ClearPass Guest 6.0

Deployment Guide

Copyright

© 2012 Aruba Networks, Inc. Aruba Networks trademarks include , Aruba Networks®, Aruba Wireless Networks®, the registered Aruba the Mobile Edge Company logo, Aruba Mobility Management System®, Mobile Edge Architecture®, People Move. Networks Must Follow®, RFProtect®, Green Island®. All rights reserved. All other trademarks are the property of their respective owners.

Open Source Code

Certain Aruba products include Open Source software code developed by third parties, including software code subject to the GNU General Public License (GPL), GNU Lesser General Public License (LGPL), or other Open Source Licenses. Includes software from Litech Systems Design. The IF-MAP client library copyright 2011 Infoblox, Inc. All rights reserved. This product includes software developed by Lars Fenneberg et al. The Open Source code used can be found at this site:

http://www.arubanetworks.com/open_source

Legal Notice

The use of Aruba Networks, Inc. switching platforms and software, by all individuals or corporations, to terminate other vendors' VPN client devices constitutes complete acceptance of liability by that individual or corporation for this action and indemnifies, in full, Aruba Networks, Inc. from any and all legal actions that might be taken against it with respect to infringement of copyright on behalf of those vendors.

Warranty

This hardware product is protected by the standard Aruba warranty of one year parts/labor. For more information, refer to the ARUBACARE SERVICE AND SUPPORT TERMS AND CONDITIONS.

Altering this device (such as painting it) voids the warranty.

www.arubanetworks.com

1344 Crossman Avenue
Sunnyvale, California 94089

Phone: 408.227.4500
Fax 408.227.4550

Contents

Chapter 1 **Preface** 15
Audience 15
Typographic Conventions 15
Contacting Support 16
..... 16

Chapter 2 **ClearPass Guest Overview** 17
About ClearPass Guest 17
Visitor Access Scenarios 18
Reference Network Diagram 19
Key Interactions 20
AAA Framework 21
Key Features 22
Visitor Management Terminology 23
ClearPass Guest Deployment Process 24
Security Policy Considerations 24
Operational Concerns 25
Network Provisioning 25
Site Preparation Checklist 25
AirGroup Deployment Process 26
Documentation and User Assistance 27
Deployment Guide and Online Help 27
Context-Sensitive Help 27
Field Help 28
Quick Help 28
If You Need More Assistance 28
Use of Cookies 29

Chapter 3 **Guest Manager** 31
Accessing Guest Manager 31
About Guest Management Processes 31
Sponsored Guest Access 32
Self Provisioned Guest Access 32
Using Standard Guest Management Features 33
Creating a Guest Account 33
Creating a Guest Account Receipt 34
Creating Multiple Guest Accounts 35
Creating Multiple Guest Account Receipts 36
Creating a Single Password for Multiple Accounts 37
Managing Guest Accounts 39
Managing Multiple Guest Accounts 43
Importing Guest Accounts 45
Exporting Guest Account Information 48
About CSV and TSV Exports 49
About XML Exports 49

MAC Authentication in ClearPass Guest 50
MAC Address Formats 50
Managing Devices 50
Changing a Device's Expiration Date 52
Disabling and Deleting Devices 53
Activating a Device 53
Editing a Device 54
Viewing Current Sessions for a Device 55
Viewing and Printing Device Details 56
MAC Creation Modes 56
Creating Devices Manually in ClearPass Guest 56
Creating Devices During Self-Registration - MAC Only 58
Creating Devices During Self-Registration - Paired Accounts 59
AirGroup Device Registration 60
Registering Groups of Devices or Services 60
Registering Personal Devices 62
Automatically Registering MAC Devices in ClearPass Policy Manager 64
Importing MAC Devices 65
Advanced MAC Features 65
2-Factor Authentication 65
MAC-Based Derivation of Role 66
User Detection on Landing Pages 66
Click-Through Login Pages 66
Active Sessions Management 67
Session States 69
RFC 3576 Dynamic Authorization 69
Filtering the List of Active Sessions 70
Disconnecting Multiple Active Sessions 71
Sending Multiple SMS Alerts 71
About SMS Guest Account Receipts 72

Chapter 4 Onboard 73

Accessing Onboard 73
About ClearPass Onboard 74
Onboard Deployment Checklist 74
Onboard Feature List 76
Supported Platforms 77
Public Key Infrastructure for Onboard 78
Certificate Hierarchy 78
Revoking Unique Device Credentials 79
Revoking Credentials to Prevent Network Access 79
Re-Provisioning a Device 79
Network Requirements for Onboard 80
Using Same SSID for Provisioning and Provisioned Networks 80
Using Different SSID for Provisioning and Provisioned Networks 80
Configuring Online Certificate Status Protocol 81
Configuring Certificate Revocation List (CRL) 81
Network Architecture for Onboard 81
Network Architecture for Onboard when Using ClearPass Guest 83
The ClearPass Onboard Process 84
Devices Supporting Over-the-Air Provisioning 84
Devices Supporting Onboard Provisioning 87
Managing Provisioned Applications 89
Configuring the User Interface for Device Provisioning 90
Customizing the Device Provisioning Web Login Page 90
Using the {nwa_mdps_config} Template Function 91

Configuring the Certificate Authority 92
Setting Up the Certificate Authority 93
Setting Up a Root Certificate Authority 94
Setting Up an Intermediate Certificate Authority 95
Obtaining a Certificate for the Certificate Authority 97
Using Microsoft Active Directory Certificate Services 97
Installing a Certificate Authority's Certificate 99
Renewing the Certificate Authority's Certificate 100
Configuring Data Retention Policy for Certificates 101
Uploading Certificates for the Certificate Authority 101
Creating a Certificate 103
Specifying the Identity of the Certificate Subject 104
Issuing the Certificate Request 106
Managing Certificates 106
Searching for Certificates in the List 107
Working with Certificates in the List 107
Working with Certificate Signing Requests 110
Importing a Code-Signing Certificate 112
Importing a Trusted Certificate 114
Requesting a Certificate 116
Providing a Certificate Signing Request in Text Format 116
Providing a Certificate Signing Request File 117
Specifying Certificate Properties 117
Configuring Provisioning Settings 118
Configuring Basic Provisioning Settings 118
Configuring Certificate Properties for Device Provisioning 118
Configuring Revocation Checks and Authorization 121
Configuring Provisioning Settings for iOS and OS X 122
Configuring Instructions for iOS and OS X 124
Configuring Reconnect Behavior for iOS and OS X 124
Configuring Provisioning Settings for Legacy OS X Devices 125
Configuring Provisioning Settings for Windows Devices 126
Configuring Provisioning Settings for Android Devices 127
Configuring Options for Legacy OS X, Windows, and Android Devices 129
Configuring Network Settings for Device Provisioning 131
Configuring Basic Network Access Settings 131
Configuring 802.1X Authentication Network Settings 134
Configuring Device Authentication Settings 135
Configuring Mutual Authentication Settings 136
Configuring Trust Settings Automatically 136
Configuring Trust Settings Manually 137
Configuring Windows-Specific Network Settings 138
Configuring Proxy Settings 139
Configuring an iOS Device VPN Connection 140
Configuring an iOS Device Email Account 141
Configuring an iOS Device Passcode Policy 143
Resetting Onboard Certificates and Configuration 145
Onboard Troubleshooting 146
iOS Device Provisioning Failures 146

Chapter 5 Configuration 147
Accessing Configuration 147
Configuring ClearPass Guest Authentication 148
Content Manager 148
Uploading Content 149
Downloading Content 149
Additional Content Actions 150
Customizing Guest Manager 151
Default Settings for Account Creation 151
About Fields, Forms, and Views 155
Business Logic for Account Creation 155
Verification Properties 155
Basic User Properties 155
Visitor Account Activation Properties 156
Visitor Account Expiration Properties 156
Other Properties 157
Standard Fields 157
Standard Forms and Views 158
Customizing Fields 159
Creating a Custom Field 160
Duplicating a Field 161
Editing a Field 161
Deleting a Field 161
Displaying Forms that Use a Field 161
Displaying Views that Use a Field 162
Customizing AirGroup Registration Forms 162
Configuring the Shared Locations and Shared Role Fields 162
Example: 165
Customizing Forms and Views 165
Editing Forms and Views 166
Duplicating Forms and Views 167
Editing Forms 167
Form Field Editor 168
Form Display Properties 168
Form Validation Properties 180
Examples of Form field Validation 181
Advanced Form Field Properties 184
Form Field Validation Processing Sequence 185
Editing Views 188
View Field Editor 189
Customizing Self Provisioned Access 190
Self-Registration Sequence Diagram 191
Creating a Self-Registration Page 192
Editing Self-Registration Pages 192
Configuring Basic Properties for Self-Registration 193
Using a Parent Page 194
Paying for Access 194
Requiring Operator Credentials 194
Editing Registration Page Properties 195
Editing the Default Self-Registration Form Settings 196
Creating a Single Password for Multiple Accounts 197
Editing Guest Receipt Page Properties 197
Editing Receipt Actions 198
Enabling Sponsor Confirmation for Role Selection 199
Editing Download and Print Actions for Guest Receipt Delivery 201

Editing Email Delivery of Guest Receipts 202
Editing SMS Delivery of Guest Receipts 203
Enabling and Editing NAS Login Properties 203
Editing Login Page Properties 204
Self-Service Portal Properties 207
Resetting Passwords with the Self-Service Portal 208
Email Receipts and SMTP Services 210
About Email Receipts 210
Configuring Email Receipts 211
Email Receipt Options 212
About Customizing SMTP Email Receipt Fields 214
Customizing Print Templates 215
Creating New Print Templates 216
Print Template Wizard 217
Modifying Wizard-Generated Templates 218
Setting Print Template Permissions 218
Customize SMS Receipt 220
SMS Receipt Fields 220
Configuring Access Code Logins 221
Customize Random Username and Passwords 221
Create the Print Template 221
Customize the Guest Accounts Form 222
Create the Access Code Guest Accounts 223

Chapter 6 Hotspot Manager 225
Accessing Hotspot Manager 225
About Hotspot Management 226
Managing the Hotspot Sign-up Interface 226
Captive Portal Integration 228
Look and Feel 228
SMS Services 228
Managing Hotspot Plans 229
Editing or Creating a Hotspot Plan 229
Managing Transaction Processors 231
Creating a New Transaction Processor 231
Managing Existing Transaction Processors 232
Managing Customer Information 232
Managing Hotspot Invoices 233
Customizing the User Interface 234
Customizing Visitor Sign-Up Page One 234
Customizing Visitor Sign-Up Page Two 236
Customizing Visitor Sign-Up Page Three 239
Viewing the Hotspot User Interface 240

Chapter 7 Administration 241
Accessing Administration 241
AirGroup Services 242
Configuring the AirGroup Services Plugin 242
Creating AirGroup Administrators 243
Creating AirGroup Operators 243
Authenticating AirGroup Users via LDAP 243
Data Retention 244
Import Configuration 244

Plugin Manager 245
Viewing Available Plugins 245
Configuring Plugins 246
Configuring the Kernel Plugin 247
Configuring the Aruba ClearPass Skin Plugin 248
SMS Services 250
Viewing SMS Gateways 250
Creating a New SMS Gateway 251
Editing an SMS Gateway 253
Sending an SMS 254
About SMS Credits 254
About SMS Guest Account Receipts 255
SMS Receipt Options 255
Working with the SMTP Carrier List 257
Support Services 260
Viewing the Application Log 260
Exporting the Application Log 262
Contacting Support 262
Viewing Documentation 262

Chapter 8 Operator Logins 265

Accessing Operator Logins 265
About Operator Logins 265
Role-Based Access Control for Multiple Operator Profiles 265
Operator Profiles 266
Creating an Operator Profile 266
Configuring the User Interface 269
Customizing Forms and Views 270
Operator Profile Privileges 271
Managing Operator Profiles 271
Configuring AirGroup Operator Device Limit 272
Local Operator Authentication 272
Creating a New Operator 272
External Operator Authentication 273
Manage LDAP Operator Authentication Servers 273
Creating an LDAP Server 274
Advanced LDAP URL Syntax 276
Viewing the LDAP Server List 276
LDAP Operator Server Troubleshooting 277
Testing Connectivity 277
Testing Operator Login Authentication 277
Looking Up Sponsor Names 278
Troubleshooting Error Messages 278
LDAP Translation Rules 279
Custom LDAP Translation Processing 281
Operator Logins Configuration 283
Custom Login Message 283
Advanced Operator Login Options 284
Automatic Logout 284

Chapter 9 Reference 285

Basic HTML Syntax 285
Standard HTML Styles 286

Smarty Template Syntax 287
Basic Template Syntax 287
Text Substitution 287
Template File Inclusion 288
Comments 288
Variable Assignment 288
Conditional Text Blocks 288
Script Blocks 288
Repeated Text Blocks 289
Foreach Text Blocks 289
Modifiers 289
Predefined Template Functions 290
dump 290
nwa_commandlink 291
nwa_iconlink 291
nwa_icontext 292
nwa_quotejs 293
nwa_radius_query 293
Advanced Developer Reference 299
nwa_assign 299
nwa_bling 300
nwa_makeid 300
nwa_nav 300
nwa_plugin 302
nwa_privilege 302
nwa_replace 302
nwa_text 303
nwa_userpref 303
nwa_youtube 303

Date/Time Format Syntax 304
nwadateformat Modifier 304
nwatimeformat Modifier 305
Date/Time Format String Reference 305

Programmer's Reference 306
NwaAlnumPassword 306
NwaBoolFormat 306
NwaByteFormat 306
NwaByteFormatBase10 306
NwaComplexPassword 307
NwaCsvCache 307
NwaDigitsPassword($len) 307
NwaDynamicLoad 307
NwaGeneratePictureString 307
NwaGenerateRandomPasswordMix 307
NwaLettersDigitsPassword 308
NwaLettersPassword 308
NwaMoneyFormat 308
NwaParseCsv 308
NwaParseXml 309
NwaPasswordByComplexity 309
NwaSmsIsValidPhoneNumber 310
NwaStrongPassword 310
NwaVLookup 310
NwaWordsPassword 311

Field, Form, and View Reference 311
GuestManager Standard Fields 311
Hotspot Standard Fields 319
SMS Services Standard Fields 320
SMTP Services Standard Fields 321
Format Picture String Symbols 322
Form Field Validation Functions 323
Form Field Conversion Functions 326
Form Field Display Formatting Functions 326
View Display Expression Technical Reference 328
LDAP Standard Attributes for User Class 330
Regular Expressions 330

Chapter 10 Glossary 333

Index 337

Figures

Figure 1 Visitor access using ClearPass Guest 18
Figure 2 Reference network diagram for visitor access 19
Figure 3 Interactions involved in guest access 20
Figure 4 Sequence diagram for network access using AAA 21
Figure 5 Sponsored guest access with guest created by operator 32
Figure 6 Guest access when guest is self-provisioned 32
Figure 7 MAC Authentication Plugin—Configuration 50
Figure 8 MAC Authentication Profile 50
Figure 9 Modify fields 59
Figure 10 Relationship of Certificates in the Onboard Public Key Infrastructure 78
Figure 11 ClearPass Onboard Network Architecture 81
Figure 12 Detailed View of the ClearPass Onboard Network Architecture 82
Figure 13 ClearPass Onboard Network Architecture when Using ClearPass Guest 83
Figure 14 ClearPass Onboard Process for iOS Devices 84
Figure 15 Sequence Diagram for the Onboard Workflow on iOS Platform 85
Figure 16 Over-the-Air Provisioning Workflow for iOS Platform 86
Figure 17 ClearPass Onboard Process for Onboard-Capable Devices 87
Figure 18 Sequence Diagram for the Onboard Workflow on Android Platform 88
Figure 19 Onboard Provisioning Workflow in the QuickConnect App 89
Figure 20 Customize Guest Manager Page (upper section) 151
Figure 21 Sample Guest Receipt Showing Aruba as the Default Site SSID 152
Figure 22 Customize Guest Manager Page, Continued (middle section) 153
Figure 23 Customize Guest Manager Page, Continued (lower section) 154
Figure 24 Steps involved in form field processing 185
Figure 25 Sequence diagram for guest self-registration 191
Figure 26 Guest Self-Registration Workflow Diagram 193
Figure 27 Customize Email Receipt page 212
Figure 28 Example of Email Receipt Test Message Content 213
Figure 29 Customize SMS Receipt page 220
Figure 30 Guest self-provisioning 226
Figure 31 The Administration Module's Left Navigation 241
Figure 32 Data Retention Policy page 244
Figure 33 Configure SMS Services Plugin 256
Figure 34 Operator profiles and visitor access control 266

Tables

Table 1 Typographic Conventions 15
Table 2 List of Key features 22
Table 3 Common Terms 23
Table 4 Site Preparation Checklist 25
Table 5 Summary of AirGroup Configuration Steps in ClearPass Guest 26
Table 6 Quick Links 27
Table 7 Operators supported in filters 40
Table 8 Operators supported in filters 44
Table 9 Operators supported in filters 51
Table 10 Tag=Value Pair Formats 61
Table 11 Operators supported in filters 70
Table 12 Onboard Deployment Checklist 74
Table 13 Onboard Features 76
Table 14 Platforms Supported by ClearPass Onboard 77
Table 15 Properties Available with the `(nwa_mdps_config)` Smarty Template Function 91
Table 16 Subject Alternative Name Fields Supported When Creating a TLS Client Certificate Signing Request 105
Table 17 Types of Certificate Supported by Onboard Certificate Management 106
Table 18 Device Information Stored in TLS Client Certificates 120
Table 19 Visitor Management Forms and Views 158
Table 20 Operators supported in filters 269
Table 21 Server Type Parameters 274
Table 22 LDAP Error Messages 278
Table 23 Template Variables 281
Table 24 Standard HTML Tags 285
Table 25 Formatting Classes 286
Table 26 Smarty Modifiers 290
Table 27 Navigation Tags 301
Table 28 Date and Time Formats 304
Table 29 Date and Time Format Strings 305
Table 30 Parsing Options 308
Table 31 NwaVLookup Options 310
Table 32 GuestManager Standard Fields 311
Table 33 Hotspot Standard Fields 319
Table 34 SMS Services Standard Fields 320
Table 35 SMPT Services Standard Fields 321
Table 36 Picture String Symbols 322
Table 37 Picture String Example Passwords 323
Table 38 Complexity Requirements 325
Table 39 Form Field Display Functions 326
Table 40 Display Expressions for Data Formatting 329
Table 41 Regular Expressions for Pattern Matching 330

ClearPass Guest provides a simple and personalized user interface through which operational staff can quickly and securely manage visitor network access.

Audience

This deployment guide is intended for system administrators and people who are installing and configuring ClearPass Guest as their visitor management solution. It describes the installation and configuration process.

Typographic Conventions

The following conventions are used throughout this manual to emphasize important concepts.

Table 1 *Typographic Conventions*

Type Style	Description
Italics	Used to emphasize important items and for the titles of books.
Boldface	Used to highlight navigation in procedures and to emphasize command names and parameter options when mentioned in text.
`Sample template code or HTML text`	Code samples are shown in a fixed-width font.
<angle brackets>	When used in examples or command syntax, text within angle brackets represents items you should replace with information appropriate to your specific situation. For example: ping <ipaddr> In this example, you would type “ping” at the system prompt exactly as shown, followed by the IP address of the system to which ICMP echo packets are to be sent. Do not type the angle brackets.
Save Changes	Tab and button names are shown in bold preceded by the appropriate icon.
	Command link icons are shown in the margin. These icons are used within the ClearPass Guest user interface to visually identify the different components of the software
	Indicates helpful suggestions, pertinent information, and important things to remember.
	Indicates a risk of damage to your hardware or loss of data.

Contacting Support

Main Site	arubanetworks.com
Support Site	support.arubanetworks.com
Airheads Social Forums and Knowledge Base	community.arubanetworks.com
North American Telephone	1-800-943-4526 (Toll Free) 1-408-754-1200
International Telephones	arubanetworks.com/support-services/aruba-support-program/contact-support/
Software Licensing Site	licensing.arubanetworks.com/login.php
End of Support information	www.arubanetworks.com/support-services/end-of-life-products/end-of-life-policy/
Wireless Security Incident Response Team (WSIRT)	arubanetworks.com/support/wsirt.php
Support Email Addresses	
Americas and APAC	support@arubanetworks.com
EMEA	emea_support@arubanetworks.com
WSIRT Email Please email details of any security problem found in an Aruba product.	wsirt@arubanetworks.com

Chapter 2

ClearPass Guest Overview

This chapter explains the terms, concepts, processes, and equipment involved in managing visitor access to a network, and helps you understand how ClearPass Guest can be successfully integrated into your network infrastructure. It is intended for network architects, IT administrators, and security consultants who are planning to deploy visitor access, or who are in the early stages of deploying a visitor access solution.

This chapter includes the following sections:

- "About ClearPass Guest"
- "Visitor Access Scenarios"
- "Reference Network Diagram"
- "Key Interactions"
- "AAA Framework"
- "Key Features"
- "Visitor Management Terminology"
- "ClearPass Guest Deployment Process"
- "AirGroup Deployment Process"
- "Documentation and User Assistance"
- "Use of Cookies"

About ClearPass Guest

ClearPass Guest provides a simple and personalized user interface through which operational staff can quickly and securely manage visitor network access. It gives your non-technical staff controlled access to a dedicated visitor management user database. Through a customizable Web portal, your staff can easily create an account, reset a password, or set an expiry time for visitors. Access permissions to ClearPass Guest functions are controlled through an operator profile that can be integrated with an LDAP server or Active Directory login.

Visitors can be registered at reception and provisioned with an individual guest account that defines their visitor profile and the duration of their visit. The visitor can be given a printed customized receipt with account details, or the receipt can be delivered wirelessly using the integrated SMS services. Companies are also able to pre-generate custom scratch cards, each with a defined network access time, which can then be handed out in a corporate environment or sold in public access scenarios.

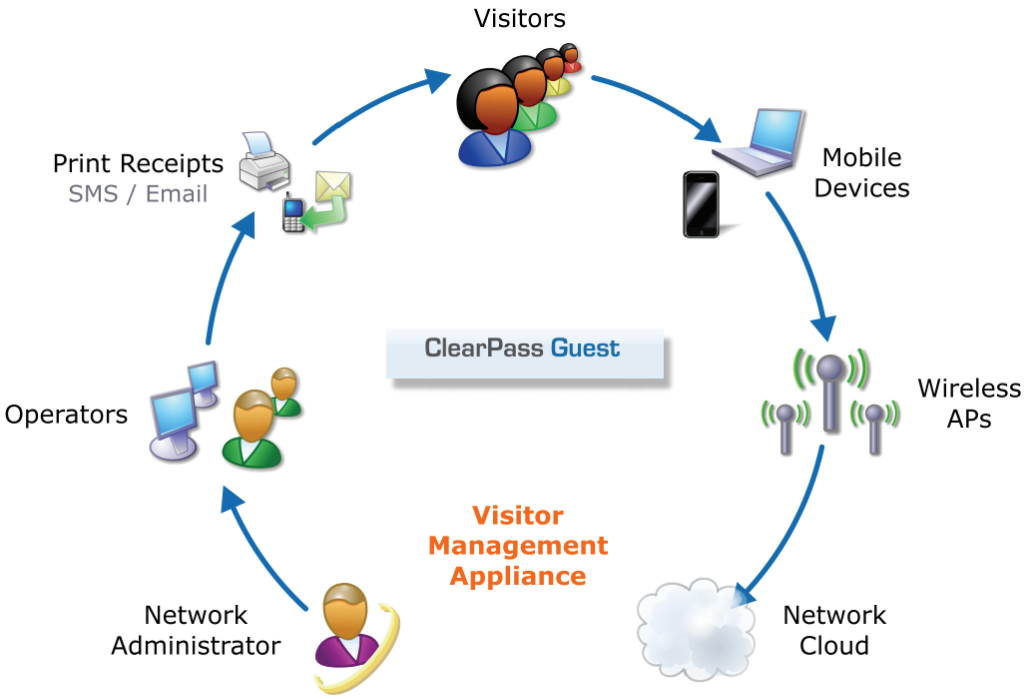
You can use the customization features to define settings that allow your visitors to self-provision their own guest accounts. Visitors register through a branded and customized Web portal, ensuring a streamlined and professional experience. Surveys can also be presented during the self-registration process and the data stored for later analysis and reporting, providing additional insight to your visitors and their network usage.

ClearPass Guest integrates with all leading wireless and NAC solutions through a flexible definition point, ClearPass Policy Manager. This ensures that IT administrators have a standard integration with the network security framework, but gives operational staff the user interface they require.

Visitor Access Scenarios

The following figure shows a high-level representation of a typical visitor access scenario.

Figure 1 *Visitor access using ClearPass Guest*



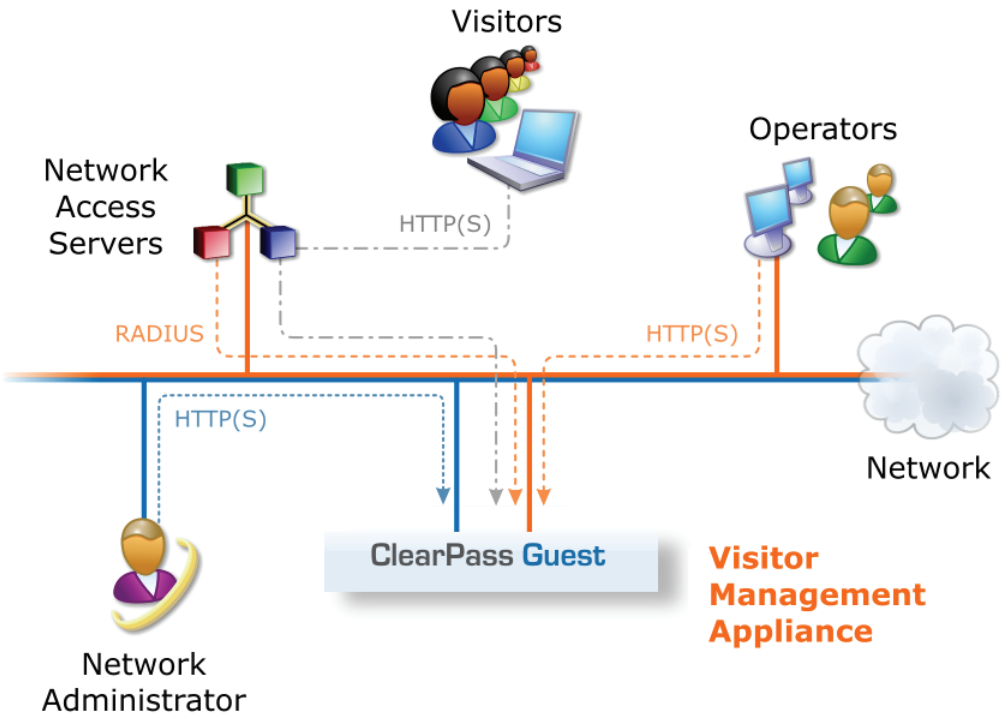
In this scenario, visitors are using their own mobile devices to access a corporate wireless network. Because access to the network is restricted, visitors must first obtain a username and password. A guest account may be provisioned by a corporate operator such as a receptionist, who can then give the visitor a print receipt that shows their username and password for the network.

When visitors use self-registration, as might be the case for a network offering public access, the process is broadly similar but does not require a corporate operator to create the guest account. The username and password for a self-provisioned guest account may be delivered directly to the visitor's Web browser, or sent via SMS or email.

Reference Network Diagram

The following figure shows the network connections and protocols used by ClearPass Guest.

Figure 2 *Reference network diagram for visitor access*

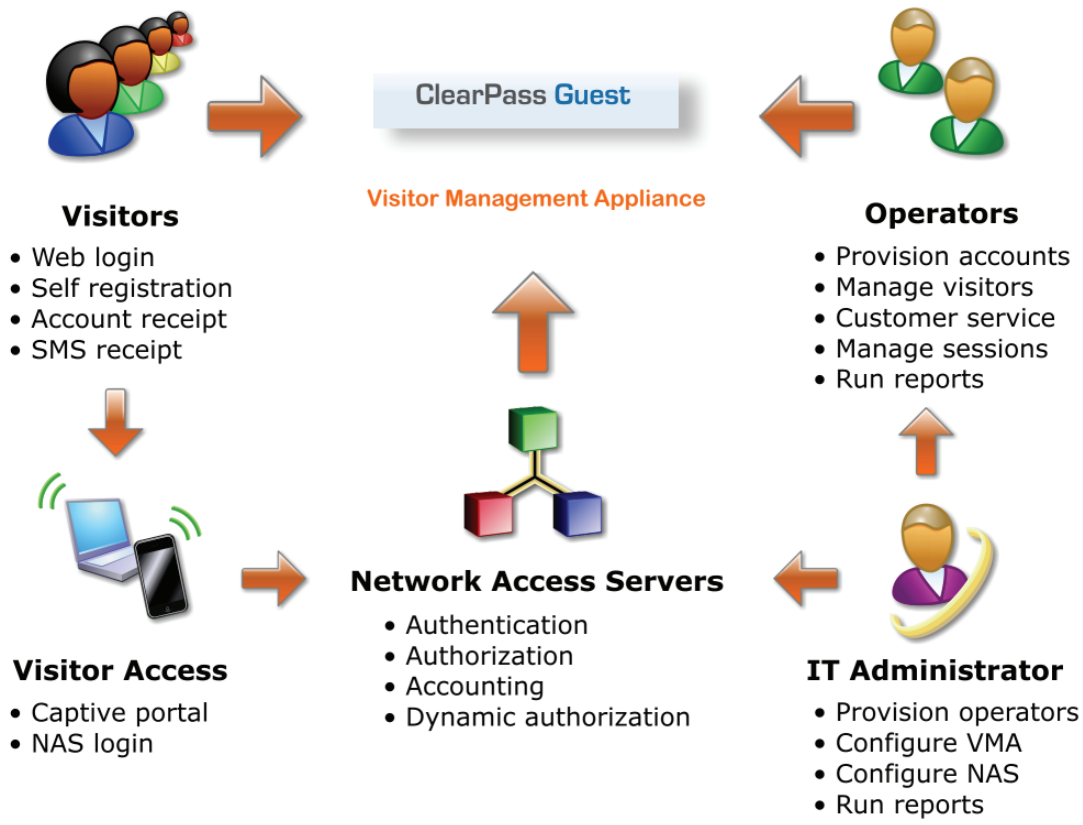


The network administrator, operators, and visitors may use different network interfaces to access the visitor management features. The exact topology of the network and the connections made to it will depend on the type of network access offered to visitors and the geographical layout of the access points.

Key Interactions

The following figure shows the key interactions between ClearPass Guest and the people and other components involved in providing guest access.

Figure 3 *Interactions involved in guest access*



ClearPass Guest is part of your network's core infrastructure and manages guest access to the network.

NAS devices, such as wireless access points and wired switches on the edge of the network, use the RADIUS protocol to ask ClearPass Policy Manager to authenticate the username and password provided by a guest logging in to the network. If authentication is successful, the guest is then authorized to access the network.

Roles are assigned to a guest as part of the context ClearPass Policy Manager uses to apply its policies. RADIUS attributes that define a role's access permissions are contained within Policy Manager's Enforcement Profile. Additional features such as role mapping for ClearPass Guest can be performed in ClearPass Policy Manager.

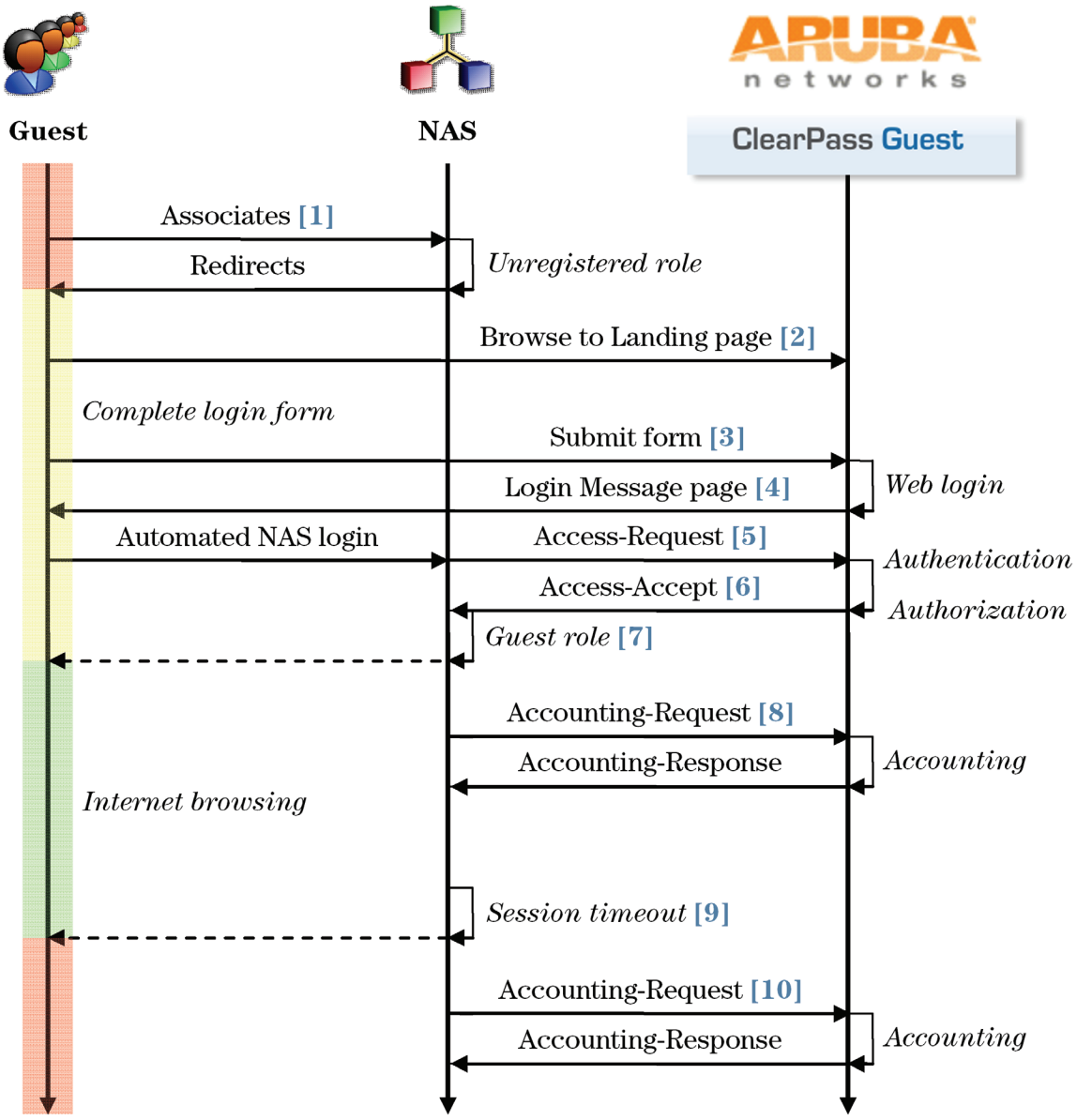
The network usage of authorized guests is monitored by the NAS and reported in summary form to ClearPass Policy Manager using RADIUS accounting, which allows administrators to generate network reports in ClearPass Insight.

AAA Framework

ClearPass Guest is built on the industry standard AAA framework, which consists of authentication, authorization, and accounting components.

The following figure shows how the different components of this framework are employed in a guest access scenario.

Figure 4 *Sequence diagram for network access using AAA*



In the standard AAA framework, network access is provided to a user according to the following process:

- The user connects to the network by associating with a local access point [1].
- A landing page is displayed to the user [2] which allows them to log in to the NAS [3], [4] using the login name and password of their guest account.
- The NAS authenticates the user with the RADIUS protocol [5].
- ClearPass Policy Manager determines whether the user is authorized, and, if so, returns vendor-specific attributes [6] that are used to configure the NAS based on the user's role and other policies [7].

- If the user's access is granted, the NAS permits the guest access to the network based on the settings provided by the ClearPass Policy Manager server.
- The NAS reports details about the user's session to the ClearPass Policy Manager server using RADIUS accounting messages [8].
- After the user's session times out [9], the NAS will return the user to an unauthorized state and finalize the details of the user's session with an accounting update [10].

Key Features

Refer to the table below for a list of key features and a cross-reference to the relevant section of this deployment guide.

Table 2 *List of Key features*

Feature	Refer to…
Visitor Access	
Web server providing content delivery for guests	"Content Manager"
Guest self-registration	"Customizing Self Provisioned Access"
Visitor Management	
Create and manage visitor accounts, individually or in groups	"Using Standard Guest Management Features"
Manage active RADIUS sessions using RFC 3576 dynamic authorization support	"Active Sessions Management"
Import and export visitor accounts	"Importing Guest Accounts"
Create guest self-registration forms	"Creating a Self-Registration Page"
Configure a self-service portal for guests	"Self-Service Portal Properties"
Local printer, SMS or email delivery of account receipts	"Editing Guest Receipt Page Properties"
Visitor Account Features	
Independent activation time, expiration time, and maximum usage time	"Business Logic for Account Creation"
Define unlimited custom fields	"Customizing Fields"
Username up to 64 characters	"GuestManager Standard Fields"

Table 2 *List of Key features (Continued)*

Customization Features	
Create new fields and forms for visitor management	"Customizing Forms and Views"
Use built-in data validation to implement visitor survey forms	"Form Validation Properties"
Create print templates for visitor account receipts	"Editing Guest Receipt Page Properties"
Administrative Management Features	
Operators defined and authenticated locally	"Local Operator Authentication"
Operators authenticated via LDAP	"External Operator Authentication"
Role based access control for operators	"Operator Profiles"
Plugin-based application features, automatically updated by ClearPass Policy Manager	"Plugin Manager"
User Interface Features	
Context-sensitive help with searchable online documentation	Documentation and User Assistance

Visitor Management Terminology

The following table describes the common terms used in ClearPass Guest and this guide.

Table 3 *Common Terms*

Term	Explanation
Accounting	Process of recording summary information about network access by users and devices.
Authentication	Verification of a user's credentials; typically a username and password.
Authorization	Controls the type of access that an authenticated user is permitted to have.
Captive Portal	Implemented by a Network Access Server to restrict network access to authorized users only.
Field	In a user interface or database, a single item of information about a user account.
Form	In a user interface, a collection of editable fields displayed to an operator.
Network Access Server	Device that provides network access to users, such as a wireless access point, network switch, or dial-in terminal server. When a user connects to the NAS device, a RADIUS access request is generated by the NAS.
Operator Profile	Characteristics assigned to a class of operators, such as the permissions granted to those operators.

Table 3 *Common Terms (Continued)*

Operator/Operator Login	User of ClearPass Guest to create guest accounts or perform system configuration.
Print Template	Formatted template used to generate guest account receipts.
Role	Type of access being granted to visitors. You can define multiple roles. Such roles could include employee, guest, team member, or press.
Sponsor	Operator
User Database	Database listing the guest accounts in ClearPass Guest.
View	In a user interface, a table displaying data, such as visitor account information, to operators.
Visitor/Guest	Someone who is permitted to access the Internet through your Network Access Server.
Visitor Account	Settings for a visitor stored in the user database, including username, password and other fields.
Web Login/NAS Login	Login page displayed to a guest user.

ClearPass Guest Deployment Process

As part of your preparations for deploying a visitor management solution, you should consider the following areas:

- Management decisions about security policy
- Decisions about the day-to-day operation of visitor management
- Technical decisions related to network provisioning

Security Policy Considerations

To ensure that your network remains secure, decisions have to be made regarding guest access:

- Do you wish to segregate guest access? Do you want a different VLAN, or different physical network infrastructure to be used by your guests?
- What resources are you going to make available to guests (for example, type of network access; permitted times of day; bandwidth allocation)?
- Will guest access be separated into different roles? If so, what roles are needed?
- How will you prioritize traffic on the network to differentiate quality of service for guest accounts and non-guest accounts?
- What will be the password format for guest accounts? Will you be changing this format on a regular basis?
- What requirements will you place on the shared secret, between NAS and the RADIUS server to ensure network security is not compromised?
- What IP address ranges will operators be using to access the server?
- Should HTTPS be required in order to access the visitor management server?

Operational Concerns

When deploying a visitor management solution, you should consider these operational concerns:

- Who is going to be responsible for managing guest accounts? What privileges will the guest account manager have? Will this person only create guest accounts or will this person also be permitted access to reports?
- Do you want guests to be able to self-provision their own network access? What settings should be applied to self-provisioned visitor accounts?
- How will operator logins be provisioned? Should operators be authenticated against an LDAP server?
- Who will manage reporting of guest access? What are the reports of interest? Are any custom reports needed?

Network Provisioning

Deploying ClearPass Guest requires provisioning the following:

- Physical location – rack space, power and cooling requirements; or deployment using virtualization
- Network connectivity – VLAN selection, IP address, and hostname
- Security infrastructure – SSL certificate

Site Preparation Checklist

The following is a checklist of the items that should be considered when setting up ClearPass Guest.

Table 4 *Site Preparation Checklist*

✓	Policy Decision
Security Policy	
	Segregated guest accounts?
	Type of network access?
	Time of day access?
	Bandwidth allocation to guests?
	Prioritization of traffic?
	Different guest roles?
	IP address ranges for operators?
	Enforce access via HTTPS?
Operational Concerns	
	Who will manage guest accounts?
	Guest account self provisioning?
	What privileges will the guest managers have?
	Who will be responsible for printing reports?

Table 4 *Site Preparation Checklist (Continued)*

Network Management Policy	
	Password format for guest accounts?
	Shared secret format?
	Operator provisioning?
Network Provisioning	
	Physical location?
	Network connectivity?
	Security infrastructure?

AirGroup Deployment Process

AirGroup allows users to register their personal mobile devices on the local network and define a group of friends or associates who are allowed to share them. You use ClearPass Guest to define AirGroup administrators and operators. AirGroup administrators can then use ClearPass Guest to register and manage an organization's shared devices and configure access according to username, role, or location. AirGroup operators (end users) can use ClearPass Guest to register their personal devices and define the group who can share them.

Table 5 summarizes the steps for configuring Aruba AirGroup functionality in ClearPass Guest. Details for these steps are provided in the relevant sections of this Guide. This table does not include the configuration steps performed in ClearPass Policy Manager or the ArubaOS controller. For complete AirGroup deployment information, refer to the AirGroup Deployment Guide and the ClearPass Policy Manager documentation.

Table 5 *Summary of AirGroup Configuration Steps in ClearPass Guest*

Step	Section in this Guide
Create AirGroup administrators	"Creating a New Operator"
Create AirGroup operators	"Creating a New Operator"
Configure an operator's device limit	"Configuring AirGroup Operator Device Limit"
To authenticate AirGroup users via LDAP: • Define the LDAP server • Define appropriate translation rules	"External Operator Authentication" "LDAP Translation Rules"
AirGroup administrator: Register devices or groups of devices	"Registering Groups of Devices or Services"
AirGroup operator: Register personal devices	"Registering Personal Devices"
(Optional) Configure device registration form with drop-down lists for existing locations and roles	"Customizing AirGroup Registration Forms"

Documentation and User Assistance

This section describes the variety of user assistance available for ClearPass Guest.

Deployment Guide and Online Help

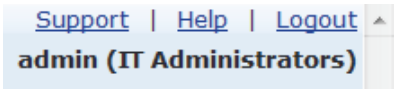
This Deployment Guide provides complete information for all ClearPass Guest features. The following quick links may be useful in getting started.

Table 6 *Quick Links*

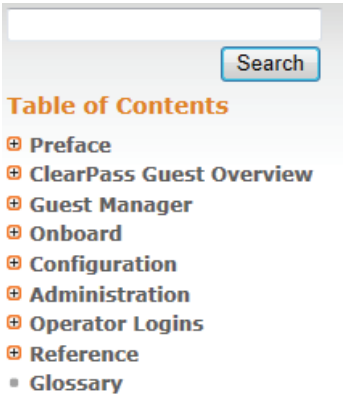
For information about...	Refer to...
What visitor management is and how it works	"About ClearPass Guest"
Using the guest management features	"Using Standard Guest Management Features"
Role-based access control for operators	"Operator Profiles"
Setting up LDAP authentication for operators	"External Operator Authentication"
Guest self-provisioning features	"Self Provisioned Guest Access"
Dynamic authorization extensions	"RFC 3576 Dynamic Authorization"
SMS receipts for guest accounts	"SMS Services"
Email receipts for guest accounts	"Email Receipts and SMTP Services"
Network administration of the appliance	"Administration"

Context-Sensitive Help

For more detailed information about the area of the application you are using, click the context-sensitive **Help** link displayed at the top right of the page. This opens a new browser tab showing the relevant section of this deployment guide.



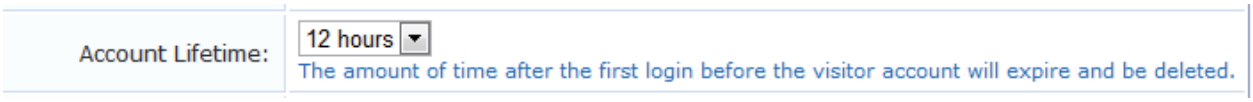
The deployment guide may be searched using the Search box in the top left corner.



Type in keywords related to your search and click the **Search** button to display a list of matches. The most relevant matches will be displayed first. Words may be excluded from the search by typing a minus sign directly before the word to exclude (for example-exclude). Exact phrase matches may also be searched for by enclosing the phrase in double quotes (for example, “word phrase”).

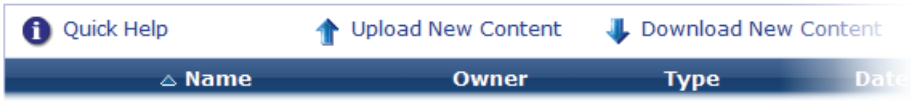
Field Help

The ClearPass Guest user interface has field help built into every form. The field help provides a short summary of the purpose of the field at the point you need it most. In many cases this is sufficient to use the application without further assistance or training.

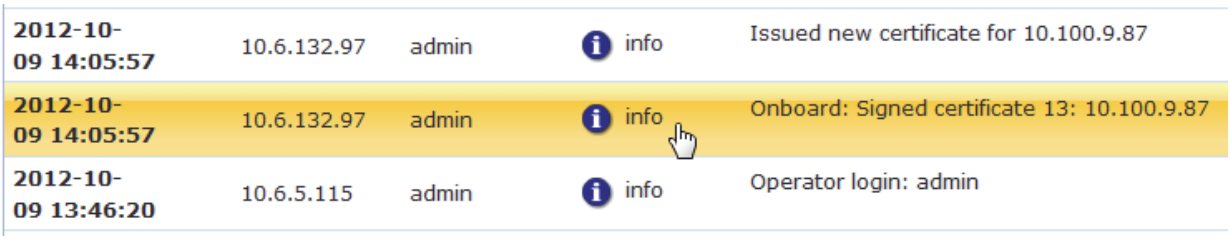


Quick Help

In list views, click the **Quick Help** tab located at the top left of the list to display additional information about the list you are viewing and the actions that are available within the list.



On some forms and views, the Quick Help icon may also be used to provide additional detail about a field.



If You Need More Assistance

If you encounter a problem using ClearPass Guest, your first step should be to consult the appropriate section in this Deployment Guide.

If you cannot find an answer here, the next step is to contact your reseller. The reseller can usually provide you with the answer or obtain a solution to your problem.

If you still need information, you can refer to the **Contact Support** command available under **Support Services** in the user interface, or see “Contacting Support” in the “Preface” chapter.

Use of Cookies

Cookies are small text files that are placed on a user's computer by Web sites the user visits. They are widely used in order to make Web sites work, or work more efficiently, as well as to provide information to the owners of a site. Session cookies are temporary cookies that last only for the duration of one user session.

When a user registers or logs in via an Aruba captive portal, Aruba uses session cookies solely to remember between clicks who a guest or operator is. Aruba uses this information in a way that does not identify any user-specific information, and does not make any attempt to find out the identities of those using its ClearPass products. Aruba does not associate any data gathered by the cookie with any personally identifiable information (PII) from any source. Aruba uses session cookies only during the user's active session and does not store any permanent cookies on a user's computer. Session cookies are deleted when the user closes his/her Web browser.

Chapter 3

Guest Manager

The ability to easily create and manage guest accounts is the primary function of ClearPass Guest. The Guest Manager module provides complete control over the user account creation process.

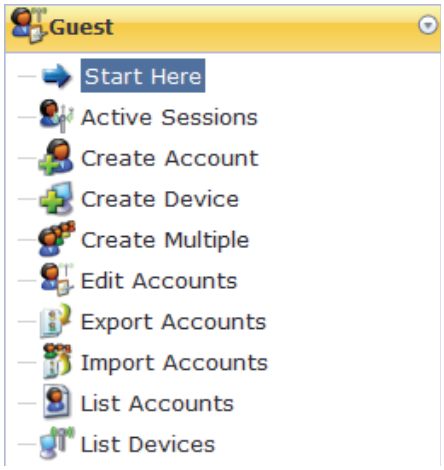
Guest Manager features for managing guest accounts let you:

- Create single or multiple guest accounts and receipts
- List guest accounts and edit individual or multiple accounts
- View and manage active sessions
- Import new accounts from a text file
- Export a list of accounts
- View MAC devices
- Create new MAC devices

Many features can also be customized. For information on customizing Guest Manager settings, forms and views, guest self-registration, and print templates, see the "Configuration" chapter.

Accessing Guest Manager

To access ClearPass Guest's guest management features, click the **Guest** link in the left navigation.



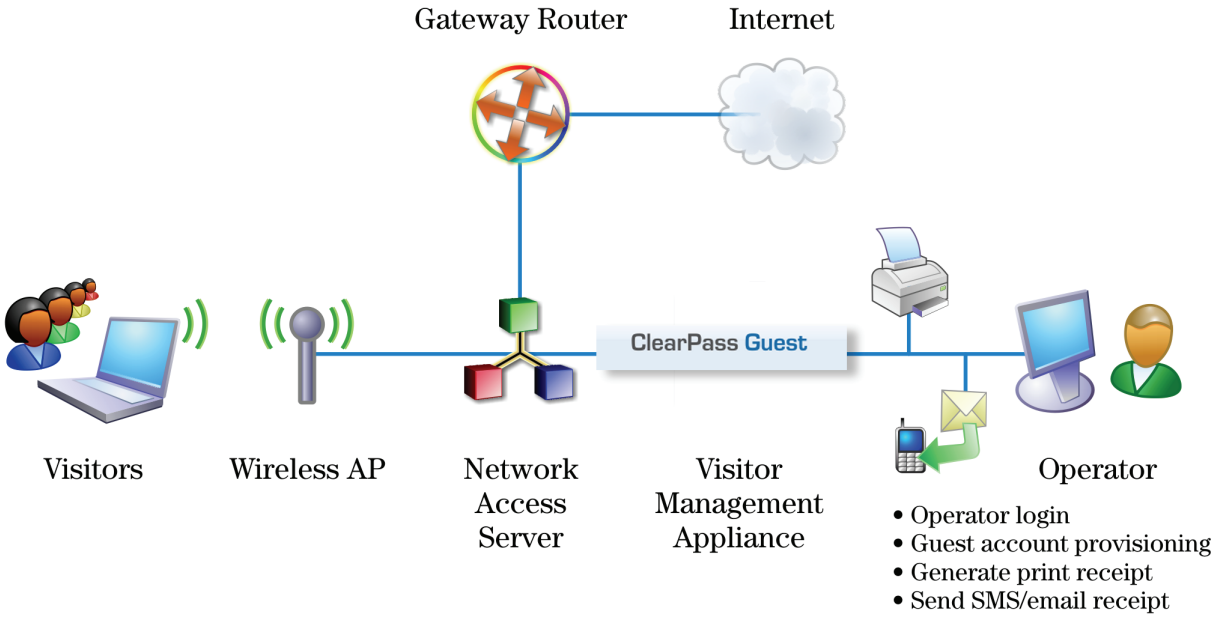
About Guest Management Processes

There are two major ways to manage guest access – either by your operators provisioning guest accounts, or by the guests self-provisioning their own accounts. Both of these processes are described in the next sections.

Sponsored Guest Access

The following figure shows the process of sponsored guest access.

Figure 5 Sponsored guest access with guest created by operator



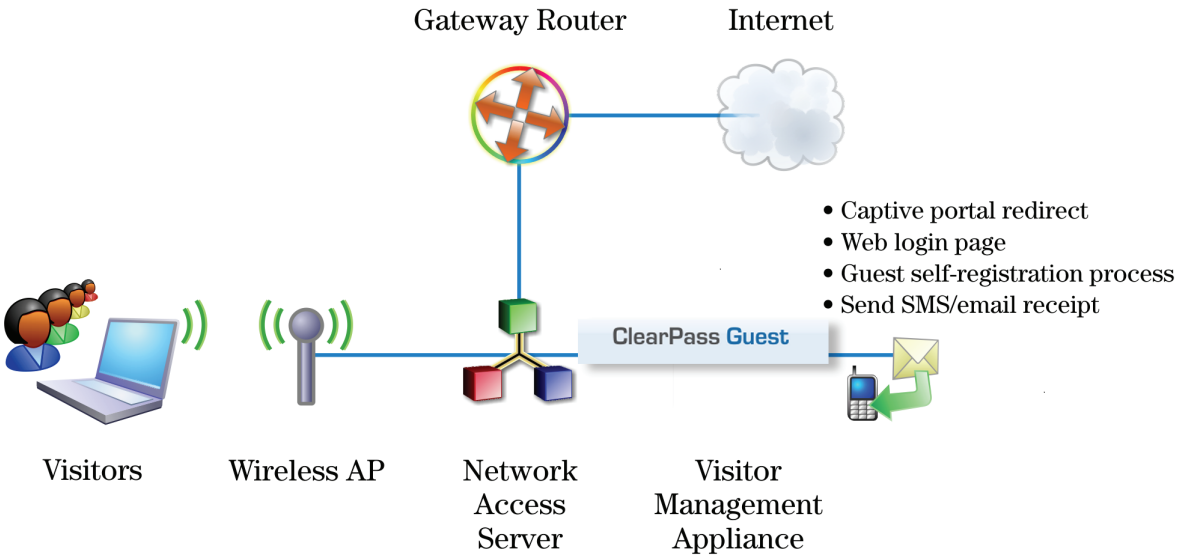
The operator creates the guest accounts and generates a receipt for the account.

The guest logs on to the Network Access Server (NAS) using the credentials provided on her receipt. The NAS authenticates and authorizes the guest's login in ClearPass Guest. Once authorized, the guest is able to access the network.

Self Provisioned Guest Access

Self-provisioned access is similar to sponsored guest access, but there is no need for an operator to create the account or to print the receipt. The following figure shows the process of self-provisioned guest access.

Figure 6 Guest access when guest is self-provisioned



The guest logs on to the Network Access Server (NAS), which captures the guest and redirects them to a captive portal login page. From the login page, guests without an account can browse to the guest self-

registration page, where the guest creates a new account. At the conclusion of the registration process, the guest is automatically redirected to the NAS to log in.

The guest can print or download a receipt, or have the receipt information sent to her by SMS or email.

The NAS performs authentication and authorization for the guest in ClearPass Guest. Once authorized, the guest is then able to access the network.

See **"Customizing Self Provisioned Access"** in the "Configuration" chapter for details on creating and managing self-registration pages.

Using Standard Guest Management Features

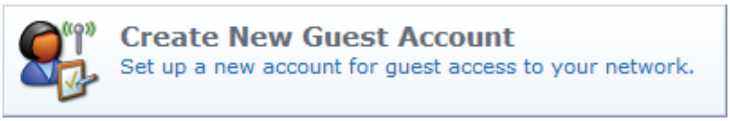
This section describes:

- How to create a single guest account and a guest account receipt
- How to create multiple guest accounts and multiple guest account receipts
- How to create a single password for multiple accounts
- How to list and edit single and multiple guest accounts

To customize guest self-registration, please see the "Configuration" chapter.

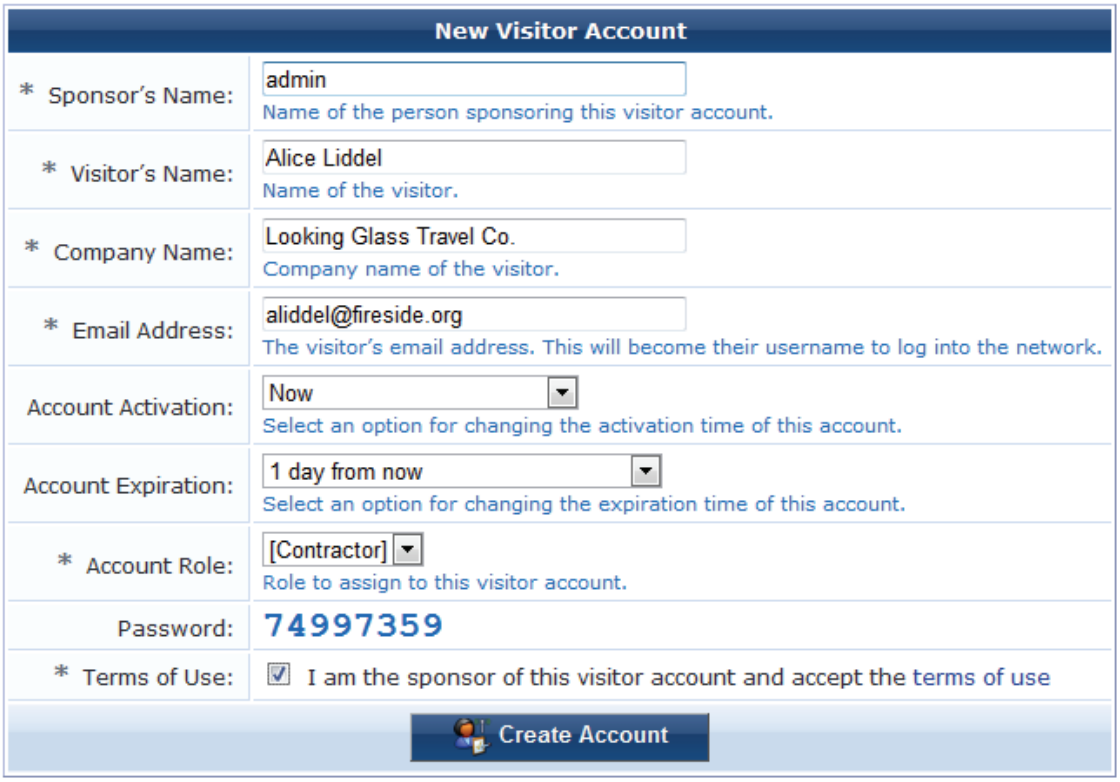
Creating a Guest Account

To create a new account, go to **Guest > Create Account**, or click the **Create New Guest Account** command link on the Guest Manager page. The New Visitor Account form opens.



NOTE

The New Visitor Account form (create_user) may be customized by adding new fields, or modifying or removing the existing fields. See "Customizing Self Provisioned Access" in the "Configuration"chapter for details about the customization process. The default settings for this form are described below.



To complete the form, first enter the visitor's details into the **Sponsor's Name**, **Visitor Name**, **Company Name** and **Email Address** fields. The visitor's email address will become their username to log into the network.

You can specify the account activation and expiration times. The visitor account cannot be used before the activation time, or after the expiration time.

The Account Role specifies what type of account the visitor should have.

A random password is created for each visitor account. This is displayed on this form, but will also be available on the guest account receipt.

You must mark the Terms of Use check box in order to create the visitor account.

Click the **Create Account** button after completing the form.

Creating a Guest Account Receipt

After you click the Create Account button on the New Visitor Account form, the details for that account are displayed.

To print a receipt for the visitor, select an appropriate template from the **Open print window using template...** list. A new Web browser window will open and the browser's Print dialog box will be displayed.

Click the **Send SMS receipt** link to send a guest account receipt via text message. Use the **SMS Receipt** form to enter the mobile telephone number to which the receipt should be sent.

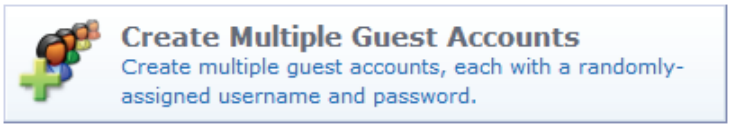
Sending SMS receipts requires the SMS Services plugin. If the administrator has enabled automatic SMS, and the visitor's phone number was typed into the **New Visitor Account** form, an SMS message will be sent automatically. A message is displayed on the account receipt page after an SMS message has been sent.

Click the **Send email receipt** link to send an email copy of the guest account receipt. Use the Email Receipt form to enter the email address to which the receipt should be sent. You can also specify the subject line for the email message. If the administrator has enabled automatic email for guest account receipts, and the visitor's email address was typed into the **New Visitor Account** form, an email receipt will be sent automatically. A message is displayed on the account receipt page after an email has been sent.

Creating Multiple Guest Accounts

The **Create Guest Accounts** form is used to create a group of visitor accounts.

To create multiple accounts, go to **Guest > Create Multiple**, or click the **Create Multiple Guest Accounts** command link on the Guest Manager page. The Create Guest Accounts form opens.



NOTE

The Create Guest Accounts form (create_multi) may be customized by adding new fields, or modifying or removing the existing fields. See "Customizing Self Provisioned Access" in the "Configuration"chapter for details about the customization process. The default settings for this form are described below.

To complete the form, you must enter the number of visitor accounts you want to create.

A random username and password will be created for each visitor account. This is not displayed on this form, but will be available on the guest account receipt.

The visitor accounts cannot be used before the activation time, or after the expiration time.

The Account Role specifies what type of accounts to create.

Click the **Create Accounts** button after completing the form.

Creating Multiple Guest Account Receipts

Once a group of guest accounts has been created, the details for the accounts are displayed.

Account Details	
Username	**91972747**
Password	20626907
Role	[Contractor]
Current State	Active
Account Activation	Friday, 26 October 2012, 03:50 PM
Account Expiration	Saturday, 27 October 2012, 03:50 PM

Account Details	
Username	**09609879**
Password	97625198
Role	[Contractor]
Current State	Active
Account Activation	Friday, 26 October 2012, 03:50 PM
Account Expiration	Saturday, 27 October 2012, 03:50 PM

Account Details	
Username	**41915905**
Password	97695485
Role	[Contractor]
Current State	Active
Account Activation	Friday, 26 October 2012, 03:50 PM
Account Expiration	Saturday, 27 October 2012, 03:50 PM

To print the receipts, select an appropriate template from the **Open print window using template...** drop-down list. A new browser window opens with the **Print** dialog displayed.

To download a copy of the receipt information in CSV format, click the **Save list for scratch cards (CSV file)** link. You will be prompted to either open or save the spreadsheet (CSV) file. The fields available in the CSV file are:

- **Number** – the sequential number of the visitor account, starting at one
- **Username** – the username for the visitor account
- **Password** – the password for the visitor account
- **Role** – the visitor account's role
- **Activation Time** – the date and time at which the account will be activated, or N/A if there is no activation time
- **Expiration Time** – the date and time at which the account will expire, or N/A if there is no activation time
- **Lifetime** – the account lifetime in minutes, or N/A if the account does not have a lifetime specified
- **Successful** – "Yes" if the account was created successfully, or "No" if there was an error creating the account

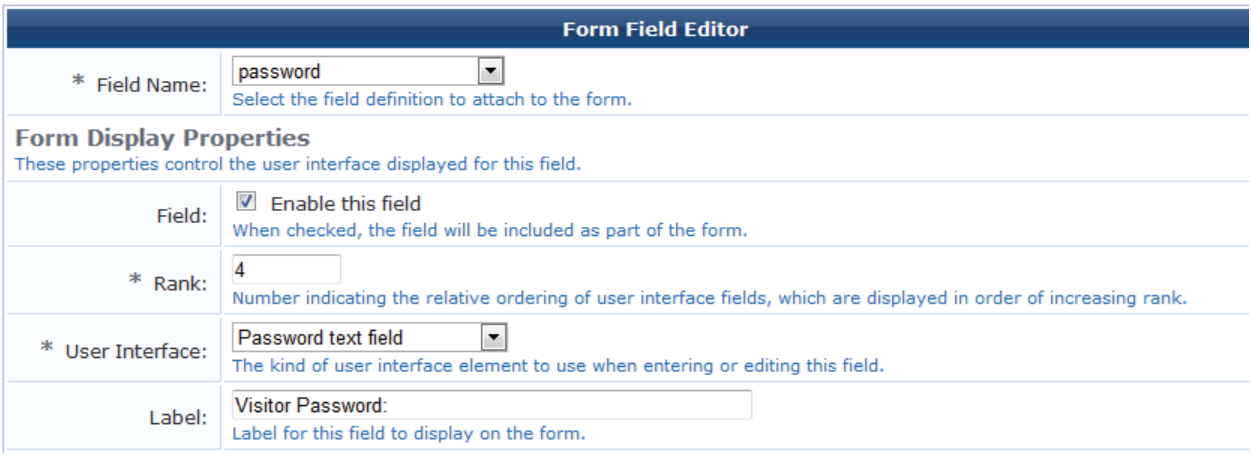
Creating a Single Password for Multiple Accounts

You can create multiple accounts that have the same password. In order to do this, you first customize the Create Multiple Guest Accounts form to include the Password field.

To include the Password field on the Create Multiple Guest Accounts form:

1. Go to **Configuration > Forms & Views**. Click the **create_multi** row, then click its **Edit Fields** link. The Customize Form Fields view opens, showing a list of the fields included in the Create Multiple Guest Accounts form and their descriptions.

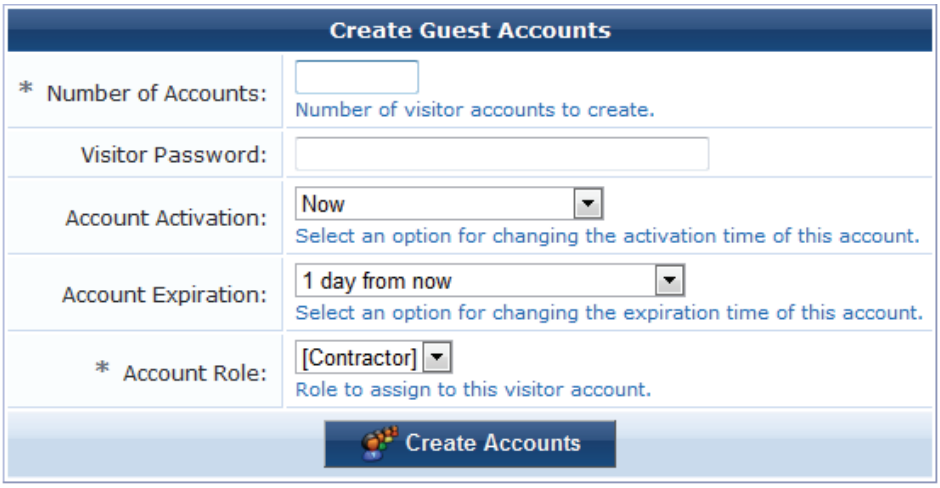
   At this point, the Password field is not listed because the Create Multiple Guest Accounts form (create_multi) has not yet been customized to include it. You will create it for the form in the next step.
2. Click on any field in the list to expand a row, then click the **Insert After** link (you can modify this placement later). The Customize Form Field form opens.
3. In the **Field Name** row, choose **password** from the drop-down list. The form displays configuration options for this field.



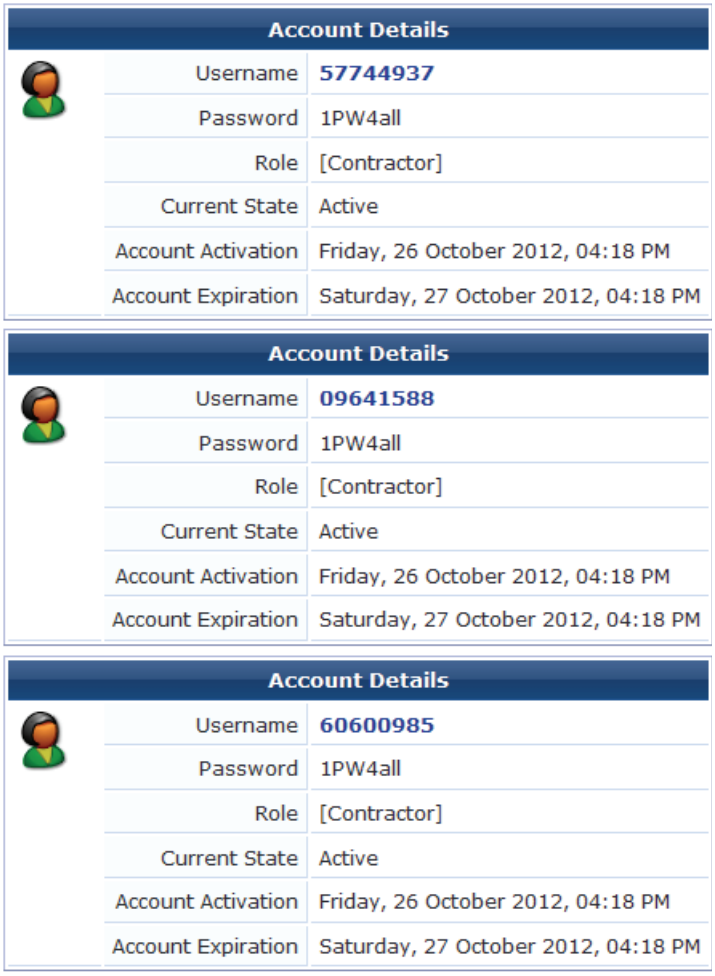
4. In the **Field** row, mark the **Enable this field** check box.
5. To adjust the placement of the password field on the Create Multiple Guest Accounts form, you may change the number in the **Rank** field.
6. In the **User Interface** row, choose **Password text field** from the drop-down list. The **Field Required** check box should now be automatically marked, and the **Validator** field should be set to **IsNonEmpty**.
7. Click **Save Changes**. The Customize Form Fields view opens again, and the password field is now included and can be edited.

To create multiple accounts that all use the same password:

1. Go to **Guest > Create Multiple**. The Create Guest Accounts form opens, and includes the Visitor Password field.

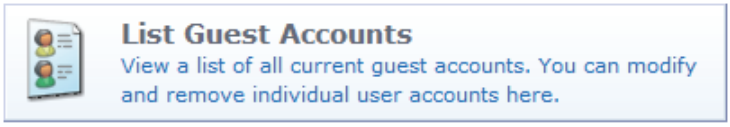


2. In the **Number of Accounts** field, enter the number of accounts you wish to create.
3. In the **Visitor Password** field, enter the password that is to be used by all the accounts.
4. Complete the other fields with the appropriate information, then click **Create Accounts**. The Finished Creating Guest Accounts view opens. The password and other account details are displayed for each account.



Managing Guest Accounts

Use the Guest Manager Accounts list view to work with individual guest accounts. To open the Guest Manager Accounts list, go to **Guest > List Accounts**.



The Guests Manager Accounts view opens.This view (guest_users) may be customized by adding new fields or modifying or removing the existing fields. See "Customizing Fields" in the "Configuration"chapter for details about this customization process. The default settings for this view are described below.

Username	Role	State	Activation	Expiration
09609879	[Contractor]	Active	40 minutes ago	2012-10-27 15:50
09641588	[Contractor]	Active	12 minutes ago	2012-10-27 16:18
41915905	[Contractor]	Active	40 minutes ago	2012-10-27 15:50
57744937	[Contractor]	Active	12 minutes ago	2012-10-27 16:18
60600985	[Contractor]	Active	12 minutes ago	2012-10-27 16:18
91972747	[Contractor]	Active	40 minutes ago	2012-10-27 15:50
ipod	[Contractor]	Expired	1.1 days ago	**Expired**
sham@a	[Contractor]	Expired	1.2 days ago	**Expired**
tom@a	[Guest]	Expired	N/A	**Expired**

Refresh 1 Showing 1 – 9 of 9 20 rows per page

The **Username**, **Role**, **State**, **Activation**, and **Expiration** columns display information about the visitor accounts that have been created:

- The value in the **Expiration** column is colored red if the account will expire within the next 24 hours. The expiration time is additionally highlighted in **boldface** if the account will expire within the next hour.
- In addition, icons in the **Username** column indicate the account's activation status:
  - —Visitor account is active
  - —Visitor account was created but is not activated yet
  - —Visitor account was disabled by Administrator
  - —Visitor account has expired
  - —Visitor account was deleted

You can use the **Filter** field to narrow the search parameters. You may enter a simple substring to match a portion of the username or any other fields that are configured for search, and you can include the following operators:

Table 7 *Operators supported in filters*

Operator	Meaning	Additional Information
=	is equal to	You may search for multiple values when using the equality (=) or inequality !=) operators. To specify multiple values, list them separated by the pipe character (|). For example, specifying the filter "role_id=2|3, custom_field=Value" restricts the accounts displayed to those with role IDs 2 and 3 (Guest and Employee), and with the field named "custom_field" set to "Value".
!=	is not equal to	
>	is greater than	
>=	is greater than or equal to	
<	is less than	
<=	is less than or equal to	
~	matches the regular expression	
!~	does not match the regular expression	

To restore the default view, click the **Clear Filter** link.

Use the paging control at the bottom of the list to jump forwards or backwards by one page, or to the first or last page of the list. You can also click an individual page number to jump directly to that page.

NOTE

When the list contains numerous user accounts, consider using the Filter field to speed up finding a specific user account.

Use the **Create** tab to create new visitor accounts using the **New Visitor Account** form. See "Creating a Guest Account" for details about this form.

Use the **More Options** tab for additional functions, including import and export of guest accounts and the ability to customize the view.

Click a user account's row to select it. You can then select from one of these actions:

- **Reset password** – Changes the password for a guest account. A new randomly generated password is displayed on the **Reset Password** form.

Click **Update Account** to reset the guest account's password. A new account receipt is displayed, allowing you to print a receipt showing the updated account details.

- **Change expiration** – Changes the expiration time for a guest account.

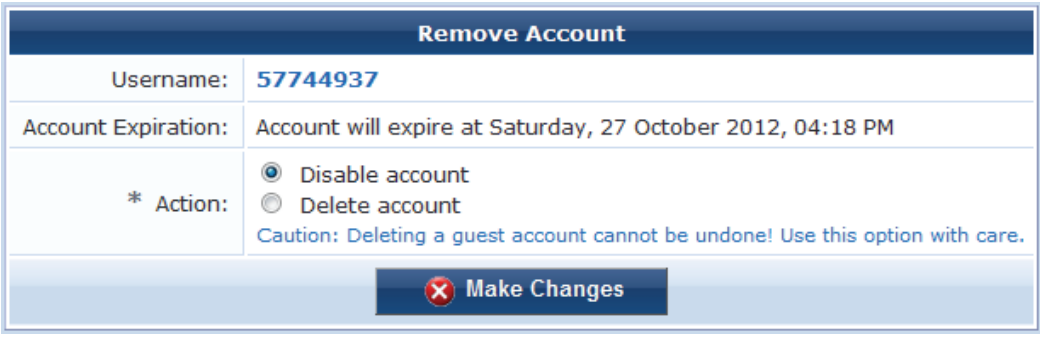
NOTE

This form (change_expiration) may be customized by adding new fields, or modifying or removing the existing fields. Refer to the "Customizing Forms and Views" section of the "Configuration" chapter for details about this customization process.

Select an option from the drop-down list to change the expiration time of the guest account.

Click **Update Account** to set the new expiration time for the guest account. A new account receipt is displayed, allowing you to print a receipt showing the updated account details.

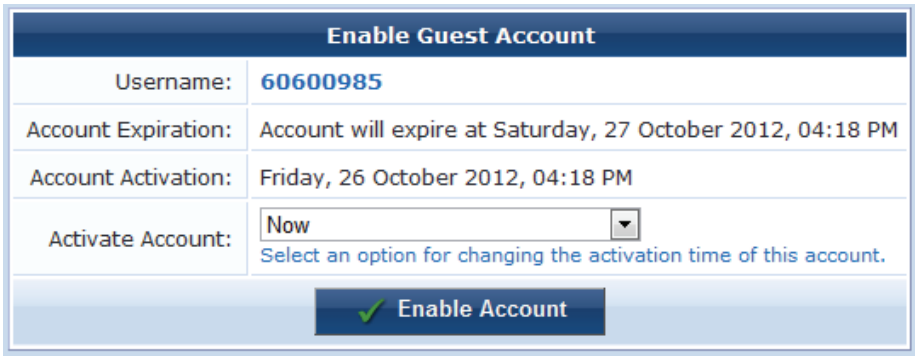
- **Remove** – Disables or deletes a guest account.



Select the appropriate **Action** radio button, and click **Make Changes** to disable or delete the account.

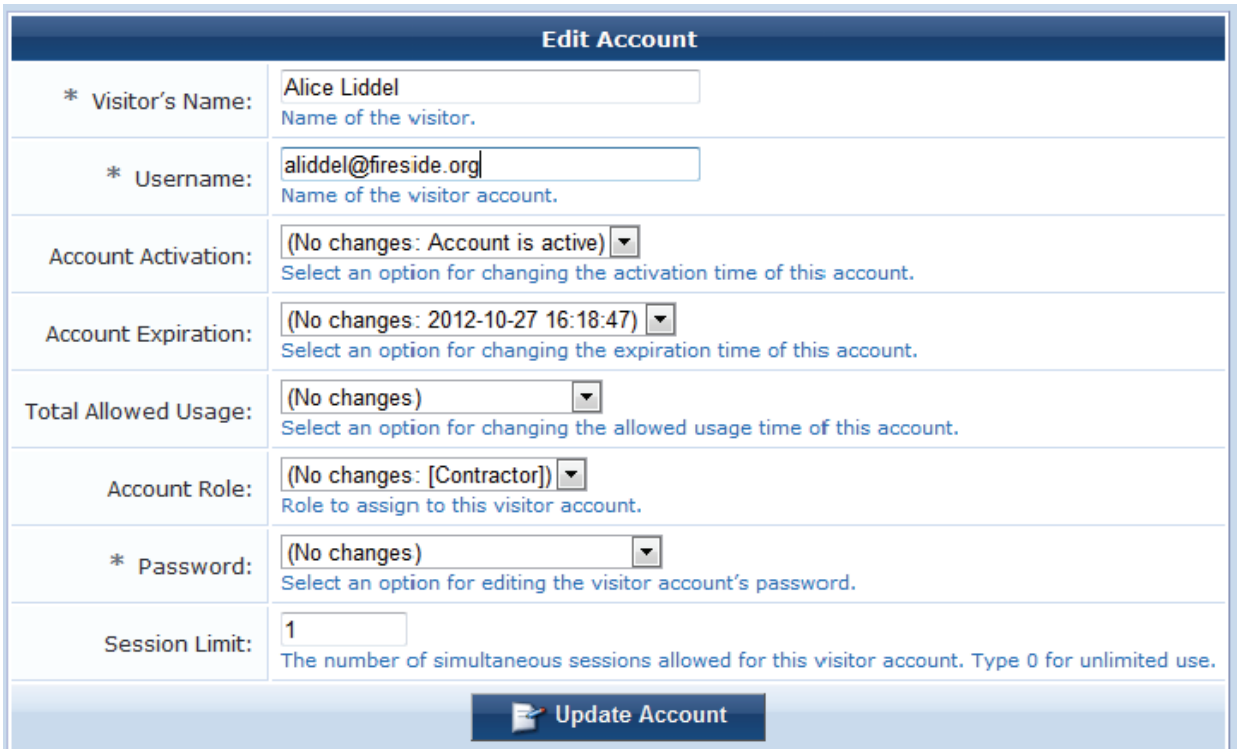
If you wish to have automatic disconnect messages sent when the enabled value changes, you can specify this in the Configuration module. See "Configuring ClearPass Guest Authentication" in the "Configuration" chapter.

- **Activate** – Re-enables a disabled guest account, or specifies an activation time for the guest account.



Select an option from the drop-down list to change the activation time of the guest account. To re-enable an account that has been disabled, choose **Now**. Click **Enable Account** to set the new activation time for the guest account. A new account receipt is displayed, allowing you to print a receipt showing the updated account details.

- **Edit** – Changes the properties of a guest account.



NOTE

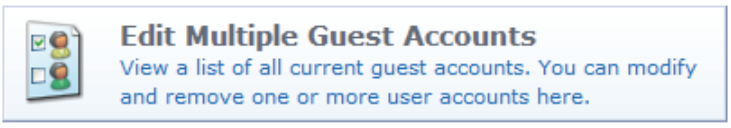
This form may be customized by adding new fields, or modifying or removing the existing fields. Refer to the "Customizing Forms and Views" section of the "Configuration"chapter for details about this customization process. This is the guest_edit form.

Click **Update Account** to update the properties of the guest account. A new account receipt is displayed, allowing you to print a receipt showing the updated account details.

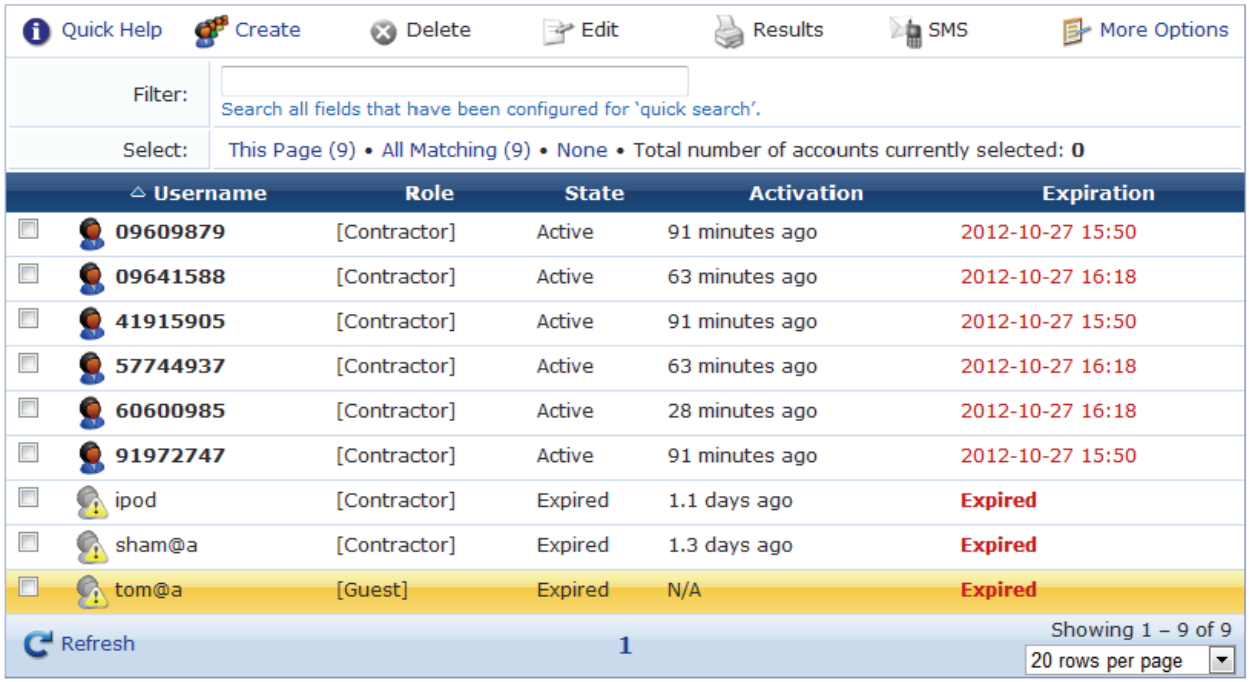
- **Sessions** – Displays the active sessions for a guest account. See "Active Sessions Management" in this chapter for details about managing active sessions.
- **Print** – Displays the guest account's receipt and the delivery options for the receipt. For security reasons, the guest's password is not displayed on this receipt. To recover a forgotten or lost guest account password, use the **Reset password** link.

Managing Multiple Guest Accounts

Use the **Edit Accounts** list view to work with multiple guest accounts. This view may be accessed by clicking the **Edit Multiple Guest Accounts** command link.



This view (guest_multi) may be customized by adding new fields or by modifying or removing the existing fields. See **"Customizing Self Provisioned Access"** in the "Configuration" chapter for details about this customization process. The default settings for this view are described below.



The **Username**, **Role**, **State**, **Activation**, and **Expiration** columns display information about the visitor accounts that have been created:

- The value in the **Expiration** column is colored red if the visitor account will expire within the next 24 hours. The expiration time is additionally highlighted in **boldface** if the visitor account will expire within the next hour.
- In addition, icons in the **Username** column indicate the account's activation status:
  - —Visitor account is active
  - —Visitor account was created but is not activated yet
  - —Visitor account was disabled by Administrator
  - —Visitor account has expired

You can use the **Filter** field to narrow the search parameters. You may enter a simple substring to match a portion of the username or any other fields that are configured for search, and you can include the following operators:

Table 8 *Operators supported in filters*

Operator	Meaning	Additional Information
=	is equal to	You may search for multiple values when using the equality (=) or inequality !=) operators. To specify multiple values, list them separated by the pipe character (|). For example, specifying the filter "role_id=2|3, custom_field=Value" restricts the accounts displayed to those with role IDs 2 and 3 (Guest and Employee), and with the field named "custom_field" set to "Value".
!=	is not equal to	
>	is greater than	
>=	is greater than or equal to	
<	is less than	
<=	is less than or equal to	
~	matches the regular expression	
!~	does not match the regular expression	

To restore the default view, click the **Clear Filter** link.

Use the paging control at the bottom of the list to jump forwards or backwards by one page, or to the first or last page of the list. You can also click an individual page number to jump directly to that page.

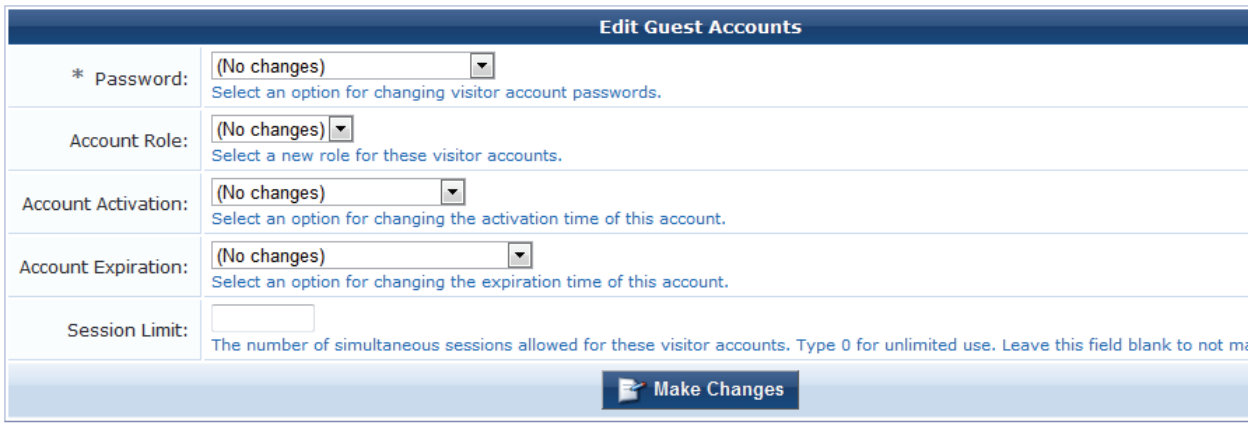
To select guest accounts, click the accounts you want to work with. You may click either the check box or the row to select a visitor account. To select or unselect all visible visitor accounts, click the check box in the header row of the table.

Use the selection row at the top of the table to work with the current set of selected accounts. The number of currently selected accounts is shown. When a filter is in effect, the “All Matching” link can be used to add all pages of the filtered result to the selection.

Use the **Create** tab to create new visitor accounts using the **Create Guest Accounts** form. See “Managing Multiple Guest Accounts” in this chapter for details about this form.

Use the **Delete** tab to delete the visitor accounts that you have selected. This option is not active if there are no visitor accounts selected.

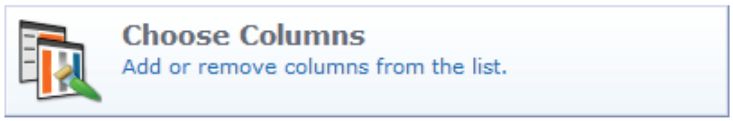
Use the **Edit** tab to make changes to multiple visitor accounts at once. This option is not active if there are no visitor accounts selected.



The Edit Guest Accounts form may be customized by adding new fields, or modifying or removing the existing fields. See "Customizing Self Provisioned Access" in the "Configuration" chapter for details about this customization process. This is the **guest_multi_form** form.

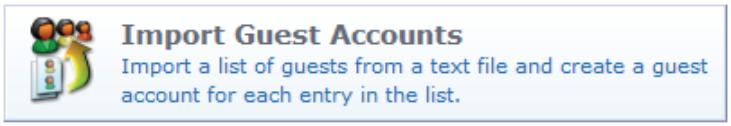
The **Results** tab will be automatically selected after you have made changes to one or more guest accounts. You can create new guest account receipts or download the updated guest account information. See "Creating Multiple Guest Account Receipts" in this chapter for more information.

The **More Options** tab includes **the Choose Columns** command link. You can click this link to open the Configuration module's Customize View Fields form, which may be used to customize the Edit Guest Accounts view.

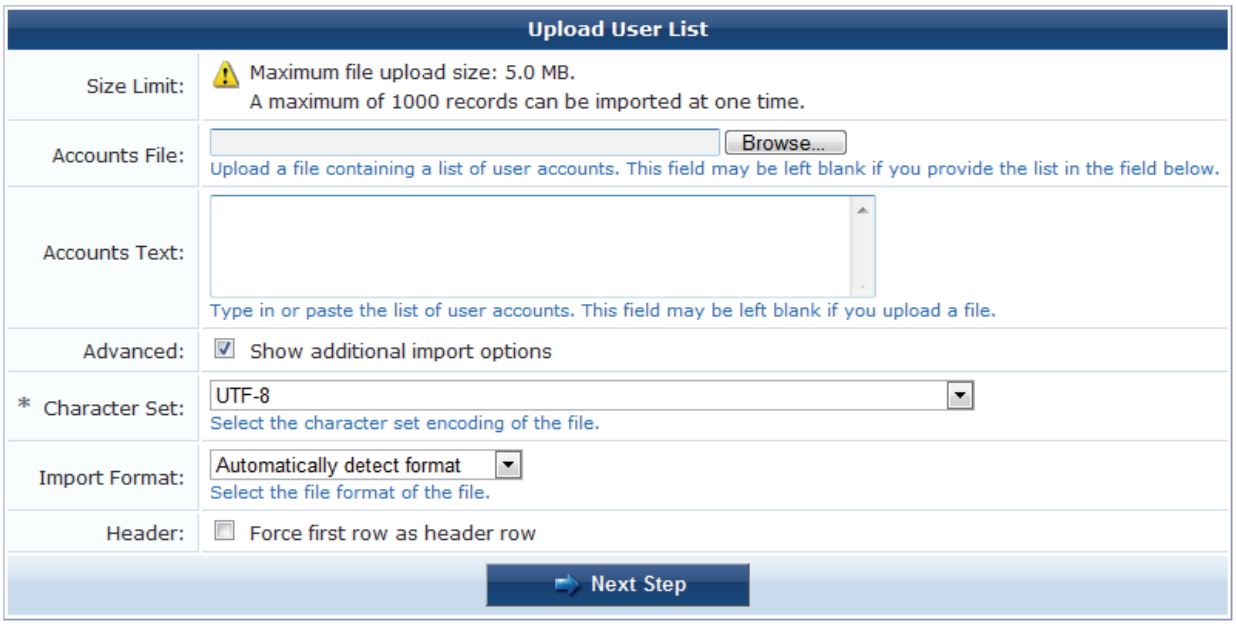


Importing Guest Accounts

Guest accounts may be created from an existing list by uploading the list to ClearPass Guest. To upload a list of existing accounts, go to **Guest > Import Accounts**, or click the **Import Guest Accounts** command link on the Guest Manager page. The Upload User List form opens.



The **Upload User List** form provides you with different options for importing guest account data.



To complete the form, you must either specify a file containing account information, or type or paste in the account information to the Accounts Text area.

Select the **Show additional import options** check box to display the following advanced import options:

- **Character Set**: ClearPass Guest uses the UTF-8 character set encoding internally to store visitor account information. If your accounts file is not encoded in UTF-8, the import may fail or produce unexpected results if non-ASCII characters are used. To avoid this, you should specify what character set encoding you are using.
- **Import format**: The format of the accounts file is automatically detected. You may specify a different encoding type if automatic detection is not suitable for your data. The **Import Format** drop-down list includes the following options:
  - **Automatically detect format** (This default option recognizes guest accounts exported from ClearPass Policy Manager in XML format)
  - **XML**
  - **Comma separated values**
  - **Tab separated values**
  - **Pipe (|) separated values**
  - **Colon (:) separated values**
  - **Semicolon (;) separated values**
- Select the **Force first row as header row** check box if your data contains a header row that specifies the field names. This option is only required if the header row is not automatically detected.

Click **Next Step** to upload the account data.

In step 2 of 3, ClearPass Guest determines the format of the uploaded account data and matches the appropriate fields are m to the data. The first few records in the data will be displayed, together with any automatically detected field names.

In this example, the following data was used:

```
username,visitor_name,password,expire_time
demo005,Demo five,secret005,2011-06-10 09:00
demo006,Demo six,secret006,2011-06-11 10:00
demo007,Demo seven,secret007,2011-06-12 11:00
```

```
demo008,Demo eight,secret008,2011-06-13 12:00
demo009,Demo nine,secret009,2011-06-13 12:00
demo010,Demo ten,secret010,2011-06-13 12:00
demo011,Demo eleven,secret011,2011-06-13 12:00
```

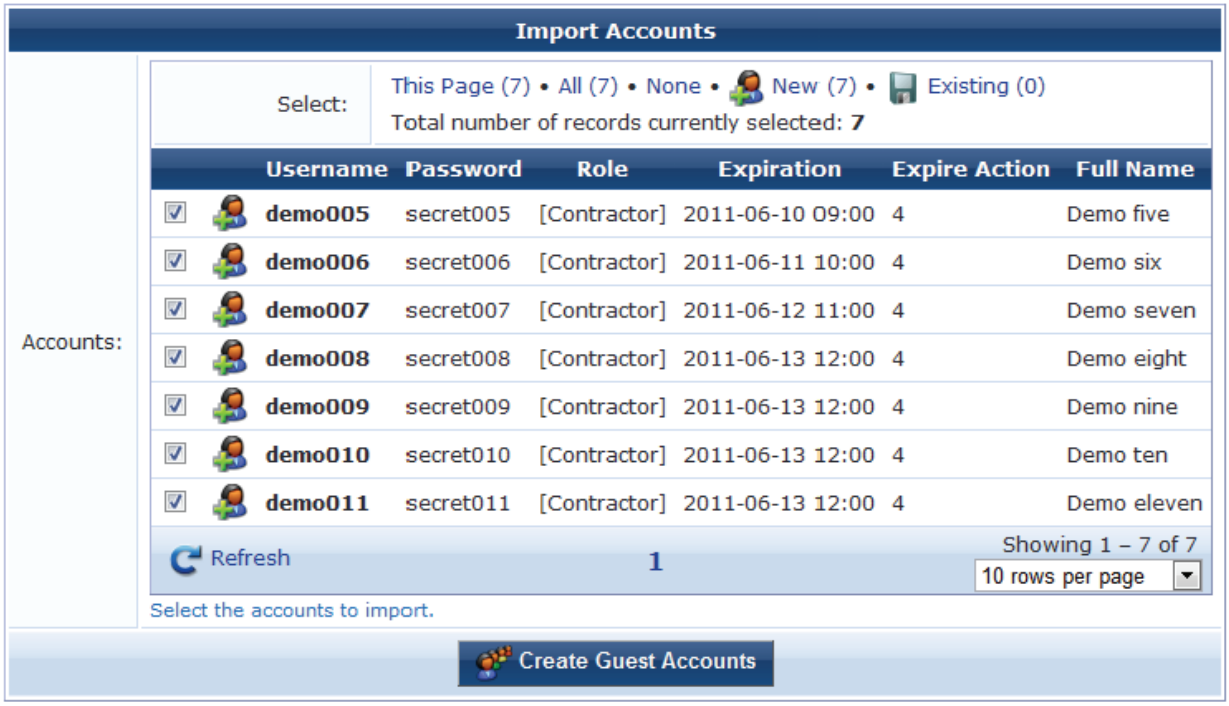
Because this data includes a header row that contains field names, the corresponding fields have been automatically detected in the data:

Record	Username	Full Name	Password	Expiration
1	username	visitor_name	password	expire_time
2	demo005	Demo five	secret005	2011-06-10 09:00
3	demo006	Demo six	secret006	2011-06-11 10:00
4	demo007	Demo seven	secret007	2011-06-12 11:00
5	demo008	Demo eight	secret008	2011-06-13 12:00
6	demo009	Demo nine	secret009	2011-06-13 12:00
7	demo010	Demo ten	secret010	2011-06-13 12:00
8	demo011	Demo eleven	secret011	2011-06-13 12:00

Use the **Match Fields** form to identify which guest account fields are present in the imported data. You can also specify the values to be used for fields that are not present in the data.

To complete the **Match Fields** form, make a selection from each of the drop-down lists. Choose a column name to use the values from that column when importing guest accounts, or select one of the other available options to use a fixed value for each imported guest account.

Click the **Next Step** button to preview the final result. Import Step 3 of 3, the Import Accounts form, opens and shows a preview of the import operation. The values of each guest account field are determined, and any conflicts with existing user accounts are shown.



The icon displayed for each user account indicates if it is a new entry ( ) or if an existing user account will be updated ( ).

By default, this form shows ten entries per page. To view additional entries, click the arrow button at the bottom of the form to display the next page, or click the **10 rows per page** drop-down list at the bottom of the form and select the number of entries that should appear on each page.

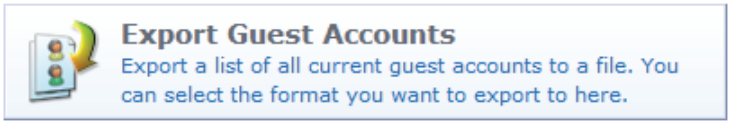
Click the check box by the account entries you want to create, or click one of the following options to select the desired accounts:

- Click the **This Page** link to select all entries on the current page.
- Click the **All** link to select all entries on all pages
- Click the **None** link to deselect all entries
- Click the **New** link to select all new entries
- Click the **Existing** link to select all existing user accounts in the list.

Click the **Create Accounts** button to finish the import process. The selected items will be created or updated. You can then print new guest account receipts or download a list of the guest accounts. See "Creating Multiple Guest Account Receipts" in this chapter for more information.

Exporting Guest Account Information

Guest account information may be exported to a file in one of several different formats.



Click the appropriate command link to save a list of all guest accounts in comma-separated values (CSV), tab-separated values (TSV), or XML format.

The Export Accounts view (guest_export) may be customized by adding new fields, or by modifying or removing the existing fields. See "Customizing Self Provisioned Access" in the "Configuration" chapter for details about this customization process.

About CSV and TSV Exports

In CSV and TSV format, the following default fields are included in the export:

- **Number** – Sequential number of the guest account in the exported data
- **User ID** – Numeric user ID of the guest account
- **Username** – Username for the guest account
- **Role** – Role for the guest account
- **Activation** – Date and time at which the guest account will be activated, or "N/A" if there is no activation time
- **Expiration** – Date and time at which the guest account will expire, or "N/A" if there is no expiration time
- **Lifetime** – The guest account's lifetime in minutes after login, or 0 if the account lifetime is not set
- **Expire Action** – Number specifying the action to take when the guest account expires (0 through 4)

About XML Exports

The default XML format consists of a **<GuestUsers>** element containing a **<GuestUser>** element for each exported guest account. The numeric ID of the guest account is provided as the "id" attribute of the **<GuestUser>** element. This format is compatible with the ClearPass Policy Manager XML format for guest users.

The values for both standard and custom fields for guest accounts are exported as the contents of an XML tag, where the tag has the same name as the guest account field.

An example XML export is given below:

```
<?xml version="1.0" encoding="UTF-8" standalone="true"?>
<TipsContents xmlns="http://www.avendasys.com/tipsapiDefs/1.0">
   <TipsHeader version="6.0" exportTime="Sun, 16 Dec 2012 16:36:03 PST"/>
   <GuestUsers>
      <GuestUser guestType="USER" enabled="true" sponsorName="55480025"
              expiryTime="2012-12-04 13:39:25" startTime="1969-12-31 16:00:00"
              password="08654361" name="55480025">
         <GuestUserTags tagValue="Hotspot Services self-provisioned guest account
              Source IP: 10.11.10.254 MAC: unknown Plan: Free Access x 1 Transaction
              Amount: $0.00 Invoice Number: P-15 Transaction ID: " tagName="notes"/>
         <GuestUserTags tagValue="2" tagName="[Role ID]"/>
         <GuestUserTags tagValue="1" tagName="do_expire"/>
         <GuestUserTags tagValue="1" tagName="simultaneous_use"/>
         <GuestUserTags tagValue="ff" tagName="Company Name"/>
         <GuestUserTags tagValue="2012-12-04 12:39:14" tagName="Create Time"/>
         <GuestUserTags tagValue="fff@df" tagName="Email"/>
         <GuestUserTags tagValue="ff" tagName="first_name"/>
         <GuestUserTags tagValue="plan0" tagName="hotspot_plan_id"/>
         <GuestUserTags tagValue="Free Access" tagName="hotspot_plan_name"/>
         <GuestUserTags tagValue="ff" tagName="last_name"/>
         <GuestUserTags tagValue="ff ff" tagName="Visitor Name"/>
         <GuestUserTags tagValue="ff" tagName="zip"/>
      </GuestUser>
```

MAC Authentication in ClearPass Guest

ClearPass Guest supports a number of options for MAC Authentication and the ability to authenticate devices.

The advanced features described in this section generally require a WLAN capable of MAC authentication with captive portal fallback. Please refer to the Aruba WLAN documentation for setting up the controller appropriately.

To verify that you have the most recent MAC Authentication Plugin installed and enabled before you configure these advanced features, go to **Administration > Plugin Manager > List Available Plugins**. For information on plugin management, see "Plugin Manager" in the "Administration"chapter.

MAC Address Formats

Different vendors format the client MAC address in different ways—for example:

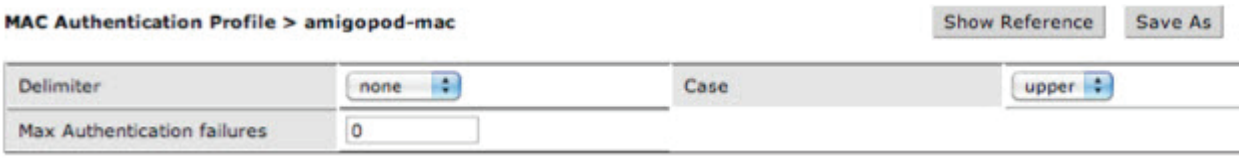
- 112233AABBCC
- 11:22:33:aa:bb:cc
- 11-22-33-AA-BB-CC

ClearPass Guest supports adjusting the expected format of a MAC address. To configure formatting of separators and case in the address, as well as user detection and device filtering for views, go to **Administration > Plugin Manager > Manage Plugins** and click the **Configuration** link for the **MAC Authentication** plugin. The MAC Authentication Configuration page opens.

Figure 7 *MAC Authentication Plugin—Configuration*

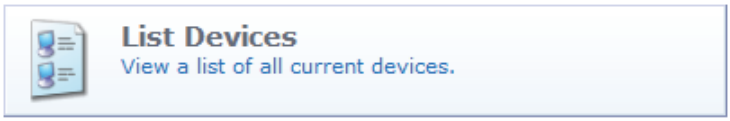
On the controller, the fields look as follows:

Figure 8 *MAC Authentication Profile*



Managing Devices

To view the list of current MAC devices, go to **Guest > List Devices**.



The Guest Manager Devices page opens.

All devices created by one of methods described in the following section are listed. Options on the form let you change a device's account expiration date; remove, activate, or edit the device; view active sessions or details for the device; or print details, receipts, confirmations, or other information.

The **MAC Address**, **Role**, **State**, **Activation**, and **Expiration** columns display information about the device accounts that have been created:

- The value in the **Expiration** column is colored red if the device account will expire within the next 24 hours. The expiration time is additionally highlighted in **boldface** if the device account will expire within the next hour.
- In addition, icons in the **MAC Address** column indicate the device account's activation status:
  - —Device account is active
  - —Device account was created but is not activated yet
  - —Device account was disabled by Administrator
  - —Device account has expired
  - —Device account was deleted

You can use the **Filter** field to narrow the search parameters. You may enter a simple substring to match a portion of any fields that are configured for search, and you can include the following operators:

Table 9 *Operators supported in filters*

Operator	Meaning	Additional Information
=	is equal to	You may search for multiple values when using the equality (=) or inequality !=) operators. To specify multiple values, list them separated by the pipe character (|). For example, specifying the filter "role_id=2|3, custom_field=Value" restricts the accounts displayed to those with role IDs 2 and 3 (Guest and Employee), and with the field named "custom_field" set to "Value".
!=	is not equal to	
>	is greater than	
>=	is greater than or equal to	
<	is less than	
<=	is less than or equal to	
~	matches the regular expression	
!~	does not match the regular expression	

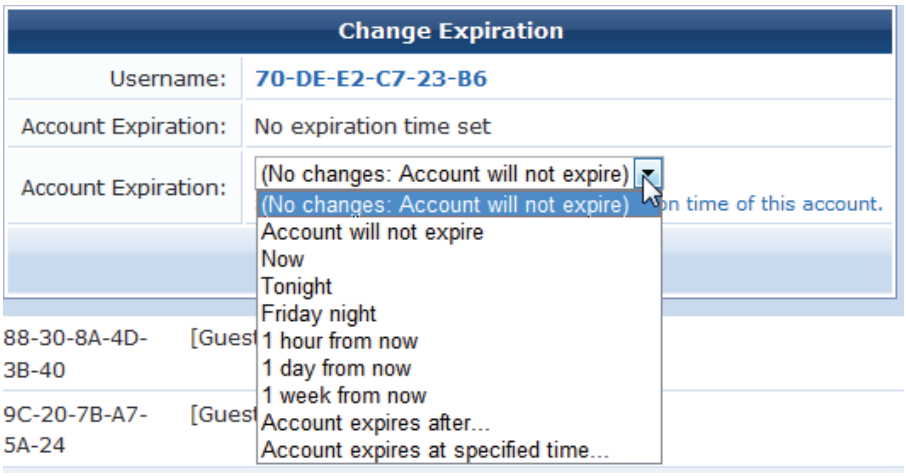
To restore the default view, click the **Clear Filter** link.

Use the paging control at the bottom of the list to jump forwards or backwards by one page, or to the first or last page of the list. You can also click an individual page number to jump directly to that page.

To select a device, click the device you want to work with.

Changing a Device's Expiration Date

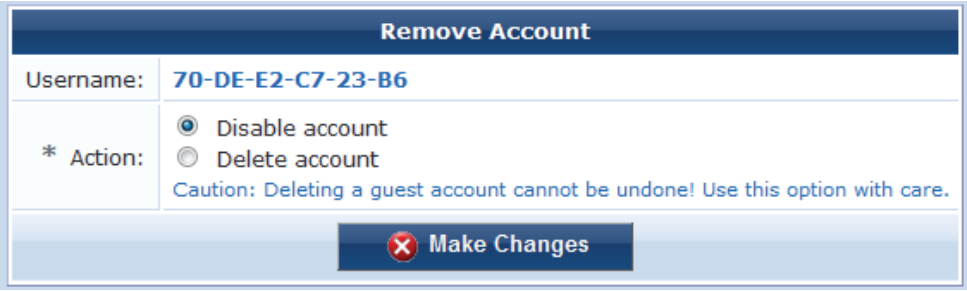
To change a device's expiration date, click the device's row in the Guest Manager Devices list, then click its **Change expiration** link. The row expands to include the Change Expiration form.



1. In the **Account Expiration** row, choose one of the options in the drop-down list to set an expiration date:
   - If you choose **Account expires after**, the **Expires After** row is added to the form. Choose an interval of hours, days, or weeks from the drop-down list.
   - If you choose **Account Expires at a specified time**, the **Expiration Time** row is added to the form. Click the button to open the calendar picker. In the calendar, use the arrows to select the year and month, click the numbers in the **Time** fields to increment the hours and minutes, then click a day to select the date.
2. If you choose any option other than "will not expire" or "now" in the Account Expiration field, the **Expire Action** row is added to the table. Use the drop-down list in this row to specify one of the following actions: delete, delete and log out, disable, or disable and log out.
3. Click **Update Account** to commit your changes.

Disabling and Deleting Devices

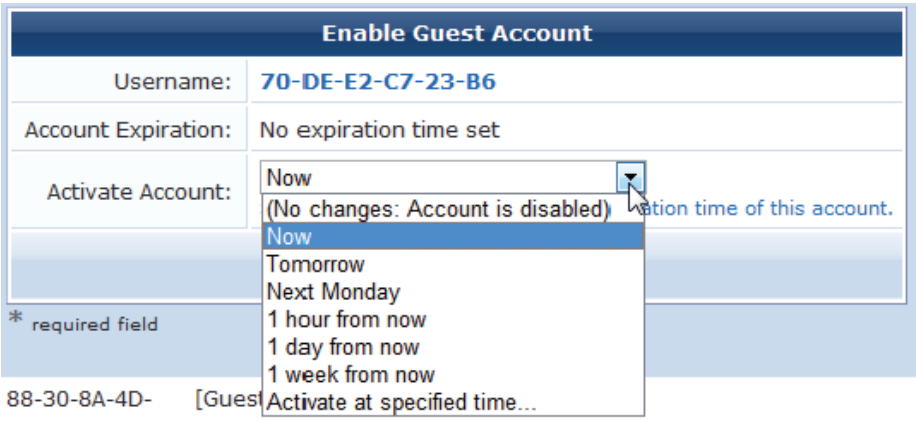
To remove a device's account by disabling or deleting it, click the device's row in the Guest Manager Devices list, then click its **Remove** link. The row expands to include the Remove Account form.



You may choose to either disable or delete the account. If you disable it, it remains in the device list and you may activate it again later. If you delete the account, it is removed from the list permanently.

Activating a Device

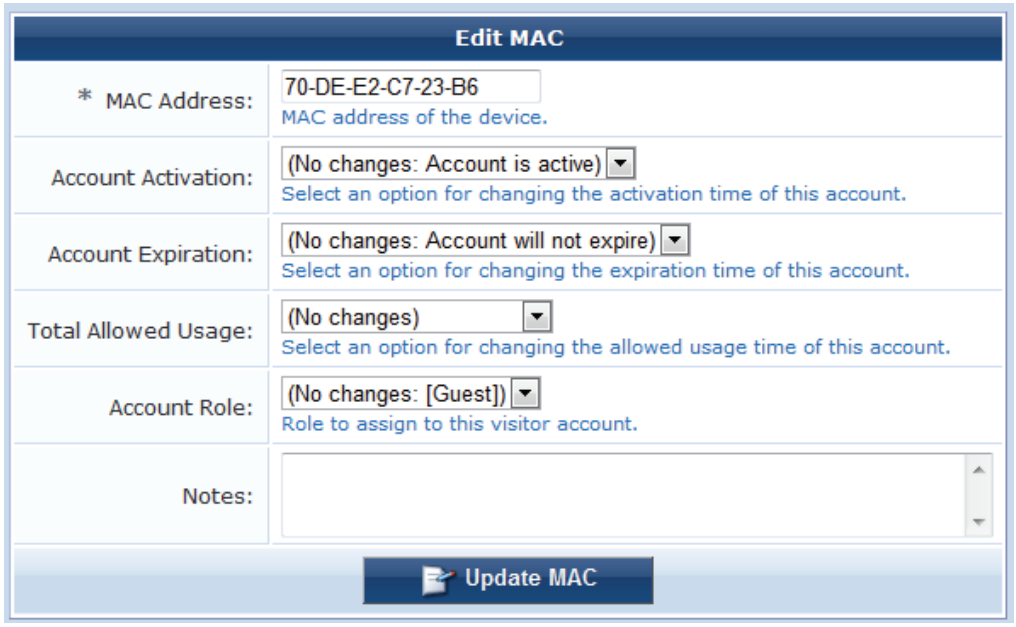
To activate a disabled device's account, click the device's row in the Guest Manager Devices list, then click its **Activate** link. The row expands to include the Enable Guest Account form.



1. In the **Activate Account** row, choose one of the options in the drop-down list to specify when to activate the account. You may choose an interval, or you may choose to specify a time.
2. If you choose **Activate at specified time**, the **Activation Time** row is added to the form. Click the button to open the calendar picker. In the calendar, use the arrows to select the year and month, click the numbers in the **Time** fields to increment the hours and minutes, then click a day to select the date.
3. Click **Enable Account** to commit your changes.

Editing a Device

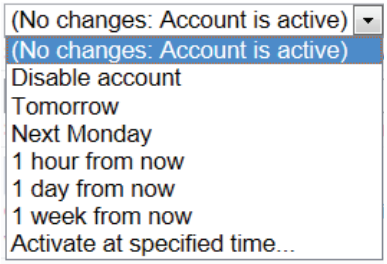
To edit a device's account, click the device's row in the Guest Manager Devices list, then click its **Edit** link. The row expands to include the Edit MAC form.



1. You can change the device's address in the **MAC Address** row.

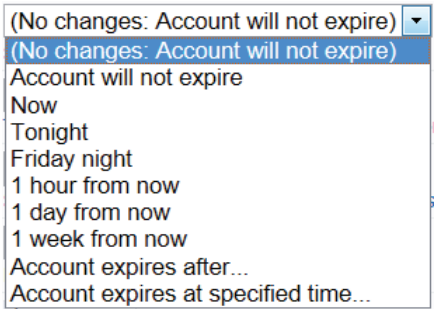
   If you need to modify the configuration for expected separator format or case, go to **Administration > Plugin Manager > Manage Plugins** and click the **Configuration** link for the **MAC Authentication** plugin.

2. If you need to change the activation time, choose one of the options in the **Account Activation** drop-down list. You may choose to activate the account immediately, at a preset interval of hours or days, or at a specified time.



   - If you choose **Activate at a specified time**, the **Activation Time** row is added to the form. Click the button to open the calendar picker. In the calendar, use the arrows to select the year and month, click the numbers in the **Time** fields to increment the hours and minutes, then click a day to select the date.

3. If you need to change the expiration time, choose one of the options in the **Account Expiration** drop-down list. You may terminate the account immediately, at a preset interval of hours or days, or at a specified time.



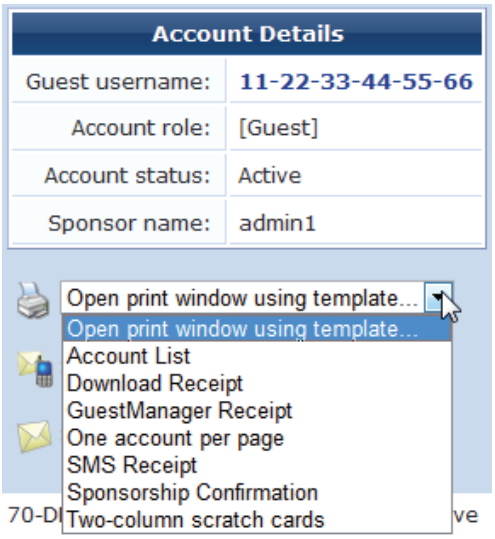
   - If you choose any time in the future, the **Expire Action** row is added to the form. Use this drop-down list to indicate the expiration action for the account—either delete, delete and log out, disable, or disable and log out. The action will be applied at the time set in the Account Expiration row.
   - If you choose **Account expires after**, the **Expires After** row is added to the form. Choose an interval of hours, days, or weeks from the drop-down list. The maximum is two weeks.
   - If you choose **Account Expires at a specified time**, the **Expiration Time** row is added to the form. Click the button to open the calendar picker. In the calendar, use the arrows to select the year and month, click the numbers in the **Time** fields to increment the hours and minutes, then click a day to select the date.
4. To change the maximum usage allowed for the account, choose an option from the **Total Allowed Usage** drop-down list. You may set the total usage to one or two hours, add one or two hours to the existing setting, or subtract one or two hours from the existing setting.
5. You can use the **Account Role** drop-down list to change the visitor's assigned role.
6. (Optional) In the **Notes** row, you may enter additional information.
7. To commit your changes, click **Update MAC**.

Viewing Current Sessions for a Device

To view any sessions that are currently active for a device, click the **Sessions** link in the device's row on the Guest Manager Devices form. The Active Sessions list opens. For more information, see "Active Sessions Management".

Viewing and Printing Device Details

To print details, receipts, confirmations, or other information for a device, click the device's row in the Guest Manager Devices list, then click its **Print** link. The row expands to include the Account Details form and a drop-down list of information that can be printed for the device.



Choosing an option in the **Open print window using template** drop-down list opens a print preview window and the printer dialog. Options include account details, receipts in various formats, a session expiration alert, and a sponsorship confirmation notice.

MAC Creation Modes

MAC device accounts may be created in three ways:

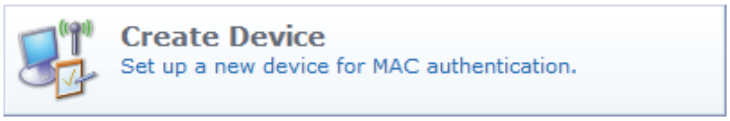
- Manually in ClearPass Guest using the Create Device form
- During guest self-registration by a mac parameter passed in the redirect URL, if the process is configured to create a MAC device account
- During guest self-registration by a mac parameter passed in the redirect URL, creating a parallel account paired with the visitor account

Creating Devices Manually in ClearPass Guest

If you have the MAC address, you can create a new device manually. You do this on the New MAC Authentication form.

To create a new device:

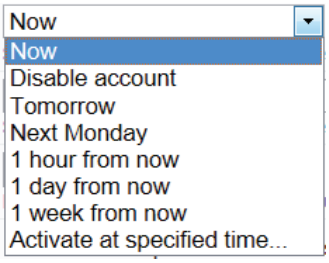
1. Go to **Guest > List Devices** and click the **Create** link, or you can go to the Guest navigation page and click the **Create Device** command.



The New MAC Authentication page opens.

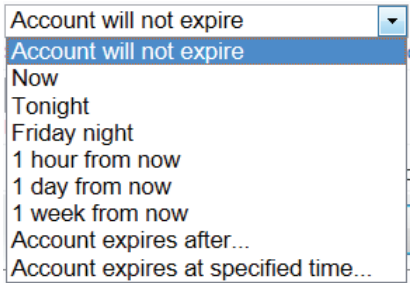
2. In the **Sponsor's Name** row, enter the name of the person sponsoring the visitor account.
3. Enter the name for the device in the **Device Name** row.
4. Enter the address in the **MAC Address** row.

   If you need to modify the configuration for expected separator format or case, go to **Administration > Plugin Manager > Manage Plugins** and click the **Configuration** link for the **MAC Authentication Plugin**.
5. Choose one of the options in the **Account Activation** drop-down list. You may choose to activate the account immediately, at a preset interval of hours or days, at a specified time, or leave the account disabled.



- If you choose **Activate at a specified time**, the **Activation Time** row is added to the form. Click the button to open the calendar picker. In the calendar, use the arrows to select the year and month, click the numbers in the **Time** fields to increment the hours and minutes, then click a day to select the date.

6. To set the account's expiration time, choose one of the options in the **Account Expiration** drop-down list. You may set the account to never expire, or to expire at a preset interval of hours or days, or at a specified time.



   - If you choose any time in the future, the **Expire Action** row is added to the form. Use this drop-down list to indicate the expiration action for the account—either delete, delete and log out, disable, or disable and log out. The action will be applied at the time set in the Account Expiration row.
   - If you choose **Account expires after**, the **Expires After** row is added to the form. Choose an interval of hours, days, or weeks from the drop-down list. The maximum is two weeks.
   - If you choose **Account Expires at a specified time**, the **Expiration Time** row is added to the form. Click the button to open the calendar picker. In the calendar, use the arrows to select the year and month, click the numbers in the **Time** fields to increment the hours and minutes, then click a day to select the date.
7. Use the **Account Role** drop-down list to assign the visitor's role.
8. In the **Terms of Use** row, first click the **terms of use** link and read the agreement, then mark the check box to agree to the terms.
9. To commit your changes and create the device, click **Create MAC**. The Account Details and print options are displayed. For more information, see "Viewing and Printing Device Details".

Creating Devices During Self-Registration - MAC Only

This section describes how to configure a guest self-registration so that it creates a MAC device account. Once the guest is registered, future authentication can take place without the need for the guest to enter their credentials. A registration can be converted to create a MAC device instead of standard guest credentials.

This requires a vendor passing a **mac** parameter in the redirect URL. ClearPass Guest does not support querying the controller or DHCP servers for the client's MAC based on IP.

To edit the registration form fields, go to **Configuration > Forms and Views**. In the **guest_register** row, click the **Edit Fields** link. The Customize Form Fields page opens. If you do not see mac or mac_auth in the list, click the **Customize fields** link above the list. Click the **Edit** link in the field's row. In the Define Custom Field form, edit the registration form fields:

- Add or enable **mac**
  - UI: **Hidden field**
  - Field Required: checked
  - Validator: **IsValidMacAddress**
- Add or enable **mac_auth**
  - UI: **Hidden field**
- Any other expiration options, role choice, surveys, and so on can be entered as usual.

Figure 9 *Modify fields*

Rank	Field	Type	Label	Description
30	visitor_company	text	Company Name:	Please enter your company name.
40	email	text	Email Address:	Please enter your email address. This will become your username to log into the network.
45	**mac**	text	MAC Address:	MAC address of the device.
47	**mac_auth**	hidden		
50	start_time	datetime	Activation Time:	Scheduled date and time at which to enable the visitor account. If blank, the account will be enabled immediately.
60	**expire_after**	hidden	Expires After:	Amount of time before this visitor account will expire.
65	expire_time	datetime	Expiration Time:	Optional date and time at which the visitor accounts will expire and be deleted. If blank, the account will not expire.

- Edit the receipt form fields:
  - Edit **username** to be a **Hidden field**
  - Edit **password** to be a **Hidden field**
- Adjust any headers or footers as needed.

When the visitor registers, they should be able to still log in via the **Log In** button. The MAC will be passed as their username and password via standard captive portal means.

The account will only be visible on the **List Devices** page.

If the guest logs out and reconnects, they should be immediately logged in without being redirected to the captive portal page.

Creating Devices During Self-Registration - Paired Accounts

Paired accounts is a means to create a standard visitor account with credentials, but to have a MAC account created in parallel that is directly tied to the visitor account. These accounts share the same role, expiration and other properties.

This requires a vendor passing a **mac** parameter in the redirect URL. ClearPass Guest does not support querying the controller or DHCP servers for the client's MAC based on IP.

To edit the registration form fields, go to **Configuration > Forms and Views**. In the **guest_register** row, click the **Edit Fields** link. The Customize Form Fields page opens. If you do not see mac or mac_auth_pair in the list, click the **Customize fields** link above the list. Click the **Edit** link in the field's row. In the Define Custom Field form, edit the registration form fields:

- Add or enable **mac**
  - UI: **Hidden field**
  - Field Required: optional
  - Validator: **IsValidMacAddress**
- Add or enable **mac_auth_pair**
  - UI: **Hidden field**
  - Initial Value: **-1**
- Any other expiration options, role choice, surveys and so on can be entered as usual.

You will see an entry under both **List Accounts** and **List Devices**. Each should have a **View Pair** action that cross links the two. Note if you delete the base account, all of its pairings will also be deleted. If RFC-3576 has been configured, all pairs will be logged out.

AirGroup Device Registration

AirGroup allows users to register their personal mobile devices on the local network and define a group of friends or associates who are allowed to share them. If AirGroup Services is enabled, AirGroup administrators can provision their organization's shared devices and manage access, and AirGroup operators can register and provision a limited number of their own personal devices for sharing. For complete AirGroup deployment information, refer to the AirGroup Deployment Guide and the ClearPass Policy Manager documentation.

Registering Groups of Devices or Services

This functionality is only available to AirGroup administrators.

To register and manage an organization's shared devices and configure device access:

1. Log in as the AirGroup administrator and go to **Guest > Create Device**. The Register Shared Device form opens.

2. In the **Device Name** field, enter the name used to identify the device.
3. In the **MAC Address** field, enter the device's MAC address.
4. In the **Shared Locations** field, enter the locations where the device can be shared. To allow the device to be shared with all locations, leave this field blank.

   Each location is entered as a tag=value pair describing the MAC address of the access point (AP) closest to the registered device. Use commas to separate the tag=value pairs in the list. Tag=value pair formats are shown in the following table.

Table 10 *Tag=Value Pair Formats*

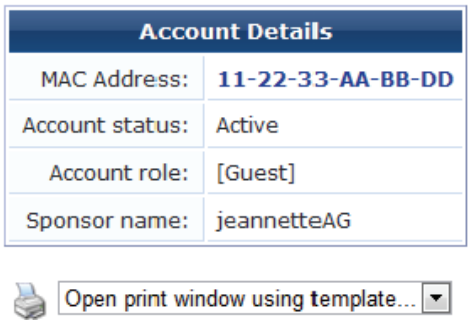
AP Type	Tag=Value Format
Name-based AP	ap-name=<name>
Group-based AP	ap-group=<group>
FQLN-based AP	fqln=<fqln>

- AP FQLNs should be configured in the format `<ap name>.<floor>.<building>.<campus>`
- Floor names should be in the format `floor <number>`
- The `<ap-name>` should not include periods (.)

Example:

```
AP105-1.Floor 1.TowerD.Aruba
```

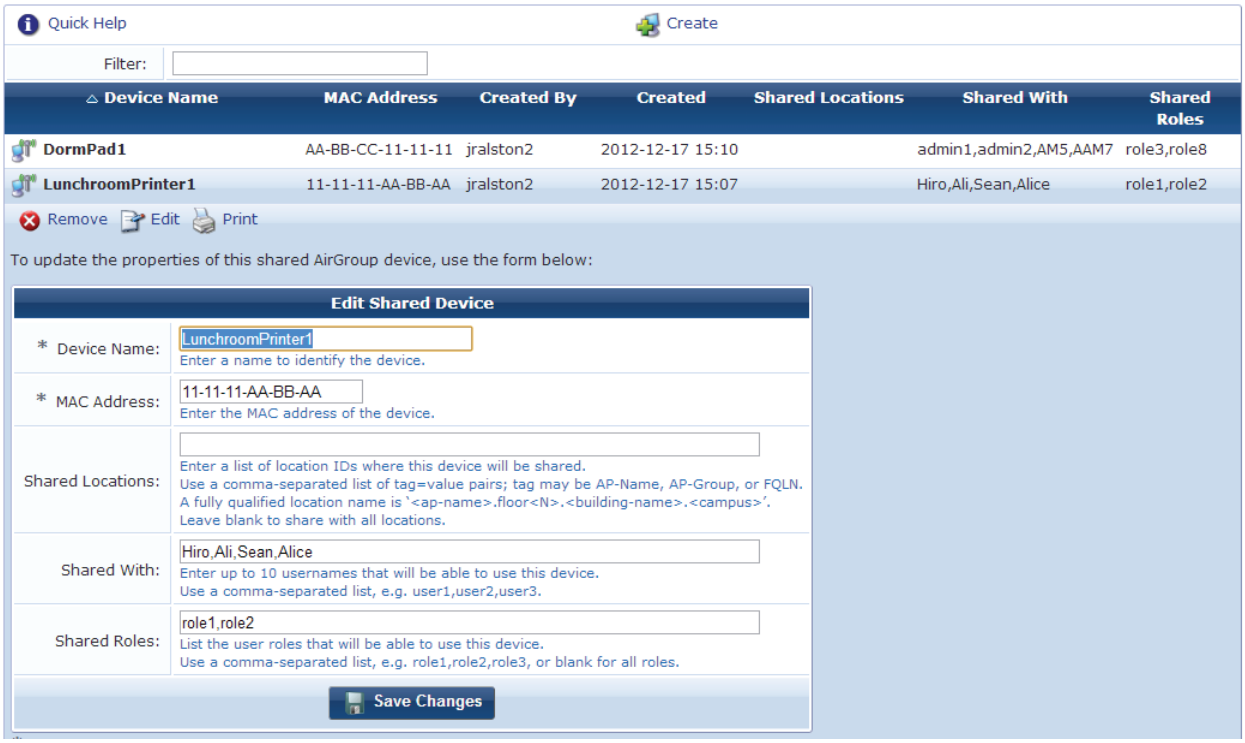
5. In the **Shared With** field, enter the usernames of your organization's staff or students who are allowed to use the device. Use commas to separate usernames in the list.
   - If the **Share With** field is left blank, this device can be accessed by all devices.
   - If users are entered in the **Shared With** field, the device can only be accessed by the specified users.
6. In the **Shared Roles** field, enter the user roles that are allowed to use the device. Use commas to separate the roles in the list.
   - To make the device available to all roles, leave this field blank.
   - If roles are entered in the Shared Roles field, the device can only be accessed by users with matching roles.
7. Click **Register Shared Device**. The Finished Creating Guest Account page opens. This page displays Account Details and provides printer options.



To view and edit your organization's shared AirGroup devices:

1. Go to **Guest > List Devices**, or click the **Manage my AirGroup Devices** link on the Create AirGroup Device page. The AirGroup Devices page opens. This page lists all the shared AirGroup devices for the organization. You can remove a device; edit a device's name, MAC address, shared locations, shared-user list, or shared roles; print device details; or add a new device.

2. To work with a device, click the device's row in the list. The form expands to include the **Remove**, **Edit**, and **Print** options.



3. To edit properties of a shared device, click the **Edit** link for the device. The row expands to include the Edit Shared Device form. You can modify the device's name, MAC address, shared locations, group of users, and shared roles.
4. When your edits are complete, click **Save Changes**.

Registering Personal Devices

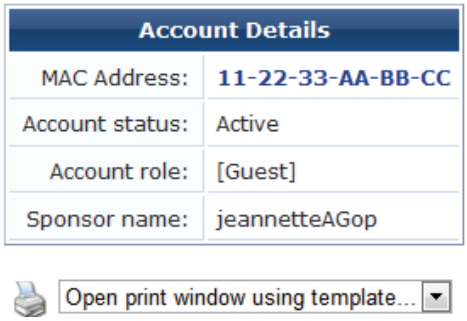
This functionality is available to AirGroup operators.

To register your personal devices and define a group who can share them:

1. Log in as the AirGroup operator and go to **Guest > Create Device**. The Register Device form opens.

2. In the **Your Name** field, enter your username for your organization.
3. In the **Device Name** field, enter the name used to identify the device.

4. In the **MAC Address** field, enter the device's MAC address.
5. In the **Shared With** field, enter the usernames of your friends or colleagues who are allowed to use the device. Use commas to separate usernames in the list. You may enter up to ten usernames.
   - If the **Shared With** field is left blank, this device can only be accessed by devices registered by the same operator or with a dot1x username that matches the operator's name.
   - If users are entered in the **Shared With** field, the device can be accessed by the device owner and by the specified users.
6. Click **Register Device**. The Finished Creating Guest Account page opens. This page displays Account Details and provides printer options.

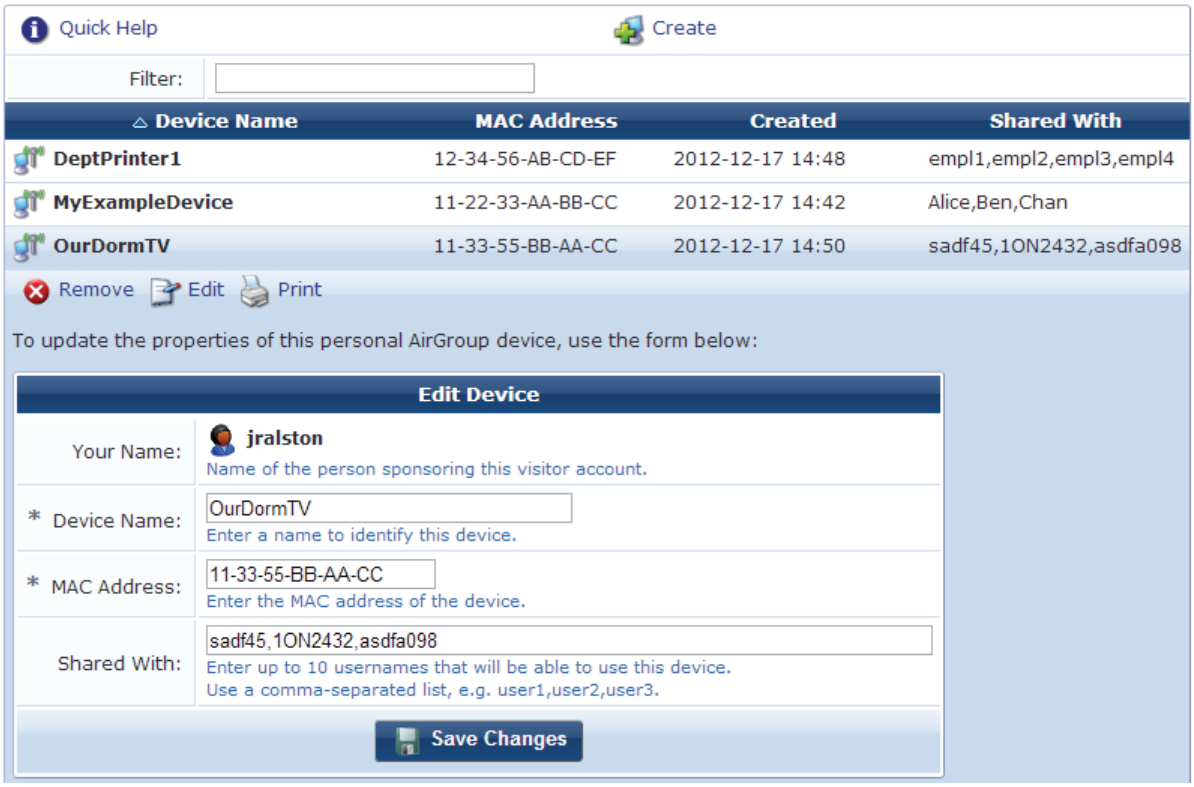


To view and edit your personal AirGroup devices, go to **Guest > List Devices**, or click the **Manage my AirGroup Devices** link on the Create AirGroup Device page. The List Device page lets you remove a device; edit a device's name, MAC address, or shared-user list; print device details; or add a new device.

To view and edit your personal AirGroup devices:

1. Go to **Guest > List Devices**, or click the **Manage my AirGroup Devices** link on the Create AirGroup Device page. The AirGroup Devices page opens. This page lists all your personal AirGroup devices. You can remove a device; edit a device's name, MAC address, or shared-user list; print device details; or add a new device.

2. To work with a device, click the device's row in the list. The form expands to include the **Remove**, **Edit**, and **Print** options.



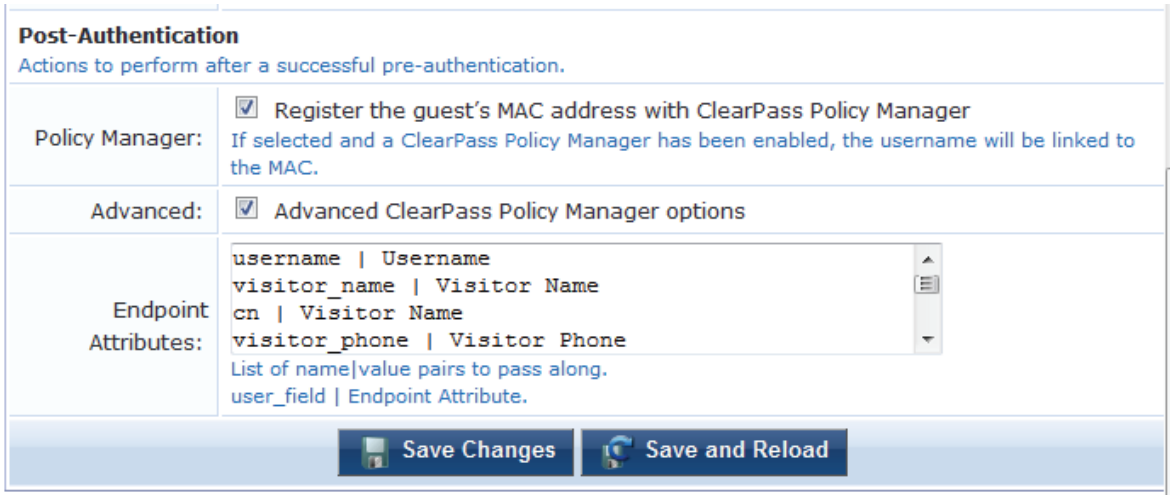
3. To edit properties of a device, click the **Edit** link for the device. The row expands to include the Edit Device form. You can modify the device's name, MAC address, and group of users.
4. When your edits are complete, click **Save Changes**.

Automatically Registering MAC Devices in ClearPass Policy Manager

If ClearPass Policy Manager is enabled, you can configure a guest MAC address to be automatically registered as an endpoint record in ClearPass Policy Manager when the guest uses a Web login page or a guest self-registration workflow. This customization option is available if a valid Local or RADIUS pre-authentication check was performed.

To configure auto-registration for an address through a Web login page:

1. Go to **Configuration > Web Logins**, click the row of the page you wish to configure, then click its **Edit** link. The RADIUS Web Login Editor form opens.
2. Scroll down to the **Post-Authentication** area.



3. In the **Policy Manager** row, mark the check box to register the guest's MAC address with ClearPass Policy Manager. The Advanced row is added to the form.
4. In the **Advanced** row, mark the check box to enable advanced options in ClearPass Policy Manager. The Endpoint Attributes row is added to the form.
5. In the **Endpoint Attributes** row, enter name|value pairs for the user fields and Endpoint Attributes to be passed.
6. Click **Save Changes** to complete this configuration and continue with other tasks, or click **Save and Reload** to proceed to Policy Manager and apply the network settings.

Importing MAC Devices

The standard **Guest > Import Accounts** form supports importing MAC devices. At a minimum the following two columns are required: **mac** and **mac_auth**.

```
mac_auth,mac,notes
1,aa:aa:aa:aa:aa:aa,Device A
1,bb:bb:bb:bb:bb:bb,Device B
1,cc:cc:cc:cc:cc:cc,Device C
```

Any of the other standard fields can be added similar to importing regular guests.

Advanced MAC Features

2-Factor Authentication

2-factor authentication checks against both credentials and the MAC address on record.

Tying the MAC to the visitor account will depend on the requirements of your deployment. In practice you would probably add **mac** as a text field to the **create_user** form. When **mac** is enabled in a self-registration it will be included in the account as long as **mac** is passed in the URL. Relying on self-registration may defeat the purpose of two-factor authentication, however.

The 2-factors are performed as follows:

1. Regular RADIUS authentication using username and password
2. Role checks the user account mac against the passed Calling-Station-Id.

Edit the user role and the attribute for **Reply-Message** or **Aruba-User-Role**. Adjust the condition from **Always** to **Enter conditional expression**.

```
return !MacEqual(GetAttr('Calling-Station-Id'), $user['mac']) && AccessReject();
```

There is an alternative syntax where you keep the condition at **Always** and instead adjust the **Value**.

```
<?= MacEqual(GetAttr('Calling-Station-Id'), $user['mac']) ? $role["name"] :
AccessReject()
```

or

```
<?= MacEqual(GetAttr('Calling-Station-Id'), $user['mac']) ? 'Employee' : AccessReject()
```

MAC-Based Derivation of Role

Depending on whether the MAC address matches a registered value, you can also adjust which role is returned. The controller must be configured with the appropriate roles and the reply attributes mapping to them as expected.

Edit the **Value** of the attribute within the role returning the role to the controller.

If you are on the registered MAC, apply the **Employee** role, otherwise set them as **Guest**.

```
<?= MacEqual(GetAttr('Calling-Station-Id'), $user['mac']) ? 'Employee' : 'Guest'
```

This can be expanded if you create multiple MAC fields. Navigate to **Customize > Fields** and duplicate **mac**. Rename it as mac_byod and then add it to the 'create_user and guest_edit forms. In this example the account has a registered employee device under mac, and a registered BYOD device under mac_byod.

```
<?= MacEqual(GetAttr('Calling-Station-Id'), $user['mac_byod']) ? 'BYOD' :
(MacEqual(GetAttr('Calling-Station-Id'), $user['mac']) ? 'Employee' : 'Guest')
```

User Detection on Landing Pages

When **mac** is passed in the redirect URL, the user is detected and a customized message displays on the landing page.

Navigate to **Administration > Plugin Manager > Manage Plugins: MAC Authentication: Configuration** and enable **MAC Detect**.

Edit the header of your redirect landing page (login or registration) and include the following:

```
<p>{if $guest_receipt.u.visitor_name}
Welcome back to the show, {$guest_receipt.u.visitor_name|htmlspecialchars}!
{else}
Welcome to the show!
{/if}</p>
```

For debugging purposes, include the following to see all the fields available:

```
{dump var=$guest_receipt export=html}
```

Click-Through Login Pages

A click-through login page will present a splash or terms screen to the guest, yet still provide MAC-auth style seamless authentication. Under this scenario, you could have people create an account, with a paired MAC, yet still have them click the terms and conditions on every new connection.

Disable MAC authentication on the controller.

Navigate to **Administration > Plugin Manager > Manage Plugins: MAC Authentication: Configuration** and enable **MAC Detect**.

Create a **Web Login**

- Authentication: **Anonymous**
- Anonymous User: **_mac** *(_mac is a special secret value)*
- Pre-Auth Check: **Local**
- Terms: **Require a Terms and Conditions confirmation**

Set the Web login as your landing page and test. Using a registered device the 'Log In' button should be enabled, otherwise it will be disabled.

You may also want to add a message so visitors get some direction.

```
<p>{if $guest_receipt.u.username}
{if $guest_receipt.u.visitor_name}
Welcome back, {$guest_receipt.u.visitor_name|htmlspecialchars}!
{else}
Welcome back.
{/if}
     Please accept the terms before proceeding.
{else}
You need to register...
{/if}</p>
```

You can hide the login form by having the final line of the header be:

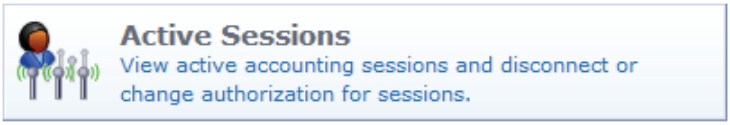
```
{if !$guest_receipt.u.username}<div style="display:none">{/if}
```

and the first line of the footer be:

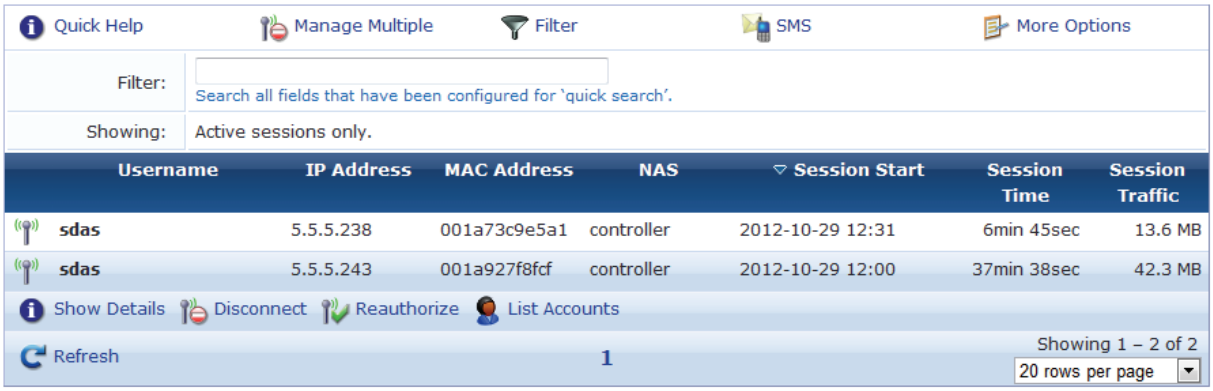
```
{if !$guest_receipt.u.username}</div>{/if}
```

Active Sessions Management

The RADIUS server maintains a list of active visitor sessions. If your NAS equipment has RFC 3576 support, the RADIUS dynamic authorization extensions allow you to disconnect or modify an active session.



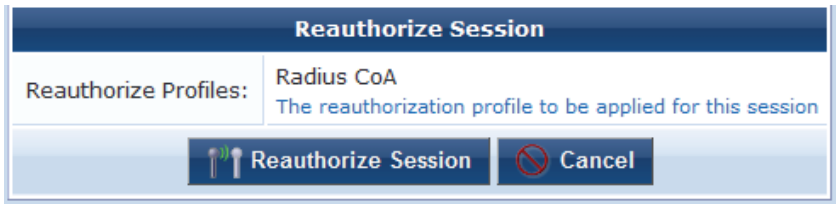
To view and manage active sessions for the RADIUS server, go to **Guest > Active Sessions**. The Active Sessions list opens. You can use this list to modify, disconnect or reauthorize, or send SMS notifications for active visitor sessions; manage multiple sessions; or customize the list to include additional fields.



- To view details for an active session, click the session's row in the list, then click its **Show Details** link. The form expands to include the Session Details view.

Session Details	
Username:	**test**
IP Address:	5.5.5.252
NAS:	controller
NAS IP Address:	10.100.9.25
NAS Port Type:	Wireless-802.11
Calling Station ID:	70DEE2C723B6
Called Station ID:	000B866D1F58
Service Type:	Onboard Service
Session ID:	R0000017f-01-508ef9f4
Session Upload:	707,722 bytes
Session Download:	45,361,239 bytes
Session End:	2012-10-29 14:51
Termination Cause:	Lost-Service

- If the NAS equipment has RFC 3576 support, you can disconnect or dynamically reauthorize active sessions. See "RFC 3576 Dynamic Authorization" for more information.
  - To disconnect an active session, click the session's row in the list, then click its **Disconnect** link. A message is displayed to show that the disconnect is in progress and acknowledge when it is complete.
  - To reauthorize a session that was disconnected, click the session's row in the list, then click its **Reauthorize** link. The Reauthorize Session form opens. Click **Reauthorize Session**. A message is displayed to show that the disconnect is in progress and acknowledge when it is complete.



  - To disconnect multiple sessions, click the **Manage Multiple** tab. The form expands to include the Manage Multiple Sessions form. For more information, see "Disconnecting Multiple Active Sessions".
- To view and work with the guest accounts associated with a session, click the session's row in the list, then click its **List Accounts** link. The Guest Manager Accounts view opens. See "Managing Guest Accounts" for more information.
- To display only sessions that meet certain criteria, click the **Filter** tab. For more information, see "Filtering the List of Active Sessions".
- To send SMS notifications to visitors, click the **SMS** tab. For more information, see "Sending Multiple SMS Alerts".
- To include additional fields in the Active Sessions list, or delete fields from it, click the **More Options** tab. The Customize View Fields page opens. For more information, see "Editing Forms" in the "Configuration" chapter.

- You can use the paging control at the bottom of the list to jump forwards or backwards by one page, or to the first or last page of the list. You can also click an individual page number to jump directly to that page.

Session States

A session may be in one of three possible states:

- **Active**—An active session is one for which the RADIUS server has received an accounting start message and has not received a stop message, which indicates that service is being provided by a NAS on behalf of an authorized client.

  While a session is in progress, the NAS sends interim accounting update messages to the RADIUS server. This maintains up-to-date traffic statistics and keeps the session active. The frequency of the accounting update messages is configurable in the RADIUS server.
- **Stale**—If an accounting stop message is never sent for a session—for example, if the visitor does not log out— that session will remain open. After 24 hours without an accounting update indicating session traffic, the session is considered 'stale' and is not counted towards the active sessions limit for a visitor account. To ensure that accounting statistics are correct, you should check the list for stale sessions and close them.
- **Closed**—A session ends when the visitor logs out or if the session is disconnected. When a session is explicitly ended in either of these ways, the NAS sends an accounting stop message to the RADIUS server. This closes the session. No further accounting updates are possible for a closed session.

RFC 3576 Dynamic Authorization

Dynamic authorization describes the ability to make changes to a visitor account's session while it is in progress. This includes disconnecting a session, or updating some aspect of the authorization for the session.

The Active Sessions page provides two dynamic authorization capabilities that apply to currently active sessions:

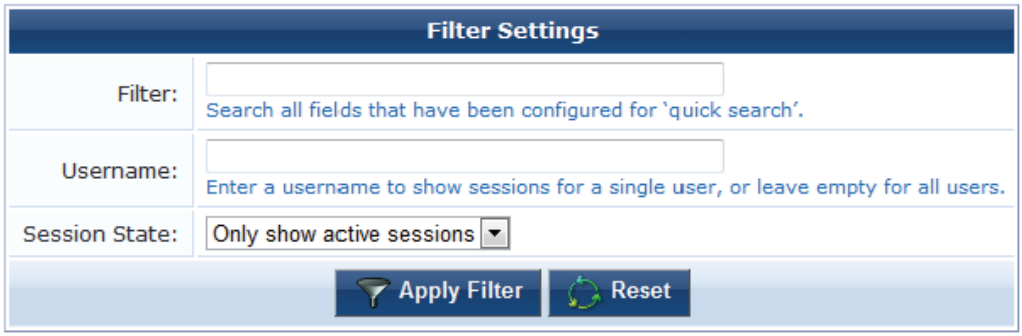
- **Disconnect** causes a Disconnect-Request message to be sent to the NAS for an active session, requesting that the NAS terminate the session immediately. The NAS should respond with a Disconnect-ACK message if the session was terminated or Disconnect-NAK if the session was not terminated.
- **Reauthorize** causes a Disconnect-Request message to be sent to the NAS for an active session. This message will contain a Service-Type attribute with the value 'Authorize Only'. The NAS should respond with a Disconnect-NAK message, and should then reauthorize the session by sending an Access-Request message to the RADIUS server. The RADIUS server's response will contain the current authorization details for the visitor account, which will then update the corresponding properties in the NAS session.

If the NAS does not support RFC 3576, attempts to perform dynamic authorization will time out and result in a 'No response from NAS' error message.

Refer to RFC 3576 for more details about dynamic authorization extensions to the RADIUS protocol.

Filtering the List of Active Sessions

You can use the **Filter** tab to narrow the search parameters and quickly find all matching sessions:



Enter a username or IP address in the **Filter** field. Additional fields can be included in the search if the "Include values when performing a quick search" option was selected for the field within the view. To control this option, use the **Choose Columns** command link on the **More Options** tab.

You may enter a simple substring to match a portion of the username or any other fields that are configured for search, and you can include the following operators:

Table 11 *Operators supported in filters*

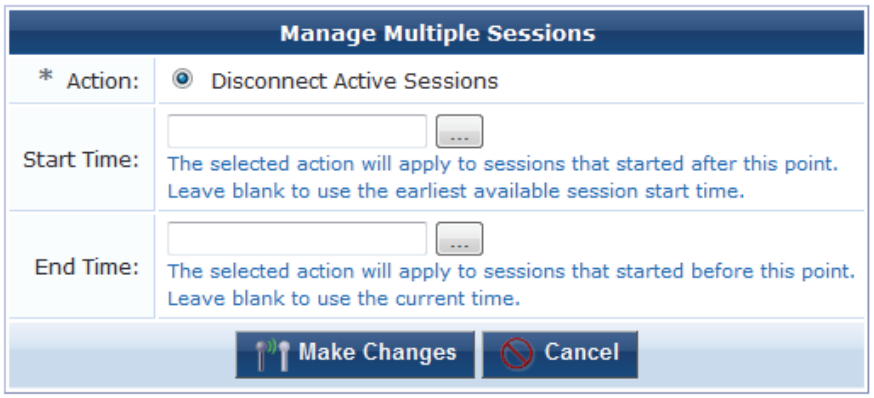
Operator	Meaning	Additional Information
=	is equal to	You may search for multiple values when using the equality (=) or inequality !=) operators. To specify multiple values, list them separated by the pipe character (|). For example, specifying the filter "role_id=2|3, custom_field=Value" restricts the accounts displayed to those with role IDs 2 and 3 (Guest and Employee), and with the field named "custom_field" set to "Value".
!=	is not equal to	
>	is greater than	
>=	is greater than or equal to	
<	is less than	
<=	is less than or equal to	
~	matches the regular expression	
!~	does not match the regular expression	

To restore the default view, click the **Clear Filter** link.

Click the **Apply Filter** button to save your changes and update the view, or click the **Reset** button to remove the filter and return to the default view.

Disconnecting Multiple Active Sessions

To disconnect multiple sessions, click the **Manage Multiple** tab. The Manage Multiple Sessions form opens.



- To close all active sessions, leave the **Start Time** and **End Time** fields empty and click **Make Changes**. All active sessions are closed and are removed from the Active Sessions list.

You can specify sessions in a time range.

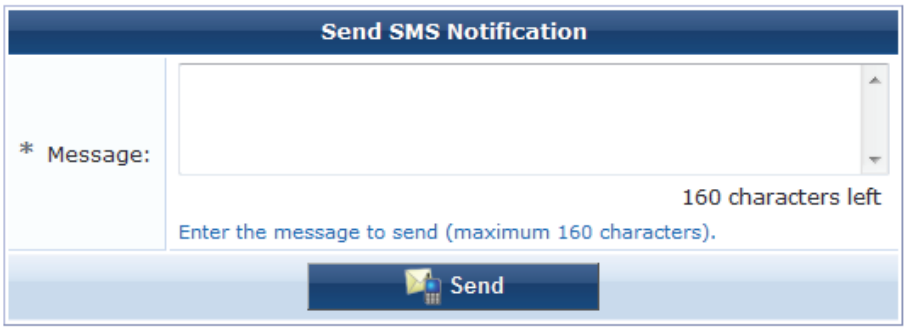
1. To close all sessions that started after a particular time, click the button in the **Start Time** row. The calendar picker opens. Use the calendar to specify the year, month, and day, and click the numbers in the **Time** fields to increment the hours and minutes. All sessions that started after the specified date and time will be disconnected.
2. To close all sessions that started before a particular time, click the button in the **End Time** row. The calendar picker opens. Use the calendar to specify the year, month, and day, and click the numbers in the **Time** fields to increment the hours and minutes. All sessions that started before the specified date and time will be disconnected.
3. Click Make Changes. The specified sessions are closed and are removed from the Active Sessions list.

Sending Multiple SMS Alerts

The SMS tab on the Active Sessions page lets you send an SMS alert message to all active sessions that have a valid phone number. An SMS alert during an active session can be used to send a group of visitors information you might want them to have immediately—for example, a special offer that will only be available for an hour, a change in a meeting's schedule or location, or a public safety announcement.

To create an SMS message:

1. Click the **SMS** tab on the Active Sessions page. The Send SMS Notification form opens.



2. Use the filter to specify the group of addresses that should receive the message. See "Filtering the List of Active Sessions". Only accounts with valid phone numbers can be sent SMS alerts.

3. Enter the message in the **Message** text box. Messages may contain up to 160 characters.
4. Click **Send**.

About SMS Guest Account Receipts

You can send SMS receipts for guest accounts that are created using either sponsored guest access or self-provisioned guest access. This is convenient in situations where the visitor may not be physically present to receive a printed receipt.

ClearPass Guest may be configured to automatically send SMS receipts to visitors, or to send receipts only on demand.

To manually send an SMS receipt:

1. Navigate to the **Guest > List Accounts** and click to expand the row of the guest to whom you want to send a receipt.
2. Click **Print** to display the Account Details view, then click the **Send SMS receipt** link. The SMS Reciept form opens. Use the fields on this form to enter the service to use, the recipient's mobile phone number, the mobile carrier, and the message text.

For more information on SMS services, see "SMS Services" in the "Administration" chapter.

Chapter 4

Onboard

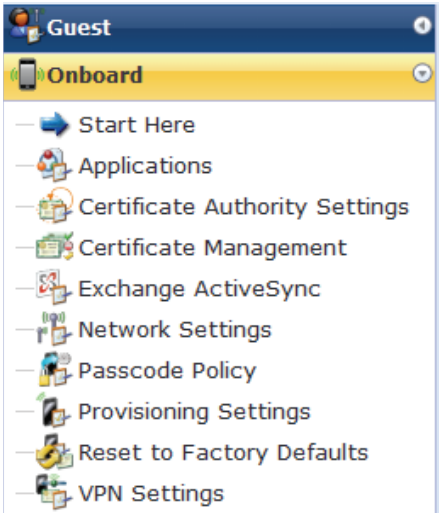
Onboarding is the process of preparing a device for use on an enterprise network by creating the appropriate access credentials and setting up the network connection parameters. ClearPass Onboard automates 802.1X configuration and provisioning for "bring your own device" (BYOD) and IT-managed devices—Windows, Mac OS X, iOS and Android—across wired, wireless, and VPNs.

ClearPass Onboard includes the following key features:

- Automatic configuration of network settings for wired and wireless endpoints.
- Provisioning of unique device credentials for BYOD and IT-managed devices.
- Support for Windows, Mac OS X, iOS and Android devices.
- Enables the revocation of unique credentials on a specific user's device.
- Leverages ClearPass profiling to identify device type, manufacturer, and model.

Accessing Onboard

To access ClearPass Onboard's device provisioning features, click the **Onboard** link in the left navigation.



About ClearPass Onboard

This section provides important information about ClearPass Onboard.

Onboard Deployment Checklist

Table 12 lists planning, configuration, and testing procedures. Use this checklist to complete your Onboard deployment.

Onboard events are stored in the Application Log for seven days by default. After seven days, significant runtime events are listed in the Audit Viewer in ClearPass Policy Manager's Monitoring module. Onboard events that are listed include:

- Changing the CA certificate
- Issuing a new certificate
- Signing a certificate signing request
- Revoking a certificate
- Deleting a certificate
- Importing a trusted certificate
- Uploading a code-signing or other certificate

Table 12 *Onboard Deployment Checklist*

Deployment Step	Reference
Planning and Preparation	
Review the Onboard feature list to identify the major areas of interest for your deployment.	"Onboard Feature List"
Review the list of platforms supported by Onboard, and identify the platforms of interest for your deployment.	"Supported Platforms"
Review the Onboard public key infrastructure, and identify any certificate authorities that will be needed during the deployment.	"Public Key Infrastructure for Onboard"
Review the network requirements and the network architecture diagrams to determine how and where to deploy the Onboard solution.	Refer to the ClearPass Policy Manager documentation, and "Network Architecture for Onboard" in this chapter
Configuration	
Configure the hostname and networking properties of the Onboard provisioning server. • DNS is required for SSL. • Ensure that hostname resolution will work for devices being provisioned.	Refer to the ClearPass Policy Manager documentation
Configure SSL certificate for the Onboard provisioning server. A commercial SSL certificate is required to enable secure device provisioning for iOS devices.	Refer to the ClearPass Policy Manager documentation
Configure the Onboard certificate authority. • Decide whether to use the Root CA or Intermediate CA mode of operation. Create the certificate for the certificate authority.	"Configuring the Certificate Authority "

Table 12 *Onboard Deployment Checklist (Continued)*

Deployment Step	Reference
Configure the data retention policy for the certificate authority.	"Configuring Data Retention Policy for Certificates"
Configure device provisioning settings. • Select certificate options for device provisioning. Select which device types should be supported.	"Configuring Provisioning Settings"
Configure network settings for device provisioning. • Set network properties. • Upload 802.1X server certificates. Set device-specific networking settings.	"Configuring Network Settings for Device Provisioning"
Configure networking equipment for non-provisioned devices. • Set authentication for the provisioning SSID, if required. Ensure the captive portal redirects non-provisioned devices to the device provisioning page.	"Network Requirements for Onboard"
Configure networking equipment to authenticate provisioned devices. • Ensure 802.1X authentication methods and trust settings are configured correctly for all EAP types that are required. Configure OCSP or CRL on the authentication server to check for client certificate validity.	"Network Requirements for Onboard"
Configure the user interface for device provisioning. • Set display options for iOS devices. • Set user interface options for other Onboard devices. Setup the device provisioning Web login page.	"Configuring the User Interface for Device Provisioning"
Testing and Verification	
Test device provisioning. • Verify that each type of device can be provisioned successfully. Verify that each type of device can join the provisioned network and is authenticated successfully.	
Test device revocation. • Revoke a device's certificate. • Verify that the device is no longer able to authenticate. Verify that re-provisioning the device fails.	

Onboard Feature List

The following features are available in ClearPass Onboard.

Table 13 *Onboard Features*

Feature	Uses
Automatic configuration of network settings for wired and wireless endpoints.	• Configure wired networks using 802.1X • Configure Wi-Fi networks using either 802.1X or pre-shared key (PSK) • Configure trusted server certificates for 802.1X • Configure Windows-specific networking settings • Configure HTTP proxy settings for client devices (Android, OS X only)
Secure provisioning of unique device credentials for BYOD and IT-managed devices.	• Configure EAP-TLS and PEAP-MSCHAPv2 without user interaction • Revoke unique device credentials to prevent network access
Support for Windows, Mac OS X, iOS, and Android devices.	• Leverage ClearPass Profiling to identify device type, manufacturer, and model • Control the user interface displayed during device provisioning
Certificate authority enables the creation and revocation of unique credentials on a specific user's device.	• Root and intermediate CA modes of operation • Supports SCEP enrollment of certificates • Supports CRL generation to list revoked certificates • Supports OCSP responder to query for certificate status • Approve certificate signing request • Reject certificate signing request • Sign certificate from uploaded certificate signing request (CSR) • Issue certificate • Revoke certificate • Display certificates • Export certificate • Renew root certificate
Provision additional settings specific to iOS devices	• Exchange ActiveSync • Passcode policy • VPN settings

Supported Platforms

The platforms supported by ClearPass Onboard and the version requirements for each platform are summarized in the following table.

Table 14 *Platforms Supported by ClearPass Onboard*

Platform	Example Devices	Version Required for Onboard Support	Notes
Apple iOS	iPhone iPad iPod Touch	iOS 4 iOS 5	1, 3
Apple Mac OS X	MacBook Pro MacBook Air	Mac OS X 10.8 “Mountain Lion” Mac OS X 10.7 “Lion”	1
		Mac OS X 10.6 “Snow Leopard” Mac OS X 10.5 “Leopard”	2
Android	Samsung Galaxy S Samsung Galaxy Tab Motorola Droid	Android 2.2 (or higher)	2
Microsoft Windows	Laptop Netbook	Windows XP with Service Pack 3 Windows Vista with Service Pack 3 Windows 7	2

Note 1: Uses the “Over-the-air provisioning” method.
Note 2: Uses the “Onboard provisioning” method.
Note 3: Onboard may also be used to provision VPN settings, Exchange ActiveSync settings, and passcode policy on these devices.

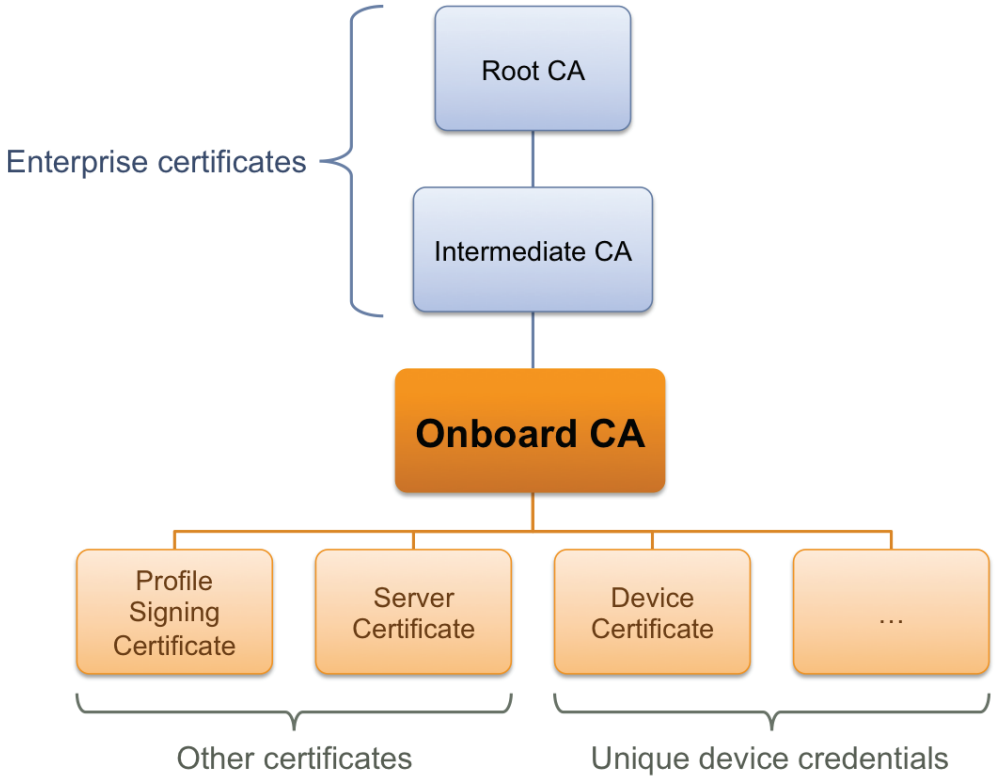
Public Key Infrastructure for Onboard

During the device provisioning process, one or more digital certificates are issued to the device. These are used as the unique credentials for a device. To issue the certificate, ClearPass Onboard must operate as a certificate authority. The following sections explain how the certificate authority works, and which certificates are used in this process.

Certificate Hierarchy

In a public key infrastructure (PKI) system, certificates are related to each other in a tree-like structure.

Figure 10 *Relationship of Certificates in the Onboard Public Key Infrastructure*



The root certificate authority (CA) is typically an enterprise certificate authority, with one or more intermediate CAs used to issue certificates within the enterprise.

Onboard may operate as a root CA directly, or as an intermediate CA. See "Configuring the Certificate Authority ".

The Onboard CA issues certificates for several purposes:

- The **Profile Signing Certificate** is used to digitally sign configuration profiles that are sent to iOS devices.
  - The identity information in the profile signing certificate is displayed during device provisioning.
- One or more **Server Certificates** may be issued for various reasons – typically, for an enterprise's authentication server.
  - The identity information in the server certificate may be displayed during network authentication.
- One or more **Device Certificates** may be issued – typically, one or two per provisioned device.
  - The identity information in the device certificate uniquely identifies the device and the user that provisioned the device.

You do not need to manually create the profile signing certificate; it is created when it is needed See "Configuring Provisioning Settings for iOS and OS X"to control the contents of this certificate.

You may revoke the profile signing certificate; it will be recreated when it is needed for the next device provisioning attempt.

Revoking Unique Device Credentials

Because each provisioned device uses unique credentials to access the network, it is possible to disable network access for an individual device. This offers a greater degree of control than traditional user-based authentication — disabling a user's account would impact all devices using those credentials.

To disable network access for a device, revoke the TLS client certificate provisioned to the device. See "Working with Certificates in the List".

Note: Revoking access for a device is only possible when using an enterprise network. Personal (PSK) networks do not support this capability.

Revoking Credentials to Prevent Network Access

Revoking a device's certificate will also prevent the device from being re-provisioned.

This is necessary to prevent the user from simply re-provisioning and obtaining a new certificate. To re-provision the device, the revoked certificate must be deleted.

If the device is provisioned with an EAP-TLS client certificate, revoking the certificate will cause the certificate authority to update the certificate's state. When the certificate is next used for authentication, it will be recognized as a revoked certificate and the device will be denied access.

Note: When using EAP-TLS authentication, you must configure your authentication server to use either OCSP or CRL to check the revocation status of a client certificate. OCSP is recommended as it offers a real-time status update for certificates.

If the device is provisioned with PEAP unique device credentials, revoking the certificate will automatically delete the unique username and password associated with the device. When this username is next used for authentication, it will not be recognized as valid and the device will be denied access.

Note: OCSP and CRL are not used when using PEAP unique device credentials. The Onboard server automatically updates the status of the username when the device's client certificate is revoked.

Re-Provisioning a Device

Because "bring your own" devices are not under the complete control of the network administrator, it is possible for unexpected configuration changes to occur on a provisioned device.

For example, the user may delete the configuration profile containing the settings for the provisioned network, instruct the device to forget the provisioned network settings, or reset the device to factory defaults and destroy all the configuration on the device.

When these events occur, the user will not be able to access the provisioned network and will need to re-provision their device.

The Onboard server detects a device that is being re-provisioned and prompts the user to take a suitable action (such as connecting to the appropriate network). If this is not possible, the user may choose to restart the provisioning process and re-provision the device.

Re-provisioning a device will reuse an existing TLS client certificate or unique device credentials, if these credentials are still valid.

If the TLS client certificate has expired then the device will be issued a new certificate. This enables re-provisioning to occur on a regular basis.

If the TLS client certificate has been revoked, then the device will not be permitted to re-provision. The revoked certificate must be deleted before the device is able to be provisioned.

Network Requirements for Onboard

For complete functionality to be achieved, ClearPass Onboard has certain requirements that must be met by the provisioning network and the provisioned network:

- The provisioning network must use a captive portal or other method to redirect a new device to the device provisioning page.
- The provisioning server (Onboard server) must have an SSL certificate that is trusted by devices that will be provisioned. In practice, this means a commercial SSL certificate is required.
- The provisioned network
- must support EAP-TLS and PEAP-MSCHAPv2 authentication methods.
- The provisioned network must support either OCSP or CRL checks to detect when a device has been revoked and deny access to the network.

Using Same SSID for Provisioning and Provisioned Networks

To configure a single SSID to support both provisioned and non-provisioned devices, use the following guidelines:

- Configure the network to use both PEAP and EAP-TLS authentication methods.
- When a user authenticates via PEAP with their domain credentials, place them into a provisioning role.
- The provisioning role should have limited network access and a captive portal that redirects users to the device provisioning page.
- When a user authenticates via PEAP with unique device credentials, place them into a provisioned role.
- When a user authenticates via EAP-TLS using an Onboard client certificate, place them into a provisioned role.

For provisioned devices, additional authorization steps can be taken after authentication has completed to determine the appropriate provisioned role.

Using Different SSID for Provisioning and Provisioned Networks

To configure dual SSIDs to support provisioned devices on one network, and non-provisioned devices on a separate network, use the following guidelines:

- Configure the provisioning SSID to use PEAP, or another suitable authentication method.
- When a user connects to the provisioning SSID, place them into a provisioning role.
  - The provisioning role should have limited network access and a captive portal that redirects users to the device provisioning page.
- When a user connects to the provisioned SSID, authenticate based on the type of credentials presented.
  - For PEAP authentication with unique device credentials, place them into a provisioned role.
  - For EAP-TLS authentication using an Onboard client certificate, place them into the provisioned role.
  - In all other cases, deny access.

As for the single-SSID case, additional authorization steps may be taken after authentication has completed to determine the appropriate provisioned role.

Configuring Online Certificate Status Protocol

Onboard supports the Online Certificate Status Protocol (OCSP) to provide a real-time check on the validity of a certificate.

To configure OCSP for your network, you will need to provide the URL of an OCSP service to your network equipment. This URL can be constructed by using the relative path mdps_ocsp.php/1.

For example, if the Onboard server's hostname is onboard.example.com, the OCSP URL to use is: http://onboard.example.com/mdps_ocsp.php/1.

Note: OCSP does not require the use of HTTPS and can be configured to use HTTP.

Configuring Certificate Revocation List (CRL)

Onboard supports generating a Certificate Revocation List (CRL) that lists the serial numbers of certificates that have been revoked.

To configure a CRL, you will need to provide its URL to your network equipment. This URL can be constructed by using the relative path mdps_crl.php?id=1.

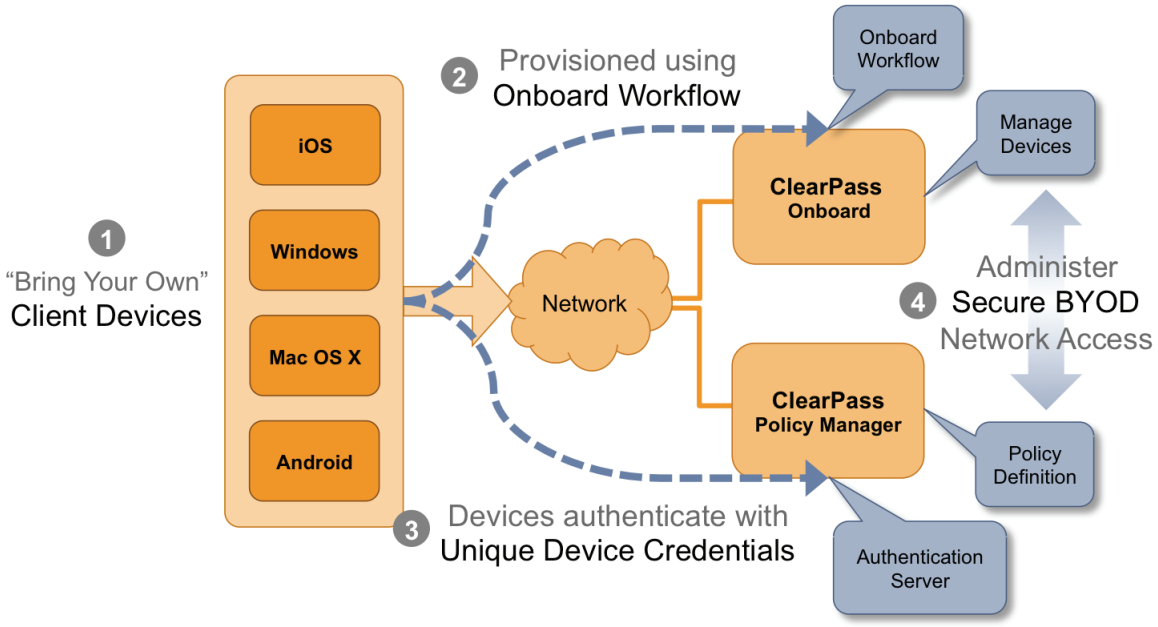
For example, if the Onboard server's hostname is onboard.example.com, the location of the CRL is: http://onboard.example.com/mdps_crl.php?id=1.

Note: A certificate revocation list does not require the use of HTTPS and can be configured to use HTTP.

Network Architecture for Onboard

The high-level network architecture for the Onboard solution is shown in the following figure.

Figure 11 *ClearPass Onboard Network Architecture*



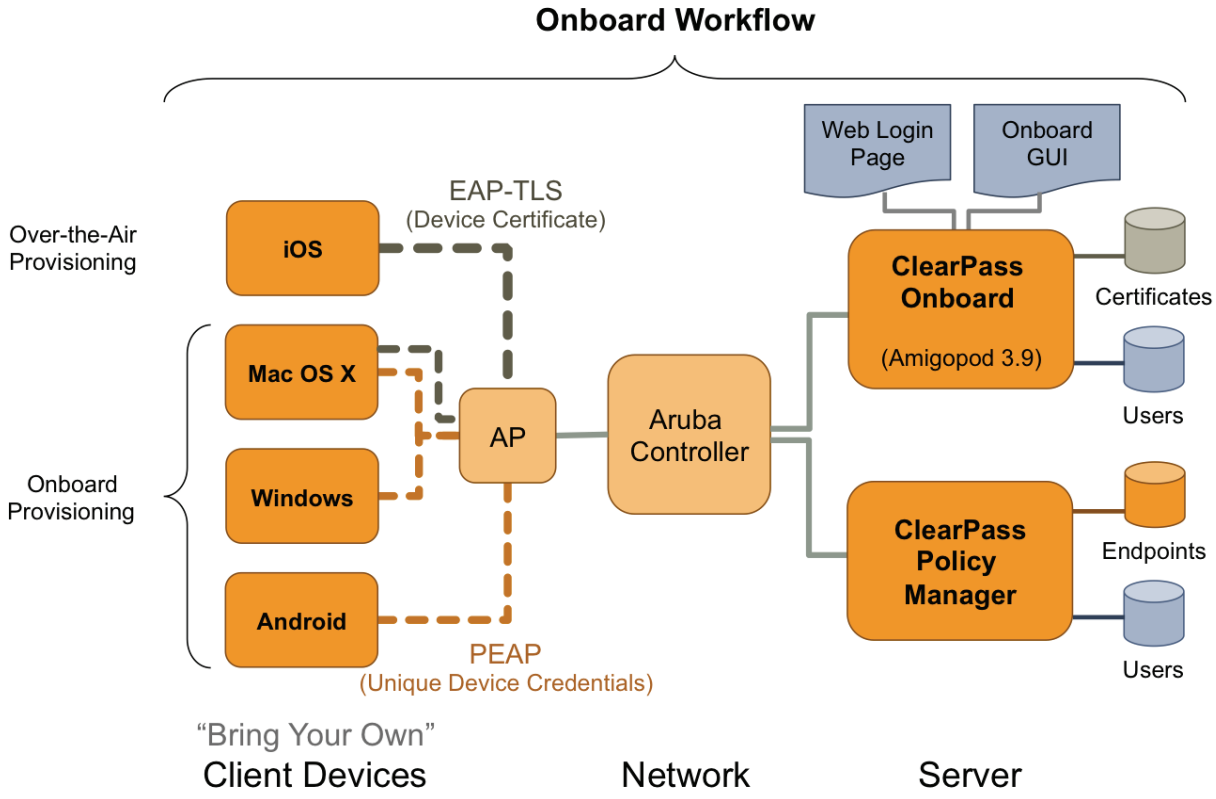
The sequence of events shown in Figure 11 is:

1. Users bring their own device to the enterprise.
2. The ClearPass Onboard workflow is used to provision the user's device securely and with a minimum of user interaction.
3. Once provisioned, the device re-authenticates to the network using a set of unique device credentials. These credentials uniquely identify the device and user and enable management of provisioned devices.

4. Administrators can configure all aspects of the provisioning workflow – including the devices that have been provisioned, policies to apply to devices and the overall user experience for BYOD.

A more detailed view of the network architecture is shown in Figure 12. This diagram shows different types of client devices using the Onboard workflow to gain access to the network. Some of the components that may be configured by the network administrator are also shown.

Figure 12 *Detailed View of the ClearPass Onboard Network Architecture*



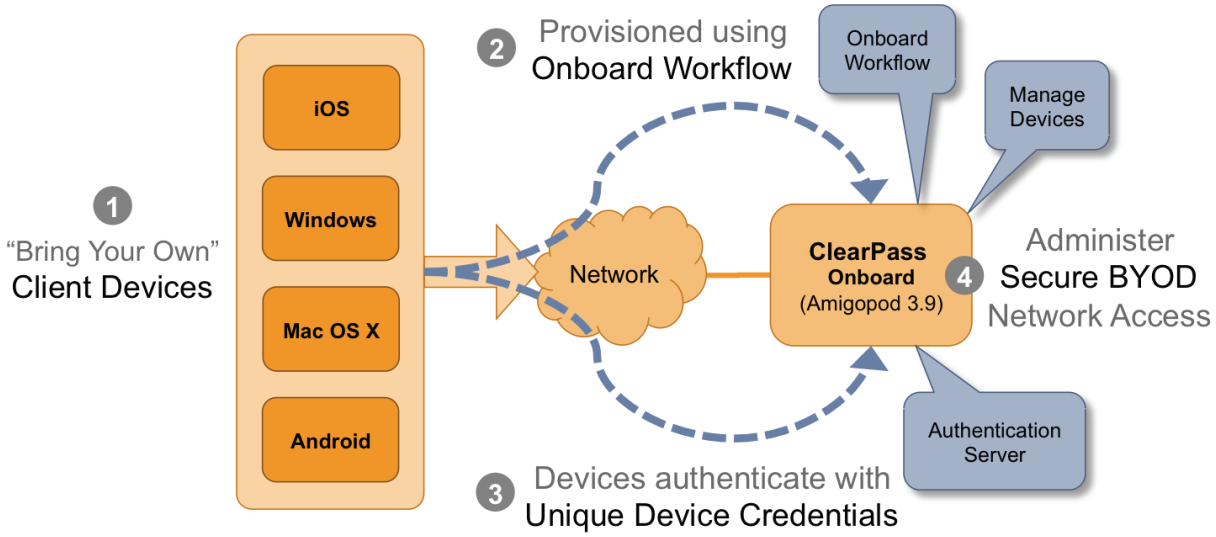
The components shown in Figure 12 are:

1. Users bring different kinds of client device with them. Onboard supports "smart devices" that use the iOS or Android operating systems, such as smartphones and personal tablets. Onboard also supports the most common versions of Windows and Mac OS X operating systems found on desktop computers, laptops and netbooks.
2. The Onboard workflow is used to provision the user's device securely and with a minimum of user interaction. The provisioning method used depends on the type of device.
   a. Newer versions of Mac OS X (10.7 and later) and iOS devices use the "over-the-air" provisioning method.
   b. Other supported platforms use the "Onboard provisioning" method.
3. Once provisioned, client devices use a secure authentication method based on 802.1X and the capabilities best supported by the device.
   a. The unique device credentials issued during provisioning are in the form of an EAP-TLS client certificate for iOS devices and OS X (10.7+) devices.
   b. Other supported devices are also issued a client certificate, but will use the PEAP-MSCHAPv2 authentication method with a unique username and strong password.
4. Administrators can manage all Onboard devices using the certificate issued to that device.

Network Architecture for Onboard when Using ClearPass Guest

ClearPass Guest supports the provisioning, authentication, and management aspects of the complete Onboard solution. Figure 13 shows the high-level network architecture for the Onboard solution when using ClearPass Guest as the provisioning and authentication server.

Figure 13 *ClearPass Onboard Network Architecture when Using ClearPass Guest*



The user experience for device provisioning is the same in Figure 13 and Figure 11, however there are implementation differences between these approaches:

- When using the ClearPass Guest RADIUS server for provisioning and authentication, EAP-TLS and PEAP authentication must be configured.

  Navigate to **RADIUS > Authentication > EAP & 802.1X** to configure a server certificate and the appropriate EAP types for the ClearPass Guest RADIUS server.

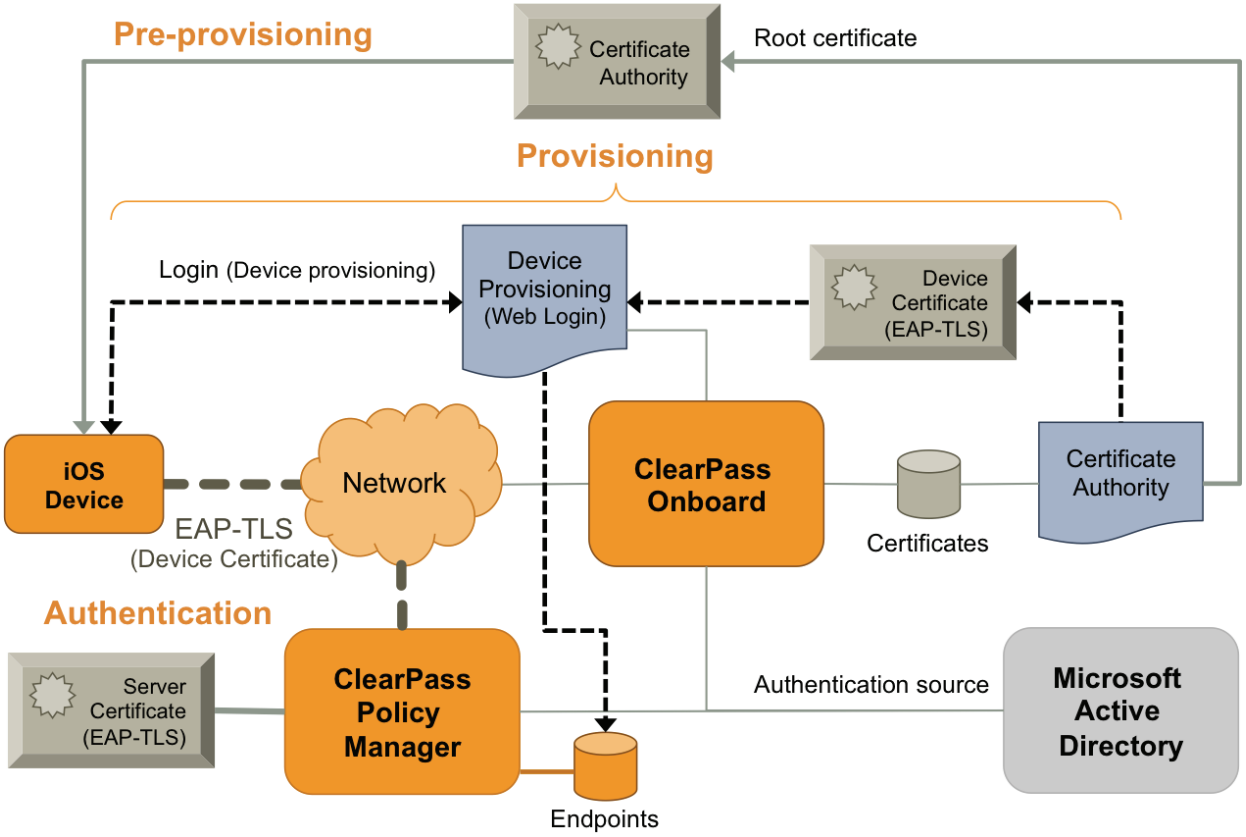
- ClearPass Policy Manager supports a rich policy definition framework. If you have complex policies to enforce, multiple authentication or authorization sources that define user accounts, or you need features beyond those available in the ClearPass Guest RADIUS server, you should deploy Policy Manager for authentication.

The ClearPass Onboard Process

Devices Supporting Over-the-Air Provisioning

ClearPass Onboard supports secure device provisioning for iOS 4, iOS 5, and recent versions of Mac OS X (10.7 "Lion" and later). These are collectively referred to as "iOS devices". The Onboard process for iOS devices is shown in Figure 14.

Figure 14 *ClearPass Onboard Process for iOS Devices*

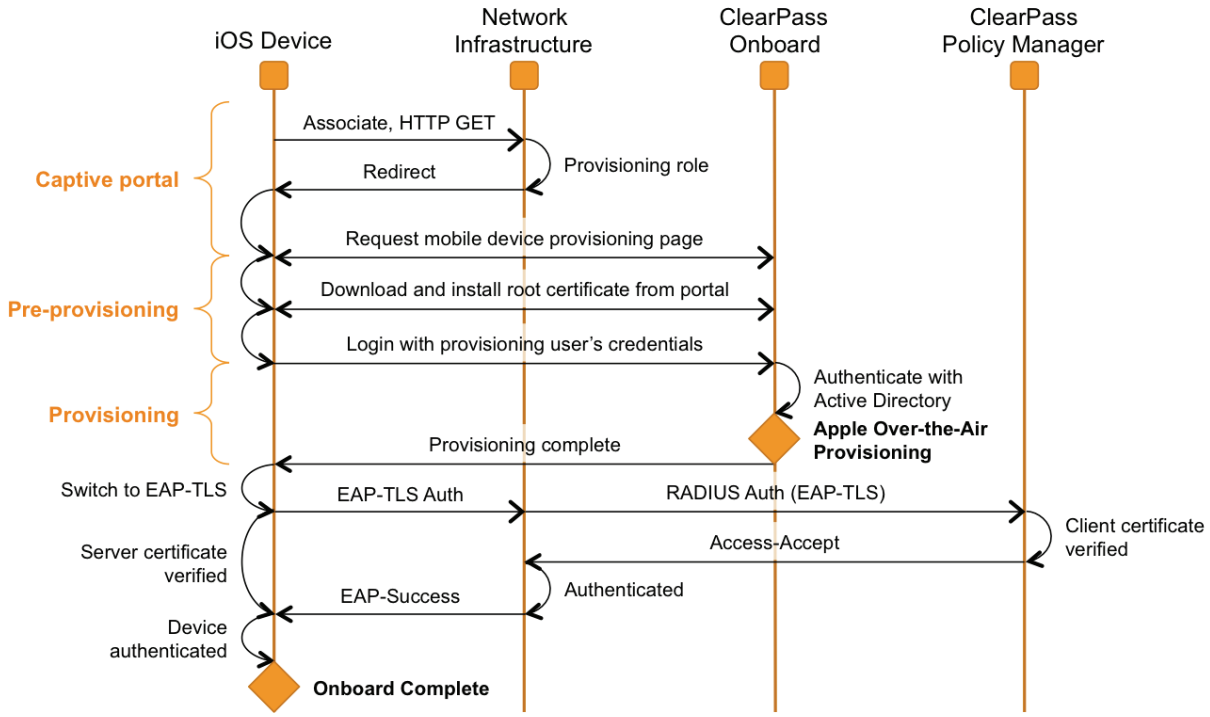


The Onboard process is divided into three stages:

1. **Pre-provisioning**. The enterprise's root certificate is installed on the iOS device.
2. **Provisioning**. The user is authenticated at the device provisioning page and then provisions their device with the Onboard server. The device is configured with appropriate network settings and a device-specific certificate.
3. **Authentication**. Once configuration is complete, the user switches to the secure network and is authenticated using an EAP-TLS client certificate.

A sequence diagram showing the interactions between each component of this workflow is shown in Figure 15.

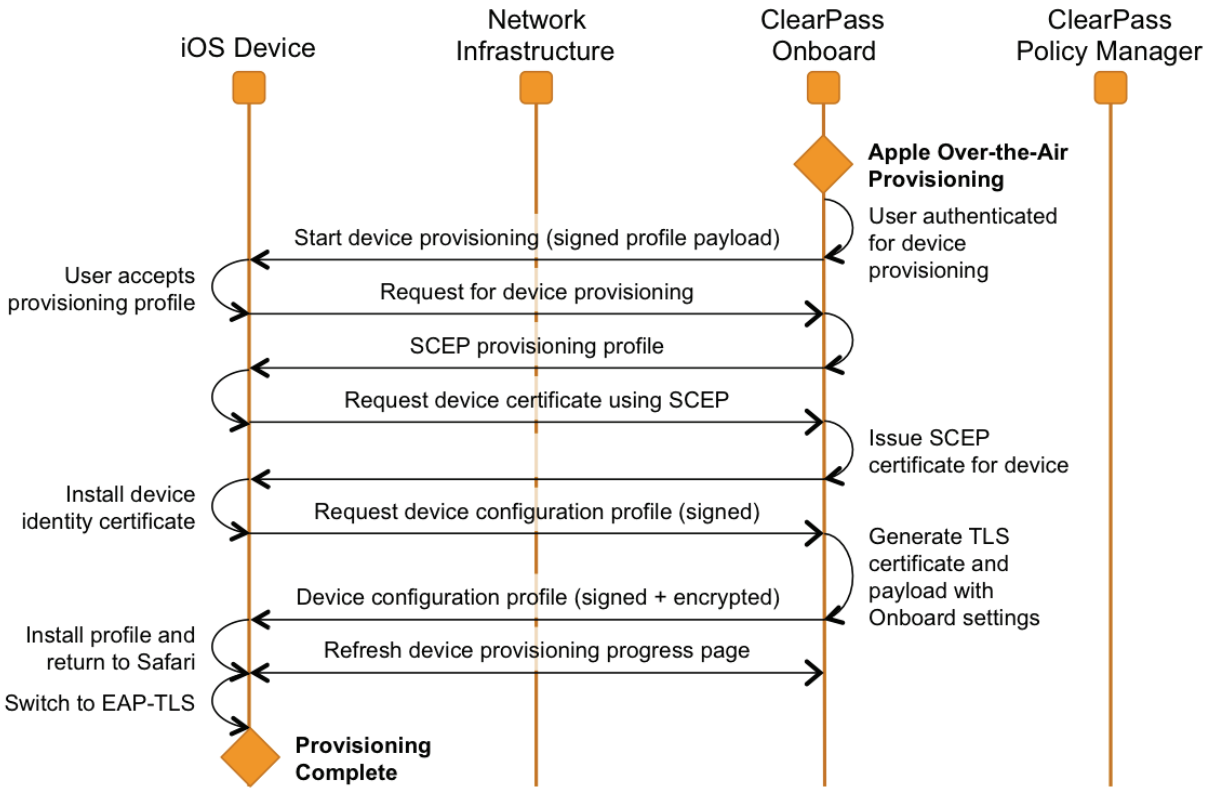
Figure 15 *Sequence Diagram for the Onboard Workflow on iOS Platform*



1. When a BYOD device first joins the provisioning network it does not have a set of unique device credentials. This will trigger the captive portal for that device, which brings the user to the mobile device provisioning page.
2. A link on the mobile device provisioning page prompts the user to install the enterprise's root certificate. Installing the enterprise's root certificate enables the user to establish the authenticity of the provisioning server during device provisioning.
3. The user then authenticates with their provisioning credentials – these are typically the user's enterprise credentials from Active Directory. If the user is authorized to provision a mobile device, the over-the-air provisioning workflow is then triggered (see Figure 16, below).
4. After provisioning has completed, the device switches to EAP-TLS authentication using the newly provisioned client certificate. Mutual authentication is performed (the authentication server verifies the client certificate, and the client verifies the authentication server's certificate).
5. The device is now onboard and is able to securely access the provisioned network.

Over-the-air provisioning is used to securely provision a device and configure it with network settings. Figure 16 shows a sequence diagram that explains the steps involved in this workflow.

Figure 16 *Over-the-Air Provisioning Workflow for iOS Platform*

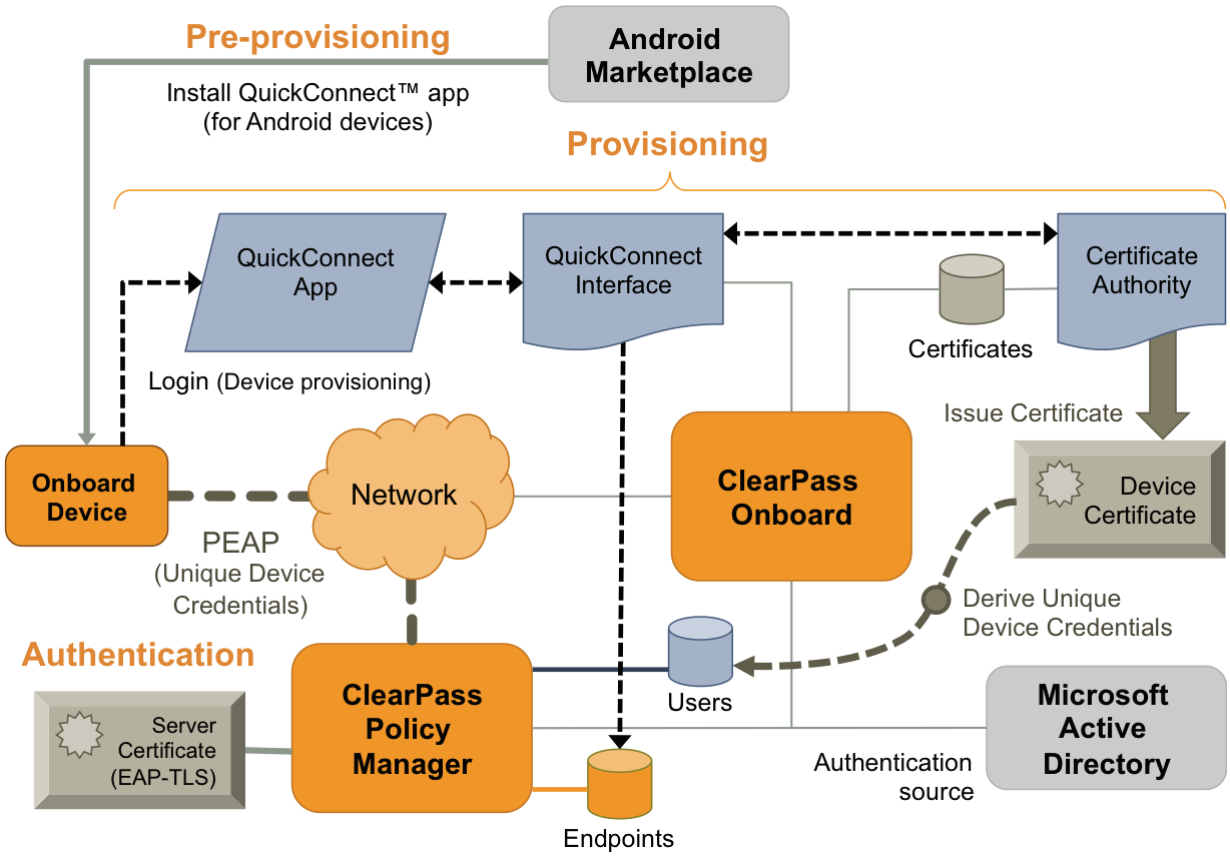


1. The only user interaction required is to accept the provisioning profile. This profile is signed by the Onboard server, so that the user can be assured of its authenticity.
2. An iOS device will have two certificates after over-the-air provisioning is complete:
   a. A Simple Certificate Enrollment Protocol (SCEP) certificate is issued to the device during the provisioning process. This certificate identifies the device uniquely, and is used to encrypt the device configuration profile so that only this device can read its unique settings.
   b. A Transport Layer Security (TLS) client certificate is issued to the device. This certificate identifies the device and the user that provisioned the device. It is used as the device's network identity during EAP-TLS authentication.

Devices Supporting Onboard Provisioning

ClearPass Onboard supports secure device provisioning for Microsoft Windows XP (service pack 3 and later), Microsoft Windows Vista, Microsoft Windows 7, Apple Mac OS X 10.5 and 10.6, and Android devices (smartphones and tablets). These are collectively referred to as “Onboard-capable devices”. The Onboard process for these devices is shown in Figure 17.

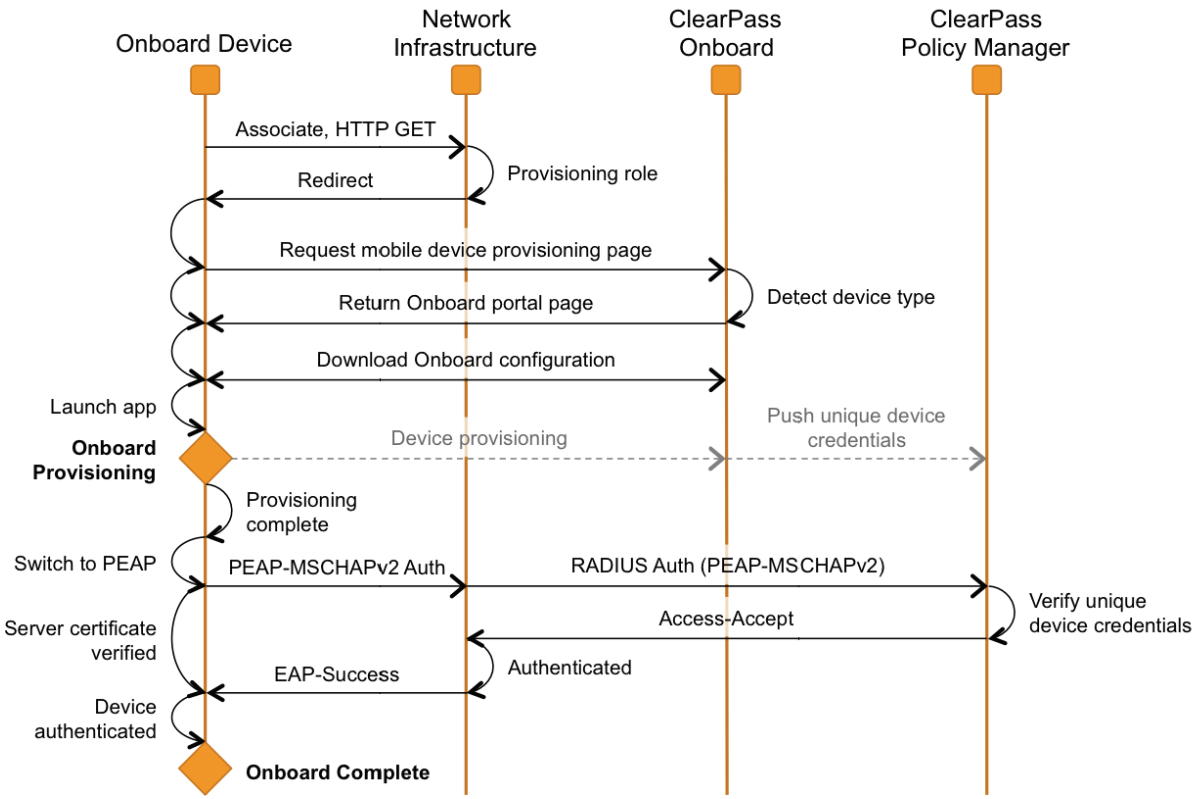
Figure 17 *ClearPass Onboard Process for Onboard-Capable Devices*



The Onboard process is divided into three stages:

1. **Pre-provisioning.** This step is only required for Android devices; the Aruba Networks QuickConnect app must be installed for secure provisioning of the device.
2. **Provisioning.** The device provisioning page detects the device type and downloads or starts the QuickConnect app. The app authenticates the user and then provisions their device with the Onboard server. The device is configured with appropriate network settings and credentials that are unique to the device. See Figure 18 for details.
3. **Authentication**. Once configuration is complete, the user switches to the secure network and is authenticated using PEAP-MSCHAPv2 unique device credentials.

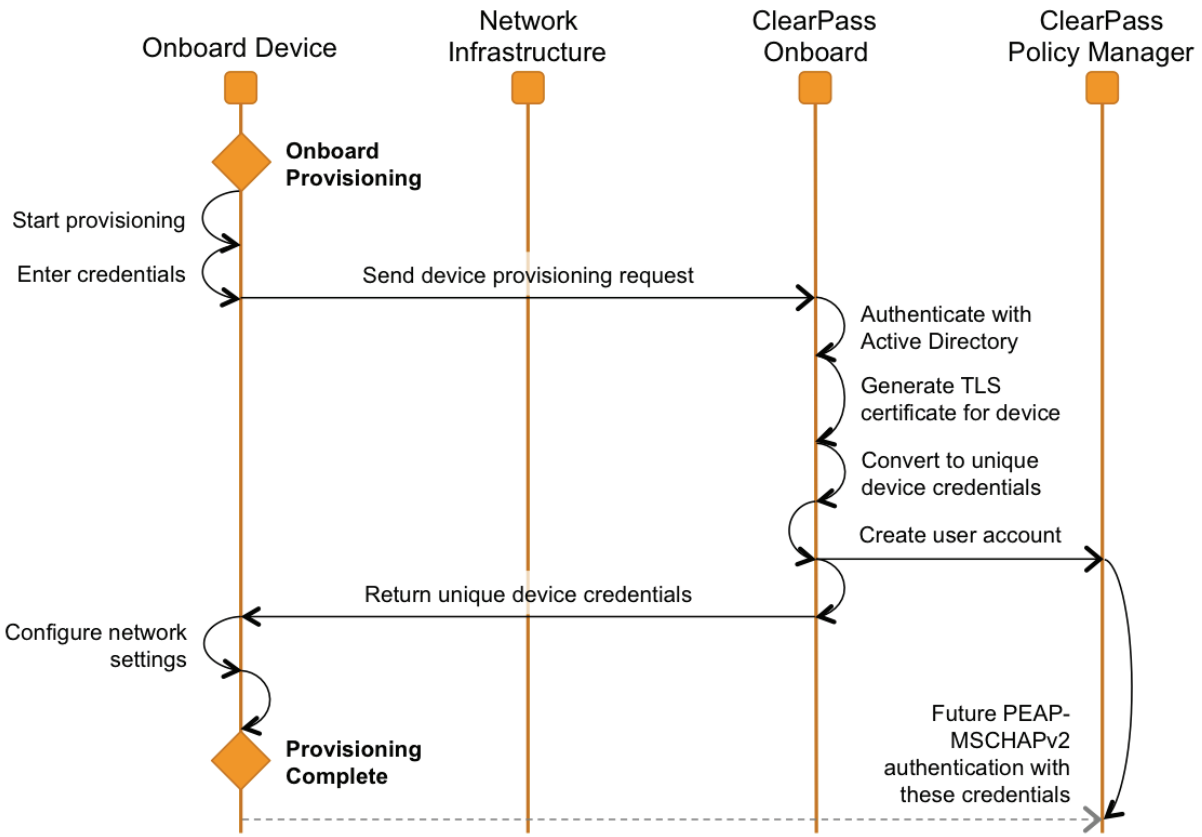
Figure 18 *Sequence Diagram for the Onboard Workflow on Android Platform*



1. When a BYOD device first joins the network it does not have a set of unique device credentials. This will trigger the captive portal for that device, which brings the user to the mobile device provisioning page.
2. The Onboard portal is displayed. The user's device type is detected, and a link is displayed depending on the device type:
   a. For Android devices, the link is to a file containing the Onboard configuration settings; downloading this file will launch the QuickConnect app on the device.
   b. For Windows and Mac, the link is to a executable file appropriate for that operating system that includes both the QuickConnect app and the Onboard configuration settings.
3. The QuickConnect app uses the Onboard provisioning workflow to authenticate the user and provision their device with the Onboard server. The device is configured with appropriate network settings and credentials that are unique to the device.
4. After provisioning has completed, the app switches the device to PEAP authentication using the newly provisioned unique device credentials. Mutual authentication is performed (the authentication server verifies the client's username and password, and the client verifies the authentication server's certificate).
5. The device is now onboard and is able to securely access the network.

The Onboard provisioning workflow is used to securely provision a device and configure it with network settings. Figure 19 shows a sequence diagram that explains the steps involved in this workflow.

Figure 19 *Onboard Provisioning Workflow in the QuickConnect App*

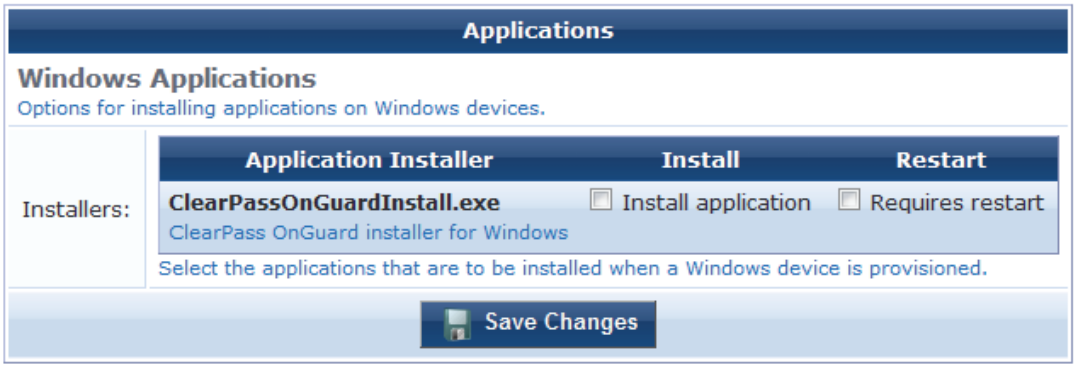


Managing Provisioned Applications

The Applications form lets you mark individual applications for installation during device provisioning, and specify whether they should be restarted when the device is provisioned. If restart is selected, you can specify whether the restart should take effect when the installation is complete or at a later time.

To manage your applications:

1. Go to **Onboard > Applications**. The Applications form opens.



2. To upload applications, click the **Content Manager** link above the form.
3. To select applications to install, mark their check boxes, then click **Save Changes**.

Configuring the User Interface for Device Provisioning

The user interface for device provisioning can be customized in three different ways:

- Customizing the Web login page used for device provisioning.

  All devices will reach the device provisioning Web login page as the first step of the provisioning process.See "Customizing the Device Provisioning Web Login Page" to make changes to the content or formatting of this page.

- Customizing the properties of the device provisioning profile for iOS and OS X devices.

  After starting the provisioning process, users of iOS and OS X are prompted to accept a configuration profile. See "Configuring Provisioning Settings for iOS and OS X" to make changes to the content of this profile.

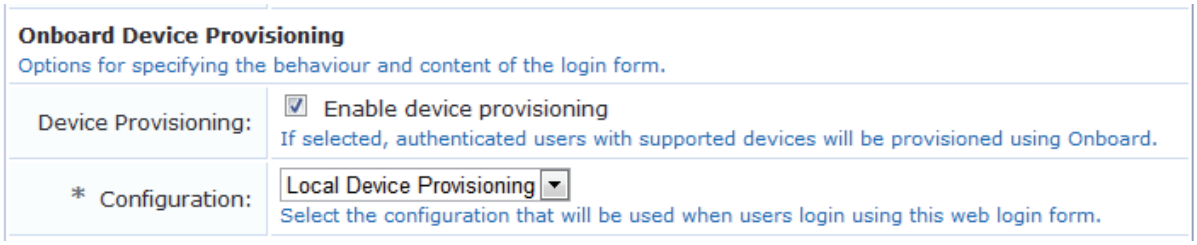
- Customizing the user interface of the QuickConnect app for Windows, Mac OS X and Android devices.

  The provisioning process for Windows, Mac OS X and Android devices uses a separate app, which has a customizable user interface. See "Configuring Options for Legacy OS X, Windows, and Android Devices" to make changes to the user interface.

Customizing the Device Provisioning Web Login Page

Onboard creates a default Web login page that is used to start the device provisioning process.

To edit this page, navigate to **Configuration > Web Logins**, click to expand the **Onboard Provisioning** row in the list, and then click **Edit**. The RADIUS Web Logins Editor form opens. Scroll to the **Onboard Device Provisioning** rows of the form.



The Onboard-specific settings required for a device provisioning page are described below:

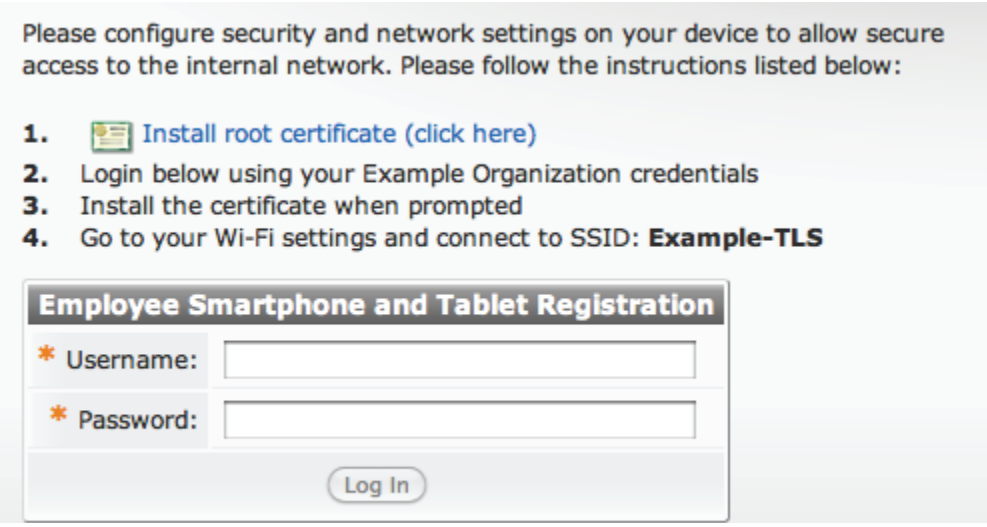
Mark the **Enable device provisioning** check box to activate the Onboard features for this Web login page.

Note: If this check box is not marked, device provisioning will be inoperative.

Select the appropriate Onboard configuration from the **Configuration** drop-down list.

To modify the instructions provided to users on the device provisioning page, edit the contents of the **Header HTML** text area.

The default instructions are displayed to the user as:



This corresponds to the following text prepopulated in the Header HTML text area:

```
<p>
Please configure security and network settings on your device to allow secure<br>
access to the internal network. Please follow the instructions listed below:<br>
<br>
<strong>1.</strong>    {nwa_iconlink icon="images/icon-
certificate22.png" text="Install root certificate (click here)"}{nwa_mdps_config
name=root_cert}{/nwa_iconlink}<br>
<strong>2.</strong>    Login below using your {nwa_mdps_config
name=organization_name} credentials<br>
<strong>3.</strong>    Install the certificate when prompted<br>
<strong>4.</strong>    Go to your Wi-Fi settings and connect to
SSID: <strong>{nwa_mdps_config name=wifi_ssid}</strong>
<br>
</p>
```

Using the {nwa_mdps_config} Template Function

Certain properties can be extracted from the Onboard configuration and used in the device provisioning page.

To obtain these properties, use the {nwa_mdps_config} Smarty template function. The “name” parameter specifies which property should be returned, as described in Table 15.

Table 15 *Properties Available with the* `(nwa_mdps_config)` *Smarty Template Function*

Name	Description

root_cert	URL of the Onboard certificate authority’s root certificate.
	Browsing to this URL will install the root certificate on the device, which is required as part of the pre-provisioning step.
	Example:
	`<a href="{nwa_mdps_config name=root_cert}">`
	`  Install Onboard root certificate`
	`</a>`

wifi_ssid	Name of the wireless network. See “Configuring Basic Network Access Settings”. Example: Connect to the network named `{nwa_mdps_config name=wifi_ssid}`

Table 15 *Properties Available with the* `(nwa_mdps_config)` *Smarty Template Function (Continued)*

Name	Description

organization_name	The organization name. See "Configuring Basic Provisioning Settings".
	Example:
	`<h2>`
	`  Welcome to {nwa_mdps_config name=organization_name}`
	`</h2>`

Configuring the Certificate Authority

To configure certificate authority settings, Navigate to **Onboard > Certificate Authority Settings**, or click the **Certificate Authority Settings** command link.

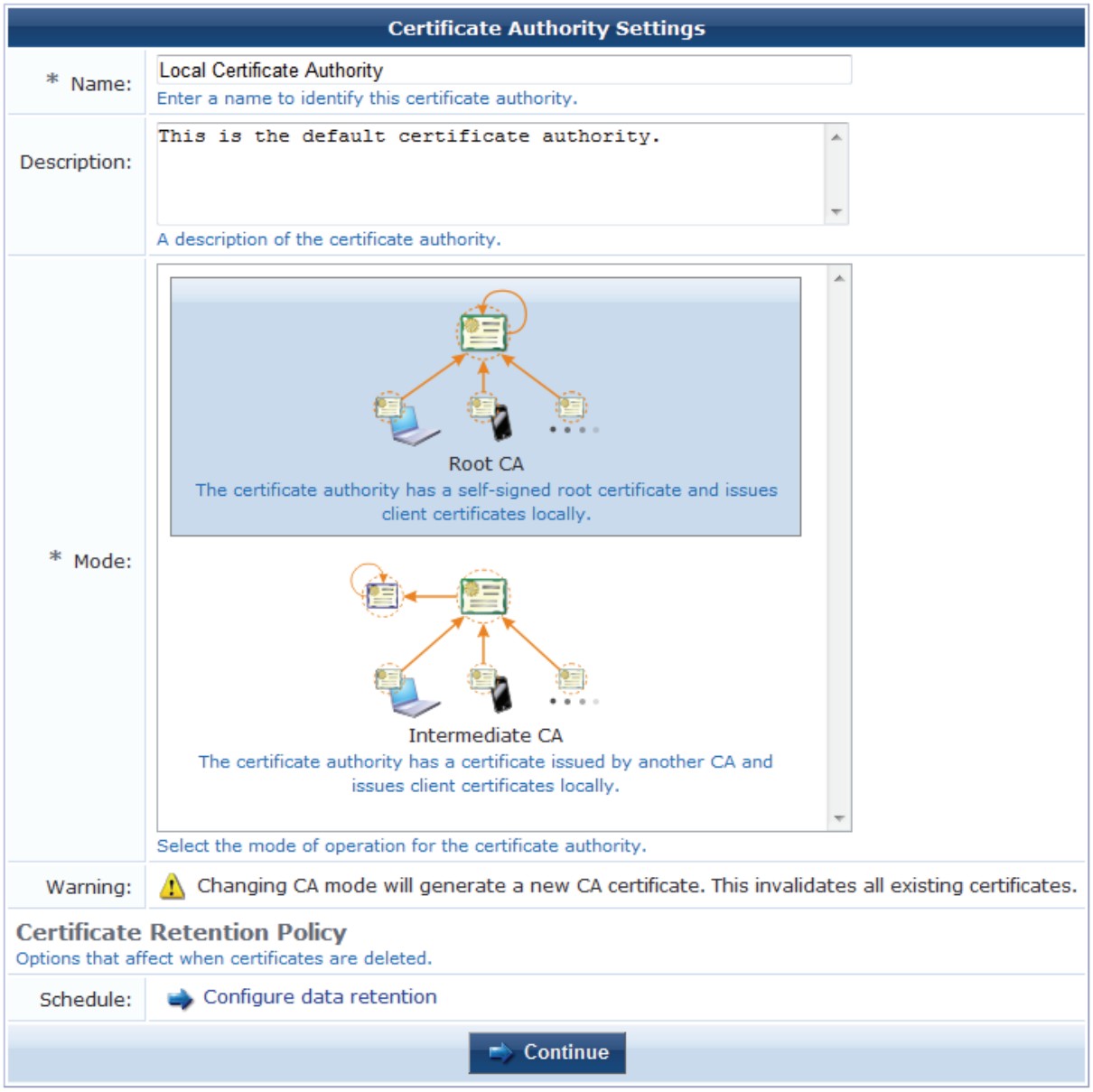
The Certificate Authority Settings form opens.

This page is used to configure the Onboard certificate authority and to perform maintenance tasks for the CA.:

- Set up a root or intermediate certificate authority (See "Setting Up the Certificate Authority")
- Determine the OCSP URL for the certificate authority
- View the trust chain for the certificate authority (See "Uploading Certificates for the Certificate Authority")
- Renew the certificate authority's certificate (See "Renewing the Certificate Authority's Certificate")
- Configure the data retention policy applied to certificates issued by the authority (See "Configuring Data Retention Policy for Certificates")

Setting Up the Certificate Authority

The Certificate Authority Settings form is used to set up the mode of operation for the certificate authority.



The **Name** and **Description** fields are used internally to identify this certificate authority for the network administrator. These values are never displayed to the user during device provisioning.

Select the appropriate mode for the certificate authority:

- **Root CA** – The Onboard certificate authority issues its own root certificate. The certificate authority issues client and server certificates using a local signing certificate, which is an intermediate CA that is subordinate to the root certificate. Use this option when you do not have an existing public-key infrastructure (PKI), or if you want to completely separate the certificates issued for Onboard devices from your existing PKI.

  Click the **Root CA** image in the **Mode** area, then click **Continue** to proceed to the second step. See "Setting Up a Root Certificate Authority".

- **Intermediate CA** – The Onboard certificate authority is issued a certificate by an external certificate authority. The Onboard certificate authority issues client and server certificates using this certificate. Use this option when you already have a public-key infrastructure (PKI), and would like to include the certificate issued for Onboard devices in that infrastructure.

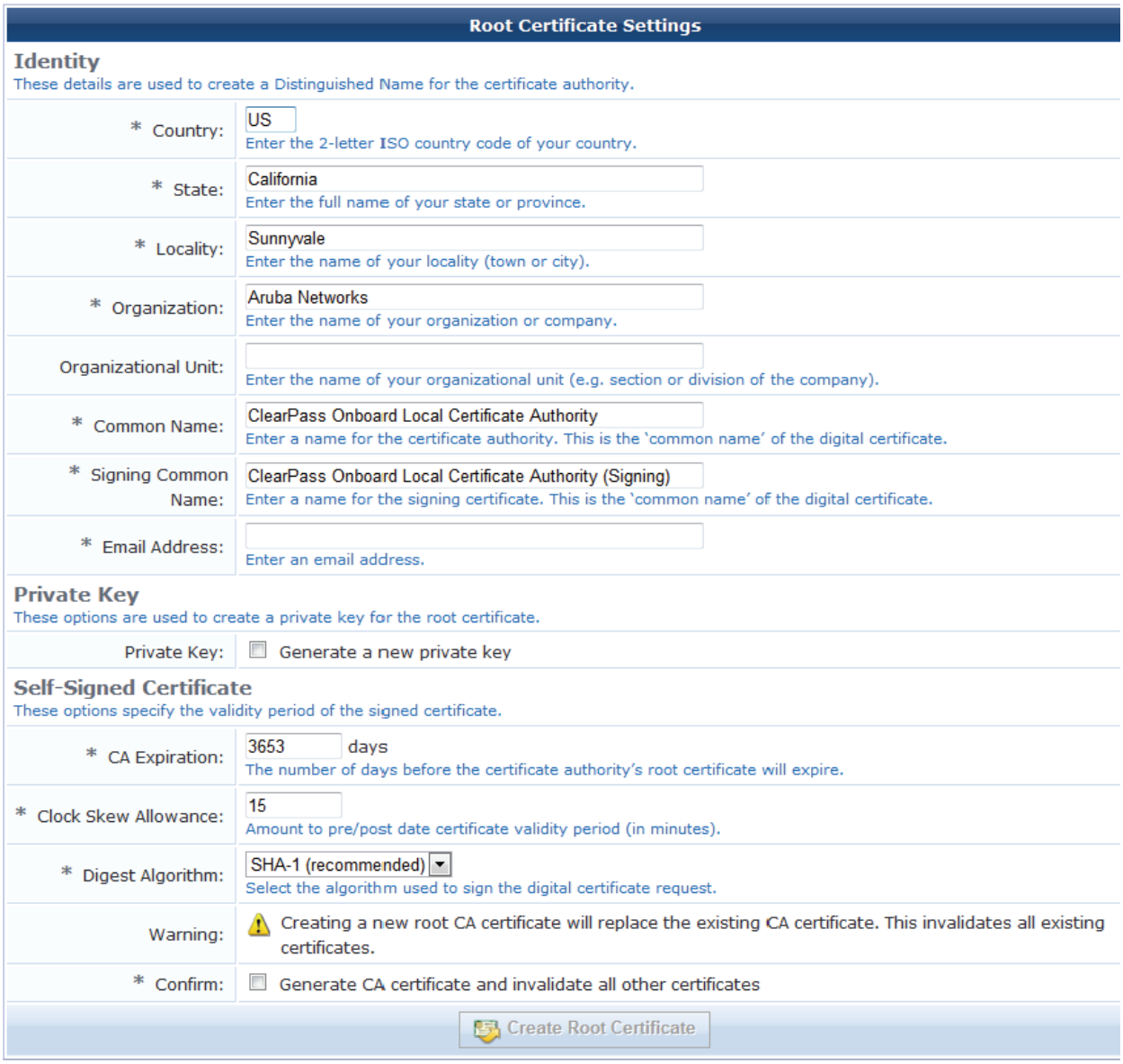
Click the **Intermediate CA** image in the **Mode** area, then click **Continue** to proceed to the second step. See "Setting Up an Intermediate Certificate Authority".

Setting Up a Root Certificate Authority

If you already have a certificate and private key for the certificate authority, see "Installing a Certificate Authority's Certificate".

After you choose **Root CA** on the Certificate Authority Settings form and click **Continue**, the Root Certificate Settings form opens. The Root Certificate Settings form is used to configure the distinguished name and properties for the certificate authority's root (self-signed) certificate.

Note: If you intend to change any of the root certificate's distinguished name properties, and you have previously created any client or server certificates or performed device provisioning using the existing root certificate, these certificates will be invalidated and deleted as the root certificate's distinguished name has changed. To avoid the complication of revoking and reissuing certificates, it is recommended that you configure the certificate authority before any device provisioning or other configuration is done.



In the **Identity** section of the form:

- Enter values in the **Country**, **State**, **Locality**, **Organization,** and **Organizational Unit** text fields that correspond to your organization. These values form part of the distinguished name for the root certificate.
- Enter a descriptive name for the root certificate in the **Common Name** text field. This value will be used to identify the root certificate as the issuer of other certificates, notably the signing certificate.
- Enter a descriptive name for the signing certificate in the **Signing Common Name** text field. This value will be used to identify the signing certificate as the issuer of client and server certificates from this certificate authority.

  The other identity information in the signing certificate will be the same as for the root certificate.
- Enter a contact email address in the **Email Address** text field. This email address will be included in the root and signing certificates, and provides a way for users of the certificate authority to contact your organization.

In the **Private Key** section:

- To create a new private key for the root certificate, mark the **Generate a new private key** check box. The form expands to include the Key Type drop-down list. Creating a new private key is only necessary if you are recreating the entire certificate authority from the beginning.

  Note: If you have previously created any client or server certificates or performed device provisioning using the existing root certificate, these certificates will be invalidated when changing the root certificate's private key.
- The **Key Type** drop-down list specifies the type of private key that should be created for the certificate. You can select one of these options:
  - **1024-bit RSA** – not recommended for a root certificate
  - **2048-bit RSA** – recommended for general use
  - **4096-bit RSA** – higher security

In the **Self-Signed Certificate** section:

- Use the **CA Expiration** field to specify the lifetime of the root certificate in days. The default value of 3653 days is a 10-year lifetime.
- The **Clock Skew Allowance** field adds a small amount of time to the start and end of the root certificate's validity period. This permits a newly issued certificate to be recognized as valid in a network where not all devices are perfectly synchronized.
- The **Digest Algorithm** drop-down list allows you to specify which hash algorithm should be used.

  Note: MD5 is not recommended for use with root certificates.

Mark the **Generate CA certificate and invalidate all other certificates** check box to confirm the changes.

Click the **Create Root Certificate** button to save the settings and generate a new root certificate.

Setting Up an Intermediate Certificate Authority

After you choose **Intermediate CA** on the Certificate Authority Settings form and click **Continue**, the Intermediate Certificate Settings form opens. The Intermediate Certificate Settings form is used to configure the distinguished name and properties for the certificate authority's certificate, which will be issued by an external certificate authority.

Note: If you intend to change any of the intermediate certificate's distinguished name properties, and you have previously created any client or server certificates or performed device provisioning using the existing intermediate certificate, these certificates will be invalidated as the intermediate certificate's distinguished name has changed. In this case, you should use the Reset to Factory Defaults form (see "Resetting Onboard

Certificates and Configuration") to delete all client certificates and re-provision all devices. You will also need to reissue any server or subordinate CA certificates.

To avoid the complication of revoking and reissuing certificates, it is recommended that you configure the certificate authority before any device provisioning or other configuration is done.

In the **Identity** section of the form:

- Enter values in the **Country**, **State**, **Locality**, **Organization,** and **Organizational Unit** text fields that correspond to your organization. These values form part of the distinguished name for the certificate authority.
- Enter a descriptive name for the certificate authority in the **Common Name** text field. This value will be used to identify the intermediate certificate as the issuer of client and server certificates from this certificate authority.
- Enter a contact email address in the **Email Address** text field. This email address will be included in the certificate authority's certificate, and provides a way for users of the certificate authority to contact your organization.

In the **Private Key** section:

- To create a new private key for the intermediate certificate, mark the **Generate a new private key** check box. The form expands to include the Key Type drop-down list. Creating a new key is only necessary if you are recreating the entire certificate authority from the beginning.

  Note: If you have previously created any client or server certificates or performed device provisioning using the existing intermediate CA certificate, these certificates will be invalidated when changing the intermediate CA's private key.

- The **Key Type** drop-down list specifies the type of private key that should be created for the certificate. You can select one of these options:
  - **1024-bit RSA** – not recommended for a certificate authority
  - **2048-bit RSA** – recommended for general use
  - **4096-bit RSA** – higher security

In the **Intermediate Certificate** section:

- The **Digest Algorithm** drop-down list allows you to specify which hash algorithm should be used.

  Note: MD5 is not recommended for use with certificate authority certificates.

Mark the **Generate CA certificate request and invalidate all other certificates** check box to confirm the changes.

Click the **Create Certificate Request** button to save the settings and generate a new certificate signing request.

Obtaining a Certificate for the Certificate Authority

The Intermediate Certificate Request page displays the certificate signing request for the certificate authority's intermediate certificate. This page is also used to renew the certificate authority's intermediate certificate when it is close to expiring.

You can copy the certificate signing request in text format using your Web browser. Use this option when you can paste the request directly into another application to obtain a certificate.

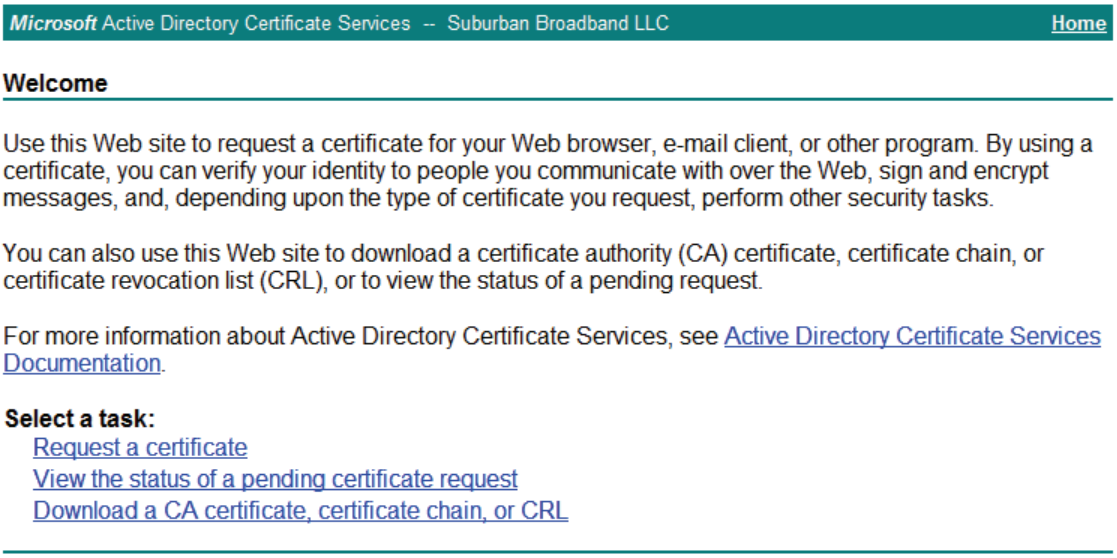
You can click the **Download the current CSR** link to download the certificate signing request as a file. Use this option when you need to provide the certificate signing request as a file to obtain a certificate.

Once you have obtained the certificate, click the Install a signed certificate link to continue configuring the intermediate certificate authority. See "Installing a Certificate Authority's Certificate".

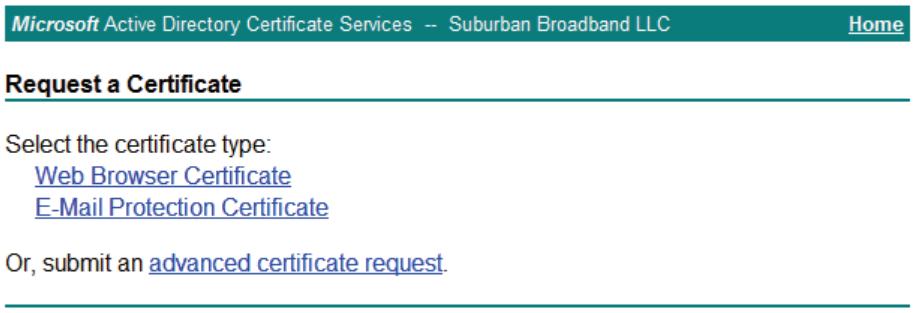
You can also click the **Change CA settings** link to return to the main Certificate Authority Settings form. Use this option to switch to a root CA, or to change the name or properties of the intermediate CA and reissue the certificate signing request.

Using Microsoft Active Directory Certificate Services

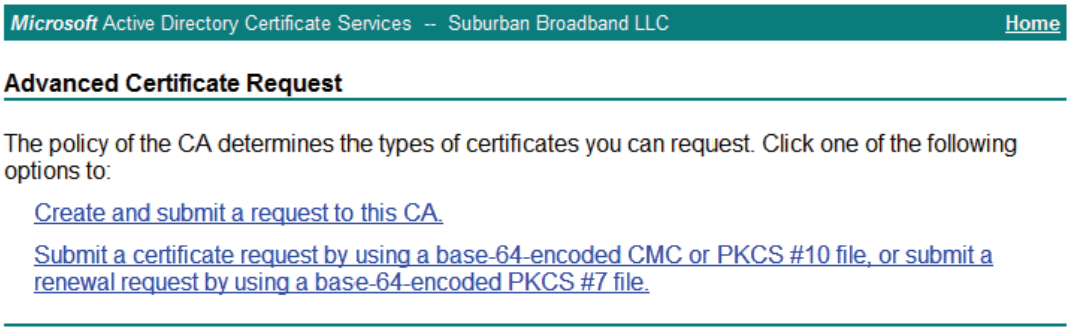
Navigate to the Microsoft Active Directory Certificate Services Web page. This page is typically found at https://yourdomain/certsrv/. The Welcome page opens.



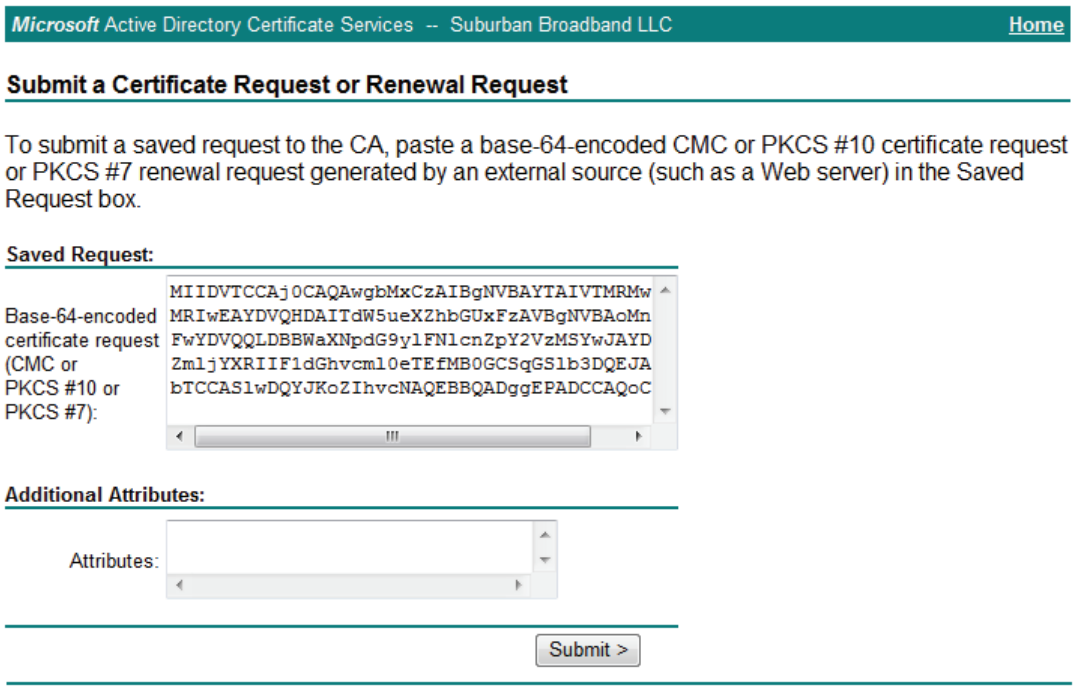
Click the **Request a Certificate** link on this page. The Request a Certificate page opens.



Click the link to submit an **advanced certificate request**. The Advanced Certificate Request page opens.



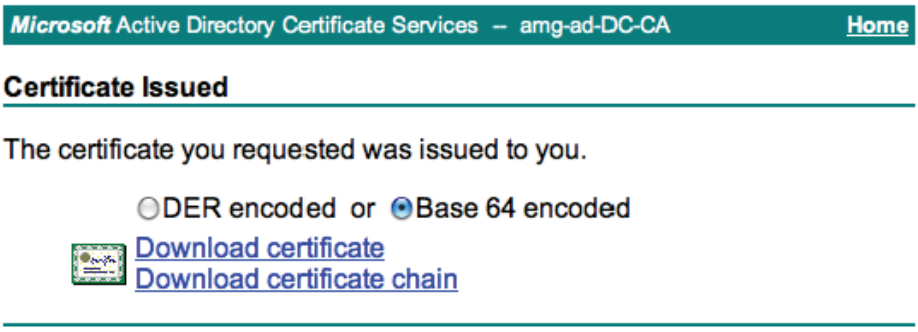
Click the link to submit a request using a **base-64-encoded CMC or PKCS #10** file. The Submit a Certificate Request or Renewal Request page is displayed.



Copy and paste the certificate signing request text into the **Saved Request** text field.

Because this certificate is for a certificate authority, select the “Subordinate Certificate Authority” in the **Certificate Template** drop-down list.

Click the **Submit** button to issue the certificate. The Certificate Issued page is displayed.



Select the **Base 64 encoded** option and then click the **Download certificate chain** link.

A file containing the intermediate certificate and the issuing certificates in the trust chain will be downloaded.

Refer to the instructions in "Installing a Certificate Authority's Certificate" for information on uploading this certificate file.

Installing a Certificate Authority's Certificate

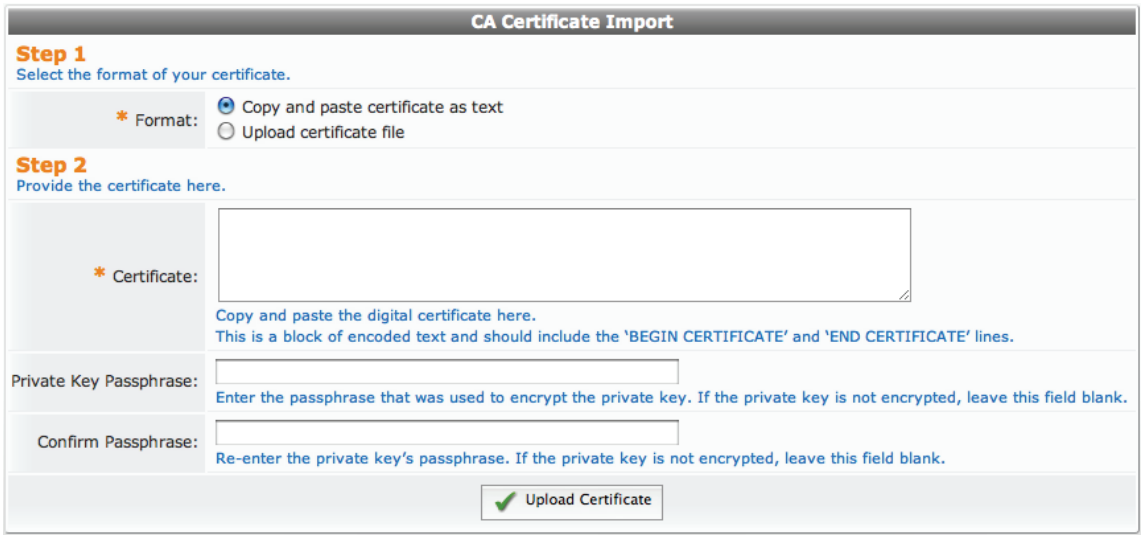
The **CA Certificate Import** page may be used to:

- Upload a certificate that has been issued by another certificate authority. This process is required when configuring an intermediate certificate authority.
  - A private key is not required, as the certificate authority has already generated one and used it to create the certificate signing request.
- Upload a certificate and private key to be used as the certificate authority's certificate. This process may be used to configure a root certificate authority.
  - A private key is required, as the certificate authority's existing private key will be replaced.

Note: This form may be used multiple times, to import each of the certificates in the trust chain. Check the message displayed above the form to determine which certificate or type of file must be uploaded next.

In the **Step 1** section of the form, select one of the following options in the **Format** radio buttons:

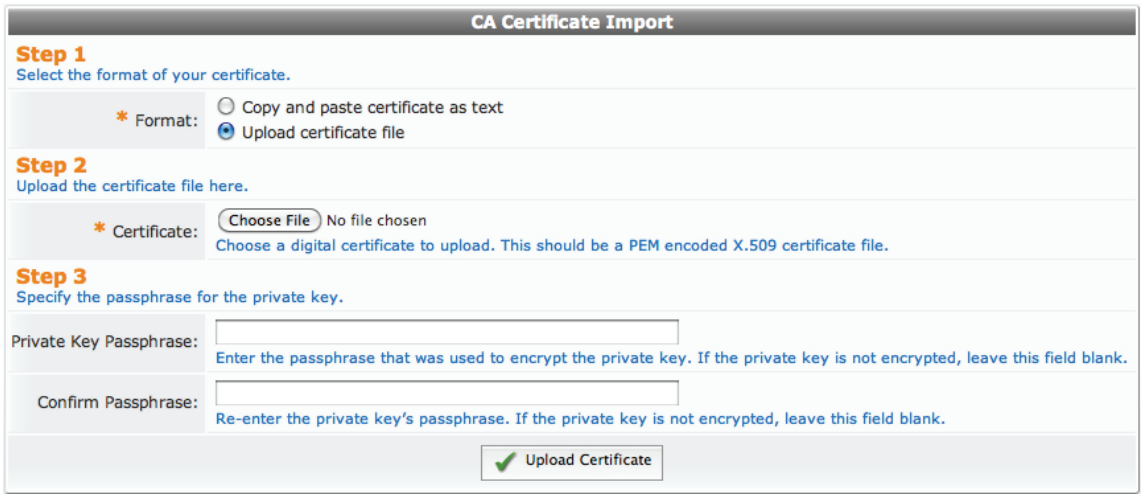
- **Copy and paste certificate as text**. The form expands to include the Step 2 fields.



  To upload a single certificate, copy and paste the certificate into the **Certificate** text field. The text must include the "BEGIN CERTIFICATE" and "END CERTIFICATE" lines. Leave the passphrase fields blank.

To upload a certificate and private key, copy and paste the certificate and private key into the **Certificate** text field. The text must include the "BEGIN CERTIFICATE" and "END CERTIFICATE" lines, as well as the "BEGIN RSA PRIVATE KEY" and "END RSA PRIVATE KEY" lines.

- **Upload certificate file** – Step 2 and Step 3 are displayed on the CA Certificate Import form.



Choose the file to upload in the **Certificate** field.

To upload a single certificate, choose a certificate file in PEM (base-64 encoded) or binary format (.crt or PKCS#7). Leave the passphrase fields blank.

To upload a certificate's private key as a separate file, choose the private key file in PEM (base-64 encoded) format. If the private key has a passphrase, enter it in the **Private Key Passphrase** and **Confirm Passphrase** fields. The private key will be automatically matched to its corresponding certificate when uploaded.

To upload a combined certificate and private key, choose a file in either PEM (base-64 encoded) or PKCS#12 format. If the private key has a passphrase, enter it in the **Private Key Passphrase** and **Confirm Passphrase** fields.

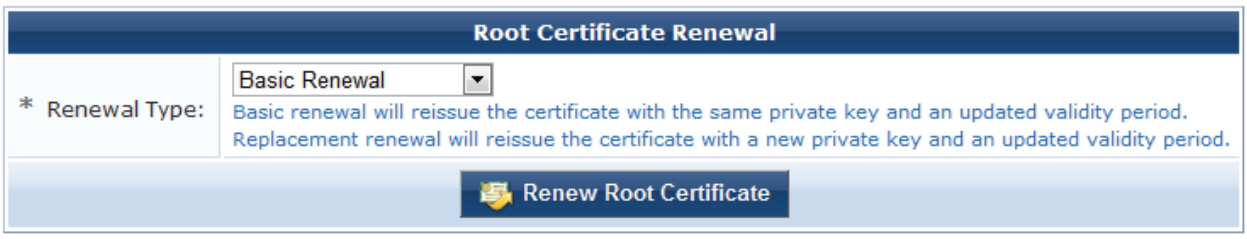
Click the ✔**Upload Certificate** button to save your changes.

- If additional certificates are required, you will remain at the same page. Check the message displayed above the form to determine which certificate or type of file must be uploaded next.
- When the trust chain is complete, it will be displayed. This completes the initialization of the certificate authority.

Renewing the Certificate Authority's Certificate

When a root certificate is close to expiration, it must be renewed.

Navigate to **Onboard** > **Certificate Authority Settings** and click the **Renew Root Certificate** link. The Root Certificate Renewal form is displayed.



Select an option in the **Renewal Type** drop-down list:

- **Basic Renewal** – Uses the same private key for the root certificate, but reissues the root CA certificate with an updated validity period. Use this option to maintain the validity of all certificates issued by the CA.
- **Replacement Renewal** – Generates a new private key for the root certificate, and reissues the root CA certificate with an updated validity period. Use this option if the root certificate has been compromised, or if you want to invalidate all certificate that were previously issued by the CA.

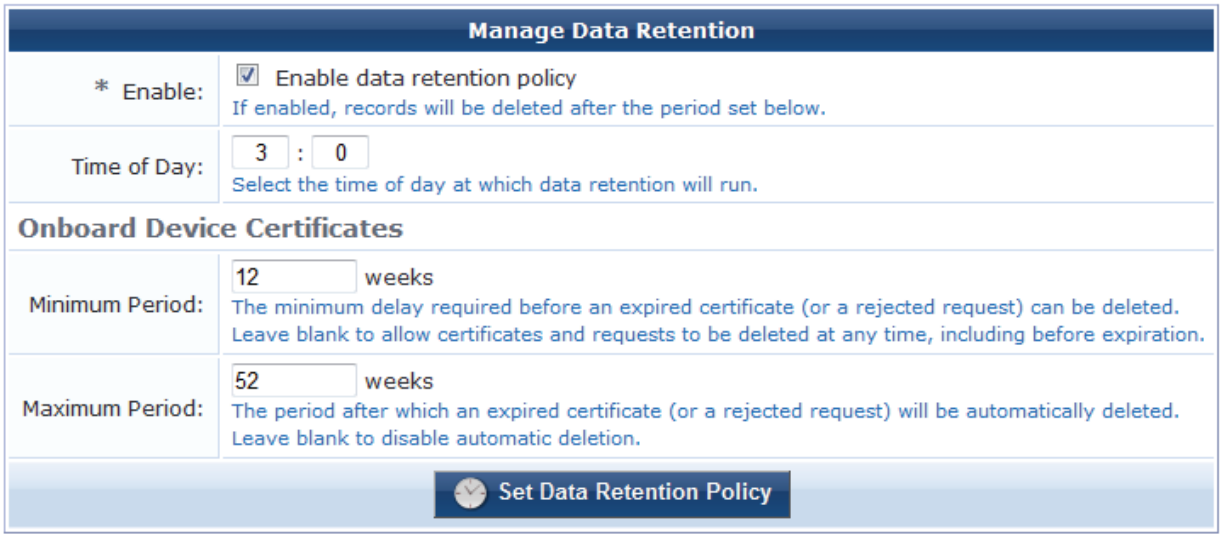
Whether you renew or replace the root certificate, you should distribute a new copy of the root certificate to all users of that certificate.

Click the **Renew Root Certificate** button to perform the renewal action.

Configuring Data Retention Policy for Certificates

The data retention policy for certificates and certificate requests can be configured by navigating to **Onboard > Certificate Authority Settings** and clicking the **Configure data retention** link.

The Manage Data Retention form is displayed.



In the Onboard Device Certificates section of the form, specify a value in the **Minimum Period** and **Maximum Period** fields that is appropriate for your organization's retention policy.

Note: Use a blank value for Minimum Period to enable the **Delete Certificate** and **Delete Request** actions in the Certificate Management list view. This is useful for testing and initial deployment.

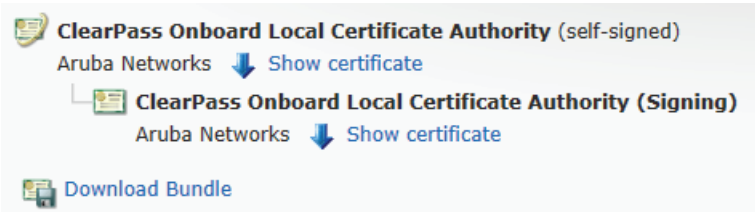
The default data retention policy specifies the values:

- Minimum Period of 12 weeks
- Maximum Period of 52 weeks

Uploading Certificates for the Certificate Authority

The Certificate Authority Trust Chain page is used to view the certificate authority's current trust chain, or to upload a new certificate in the trust chain when configuring a certificate authority.

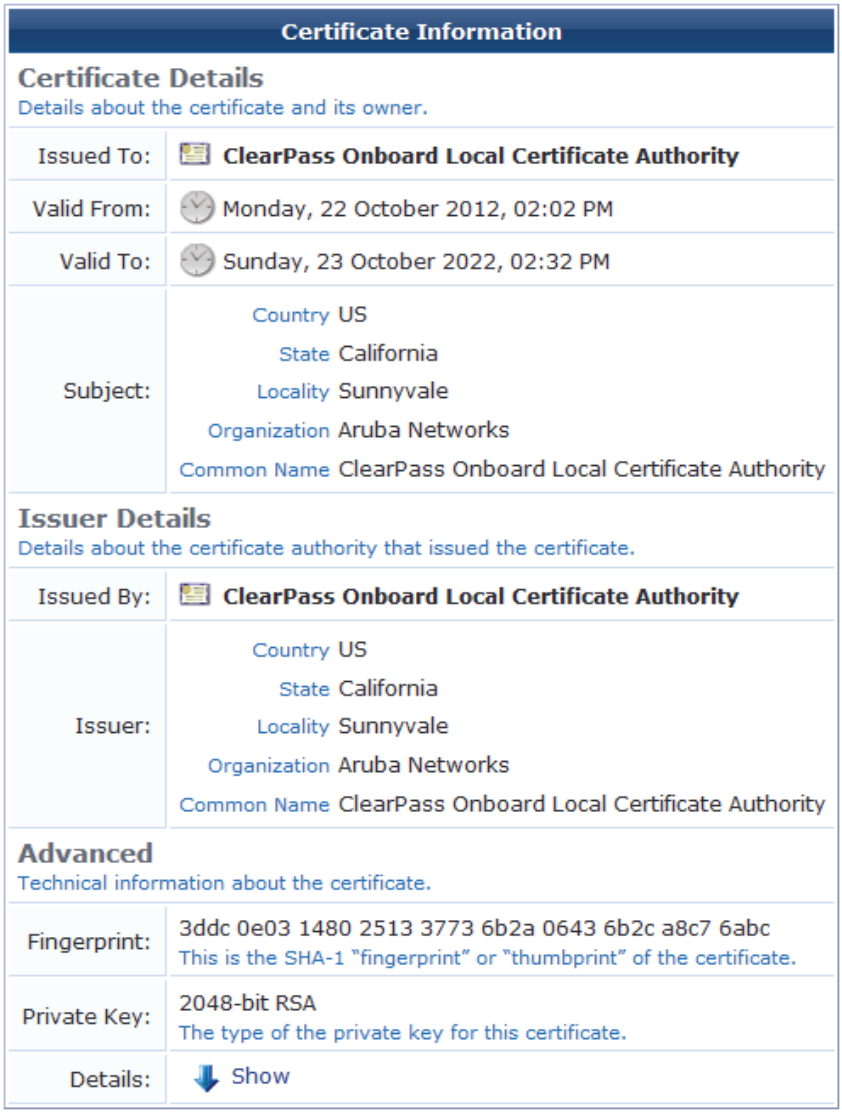
To view the Certificate Authority's trust chain, go to **Onboard** > **Certificate Authority Settings** and click the **View CA Certificate** link at the top of the page. The Certificate Authority Trust Chain page is displayed. This page shows a graphical representation of the certificates that make up the trust chain.



The first certificate listed is the root certificate. Root certificates are always self-signed and are explicitly trusted by clients.

Each additional certificate shown is an intermediate certificate. The last certificate in the list is the signing certificate that is used to issue client and server certificates.

To view the properties of a certificate in the trust chain, click the **Show certificate** link. The Certificate Information view opens.



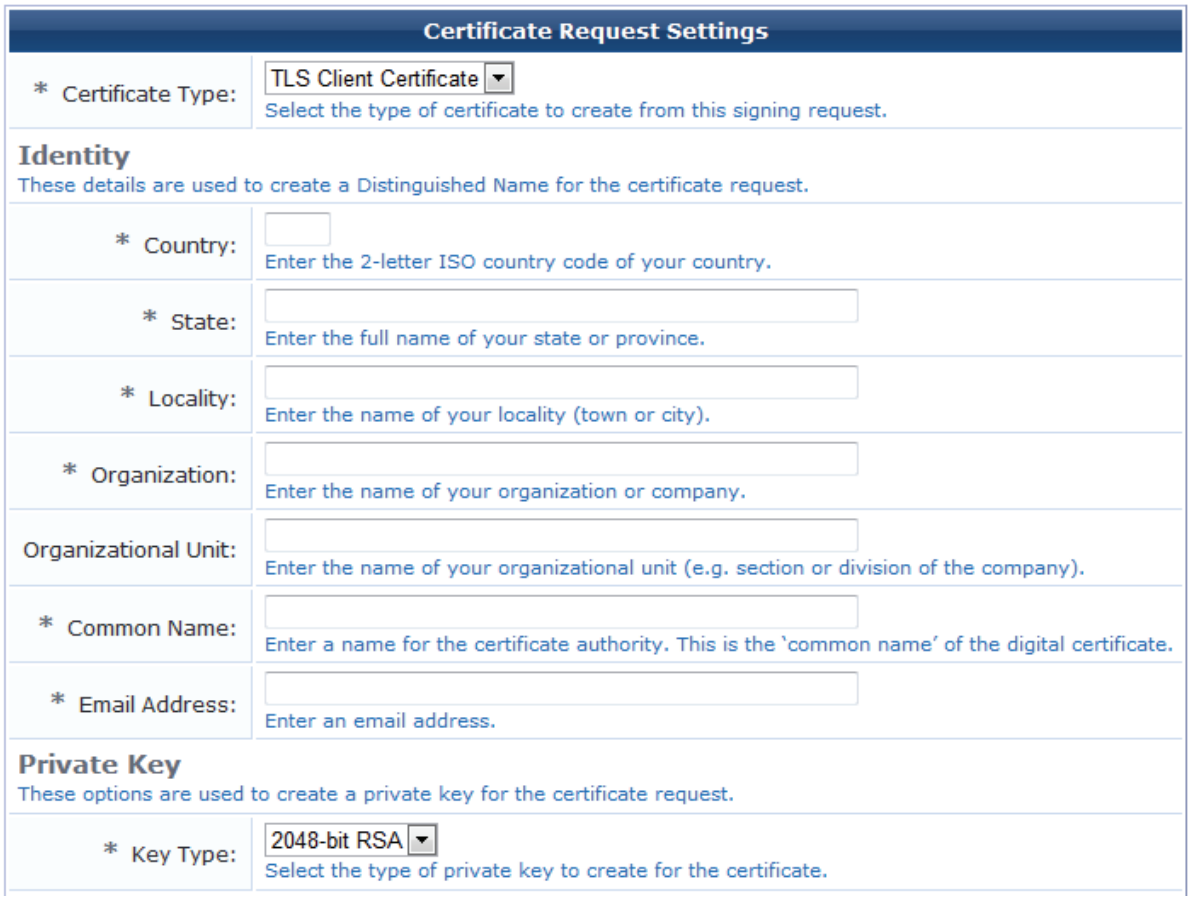
To export a certificate:

1. Click the **Download Bundle** link. The Export Certificate form opens.

2. In the **Format** row, choose the certificate format. The form expands to include configuration options for that format.
3. Complete the fields with the appropriate information, then click **Export Certificate**.

Creating a Certificate

From the **Certificate Management** page, click the **Generate a new certificate signing request** link to access the Certificate Request form.



To create a new certificate or certificate signing request, first select the type of certificate you want to create from the **Certificate Type** drop-down list:

- **TLS Client Certificate—**Use this option when the certificate is to be issued to a client, such as a user or a user's device.
  - When this option is selected, the issued certificate's extended key usage property will contain a value of "Client Auth", indicating that the certificate may be used to identify a client.
- **Trusted Certificate**—Use this option when the certificate is to be issued to a network server, such as a Web server or as the EAP-TLS authentication server.
  - When this option is selected, the issued certificate's extended key usage property will contain a value of "Server Auth", indicating that the certificate may be used to identify a server.
- **Certificate Authority**—Use this option when the certificate is for an subordinate certificate authority.
  - When this option is selected, the issued certificate will contain an extension identifying it as an intermediate certificate authority, and the extended key usage property will contain the three values "Client Auth", "Server Auth" and "OCSP Signing".
- **Code Signing**—Use this option for signing the Windows provisioning application.

Specifying the Identity of the Certificate Subject

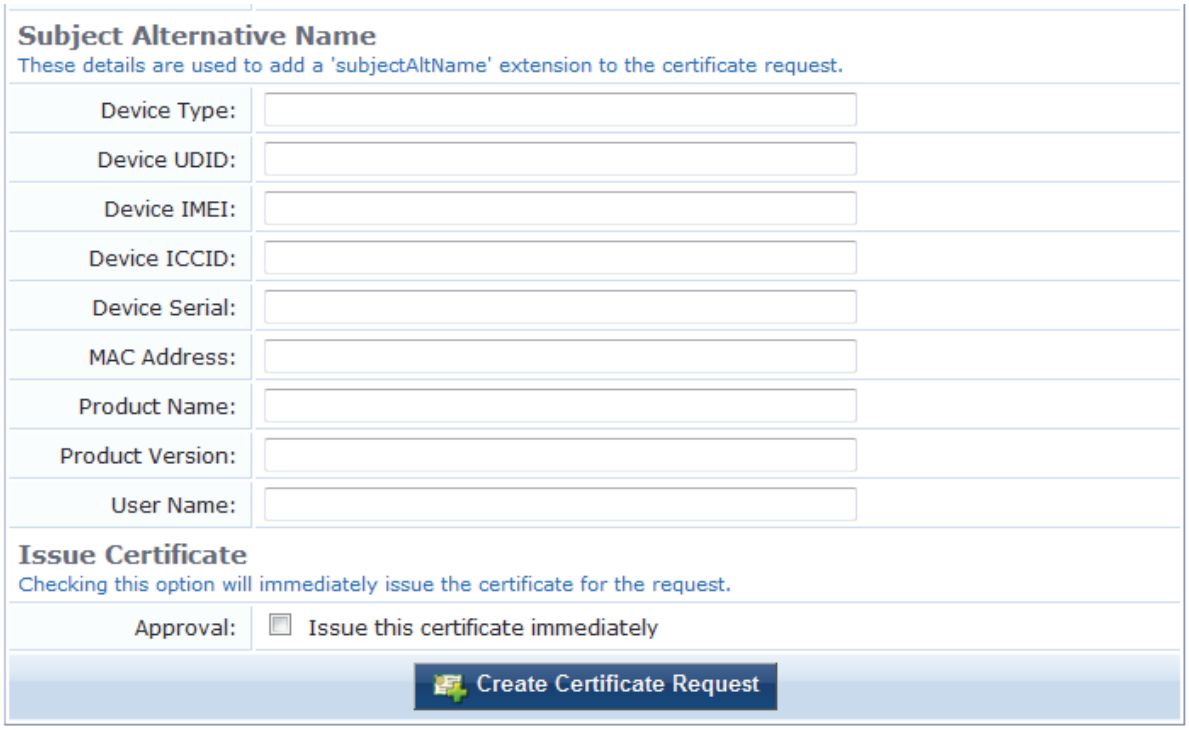
In the first part of the form, provide the identity of the person or device for which the certificate is to be issued (the "subject" of the certificate). Together, these fields are collectively known as a distinguished name, or "DN".

- Country
- State
- Locality
- Organization
- Organizational Unit
- Common Name – this is the primary name used to identify the certificate
- Email Address

The **Key Type** drop-down list specifies the type of private key that should be created for the certificate. You can select one of these options:

- **1024-bit RSA** – lower security
- **2048-bit RSA** – recommended for general use
- **4096-bit RSA** – higher security

Note: Using a private key containing more bits will increase security, but will also increase the processing time required to create the certificate and authenticate the device. The additional processing required will also affect the battery life of a mobile device. It is recommended to use the smallest private key size that is feasible for your organization.



If you have selected **TLS Client** as the certificate type, the Subject Alternative Name section is also shown. The alternative name can be used to specify additional identification details for the certificate's subject. If one or more of these options are provided, the issued certificate will contain a subjectAltName extension with the specified values.

Table 16 explains the fields that may be included as part of the subject alternative name.

Table 16 *Subject Alternative Name Fields Supported When Creating a TLS Client Certificate Signing Request*

Name	Description
Device Type	Type of device, such as "iOS", "Android", etc.
Device UDID	Unique device identifier (UDID) for this device. This is typically a 64-bit, 128-bit or 160-bit number represented in hexadecimal (16, 32 or 40 characters, respectively).
Device IMEI	International Mobile Equipment Identity (IMEI) number allocated to this device.
Device ICCID	Integrated Circuit Card Identifier (ICCID) number from the Subscriber Identity Module (SIM) card present in the device.
Device Serial	Serial number of the device.
MAC Address	IEEE MAC address of this device.
Product Name	Product string identifying the device and often including the hardware version information.
Product Version	Software version number for the device.
User Name	Username of the user who provisioned the device.

Issuing the Certificate Request

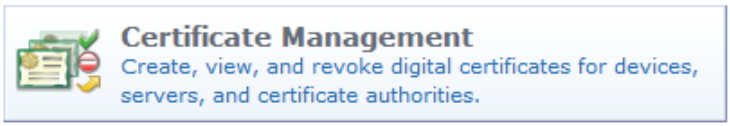
Mark the **Issue this certificate immediately** check box to automatically create the certificate.

Click the **Create Certificate Request** button to save your changes.

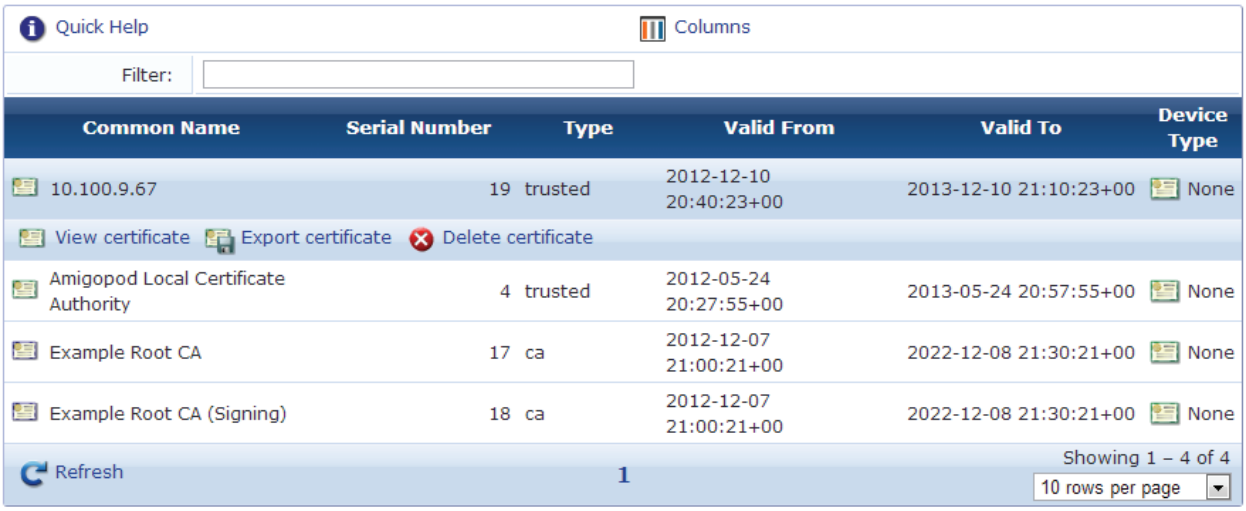
- If the "Issue this certificate immediately" check box is marked, the certificate will be issued immediately and will be displayed in the Certificate Management list view.
- If the "Issue this certificate immediately" check box is
- **not** marked, the certificate request will be displayed in the Certificate Management list view. The certificate can then be issued or rejected at a later time.

Managing Certificates

To view the list of certificates and work with them, go to **Onboard > Certificate Management**, or click the **Certificate Management** command link.



The Certificate Management list view opens. This list displays all of the certificates and certificate requests in the Onboard system.



Information provided in the Certificate Management list includes common name, serial number (if available), certificate type, validity date range, and device type—iOS, Android, Windows, or None (if not associated with a device type). Table 17 lists the types of certificate that are displayed in this list.

Table 17 *Types of Certificate Supported by Onboard Certificate Management*

	Certificate Type	"Type" Column	Notes
	Root certificate	ca	Self-signed certificate for the certificate authority
	Intermediate certificate	ca	Issued by the root CA or another intermediate CA

Table 17 *Types of Certificate Supported by Onboard Certificate Management (Continued)*

	Certificate Type	"Type" Column	Notes
	Profile signing certificate	profile-signing	Issued by the certificate authority
	Certificate signing request	tls-client or trusted	The type shown depends on the kind of certificate requested
	Rejected certificate signing request	tls-client or trusted	Certificate request that was rejected due to an administrator decision
	Device certificate	scep-client	Issued to iOS and OS X (10.7+) devices only
	Client certificate	tls-client	Identity certificate issued to a specific user's device
	Server certificate	trusted	Identity certificate issued to a server
	Code-signing certificate	ca	Used for signing the Windows provisioning application
	Revoked certificate	--	Certificate that has been administratively revoked and is no longer valid
	Expired certificate	--	Certificate that is outside its validity period and is no longer valid

Searching for Certificates in the List

The **Filter** field can be used to quickly search for a matching certificate. Type a username into this field to locate all certificates matching that username quickly.

The filter is applied to all columns displayed in the list view. To search by another field, such as MAC address, device type, or device serial number, click the **Columns** tab, select the appropriate column(s), and then click the **Save and Reload** button. The list view will refresh to update the results of the filter.

Click the **Clear Filter** link to restore the default view.

Use the paging control at the bottom of the list to jump forwards or backwards by one page, or to the first or last page of the list. You can also click an individual page number to jump directly to that page.

Note: When the list contains many thousands of certificates, consider using the Filter field to speed up finding a specific certificate.

Click the column headers to sort the list view by that column. Click the column header a second time to reverse the direction of the sort.

Working with Certificates in the List

Click on a certificate to select it. You can then select from one of these actions:

- **View certificate** – Displays the properties of the certificate. Click the **Cancel** button to close the certificate properties.
- **Export certificate** – Displays the Export Certificate form.

Use the **Format** drop-down list to select the format in which the certificate should be exported. The following formats are supported:

- **PKCS#7 Certificates (.p7b)**—Exports the certificate, and optionally the other certificates forming the trust chain for the certificate, as a PKCS#7 container.
- **Base-64 Encoded (.pem)**—Exports the certificate as a base-64 encoded text file. This is also known as "PEM format". You may optionally include the other certificates forming the trust chain for the certificate.
- **Binary Certificate (.crt)**—Exports the certificate as a binary file. This is also known as "DER format".
- **Open SSL Text Format**—Exports the certificate as a full openssl text-format output, allowing you to view advanced details such as X509v3 extensions. It also includes the certificate in .pem format appended to the .txt file.
- **PKCS#12 Certificate & Key (.p12)**—Exports the certificate and its associated private key, and optionally any other certificates required to establish the trust chain for the certificate, as a PKCS#12 container. This option is only available if the private key for the certificate is available to the server. If you select the PKCS#12 format, you must enter a passphrase to protect the private key stored in the file. **Note:** To protect against brute-force password attacks and ensure the security of the private key, you should use a strong passphrase – one consisting of several words, mixed upper- and lower-case letters, and punctuation or other symbol characters.

Click the **Export Certificate** button to download the certificate file in the selected format.

- **Revoke certificate** – Displays the Revoke Certificate form.

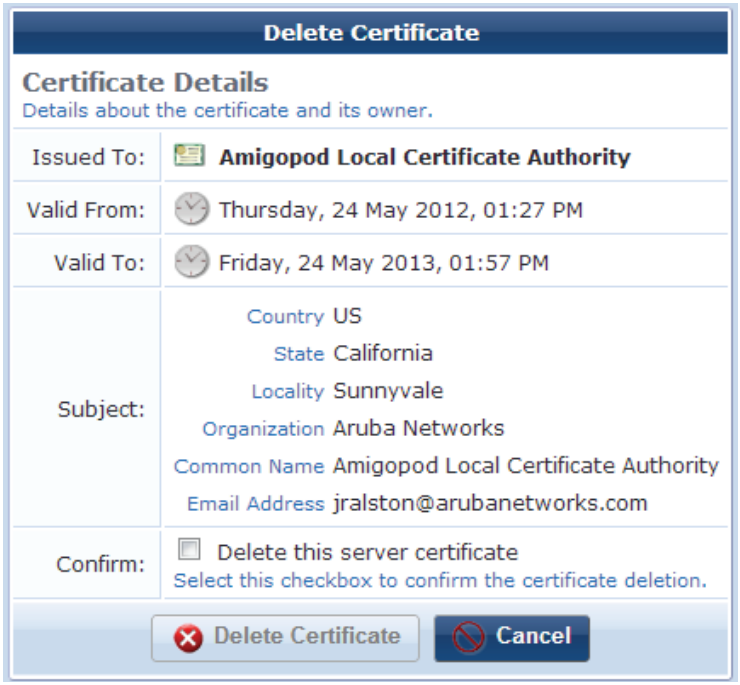
  Mark the **Revoke this client certificate** check box to confirm that the certificate should be revoked, and then click the **Revoke Certificate** button.

  Once the certificate has been revoked, future checks of the certificate's validity using OCSP or CRL will indicate that the certificate is no longer valid.

  Note: Due to the way in which certificate revocation lists work, a certificate cannot be un-revoked. A new certificate must be issued if a certificate is revoked in error.

  Note: Revoking a device's certificate will also prevent the device from being re-provisioned. This is necessary to prevent the user from simply re-provisioning and obtaining a new certificate. To re-provision the device, the revoked certificate must be deleted.

- **Delete certificate** – Removes the certificate from the list. Trusted certificates that were imported into Onboard may be deleted at any time after import. For all other certificates, this option is only available if the data retention policy is configured to permit the certificate's deletion. See "Configuring Data Retention Policy for Certificates".



The Delete Certificate form is displayed. Mark the **Delete this client certificate** check box to confirm the certificate's deletion, and then click the **Delete Certificate** button.

Working with Certificate Signing Requests

Certificate signing requests can be managed through the Certificate Management list view. This allows for server certificates, subordinate certificate authorities, and other client certificates not associated with a device to be issued by the Onboard certificate authority.

Click on a certificate request to select it. You can then select from one of these actions:

- **View request** – Displays the properties of the certificate request. Click the **Cancel** button to close the certificate request properties.
- **Export request** – Displays the Export Certificate Request form.

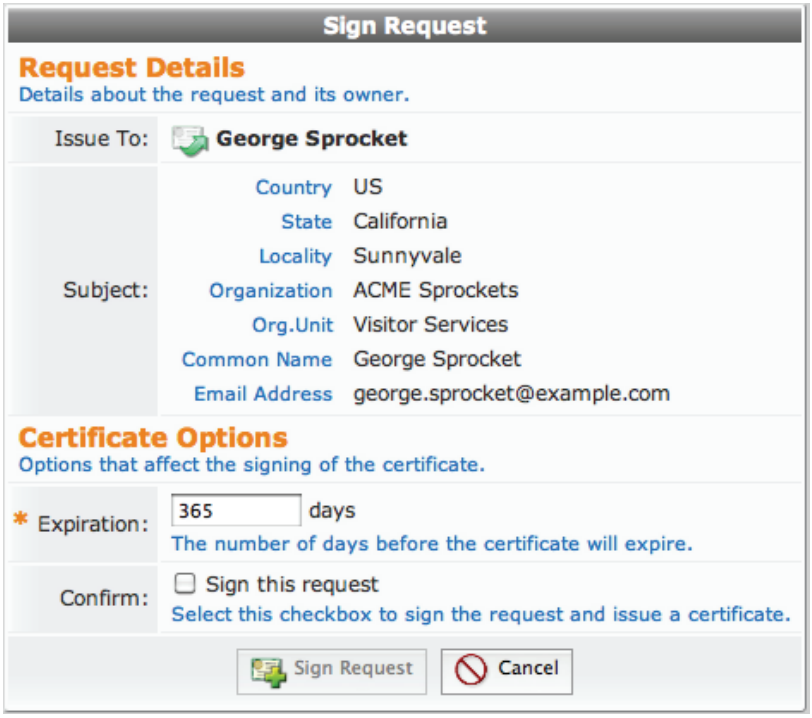
Use the **Format** drop-down list to select the format in which the certificate signing request should be exported. The following formats are supported:

- **PKCS#10 Certificate Request (.p10)** – Exports the certificate signing request in binary format.
- **Base-64 Encoded (.pem)** – Exports the certificate signing request as a base-64 encoded text file. This is also known as "PEM format".

If you choose Base-64 Encoded, the form expands to include the **Trust Chain** row. You can use this option to create and export a certificate bundle that includes the Intermediate CA and Root CA and can be imported in ClearPass Policy Manager as the server certificate (ClearPass Policy Manager does not accept PKCS#7). To include the trust chain in a certificate bundle that can be imported as the server certificate in ClearPass Policy Manager, mark the **Include certificate trust chain** check box, then click the **Export Certificate** button.

Click the **Export Request** button to download the certificate signing request file in the selected format.

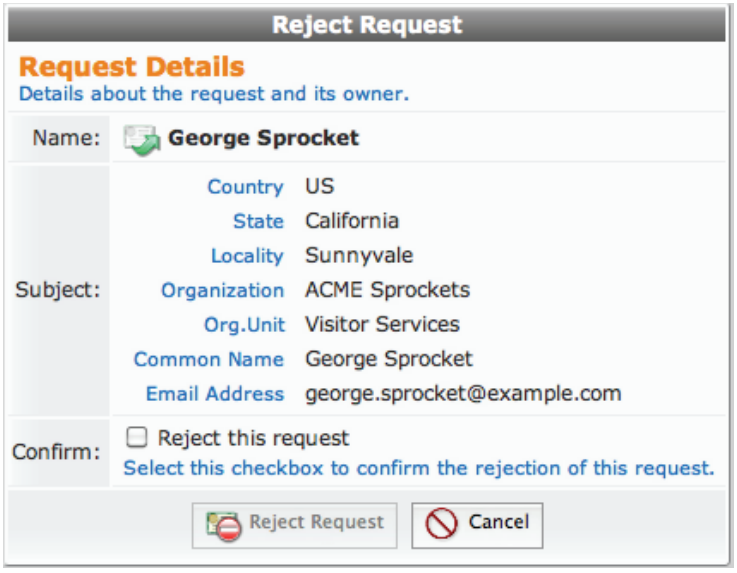
- **Sign request** – Displays the Sign Request form. Use this action to approve the request for a certificate and issue the certificate.



Use the **Expiration** text field to specify how long the issued certificate should remain valid.

Mark the **Sign this request** check box to confirm that the certificate should be issued, and then click the **Sign Request** button. The certificate will be issued and will then replace the certificate signing request in the list view.

- **Reject request** – Displays the Reject Request form. Use this action to reject the request for a certificate. Rejected requests are automatically deleted according to the data retention policy.



  Mark the **Reject this request** check box to confirm that the certificate signing request should be rejected, and then click the **Reject Request** button.

- **Delete request** – Removes the certificate signing request from the list. This option is only available if the data retention policy is configured to permit the certificate signing requests's deletion. See "Configuring Data Retention Policy for Certificates".

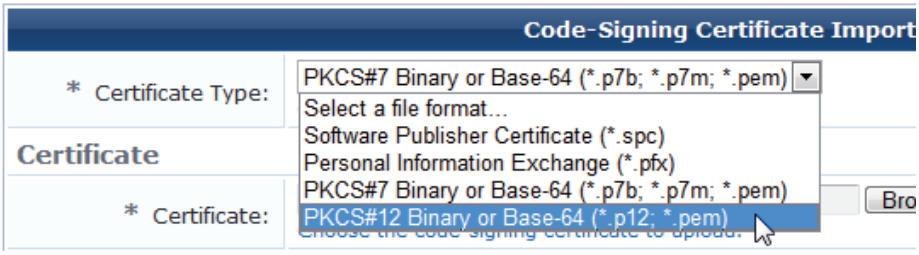
  The Delete Request form is displayed. Mark the **Delete this request** check box to confirm the certificate signing request's deletion, and then click the **Delete Request** button.

Importing a Code-Signing Certificate

Onboard supports importing a code-signing certificate chain and private key for signing the Windows provisioning application. Certificates can be uploaded as PFX, PKCS-12, SPC, or PKCS-7, and can include a chain of certificates. An operator's profile must include the Import Code-Signing Certificate privilege in order to access this feature.

To import a code-signing certificate:

1. Go to **Onboard > Certificate Management** or **Onboard > Provisioning Settings** and click the **Upload a code-signing certificate** link at the top of the page. The Code-Signing Certificate Import form opens.



2. In the **Certificate Type** drop-down list, choose the file type—either **SPC**, **PFX**, **PKCS-7**, or **PKCS-12**. The form expands to include the Certificate area, with fields for uploading the certificate, uploading the private key, and entering the passphrase.

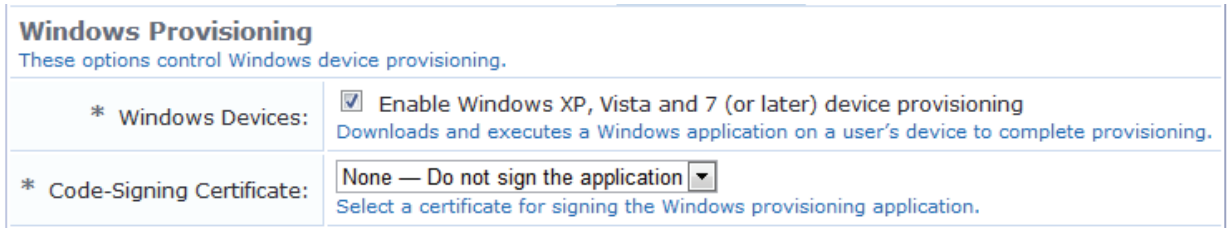
   For PFX and PKCS-12 files, the private key must be included in the certificate file, so the Private Key upload option is not available in the form. The private key passphrase is required.

   For SPC and PKCS-7 files, a PEM-encoded private key must be uploaded separately using the Private Key upload option on the form. If it is encrypted, the passphrase must also be provided.

3. Click **Upload Certificate**. The certificate chain is displayed.

To use the certificate for code-signing:

1. Go to **Onboard > Provisioning Settings** and scroll to the **Windows Provisioning** section of the form.



2. In the **Code-Signing Certificate** drop-down list, select the uploaded certificate.

To create a test certificate:

1. Go to **Onboard > Certificate Management** and click the Gen**erate a new certificate signing request** link. The Certificate Request Settings form opens.
2. In the **Certificate Type** drop-down list, choose **Code-Signing**.
3. Complete the rest of the form with your information. Mark the **Issue this certificate immediately** check box, then click **Create Certificate Request**.

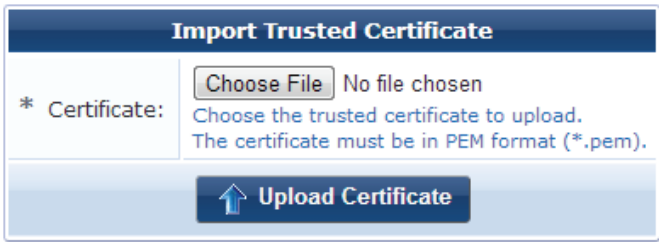
   The test certificate is displayed in the list on the Certificate Management page, and can be selected on the Provisioning Settings form.

Importing a Trusted Certificate

Onboard's Certificate Management page supports importing trusted certificates. Certificates may be uploaded in PEM format (*.pem).

To import a trusted certificate:

1. Go to **Onboard > Certificate Management** and click the **Upload a trusted certificate** link in the upper-right corner. The Import Trusted Certificate form opens.



2. Click **Choose File** to browse to the certificate on your system, then click **Upload Certificate**. A confirmation message is displayed, and the imported certificate is included in the Certificate Management list. You can click the Show Certificate link next to the certificate's name to view the certificate's details.

3. You can use the following additional options in the upper-right corner of the Import Trusted Certificate page:
   - Click the **Upload another trusted certificate** link to upload additional certificates.
   - Click the **Edit <certificate name> trust settings** link to open the Trust tab of the Network Settings form.

Requesting a Certificate

From the Certificate Management page, click the **Upload a certificate signing request** link to access the Certificate Signing Request form.

Providing a Certificate Signing Request in Text Format

If you have a certificate signing request in text format, click the **Copy and paste certificate signing request as text** radio button.

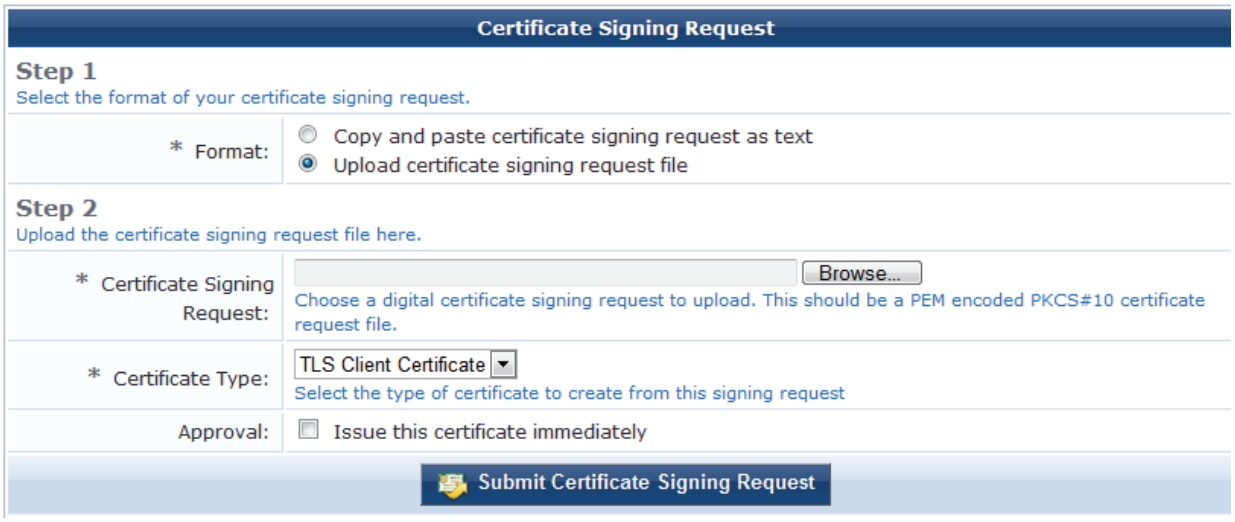
Paste the text into the **Certificate Signing Request** text field. Be sure to include the complete block of text, including the beginning and ending lines.

A complete certificate signing request looks like the following:

```
-----BEGIN CERTIFICATE REQUEST-----
MIIB7DCCAVUCAQAwgasxCzAJBgNVBAYTAlVTMRMwEQYDVQQIEwpDYWxpZm9ybmlh
MRIwEAYDVQQHEwlTdW5ueXZhbGUxFzAVBgNVBAoTDkFDTUUgU3Byb2NrZXRzMRkw
FwYDVQQLExBWaXNpdG9yIFNlcnZpY2VzMR4wHAYDVQQDExVBdXRoZW50aWNhdGlv
biBTZXJ2ZXIxHzAdBgkqhkiG9w0BCQEWEGluZm9AZXhhbXBsZS5jb20wgZ8wDQYJ
KoZIhvcNAQEBBQADgY0AMIGJAoGBALR4wRSH26wlcf3OEPEIh34iXRQIUrnYnDfo
+ZezeB/i4NZUhRvLMvhPW7DcLpiZJ17ILj3aPPUXWDBYYiiuOkmuFX3dG7eKCLMH
Z4E9z1ozK5Znm8cWIj56kg69le7QrAZBYrd5QaBTMxEe0F9CGFsYbFx1viMUMxN6
EJILaCTBAgMBAAGgADANBgkqhkiG9w0BAQUFAAOBgQB8/So9KU5BS3oxjyxftIwF
dWvNP2CNruKyQaba5RQ1ixdHAsPE+3uYIHNvlqqIpSzBlfYkr21S4DdR3SSC3bXy
t4l/fyMuC1cEG/RpPSxdDALpeT8MuoGV1JonKo2BDitOEd4y5SXGmHmDBHrPW2Nd
gthkrtBb/a2WAkNcRfDuiQ==
-----END CERTIFICATE REQUEST-----
```

Providing a Certificate Signing Request File

Alternatively, if you have the certificate signing request as a file, click the **Upload certificate signing request file** radio button.



Use the Certificate Signing Request field to select the appropriate file for upload.

Note: The file should be a base-64 encoded (PEM format) PKCS#10 certificate signing request.

Specifying Certificate Properties

Select the type of certificate from the **Certificate Type** drop-down list. Choose from one of the following options:

- **TLS Client Certificate** – Use this option when the certificate is to be issued to a client, such as a user or a user's device.
  - When this option is selected, the issued certificate's extended key usage property will contain a value of "Client Auth", indicating that the certificate may be used to identify a client.
- **TLS Server Certificate** – Use this option when the certificate is to be issued to a network server, such as a Web server or as the EAP-TLS authentication server.
  - When this option is selected, the issued certificate's extended key usage property will contain a value of "Server Auth", indicating that the certificate may be used to identify a server.
- **Certificate Authority** – Use this option when the certificate is for an subordinate certificate authority.
  - When this option is selected, the issued certificate will contain an extension identifying it as an intermediate certificate authority, and the extended key usage property will contain the three values "Client Auth", "Server Auth" and "OCSP Signing".

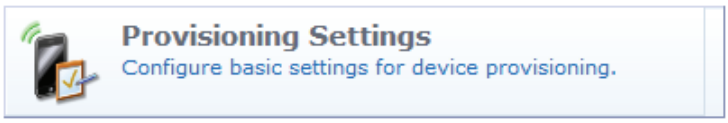
Mark the **Issue this certificate immediately** check box to automatically issue the certificate.

Click the **Submit Certificate Signing Request** button to save your changes.

- If the "Issue this certificate immediately" check box is marked, the certificate will be issued immediately and will be displayed in the Certificate Management list view.
- If the "Issue this certificate immediately" check box is **not** marked, the certificate request will be displayed in the Certificate Management list view. The certificate can then be issued or rejected at a later time.

Configuring Provisioning Settings

To configure basic device provisioning settings, go to **Onboard > Provisioning Settings**, or click the **Provisioning Settings** command link. The Device Provisioning Settings page opens.



This page is used to configure the settings for ClearPass Onboard device provisioning, including:

- The organization name displayed during device provisioning
- Properties for the certificates issued to devices when they are provisioned
- Which operating systems should be supported
- Authorization properties – the number of devices that a user may provision

The Device Provisioning form is organized in tabbed pages, with separate tabs for general, iOS & OS X, Legacy OS X, Windows, Android, and Onboard Client information.

Configuring Basic Provisioning Settings

To configure basic provisioning settings:

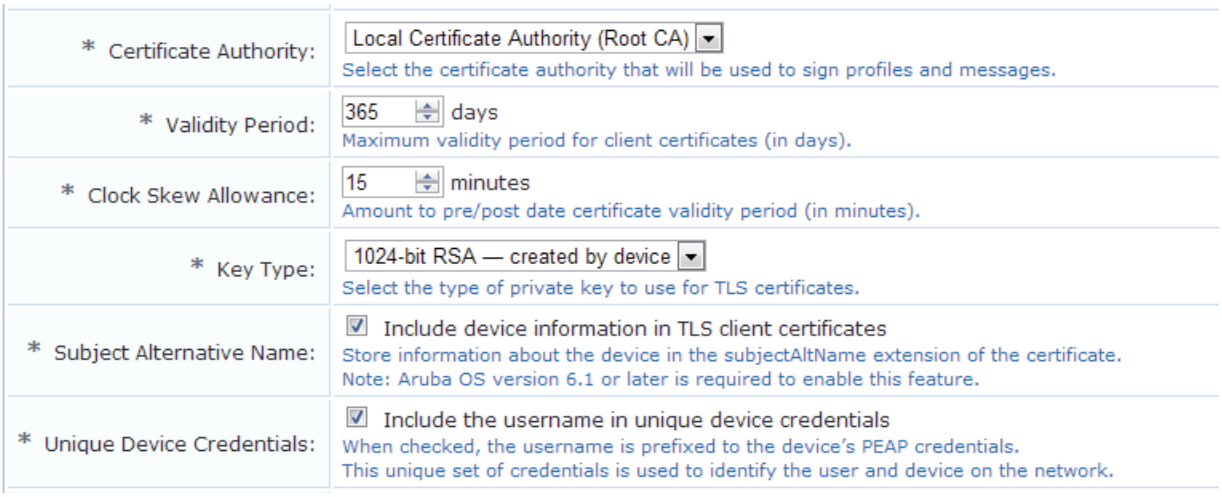
1. Go to **Onboard > Provisioning Settings** and click the **General** tab. The first part of the Device Provisioning Settings form's General tab is used to specify basic information about Onboard provisioning.

2. The **Name** and **Description** fields are used internally to identify this set of Onboard settings for the network administrator. These values are never displayed to the user during device provisioning.
3. Use the **Organization** field to provide the name of your organization; this will be displayed to the user during the device provisioning process.

Configuring Certificate Properties for Device Provisioning

To specify the properties for certificates issued to devices:

1. Go to **Onboard > Provisioning Settings**, click the **General** tab, and scroll to the **Certificate Authority** row.



2. The **Certificate Authority** drop-down list can be used to select a different certificate authority. By default, there is only a single certificate authority.
3. Use the **Validity Period** text field to specify the maximum length of time for which a client certificate issued during device provisioning will remain valid.
4. The **Clock Skew Allowance** text field adds a small amount of time to the start and end of the client certificate's validity period. This permits a newly issued certificate to be recognized as valid in a network where not all devices are perfectly synchronized.

   For example, if the current time is 12:00, and the clock skew allowance is set to the default value of 15 minutes, then the client certificate will be issued with a "not valid before" time of 11:45. In this case, if the authentication server that receives the client certificate has a time of 11:58, it will still recognize the certificate as valid. If the clock skew allowance was set to 0 minutes, then the authentication server would not recognize the certificate as valid until its clock has reached 12:00.

   The default of 15 minutes is reasonable. If you expect that all devices on the network will be synchronized then the value may be reduced. A setting of 0 minutes is not recommended as this does not permit any variance in clocks between devices.

   When issuing a certificate, the certificate's validity period is determined as follows:

   - The "not valid before" time is set to the current time, less the clock skew allowance.
   - The "not valid after" time is first calculated as the earliest of the following:
   - The current time, plus the maximum validity period.
   - The expiration time of the user account for whom the device certificate is being issued.
   - The "not valid after" time is then increased by the clock skew allowance.
5. The **Key Type** drop-down list specifies the type of private key that should be created when issuing a new certificate. You can select one of these options:
   - **1024-bit RSA – created by device**: Lower security. Uses SCEP to provision the EAP-TLS certificate.
   - **2048-bit RSA – created by device**: Recommended for general use. Uses SCEP to provision the EAP-TLS certificate.
   - **1024-bit RSA – created by server**: Lower security.
   - **2048-bit RSA – created by server**: Recommended for general use.
   - **4096-bit RSA – created by server**: Higher security.

   Note: Using a private key containing more bits will increase security, but will also increase the processing time required to create the certificate and authenticate the device. The additional processing required will also affect the battery life of a mobile device. It is recommended to use the smallest private

key size that is feasible for your organization. The “created by device” options use SCEP to provision the EAP-TLS device certificate, so the private key is known only to the device rather than also known by the user. When a “created by device” option is selected, the generated key is used instead of a username/password authentication defined in Network Settings.

6. Mark the **Include device information in TLS client certificates** check box to include additional fields in the TLS client certificate issued for a device. These fields are stored in the subject alternative name (subjectAltName) of the certificate. Refer to Table 18 for a list of the fields that are stored in the certificate when this option is enabled.

   Storing additional device information in the client certificate allows for additional authorization checks to be performed during device authentication.

   Note: If you are using an Aruba Controller to perform EAP-TLS authentication using these client certificates, you must have Aruba OS 6.1 or later to enable this option.

Table 18 *Device Information Stored in TLS Client Certificates*

Name	Description	OID
Device ICCID	Integrated Circuit Card Identifier (ICCID) number from the Subscriber Identity Module (SIM) card present in the device. This is only available for devices with GSM (cellular network) capability, where a SIM card has been installed.	mdpsDeviceIccid (.4)
Device IMEI	International Mobile Equipment Identity (IMEI) number allocated to this device. This is only available for devices with GSM (cellular network) capability.	mdpsDeviceImei (.3)
Device Serial	Serial number of the device.	mdpsDeviceSerial (.9)
Device Type	Type of device, such as “iOS”, “Android”, etc.	mdpsDeviceType (.1)
Device UDID	Unique device identifier (UDID) for this device. This is typically a 64-bit, 128-bit or 160-bit number represented in hexadecimal (16, 32, or 40 characters, respectively).	mdpsDeviceUdid (.2)
MAC Address	IEEE MAC address of this device. This element may be present multiple times, if a device has more than one MAC address (for example, an Ethernet port and a Wi-Fi adapter).	mdpsMacAddress (.5)
Product Name	Product string identifying the device and often including the hardware version information.	mdpsProductName (.6)
Product Version	String containing the software version number for the device.	mdpsProductVersion (.7)
User Name	String containing the username of the user who provisioned the device.	mdpsUserName (.8)

Note: Object Identifier. These OIDs are relative to the ClearPass Guest base OID, which is 1.3.6.1.4.1.14823.1.5.1.

Configuring Revocation Checks and Authorization

To specify automatic certificate revocation checks and to configure device authorization:

1. Go to **Onboard > Provisioning Settings**, click the **General** tab, and scroll to the **Authority Info Access** row.

2. Specify one of the following options in the **Authority Info Access** drop-down list to control automatic certificate revocation checks:
   - **Do not include OCSP responder URL** – The Authority Info Access extension is not included in the client certificate. Certificate revocation checking must be configured manually on the authentication server. This is the default option.
   - **Include OCSP responder URL** – The Authority Info Access extension is added to the client certificates, with the OCSP responder URL set to a predetermined value. This value is displayed as the "OCSP URL".
   - **Specify an OCSP responder URL** – The Authority Info Access extension is added to the client certificates, with the OCSP responder URL set to a value defined by the administrator. This value may be specified in the "OCSP URL" field.
3. In the **Unsupported Device** text box, enter instructions to be displayed to the user if they attempt to provision an unsupported device. The text can be entered as HTML code, and you can use Smarty template functions. If this field is left empty, the following default text will be displayed: "Your operating system is not supported. Please contact your network administrator."

4. In the **Authorization** area of the form, enter a number in the **Maximum Devices** field to limit the maximum number of devices that each user may provision.

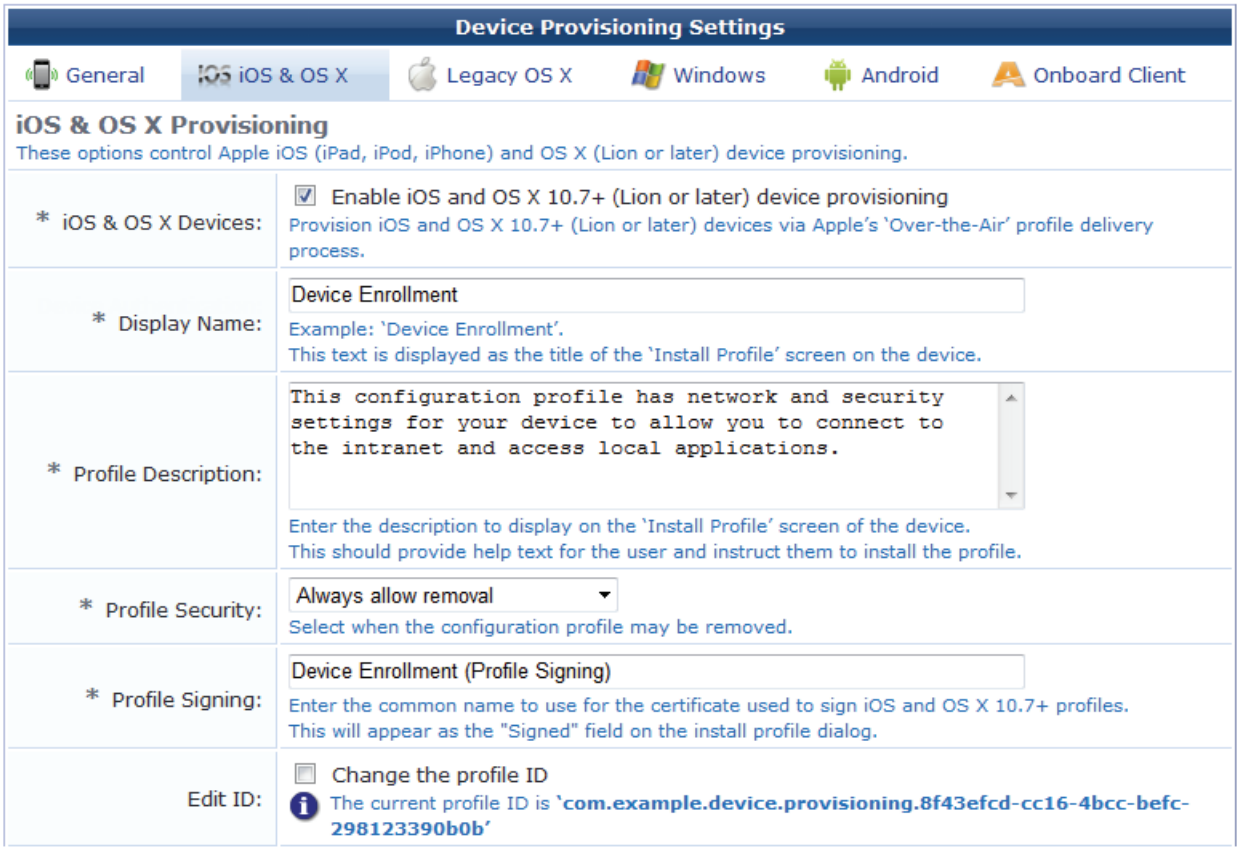
   Devices are recognized as unique when they have a different MAC address, or a different device identifier (when the MAC address is not available).

5. When your entries are complete in this tab, click **Save Changes**. You can click **Next** to continue to the next tab.

Configuring Provisioning Settings for iOS and OS X

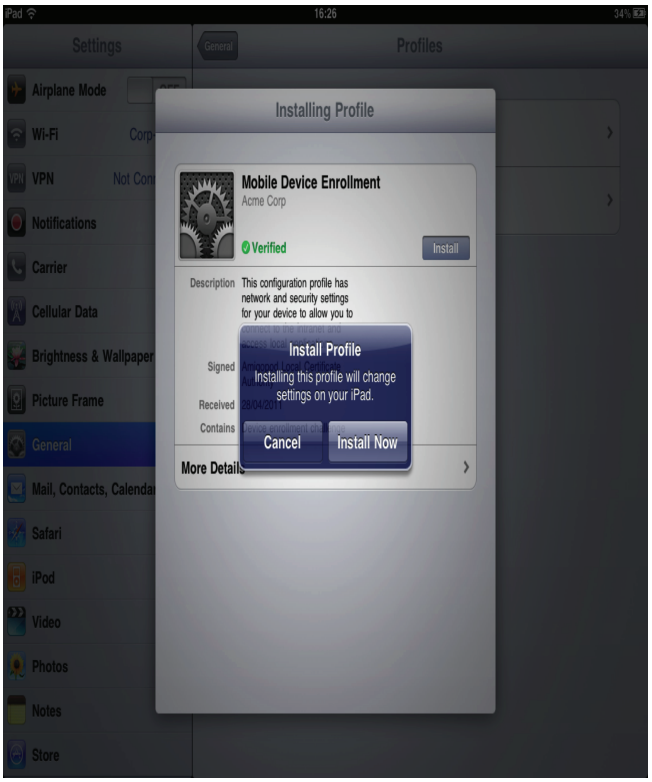
To specify provisioning settings related to iOS and OS X devices:

1. Go to **Onboard > Provisioning Settings** and click the **iOS & OS X** tab.



2. In the **iOS & OS X Devices** row, mark the **Enable iOS and OS X 10.7+ (Lion or later) device provisioning** check box to enable provisioning for these devices.

3. Use the **Display Name** and **Profile Description** text fields to control the user interface displayed during device provisioning.



4. In the **Profile Security** row, select one of the following options from the drop-down list to control how a device provisioning profile may be removed:
   - **Always allow removal** – The user may remove the device provisioning profile at any time, which will also remove the associated device configuration and unique device credentials.
   - **Remove only with authorization** – The user may remove the device provisioning profile if they also provide a password. The administrator must specify the password in the "Removal Password" and "Confirm Removal Password" fields.
   - **Never allow removal** – The user cannot remove the device provisioning profile. This option should be used with caution, as the only way to remove the profile is to reset the device to factory defaults, and destroy all data on the device.
5. Use the **Profile Signing** text field to specify the display name of the certificate used to sign the configuration profile. This certificate will be automatically created by the certificate authority, and appears as the "Sign**ed" field on the device when the user authorizes the device provisioning.**
6. In the **Edit ID** row, Mark the **Change the profile ID** check box to change the unique value associated with the configuration profile. This value is used to identify the configuration settings as being from a particular source, and should be globally unique.

   When an iOS device receives a new configuration profile that has the same profile ID as an existing profile, the existing profile will be replaced with the new profile.

   Note: Changing the profile ID will affect any device that has already been provisioned with the existing profile ID. The default value is automatically generated and is globally unique. You should only change this value during initial configuration of device provisioning.

Configuring Instructions for iOS and OS X

To edit the instruction text shown during provisioning for iOS and OS X devices:

1. Go to **Onboard > Provisioning Settings,** click the **iOS & OS X** tab, and scroll to the **Instructions** area of the form.
2. In the **Before Provisioning** text box, enter the instructions that are shown to the user before they provision their device. The text can be entered as HTML code, and you can use Smarty template functions. If this field is left empty, the default text will be displayed.
3. In the **After Provisioning** text box, enter the instructions that are shown to the user after they have provisioned their device. The text can be entered as HTML code, and you can use Smarty template functions. If this field is left empty, the default text will be displayed.
4. In the **iOS-4 Same SSID** text box, enter the instructions that are shown to the user of an iOS 4 device if they attempt to provision their device while connected to an SSID that will be provisioned. "Same SSID" provisioning is not supported. The text can be entered as HTML code, and you can use Smarty template functions. If this field is left empty, the default text will be displayed.

Configuring Reconnect Behavior for iOS and OS X

Reconnect is only supported by iOS 5+ and OS X 10.7+ (Lion or later) devices.

To configure the reconnect behavior iOS and OS X devices:

1. Go to **Onboard > Provisioning Settings,** click the **iOS & OS X** tab, and scroll to the **Reconnect** area of the form.
2. In the **Allow Automatic Reconnect** row, mark the check box if you want to allow the device to be automatically reconnected to the provisioned network. Automatic reconnect only applies when there is a single network configured to "Automatically join network."
3. In the **Allow Manual Reconnect** row, mark the check box if you want to allow the device to be manually reconnected to the provisioned network. Manual reconnect only applies when automatic reconnect is not allowed or not applicable.
4. In the **Manual Reconnect Interface** row, enter the text that will be shown to the user if manual reconnect is allowed and applicable. Enter the text as HTML code. You can use Smarty template functions. If this field is left empty, the default text will be displayed.
5. In the **Connect Success** row, enter the text that will be shown to the user after successful reconnect. Enter the text as HTML code. You can use Smarty template functions. If this field is left empty, the default text will be displayed.
6. In the **Connect Failure** row, enter the text that will be shown to the user after a failed reconnect or if the device does not support reconnection (for example, for iOS 4 and earlier devices). Enter the text as HTML code. You can use Smarty template functions. If this field is left empty, the default text will be displayed.
7. In the **After Connect** row, enter the text that will be shown after a reconnect attempt, regardless of success or failure. Enter the text as HTML code. You can use Smarty template functions. If this field is left empty, the default text will be displayed.

To configure delay and timeout settings:

1. Mark the check box in the **Advanced Settings** row. The form expands to include these options.
2. In the **Disconnect Delay** row, enter the duration in seconds for the Web server to wait after receiving a disconnect request before it sends the request to the controller. This delay gives the client time to receive a valid HTTP response before begin disconnected from the network.
3. In the **Reconnect Delay** row, enter the duration in seconds for the client to wait after sending a disconnect request to the Web server before it sends a reconnect request. This duration must give the Web server and the controller adequate time to negotiate a disconnect for the device first.

4. In the **Reconnect Timeout** row, enter the duration in seconds for the client to wait for a valid response after sending a reconnect request to the Web server. This duration must allow enough time for the client to be reconnected to the network (using the newly-installed settings) and for the Web server to then acknowledge the HTTP request.
5. When your entries are complete in this tab, click **Save Changes**. You can click **Next** to continue to the next tab, or **Previous** to return to the previous tab.

Configuring Provisioning Settings for Legacy OS X Devices

To specify provisioning settings related to legacy OS X 10.5 and 10.6 (Leopard and Snow Leopard) devices:

1. Go to **Onboard > Provisioning Settings** and click the **Legacy OS X** tab.

2. To enable provisioning OS X 10.5 and 10.6 devices, mark the check box in the **OS X 10.5/6 Devices** row.
3. In the **Before Provisioning** text box, enter the instructions that are shown to the user before they provision their device. The text can be entered as HTML code, and you can use Smarty template functions. If this field is left empty, the default text will be displayed.
4. In the **After Provisioning** text box, enter the instructions that are shown to the user after they have provisioned their device. The text can be entered as HTML code, and you can use Smarty template functions. If this field is left empty, the default text will be displayed.
5. You may use the **Insert content item** drop-down list to add an image file or other content item.
6. When your entries are complete in this tab, click **Save Changes**. You can click **Next** to continue to the next tab, or **Previous** to return to the previous tab.

Configuring Provisioning Settings for Windows Devices

To specify provisioning settings related to Windows devices:

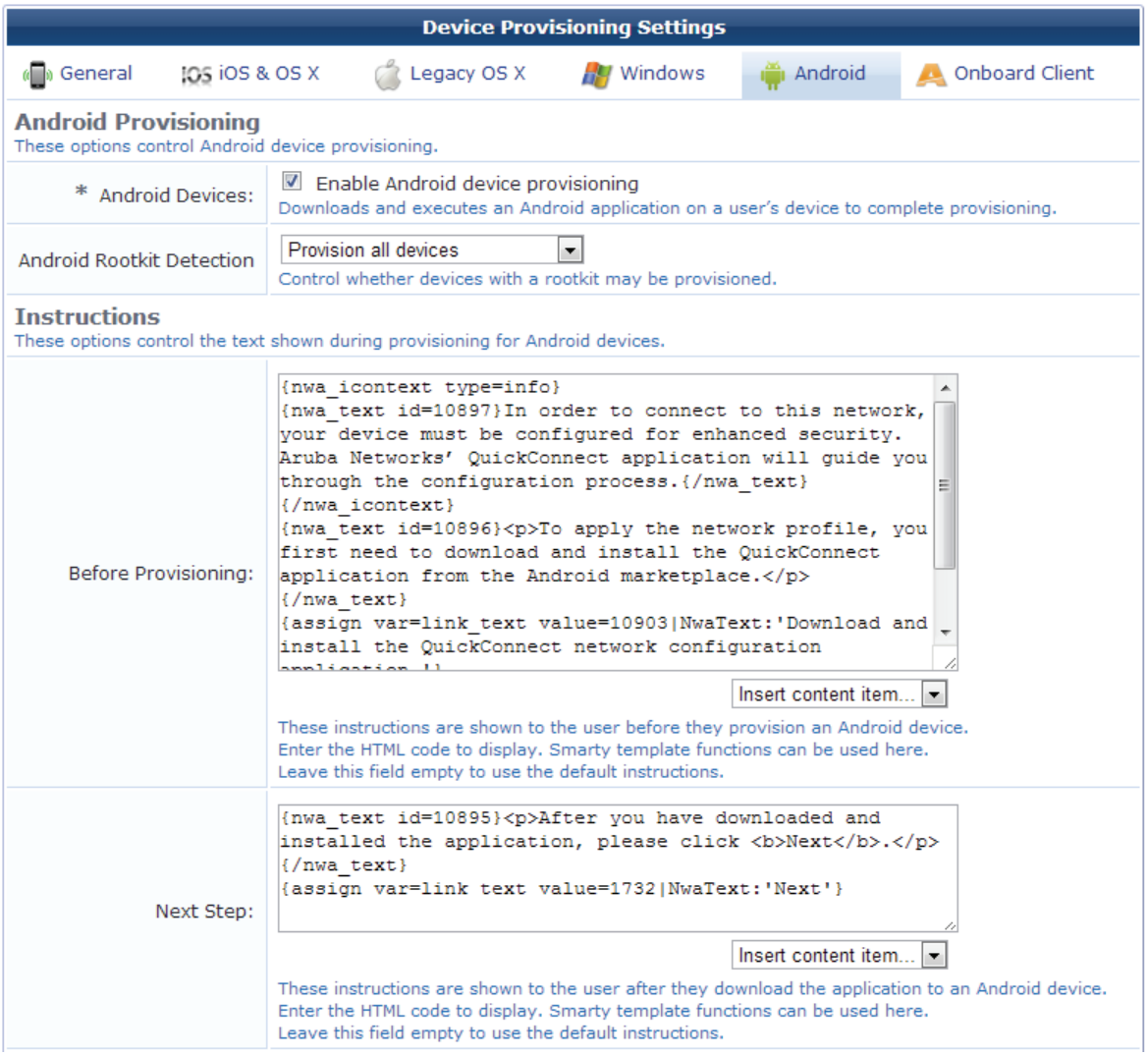
1. Go to **Onboard > Provisioning Settings** and click the **Windows** tab.

2. To enable provisioning Windows devices, mark the check box in the **Windows Devices** row.
3. In the **Code-Signing Certificate** drop-down list, select a certificate for signing the provisioning application, or leave the default setting of **None-Do not sign the application**.
4. In the **Before Provisioning** text box, enter the instructions that are shown to the user before they provision their device. The text can be entered as HTML code, and you can use Smarty template functions. If this field is left empty, the default text will be displayed.
5. In the **After Provisioning** text box, enter the instructions that are shown to the user after they have provisioned their device. The text can be entered as HTML code, and you can use Smarty template functions. If this field is left empty, the default text will be displayed.
6. You may use the **Insert content item** drop-down list to add an image file or other content item.
7. When your entries are complete in this tab, click **Save Changes**. You can click **Next** to continue to the next tab, or **Previous** to return to the previous tab.

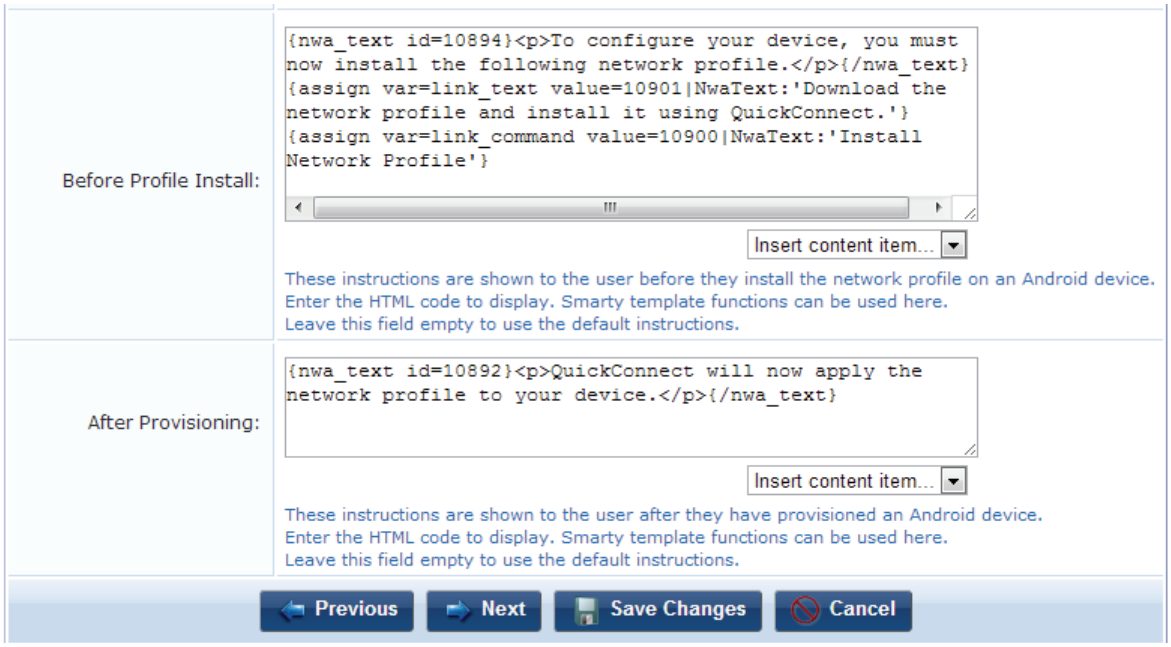
Configuring Provisioning Settings for Android Devices

To specify provisioning settings related to Android devices:

1. Go to **Onboard > Provisioning Settings** and click the **Android** tab.



2. To enable provisioning Android devices, mark the check box in the **Android Devices** row.
3. In the **Android Rootkit Detection** drop-down list, choose one of the following options:
   - **Provision all devices**— All Android devices will be provisioned.
   - **Do not provision rooted devices**—Onboard will detect a jailbroken Android device and will not provision the network if the device has been compromised.
4. In the **Before Provisioning** text box, enter the instructions that are shown to the user before they provision their device. The text can be entered as HTML code, and you can use Smarty template functions. If this field is left empty, the default text will be displayed.
5. In the **Next Step** text box, enter the instructions that are shown to the user after they download the application to their device. The text can be entered as HTML code, and you can use Smarty template functions. If this field is left empty, the default text will be displayed.



6. In the **Before Profile Install** text box, enter the instructions that are shown to the user before they install the network profile on their device. The text can be entered as HTML code, and you can use Smarty template functions. If this field is left empty, the default text will be displayed.
7. In the **After Provisioning** text box, enter the instructions that are shown to the user after they have provisioned their device. The text can be entered as HTML code, and you can use Smarty template functions. If this field is left empty, the default text will be displayed.
8. You may use the **Insert content item** drop-down list to add an image file or other content item.
9. When your entries are complete in this tab, click **Save Changes**. You can click **Next** to continue to the next tab, or **Previous** to return to the previous tab.

Configuring Options for Legacy OS X, Windows, and Android Devices

The Onboard Client tab is used to edit basic configuration option for Windows, Android, and legacy OS X (10.5 and 10.6) devices.

To specify provisioning settings related to these Onboard-capable devices:

1. Go to **Onboard > Provisioning Settings** and click the **Onboard Client** tab.

2. In the **Provisioning Address** drop-down list, choose the hostname or IP address to use for device provisioning:
   - **The system's hostname (requires DNS resolution)** – Select this option to use the system hostname for device provisioning.

     Note that this option requires that the device be able to resolve the listed hostname at the time the device is provisioned.
   - **The system's IP address (*network adapter name*)** – Select this option to use the IP address of the system for device provisioning. The drop-down list includes one option for each of the IP addresses detected on the system.

     Use this option when DNS resolution of the system's hostname is not available for devices that are in a provisioning role.

- **Other IP address or hostname**… – Select this option to override the hostname or IP address to be specified during device provisioning. The administrator must enter the hostname or IP address in the "Address" text field.

  Use this option when special DNS or NAT conditions apply to devices that are in a provisioning role.

3. If you chose **Other IP address or hostname** in the Provisioning Address drop-down list, use the **Address** field to enter a hostname or IP address.
4. The **Provisioning Access** warning message is displayed when HTTPS is not required for guest access. HTTPS is recommended for all deployments as it secures the unique device credentials that will be issued to the device.

   Note: When using HTTPS for device provisioning, you must obtain a commercial SSL certificate. Self-signed SSL certificates, and SSL server certificates that have been issued by an untrusted or unknown root certificate authority, will cause iOS device provisioning to fail with the message "The server certificate for … is invalid".
5. The **Validate Certificate** drop-down list is used to specify whether the SSL server's certificate should be validated as trusted. When this option is set to **Yes, validate this web server's certificate (recommended)**, a certificate validation failure on the client device will cause device provisioning to fail. This is the default option.

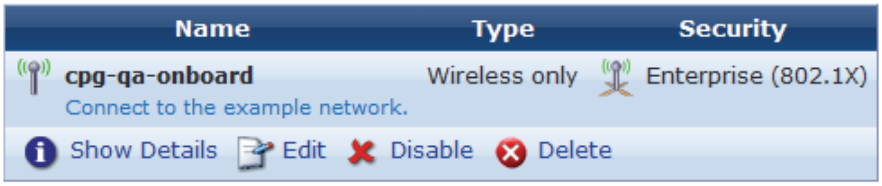
   You should change this option to **No, do not validate this web server's certificate** only during testing, or if you are waiting for a commercial SSL certificate.
6. To display your enterprise's logo, select an image from the list in the **Logo Image** field. Navigate to **Administration** > **Content Manager** to upload new images for use as the logo.

   The native size of the logo used in the QuickConnect client is 188 pixels wide, 53 pixels high. You may use an image of a different size and it will be scaled to fit, but for the best quality results it is recommended that you provide an image that is already the correct size.
7. The **Wizard Title** text field may be used to specify the text displayed to users when they launch the QuickConnect app to provision their device.
8. If provided, the **Password Recovery URL** and **Helpdesk URL** fields may be used to provide additional resources to users who encounter trouble in provisioning their devices.

   Note: Ensure that users in the provisioning role can access these URLs.
9. When your entries are complete in this tab, click **Save Changes**. You can click **Previous** to return to the previous tab.

Configuring Network Settings for Device Provisioning

To configure the network settings that will be sent to a provisioned device, go to **Onboard > Network Settings**, or click the **Network Settings** command link. The **Network Settings** list view opens.



All networks that have been provisioned are included in the list. To view details for a network, or to configure a network, click the network's row in the list. The row expands to include the **Show Details**, **Edit**, **Disable** or **Enable**, and **Delete** options.

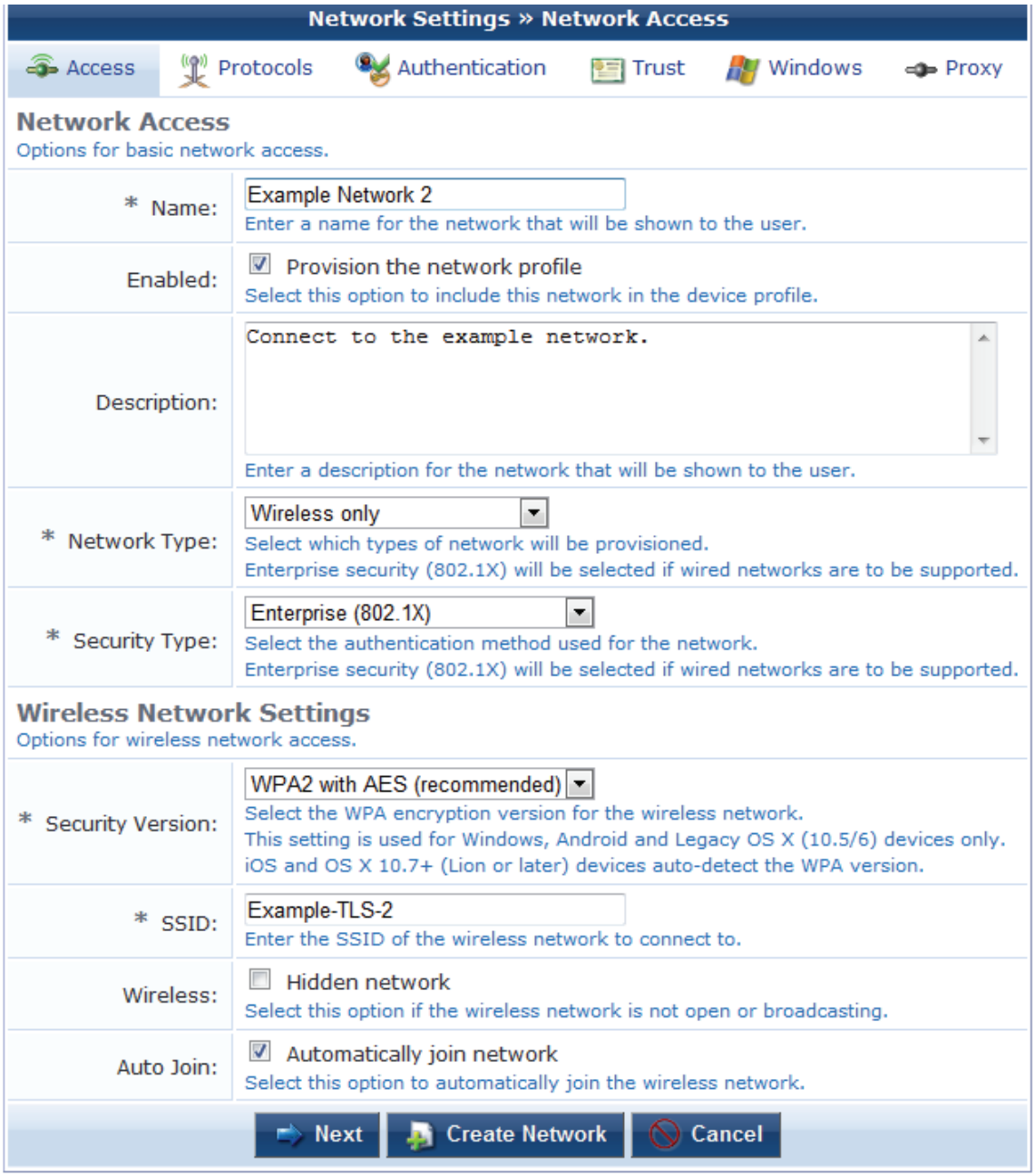
Configuring Basic Network Access Settings

1. To configure the network settings that will be provisioned to devices, click the network's **Edit** link. To create a new network, click the **Create new network** link in the upper-right corner. The Network Access form opens with the **Access** tab displayed.

   The configuration process is the same for editing an existing network and for creating a new network. The Network Access form is divided into several tabs:

   - **Access** – Specifies basic network properties, such as the name of the wireless network and the type of security that is used. See "Configuring Basic Network Access Settings".
   - **Protocols** – Specifies the 802.1X authentication protocols that are used by the network. See "Configuring 802.1X Authentication Network Settings".
   - **Authentication** – Specifies the type of device authentication to be used for the network. See "Configuring Device Authentication Settings".
   - **Trust** – Specifies options related to mutual authentication. See "Configuring Mutual Authentication Settings".
   - **Windows** – Specifies networking options used only by devices using the Windows operating system. See "Configuring Windows-Specific Network Settings".
   - **Proxy** – Specifies a proxy server to be used by devices connecting to the network. See "Configuring Proxy Settings".

   Note: Navigating between different tabs will save the changes you have made. The modified settings are indicated with a "#" marker in the tab. The settings used for device provisioning are not modified until you click **Create Network.**



2. To edit the network's basic and wireless network access options, click the **Access** tab.
3. If you need to edit the network's name, enter the new name in the **Name** field.
4. You can use the check box in the **Enabled** row to enable or disable the network in the device profile.
5. (Optional) You may enter additional identifying information in the **Description** field.
6. The options available in the **Network Type** drop-down list are:
   - **Both — Wired and Wireless** – Configures both wired (Ethernet) and wireless network adapters. Use this option when you have 802.1X configured for all types of network access.
   - **Wireless only** – Configures only wireless network adapters.
   - **Wired only** – Configures only wired (Ethernet) network adapters.
7. The options available in the **Security Type** drop-down list are:
   - **Enterprise (802.1X)** – Use this option to setup a network that requires user authentication.
   - This option is the only available choice when the Network Type is set to "Wired only".

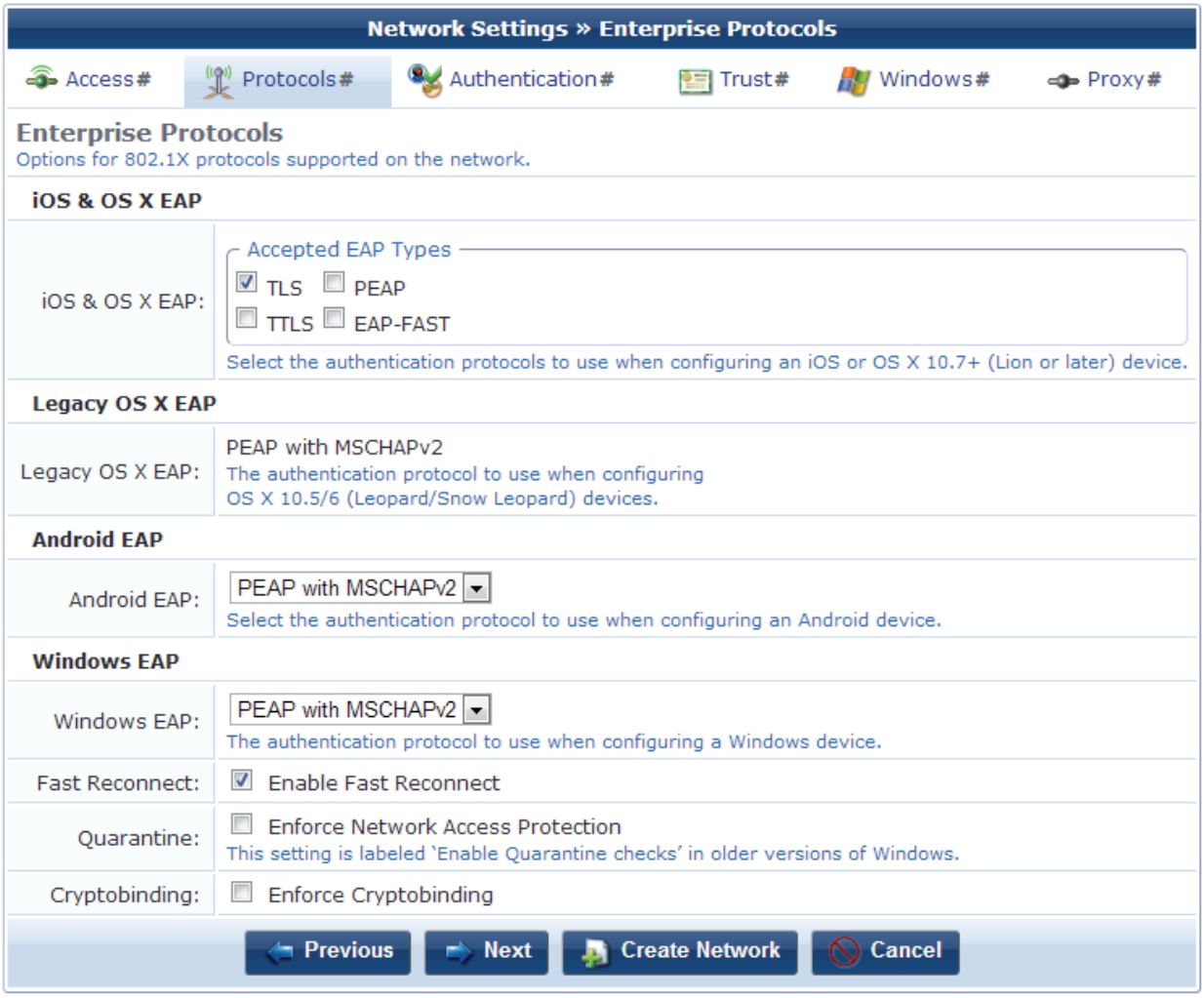
- **Personal (PSK)** – Use this option to setup a network that requires only a pre-shared key (password) to access the network.

This option is only available when the Network Type is set to “Wireless only”.

8. The **Security Type** field lets you set the encryption version for the wireless network to **WPA** or **WPA2**.
9. If you have selected the **Personal (PSK)** security type, you must provide the pre-shared key in the **Password** field. Selecting this security type will hide the **Protocols**, **Authentication**, and **Trust** tabs.
10. In the **Wireless Network Settings** area:
    - The drop-down list in the **OS X Profile** row allows you to select the type of profile to create when an OS X 10.7 (or later) device is provisioned. To create a per-user profile, select the **User** option. To create a system profile, select the **System** option. The System option can be used in settings where the device has several users and a single profile might be preferred to individual user profiles—for example, where an iMac in a high school classroom is used by all the students.
    - In the **Auto Join** row, you can mark the **Automatically join network** check box to specify that the device should be automatically connected to the network when it is provisioned. If only one network is available to the user, the device will be connected automatically. If multiple networks are available, the user will be able to choose the network to connect to. If the **Automatically join network** option is not selected on this form, an option to manually connect to the network will be shown to the user.
11. Do one of the following:
    - Click the **Next** button to continue to the **Protocols** tab.
    - Click the **Create Network** button to make the new network configuration settings take effect
    - Click the **Cancel** button to discard your changes and return to the main Onboard configuration user interface.

Configuring 802.1X Authentication Network Settings

Click the **Protocols** tab to display the Enterprise Protocols form.



Use this form to specify the authentication methods required by your network infrastructure.

- The **iOS & OS X EAP** option supports TLS, TTLS, PEAP, and EAP-FAST.
- The **Legacy OS X EAP** option supports only PEAP with MSCHAPv2.
- The **Android EAP** option supports PEAP with MSCHAPv2, PEAP with GTC, TTLS with MSCHAPv2, TTLS with GTC, TTLS with PAP, and TLS.
- The **Windows EAP** option supports PEAP with MSCHAPv2 and TLS.

These best practices are recommended when choosing the 802.1X authentication methods to provision:

- Configure PEAP with MSCHAPv2 for Onboard devices – Android, Windows, and legacy OS X (10.5/10.6).
- Configure EAP-TLS for iOS devices and OS X (10.7 or later).
- Other EAP methods, while possible, are limited in their applicability and should only be used if you have a specific requirement for that method.

The **Windows EAP** options that may be specified include:

- **Enable Fast Reconnect** – Fast Reconnect is a PEAP property that enables wireless clients to move between wireless access points on the same network without being re-authenticated each time they associate with a new access point. If TLS is selected, Fast Reconnect is not available.
- **Enforce Network Access Protection**– Enable this option to obtain a system statement-of-health (SSoH) from the OnGuard or Microsoft NAP Agent and send it to the authentication server during the

802.1X authentication process. Use this option to enforce network access control (NAC) protections on the network. If TLS is selected, Enforce Network Access Protection is not available.

- **Enforce Cryptobinding** – Cryptobinding is a process that protects the authentication protocol negotiation against man-in-the-middle attacks. The cryptobinding request and response performs a two-way handshake between the peer and the authentication server using key materials. If TLS is selected, Enforce Cryptobinding is not available.
- Do one of the following:
  - Click the Previous button to return to the Access tab.
  - Click the Next button to continue to the Authentication tab.
  - Click the **Create Network** button to make the new network configuration settings take effect
  - Click the Cancel button to discard your changes and return to the main Onboard configuration user interface.

Configuring Device Authentication Settings

Click the **Authentication** tab to display the Enterprise Authentication form.

1. Select one of these options in the **iOS & OS X Credentials** drop-down list:
   - **Certificate** – A device certificate will be provisioned and used for EAP-TLS client authentication. When this option is selected, **EAP-TLS** must be selected on the **Protocols** tab.
   - **Username & Password** – A device certificate will be provisioned, but the client authentication will use unique device credentials (as for Onboard devices). When this option is selected, **EAP-TTLS** or **PEAP** must be selected on the **Protocols** tab.
2. The **Windows Authentication** options that may be selected are:
   - **Machine Only** – Use computer-only credentials.
   - **User Only** – Use user-only credentials
   - **Machine Or User** – Use computer-only credentials or user-only credentials. When a user is logged on, the user's credentials are used for authentication. When no user is logged on, computer-only credentials are used for authentication.
   - **Guest –** Use guest-only credentials.
3. Do one of the following:
   - Click the **Previous** button to return to the Protocols tab.

- Click the **Next** button to continue to the Trust tab.
- Click the **Create Network** button to make the new network configuration settings take effect
- Click the **Cancel** button to discard your changes and return to the main Onboard configuration user interface.

Configuring Mutual Authentication Settings

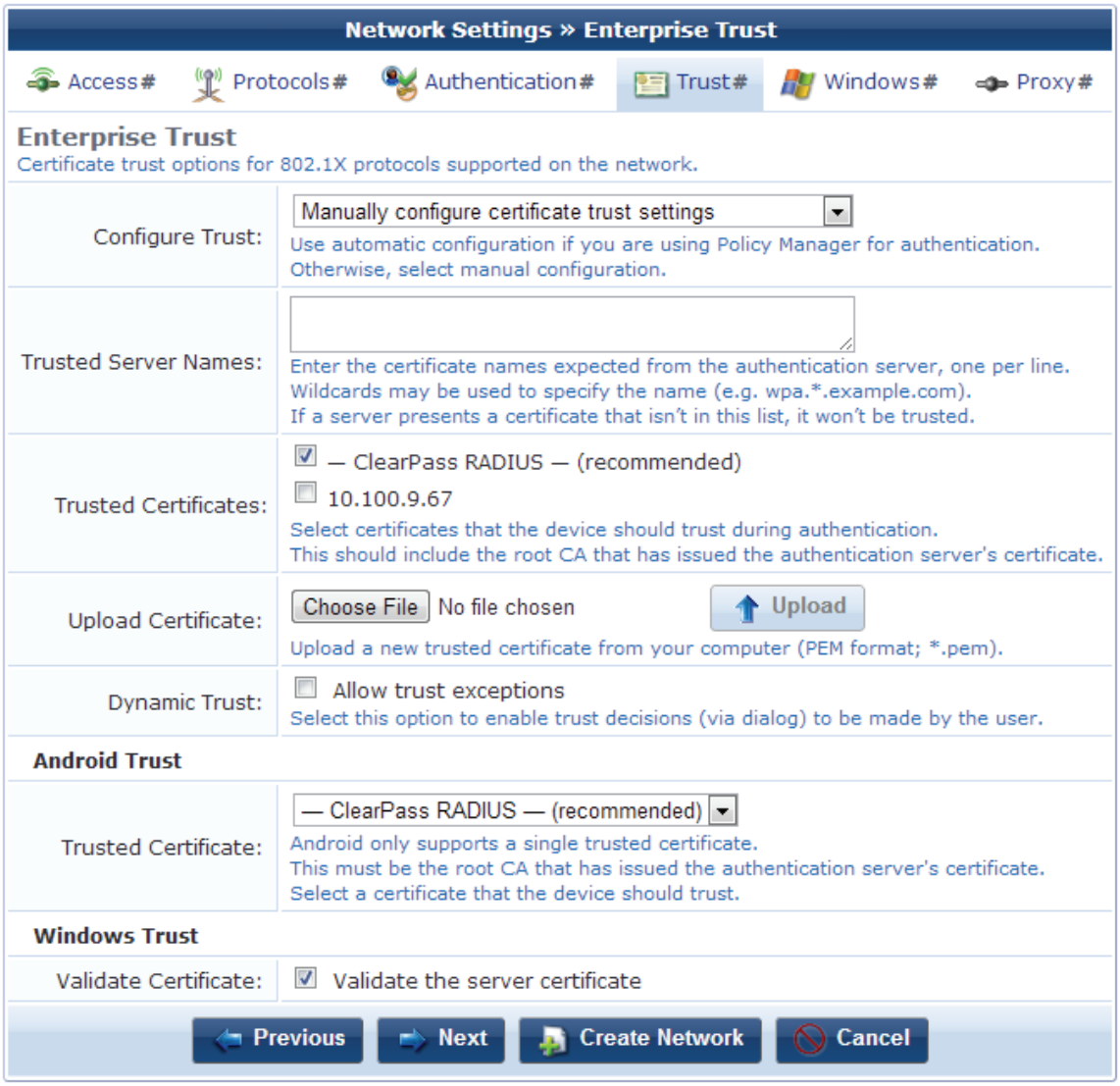
Click the **Trust** tab to display the Enterprise Trust form. Use this form to create the network settings that will be sent to a provisioned device.

Configuring Trust Settings Automatically

1. When you open this tab, the default selection in the **Configure Trust** field is **Automatically configure trust settings (recommended)**. With this option selected, Onboard automatically determines the appropriate certificate trust configuration for your deployment.
2. If the deployment is not using the built-in CA, you may use the **Trusted Server Names** text field to enter the certificate names to accept from the authentication server. Only certificates included in this list will be trusted. Enter each server name on a separate line. You can use wildcards.
3. Do one of the following:
   - Click the **Previous** button to return to the Authentication tab
   - Click the **Next** button to continue to the Windows tab
   - Click the **Create Network** button to make the new network configuration settings take effect
   - Click the **Cancel** button to discard your changes and return to the main Onboard configuration user interface

Configuring Trust Settings Manually

1. To change the recommended default setting and configure trust settings manually, choose **Manually configure certificate trust settings** in the **Configure Trust** drop-down list. The form expands to include configuration options.



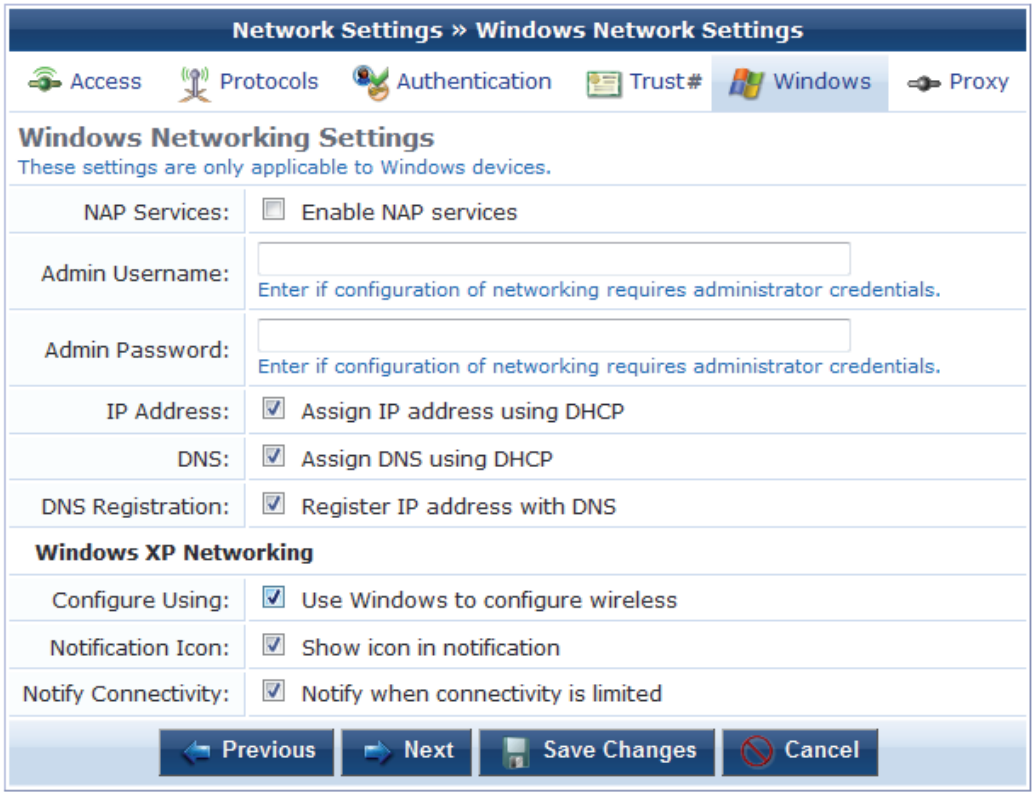
2. If the deployment is not using the built-in CA, you may use the **Trusted Server Names** text field to enter the certificate names to accept from the authentication server. Only certificates included in this list will be trusted. Enter each server name on a separate line. You can use wildcards.
3. In the **Trusted Certificates** row, mark the check box for each server certificate that the client should trust. You should include the root certificate that issued the authentication server's certificate, and you should provide the certificate for each authentication server a provisioned device will use.
4. You can use the **Upload Certificate** options to import additional trusted certificates or certificate signing requests. Click **Choose File** to navigate to the file on your computer, then click **Upload**. The certificate is imported, and the certificate name is displayed above the form. You can click the **Show certificate** link next to the name to view certificate details. The certificate is also displayed in the Certificate Management list with the type "trusted."
5. In the **Dynamic Trust** row, you should avoid marking the **Allow trust exceptions** check box – the network administrator should make all trust decisions. Users will not generally review certificates for potential issues before accepting them. If you wish to enable trust decisions to be made by the user, you

may unmark the **Allow trust exceptions** check box. Be aware that this is an insecure configuration, as a user can override a security warning if a man-in-the-middle attack occurs.

6. In the **Android Trust** area, use the **Trusted Certificate** drop-down list to select a certificate the device should trust. Android supports only a single trusted certificate; this must be the root CA that issued the authentication server's certificate. Be aware that if **None** is selected, 802.1x authentication might not work.
7. In the **Windows Trust** area, mark the **Validate the server certificate** check box. This ensures that the provisioned device will check the server certificate is valid before using the server for authentication. If this check box is unmarked, the configuration will not be secure. An attacker could provide another server certificate which the client would not verify.
8. Do one of the following:
   - Click the **Previous** button to return to the Authentication tab.
   - Click the **Next** button to continue to the Windows tab.
   - Click the **Create Network** button to make the new network configuration settings take effect
   - Click the **Cancel** button to discard your changes and return to the main Onboard configuration user interface.

Configuring Windows-Specific Network Settings

Click the **Windows** tab to display the Windows Network Settings form.



Network Access Protection (NAP) is a feature in Windows Server 2008 that controls access to network resources based on a client computer's identity and compliance with corporate governance policy. NAP allows network administrators to define granular levels of network access based on who a client is, the groups to which the client belongs, and the degree to which that client is compliant with corporate governance policy. If a client is not compliant, NAP provides a mechanism to automatically bring the client back into compliance and then dynamically increase its level of network access.

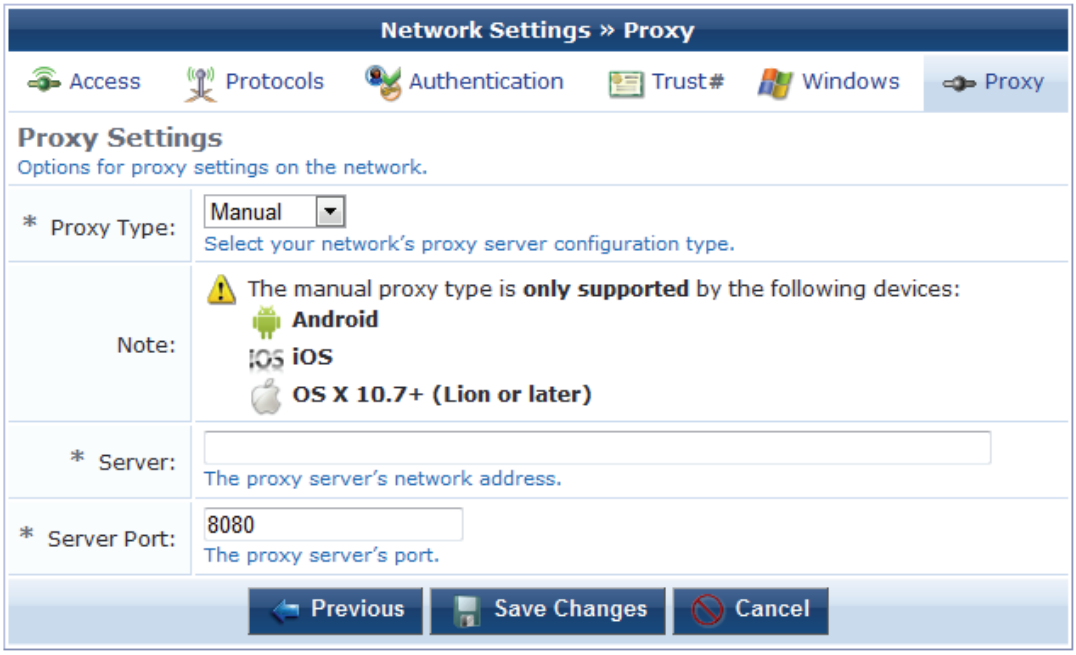
Deploying NAP requires a NAP-compatible authentication server, so that appropriate policies may be implemented based on the statement of health provided by the NAP client.

To enable NAP for Microsoft Windows clients, mark the **Enable NAP services** check box on this tab. You will also need to mark the **Enable Quarantine Checks** check box on the **Protocols** tab.

- Do one of the following:
  - Click the **Previous** button to return to the Trust tab.
  - Click the **Next** button to continue to the Proxy tab.
  - Click the **Create Network** button to make the new network configuration settings take effect
  - Click the **Cancel** button to discard your changes and return to the main Onboard configuration user interface.

Configuring Proxy Settings

Click the **Proxy** tab to display the Proxy Settings form.

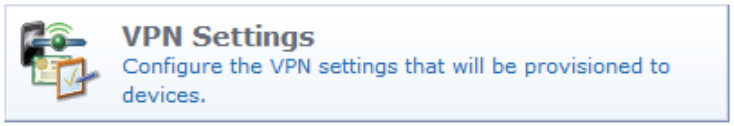


Select one of these options in the **Proxy Type** drop-down list:

- **None** – No proxy server will be configured.
- **Manual** – A proxy server will be configured, if the device supports it. Specify the proxy server settings in the **Server** and **Server Port** fields.
- **Automatic** – The device will configure its own proxy server, if the device supports it. Specify the location of a proxy auto-config file in the **PAC URL** text field.
- Do one of the following:
  - Click the **Previous** button to return to the Windows tab.
  - Click the **Create Network** button to make the new network configuration settings take effect
  - Click the **Cancel** button to discard your changes and return to the main Onboard configuration user interface.

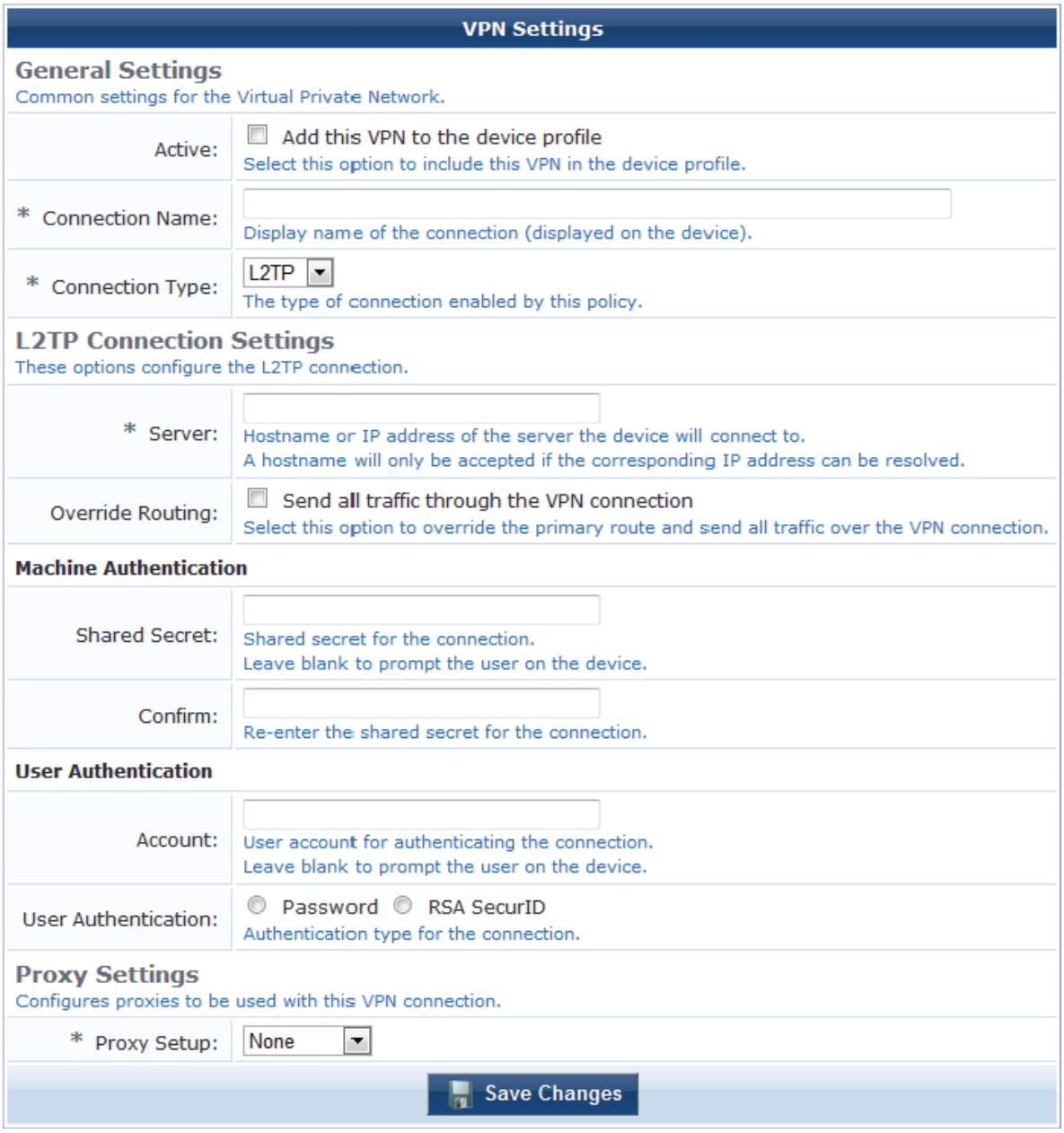
Configuring an iOS Device VPN Connection

To configure the VPN settings that will be sent to a device, go to **Onboard > VPN Settings**, or click the **VPN Settings** command link. The VPN Settings page opens.



This page is used to automatically configure virtual private networking (VPN) settings on the iOS device. Use this option when you have deployed a VPN infrastructure and want to automatically provide the secure connection settings to users at the time of device provisioning.

Note: Onboard VPN settings can only be used with iOS 4 and iOS 5 devices. Other platforms are not supported.



Mark the **Add this VPN to the device profile** check box to enable provisioning of VPN settings.

The **Display Name** text field specifies the name for this VPN connection. This will be displayed on the device in the Settings app. To help the user identify the connection easily, include your organization's name in the Display Name field. For example, use "ACME Sprockets VPN".

Select the appropriate **Connection Type** from the drop-down list:

- **L2TP** – Connection uses the Layer 2 Tunneling Protocol. Complete the fields shown in the L2TP Connection Settings section of the form.
- **PPTP** – Connection uses the Point-to-Point Tunneling Protocol. Complete the fields shown in the PPTP Connection Settings section of the form.
- **IPSec** – Connection uses the Internet Protocol with security extensions. Complete the fields shown in the IPSec Connection Settings section of the form.

  The **Authentication Type** drop-down list provides these options when configuring an IPSec VPN:

  - **Identity Certificate** – The client certificate issued during device provisioning will also be used as the identity certificate for VPN connections. This option requires configuring your VPN server to allow IPSec authentication using a client certificate.
  - **Shared Secret / Group Name** – An optional group name may be specified. A shared secret (pre-shared key) is used to establish the IPSec VPN. Authentication is performed with a username and password.

The Proxy Settings section of the form specifies a proxy server that is used when the VPN connection is active. Select one of these options in the **Proxy Setup** drop-down list:

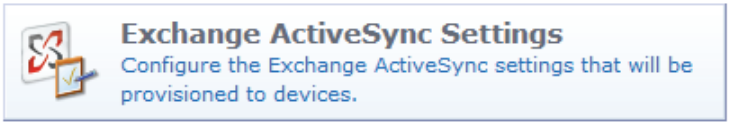
- **None** – No proxy server will be configured with this VPN profile.
- **Manual** – A proxy server will be configured with this VPN profile. Specify the proxy server settings in the **Server** and **Port** fields.

  If authentication is required to access this proxy, you may specify the username and password using the **Authentication** and **Password** text fields.
- **Automatic** – The proxy server will be automatically configured with this VPN profile. Specify the location of a proxy auto-config file in the **Proxy Server URL** text field.

Click the **Save Changes** button to save the VPN connection profile and return to the main Onboard configuration user interface.

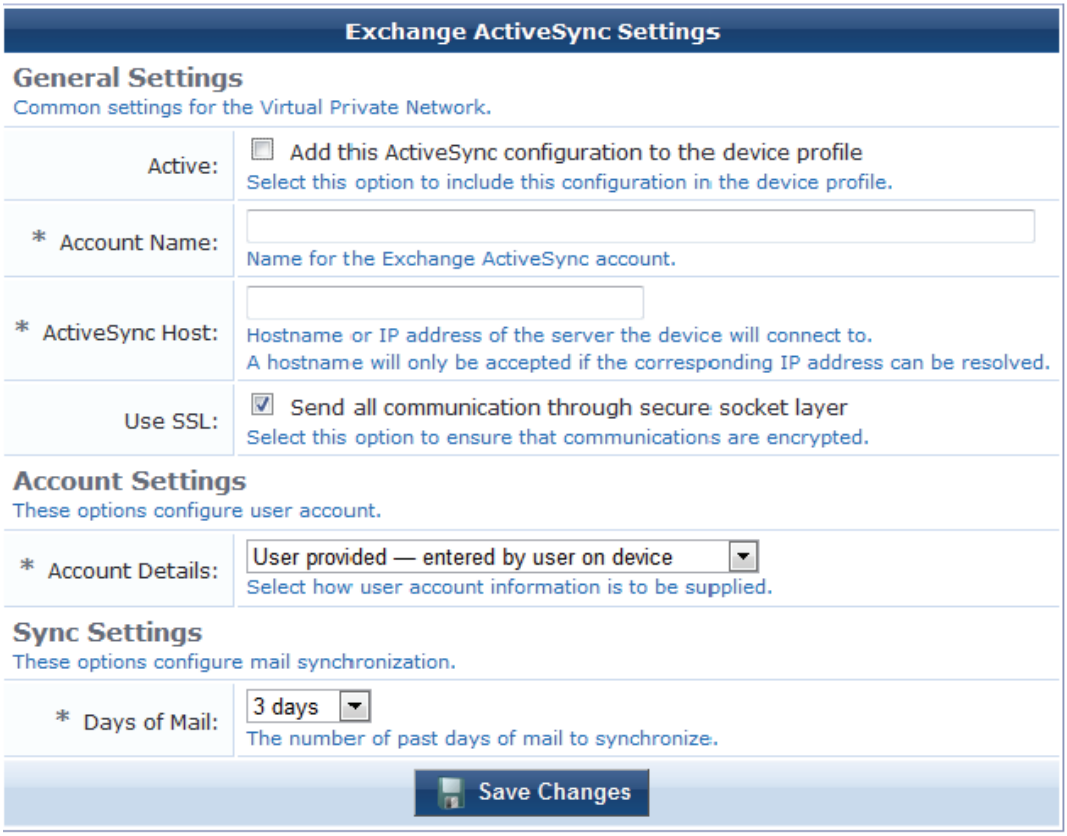
Configuring an iOS Device Email Account

To configure the Exchange ActiveSync settings that will be sent to a device, go to **Onboard > Exchange ActiveSync**, or click the **Exchange ActiveSync** command link. The Exchange ActiveSync Settings page opens.



This page is used to automatically configure an email account on the iOS device. Use this option when you have an Exchange mail server and want to automatically provide the email settings to users provisioning their mobile devices.

Note: Onboard Exchange ActiveSync settings can only be used with iOS 4 and iOS 5 devices. Other platforms are not supported.



Mark the **Add this ActiveSync configuration to the device profile** check box to enable email account provisioning.

The **Account Name** text field specifies the name for this email account. This will be displayed on the device in the Settings app, and also within the Mail app to identify the mailbox. To help the user identify this mailbox easily, include your organization's name in the Account Name field. For example, use "ACME Sprockets Mail".

In the **Account Settings** group, choose one of the following options from the **Account Details** drop-down list:

- **User provided — entered by user on device.** This option requires the user to enter their credentials on the device to access their email.
- **Identity certificate — created during provisioning.** This option uses the device's TLS client certificate to authenticate the user. Using this option requires configuration of the ActiveSync server to authenticate a user based on the client certificate.
- **Shared preset values — testing only**. This option provides a fixed set of credentials to the device. These settings cannot be modified for each user when provisioning a device, so it is recommended that these settings only be used when testing Exchange integration.

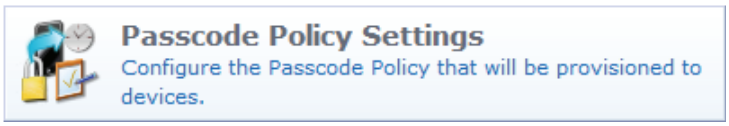
In the **Sync Settings** group, choose one of the following options from the **Days of Mail** drop-down list:

- No Limit
- 1 day
- 3 days
- 1 week
- 2 weeks
- 1 month

Click the **Save Changes** button to save the Exchange ActiveSync profile and return to the main Onboard configuration user interface.

Configuring an iOS Device Passcode Policy

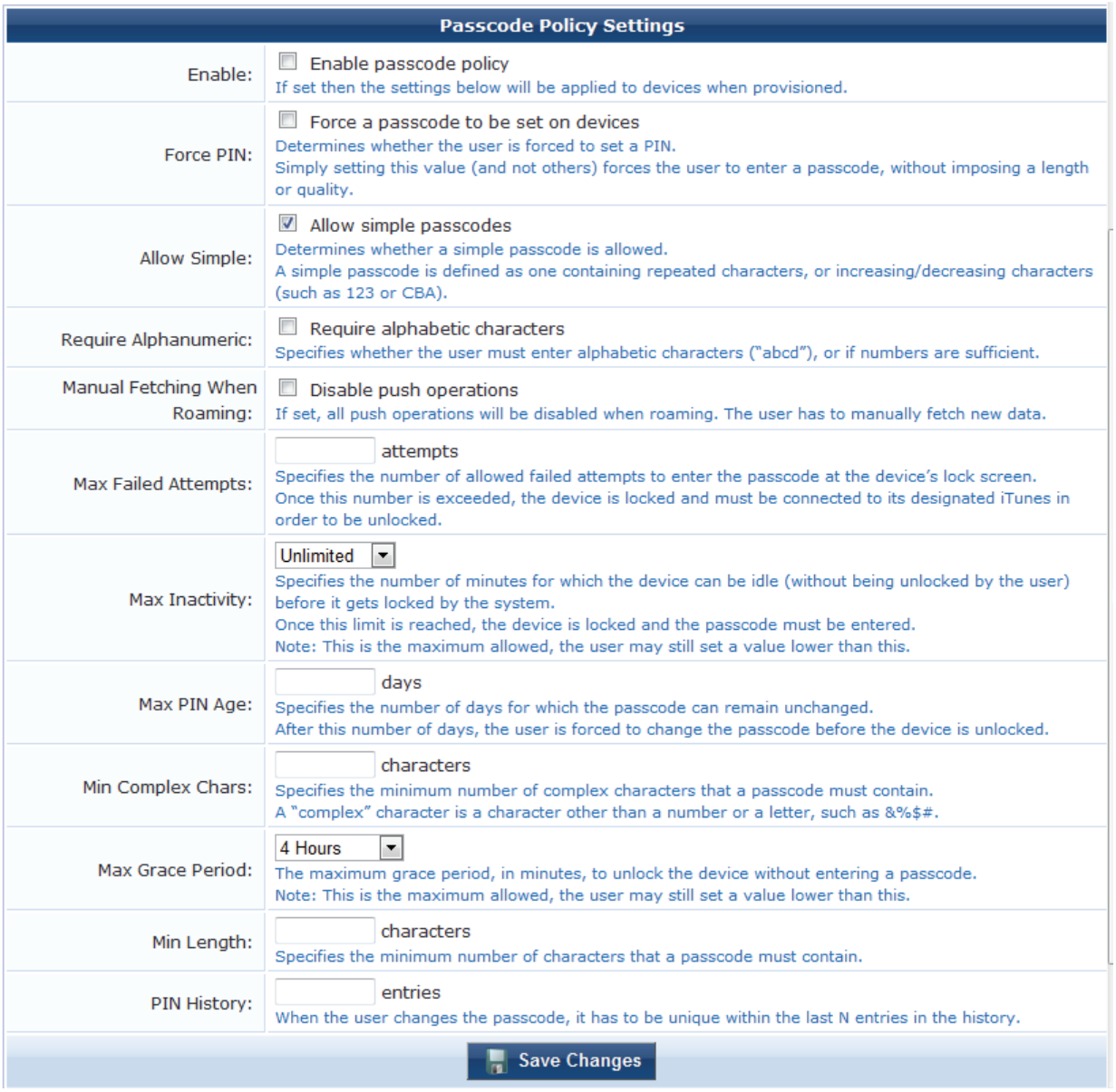
To make changes to the Passcode Policy configuration that will be sent to a device, go to **Onboard > Passcode Policy**, or click the **Passcode Policy** command link. The Passcode Policy Settings page opens.



This page is used to configure a passcode policy that is applied to iOS devices when provisioned.

Typically, you would enable this policy when provisioning a corporate-owned device, or if you are allowing a user to access sensitive information remotely.

NOTE: Onboard Passcode Policy settings can only be used with iOS 4 and iOS 5 devices. Other platforms are not supported.



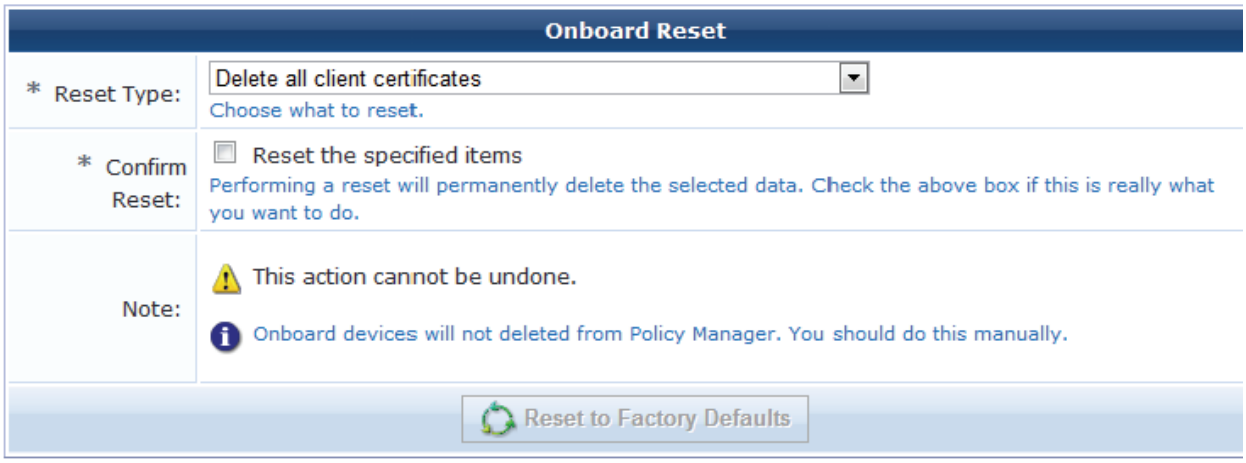
To enable the passcode policy on all iOS devices, mark the **Enable passcode policy** check box and configure the remaining options according to your enterprise's security requirements.

Click the **Save Changes** button to save the passcode policy settings and return to the main Onboard configuration user interface.

Resetting Onboard Certificates and Configuration

To delete certificates, re-create the Onboard Web login page, or reset configuration to factory default settings, go to **Onboard > Reset to Factory Defaults**, or click the **Reset to Factory Defaults** command link. The Reset to Factory Defaults page opens.

This page is used to delete certificates, or restore the default configuration for Onboard. These options are useful while trailing the Onboard workflow with a set of test devices.



Select one of the following options in the **Reset Type** drop-down list:

- **Delete all client certificates** – Removes all client certificates from Certificate Management. The certificate authority's root certificate, intermediate certificate, profile signing certificate, and any server certificates are not affected. The provisioning settings for iOS and Onboard-capable devices are not modified.
- **Delete all certificates** – Removes all certificates from Certificate Management, including the certificate authority's root certificate, intermediate certificate, profile signing certificate, and any server certificates. The default certificate authority certificate will be recreated. The provisioning settings for iOS and Onboard-capable devices are not modified.
- **Re-create the Onboard weblogin page** – Select this option to create the default device_provisioning Web login page, if it has been deleted or has been modified and no longer functions correctly. All certificates and settings are left unmodified.
- **Delete all certificates and reset configuration to factory defaults** – Removes all certificates from Certificate Management, including the certificate authority's root certificate, intermediate certificate, profile signing certificate, and any server certificates. The provisioning settings for iOS and Onboard-capable devices are restored to the default settings. The default certificate authority will be recreated.

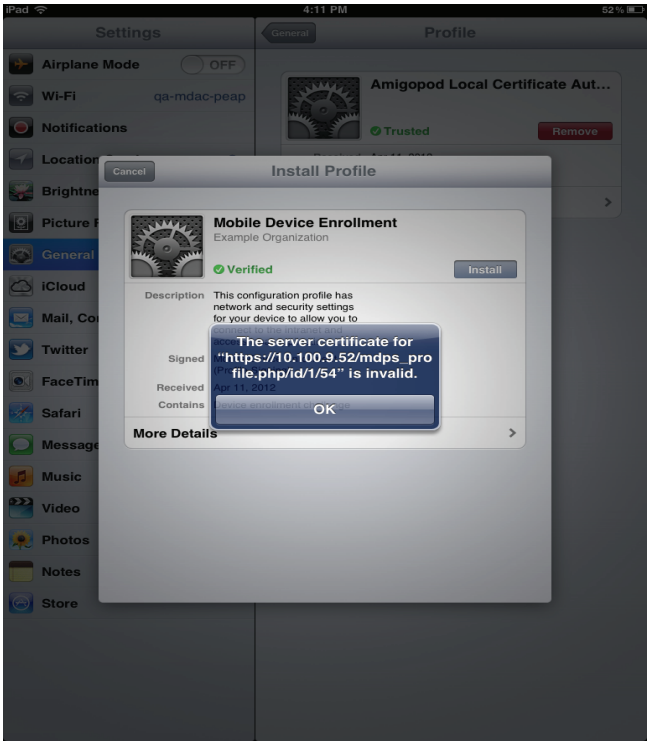
Mark the **Reset the specified items** check box to indicate that the reset operation should be performed, and then click **Reset to Factory Defaults** to perform the operation.

Onboard Troubleshooting

If you encounter a problem that is not listed here, refer to the "Onboard Deployment Checklist" and check each of the configuration steps listed there.

iOS Device Provisioning Failures

Symptom: Device provisioning fails on iOS with the message "The server certificate for https://… is invalid".



Resolution: When using HTTPS for device provisioning, you **must** obtain a commercial SSL certificate.

Self-signed SSL certificates, and SSL server certificates that have been issued by an untrusted or unknown root certificate authority, will cause iOS device provisioning to fail with the message "The server certificate for … is invalid".

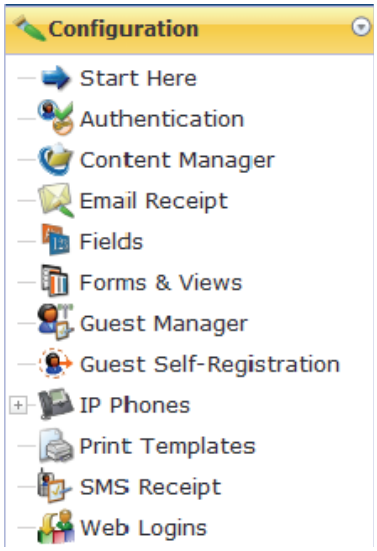
A workaround for this issue is to install an appropriate root certificate on the iOS device. This root certificate must be the Web server's SSL certificate (if it is a self-signed certificate), or the certificate authority that issued the SSL certificate. This is not recommended for production deployments as it increases the complexity of deployment for users with iOS devices.

ClearPass Guest's built-in Configuration editor lets you customize many aspects of the appearance, settings, and behavior of the application. Areas you can customize include:

- Guest Manager configuration
- Fields, forms, and views in ClearPass Guest
- Guest self-registration processes and forms
- Format and appearance of visitor account receipts
- Settings for emailing visitor account receipts
- Self-provisioning features of your wireless network
- Content asset management
- Visitor account provisioning services for IP phones
- SMS visitor account receipt settings
- Web login pages

Accessing Configuration

To access ClearPass Guest's application customization features, click the **Configuration** link in the left navigation.

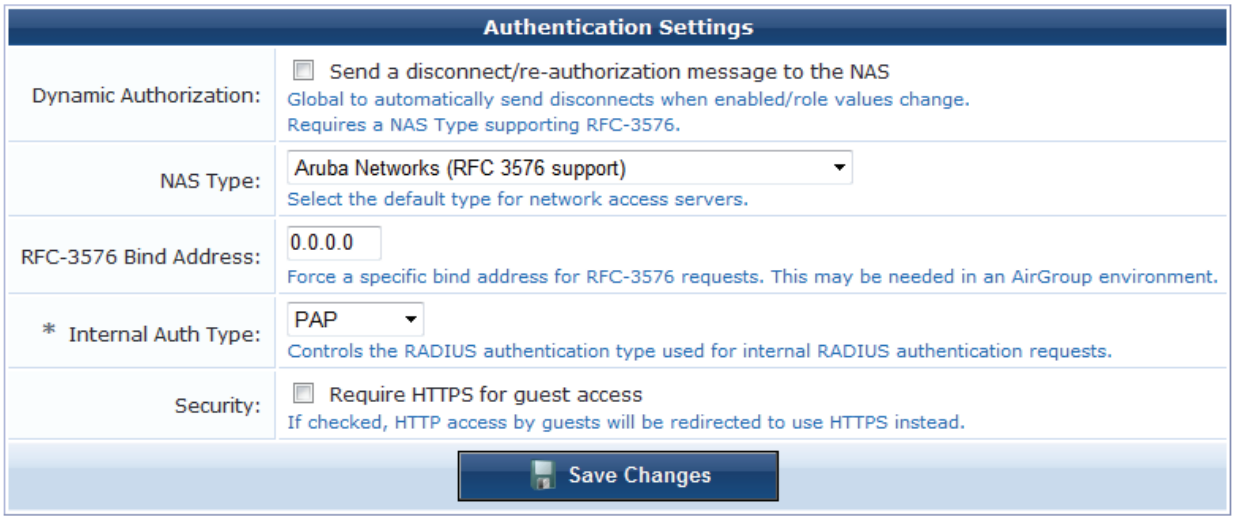


Configuring ClearPass Guest Authentication

You can use the Configuration module to modify authentication settings for the ClearPass Guest application.

To configure ClearPass Guest's authentication settings:

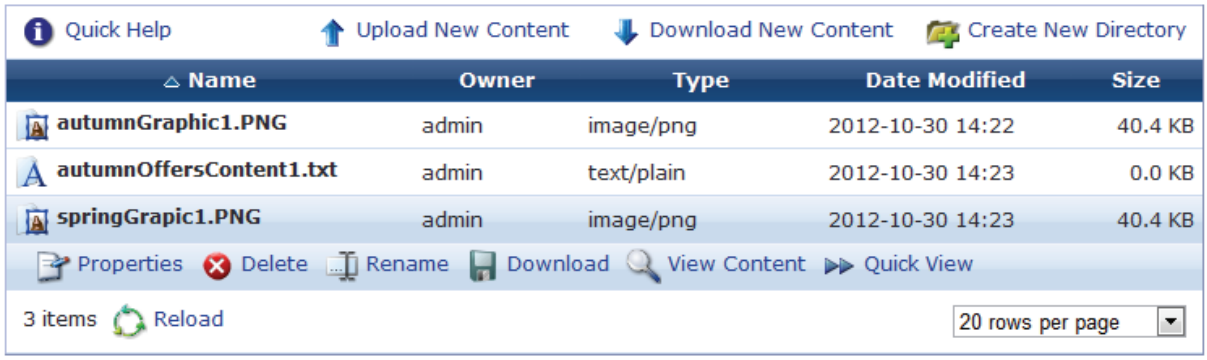
1. Go to **Configuration > Authentication**. The Authentication Settings form opens.



2. To send automatic disconnect or re-authorization messages when enabled or role values change, mark the check box in the **Dynamic Authorization** row. This requires a network access server (NAS) type that supports RFC-3576.
3. In the **NAS Type** row, use the drop-down list to choose the default type for network access servers.
4. To force a specific bind address for RFC-3576 requests, enter a value in the **RFC-3576 Bind Address** row. This might be needed in an AirGroup environment.
5. In the **Internal Auth Type** row, choose a type from the drop-down list. Choices in list include **PAP**, **CHAP**, and **MS-CHAP**. The internal authentication type controls the RADIUS authentication used for internal RADIUS requests.
6. To redirect HTTP access to use HTTPS instead, mark the check box in the **Security** row.

Content Manager

The Content Manager allows you to upload content items to ClearPass Guest. Content items are assets such as text, images, and animations that are made available for guest access using the application's built-in Web server. To work with your content items, go to **Configuration > Content Manager**.



You can add content items by using your Web browser to upload them. You can also copy a content item stored on another Web server by downloading it.

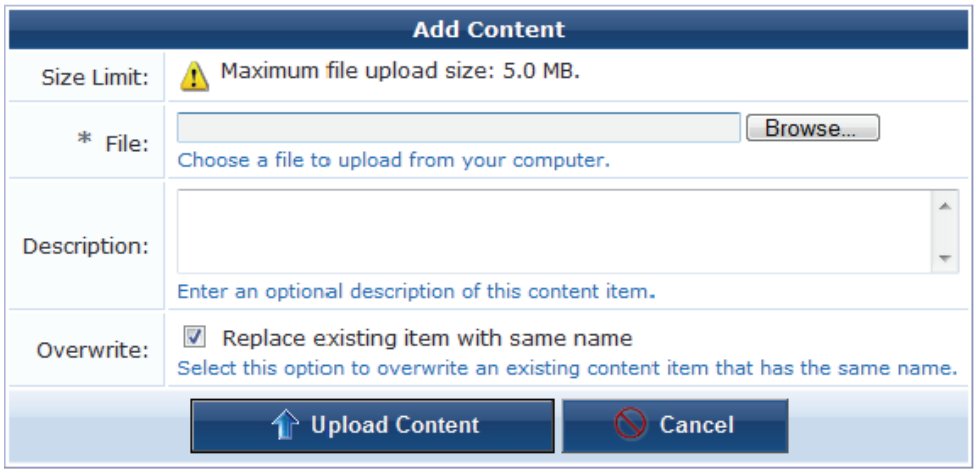
To use a content item, you can insert a reference to it into any custom HTML editor within the application. To do this, select the content item you want to insert from the drop-down list located in the lower right corner of the editor. The item will be inserted using HTML that is most suited to the type of content inserted.

To manually reference a content item, you can use the URL of the item directly. For example, an item named **logo.jpg** could be accessed using a URL such as: **http://192.168.88.88/public/logo.jpg**.

Uploading Content

To add a new content item using your Web browser:

1. Go to **Configuration > Content Manager**, then click the **Upload New Content** tab. The Add Content form opens.



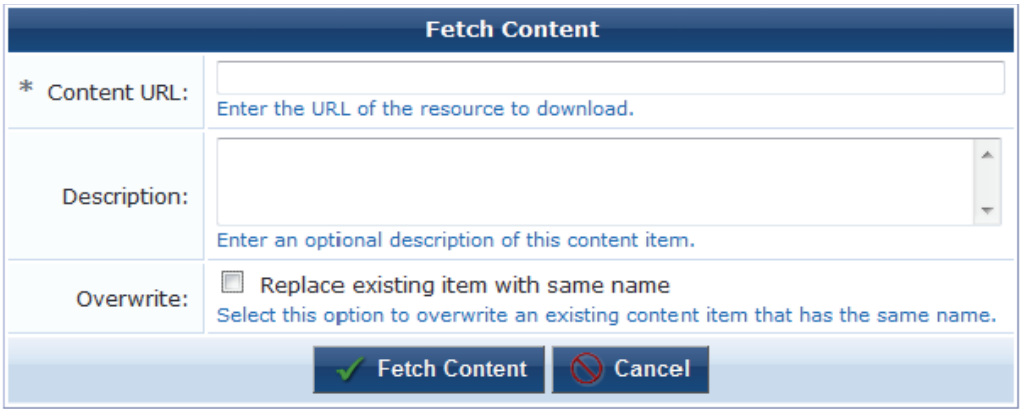
2. In the **File** row, click **Browse** to navigate to the file you wish to upload. The Maximum file size is 15 MB.

   You can upload single content files, multiple content asset files and folders, or a Web deployment archive. To upload multiple assets, first compress the files as a “tarball” or zip file, then browse to it in the File field. Allowed file formats are .tgz, .tar.gz, .tb2, .tar.bz2, or .zip. When you have uploaded the file, the Extract option lets you create the new directory, navigate into it, and view and extract the files. Directory structure is preserved when extracting.
3. (Optional) You may enter a description of the content assets in the **Description** text area.
4. To overwrite a previous file of the same name, mark the **Overwrite** check box.
5. Click **Upload Content** to upload the file. The file is displayed in the list view and will be placed in the **public** directory on the Web server. You can reference the file when creating custom HTML templates.

Downloading Content

To download a file from the Internet for use in ClearPass Guest:

1. Go to **Configuration > Content Manager**, then click the **Download New Content** tab. The **Fetch Content** form is displayed.

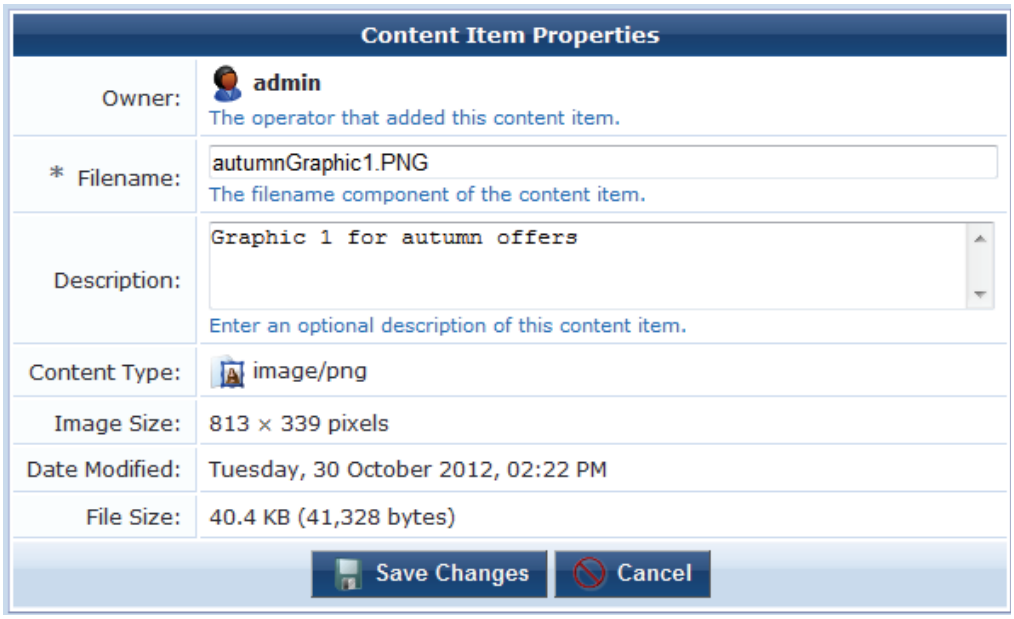


After you have completed the form, click the **Fetch Content** button to have the file downloaded. The file is placed in the **public** directory on the Web server. You are then able to reference this file when creating custom HTML templates.

Additional Content Actions

To work with your content items:

1. Go to **Configuration > Content Manager**, then click the item's row in the list. The row expands to include the **Properties**, **Delete**, **Rename**, **Download**, **View Content**, and **Quick View** options.
2. The **Properties** link allows you to view and edit the properties of the item. Editable properties include the content item's filename and description. Read-only properties include the content type, modification time, file size, and other content-specific properties such as the image's size.



3. You can use the **Delete** link to delete the content item. You will be asked to confirm the deletion.
4. You can use the using the **Rename** link to rename the content item.
5. To save a copy of the content item using your Web browser, click the **Download** link.
6. To open a new window to view the item, use the **View Content** link.
7. The **Quick View** link can be used to display certain types of content inline, such as images and text. The item is displayed below its row in the list. The Quick View link is not available for all content types.

Customizing Guest Manager

Guest Manager allows the entire guest account provisioning process to be customized. This is useful in many different situations, such as:

- **Self-registration** – Allow your guests to self-register and create their own temporary visitor accounts.
- **Visitor surveys** – Define custom fields to store data of interest to you, and collect this information from guests using customized forms.
- **Branded print receipts** – Add your own branding images and text to print receipts.
- **SMS and email receipts** – Include a short text message with your guest's username and password, or send HTML emails containing images.
- **Advanced customization** – ClearPass Guest is flexible and can be used to provide location sensitive content and advertising.

Default Settings for Account Creation

The Guest Manager plugin configuration holds the default settings for account creation.

To modify settings for the Guest Manager plugin configuration, go to **Configuration** and click the **Guest Manager Settings** command link, or, from the **Guest Manager** page, click the **Guest Manager Settings** command link.

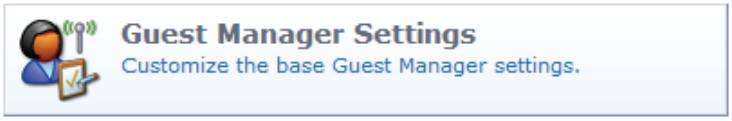
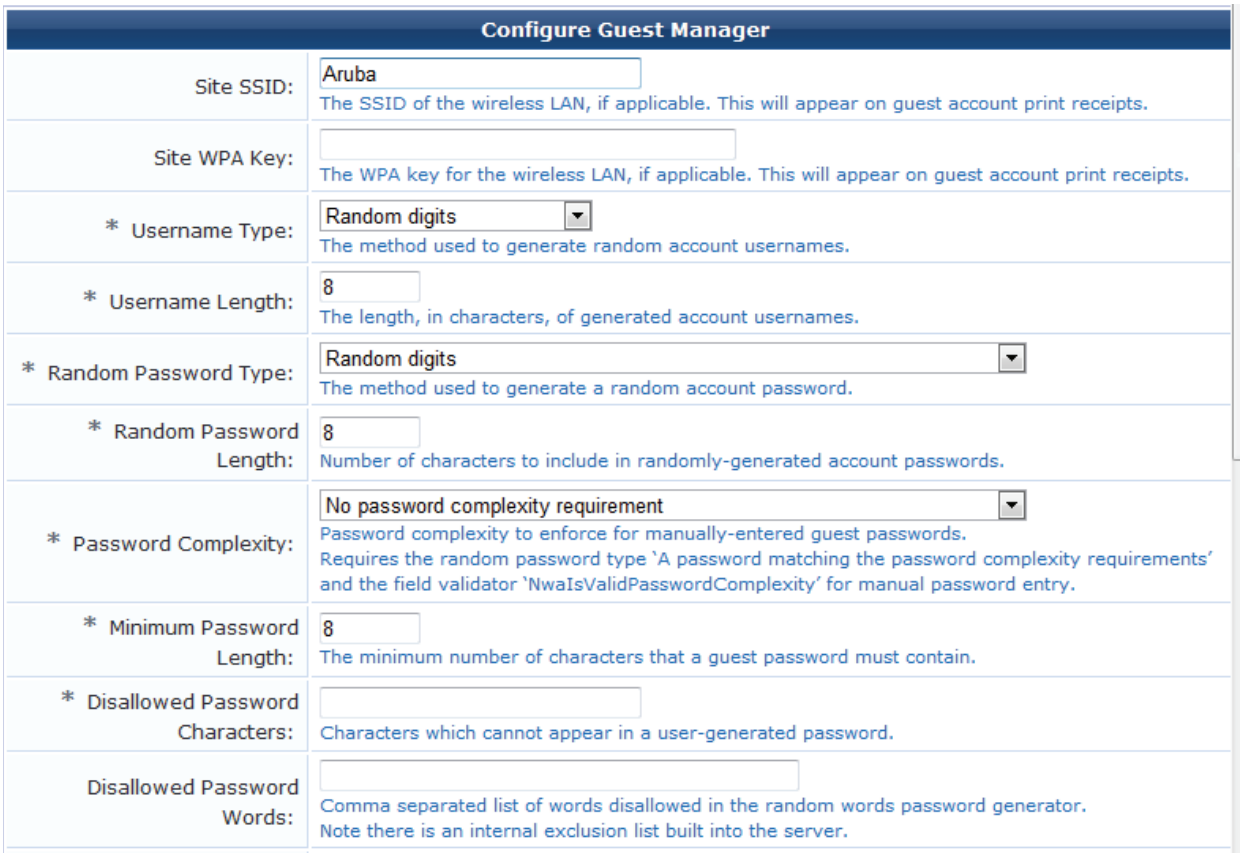
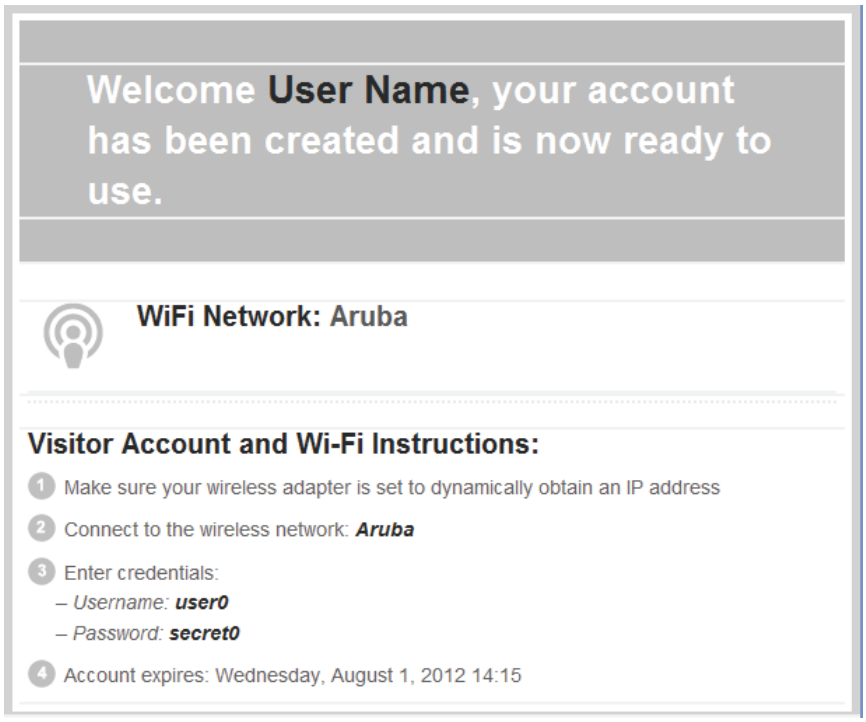


Figure 20 *Customize Guest Manager Page (upper section)*



- **Site SSID**—The Site SSID is the public name of the wireless local area network (WLAN). The default setting for this field is **Aruba**, and can be changed. The site SSID is displayed in the guest receipt as the **WiFi Network**, as shown below:

Figure 21 *Sample Guest Receipt Showing Aruba as the Default Site SSID*

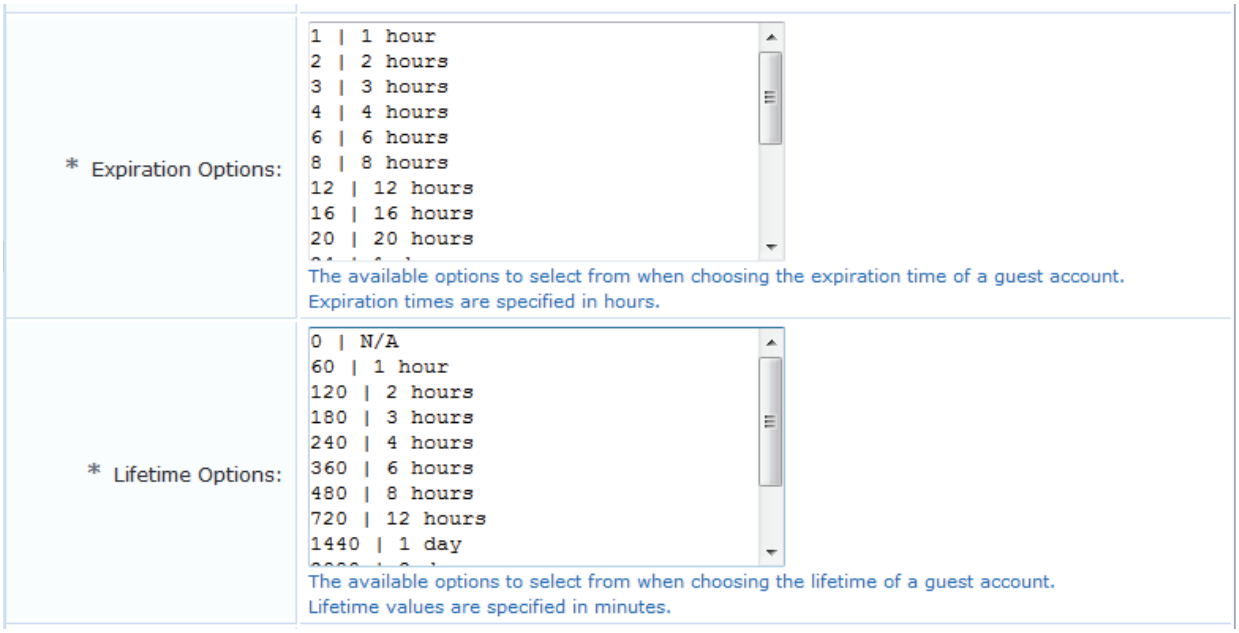


- **Site WPA Key**—The encryption key used to secure the wireless network. If a value is entered in this field, it will appear on guest print receipts.
- **Username Type**—The default method used to generate random account usernames (when creating groups of accounts). This may be overridden by using the **random_username_method** field.
  - **Username Length**—This field is displayed if the **Username Type** is set to "Random digits", "Random letters", "Random letters and digits" or "Sequential numbering". The default length of random account usernames (when creating groups of accounts). This may be overridden by using the **random_username_length** field.
  - **Username Format—**This field is displayed if the **Username Type** is set to "Format picture". It sets the format of the username to be created. See "Format Picture String Symbols" in the "Reference" chapter for a list of the special characters that may be used in the format string. This may be overridden by using the **random_username_picture** field.
- **Random Password Type—**The default method used to generate random account passwords (when creating groups of accounts). This may be overridden by using the **random_password_method** field.
  - **Random Password Length—**The default length of random account passwords (when creating groups of accounts). This may be overridden by using the **random_password_length** field
  - **Password Format—**This field is displayed if the **Password Type** field is set to "Format picture". It sets the format of the password to be created. See "Format Picture String Symbols" in the "Reference" chapter for a list of the special characters that may be used in the format string. This may be overridden by using the **random_password_picture** field.
- **Password Complexity—**The policy to enforce when guests change their account passwords using the guest self-service user interface. Different levels of password complexity can require guests to select passwords that contain different combinations of uppercase letters, lowercase letters, digits and symbols (`!#$%&()*+,-./:;<=>?@[\\]^_{|}~`,). The available options for this setting are:
  - No password complexity requirement

- At least one uppercase and one lowercase letter
- At least one digit
- At least one letter and one digit
- At least one of each: uppercase letter, lowercase letter, digit
- At least one symbol
- At least one of each: uppercase letter, lowercase letter, digit, and symbol

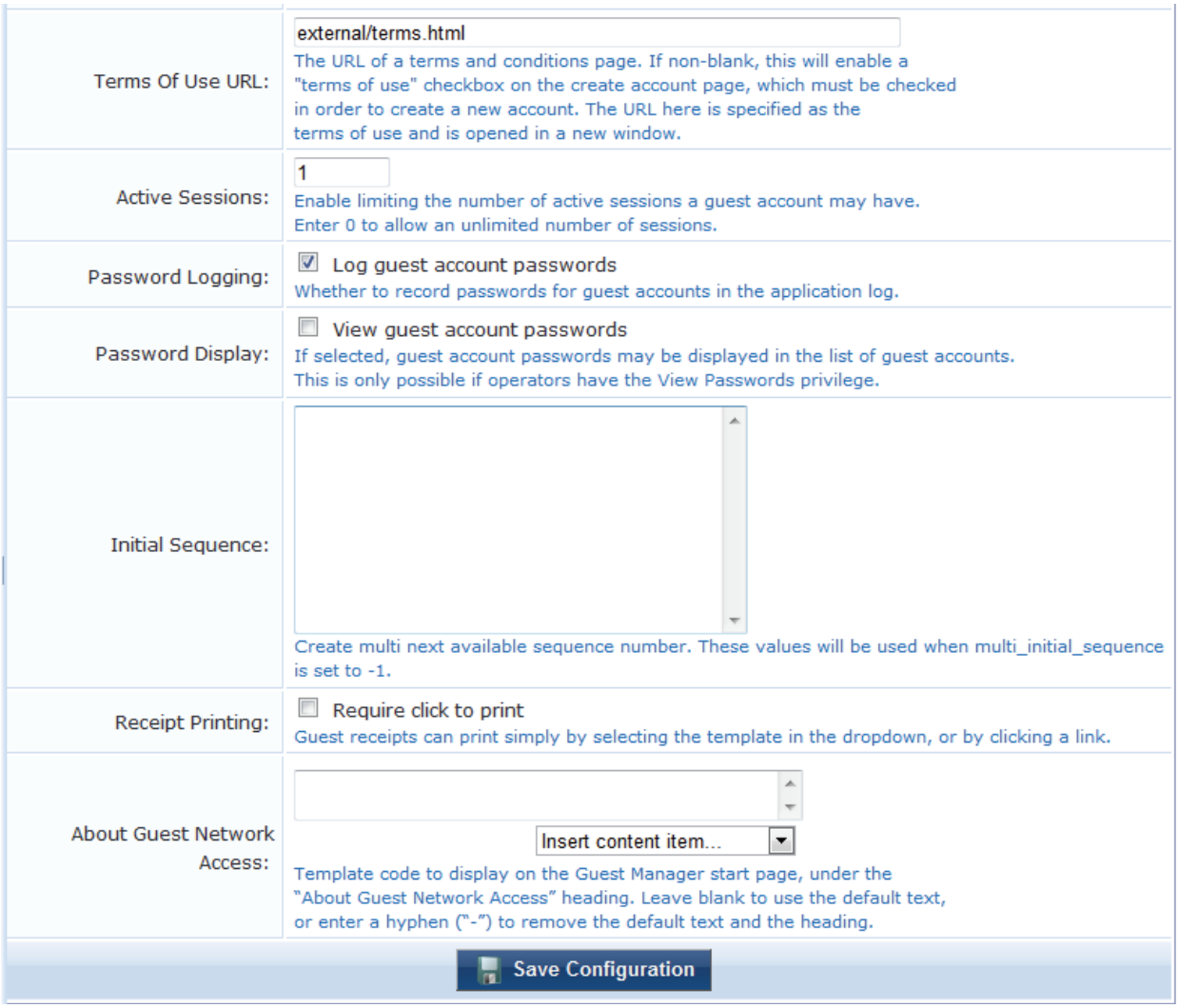
- **Minimum Password Length—**The minimum acceptable password length for guests changing their account passwords.
- **Disallowed Password Characters—**Special characters that should not be allowed in a guest password. Spaces are not allowed by default. You can specify special characters, numbers, and letters to exclude from passwords—for example, letters and numbers that can look similar, such as i, l, 1, 0, O, o, 5, S.
- **Disallowed Password Words—**Enter a comma- separated list of words that are disallowed and will not be created by the random words password generator.

Figure 22 *Customize Guest Manager Page, Continued (middle section)*



- **Expiration Options—**Default values for relative account expiration times. These options are displayed as the values of the “Expires After” field when creating a user account.
- **Lifetime Options**—Default values for account lifetimes. These options are displayed as the values of the “Account Lifetime” field when creating a user account.

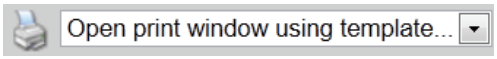
Figure 23 *Customize Guest Manager Page, Continued (lower section)*



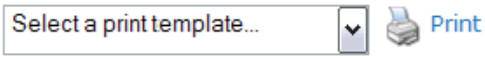
- **Terms of Use URL**—URL of a terms and conditions page provided to sponsors. You may upload an HTML file describing the terms and conditions of use using the Content Manager (See **"Content Manager"**). If this file is called **terms.html** then the Terms of Use URL should be **public/terms.html**.
- **Active Sessions**—Default maximum number of active sessions that should be allowed for a guest account. This may be overridden by using the **simultaneous_use** field when creating or editing a guest account.
- **Password Logging**—By default, the passwords for created guest accounts are logged in the application log and may be recovered from there. For increased security, you may prevent this password from being logged by unselecting this check box.
- **Password Display—**Select the "View guest account passwords" to enable the display of visitor account passwords in the user list. To reveal passwords, the **password** field must be added to the "guest_users" or "guest_edit" view, and the operator profile in use must also have the View Passwords privilege.
- **Initial Sequence—**This field contains the next available sequence number for each username prefix that has been used. Automatic sequence numbering is used when the value of the **multi_initial_sequence** field is set to -1. The username prefix is taken from the **multi_prefix** field when usernames are automatically generated using the "nwa_sequence" method. You can edit the values stored here to change the next sequence numbers that will be used. This is an automatically managed field; in most situations there is no need to edit it.

- **Receipt Printing—**Select the "Require click to print" option to change the behavior of the receipt page.

  When this option is not selected, the default behavior is to provide a drop-down list of print templates and to open a new window when one is selected:

  When "Require click to print" is selected, the receipt page provides a drop-down list of print templates and a **Print** link that must be clicked to display the account receipt:

- **About Guest Network Access**—Allows the text displayed to operators on the Guest Manager start page to be customized, or removed (if a single hyphen "-" is entered).

About Fields, Forms, and Views

- A field is a named item of information. It may be used to display information to a user as static text, or it may be an interactive field where a user can select an option or enter text.
- A form is a group of fields that is used to collect information from an operator.
- A view is a grouping of fields that is used to display information to an operator.

Business Logic for Account Creation

When guest accounts are created, there are certain rules that must be followed in order to create a valid account. These rules apply to all accounts, regardless of how the account was created.

The business logic rules that control all guest account creation are described below. To see the display name corresponding to a field name, go to **Configuration > Fields** and scroll to the field name. Display names are shown in the Column Title column.

Verification Properties

- **creator_accept_terms**: This field must be set to 1, indicating the creator has accepted the terms of use for creating the account. If the field is not present or is not set to 1, the visitor account is not created.
- **password2**: If this field is specified, its value must be equal to the "password" field, or else the visitor account is not created.
- **auto_update_account**: If this field is present and set to a non-zero value, account creation will not fail if the username already exists – any changes will be merged into the existing account using an update instead.

Basic User Properties

- **username**: This field is the name for the visitor account and may be provided directly. If this field is not specified, then use the email address from the **email** field, and if that is also not specified, then randomly generate a username (according to the value of the **random_username_method** and **random_username_length** fields).
- **modify_password**: This field controls password modification for the visitor account. It may be set to one of these values:
  - "reset" to randomly generate a new password according to the values of the **random_password_method** and **random_password_length** fields
  - "password" to use the password specified in the **password** field
  - "random_password" to use the password specified in the **random_password** field

- If blank or unset, the default password behavior is used, which is to use any available value from the **random_password** field and the **password** field, or assume that "reset" was specified otherwise.

- **password**: This field is the password for the visitor account and may be provided directly. If this field is not specified, then randomly generate a password (according to the values of the **random_password_method** and **random_password_length** fields).
- **role_id**: This field is the role to assign to the visitor account and may be specified directly. If this field is not specified, then determine the role ID from the **role_name** field. If no valid role ID is able to be determined, the visitor account is not created.
- **simultaneous_use**: This field determines the maximum number of concurrent sessions allowed for the visitor account. If this field is not specified, the default value from the GuestManager configuration is used.
- **random_username_method** – The method used to generate a random account username. If not specified, the default value from the GuestManager configuration is used.
- **random_username_length** – The length in characters of random account usernames. If not specified, the default value from the GuestManager configuration is used.
- **random_password_method** – The method used to generate a random account password. If not specified, the default value from the GuestManager configuration is used.
- **random_password_length** – The length in characters of random account passwords. If not specified, the default value from the GuestManager configuration is used.

Visitor Account Activation Properties

- **enabled**: This field determines if the account is enabled or disabled; if not specified, the default is 1 (account is enabled).
- **do_schedule**, **modify_schedule_time**, **schedule_after** and **schedule_time**: These fields are used to determine the time at which the visitor account will be activated.
  - If **modify_schedule_time** is "none", then the account is disabled and has no activation time set.
  - If **modify_schedule_time** is "now", then the account is enabled and has no activation time set.
  - If **modify_schedule_time** is a value that specifies a relative time change, for example "+1h", then the visitor account's activation time is modified accordingly.
  - If **modify_schedule_time** is a value that specifies an absolute time, for example "2010-12-31 17:00", then the visitor account's activation time is set to that value.
  - If **modify_schedule_time** is "schedule_after" or "schedule_time", then the activation time is determined according to the **schedule_after** or **schedule_time** fields as explained below.
  - If **schedule_after** is set and not zero, then add that time in hours to the current time and use it as the activation time (setting **do_schedule** to 1); **enabled** will be set to zero.
  - Otherwise, if **schedule_after** is zero, negative or unset, and **schedule_time** has been specified, use that activation time (set **do_schedule** to 1 and **enabled** to 0). If the **schedule_time** specified is in the past, set **do_schedule** to 0 and **enabled** to 1.
  - Otherwise, if **schedule_time** if not specified, then the visitor account has no activation time and **do_schedule** will default to zero.

Visitor Account Expiration Properties

- **do_expire**, **modify_expire_time**, **expire_after** and **expire_time**: These fields are used to determine the time at which the visitor account will expire.
  - If **modify_expire_time** is "none", then the account has no expiration time set.
  - If **modify_expire_time** is "now", then the account is disabled and has no expiration time set.

- If **modify_expire_time** is a value that specifies a relative time change, for example “+1h”, then the visitor account’s expiration time is modified accordingly.
- If **modify_expire_time** is a value that specifies an absolute time, for example “2010-12-31 17:00”, then the visitor account’s expiration time is set to that value.
- If **modify_expire_time** is “expire_after” or “expire_time”, then the expiration time is determined according to the **expire_after** or **expire_time** fields as explained below.
- If **expire_after** is set and not zero and the account will be activated immediately, then add the value in hours to the current time to determine the expiration time.
- If **expire_after** is set and not zero and account activation is set for a future time (schedule_time) instead of the current time, then the expiration time is calculated relative to the activation time instead of the current time.
- Otherwise, if **expire_after** is zero, negative or unset, and **expire_time** has been specified, use that expiration time. If the **expire_time** specified is in the past, set **do_expire** to 0 and ignore the specified expiration time.
- Otherwise, if **expire_time** is not specified, then the **expire_time** is not set and **do_expire** will always be set to zero.
- If the **do_expire** field is not included in the form, the default expiration action is 4, Logout and Delete. This can be configured on the Customize Guest Manager page.

- **expire_postlogin**: This field determines the amount of time after the initial login for which the visitor account will remain valid. If this field is not specified, the default value is 0 (account lifetime not set).
- **expire_usage**: This field determines the total amount of login time permitted for the visitor account. If this field is not specified, the default value is 0 (account usage is unlimited).

Other Properties

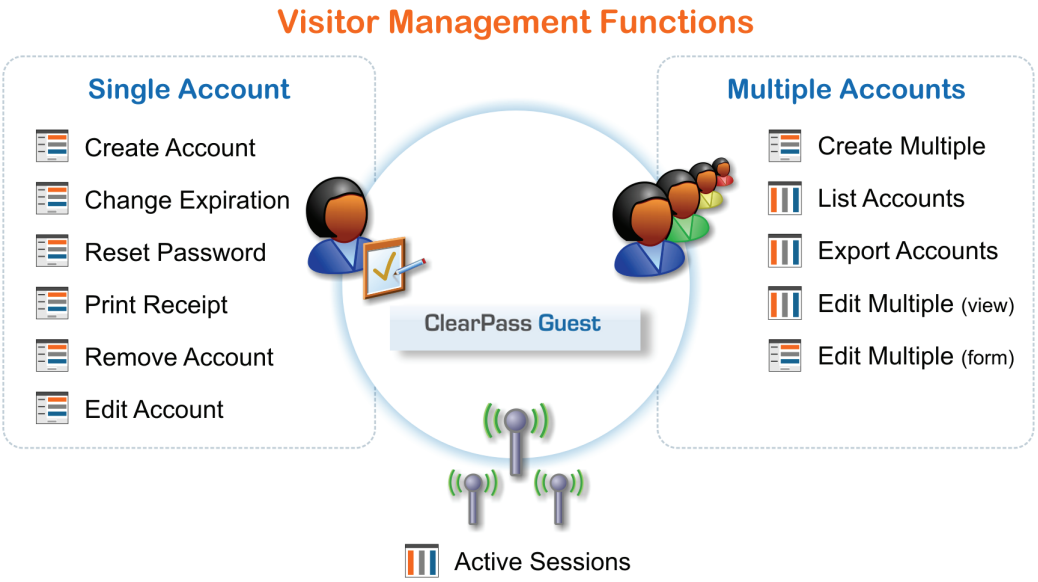
- All other properties specified at creation time are stored with the visitor account (for example, **email**, **visitor_name**, **visitor_company**, **visitor_phone**, **sponsor_name** as well as any custom fields that have been defined)

Standard Fields

See “Field, Form, and View Reference” in the “Reference” chapter for a listing of the standard fields shipped with ClearPass Guest.

Standard Forms and Views

The figure below shows the standard forms and views in the application.



The table below lists all the forms and views used for visitor management.

Table 19 *Visitor Management Forms and Views*

Name	Type	Visitor Management Function	Editable?
change_expiration	Form	Change Expiration	Yes
create_multi	Form	Create Multiple	Yes
create_user	Form	Create Account	Yes
guest_edit	Form	Edit Account	Yes
guest_export	View	Export Accounts	Yes
guest_multi	View	Edit Multiple Accounts	Yes
guest_multi_form	Form	Edit Multiple Accounts	Yes
guest_receipt	Form	Print Receipt	No
guest_register	Form	Guest Self-Registration	Yes
guest_register_receipt	Form	Guest Self-Registration Receipt	Yes
guest_sessions	View	Active Sessions	Yes
guest_users	View	List Accounts	Yes
remove_account	Form	Remove Account	No
reset_password	Form	Reset Password	No

These forms are accessed directly:

- **create_multi** form – multiple account creation
- **create_user** form – sponsored account creation
- **guest_register** form – guest self-registration form

These forms are accessed through the action row of the **guest_users** view:

- **change_expiration** form – change expiration time for a single account
- **guest_multi_form** form – editing multiple accounts
- **guest_edit** form – editing single account
- **reset_password** form – reset password for a single account

These forms are the standard self-registration forms:

- **guest_register** form – self-registration form
- **guest_register_receipt** form – self-registration receipt

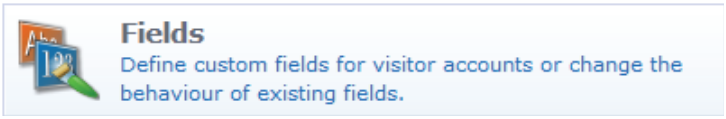
These standard views are defined in Guest Manager:

- **guest_export** view – view used when exporting guest account information
- **guest_multi** view – displays a list of guest accounts optimized for working with multiple accounts
- **guest_sessions** view – displays a list of current or historical sessions (See "Active Sessions Management" in the "Guest Manager"chapter.)
- **guest_users** view – displays a list of guest accounts optimized for working with individual accounts

Customizing Fields

Custom fields are fields that you define yourself to cater for areas of interest to your organization. You are able to define custom fields for your guest accounts as well as edit the existing fields.

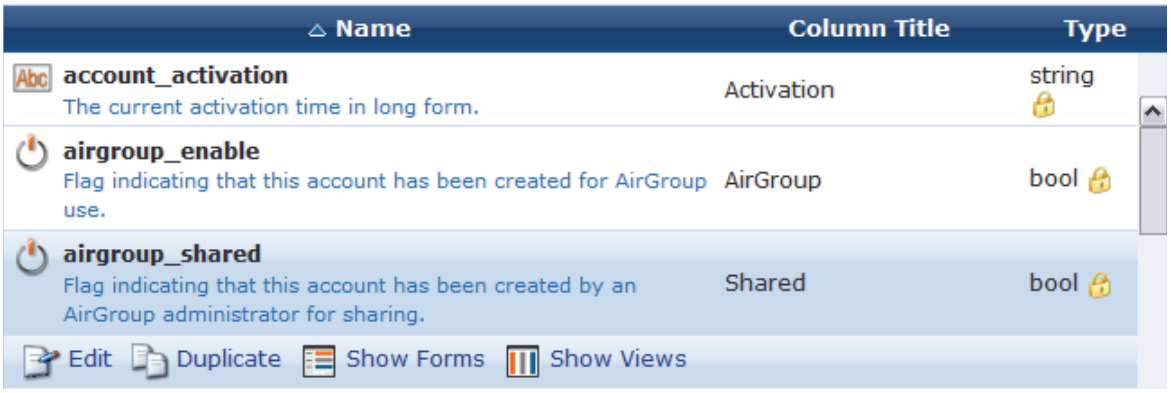
In addition you can delete and duplicate fields. For your convenience you are also able to list any forms or views that use a particular field.



NOTE

Fields that have a lock symbol cannot be deleted.

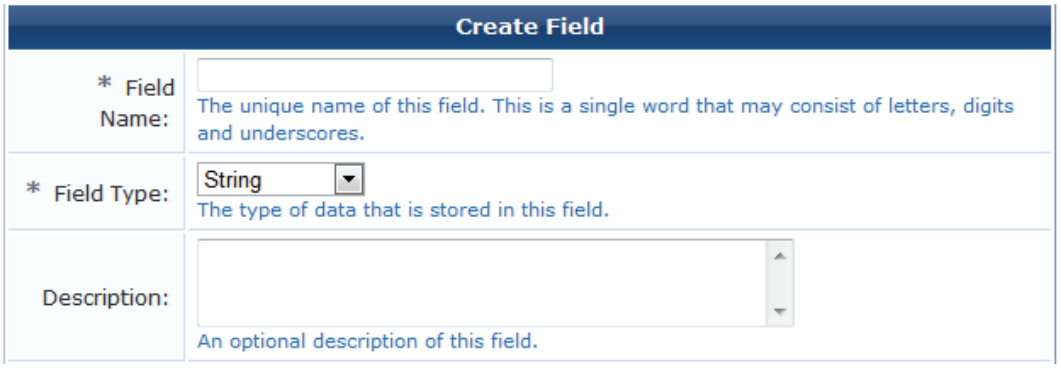
A complete list of fields is displayed when you click the **Fields** command link on the **Customize Guest Manager** page.



To display only the fields that you have been created, click the **Custom Fields Only** link in the bottom row of the list view. To return to displaying all fields, click the **All Fields** link.

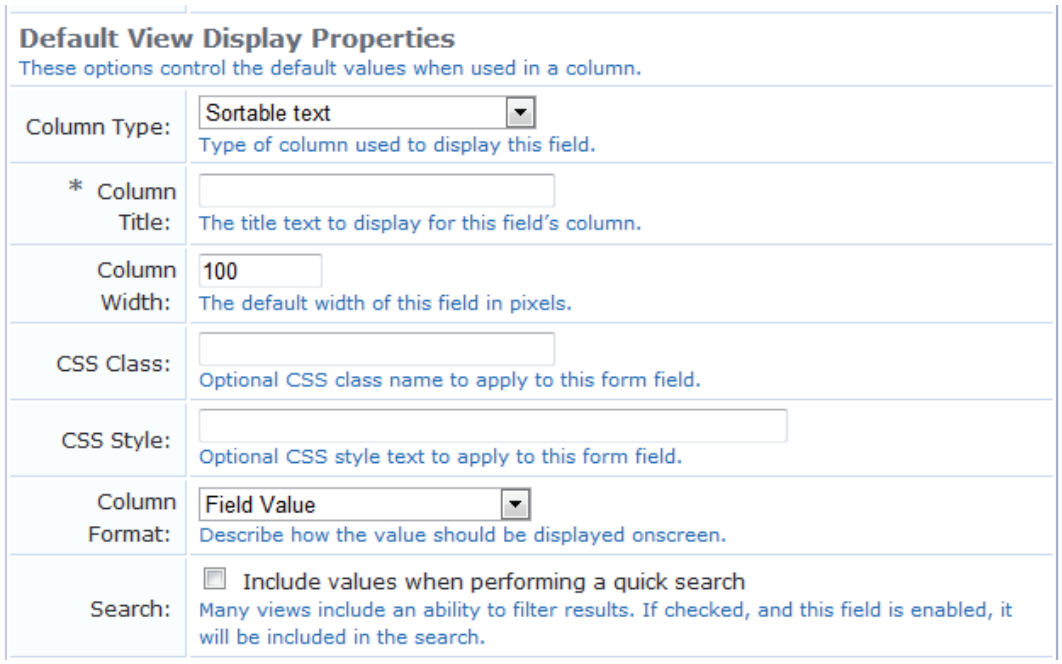
Creating a Custom Field

To create a custom field, click the **Create** tab at the top of the window or the **Create a new field** link at the bottom of the window. The Create Field form is displayed.

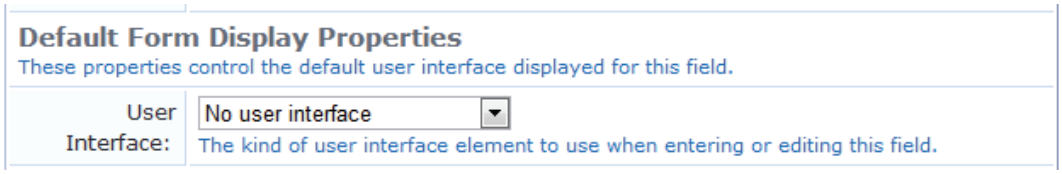


The Field Name is not permitted to have spaces but you can use underscores. Enter a description in the Description field. You can enter multiple-line descriptions which result in separate lines displayed on the form.

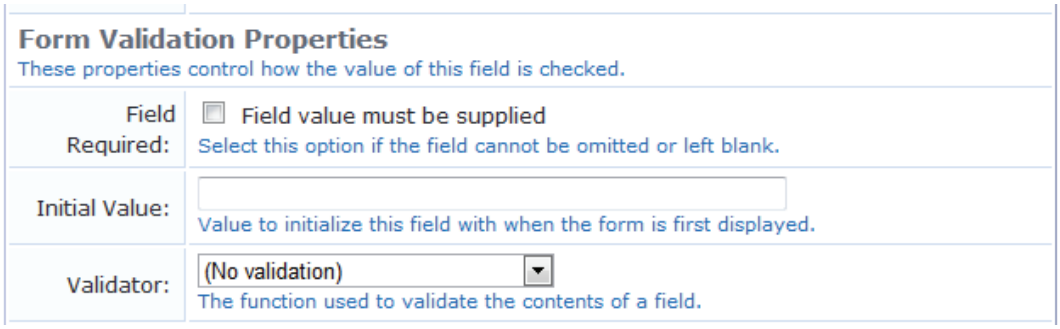
The Field Type can be one of String, Integer, Boolean or No data type. The No data type field would be used as a label, or a submit button.



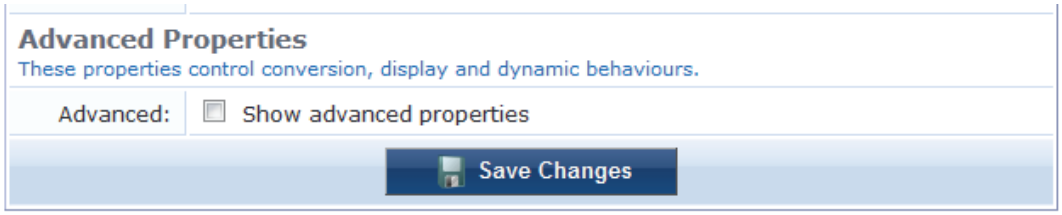
You can specify the default properties to use when adding this field to a view. See **"View Field Editor"** for a description of the view display fields, including the Column Type and Column Format fields.



You can specify the default properties to use when adding the field to a form. See **"View Field Editor"** for a list of the available user interface types.



You can specify the default validation rules that should be applied to this field when it is added to a form. See **"Form Validation Properties"** in this chapter for further information about form validation properties.



Select the **Show advanced properties** check box to reveal additional properties related to conversion, display and dynamic form behavior. See **"View Field Editor"** in this chapter for more information about advanced properties.

Click the **Save Changes** button to complete the creation of a new field. The new field is added at the top of the field list. To change the position of the new field, you can re-sort the list or you can reload the page.

Duplicating a Field

To duplicate a field, click the field to be duplicated, then click the **Duplicate** link. The field is copied and a number appended to the end of the field name—for example, if you were to duplicate the **card_code** field, the duplicated field would be **card_code_1**. To rename the field, click **Edit**.

Editing a Field

You are able to alter the properties of the field by making changes to the Field Name, Field Type or Description when you click the **Edit** link. This link is available when you click a field in the list view.

Click the **Save Changes** button to have the changes made permanent.

Deleting a Field

Fields that do not have a lock symbol can be deleted by clicking on the **Delete** link. You will be asked to confirm the deletion. If you want the deletion to take place you are informed when the deletion has been completed. A field that is currently in use on a form or view may not be deleted.

Displaying Forms that Use a Field

Click the **Show Forms** link to see a list of forms that use the selected field.

The list displays the forms that use the selected field. It also allows you to edit the form's fields by clicking on the **Edit Fields** link. Clicking on the **Use** link opens the form using that field.

If the field is used on multiple forms, you are able to select which form you would like to view.

Displaying Views that Use a Field

You are able to click the **Show Views** link to see a list of views that use the selected field.

The list displays the views that use the selected field. It also allows you to edit the view's fields by clicking on the **Edit Fields** link. Clicking on the **Use** link displays the view.

If the field is used on multiple views, you are able to select which view you would like to see.

Customizing AirGroup Registration Forms

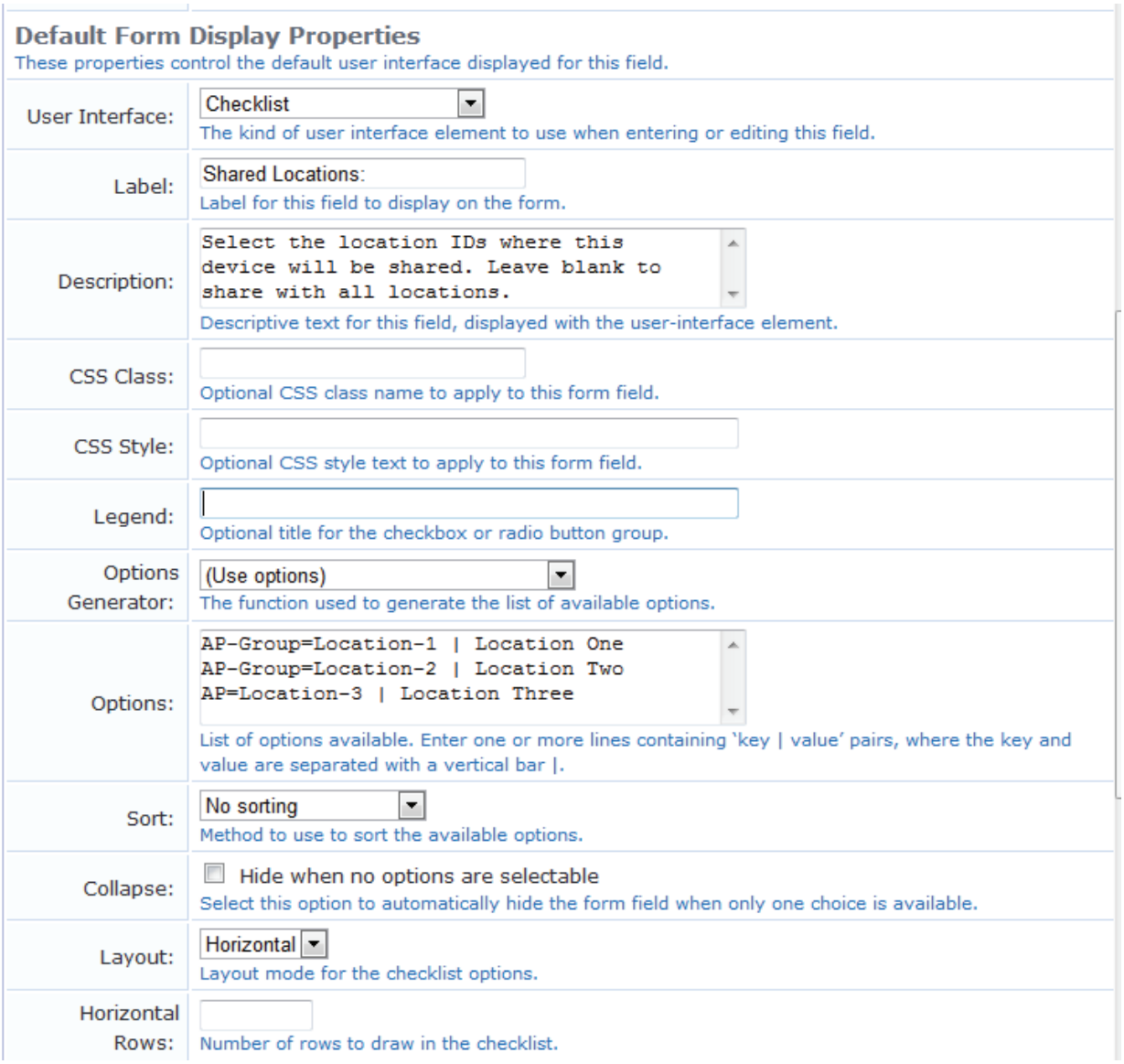
AirGroup allows users to register their personal mobile devices on the local network and define a group of friends or associates who are allowed to share them. If AirGroup Services is enabled, AirGroup administrators can provision their organization's shared devices and manage access, and AirGroup operators can register and provision a limited number of their own personal devices for sharing. For complete AirGroup deployment information, refer to the AirGroup Deployment Guide and the ClearPass Policy Manager documentation.

On the device registration forms for AirGroup administrators and operators, the default **Shared Locations** and **Shared Roles** fields are text boxes where the user enters the information. These fields can be configured as selection options populated with existing locations or roles.

Configuring the Shared Locations and Shared Role Fields

To configure a predefined list of shared locations or shared roles:

1. Go to **Configuration > Fields** and click the **airgroup_shared_location** or **airgroup_shared_role row**. The form expands to include the **Edit**, **Duplicate**, **Show Forms**, and **Show Views** links.
2. Click the **Edit** link. The Define Custom Field form opens. Scroll to the **Default Form Display Properties** section.



3. In the **User Interface** drop-down list, select **Checklist**.
4. In the **Description** text box, delete the existing text, then enter **Select the location IDs where this device will be shared. Leave blank to share with all locations**.
5. Delete any text from the **CSS Class** and the **CSS Style** fields.
6. In the **Options Generator** drop-down list, select **(Use options)**.
7. In the **Options** text box, enter a list of values to use as the checklist options that presented to the user.

   The values you enter in the Options text box control both the values stored in the shared_location field in the database as well as the text displayed to the user in the checklist. Use the following format:

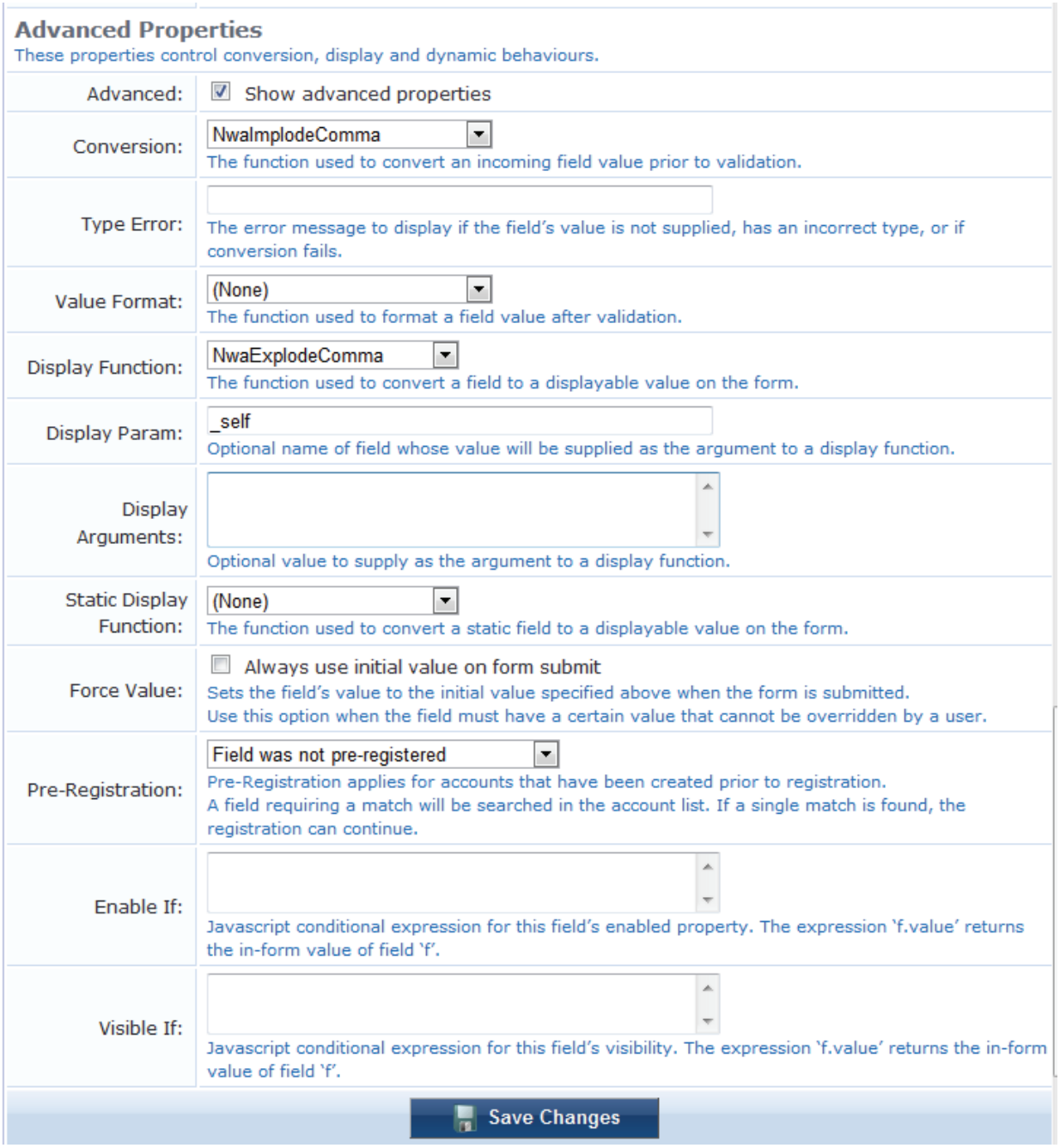
   ```
   tag1=value1 | Option 1
   tag2=value2 | Option 2
   ```

   ...where the tag=value pair **tag1=value1** represents the value stored in the shared_location field in the database, the pipe character (|) is a separator, and **Option 1** represents the text displayed in the checklist.
8. (Optional) To sort the locations by key or value, choose an option from the **Sort** drop-down list.
9. (Optional) To control the layout of the checklist on the form, first use the **Layout** drop-down list to select either **Vertical** or **Horizontal**. The name of the next field changes to correspond to your choice

in this field. Enter the appropriate number in the **Vertical Rows** or **Horizontal Rows** field. If the Layout field is left blank, the default layout of a single list of checklist options is displayed.

To ensure the values are stored correctly as a comma-separated list:

1. Scroll to the **Advanced Properties** section of the form and mark the check box in the **Advanced** row. The form expands to include the advanced options.



2. In the **Conversion** drop-down list, select **NwaImplodeComma**. The form expands to include the Type Error row.
3. In the **Display Function** drop-down list, select **NwaExplodeComma**. The form expands to include the Display Param and Display Arguments rows.
4. In the **Display Param** text field, enter the value **_self**. Be sure to include the leading underscore character.
5. Click **Save Changes**.

Example:

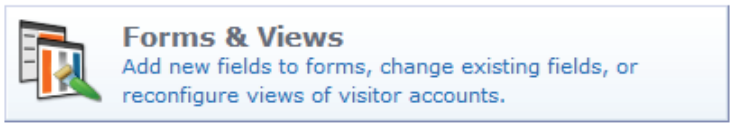
If the layout is set to vertical and the following options are specified:

```
AP-Group=Location-1 | Location One
AP-Group=Location-2 | Location Two
AP-Location-3 | Location Three
```

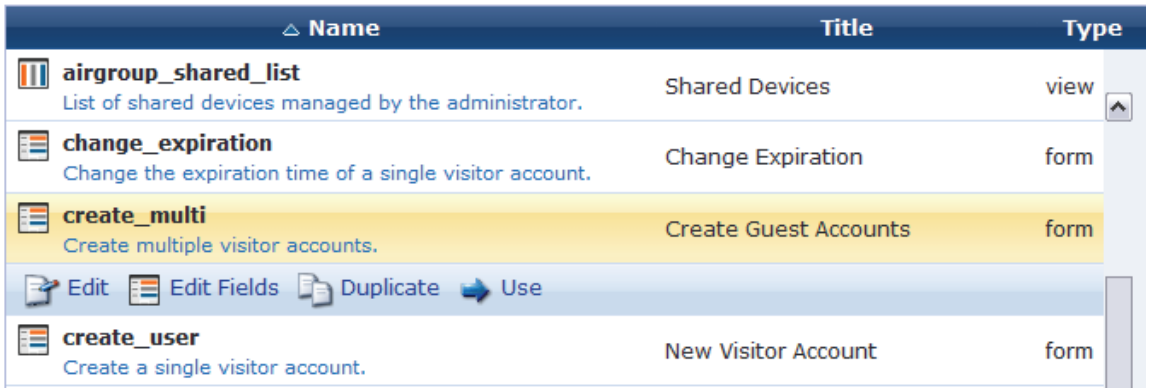
The user interface appears as follows:

Customizing Forms and Views

You are able to view a list of forms and views. From this list view, you can change the layout of forms or views, add new fields to a form or view, or alter the behavior of an existing field.



To view or customize forms and views, go to **Configuration > Forms & Views**. The Customize Forms and Views page opens.

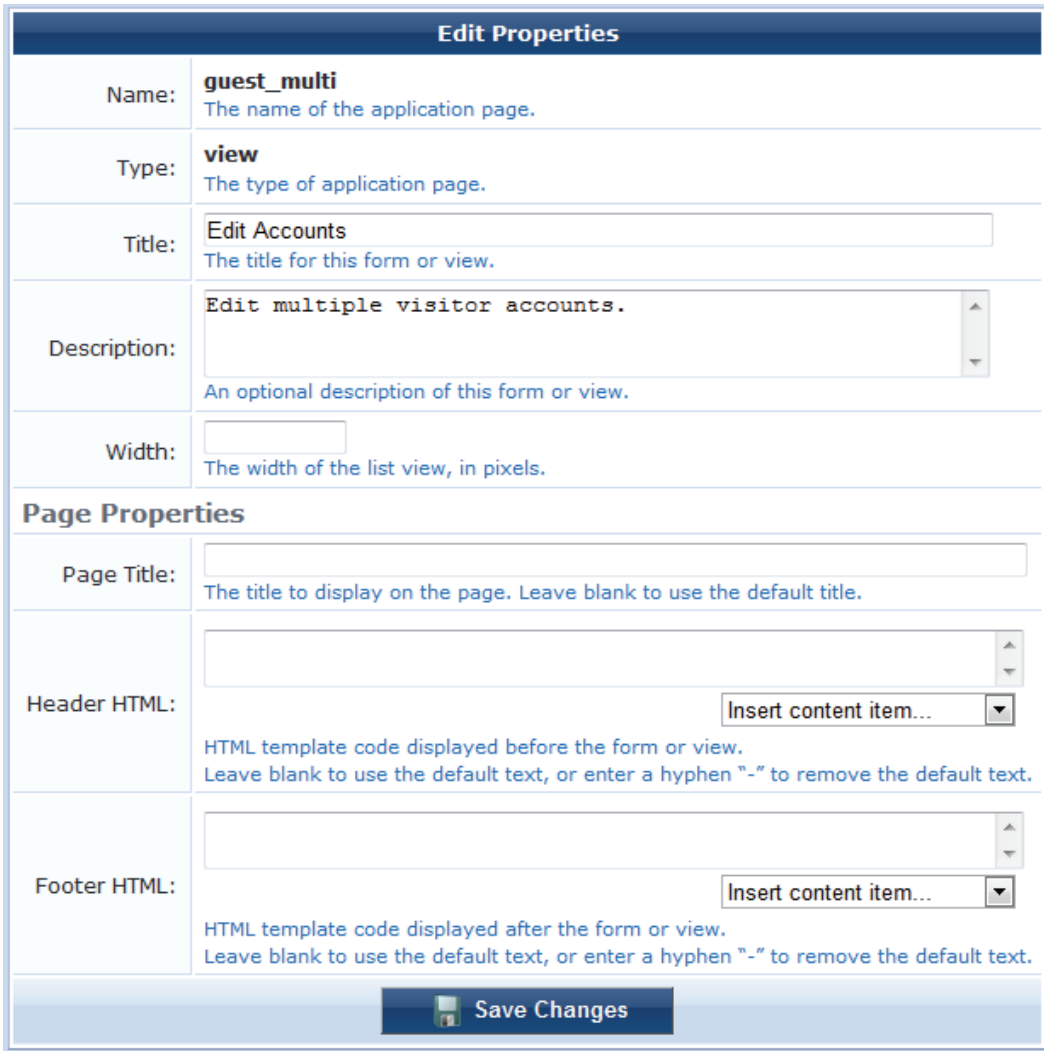


You can open a form or view directly from the Forms and Views page. To open form or view to use it, go to **Configuration > Forms & Views**, click the form's or view's row in the list, then click its **Use** link. The form or view opens in a separate browser tab, and the Forms and Views tab stays open so you can work in both.

An asterisk (*) shown next to a form or view indicates that the form or view has been modified from the defaults. You can click the **Reset to Defaults** link to remove your modifications and restore the original form. Resetting a form or view is a destructive operation and cannot be undone. You will be prompted to confirm the form or view reset before it proceeds.

Editing Forms and Views

You can change the general properties of a form or view such as its title and description. To edit the form or view, go to **Configuration > Forms & Views**, click the form's or view's row in the list, then click its **Edit** link. The row expands to include the Edit Properties form.



The **Width** field is only displayed for views. It specifies the total width of the list view in pixels. If blank, a default value is used.

You can customize the page title, header HTML, and footer HTML for many forms and views (for example, Create Guest Account, Edit Guest Accounts, and others). When these options are available, the **Page Properties** area is included on the Edit Properties form.

Duplicating Forms and Views

You can make a copy of a form or view to use as a template in order to provide different forms and views to different operator profiles. See "Role-Based Access Control for Multiple Operator Profiles" in the "Operator Logins" chapter for a description. This enables you to provide different views of the underlying visitor accounts in the database depending on the operator's profile.

To make a copy of the form or view, go to **Configuration > Forms & Views**, click the form's or view's row in the list, then click its **Duplicate** link. The copy is added to the Forms and Views list.

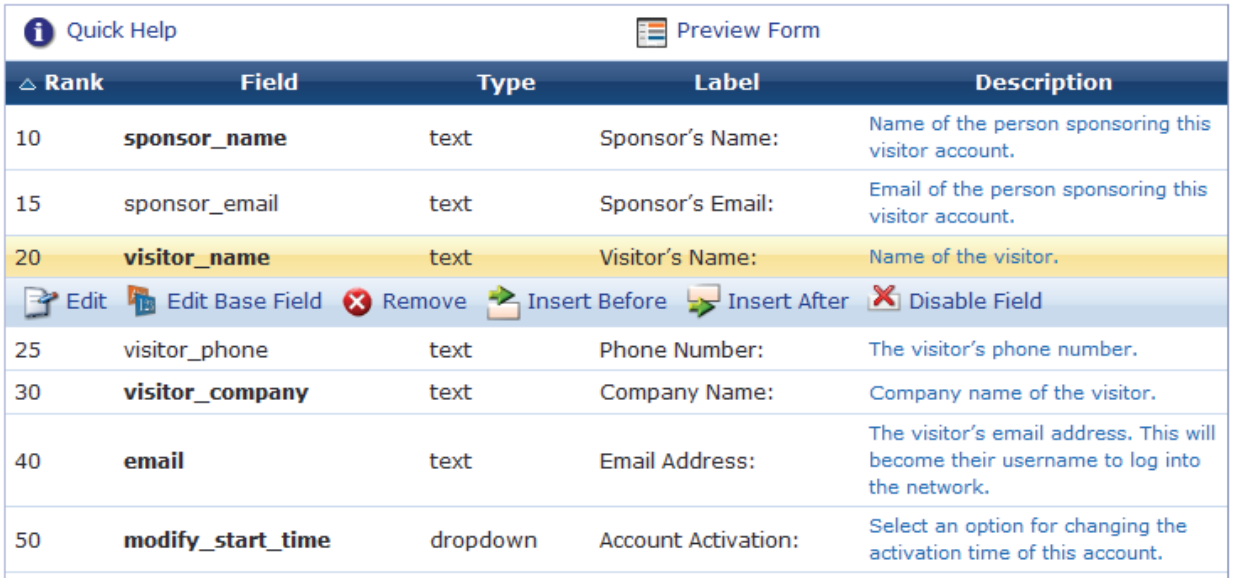
The name of the duplicated form or view is the same as the original with a number appended. This name cannot be changed. Use the **Title** and **Description** properties of the duplicated item to describe the intended purpose for the form or view.

Click the **Show Usage** link for a duplicated form or view to see the operator profiles that are referencing it.

Click the **Delete** link for a duplicated form or view to remove the copy. A duplicated item cannot be removed if it is referenced by an operator login account or an operator profile.

Editing Forms

To add a new field to a form, reorder the fields, or make changes to an existing field, go to **Configuration > Forms & Views**, click the form's row in the Customize Forms & Views list, and then click the **Edit Fields** link. The **Customize Form Fields** view opens.



Form fields have a **Rank** number, which specifies the relative ordering of the fields when displaying the form. The Customize Form Fields editor always shows the fields in order by rank.

The **Type** of each form field is displayed. This controls what kind of user interface element is used to interact with the user. The **Label** and **Description** displayed on the form are also shown in the list view.

To work with a form field, click its row in the list. The row expands to include the **Edit**, **Edit Base Field**, **Remove**, **Insert Before**, **Insert After**, and **Disable Field** options.

To make changes to an existing field, click its **Edit** link. The Form Field Editor opens. Any changes made to the field using this editor will apply only to this field on this form.

To make changes to an existing field's definition, click its **Edit Base Field** link. Any changes made to the field using this editor will apply to all forms that are using this field (except where the form field has already been modified to be different from the underlying field definition).

The **Insert Before** and **Insert After** links can be used to add a new field to the form. Clicking one of these links will open a blank form field editor and automatically set the rank number of the new field.

Use the **Preview Form** tab at the top of the list view to see what the form looks like. This preview form can be submitted to test the field validation rules you have defined. If all fields are able to be validated, the form submit is successful and a summary of the values submitted is displayed. This allows you to verify any data conversion and formatting rules you have set up.

Form Field Editor

The form field editor is used to control both the data gathering aspects and user interface characteristics of a field.

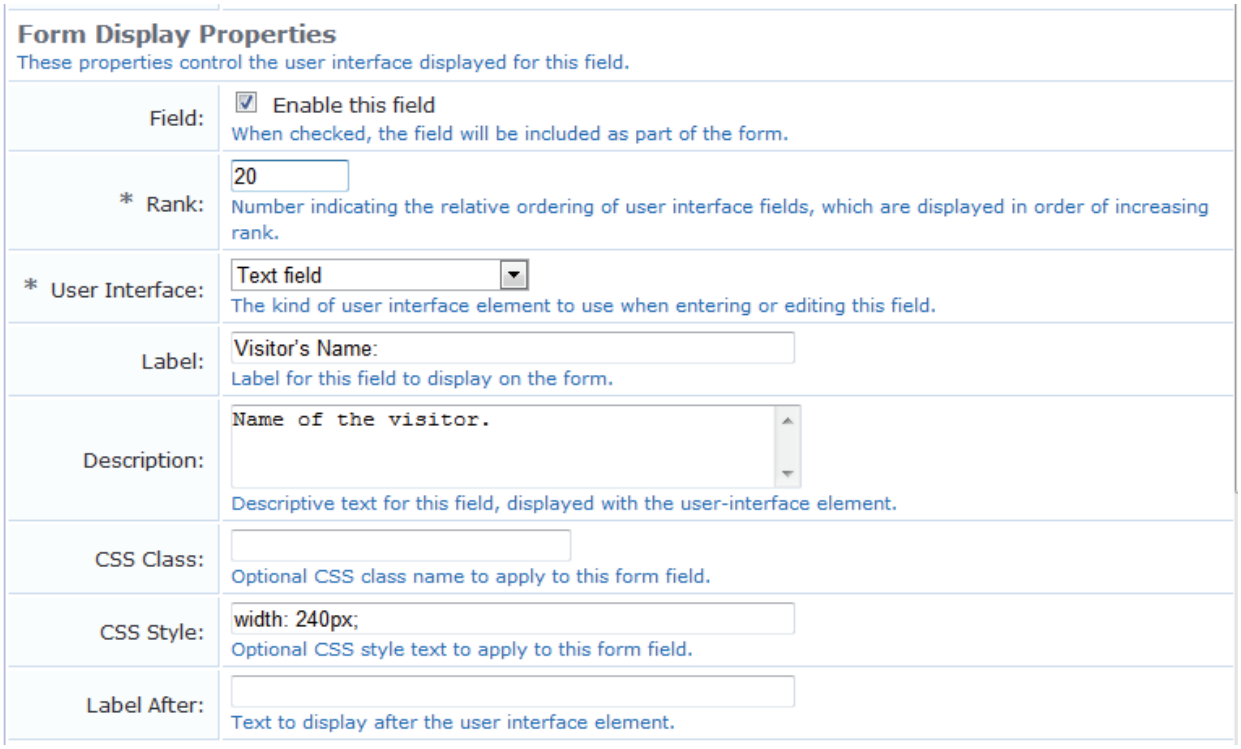
Each field can only appear once on a form. The **Field Name** selects which underlying field is being represented on the form.

The remainder of the form field editor is split into three sections:

- Form Display Properties
- Form Validation Properties
- Advanced Properties

See "Form Display Properties" for detailed descriptions of these form sections.

Form Display Properties

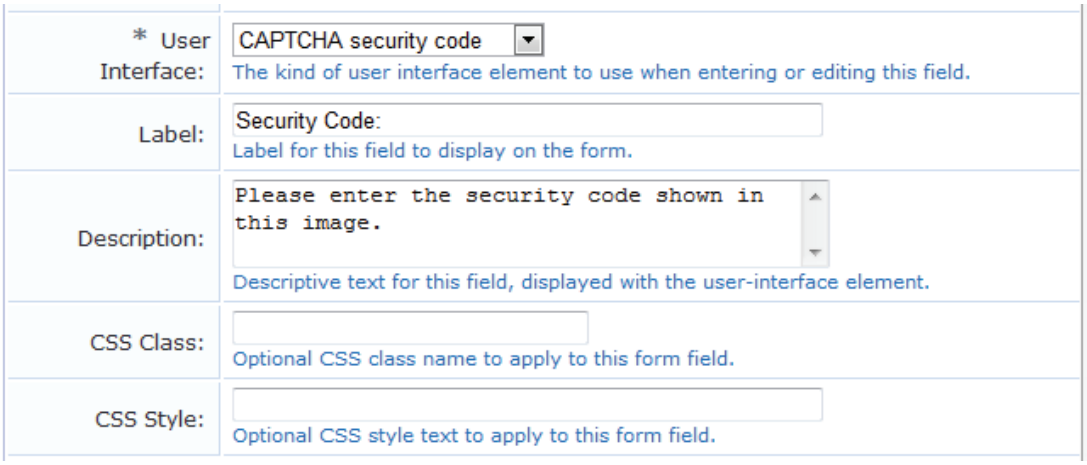


The form display properties control the user interface that this field will have. Different options are available in this section, depending on the selection you make in the User Interface drop-down list.

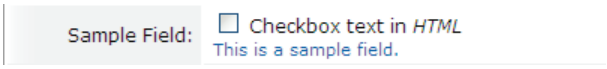
The available user interface elements are listed below, together with an example of each.

- **(Use default)** – The default user interface type defined for the field will be used.
- **No user interface** – The field does not have a user interface specified. Using this value will cause a diagnostic message to be displayed ("Form element is missing the 'ui' element") when using the form.
- **CAPTCHA security code** – A distorted image of several characters will be displayed to the user, as shown below:

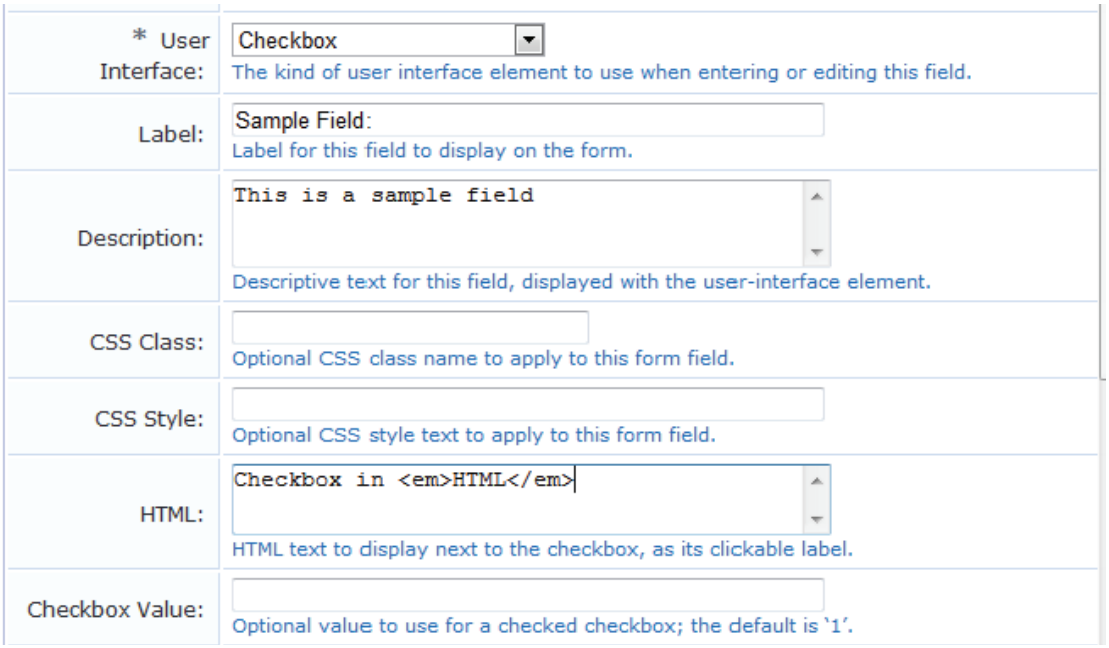
The image may be regenerated, or played as an audio sample for visually impaired users. When using the recommended validator for this field (NwaCaptchaIsValid), the security code must be matched or the form submit will fail with an error.



- **Check box** – A check box is displayed for the field, as shown below:



The check box label can be specified using HTML. If the check box is selected, the field is submitted with its value set to the check box value (default and recommended value 1). If the check box is not selected, the field is not submitted with the form.

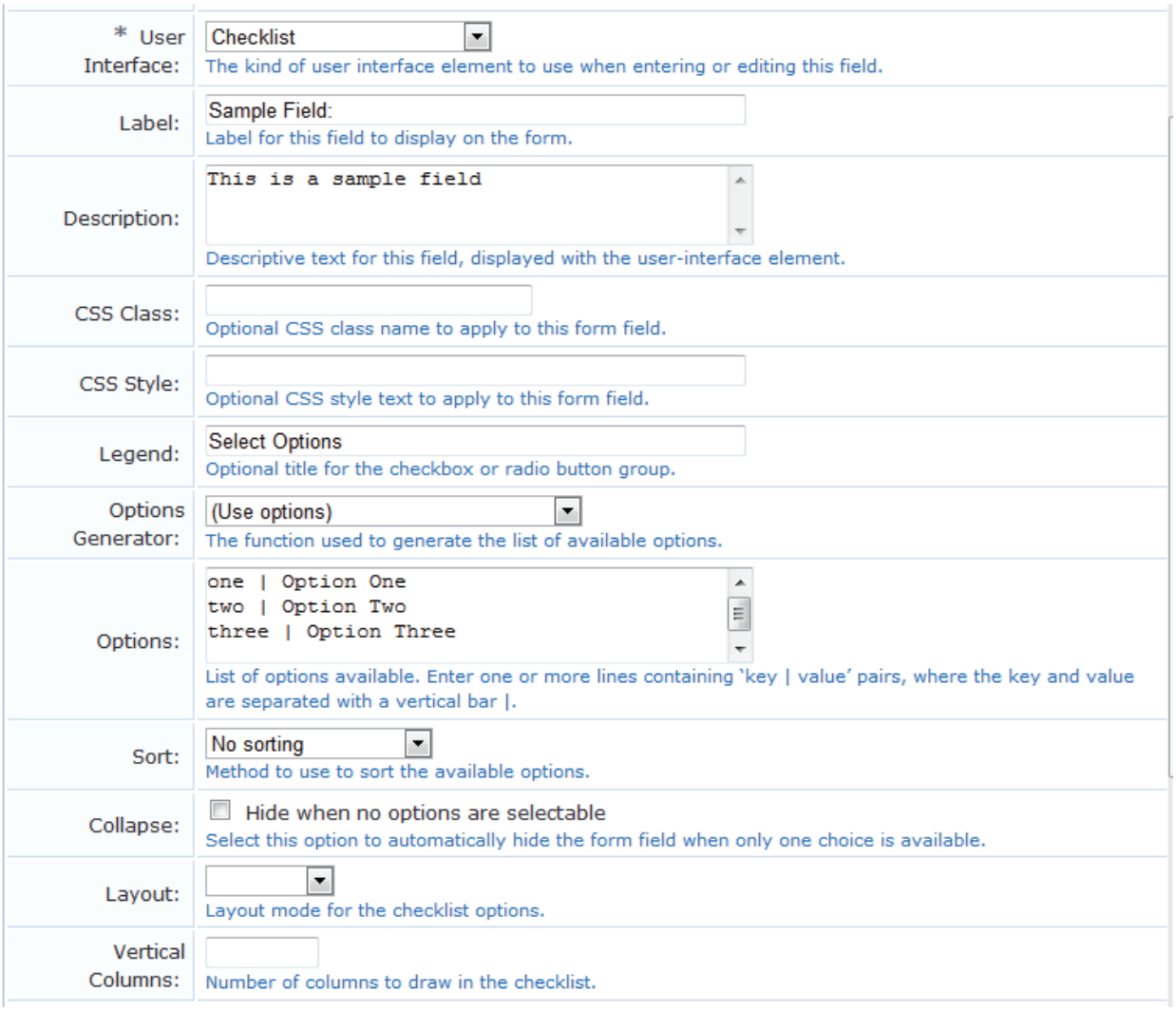


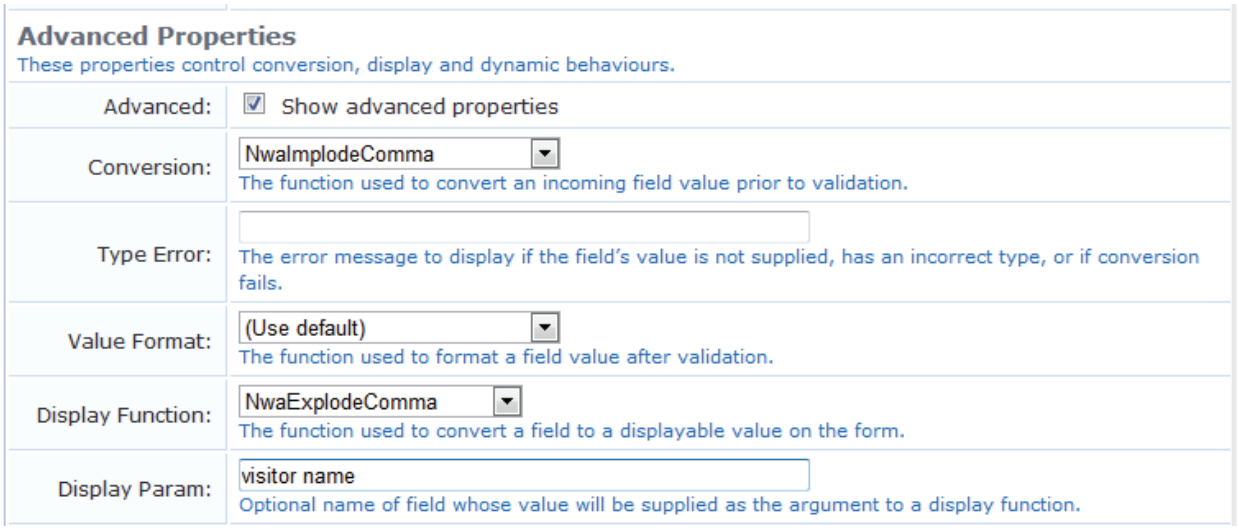
- **Checklist** – A list of check boxes is displayed, as shown below:

The text displayed for each check box is the value from the options list. Zero or more check boxes may be selected. This user interface type submits an array of values containing the option key values of each selected check box. Because an array value may not be stored directly in a custom field, you should use the conversion and value formatting facilities to convert the array value to and from a string when using this user interface type.

To store a comma-separated list of the selected values, enable the **Advanced** options, select "NwaImplodeComma" for **Conversion**, select "NwaExplodeComma" for **Display Function** and enter the field's name for **Display Param**.

The "Vertical" and "Horizontal" layout styles control whether the check boxes are organized in top-to-bottom or left-to-right order. The default is "Vertical" if not specified. When using these options, you may also specify the desired number of columns or rows to adjust the layout appropriately.





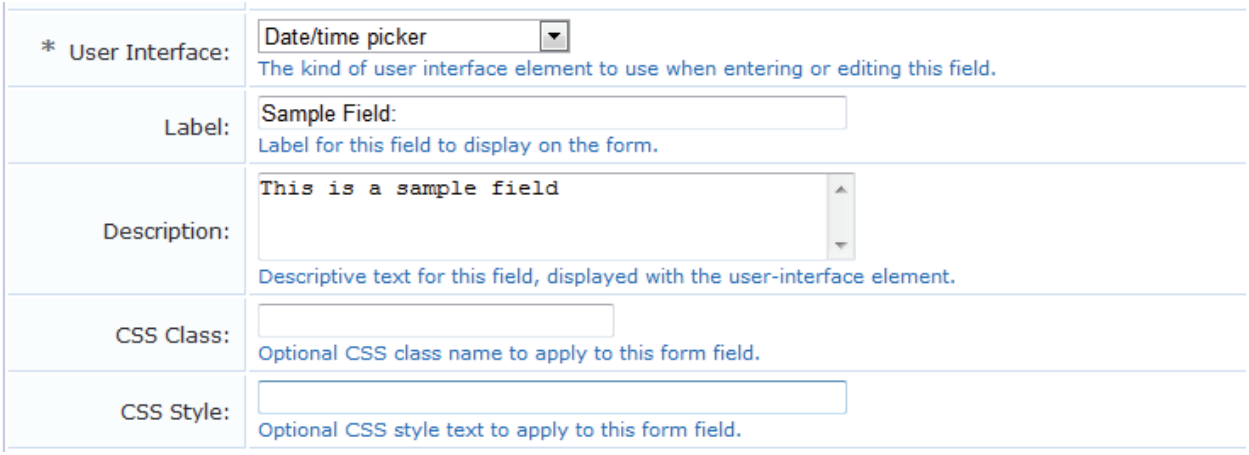
How this works: Suppose the first two check boxes are selected (in this example, with keys "one" and "two"). The incoming value for the field will be an array containing 2 elements, which can be written as `array("one", "two")`. The NwaImplodeComma conversion is applied, which converts the array value into the string value "one,two", which is then used as the value for the field. Finally, when the form is displayed and the value needs to be converted back from a string, the NwaExplodeComma display

function is applied, which turns the “one,two” string value into an array value `array("one", "two")`, which is used by the checklist to mark the first two items as selected.

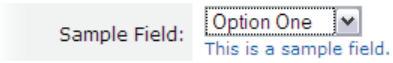
- **Date/time picker** – A text field is displayed with an attached button that displays a calendar and time chooser. A date may be typed directly into the text field, or selected using the calendar:

The text value typed is submitted with the form. If using a date/time picker, you should validate the field value to ensure it is a date.

Certain guest account fields, such as **expire_time** and **schedule_time**, require a date/time value to be provided as a UNIX time value. In this case, the conversion and display formatting options should be used to convert a human-readable date and time to the equivalent UNIX time and vice versa.

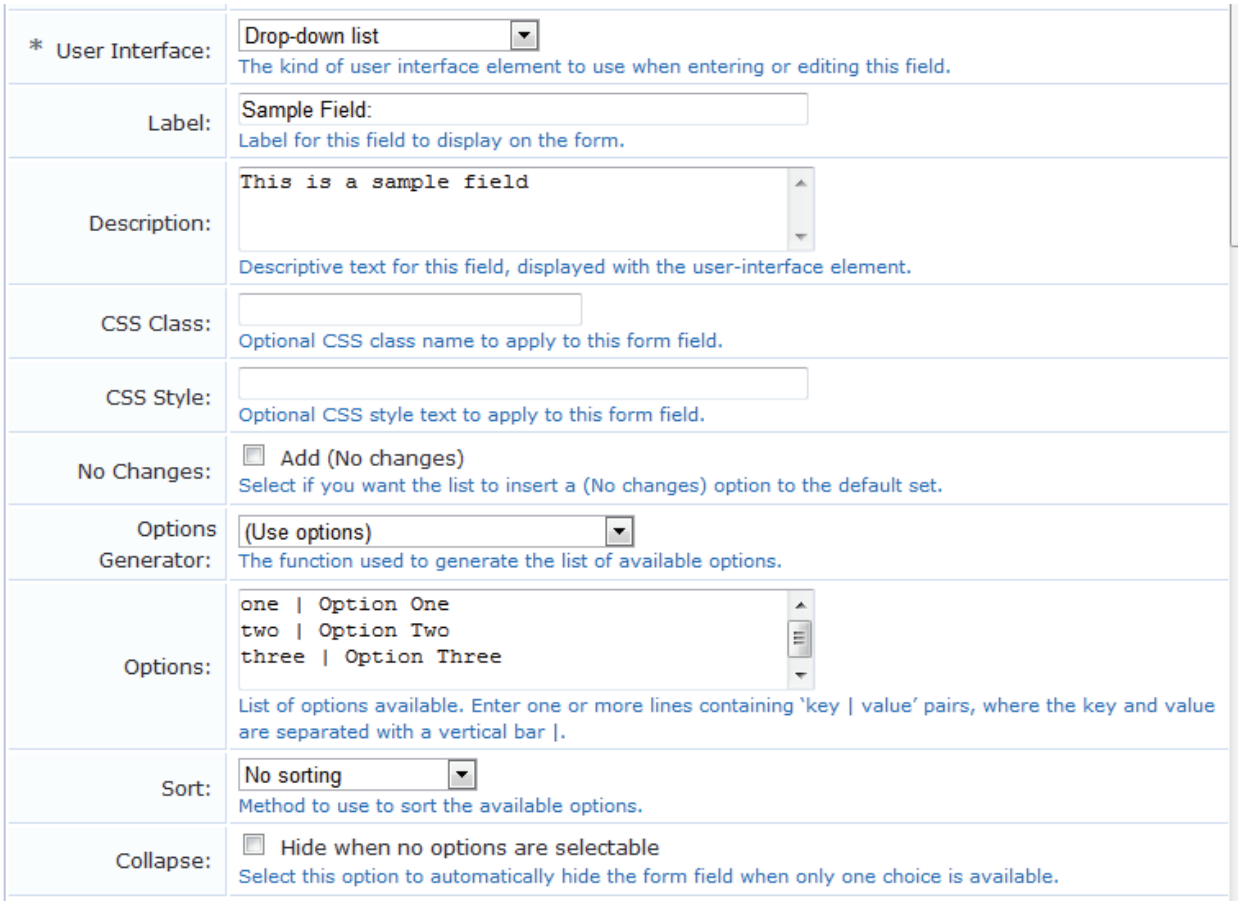


- **Drop-down list** – The field is displayed allowing a single choice from a drop-down list.



The text displayed for each option is the value from the options list. When the form is submitted, the key of the selected value becomes the value of the field.

If the “Hide when no options are selectable” check box is selected, and there is only a single option in the drop-down list, it will be displayed as a static text item rather than as a list with only a single item in it.

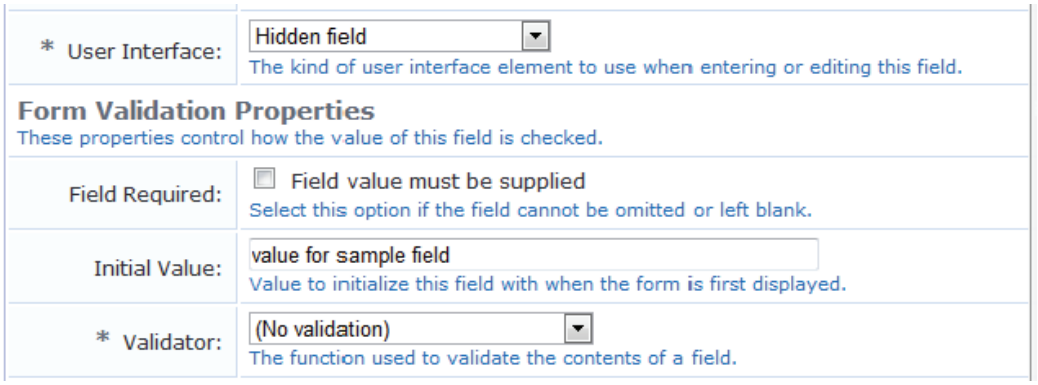


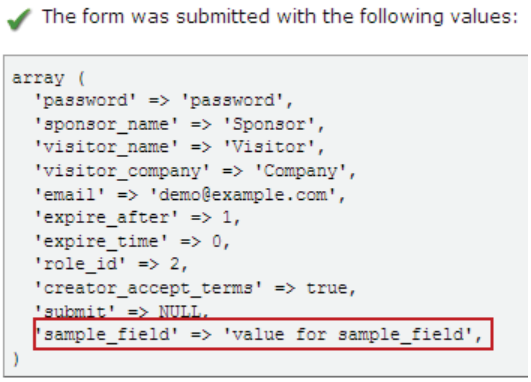
- **File upload** – Displays a file selection text field and dialog box (the exact appearance differs from browser to browser).

  File uploads cannot be stored in a custom field. This user interface type requires special form implementation support and is not recommended for use in custom fields.

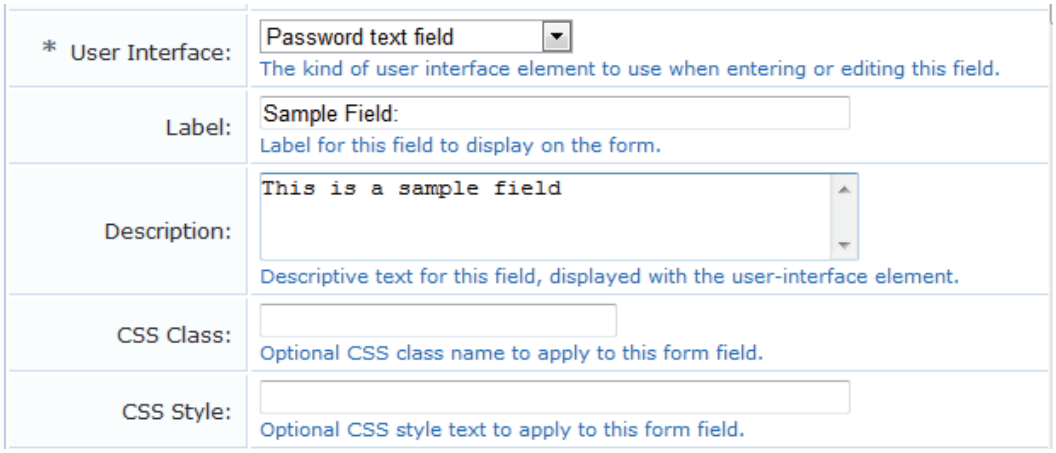
- **Hidden field** – If Hidden Field is selected in the User Interface drop-down list, the field is not displayed to the user, but is submitted with the form. This option is often used to force a specific value such as a user's role or an expiration date. However, it is possible for someone to use browser tools to modify the initial value when the form is submitted. If the value should be forced, use the **Force Value** setting under **Advanced Properties** to ensure the value cannot be overridden. For more information, see "Advanced Form Field Properties".

  To set the value to submit for this field, use the **Initial Value** option in the form field editor.



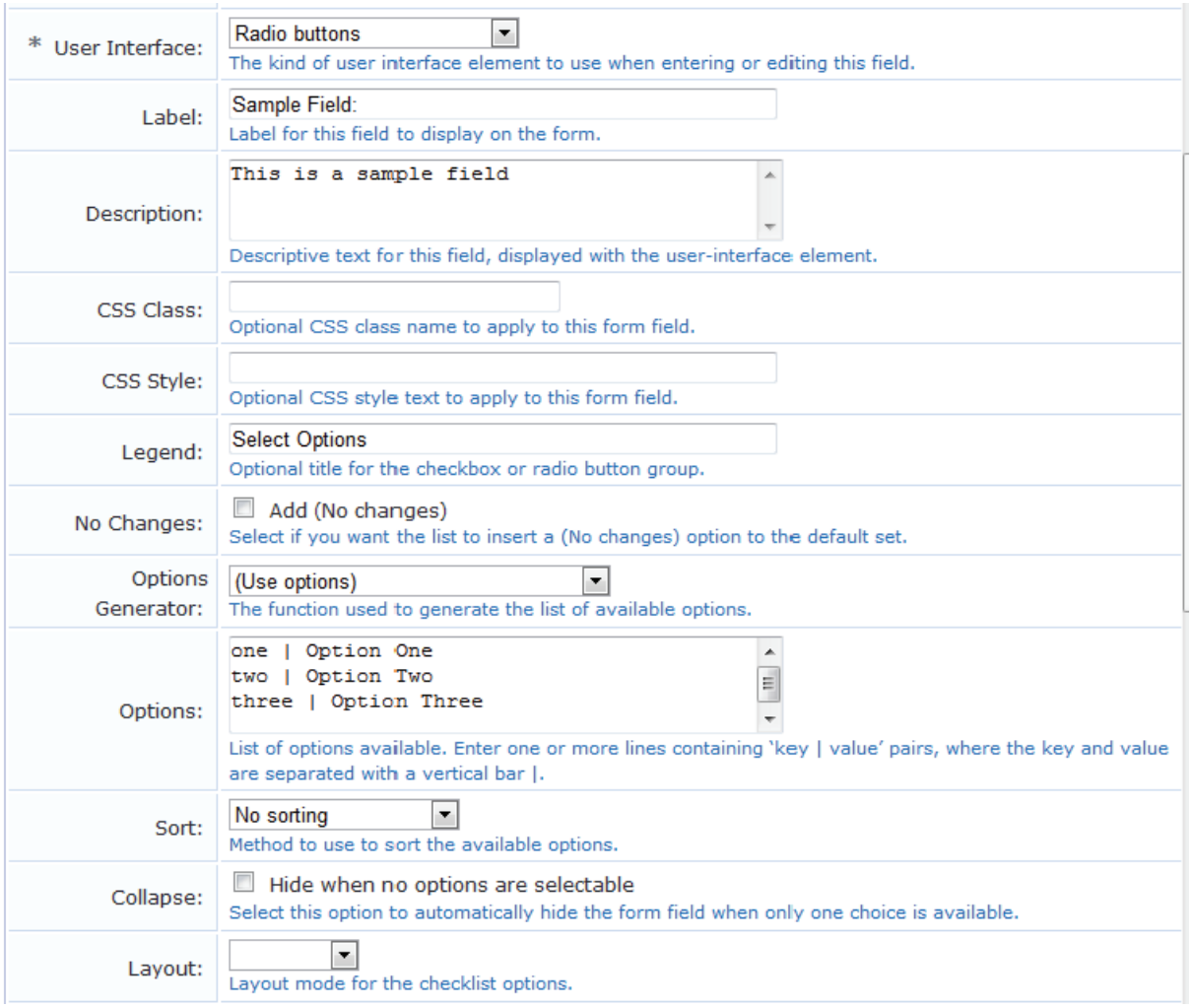


- **Password text field** – The field is displayed as a text field, with input from the user obscured. The text typed in this field is submitted as the value for the field.



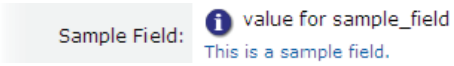
- **Radio buttons** – The field is displayed as a group of radio buttons, allowing one to be selected, as shown below:

The text displayed for each option is the value from the options list. When the form is submitted, the key of the selected value becomes the value of the field.



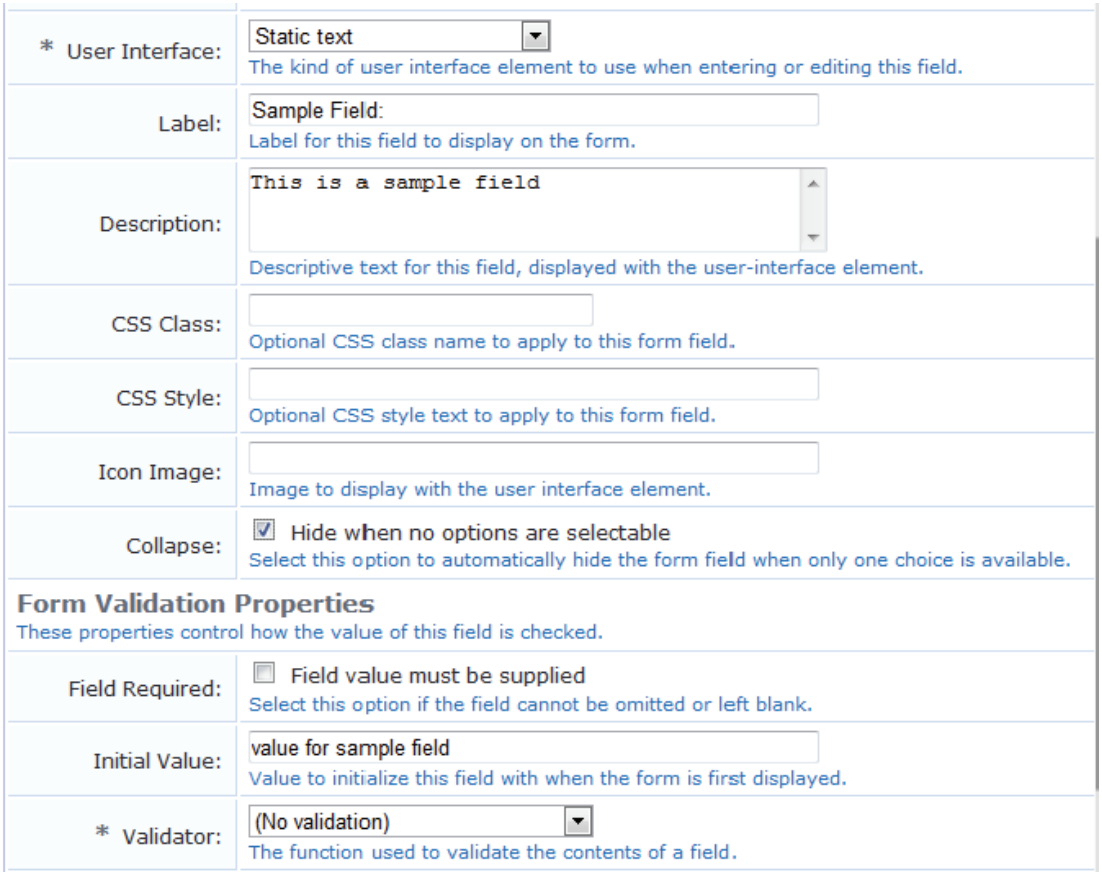
The "Vertical" and "Horizontal" layout styles control whether the radio buttons are organized in top-to-bottom or left-to-right order. The default is "Vertical" if not specified.

- **Static text** – The field's value is displayed as a non-editable text string. An icon image may optionally be displayed before the field's value. A hidden element is also included for the field, thereby including the field's value when the form is submitted.



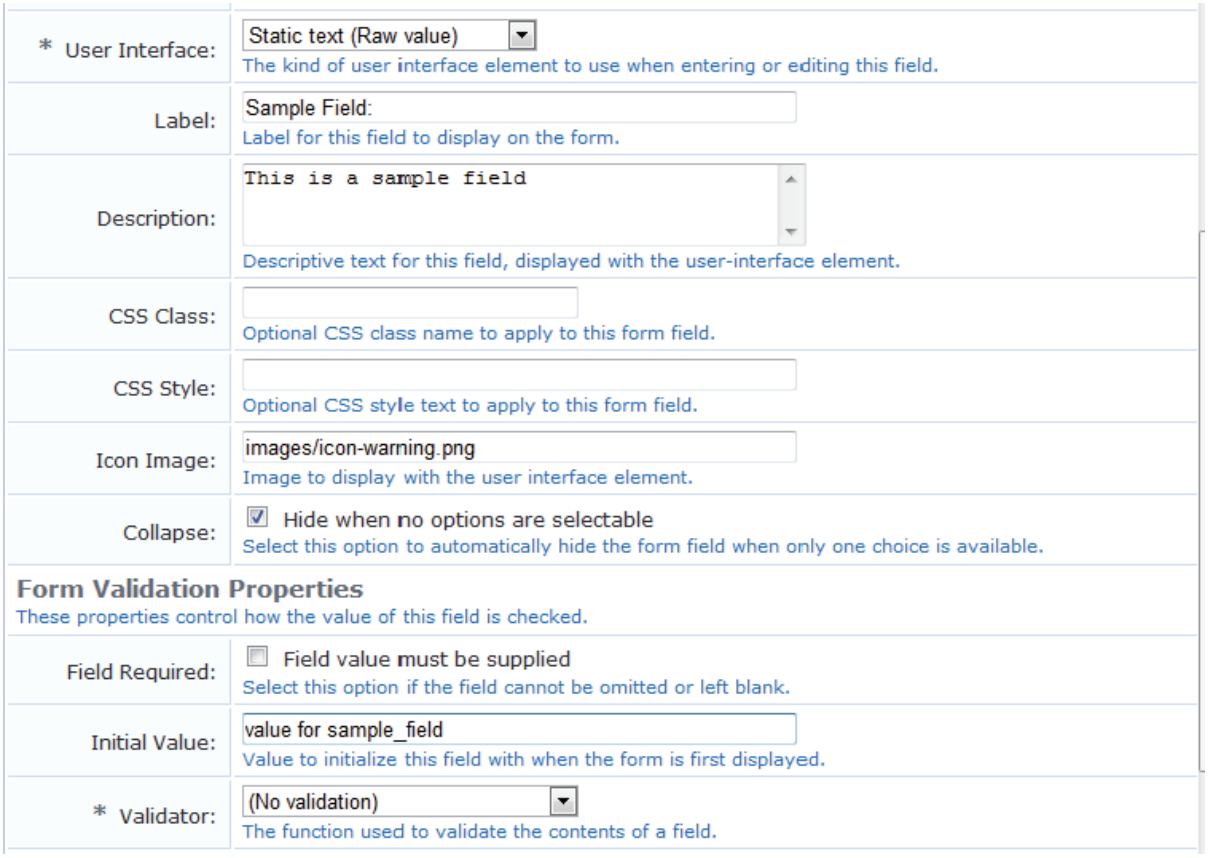
If the **Hide when no options are selectable** check box is selected in the **Collapse** row, the field will be hidden if its value is blank.

To set the value of this field, use the **Initial Value** option in the **Form Validation Properties** area of the form field editor.



- **Static text (Raw value)** – The field's value is displayed as a non-editable text string. HTML characters in the value are not escaped, which allows you to display HTML markup such as images, links and font formatting.

Use caution when using this type of user interface element, particularly if the field's value is collected from visitors. Allowing HTML from untrusted sources is a potential security risk.

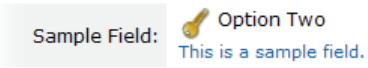


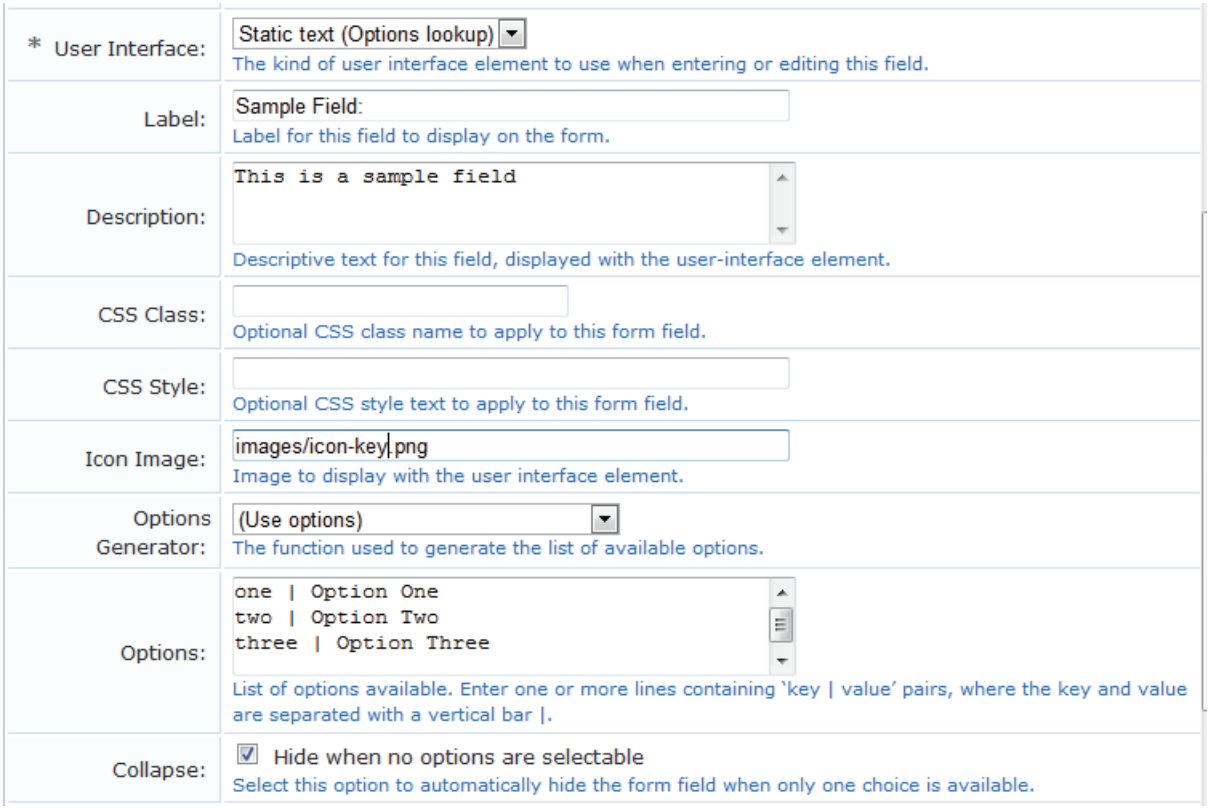
If the **Hide when no options are selectable** check box is selected in the **Collapse** row, the field will be hidden if its value is blank.

To set the value of this field, use the **Initial Value** option in the **Form Validation Properties** area of the form field editor.

- **Static text (Options lookup)** – The value of the field is assumed to be one of the keys from the field's option list. The value displayed is the corresponding value for the key, as a non-editable text string.

  An icon image may optionally be displayed before the field's value. A hidden element is also included for the field, thereby including the field's value when the form is submitted.

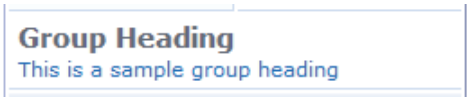




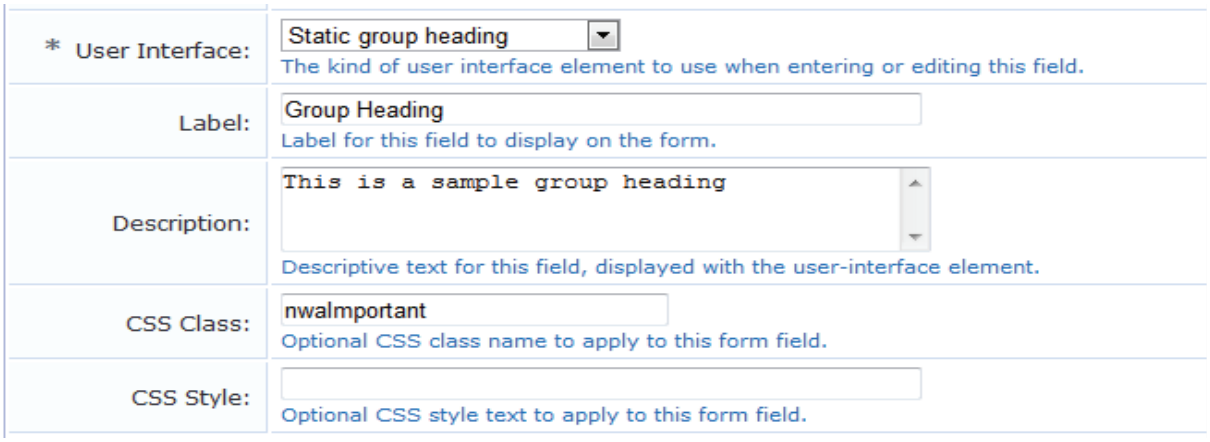
If the **Hide when no options are selectable** check box is selected in the **Collapse** row, the field will be hidden if its value is blank.

To set the value of this field, use the **Initial Value** option in the **Form Validation Properties** area of the form field editor.

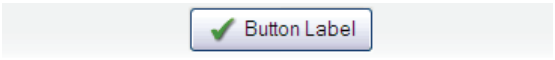
- **Static group heading** – The label and description of the field is used to display a group heading on the form, as shown below. The field's value is not used, and the field is not submitted with the form.

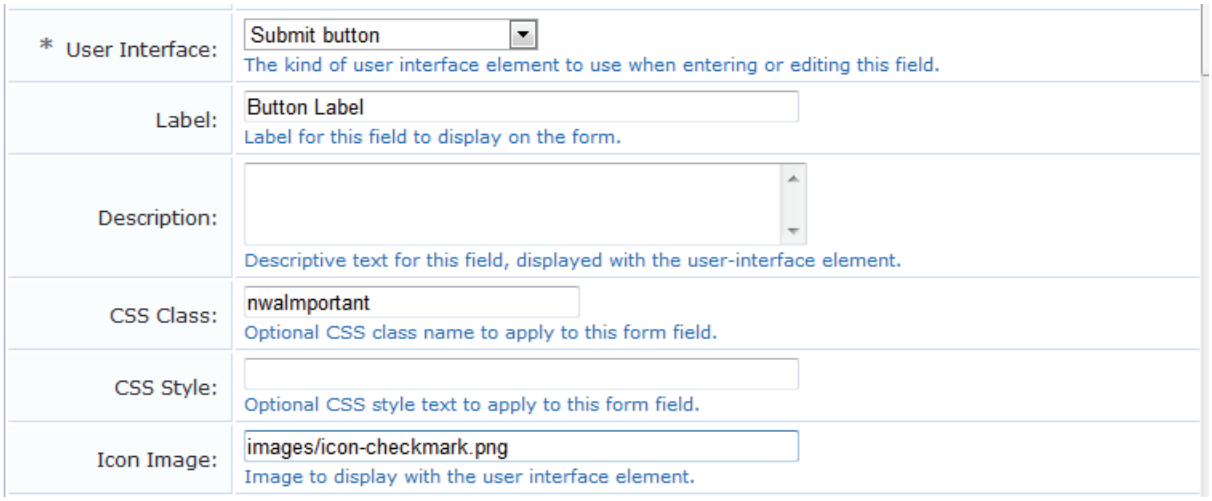


When using this user interface element, it is recommended that you use the "nwaImportant" CSS class to visually distinguish the group heading's title.



- **Submit button** – The field is displayed as a clickable form submit button, with the label of the field as the label of the button.

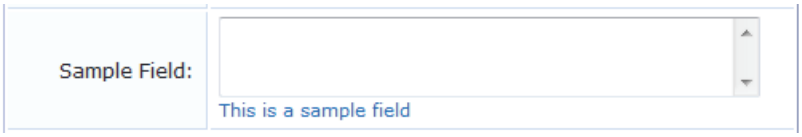


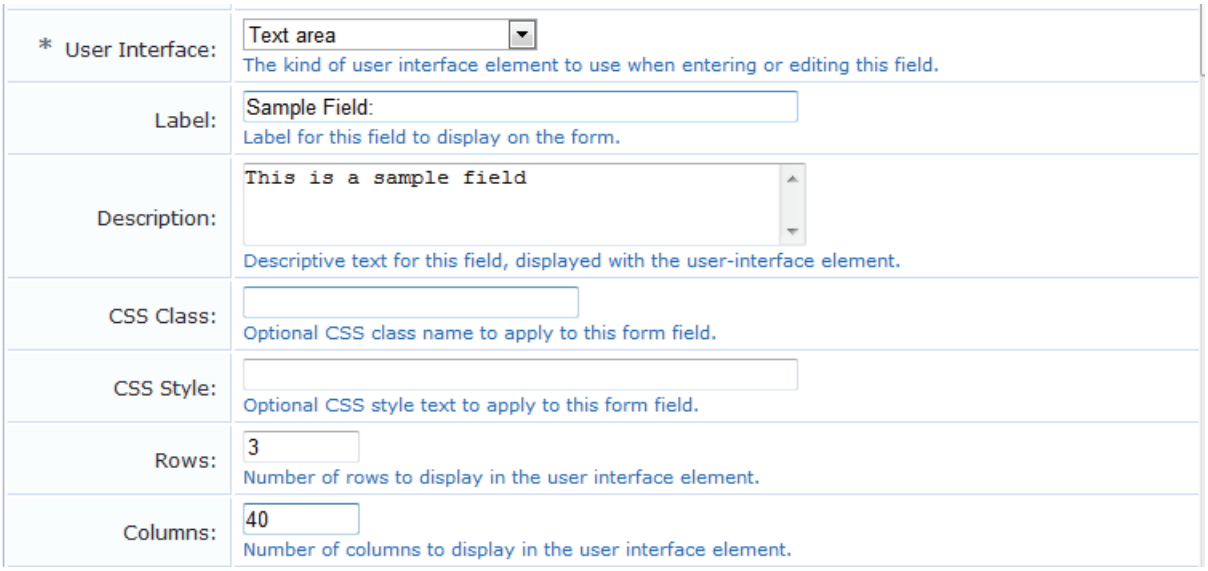


The description is not used. The field's value is ignored, and will be set to NULL when the form is submitted. To place an image on the button, an icon may be specified.

To match the existing user interface conventions, you should ensure that the submit button has the highest rank number and is displayed at the bottom of the form.

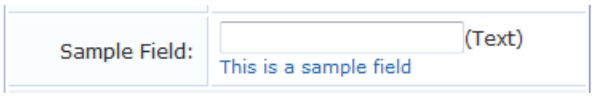
- **Text area** – The field is displayed as a multiple-line text box. The text typed in this box is submitted as the value for the field.



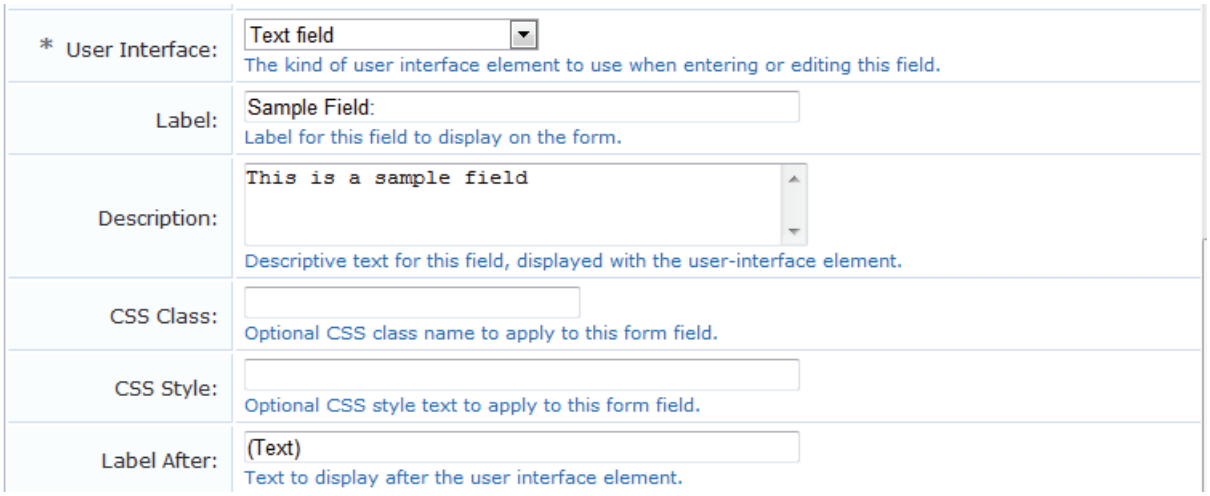


It is recommended that you specify the desired minimum dimensions of the text area, either with the **Rows** and **Columns** options, or by specifying a width in the **CSS Style** option (for example, "width: 460px; height: 100px;" specifies a 460 x 100 pixel minimum area).

- **Text field** – The field is displayed as a single-line text box. The text typed in this box is submitted as the value for the field.

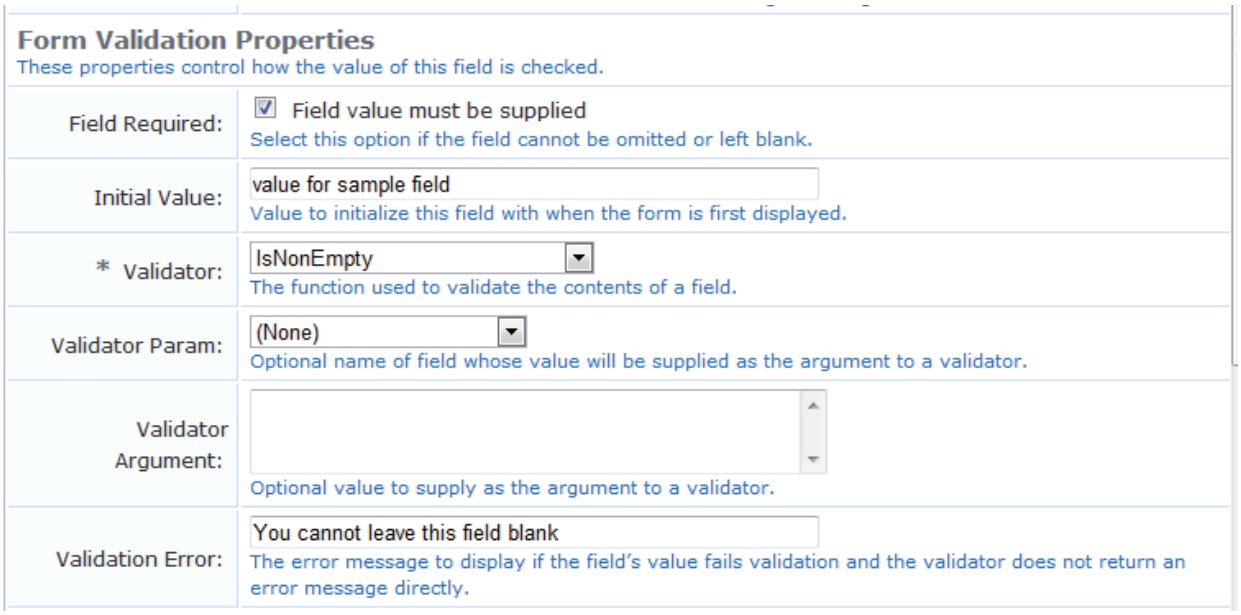


A short text label may be placed after the text box using the **Label After** option.



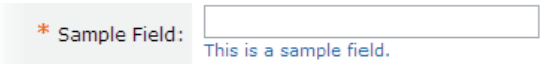
Form Validation Properties

The form validation properties control the validation of data entered into a form. By specifying appropriate validation rules, you can detect when users attempt to enter incorrect data and require them to correct their mistake.



The initial value for a form field may be specified. Use this option when a field value has a sensible default. The initial value should be expressed in the same way as the field's value. In particular, for drop-down list and radio button selections, the initial value should be the key of the desired default option. Likewise, for date/time fields that have a display function set, the initial value should be a value that can be passed to the display function.

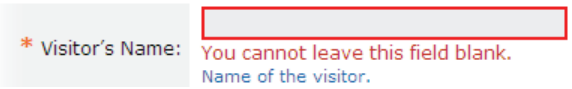
Select the **Field value must be supplied** check box to mark the field as a required field. Required fields are marked with an asterisk, as shown below:



An optional field may be left blank. In this case, the field is not validated as there is no value for the field. However, any value that is supplied for an optional field is subject to validation checks.

All values supplied for a required field are always validated, including blank values.

Validation errors are displayed to the user by highlighting the field(s) that are in error and displaying the validation error message with the field:



All fields must be successfully validated before any form processing can take place. This ensures that the form processing always has user input that is known to be valid.

To validate a specific field, choose a validator from the drop-down list. See "Form Field Validation Functions" in the "Reference" chapter for a description of the built-in validators.

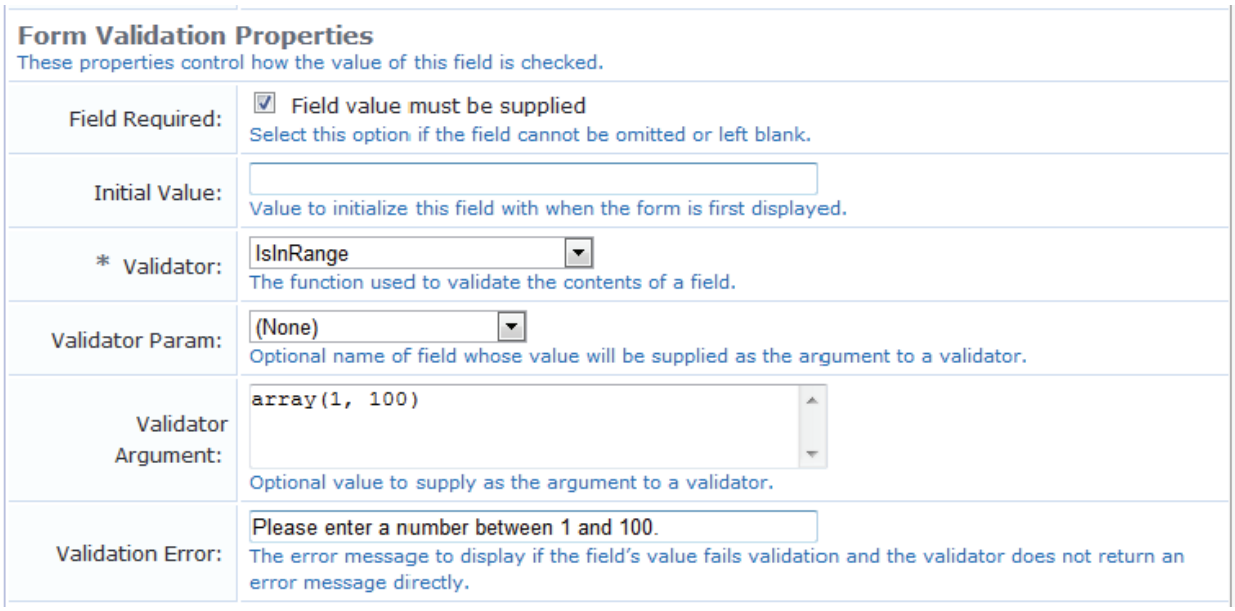
The Validator Param is the name of a field on the form, the value of which should be passed to the validator as its argument. This could be used to validate one field based on the contents of another. However, in most deployments this does not need to be set.

Set the Validator Param to its default value, "(Use argument)", to provide a fixed value as the argument to the validator.

The Validator Argument is used to provide further instructions to the selected validator. Not all validators require an argument; a validator such as **IsValidEmail** is entirely self-contained and will ignore the Validator Argument. Validators such as **IsEqual**, **IsInRange** and **IsRegexMatch** use the argument to perform validation.

Examples of Form field Validation

Example 1 – To create a form field that requires an integer value between 1 and 100 (inclusive) to be provided, use the following settings in the form field editor:



NOTE

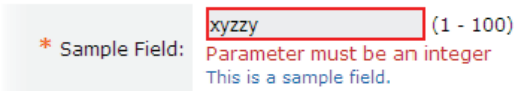
The form field will contain an integer value, so you should set the field's type to Integer when creating it.

Use the PHP syntax **array(1, 100)** to specify the minimum and maximum values for the **IsInRange** validator. After saving changes on the form, this value will be internally converted to the equivalent code:

```
array (
  0 => 1,
  1 => 100,
)
```

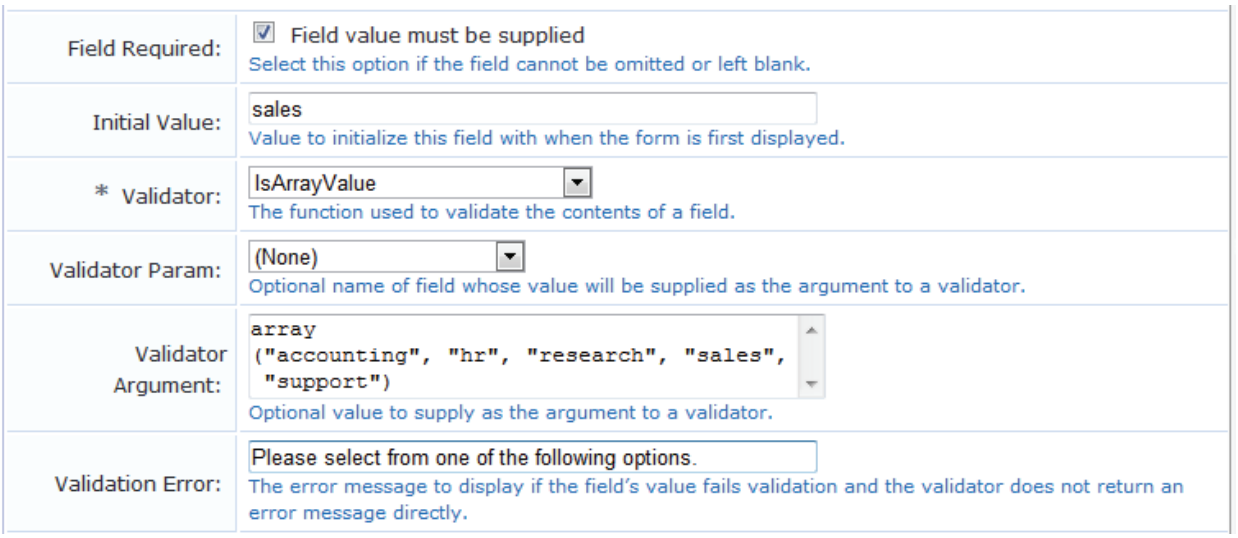
With these validator settings, users that enter an invalid value will now receive a validation error message:

Furthermore, note that blank values, or non-numeric values, will result in a different error message:



The reason for this is that in this case, the validation has failed due to a type error – the field is specified to have an integer type, and a blank or non-numeric value cannot be converted to an integer. To set the error message to display in this case, use the Type Error option under the Advanced Properties.

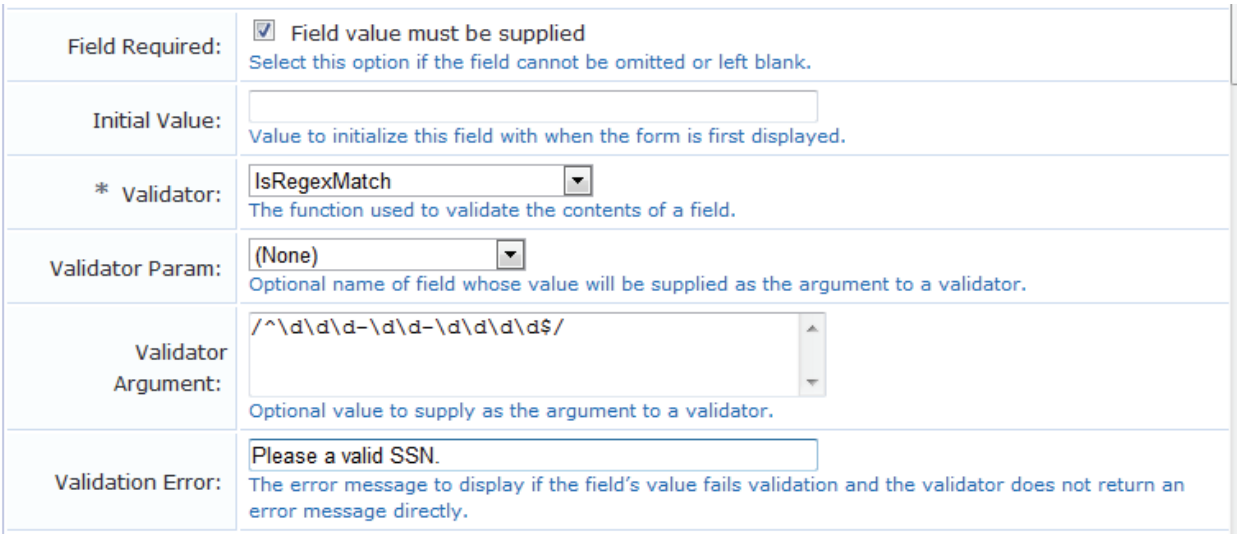
Example 2 – To create a form field that accepts one of a small number of string values, use the following settings in the form field editor:



This example could be used for a string field named **visitor_department**. Because the values are known in advance, a drop-down list is the most suitable user interface. An initial value for the form field, as shown above, could be used if most visitors are in fact there to visit the sales team.

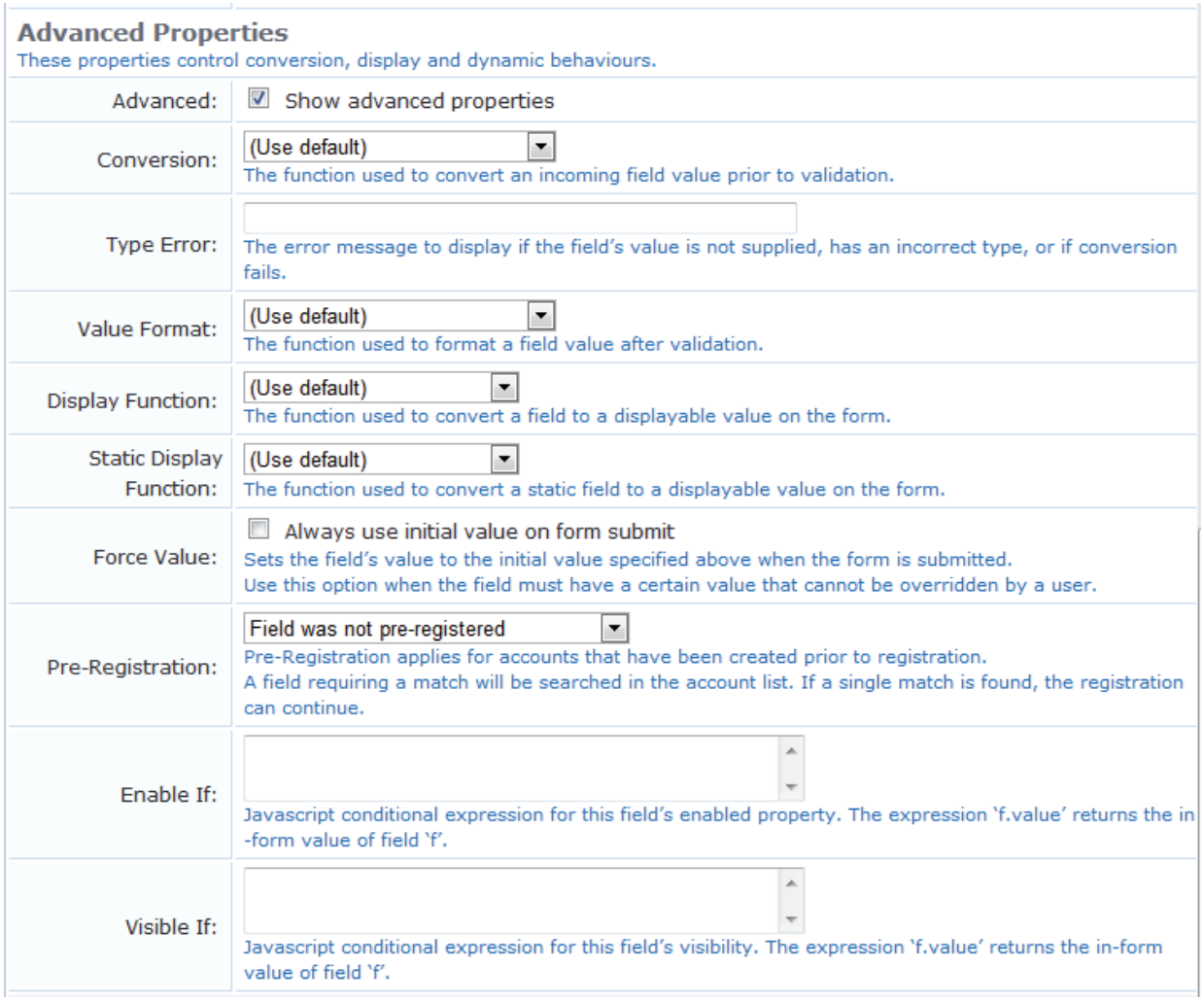
To match against a list of options used for a drop-down list or set of radio buttons, you can use the IsInOptionsList validator.

Example 3 – To create a form field that validates U.S. social security numbers using a regular expression, use the following settings in the form field editor:



Note that the regular expression used here includes beginning and ending delimiters (in this case the **/** character), and ensures that the whole string matches by the start-of-string marker **^** and the end-of-string marker **$**. The construct **\d** is used to match a single digit. Many equivalent regular expressions could be written to perform this validation task. See "Regular Expressions" in the "Reference" chapter for more information about regular expressions.

Advanced Form Field Properties



The Advanced Properties control certain optional form processing behaviors. You can also specify JavaScript expressions to build dynamic forms similar to those found elsewhere in the application.

On the Customize Form Fields page, select the **Show advanced properties** check box to display the advanced properties in the form field editor.

The **Conversion**, **Value Format**, and **Display Function** options can be used to enable certain form processing behavior. See "Form Field Conversion Functions" and **"Form Field Display Formatting Functions"** in the "Reference" chapter.

In the **Force Value** row, use the **Always use initial value on form submit** check box to prevent attempts to override the value set for a field. When this option is set, if a user modifies the field's value, it reverts to the specified initial value when the form is submitted. A similar effect can be achieved by using appropriate validation rules, but selecting this check box is easier. Using this option is recommended for hidden fields, particularly those related to security, such as role ID or expiration date.

For pre-registered guest accounts, some fields may be completed during pre-registration and some fields may be left for the guest to complete at registration. You can use the **Pre-Registration** field to specify whether the guest's entry must match the preliminary value provided for a field during pre-registration.

- If a value was not provided for a field when the account was created, choose **Field was not pre-registered** from the drop-down list.
- If a preliminary value was provided for the field but the guest's entered value does not need to match case or all characters, choose **Guest must supply field** from the drop-down list. For example, a bulk

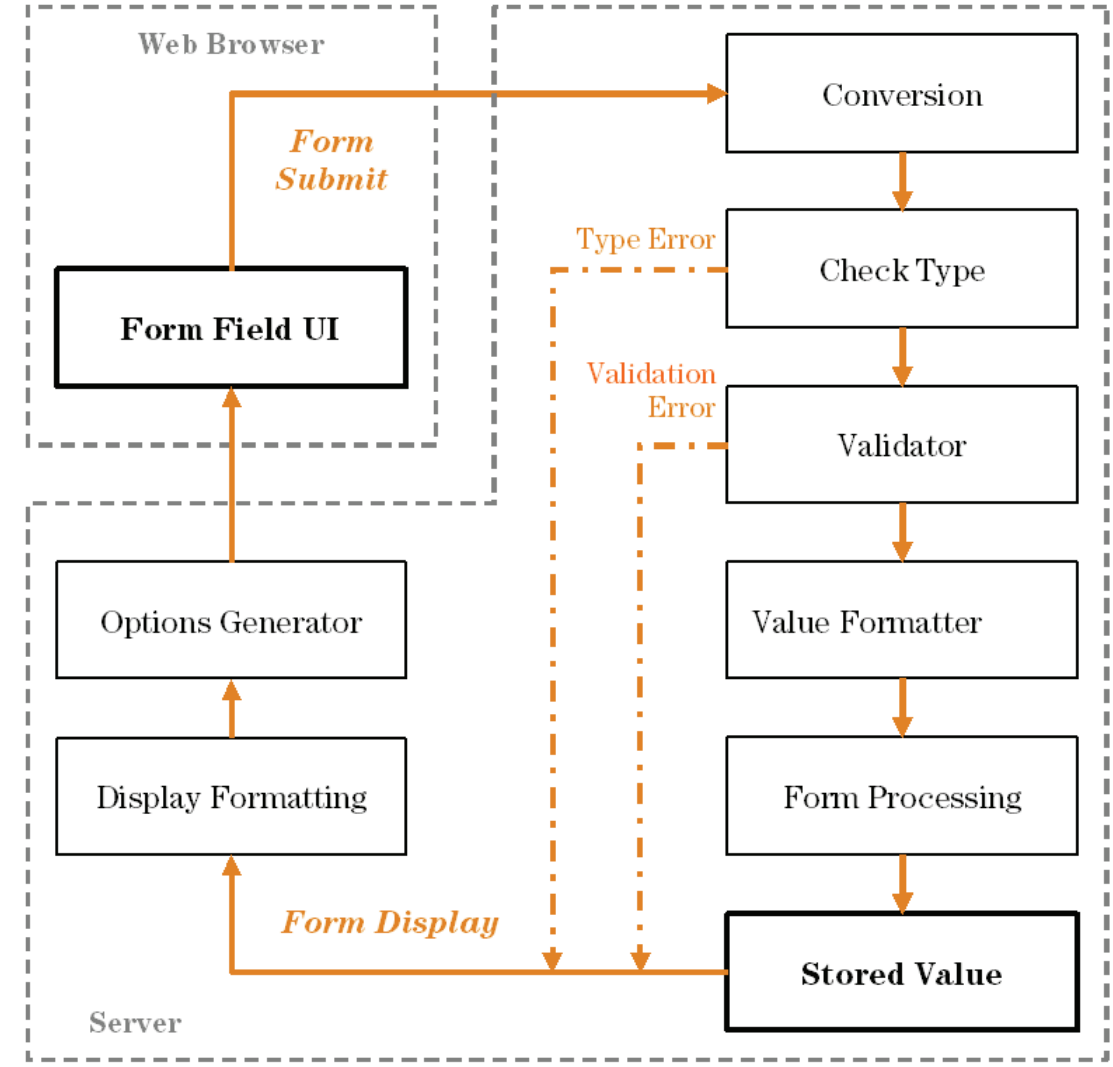
account creation might use random usernames, and each visitor's entry in that field would not need to match exactly.

- If a preliminary value was provided for the field and the guest's entered value must match case or all characters, choose **Guest must supply field (match case)** from the drop-down list. If the guest's entry does not successfully match the preregistered value, the account registration will not succeed. For example, if a list of email addresses and phone numbers was imported for pre-registration, each visitor's entries for those fields at registration must match.

Form Field Validation Processing Sequence

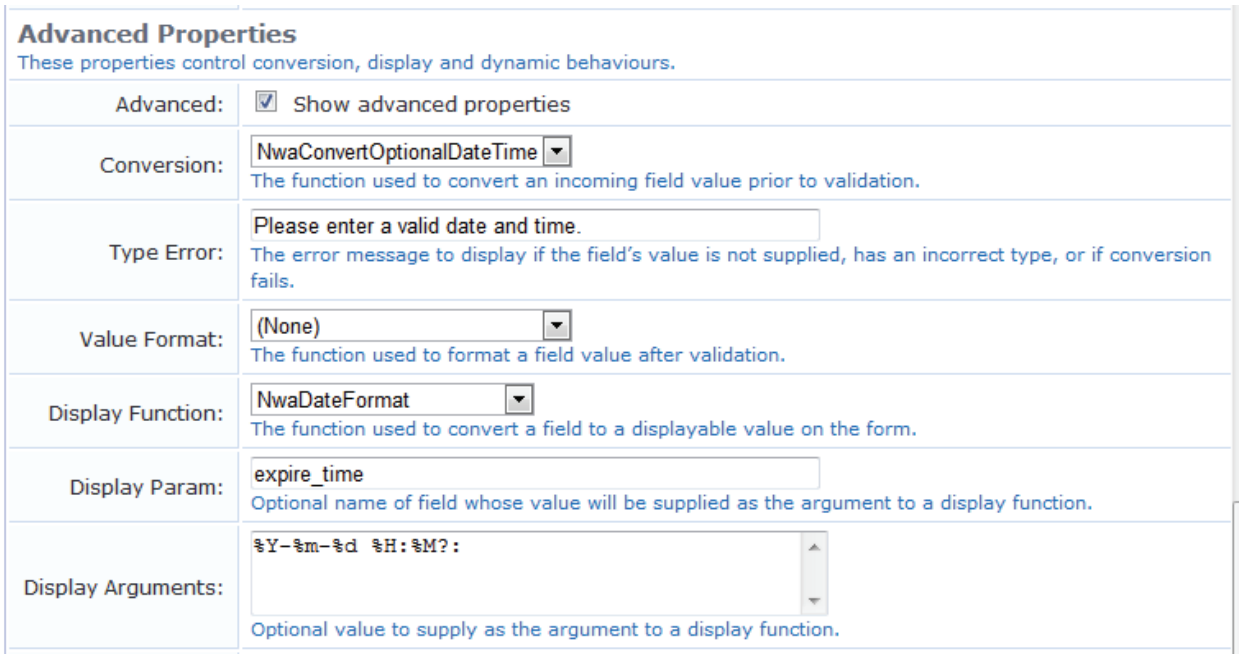
The following figure shows the interaction between the user interface displayed on the form and the various conversion and display options.

Figure 24 *Steps involved in form field processing*



The Conversion step should be used when the type of data displayed in the user interface is different from the type required when storing the field.

For example, consider a form field displayed as a date/time picker, such as the **expire_time** field used to specify an account expiration time on the **create_user** form. The user interface is displayed as a text field, but the value that is required for the form processing is a UNIX time (integer value).

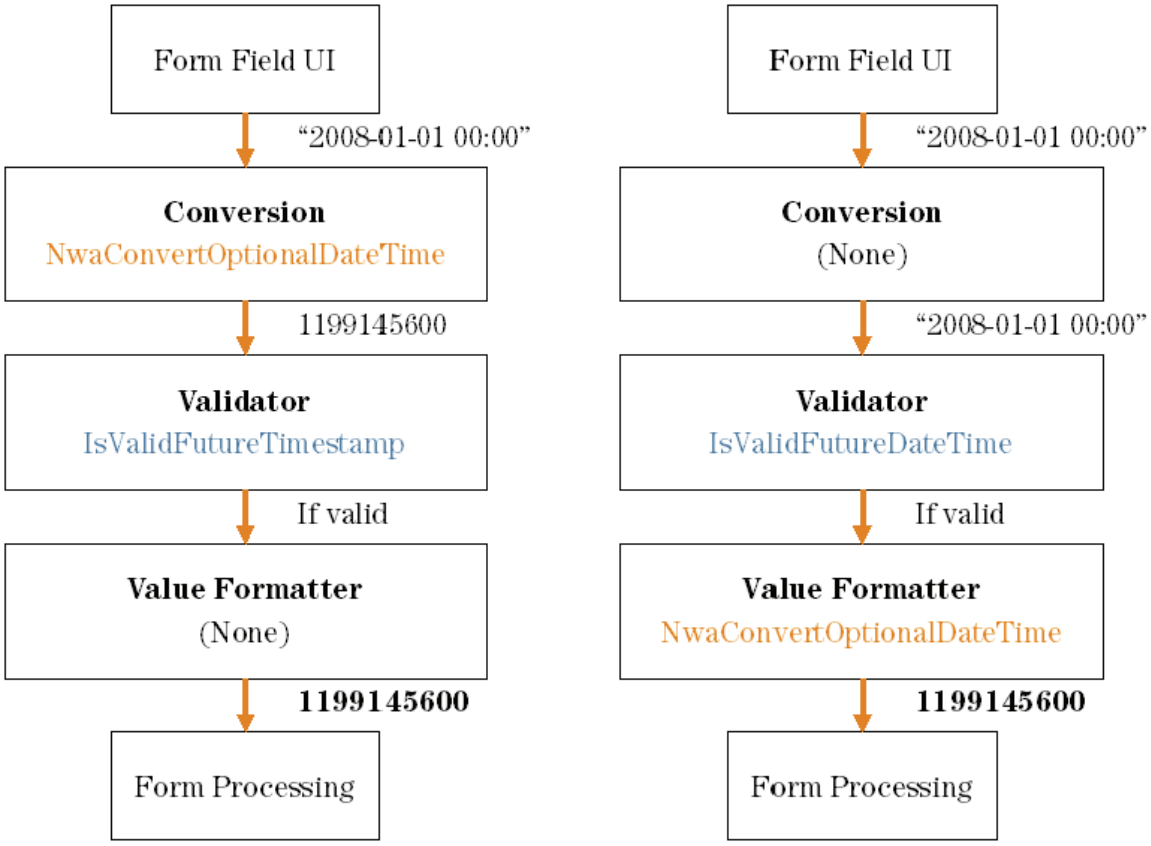


In this case, the Conversion function is set to NwaConvertOptionalDateTime to convert the string time representation from the form field (for example, "2008-01-01") to UNIX time (for example, 1199145600).

The Validator for the **expire_time** field is **IsValidFutureTimestamp**, which checks an *integer* argument against the current time.

The Value Formatter is applied after validation. This may be used in situations where the validator requires the specific type of data supplied on the form, but the stored value should be of a different type. In the **expire_time** field example, this is not required, and so the value formatter is not used. However, if the Conversion function had not been used, and the Validator had been set to **IsValidFutureDateTime** (which checks a *string* date/time value), then the Value Formatter would need to be set to NwaConvertOptionalDateTime to perform the data conversion before the form processing.

A comparison of these two approaches is shown below to illustrate the difference:



When using a Conversion or Value Format function, you will almost always have to set up a Display Function for the form field. This function is used to perform the conversion in the reverse direction – between the internal stored value and the value displayed in the form field.

See "Form Field Conversion Functions" in the "Reference" chapter for a detailed list of the options available to you for the Conversion and Value Format functions.

The **Display Param** is the name of a form field, the value of which will be passed to the Display Function. In almost all cases this option should contain the name of the form field.

Display Arguments are available for use with a form field and are used to control the conversion process. In the case of the **expire_time** form field, the Display Function is set to **NwaDateFormat** to perform a conversion from a UNIX time to a date/time string, and the Display Argument specifies the format to use for the conversion.

See "Form Field Display Formatting Functions" in the "Reference" chapter for a detailed list of the options available to you for the Display Function and Static Display Function.

The **Enable If** and **Visible If** options in the form field editor allow you to specify JavaScript expressions. The result obtained by evaluating these expressions is used to enable/disable, or show/hide the form field in real time, while an operator is using the form.

Unlike the other parts of the form field editor, the **Enable If** and **Visible If** expressions are evaluated by the operator's Web browser. These expressions are not used by the server for any other purpose.

The expression must be a Boolean expression in the JavaScript language; statements and other code should not be included as this will cause a syntax error when the form is displayed in a Web browser.

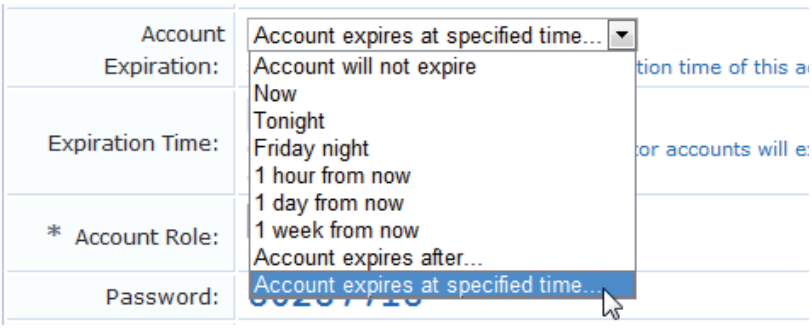
Because of the scoping rules of JavaScript, all of the user interface elements that make up the form are available as variables in the local scope with the same name as the form field. Thus, to access the current

value of a text field named **sample_field** in a JavaScript expression, you would use the code **sample_field.value**.

Most user interface elements support the **value** property to retrieve the current value. For check boxes, however, use the **checked** property to determine if the check box is currently selected.

The most practical use for this capability is to hide a form field until a certain value of some other related field has been selected.

For example, the default **create_user** form has an **Account Expiration** drop-down list. One of the values in this list is special: the **-1** option displays the value **Account expires at a specified time...**



When this option is selected, the form expands to include the **Expires After** row, allowing the user to specify a time other than one of the options in the list.

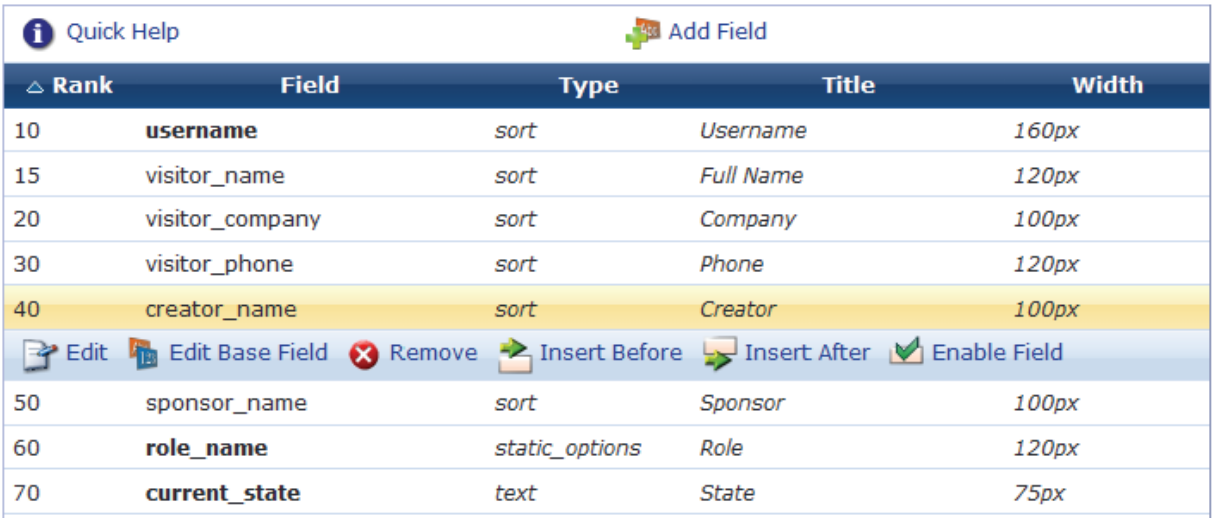
The **expire_time** field uses the JavaScript expression **expire_after.value < 0** for the **Visible If** option. When the -1 option has been selected, this condition will become true and the field will be displayed.

Additional examples of the **Visible If** conditional expressions can be found in the **guest_edit** form.

Editing Views

A view consists of one or more columns, each of which contains a single field. You can change which fields are displayed and how each field is displayed. You can also define your own fields using the Customize Fields page, and then add them to a view by choosing appropriate display options for each new column.

To add a new field to a view, reorder the fields, or make changes to an existing field in a view, select the view in the **Customize Forms & Views** list and click the **Edit Fields** link. This opens the Customize View Fields editor.



View fields have a **Rank** number, which specifies the relative ordering of the columns when displaying the view. The Customize View Fields editor always shows the columns in order by rank.

The **Type** of each field is displayed. This controls what kind of user interface element is used to display the column, and whether the column is to be sortable or not. The **Title** of the column and the **Width** of the column are also shown in the list view. Values displayed in *italics* are default values defined for the field being displayed.

Click a view field in the list view to select it.

Use the **Edit** link to make changes to an existing column using the View Field Editor. Any changes made to the field using this editor will apply only to this field on this view.

Use the **Edit Base Field** link to make changes to an existing field definition. Any changes made to the field using this editor will apply to all views that are using this field (except where the view field has already been modified to be different from the underlying field definition).

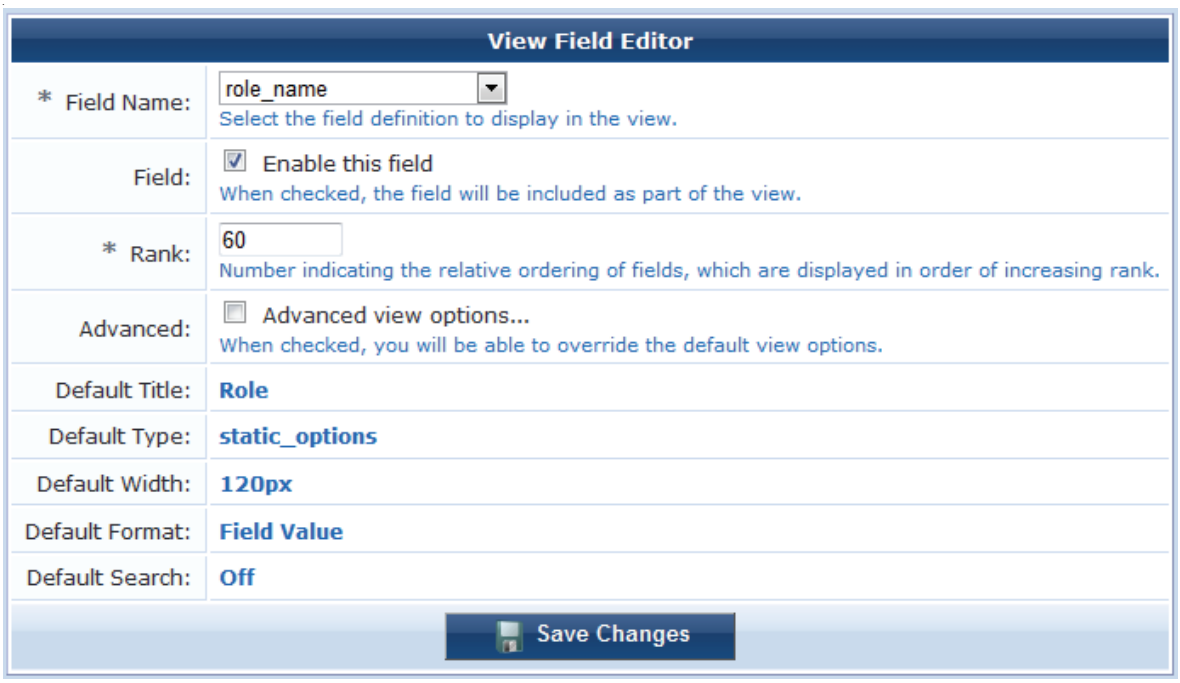
The **Insert Before** and **Insert After** links can be used to add a new column to the view. Clicking one of these links will open a blank view field editor and automatically set the rank number of the new column.

Use the **Enable Field** and **Disable Field** links to quickly turn the display of a column on or off.

Click the **Add Field** tab to add a new column to the view.

View Field Editor

The view field editor is used to control the data-display aspects of a column within the view.



Each column in a view displays the value of a single field.

To use the default view display properties for a field, you only need to select the field to display in the column and then click the **Save Changes** button.

To customize the view display properties, click the **Advanced view options...** check box.

The column type must be one of the following:

- **Text** – The column displays a value as text.
- **Sortable text** – The column displays a value as text, and may be sorted by clicking on the column heading.
- **Sortable text, case-insensitive** – The same as "Sortable text", but the column sorting will treat uppercase and lowercase letters the same.

- **Sortable numeric** – The column displays a numeric value, and may be sorted by clicking on the column heading.

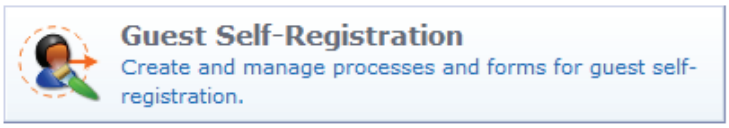
The Column Format may be used to specify how the field's value should be displayed. You may choose from one of the following:

- **Field Value** – The value of the field is displayed as plain text.
- **Field Value (Un-Escaped)** – The value of the field is displayed as HTML.
- **Boolean – Yes/No** – The value of the field is converted to Boolean and displayed as "Yes" or "No".
- **Boolean – Enabled/Disabled** – The value of the field is converted to Boolean and displayed as "Enabled" or "Disabled".
- **Boolean – On/Off** – The value of the field is converted to Boolean and displayed as "On" or "Off".
- **Date** – The value of the field is assumed to be a UNIX timestamp value and is displayed as a date and time.
- **Duration (from seconds)** – The value of the field is assumed to be a time period measured in seconds and is displayed as a duration (for example, "23 seconds", "45 minutes")
- **Duration (from minutes)** – The value of the field is assumed to be a time period measured in minutes and is displayed as a duration (for example, "45 minutes", "12 hours")
- **Use form options** – The value of the field is assumed to be one of the keys from the field's option list. The value displayed is the corresponding value for the key.
- **Custom expression…** – The Display Expression text area is displayed allowing a custom JavaScript expression to be entered. See "View Display Expression Technical Reference" in the "Reference" chapter for technical information about this display expression and a list of the functions that are available to format the value.

The Display Expression is a JavaScript expression that is used to generate the contents of the column. Generally, this is a simple expression that returns an appropriate piece of data for display, but more complex expressions can be used to perform arbitrary data processing and formatting tasks.

Customizing Self Provisioned Access

Guest self-registration allows an administrator to customize the process for guests to create their own visitor accounts.



The registration process consists of a data collection step (the 'register page') and a confirmation step (the 'receipt page').

You can define what information is collected from visitors on the registration page. New fields and data validation rules can be defined with the custom form editor. Specific details about the type of visitor accounts created are also set here.

The receipt page also includes a form, although typically this form will only contain static information about the guest account. Several different actions can be included on the receipt page, enabling visitors to obtain their receipt in different ways.

The receipt page can also be used to automatically log the guest into a Network Access Server, enabling them to start using the network immediately.

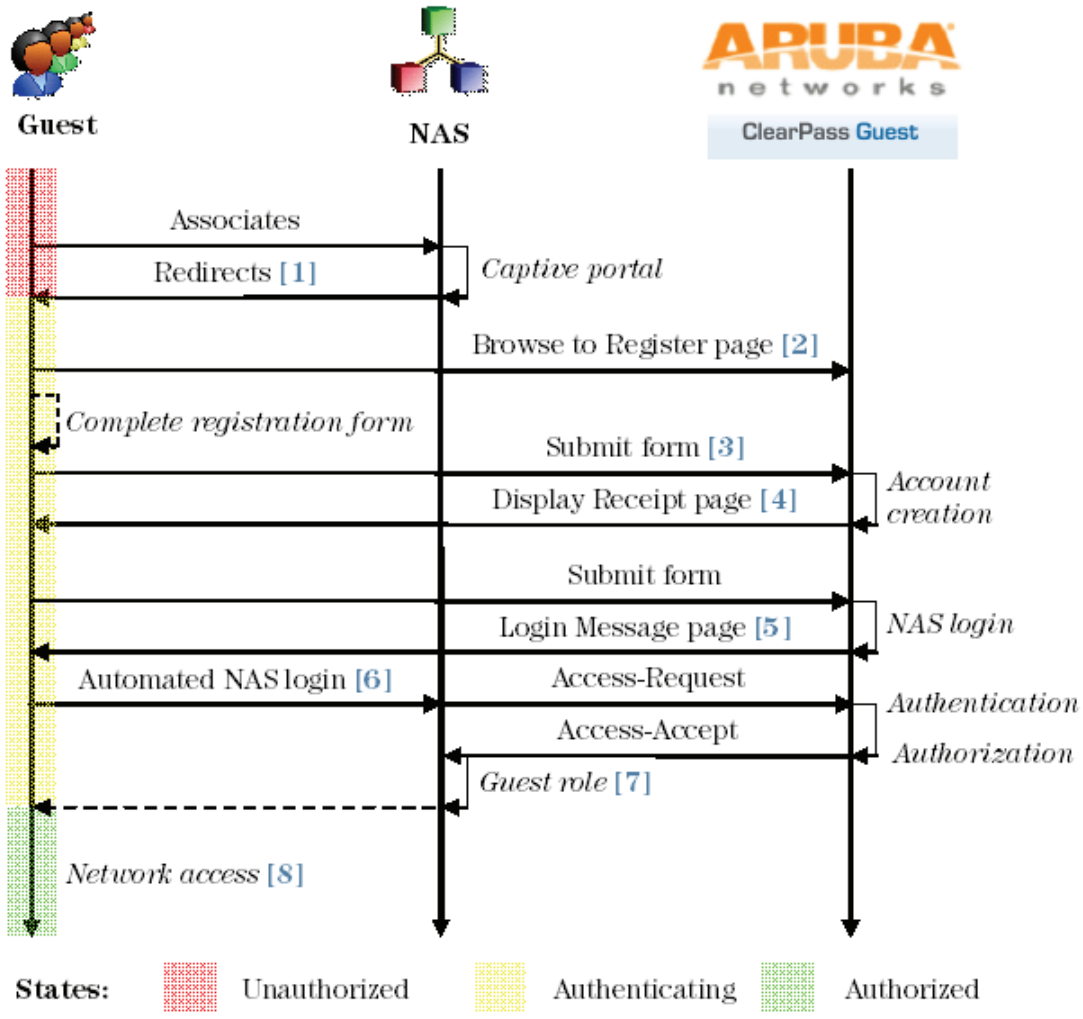
Detailed user interface customization can be performed for all parts of the self-registration process. You can define page titles, template code for the page header and footer, and choose a skin that controls the overall look and feel of self-registration. The default user interface customization can be disabled.

Self-Registration Sequence Diagram

To set up a captive portal with guest self-registration, configure your Network Access Servers to redirect guests to the URL of the 'Go To' link. To complete the portal, ensure that the NAS is configured to authorize users with the ClearPass Guest RADIUS server, and set up the self-registration NAS login to redirect registered guests back to the NAS.

This process is shown below.

Figure 25 *Sequence diagram for guest self-registration*



The captive portal redirects unauthorized users [1] to the register page [2]. After submitting the registration form [3], the guest account is created and the receipt page is displayed [4] with the details of the guest account. If NAS login is enabled, submitting the form on this page will display a login message [5] and automatically redirect the guest to the NAS login [6]. After authentication and authorization the guest's security profile is applied by the NAS [7], enabling the guest to access the network [8].

Creating a Self-Registration Page

To create a new guest self-registration page, go to **Configuration > Guest Self-Registration** and click the **Create new self-registration page** link. The **Customize Guest Registration** form is displayed.

The Register Page is the name of a page that does not already exist. There are no spaces in this name. This page name will become part of the URL used to access the self provisioning page. For example, the default "guest_register" page is accessed using the URL **guest_register.php**.

Click the **Save Changes** button to save the self registration page. A diagram of the self registration process is displayed.

Click the **Save and Continue** button to proceed to the next step of the setup.

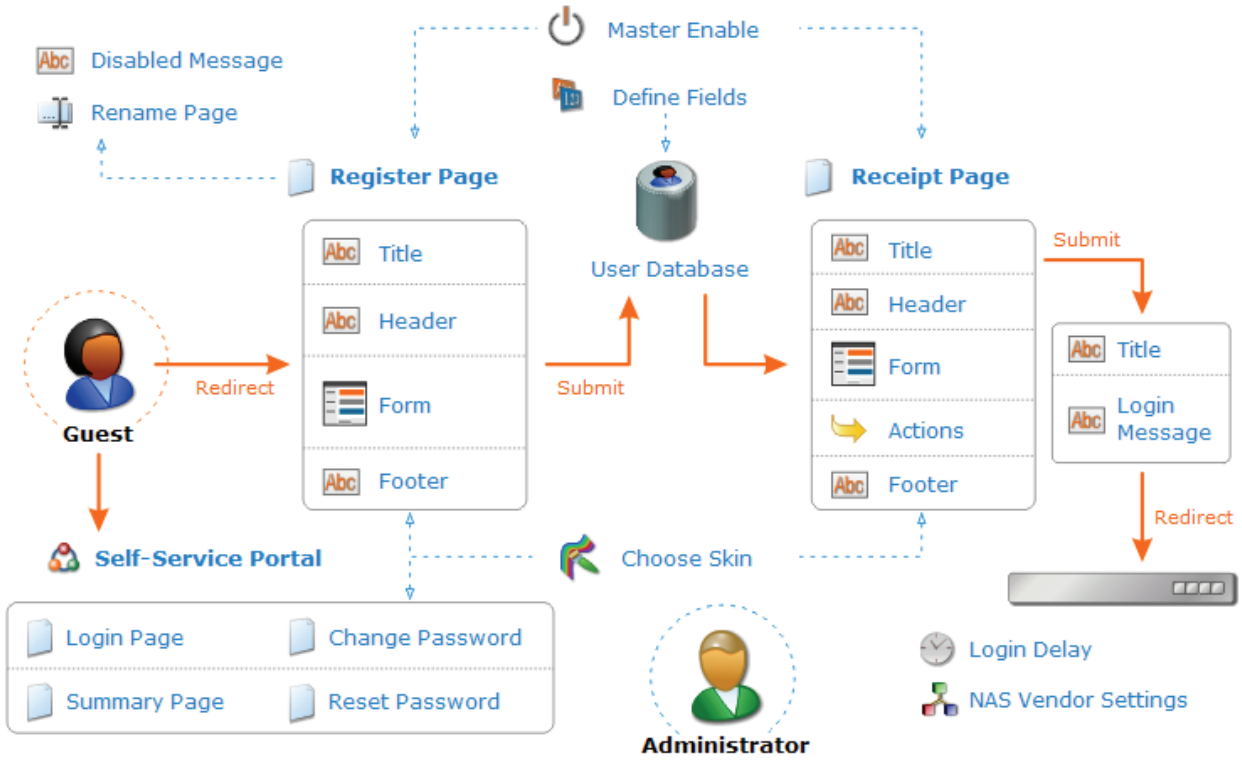
Once a self registration page has been created you are able to edit, delete, duplicate or go to it, providing self-registration has been enabled.

Editing Self-Registration Pages

The guest self-registration process is displayed in graphical form, shown below in Figure 26. The workflow for the guest is shown using solid orange arrows, while the administrator workflow is shown with dotted blue arrows. To access this page in the WebUI:

1. Navigate to **Configuration > Guest Self-Registration**
2. Select an entry in the **Guest Self-Registration** list, then click **Edit**.
3. The **Customize Guest Registration** workflow page appears, as shown below

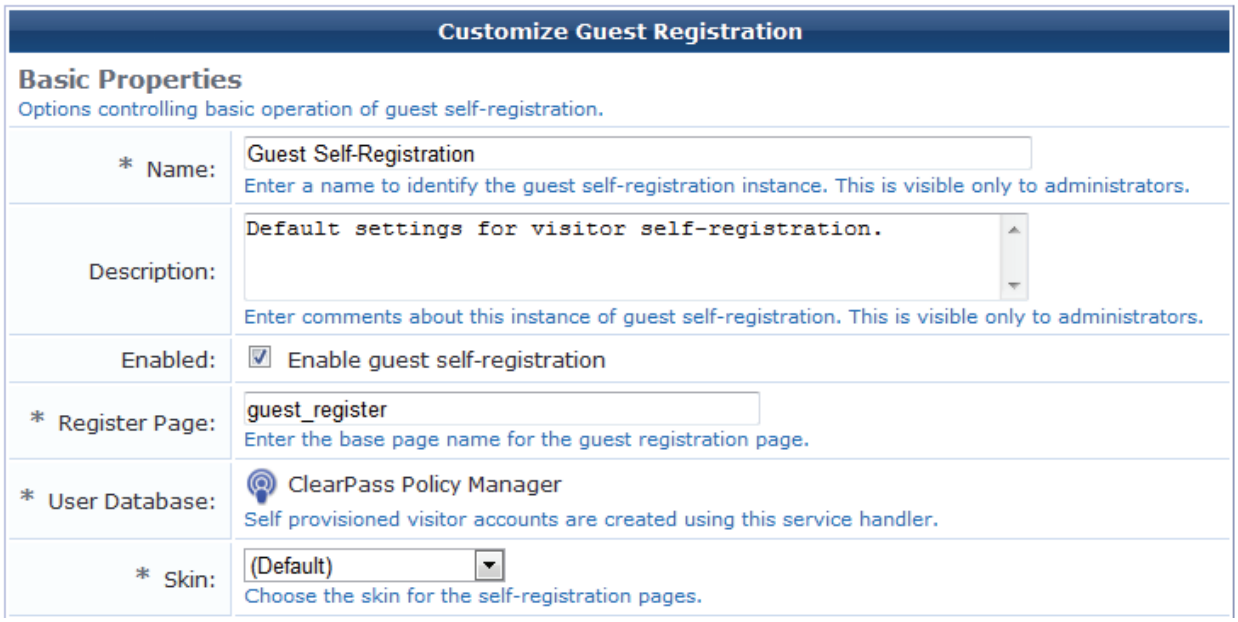
Figure 26 Guest Self-Registration Workflow Diagram



A guest self-registration page consists of many different settings, which are divided into groups across several pages. Click an icon or label in the diagram to jump directly to the editor for that item.

Configuring Basic Properties for Self-Registration

Click the **Master Enable**, **User Database**, **Choose Skin**, or **Rename Page** links to edit the basic settings for guest self-registration.



The Basic Properties window has configurable settings such as Name, Description, enabling guest-self registration, Register Page, Parent, and Authentication.

Using a Parent Page

To use the settings from a previously configured self-registration page, select an existing page name from the **Parent** drop-down menu. This is useful if you need to configure multiple registrations. You can always override parent page values by editing field values yourself. To create a self-registration page with new values, select the **Guest Self-Registration (guest_register)** option from the **Parent** field drop-down menu.

Paying for Access

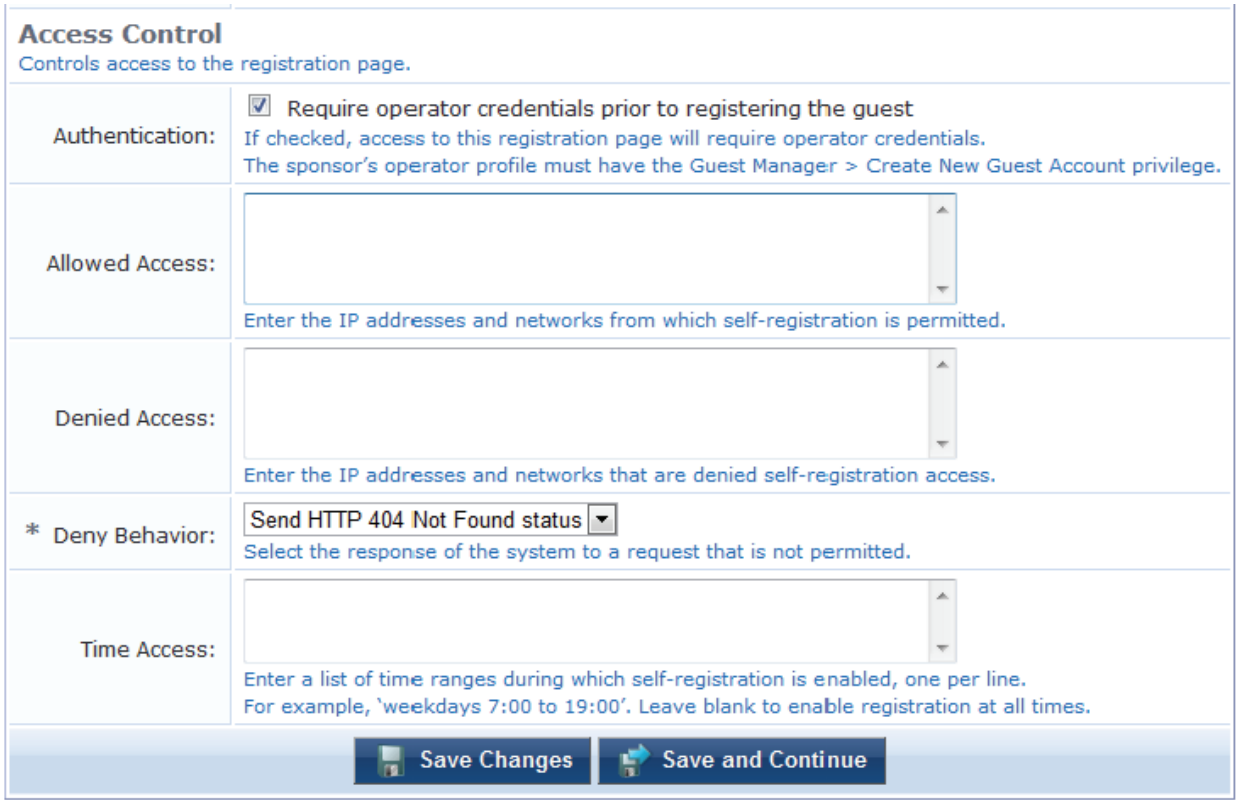
If you select a standalone self -registration, **(No parent- standalone)** option you can also configure the Hotspot option. You can configure this setting so that registrants have to pay for access.

Requiring Operator Credentials

If you want to require an operator to log in with their credentials before they can create a new guest account, select the **Require operator credentials prior to registering guest** check box. The sponsor's operator profile must have the **Guest Manager > Create New Guest Account privilege** already configured.

If you choose this option, the authenticated page it produces for creating accounts is very simple, and does not include navigation or other links that would otherwise be available in the operator user interface.

You can specify access restrictions for the self-registration page in the **Access Control** section of this form.



The **Allowed Access** and **Denied Access** fields are access control lists that determine if a client is permitted to access this guest self-registration page. You can specify multiple IP addresses and networks, one per line, using the following syntax:

- **1.2.3.4** – IP address
- **1.2.3.4/24** – IP address with network prefix length
- **1.2.3.4/255.255.255.0** – IP address with explicit network mask

Use the **Deny Behavior** drop-down list to specify the action to take when access is denied. The **Time Access** field allows you to specify the days and times that self-registration is enabled. Times must be entered in 24-hour clock format. For example:

- Mondays, Wednesdays and Fridays, 8:00 to 17:00
- Weekdays, 6:00 to 18:00
- Weekends 10:00 to 22:00 and Thursday 11:00 to 13:00

The access control rules will be applied in order, from the most specific match to the least specific match.

Access control entries are more specific when they match fewer IP addresses. The most specific entry is a single IP address (for example, **1.2.3.4**), while the least specific entry is the match-all address of **0.0.0.0/0**.

As another example, the network address **192.168.2.0/24** is less specific than a smaller network such as **192.168.2.192/26**, which in turn is less specific than the IP address **192.168.2.201** (which may also be written as **192.168.2.201/32**).

To determine the result of the access control list, the most specific rule that matches the client's IP address is used. If the matching rule is in the **Denied Access** field, then the client will be denied access. If the matching rule is in the **Allowed Access** field, then the client will be permitted access.

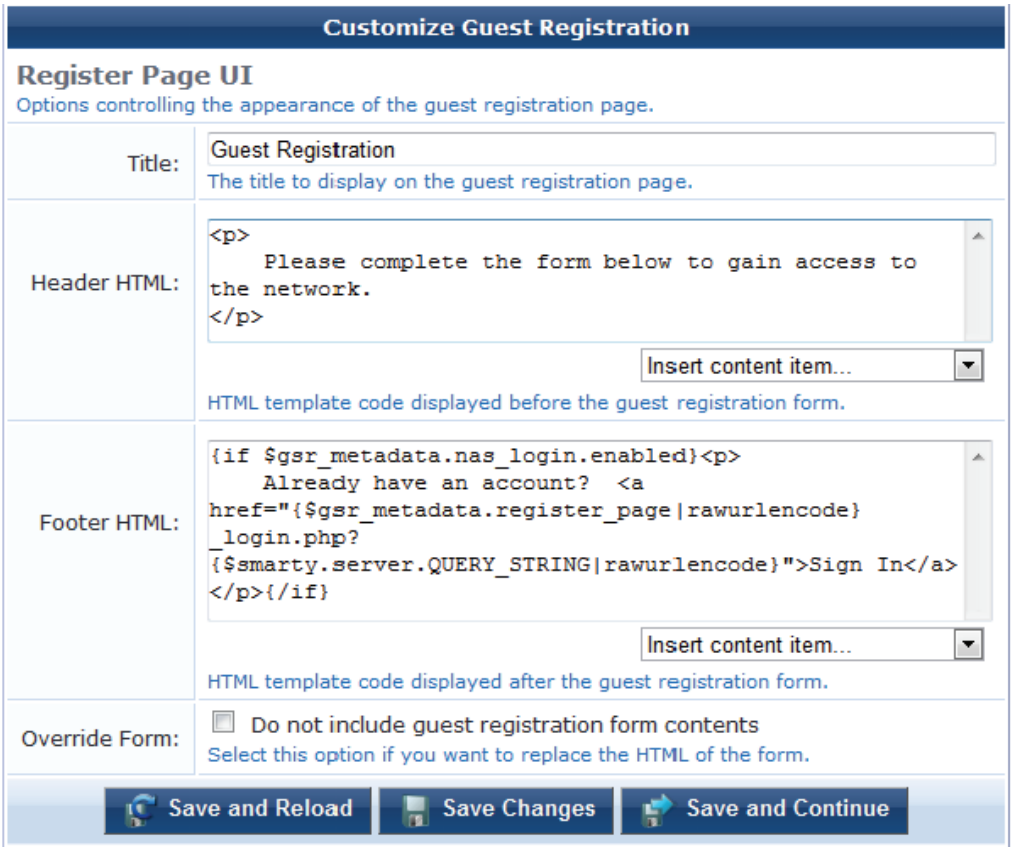
If the **Allowed Access** field is empty, all access will be allowed, except to clients with an IP address that matches any of the entries in the **Denied Access** field. This behavior is equivalent to adding the entry **0.0.0.0/0** to the **Allowed Access** field.

If the **Denied Access** list is empty, only clients with an IP address that matches one of the entries in the Allowed Access list will be allowed access. This behavior is equivalent to adding the entry **0.0.0.0/0** to the Denied Access list.

Editing Registration Page Properties

To edit the properties of the registration page:

1. Navigate to **Configuration > Guest Self-Registration**
2. Select an entry in the **Guest Self-Registration** list and click its **Edit** link. The **Customize Guest Registration** workflow page appears, as shown in Figure 26.
3. Click the **Register Page** link, or one of the **Title**, **Header,** or **Footer** fields for the Register Page.



Template code for the title, header, and footer may be specified. See "Smarty Template Syntax" in the "Reference" chapter for details on the template code that may be inserted.

Select the **Do not include guest registration form contents** check box to override the normal behavior of the registration page, which is to display the registration form between the header and footer templates.

Click the **Save and Reload** button to update the self-registration page and launch or refresh a second browser window to show the effects of the changes.

Click the **Save Changes** button to return to the process diagram for self-registration.

Click the **Save and Continue** button to update the self-registration page and continue to the next editor.

Editing the Default Self-Registration Form Settings

Click the **Form** link for the Register Page to edit the fields on the self-registration form.

The default settings for this form are as follows:

- The **visitor_name** and **email** fields are enabled. The email address of the visitor will become their username for the network.
- The **expire_after** field is hidden, and set to a value of 24 by default; this sets the default expiration time for a self-registered visitor account to be 1 day after it was created.
- The **role_id** field is hidden, and set to a value of 2 by default; this sets the default role for a self-registered visitor account to the built-in Guest role.
- The **auto_update_account** field is set by default. This is to ensure that a visitor who registers again with the same email address has their existing account automatically updated.

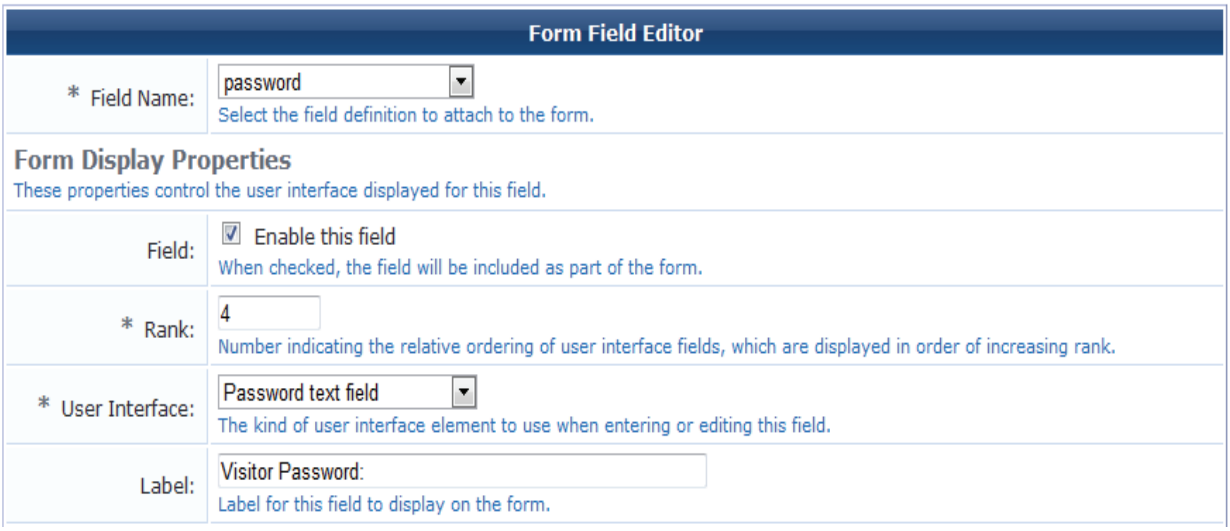
Creating a Single Password for Multiple Accounts

You can create multiple accounts that have the same password. In order to do this, you first customize the Create Multiple Guest Accounts form to include the Password field.

To include the Password field on the Create Multiple Guest Accounts form:

1. Go to **Configuration > Forms & Views**. Click the **create_multi** row, then click its **Edit Fields** link. The Customize Form Fields view opens, showing a list of the fields included in the Create Multiple Guest Accounts form and their descriptions.

   At this point, the Password field is not listed because the Create Multiple Guest Accounts form (create_multi) has not yet been customized to include it. You will create it for the form in the next step.
2. Click on any field in the list to expand a row, then click the **Insert After** link (you can modify this placement later). The Customize Form Field form opens.
3. In the **Field Name** row, choose **password** from the drop-down list. The form displays configuration options for this field.



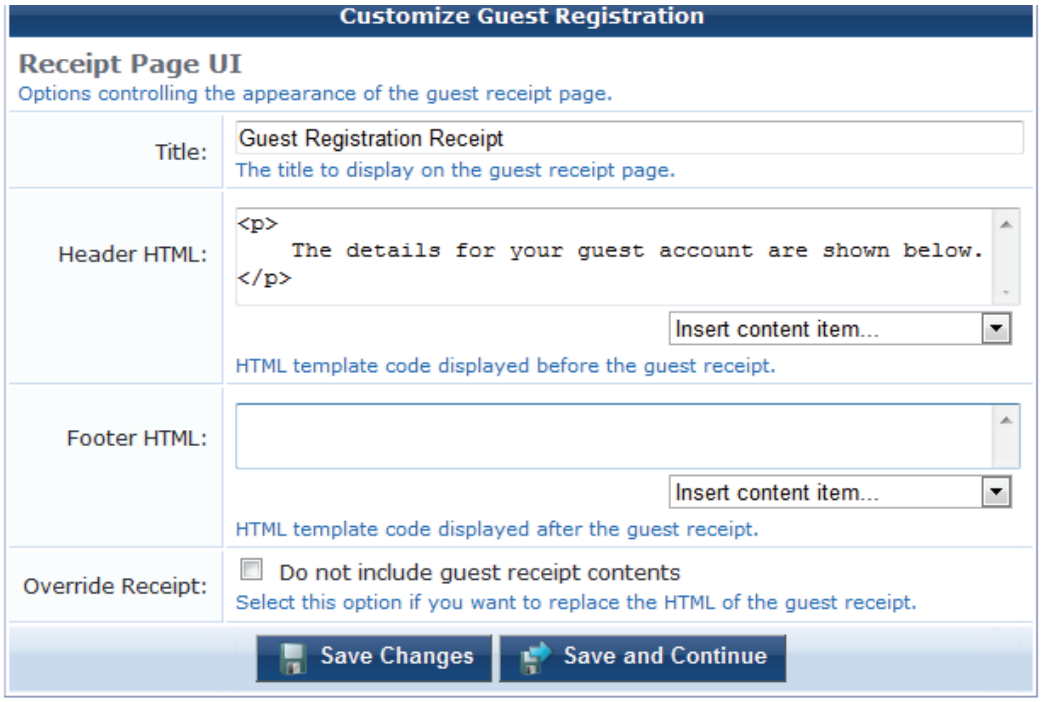
4. In the **Field** row, mark the **Enable this field** check box.
5. To adjust the placement of the password field on the Create Multiple Guest Accounts form, you may change the number in the **Rank** field.
6. In the **User Interface** row, choose **Password text field** from the drop-down list. The **Field Required** check box should now be automatically marked, and the **Validator** field should be set to **IsNonEmpty**.
7. Click **Save Changes**. The Customize Form Fields view opens again, and the password field is now included and can be edited.

To create the multiple accounts that all use the same password, see "Creating a Single Password for Multiple Accounts" in the "Guest Manager" chapter.

Editing Guest Receipt Page Properties

To edit the properties of the guest receipt page:

1. Navigate to **Configuration > Guest Self-Registration**
2. Select an entry in the **Guest Self-Registration** list and click its **Edit** link. The **Customize Guest Registration** workflow page appears, as shown in Figure 26.
3. Click the **Receipt Page** link or one of the **Title**, **Header,** or **Footer** fields for the Receipt Page to edit the properties of the receipt page. This page is shown to guests after their visitor account has been created.

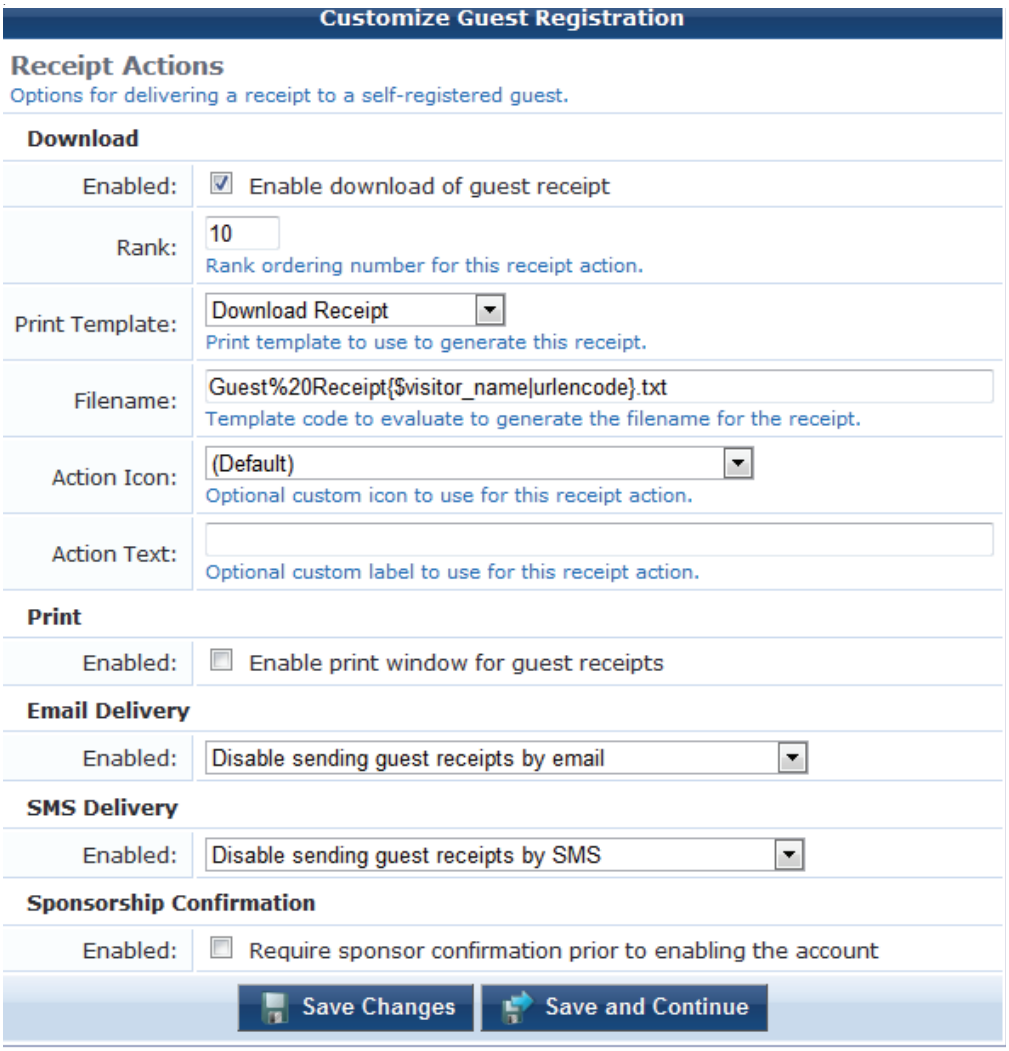


Click the **Save Changes** button to return to the process diagram for self-registration.

Editing Receipt Actions

To edit the actions that are available once a visitor account has been created:

1. Navigate to **Configuration > Guest Self-Registration.**
2. Select an entry in the **Guest Self-Registration** list and click its **Edit** link. The **Customize Guest Registration** workflow page appears, as shown in Figure 26.
3. In the **Receipt Page** area of the diagram, click the **Actions** link. The Receipt Actions form opens.



Enabling Sponsor Confirmation for Role Selection

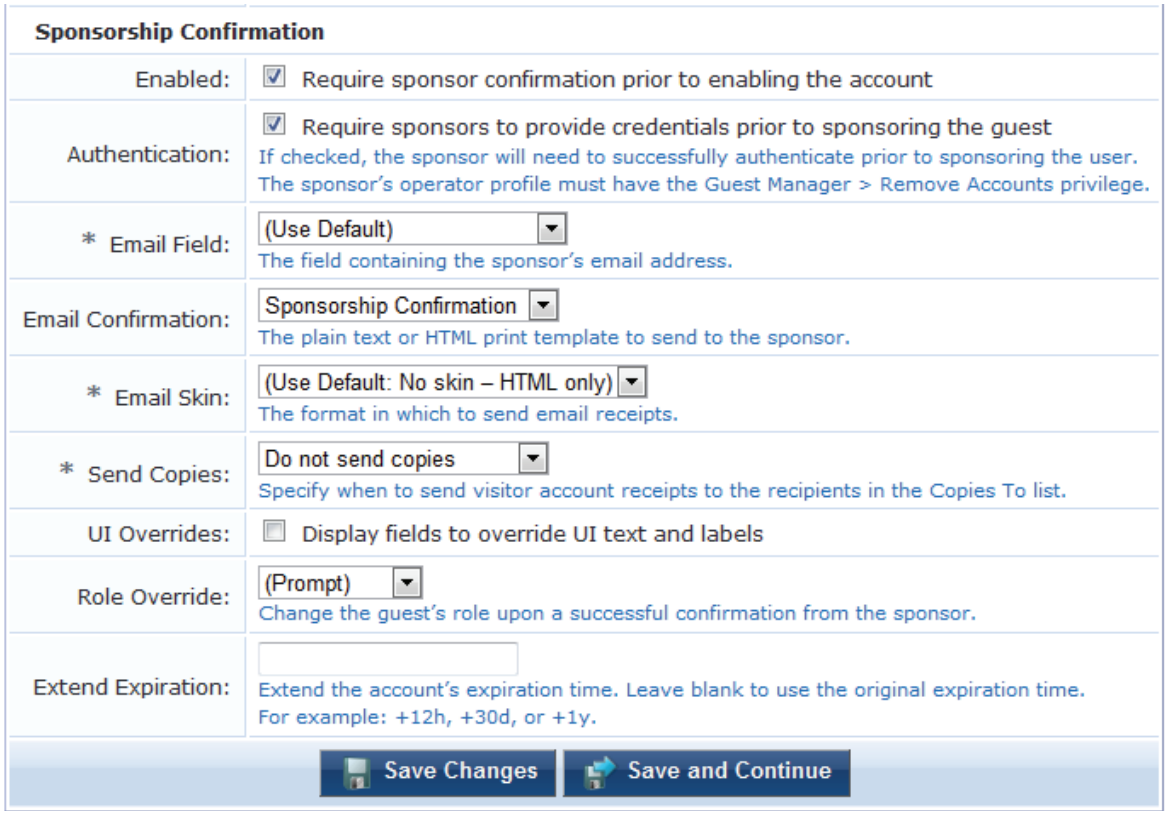
You can allow the sponsor to choose the role for the user account at the time the sponsor approves the self-registered account.

To enable role selection by the sponsor:

1. Go to **Configuration > Guest Self-Registration**. Click the **Guest Self-Registration** row, then click its **Edit** link. The Customize Guest Registration diagram opens.
2. In the **Receipt Page** area of the diagram, click the **Actions** link.

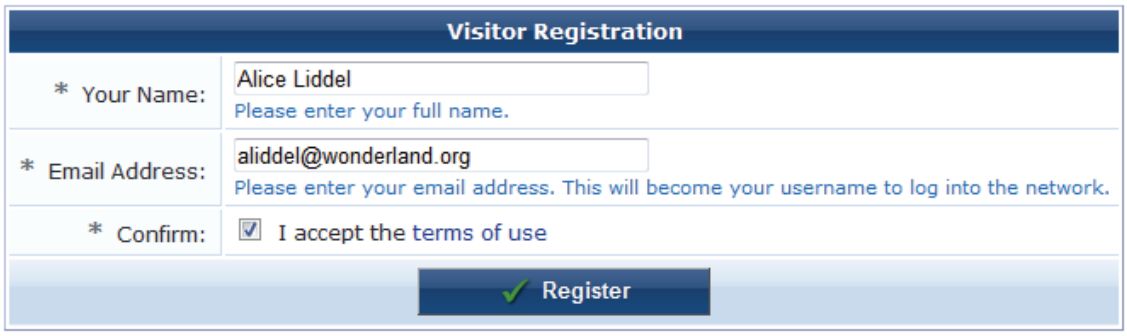
The Receipt Actions form opens.

3. In the **Sponsorship Confirmation** area at the bottom of the form, mark the **Enabled** check box for **Require sponsor confirmation prior to enabling the account**. The form expands to let you configure this option.



4. In the **Authentication** row, mark the check box for **Require sponsors to provide credentials prior to sponsoring the guest**.
5. In the **Role Override** row, choose **(Prompt)** from the drop-down list.
6. Complete the rest of the form with the appropriate information, then click **Save Changes**. The Customize Guest Registration diagram opens again.
7. You can click the **Launch this guest registration page** link at the upper-right corner of the Customize Guest Registration diagram to preview the Guest Registration login page.

The Guest Registration login page is displayed as the guest would see it.



When a guest completes the form and clicks the **Register** button, the sponsor receives an email notification.

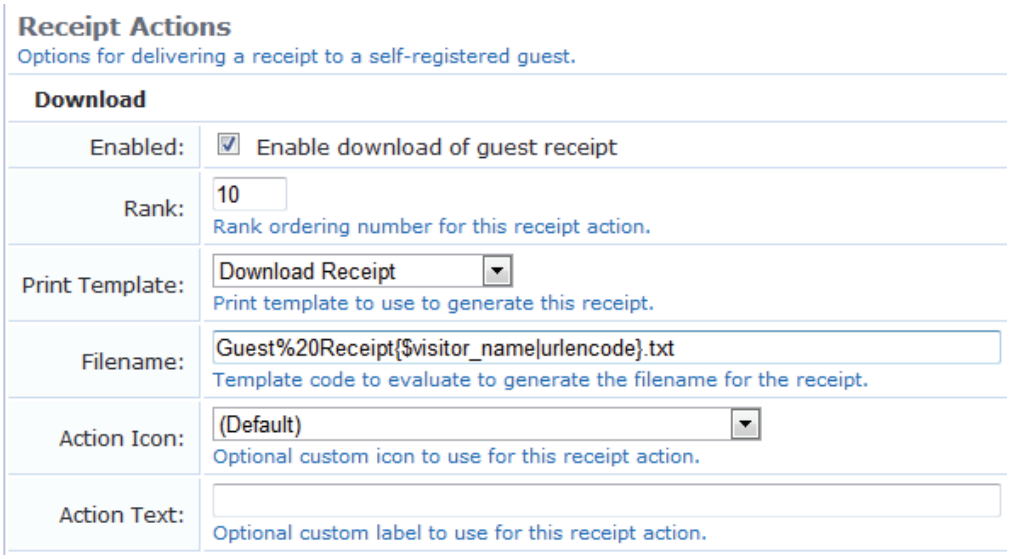
8. To confirm the guest's access, the sponsor clicks the **click here** link in the email, and is redirected to the Guest Registration Confirmation form.

9. In the **Account Role** drop-down list, the sponsor chooses the role for the guest, then clicks the **Confirm** button.

Editing Download and Print Actions for Guest Receipt Delivery

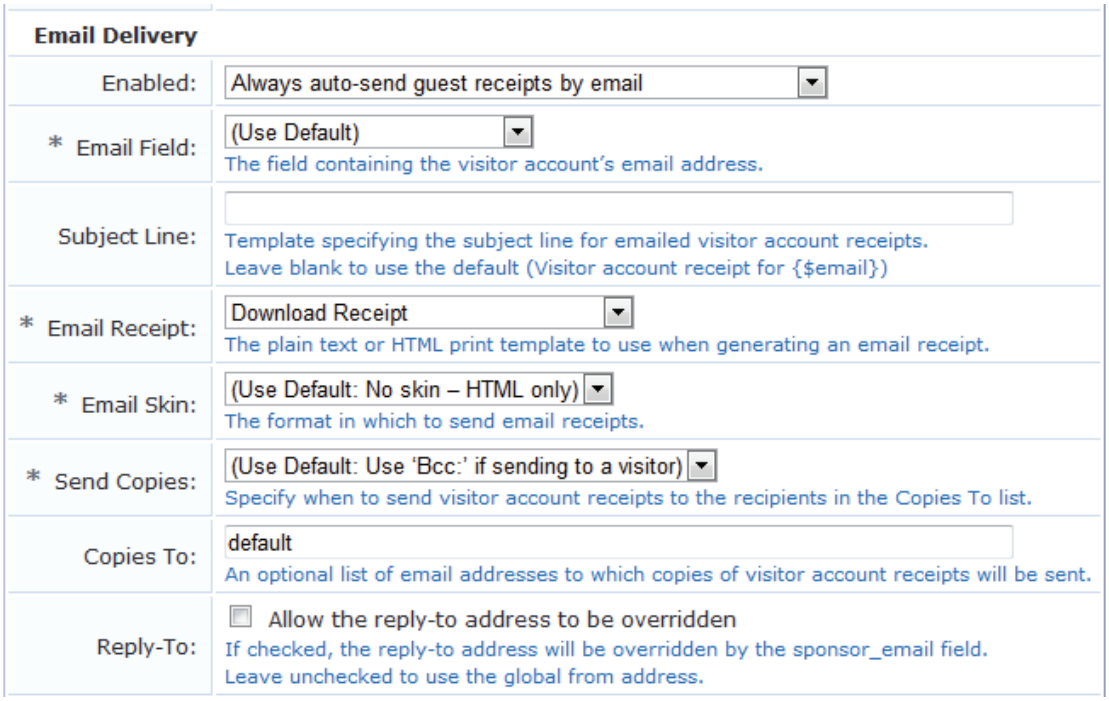
To enable the template and display options to deliver a receipt to the user as a downloadable file, or display the receipt in a printable window in the visitor's browser:

1. Go to **Configuration > Guest Self-Registration**. Click the **Guest Self-Registration** row, then click its **Edit** link. The Customize Guest Registration diagram opens.
2. In the **Receipt Page** area of the diagram, click the **Actions** link. The Receipt Actions form opens.
3. Select either the **Enable download of guest receipt** check box in the **Download** area, or the **Enable print window for guest receipts** check box in the **Print** area. The form expands to include configuration options.



Editing Email Delivery of Guest Receipts

The Email Delivery options available for the receipt page actions allow you to specify the email subject line, the print template and email format, and other fields relevant to email delivery.

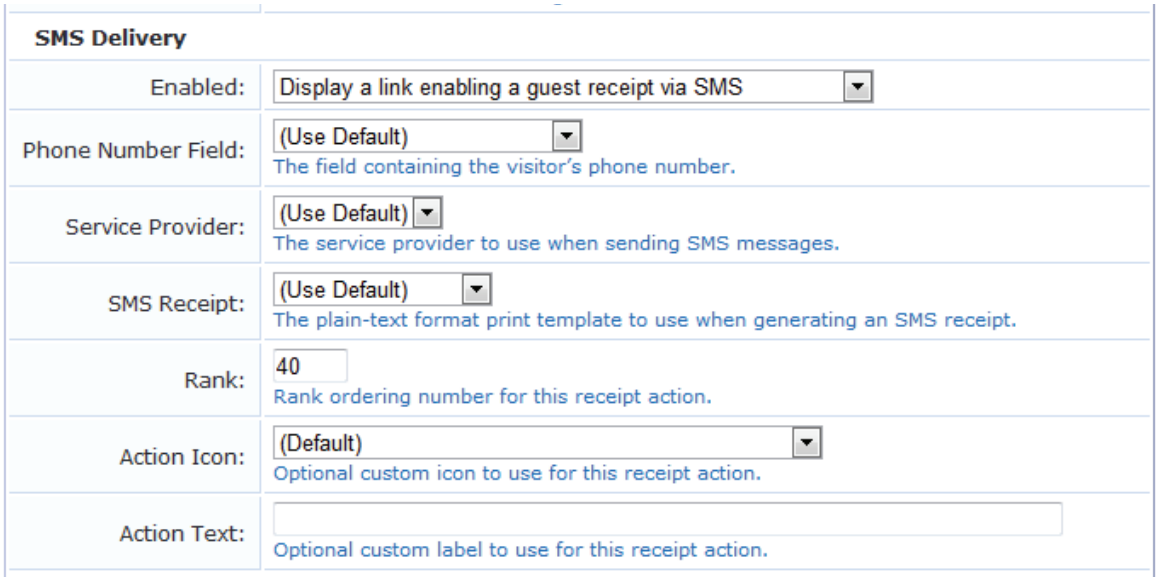


When email delivery is enabled, the following options are available to control email delivery:

- **Disable sending guest receipts by email** – Email receipts are never sent for a guest registration.
- **Always auto-send guest receipts by email** – An email receipt is always generated using the selected options, and will be sent to the visitor's email address.
- **Auto-send guest receipts by email with a special field set** – If the Auto-Send Field available for this delivery option is set to a non-empty string or a non-zero value, an email receipt will be generated and sent to the visitor's email address. The auto-send field can be used to create an "opt-in" facility for guests. Use a check box for the **auto_send_smtp** field and add it to the **create_user** form, or a guest self-registration instance, and email receipts will be sent to the visitor only if the check box has been selected.
- **Display a link enabling a guest receipt via email** – A link is displayed on the receipt page; if the visitor clicks this link, an email receipt will be generated and sent to the visitor's email address.
- **Send an email to a list of fixed addresses** – An email receipt is always generated using the selected options, and will be sent only to the list of email addresses specified in "Copies To".

Editing SMS Delivery of Guest Receipts

The SMS Delivery options available for the receipt page actions allow you to specify the print template to use, the field containing the visitor's phone number, and the name of an auto-send field.



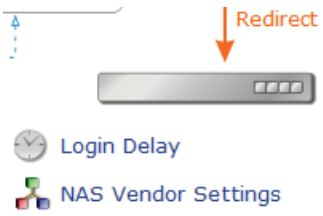
These options under Enabled are available to control delivery of SMS receipts:

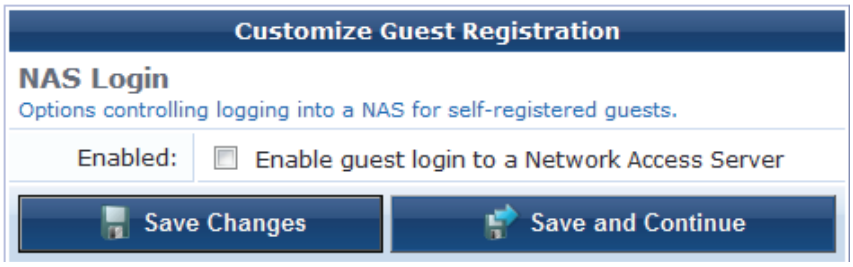
- **Disable sending guest receipts by SMS** – SMS receipts are never sent for a guest registration.
- **Always auto-send guest receipts by SMS** – An SMS receipt is always generated using the selected options, and will be sent to the visitor's phone number.
- **Auto-send guest receipts by SMS with a special field set** – If the Auto-Send Field is set to a non-empty string or a non-zero value, an SMS receipt will be generated and sent to the visitor's phone number. The **auto-send** field can be used to create an "opt-in" facility for guests. Use a check box for the **auto_send_sms** field and add it to the **create_user** form, or a guest self-registration instance, and SMS messages will be sent to the specified phone number only if the check box has been selected.
- **Display a link enabling a guest receipt via SMS** – A link is displayed on the receipt page; if the visitor clicks this link, an SMS receipt will be generated and sent to the visitor's phone number. Only one SMS receipt per guest registration can be sent in this way.

Enabling and Editing NAS Login Properties

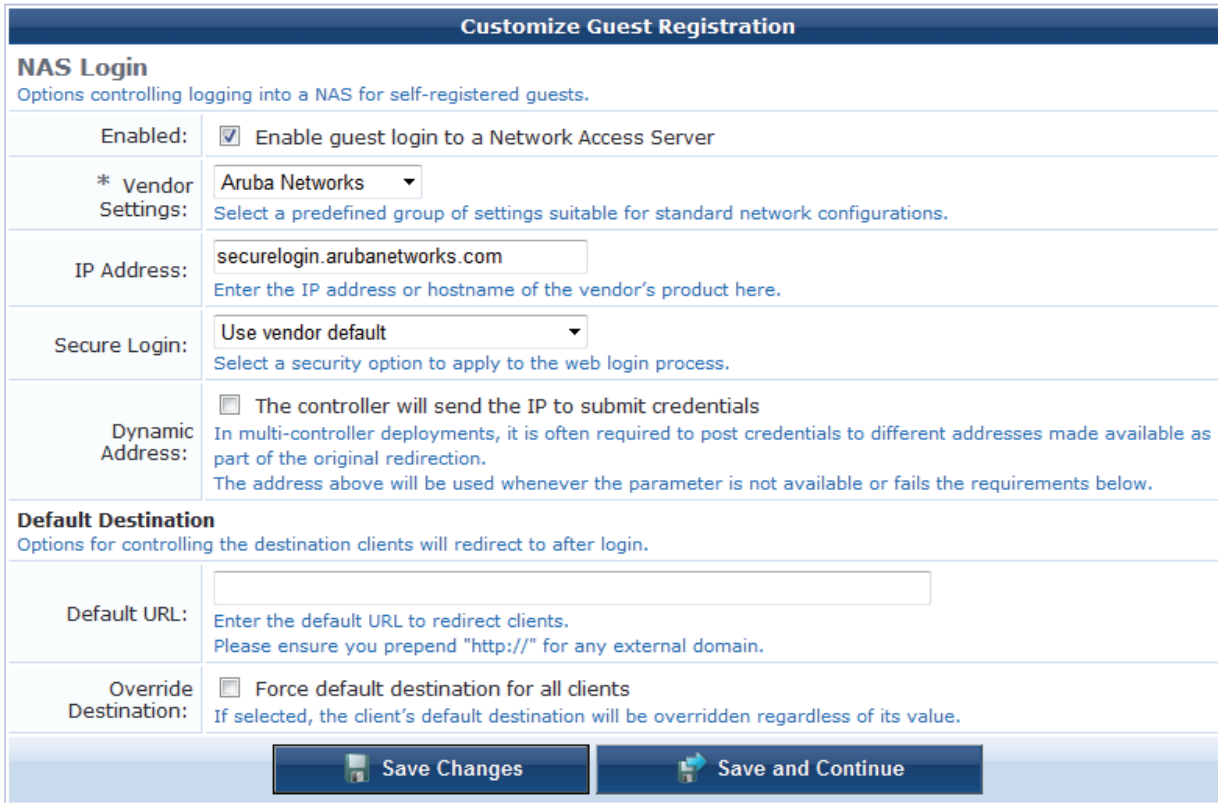
To enable and edit the properties for automatic NAS login:

1. Go to **Configuration > Guest Self-Registration**. Click to expand the **Guest Self-Registration** row in the form, then click its **Edit** link. The Customize Guest Self-Registration diagram opens.
2. In the lower-right corner of the diagram, click the **NAS** box or the **NAS Vendor Settings** link. The NAS Login form opens.





3. Mark the **Enabled** check box to expand the form.



If automatic guest login is not enabled, the submit button on the receipt page will not be displayed, and automatic NAS login will not be performed.

Editing Login Page Properties

The login page is displayed if automatic guest login is enabled and a guest clicks the submit button from the receipt page to log in.

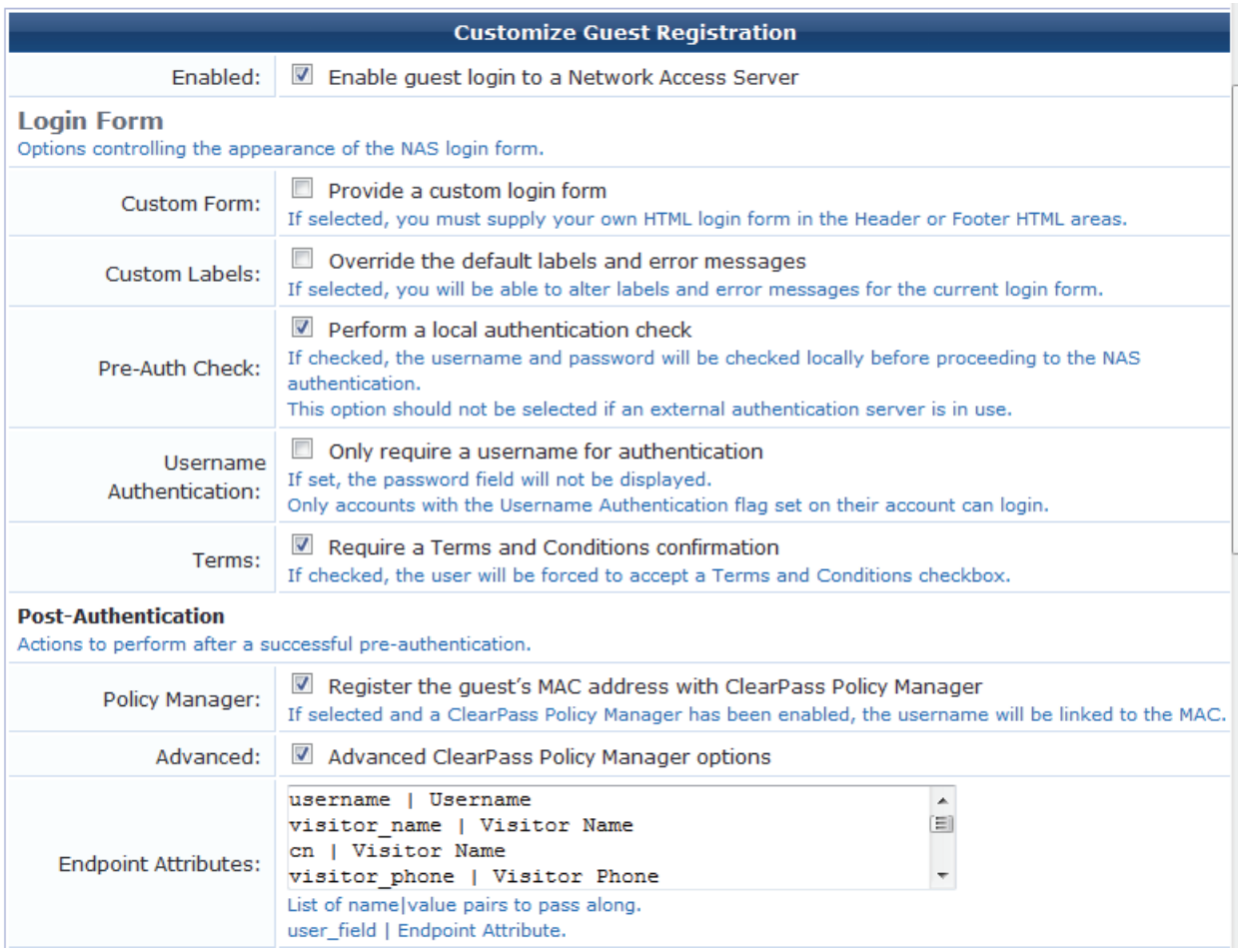
To edit the properties of the login page:

1. Go to **Configuration > Guest Self-Registration**. Click to expand the **Guest Self-Registration** row in the form, then click its **Edit** link. The Customize Guest Self-Registration diagram opens.
2. In the **Receipt Page** area of the diagram, click the **Title** or **Login Message** fields for the login page to edit the properties of the login page, then mark the **Enable guest login to a Network Access Server** check box.The form expands to include configuration options.

The login page is also a separate page that can be accessed by guests using the login page URL. The login page URL has the same base name as the registration page, but with **_login** appended. To determine the login page URL for a guest self-registration page, first ensure that the **Enable guest login to a Network Access Server** option is checked, and then use the **Launch network login** link from the self-registration process diagram, as shown below:

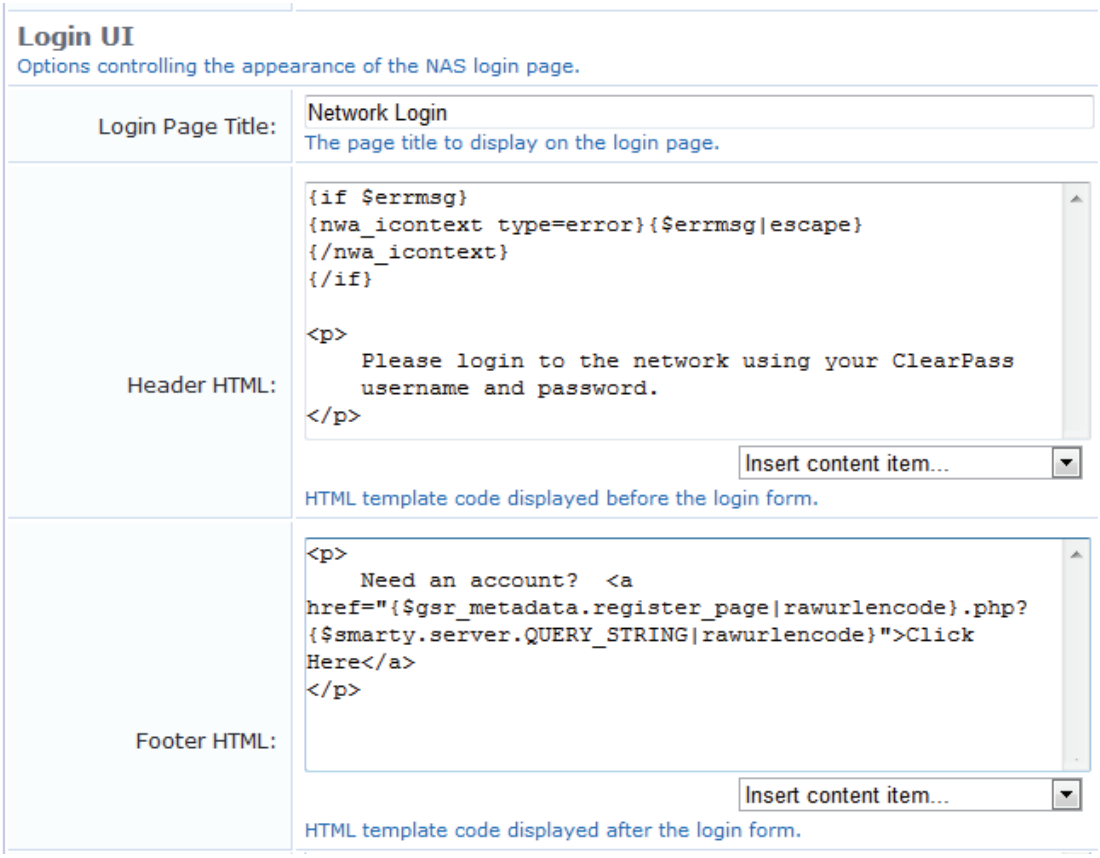
Launch this guest registration page
Launch network login

The options available under the **Login Form** heading may be used to customize the login page.

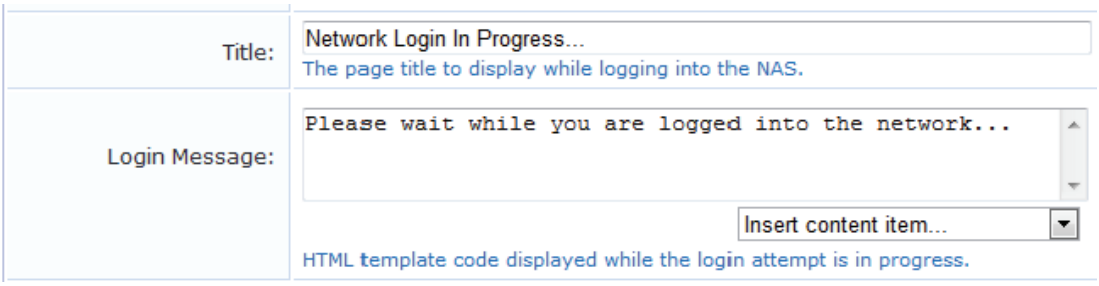


The login page consists of two separate parts: the login form page, and a login message page.

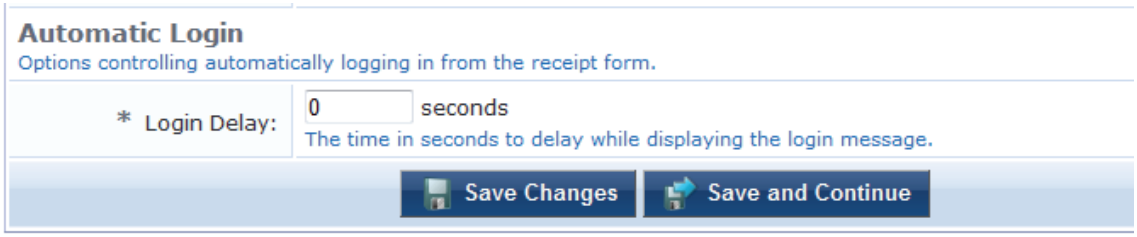
The login form page contains a form prompting for the guest's username and password. The title, header and footer of this page can be customized. If the **Provide a custom login form** option is selected, then the form must also be provided in either the Header HTML or Footer HTML sections.



The login message page is displayed after the login form has been submitted, while the guest is being redirected to the NAS for login. The title and message displayed on this page can be customized.



The login delay can be set; this is the time period, in seconds, for which the login message page is displayed.



Click the **Save Changes** button to return to the process diagram for self-registration.

Self-Service Portal Properties

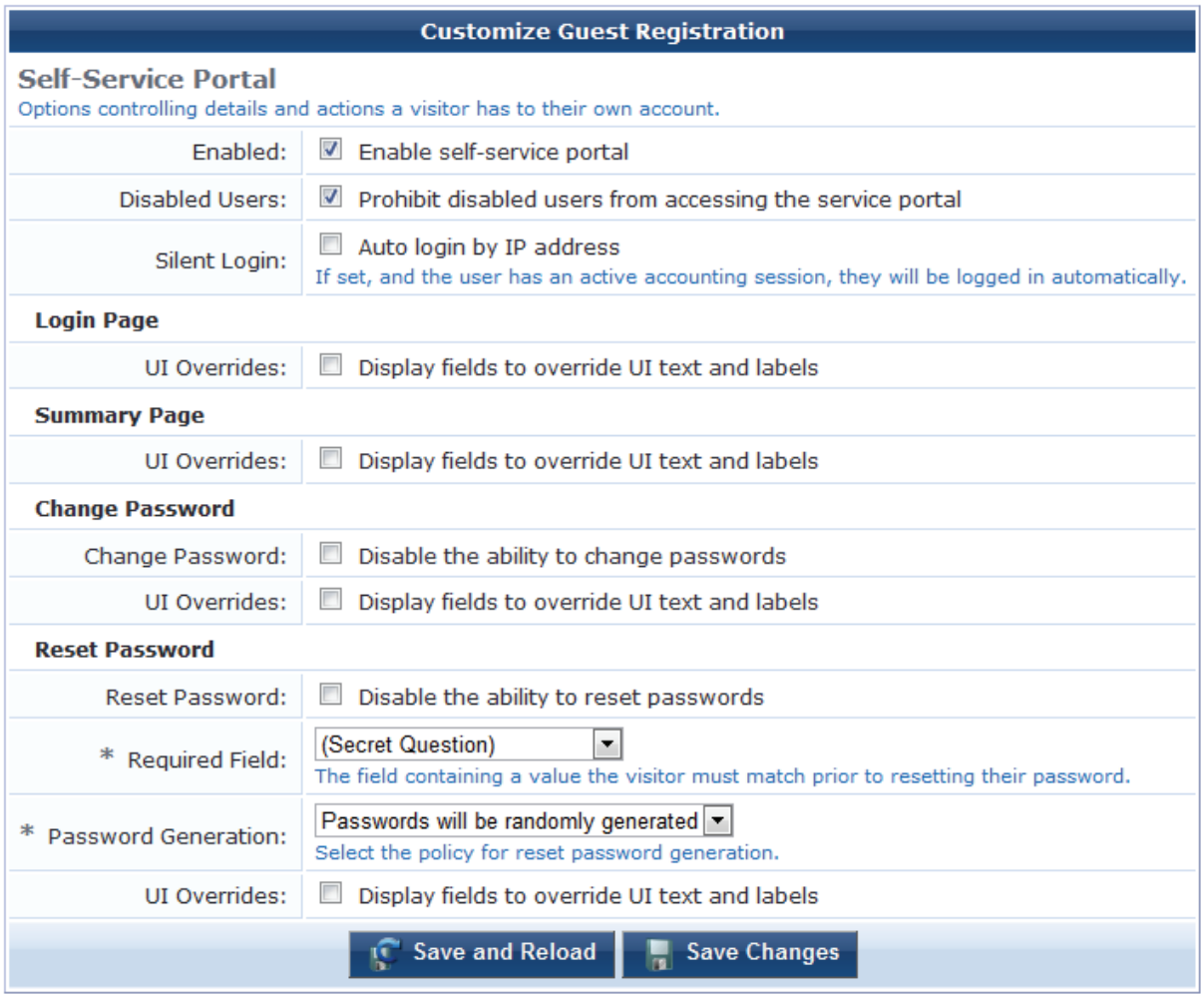
To edit the properties of the self-service portal:

1. Go to **Configuration > Guest Self-Registration**. Click to expand the **Guest Self-Registration** row in the form, then click its **Edit** link. The Customize Guest Self-Registration diagram opens.
2. Click the **Self-Service Portal** link or one of the **Login Page**, **Summary Page**, **Change Password,** or **Reset Password** links for the Self-Service Portal.
3. Mark the **Enable self-service portal** check box.The form expands to include configuration options.

The self-service portal is accessed through a separate link that must be published to guests. The page name for the portal is derived from the registration page name by appending "_portal".

When the self-service portal is enabled, a **Go To Portal** link is displayed on the list of guest self-registration pages, and may be used to determine the URL that guests should use to access the portal.

The portal offers guests the ability to log in with their account details, view their account details, or change their password. Additionally, the Reset Password link provides a method allowing guests to recover a forgotten account password.



To adjust the user interface, use the override check boxes to display additional fields on the form. These fields allow you to customize all text and HTML displayed to users of the self-service portal.

The behavioral properties of the self-service portal are described below:

- The "Enable self-service portal" check box must be selected for guests to be able to access the portal. Access to the portal when it is disabled results in a disabled message being displayed; this message may be customized using the "Disabled Message" field.

- The “Disabled Users” check box controls whether a user account that has been disabled is allowed to log in to the portal.
- The “Change Password” check box controls whether guests are permitted to change their account password using the portal.
- The “Reset Password” check box controls whether guests are permitted to reset a forgotten account password using the portal. If this check box is enabled, the “Required Field” may be used to select a field value that the guest must match in order to confirm the password reset request.

If the “Auto login by IP address” option is selected, a guest accessing the self-service portal will be automatically logged in if their client IP address matches the IP address of an active RADIUS accounting session (that is, the guest’s HTTP client address is the same as the RADIUS Framed-IP-Address attribute for an active session).

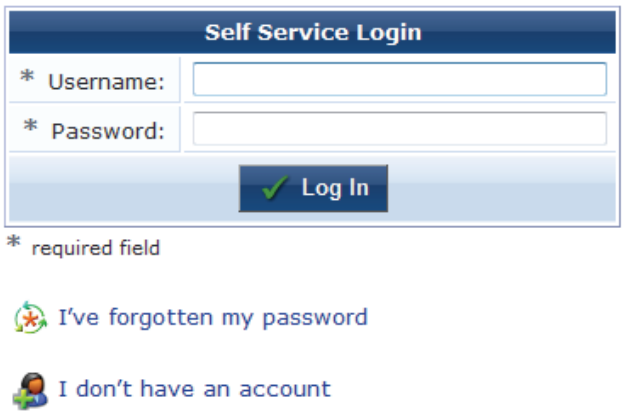
The Password Generation drop-down list controls what kind of password reset method is used in the portal. The default option is “Passwords will be randomly generated”, but the alternative option “Manually enter passwords” may be selected to enable guests to select their own password through the portal.

Click the **Save Changes** button to return to the process diagram for self-registration.

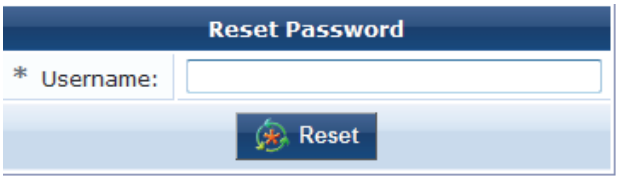
Resetting Passwords with the Self-Service Portal

The self-service portal includes the ability to reset a guest account’s password.

The default user interface for the self-service portal is shown below:



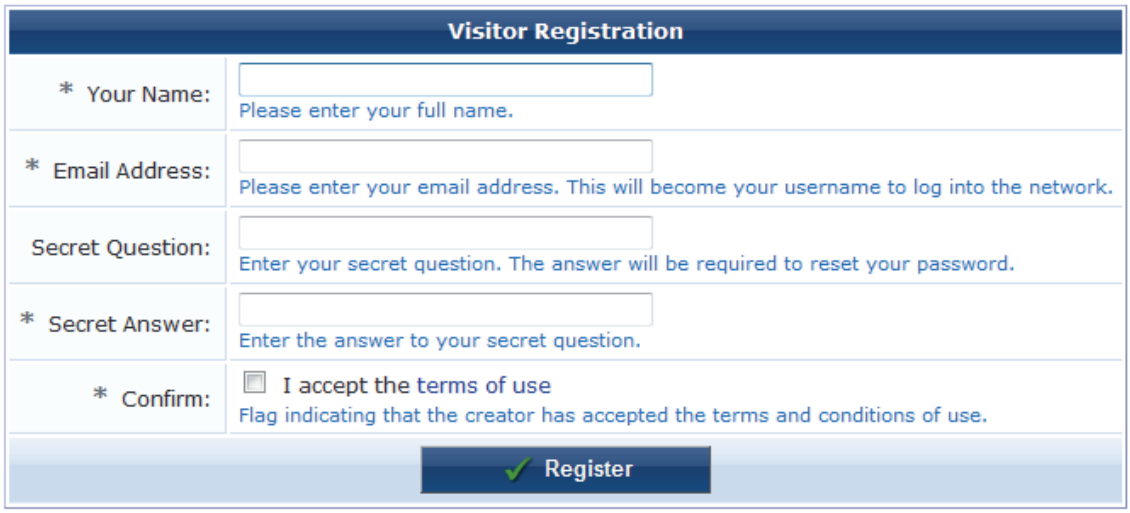
Clicking the **I’ve forgotten my password** link displays a form where the user password may be reset:



Entering a valid username will reset the password for that user account, and will then display the receipt page showing the new password and a login option (if NAS login has been enabled).

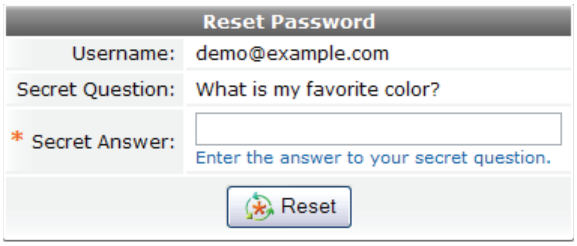
This feature allows the password to be reset for any guest account on the system, which may pose a security risk. It is strongly recommended that when this feature of the self-service portal is enabled, guest registrations should also store a secret question/secret answer field.

To enable a more secure password reset operation, first enable the **secret_question** and **secret_answer** fields to the registration form. The default appearance of these fields is shown below:

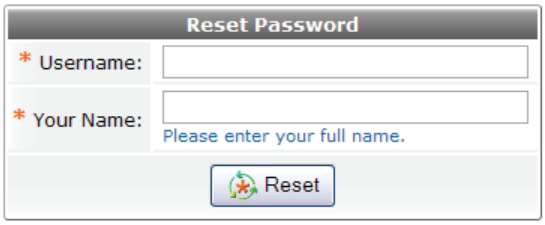


Next, enable the **Required Field** option in the Self-Service Portal properties. Setting this to **(Secret Question)** will ask the guest the **secret_question** and will only permit the password to be reset if the guest supplies the correct **secret_answer** value.

With these settings, the user interface for resetting the password now includes a question and answer prompt after the username has been determined:



Selecting a different value for the “Required Field” allows other fields of the visitor account to be checked. These fields should be part of the registration form. For example, selecting the **visitor_name** field as the “Required Field” results in a **Reset Password** form like this:



Email Receipts and SMTP Services

With SMTP Services, you can configure ClearPass Guest to send customized guest account receipts to visitors and sponsors by email. Email receipts may be sent in plain text or HTML format. You may also send email receipts using any of the installed skins to provide a look and feel.

To use the email sending features, you must have the **SMTP Services Plugin** installed.

About Email Receipts

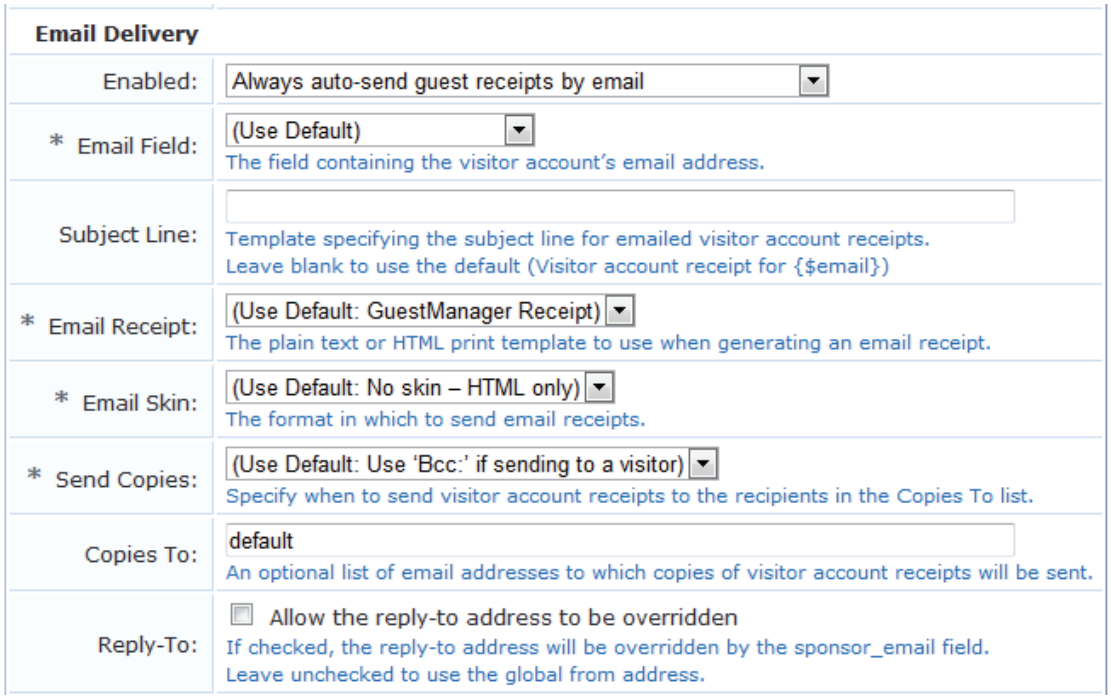
You can send email receipts for guest accounts that are created using either sponsored guest access or self-provisioned guest access. This is convenient in situations where the visitor may not be physically present to receive a printed receipt.

ClearPass Guest may be configured to automatically send email receipts to visitors, or to send receipts only on demand. Email receipts may be sent manually from the guest account receipt page by clicking the **Send email receipt** link displayed there.

When using guest self-registration, the email delivery options available for the receipt page actions allow you to specify the email subject line, the print template and email format, and other fields relevant to email delivery.

To configure these email delivery options:

1. Go to **Configuration > Guest Self-Registration**. Click to expand the **Guest Self-Registration** row in the form, then click its **Edit** link. The Customize Guest Self-Registration diagram opens.
2. In the **Receipt Page** area, click the **Actions** link. The Receipt Actions form opens.
3. Scroll to the **Email Delivery** section of the form and choose one of the options from the **Enabled** drop-down list. The form expands to include configuration options for email delivery.



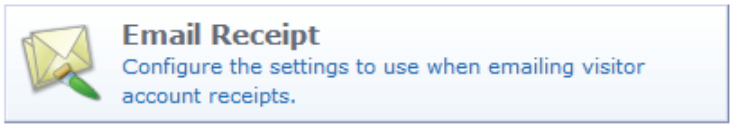
The following options are available in the **Enabled** drop-down list to control email delivery:

- **Disable sending guest receipts by email** – Email receipts are never sent for a guest registration.
- **Always auto-send guest receipts by email** – An email receipt is always generated using the selected options, and will be sent to the visitor's email address.

- **Auto-send guest receipts by email with a special field set** – If the **Auto-Send** Field is set to a non-empty string or a non-zero value, an email receipt will be generated and sent to the visitor's email address. The **auto-send** field can be used to create an "opt-in" facility for guests. Use a check box for the **auto_send_sms** field and add it to the **create_user** form, or a guest self-registration instance, and SMS messages will be sent to the specified phone number only if the check box has been selected.
- **Display a link enabling a guest receipt via email** – A link is displayed on the receipt page; if the visitor clicks this link, an email receipt will be generated and sent to the visitor's email address.
- **Send an email to a list of fixed addresses** – An email receipt is always generated using the selected options, and will be sent only to the list of email addresses specified in the "Copies To" field.

Configuring Email Receipts

You can configure the default settings used when generating an email receipt by going to **Configuration > Email Receipt**.



See **"Email Receipt Options"** for details about the email receipt options.

Email Receipt Options

The **Customize Email Receipt** form may be used to set default options for visitor account email receipts. To configure email receipt options, go to **Configuration > Email Receipt**. The Customize Email Receipt form opens.

Figure 27 *Customize Email Receipt page*

1. The **Subject Line** may contain template code, including references to guest account fields. The default value, `Visitor account receipt for {$email}`, uses the value of the **email** field. See "Smarty Template Syntax" in the "Reference" chapter for more information on template syntax.
2. The **Skin** drop-down list allows you to specify a skin to be used to provide the basic appearance of the email. You may select from one of the installed skins, or use one of these special options:
   - **No skin – Plain text only** – A skin is not used, and the email will be sent in plain text format. Use this option to remove all formatting from the email.
   - **No skin – HTML only** – A skin is not used, but the email will be sent in HTML format. Use this option to provide a basic level of formatting in the email.

- **No skin – Native receipt format** – A skin is not used. The email will be sent in either plain text or HTML format, depending on the type of print template that was selected.
- **Use the default skin** – The skin currently marked as the default skin is used.

When sending an email message using HTML formatting, the images and other resources required to display the page will be included in the message.

3. Use the **Copies To** field to create a list of additional email addresses that are designated to receive copies of the generated email receipts.
4. Choose a value from the **Send Copies** drop-down list to specify how copies of the email receipts will be sent to the additional email addresses listed in the Copies To field:
   - **Do not send copies** – The Copies To list is ignored and email is not copied.
   - **Always send using 'cc:'** – The Copies To list is always sent a copy of any guest account receipt (even if no guest account email address is available).
   - **Always send using 'bcc:'** – The Copies To list is always sent a blind copy of any guest account receipt (even if no guest account email address is available).
   - **Use 'cc:' if sending to a visitor** – If a guest account email address is available, the email addresses in the Copies To list will be copied.
   - **Use 'bcc:' if sending to a visitor** – If a guest account email address is available, the email addresses in the Copies To list will be blind copied.
5. To preview and verify the appearance of the email receipt, you can send yourself or another person a test message. In the **Test Mail Settings** area, enter the test message recipient's email address, then click **Send Test Message**. The test message is sent immediately.

Figure 28 *Example of Email Receipt Test Message Content*

6. When all fields on the form are completed, click **Save and Close**.

About Customizing SMTP Email Receipt Fields

The behavior of email receipt operations can be customized with certain guest account fields. You do this on a per-user basis.

- **smtp_enabled** – This field may be set to a non-zero value to enable sending an email receipt. If unset, the default value from the email receipt configuration is used. The special values "_Auto" (Always auto-send guest receipts by email), "_AutoField" (Auto-send guest receipts by email with a special field set), "_Click" (Display a link enabling a guest receipt via email), and "_Cc" (Send an email to a list of fixed addresses) may also be used.
- **smtp_subject** – This field specifies the subject line for the email message. Template variables appearing in the value will be expanded. If the value is "default", the default subject line from the email receipt configuration is used.
- **smtp_template_id** – This field specifies the print template ID to use for the email receipt. If blank or unset, the default value from the email receipt configuration is used.
- **smtp_receipt_format** – This field specifies the email format to use for the receipt. It may be one of "plaintext" (No skin – plain text only), "html_embedded" (No skin – HTML only), "receipt" (No skin – Native receipt format), "default" (Use the default skin), or the plugin ID of a skin plugin to specify that skin. If blank or unset, the default value from the email receipt configuration is used.
- **smtp_email_field** – This field specifies the name of the field that contains the visitor's email address. If blank or unset, the default value from the email receipt configuration is used. Additionally, the special value "_None" indicates that the visitor should not be sent any email.
- **smtp_auto_send_field** – This field specifies the name of the field that contains the auto-send flag. If blank or unset, the default value from the email receipt configuration is used. Additionally, the special values "_Disabled" and "_Enabled" may be used to never send email or always send email, respectively.
- **smtp_cc_list** – This field specifies a list of additional email addresses that will receive a copy of the visitor account receipt. If the value is "default", the default carbon-copy list from the email receipt configuration is used.
- **smtp_cc_action** – This field specifies how to send copies of email receipts. It may be one of "never", "always_cc", "always_bcc", "conditional_cc", or "conditional_bcc". If blank or unset, the default value from the email receipt configuration is used.

The logic used to send an email receipt is:

- If email receipts are disabled, take no action.
- Otherwise, check the auto-send field.
  - If it is "_Disabled" then no receipt is sent.
  - If it is "_Enabled" then continue processing.
  - If it is any other value, assume the **auto-send** field is the name of another guest account field. Check the value of that field, and if it is zero or the empty string then no receipt is sent.
- Determine the email recipients:
  - Address the email to the value specified by the **email** field in the visitor account. If the **email** field is "_None" then do not send an email directly to the visitor.
  - Depending on the value of the Send Copies setting, add the email addresses from the Copies To: list to the email's "Cc:" or "Bcc:" list.
- If there are any "To:", "Cc:" or "Bcc:" recipients, generate an email message using the specified print template and send it to the specified recipient list.
- **smtp_warn_before_subject** – This field overrides what is specified in the subject line under Logout Warnings on the email receipt. If the value is "default", the default subject line under the Logout Warnings section on the email receipt configuration is used.

- **smtp_warn_before_template_id** – This field overrides the print template ID specified under Logout Warnings on the email receipt. If the value is “default”, the default template ID under the Logout Warnings section on the email receipt configuration is used.
- **smtp_warn_before_receipt_format** – This field overrides the email format under Logout Warnings to use for the receipt. It may be one of “plaintext” (No skin – plain text only), “html_embedded” (No skin – HTML only), “receipt” (No skin – Native receipt format), “default” (Use the default skin), or the plugin ID of a skin plugin to specify that skin. If blank or unset, the default value in the Email Field under the Logout Warnings on the email receipt configuration is used.
- **smtp_warn_before_cc_list** – This overrides the list of additional email addresses that receive a copy of the visitor account receipt under Logout Warnings on the email receipt.If the value is “default”, the default carbon-copy list under Logout Warnings from the email receipt configuration is used.
- **smtp_warn_before_cc_action** – This field overrides how copies are sent as indicated under Logout Warnings on the email receipt. to send copies of email receipts. It may be one of “never”, “always_cc”, “always_bcc”, “conditional_cc”, or “conditional_bcc”. If blank or unset, the default value from the email receipt configuration is used.
- **warn_before_from_sponsor** – This field overrides the **Reply To** field (that is, the sponsor_email field of a user, or the admin's email) under the Logout Warnings on the email receipt. If the value is “default”, the **Reply To** field under Logout Warnings from the email receipt configuration is used.
- **warn_before_from** – This field overrides the **Override From** field under the Logout Warnings on the email receipt. If the value is “default”, the **Override From** field under Logout Warnings from the email receipt configuration is used.

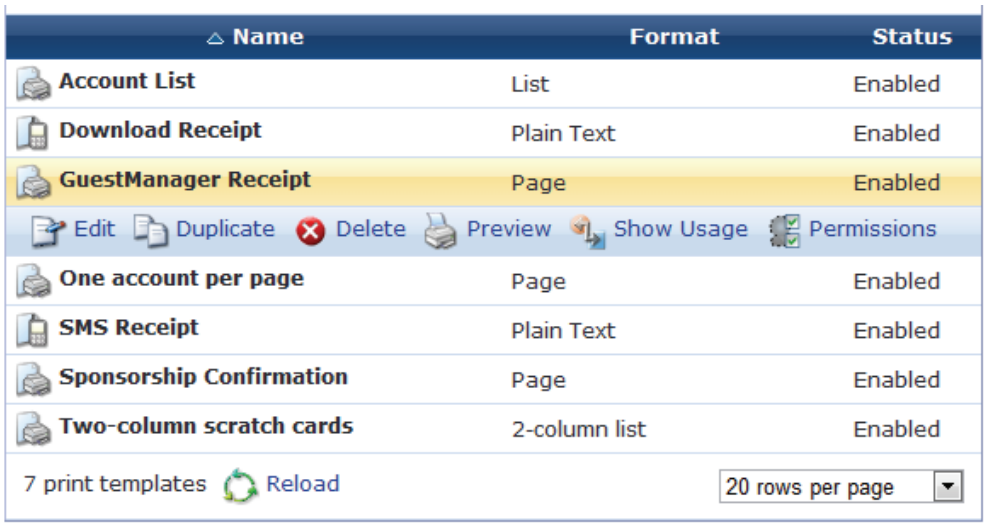
Customizing Print Templates

Print templates are used to define the format and appearance of a guest account receipt. To work with print templates, go to **Configuration > Print Templates**. The Print Templates view opens.

Click a print template’s row in the list to select it. The template’s row expands to include the **Edit**, **Duplicate**, **Delete**, **Preview**, **Show Usage**, and **Permissions** options.

The **Edit code** action is displayed for a print template when it has been created using the wizard, but subsequently modified. See **“Modifying Wizard-Generated Templates”** in this chapter for further information.

Options to show where a print template is being used, and to control individual permissions for a print template, are also available when selecting a print template. See “Setting Print Template Permissions”.



Plain text print templates may be used with SMS services to send guest account receipts; see"About SMS Guest Account Receipts" in the "Administration" chapter for details. Because SMS has a 160 character limit, the number of character used in the plain text template will be displayed below the preview. If you are including a guest account's email address in the SMS, remember to allow for lengthy email addresses (up to 50 characters is a useful rule of thumb).

Creating New Print Templates

To define a new print template, click the **Create new print template** link. This opens a window with four parts. The first part lists the variables that can be used in the template together with their meaning and an example of each.

Variable	Description	Example
{$u.username}	User account name	12345678
{$u.password}	User account password	87654321
{$u.enabled}	Non-zero if the guest account is enabled	1
{$u.role_name}	Role assigned to guest account	Guest
{$u.start_time}	Time at which the guest account will become active	1155772123
{$u.expire_time}	Time at which the guest account will expire	1155858523
{$u.expire_postlogin}	Lifetime of the guest account login in minutes after login	120
{$u.visitor_name}	User's name	Susan Guest
{$u.visitor_company}	User's company name	Acme Sprockets
{$u.sponsor_name}	Sponsor's name	John Sponsor
{$u.*custom_field*}	Custom fields attached to the account	
{$action}	Action taken on account (create, delete or edit)	create
{$source}	Source of account action (create_user, reset_password, etc.)	create_user

This section is followed by three other sections: the body, the header and the footer. Each section must be written in HTML. There is provision in each section for the insertion of multiple content items such as logos.

You are able to add Smarty template functions and blocks to your code. These act as placeholders to be substituted when the template is actually used.

See "Smarty Template Syntax" in the "Reference" chapter for further information on Smarty template syntax.

You are able to use an **{if}** statement to define a single print template that caters for multiple situations. For example if you want to customize the print template to display different content depending on the action that has been taken, the following code could be used:

```
{if $action == "create"}
<p>
 Your guest account has been created and is now ready to use!
</p>
<ul>
{if $site_ssid}
 <li>Connect to the wireless network named: <b>{$site_ssid}</b></li>
{/if}
 <li>Make sure your network adapter is set to 'DHCP - Obtain an IP address
Automatically'.</li>
 <li>Open your Web browser.</li>
 <li>Enter your username and password in the spaces provided.</li>
</ul>
{elseif $action == "edit"}
```

```
<p>
 Your guest account has been updated.
</p>
{elseif $action == "delete"}
{/if}
<table {$table_class_content} width="500">
 <tbody>
{if $u.guest_name}
<tr>
<th class="nwaLeft">guest name</th>
<td class="nwaBody">{$u.guest_name}</td>
</tr>
{/if}
```

If this code is placed in the User Account HTML section it will cater for the create, edit and delete options.

Print Template Wizard

The **Create new print template using wizard** link provides a simplified way to create print templates by selecting a basic style and providing a logo image, title and content text, and selecting the guest account fields to include.

A real-time preview allows changes made to the design to be viewed immediately.

To use the Print Template Wizard, first select a style of print template from the Style list. Small thumbnail images are shown to indicate the basic layout of each style. There are four built-in styles:

- **Table** – Best for square or nearly square logo images, and well suited for use with “scratch card” guest accounts.
- **Simple** – Best for wide or tall logo images and for situations where an operator will print a page with guest account details.
- **Centered** – Best for wide logo images; less formal design.
- **Label Printer** – These print template styles are designed for small thermal printers in various widths. On-screen assistance is provided when printing to ensure that a consistent result can be obtained.

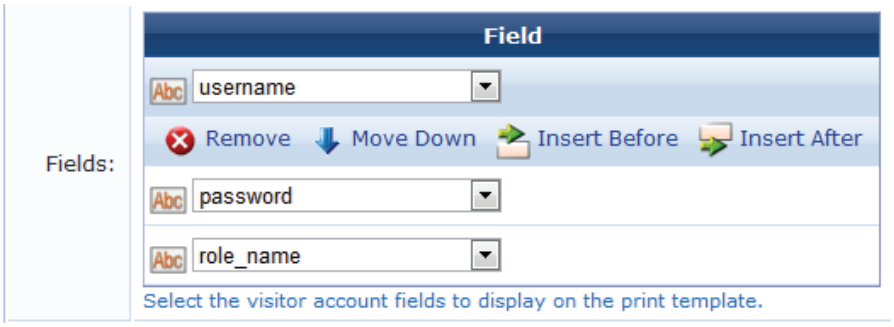
Click the **Preview at right** or **Preview at bottom** link at the top of the page to move the real-time preview of the print template.

Each of the basic styles provides support for a logo image, title area, subtitle area, notes area, and footer text. These items can be customized by typing in an appropriate value in the Print Template Wizard.

As the print template is a HTML template, it is possible to use HTML syntax as well as Smarty template code in these areas. See the **“Reference”** chapter for reference material about HTML and Smarty template code.

The print template may also contain visitor account fields. The value of each field is displayed in the print template. By default, the wizard sets up the template with the **username**, **password** and **role_name** fields, but these may be customized.

Options in the **Fields** row let you add, remove, or change the order of fields. Use the drop-down list to choose the field name, then click the icon at the left of the drop-down list. The field’s row expands to include the option links.



Use the **Remove**, **Move Up**, **Move Down**, **Insert Before**, and **Insert After** links to adjust the fields that are to be included on the print template.

Click the **Create Template** button to save your newly created print template and return to the list.

Modifying Wizard-Generated Templates

Once you have created a print template using the print template wizard, you can return to the wizard to modify it.

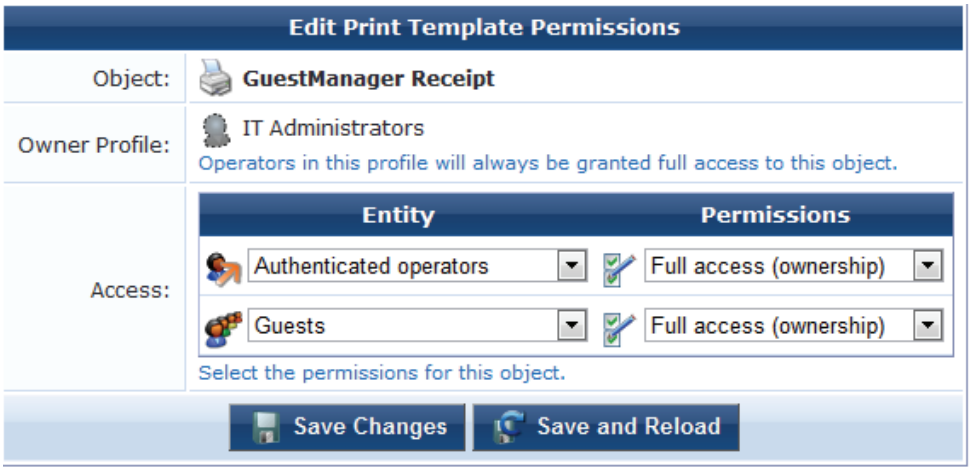
Click the **Edit print template code (Advanced)** link to use the standard print template editor. See "Creating New Print Templates" for a description.

NOTE

If you use the wizard to edit a print template after changes have been made to it outside the wizard, those changes will be lost. This is indicated with the warning message "The print template code has been modified. Making changes using the wizard will destroy any changes made outside of the wizard."

Setting Print Template Permissions

On the **Configuration > Print Templates** list view, the **Permissions** link for a template can be used to control access to an individual print template at the level of an operator profile. The Permissions link is only displayed if the current operator has the Object Permissions privilege. This privilege is located in the Administrator group of privileges.



The permissions defined on this screen apply to the print template identified in the "Object" line.

The owner profile always has full access to the print template.

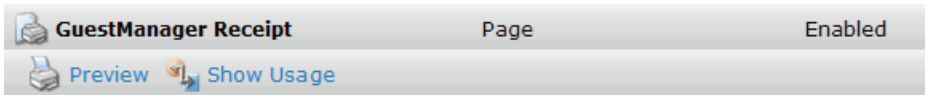
To control access to this print template by other entities, add or modify the entries in the "Access" list. To add an entry to the list, or remove an entry from the list, click one of the icons in the row. A **Delete** icon and an **Add** icon will then be displayed for that row.

Select one of the following entities in the Entity drop-down list:

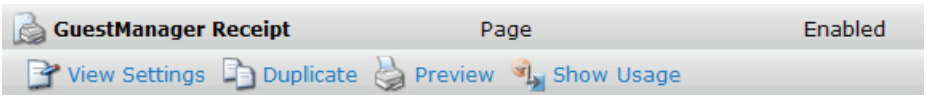
- **Operator Profiles** – a specific operator profile may be selected. The corresponding permissions will apply to all operators with that operator profile.
- Other Entities
  - **Authenticated operators** – the permissions for all operators (other than the owner profile) may be set using this item. Permissions for an individual operator profile will take precedence over this item.
  - **Guests** – the permissions for guests may be set using this item.

The permissions for the selected entity can be set using the Permissions drop-down list:

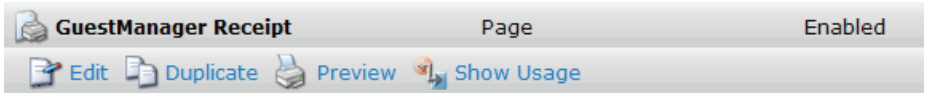
- **No access** – the print template is not visible in the list, and cannot be used, edited, duplicated, or deleted.
- **Visible-only access** – the print template is visible in the list, but cannot be edited, duplicated, or deleted.



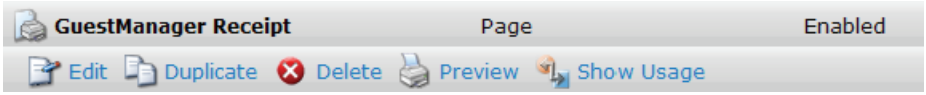
- **Read-only access** – the print template is visible in the list, and the settings for it may be viewed. The print template cannot be edited or deleted.



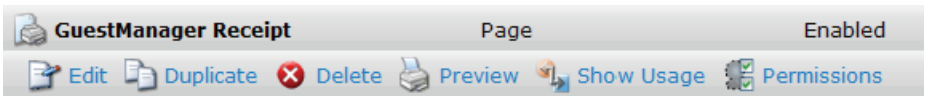
- **Update access** – the print template is visible in the list, and may be edited. The print template cannot be deleted and the permissions for the print template cannot be modified.



- **Update and delete access** – the print template is visible in the list, and may be edited or deleted. The permissions for the print template cannot be modified.



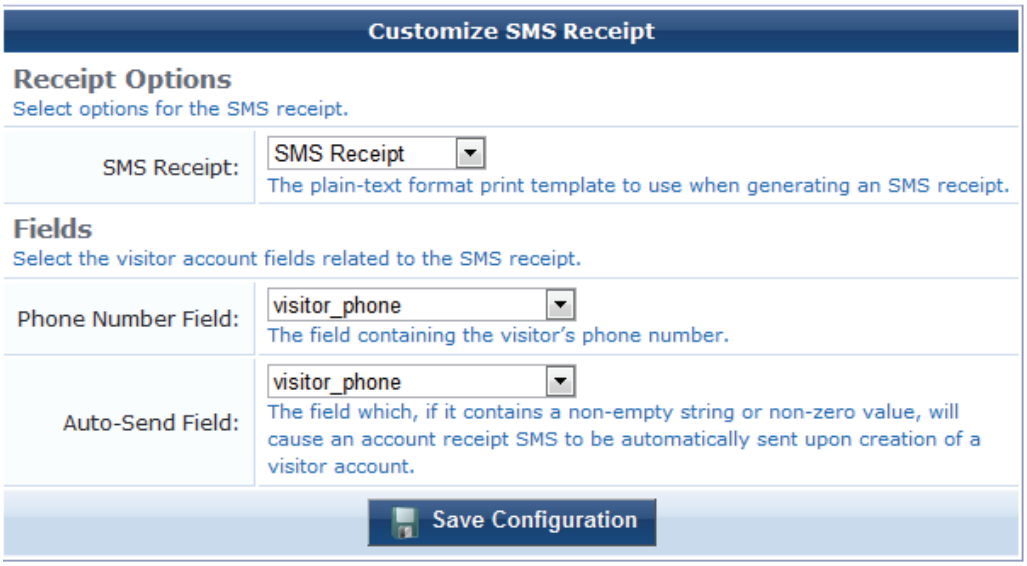
- **Full access (ownership)** – the print template is visible in the list, and may be edited or deleted. The permissions for the print template can be modified, if the operator has the Object Permissions privilege.



Customize SMS Receipt

Navigate to **Configuration > SMS Receipts** to configure SMS receipt options. These fields are described for the SMS plugin configuration page. Use the SMS receipt page for further customization. For information on standard SMS services, see "SMS Services" in the "Administration" chapter.

Figure 29 *Customize SMS Receipt page*



SMS Receipt Fields

The behavior of SMS receipt operations can be customized with certain guest account fields. You can override global settings by setting these fields.

- **sms_enabled** – This field may be set to a non-zero value to enable sending an SMS receipt. If unset, the default value is true.
- **sms_handler_id** – This field specifies the handler ID for the SMS service provider. If blank or unset, the default value from the SMS plugin configuration is used.
- **sms_template_id** – This field specifies the print template ID for the SMS receipt. If blank or unset, the default value from the SMS plugin configuration is used.
- **sms_phone_field** – This field specifies the name of the field that contains the visitor's phone number. If blank or unset, the default value from the SMS plugin configuration is used.
- **sms_auto_send_field** – This field specifies the name of the field that contains the auto-send flag. If blank or unset, the default value from the SMS plugin configuration is used. Additionally, the special values "**_Disabled**" and "**_Enabled**" may be used to never send an SMS or always send an SMS, respectively.

The logic used to send an SMS receipt is:

- If SMS receipts are disabled, take no action.
- Otherwise, check the auto-send field.
  - If it is "_Disabled" then no receipt is sent.
  - If it is "_Enabled" then continue processing.
  - If it is any other value, assume the **auto-send** field is the name of another guest account field. Check the value of that field, and if it is zero or the empty string then no receipt is sent.

- Determine the phone number – if the **phone number** field is set and the value of this field is at least 7 characters in length, then use the value of this field as the phone number. Otherwise, if the value of **the auto-send** field is at least 7 characters in length, then use the value of this field as the phone number.
- If the phone number is at least 7 characters long, generate a receipt using the specified plain-text print template and send it to the specified phone number.

Configuring Access Code Logins

This section explains how to configure the Guest Manager to create multiple accounts that have the ability to log in with only the username. We will refer to this as an **Access Code**.

Customize Random Username and Passwords

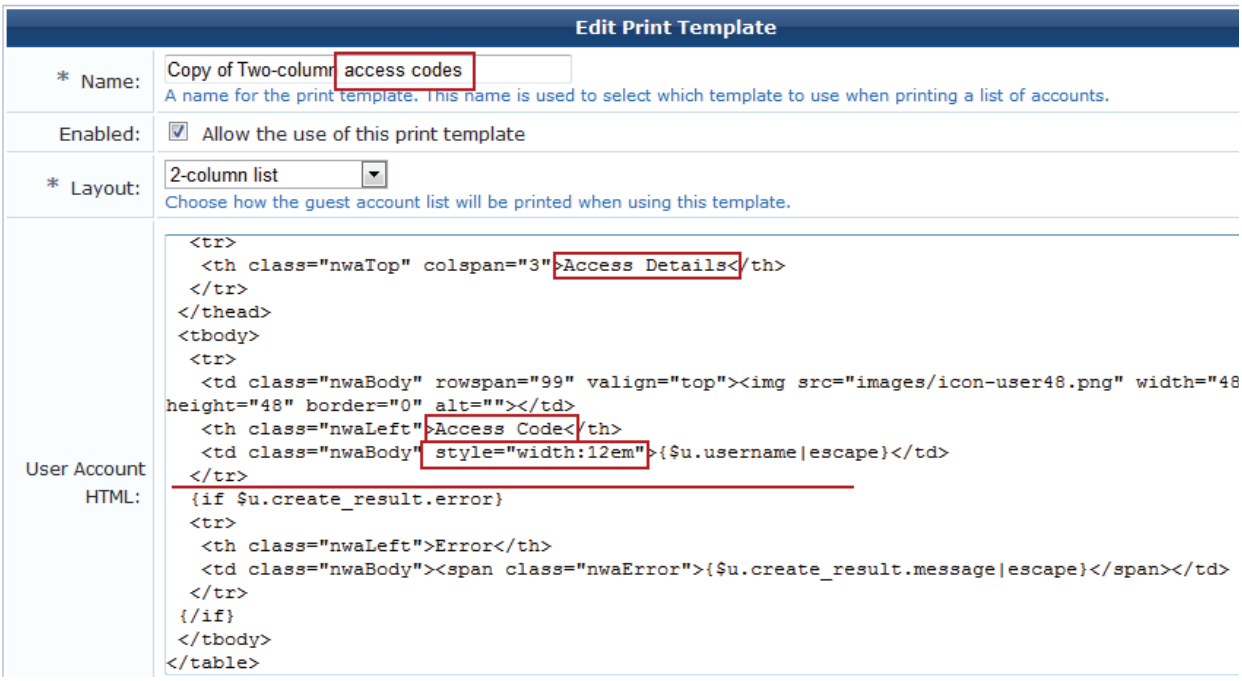
In this example we will set the random usernames and passwords to be a mix of letters and digits.

1. Navigate to **Configuration > Guest Manager**. The **Configure Guest Manager** form opens.
2. In the **Username Type** field, select **Random Letters and digits**. Note that the generator matching the complexity will also include a mix of upper and lower case letters.
3. In the **Username Length field**, select **8** characters.
4. Configure other settings. See "Default Settings for Account Creation" for a description. Click **Save Configuration** to save your changes.

Create the Print Template

By default, the print templates include username, password, and expiration, as well as other options. For the purpose of access codes, we only want the username presented. This access code login example bases the print template off an existing scratch card template.

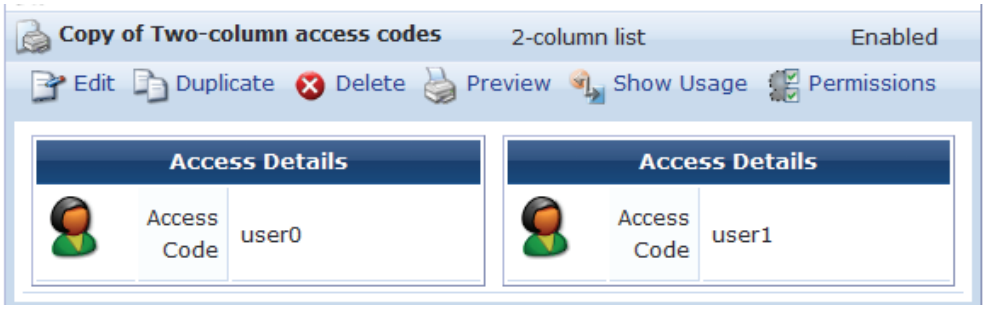
1. Navigate to **Configuration > Print Templates.**
2. Select **Two-column scratch cards** and click **Duplicate**.
3. Select the **Copy of Two-column scratch cards** template, then click **Edit**.
4. In the **Name** field, substitute **Access Code** for **Username** as shown below.



5. Remove extraneous data from the **User Account HTML** field. Example text is shown below.

```
<table {$table_class_content}>
 <thead>
<tr>
   <th class="nwaTop" colspan="3">Access Details</th>
  </tr>
 </thead>
 <tbody>
  <tr>
   <td class="nwaBody" rowspan="99" valign="top"><img src="images/icon-user48.png"
width="48" height="48" border="0" alt=""></td>
   <th class="nwaLeft">Access Code</th>
<td class="nwaBody" style="width:12em">{$u.username|htmlspecialchars}</td>
  </tr>
 {if $u.create_result.error}
  <tr>
   <th class="nwaLeft">Error</th>
<td class="nwaBody"><span class="nwaError">{$u.create_result.message}</span></td>
  </tr>
 {/if}
 </tbody>
</table>
```

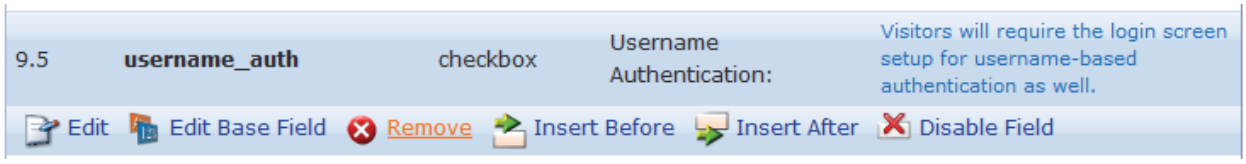
6. Click **Save Changes** to save your settings.
7. To preview the new template, select the template in the Guest Manager Print Templates list, then click Preview. The template created in this example appears as shown below.



Customize the Guest Accounts Form

Next, modify the **Guest Accounts** form to add a flag that to allows access-code based authentication.

1. Navigate to **Configuration > Forms & Views.**
2. In the **Customize Forms & Views** list, select **create_multi** and then click **Edit Fields**.
3. In the **Edit Fields** list, look for a field named **username_auth**. If the field exists, but is not bolded and enabled, select it and click **Enable Field**.



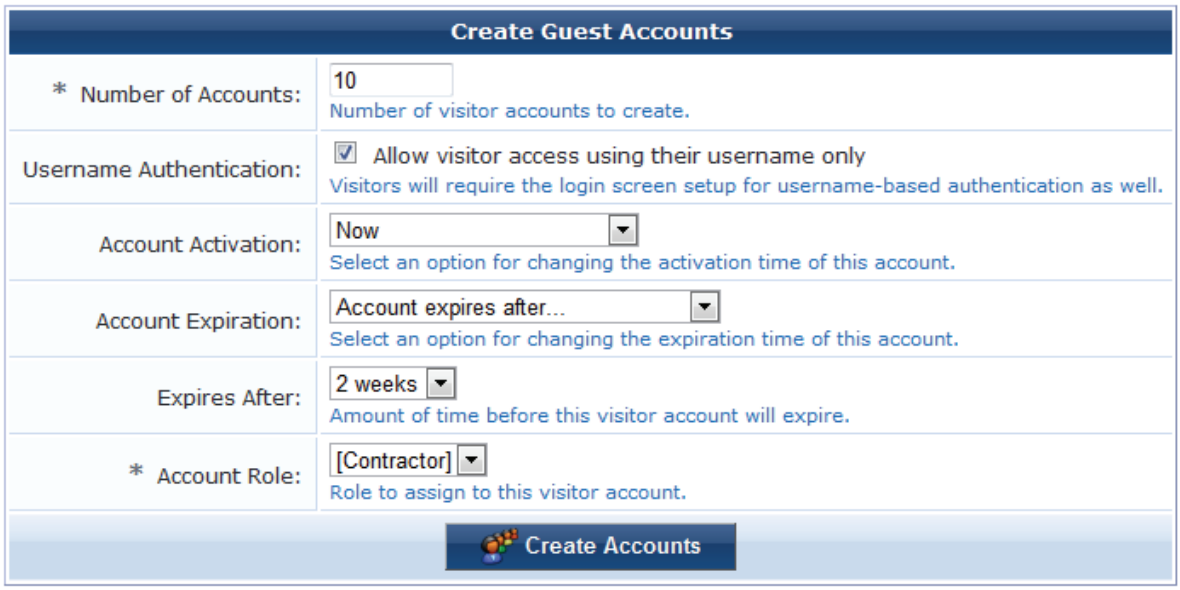
   If the field does *not* exist, select any field in the list (for example, **num_accounts**) and select **Insert After**. Click the **Field Name** drop-down list, select **username_auth** and allow the page to refresh. The defaults should be acceptable, but feel free to customize the label or description.

4. Click **Save Changes** to save your settings. Once the field is enabled or inserted, you should see it bolded in the list of fields.

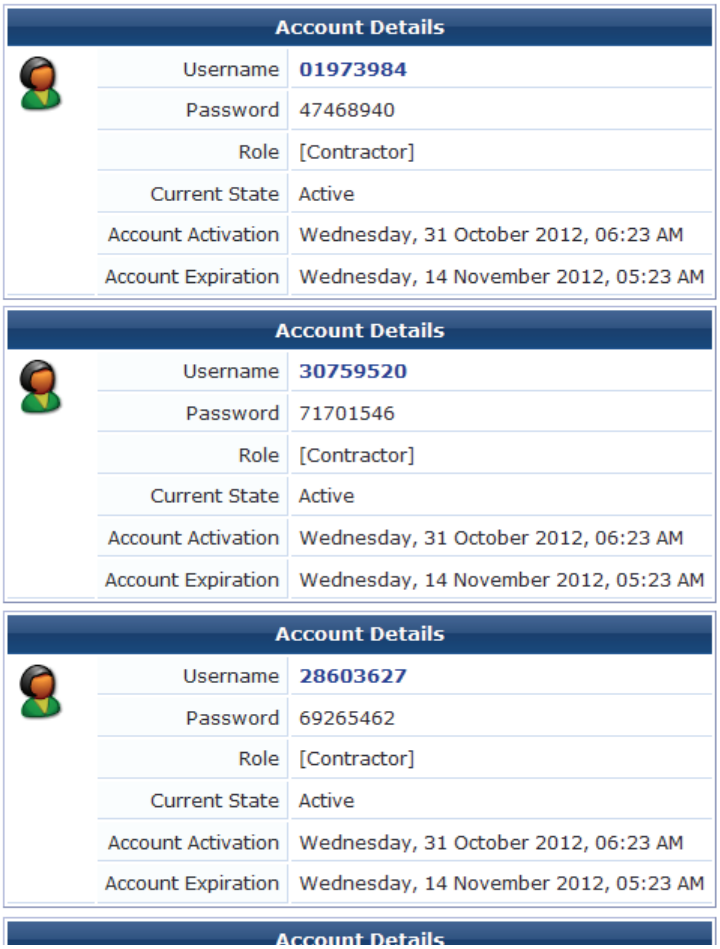
Create the Access Code Guest Accounts

Once the account fields have been customized, you can create new accounts.

1. Navigate to **Guest > Create Multiple.**
2. Mark the check box in the **Username Authentication** row that was added in the procedure above. (If you do not select this check box and if the username is entered on the login screen, the authentication will be denied.) The example shown below will create 10 accounts that will expire in two weeks, or fours hours after the visitors first log in, whichever comes first.



3. Click **Create Accounts** to display the **Finished Creating Guest Accounts** page. If you create a large number of accounts, they are created at one time but might not all be displayed at the same time. (This will not affect the printing action in the following step.)



Account Details	
Username	**01973984**
Password	47468940
Role	[Contractor]
Current State	Active
Account Activation	Wednesday, 31 October 2012, 06:23 AM
Account Expiration	Wednesday, 14 November 2012, 05:23 AM

Account Details	
Username	**30759520**
Password	71701546
Role	[Contractor]
Current State	Active
Account Activation	Wednesday, 31 October 2012, 06:23 AM
Account Expiration	Wednesday, 14 November 2012, 05:23 AM

Account Details	
Username	**28603627**
Password	69265462
Role	[Contractor]
Current State	Active
Account Activation	Wednesday, 31 October 2012, 06:23 AM
Account Expiration	Wednesday, 14 November 2012, 05:23 AM

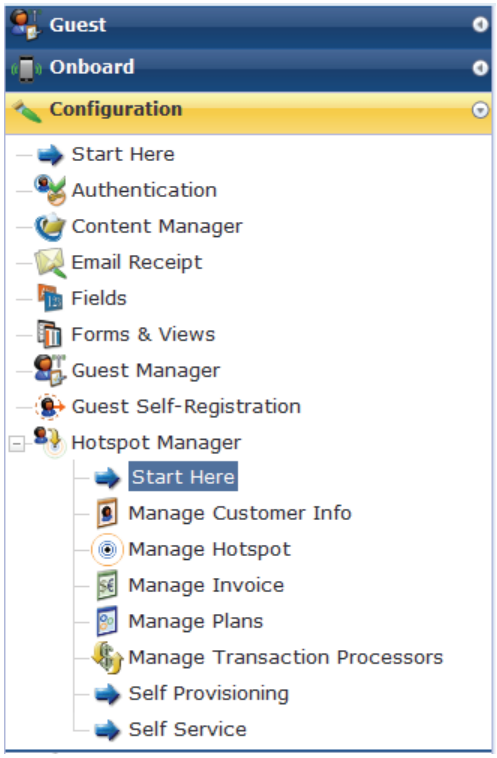
Account Details

4. Confirm that the accounts settings are as you expected with respect to letters and digits in the username and password, expiration, and role.
5. Click the **Open print window using template** drop-down list and select the new print template you created using this procedure See **"Create the Print Template"** for a description of this procedure. A new window or tab will open with the cards.

The Hotspot Manager controls self-provisioned guest or visitor accounts. This is where the customer is able to create his or her own guest account on your network for access to the Internet. This can save you time and resources when dealing with individual accounts.

Accessing Hotspot Manager

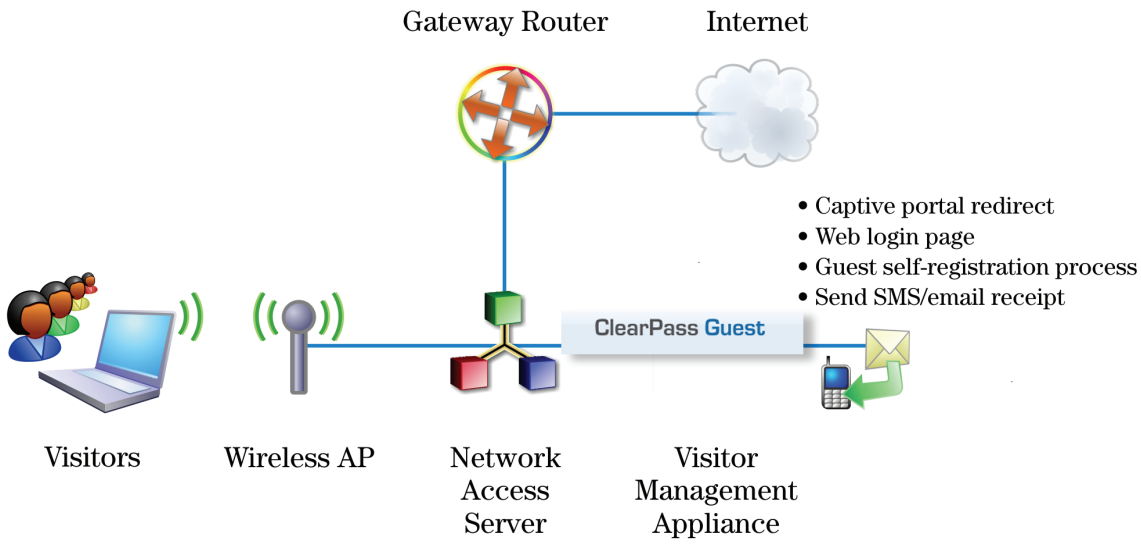
To access ClearPass Guest's hotspot management features, click the **Configuration** link in the left navigation, then click **Hotspot Manager**.



About Hotspot Management

The following diagram shows how the process of customer self provisioning works.

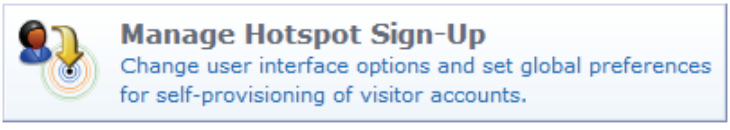
Figure 30 *Guest self-provisioning*



- Your customer associates to a local access point and is redirected by a captive portal to the login page.
- Existing customers may log in with their Hotspot username and password to start browsing.
- New customers click the **Hotspot Sign-up** link.
- On page 1, the customer selects one of the Hotspot plans you have created.
- On page 2, the customer enters their personal details, including credit card information if purchasing access.
- The customer's transaction is processed, and, if approved, their visitor account is created according to the appropriate Hotspot plan.
- On page 3, the customer receives an invoice containing confirmation of their transaction and the details of their newly created visitor account.
- The customer is automatically logged in with their username and password, providing instant Hotspot access.

Managing the Hotspot Sign-up Interface

You can enable visitor access self provisioning by navigating to **Configuration > Hotspot Manager** and selecting the **Manage Hotspot Sign-up** command.



The Hotspot Preferences form opens. This form allows you to change user interface options and set global preferences for the self-provisioning of visitor accounts.

The **Enable visitor access self-provisioning** check box must be ticked for self-provisioning to be available.

The **Require HTTPS** field, when enabled, redirects guests to an HTTPS connection for greater security.

The **Service Not Available** Message allows a HTML message to be displayed to visitors if self-provisioning has been disabled. **See** **"Smarty Template Syntax"** in the Reference chapter for details about the template syntax you may use to format this message.

Click the **Save Changes** button after you have entered all the required data.

Captive Portal Integration

To start the visitor self-provisioning process, new visitor registration is performed by redirecting the visitor to the URL specified on the Hotspot Preferences page; for example: https://guest.spiffywidgets.com/hotspot_plan.php. The Hotspot Sign-Up page opens to the first page of the wizard, Choose Plan.

The hotspot_plan.php page accepts two parameters:

- The **source** parameter is the IP address of the customer.
- The **destination** parameter is the original URL the customer was attempting to access (that is, the customer's home page). This is used to automatically redirect the customer on successful completion of the sign-up process.

For browsers without JavaScript, you may use the **<noscript>** tag to allow customers to sign up:

```
<noscript>
   <a href="https://guest.spiffywidgets.com/hotspot_plan.php">Hotspot Sign-Up</a>
</noscript>
```

However, in this situation the MAC address of the customer will not be available, and no automatic redirection to the customer's home page will be made. You may want to recommend to your customers that JavaScript be enabled for best results.

Look and Feel

The skin of a Web site is its external look and feel. It can be thought of as a container that holds the application, its style sheet (font size and color for example), its header and footer and so forth.

The default skin used by ClearPass Guest is the one that is enabled in the Plugin Manager. The skin is seen by all users on the login page.

SMS Services

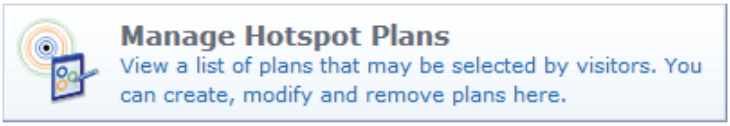
Configure the following settings in the **SMS Services** section of the **Hotspot Preferences** form to override the default SMS settings with your own custom configuration.

- **SMS Receipt**: Click this drop-down list to select the template you want to use for SMS receipts. The default value is **SMS Receipt**.
- **Phone Number Field**: Click this drop down list and identify the field that contains the visitor's phone number. The default value is **visitor_phone**.
- **Auto**-Send Field: Click this drop-down list and select the field which, when configured with any string or non-zero value, will trigger the automatic sending of an SMS receipt. The default value of this field is **auto_send_sms**.

Managing Hotspot Plans

Your Hotspot plans determine how a customer is to pay for Internet access when connected through ClearPass Guest. You also have the option to allow free access.

To view the list of hotspot plans your visitors can select and to access plan management, go to **Configuration > Hotspot Manager > Manage Plans**.



The Manage Hotspot Plans page opens, showing the list of default plans. Plans that are enabled have their name in **bold** and their icon in color: . Plans that are not enabled have their icon in gray: .

Plan Name	Description	Actions
Free Access	Free basic wireless access. Limited to 64 kbit, Web browsing traffic only, and a maximum of one hour.	Edit Delete
Hourly Access	Wireless access charged at $2.95 per hour. Offers full Internet access at 128 kbit/sec.	Edit Delete
Daily Access	Wireless access charged at $24.95 per day (24 hours). Offers full Internet access at 256 kbit/sec.	Edit Delete
Weekly Access	Wireless access charged at $54.95 per week (7 days). Offers full Internet access at 256 kbit/sec.	Edit Delete

- To create or edit an existing plan, see "Editing or Creating a Hotspot Plan".
- To delete a plan, click the **Delete** button in the plan's row. When a plan is deleted it is not possible to undo the deletion.

Editing or Creating a Hotspot Plan

When you create or edit a hotspot plan, you can customize which plans are available for selection, and any of the plan's details, such as its description, cost to purchase, allocated role, and the format of the customer's generated username and password.

1. To create or edit a plan, first go to **Configuration > Hotspot Manager > Manage Plans**, then:
   - To create a new plan, click the **Create Hotspot plan** link in the upper-right corner. The Create Hotspot Plan form opens.
   - To edit a plan, click the **Edit** link in the plan's row. The **Edit Hotspot Plan** form opens.

   The procedures are the same for both the Create Hotspot Plan and the Edit Hotspot Plan forms.

2. In the **Plan Details** area, enter a name for the plan and descriptions to display in the UI and the customer invoice.
3. To enable the plan, leave the **Enabled** check box marked. To disable the plan, unmark this check box. Disabled plans are not displayed to customers.
4. In the **User Account Details** area, you can specify the usage of numbers, letters, and symbols in the generated username and password. To use only digits, leave the value in the **Generated Username** and **Generated Password** fields set to **######**. To indicate a different combination of numbers, letters, or symbols, use the following parameters:
   - The number or hash symbol (**#**) is replaced with a random digit (0-9)
   - The dollar symbol (**$**) is replaced with a random letter
   - The underscore symbol (_) is replaced with a random lowercase letter
   - The carat symbol (**^**) is replaced with a random uppercase letter
   - The asterisk symbol (*) is replaced with a random letter or digit

- The “at” symbol (**@**) is replaced with a random letter or digit, excluding vowels
- The exclamation symbol (**!**) is replaced with a random punctuation symbol
- The ampersand symbol (**&**) is replaced with a random character (letter, digit, or punctuation symbol)
- All other characters are used without modification

For more information, see “Format Picture String Symbols” in the “Reference” chapter.

5. Complete the rest of the fields appropriately for your organization’s needs, then click **Create Plan** or **Edit Plan**. The Manage Hotspot Plans list opens with the new plan displayed.

Managing Transaction Processors

Your hotspot plan must also identify the transaction processing gateway used to process credit card payments. ClearPass Guest supports plugins for the following transaction processing gateways:

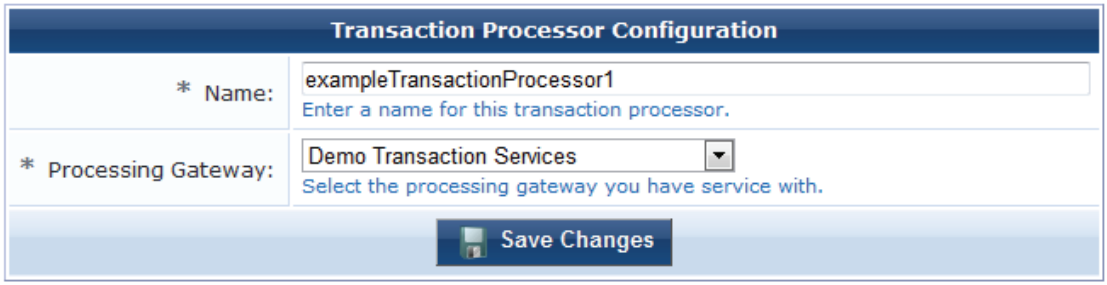
- Authorize.Net AIM
- CyberSource
- eWAY
- Netregistry
- Paypal
- WorldPay

ClearPass Guest also includes a Demo transaction processor that you can use to create hotspot forms and test hotspot transactions.

Creating a New Transaction Processor

To define a new transaction processor:

1. Go to **Configuration > Hotspot Manager**, click **Manage Transaction Processors**, then select **Create new transaction processor**.



2. In the **Name** field, enter a name for the transaction processor.
3. In the **Processing Gateway** drop-down list, select the gateway with which you have a service account. The form expands to include additional configuration fields for that gateway type.

   Each transaction processing gateway type requires unique merchant identification, password, and configuration information. Depending on the gateway provider, these configuration items will include some of the following:

   - API Login
   - API Password
   - API Username
   - Auto Email
   - Beagle Anti-Fraud

- Business Center Login
- Customer ID
- Installation ID
- Logging
- Merchant ID
- Mode
- Production Environment URL
- Shared Secret
- Signature
- Test Environment URL
- Test WSDL
- Transaction Key
- Transaction Password
- Transactions Timeout

If your transaction processor requires visitors to enter their address, ClearPass Guest will automatically include address fields in the guest self-registration forms that use that transaction processor.

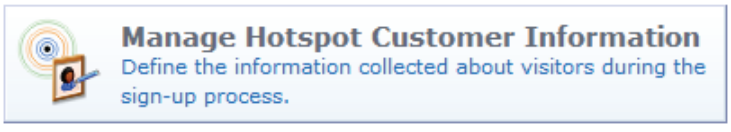
Managing Existing Transaction Processors

Once you define a transaction processor, it will appear in the transaction processor list. When you select an individual processors in the list, the list displays a menu that allows you to perform the following actions:

- **Edit** – changes the properties of the specified transaction processor
- **Delete** – removes the processor from the Transaction Processors list
- **Duplicate** – creates a copy of a transaction processor
- **Show Usage** – opens a window in the Transaction Processors list that shows if the profile is in use, and lists any hotspots associated with that transaction processor. Each entry in this window appears as a link to the **General Hotspot References** form that lets you change the transaction processor associated with that hotspot.

Managing Customer Information

You can customize the fields that the customer sees, the details of these fields, and the order in which they are presented. To customize the fields, go to **Configuration > Hotspot Manager > Manage Hotspot Customer Information**.



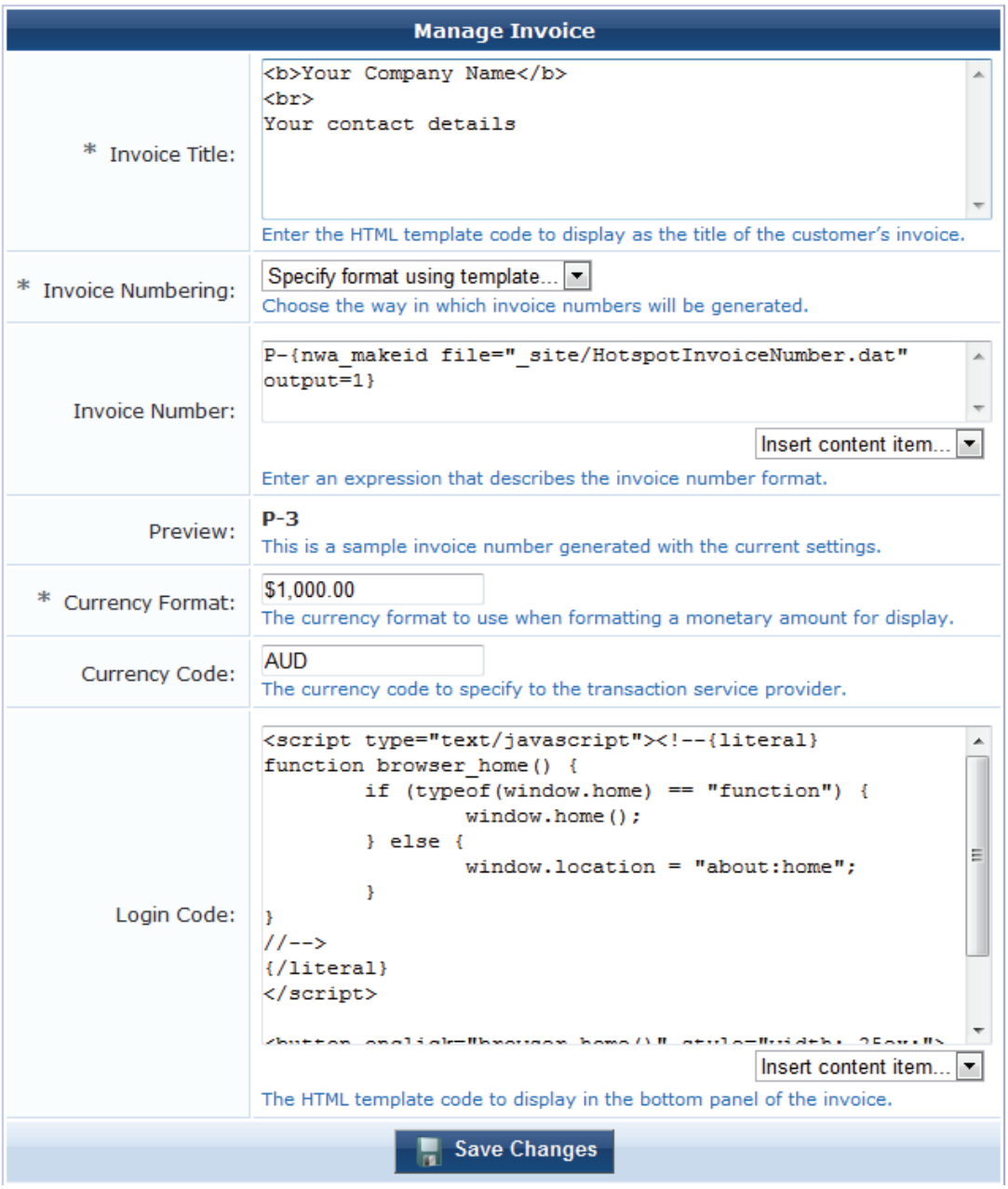
The Customize Form Fields view opens for the customer information form. See "Duplicating Forms and Views" in the "Configuration" chapter for instructions for completing the form field editor.

Managing Hotspot Invoices

After the customer's transaction has been processed successfully, the customer receives an invoice containing confirmation of their transaction and the details of their newly created hotspot user account. You can customize the title shown on the invoice and how the invoice number is created. You can also customize the currency displayed on the invoice.

To customize the hotspot invoice:

1. Go to **Configuration > Hotspot Manager > Manage Hotspot Invoice**. The Manage Hotspot Invoice form opens.



2. The **Invoice Title** must be written in HTML. See "Basic HTML Syntax" in the Reference chapter for details about basic HTML syntax.
3. Complete the rest of the fields appropriately. You can use Smarty functions on this page. See "Smarty Template Syntax" in the Reference chapter for further information on these. You can also insert content

items such as logos or prepared text. See "Customizing Self Provisioned Access" in the "Configuration" chapter for details on how to do this.

4. Click **Save Changes**.

Customizing the User Interface

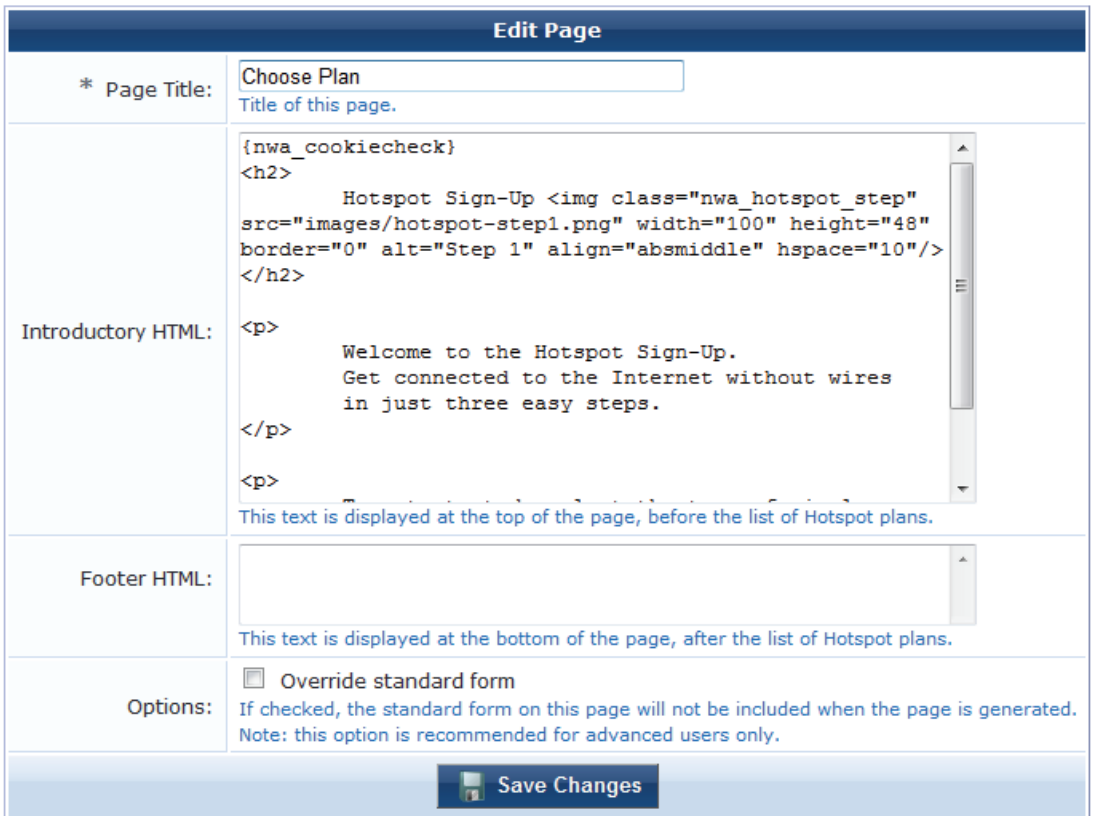
Each aspect of the user interface your hotspot customers see can be customized.

Customizing Visitor Sign-Up Page One

Page one of the guest self-provisioning process asks the guest to select a plan. An example of the default "Choose Plan" page is shown below.

To customize how this page is displayed to the guest, go to **Configuration > Hotspot Manager > Manage Hotspot Sign-Up**, then click the **Customize page 1 (Choose Plan)** link in the upper-right corner.

The Edit Hotspot Plan Selection Page form opens. You can use this form to edit the title, introductory text, and footer of the "Choose Plan" page. The introduction and the footer are HTML text that can use template syntax. See "Smarty Template Syntax" in the Reference chapter.


Edit Page
* Page Title:
Choose Plan
Title of this page.
Introductory HTML:
{nwa_cookiecheck}
<h2>
Hotspot Sign-Up <img class="nwa_hotspot_step"
src="images/hotspot-step1.png" width="100" height="48"
border="0" alt="Step 1" align="absmiddle" hspace="10"/>
</h2>

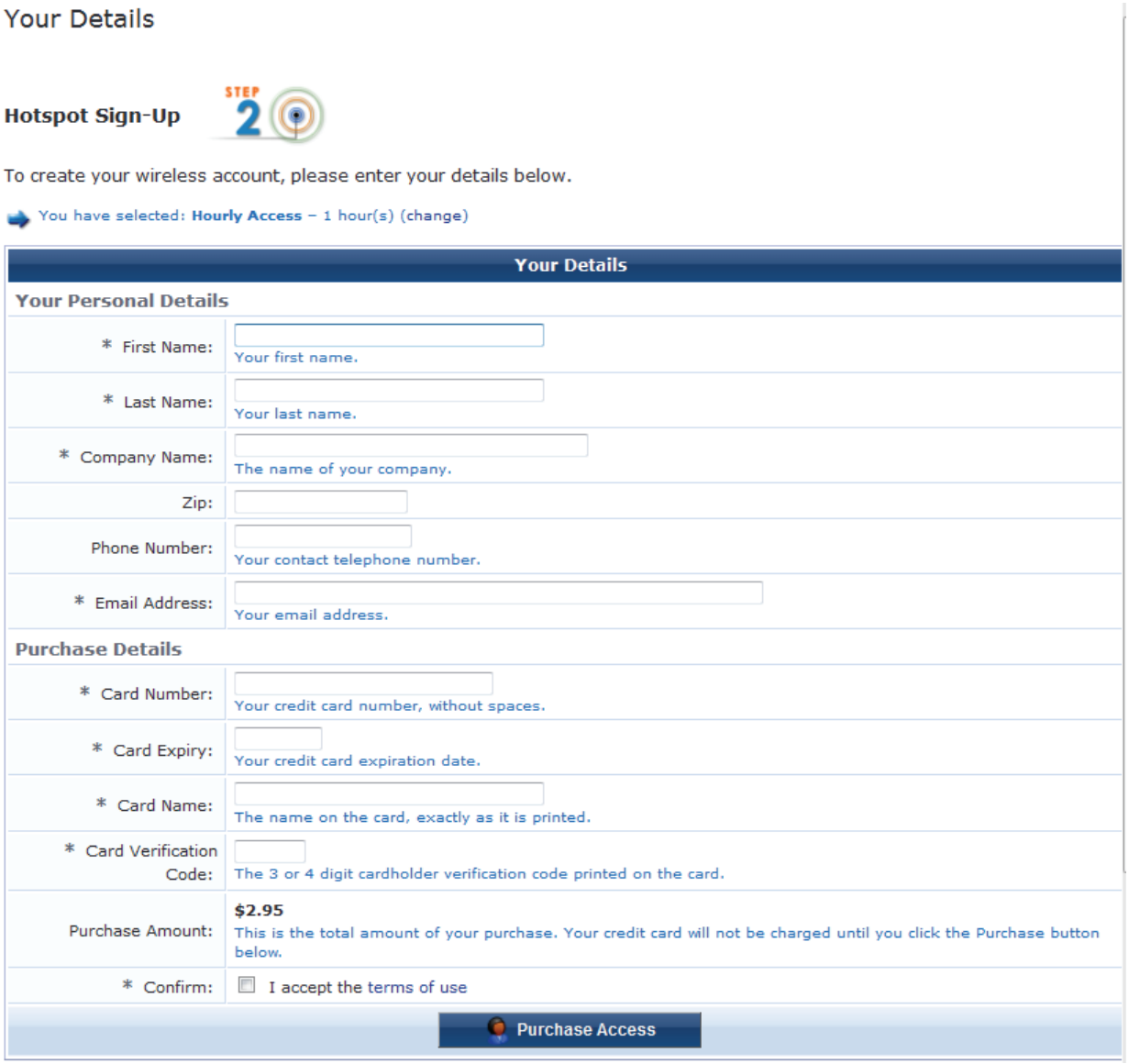
Welcome to the Hotspot Sign-Up.
Get connected to the Internet without wires
in just three easy steps.

This text is displayed at the top of the page, before the list of Hotspot plans.
Footer HTML:
This text is displayed at the bottom of the page, after the list of Hotspot plans.
Options:
Override standard form
If checked, the standard form on this page will not be included when the page is generated.
Note: this option is recommended for advanced users only.
Save Changes

Customizing Visitor Sign-Up Page Two

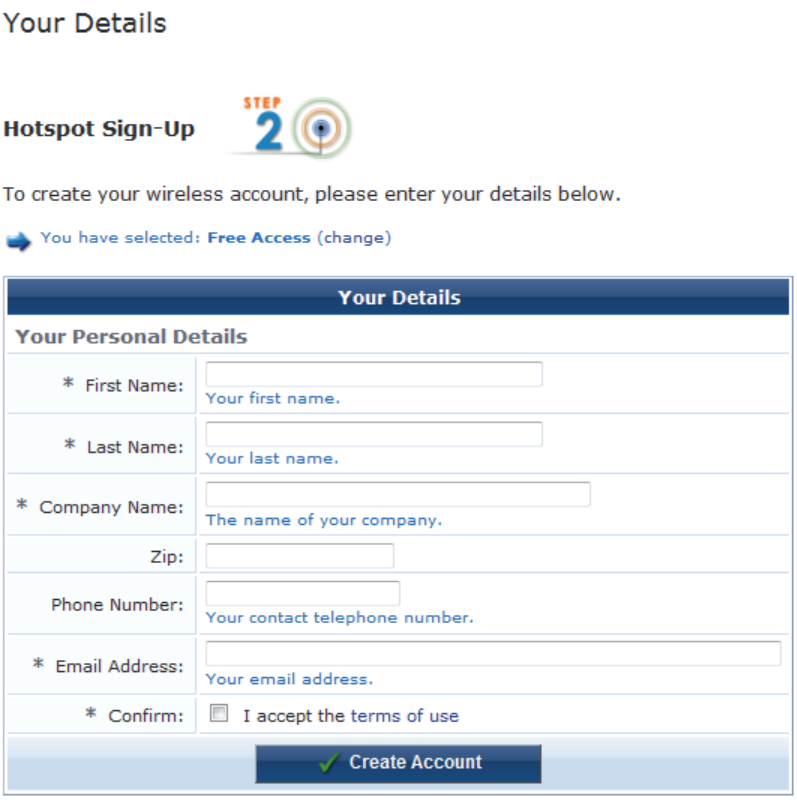
Page two of the guest self-provisioning process asks the guest to provide their personal details and payment method.

The example below shows the default “Your Details” page if the customer chooses to pay for the Hourly Access plan.



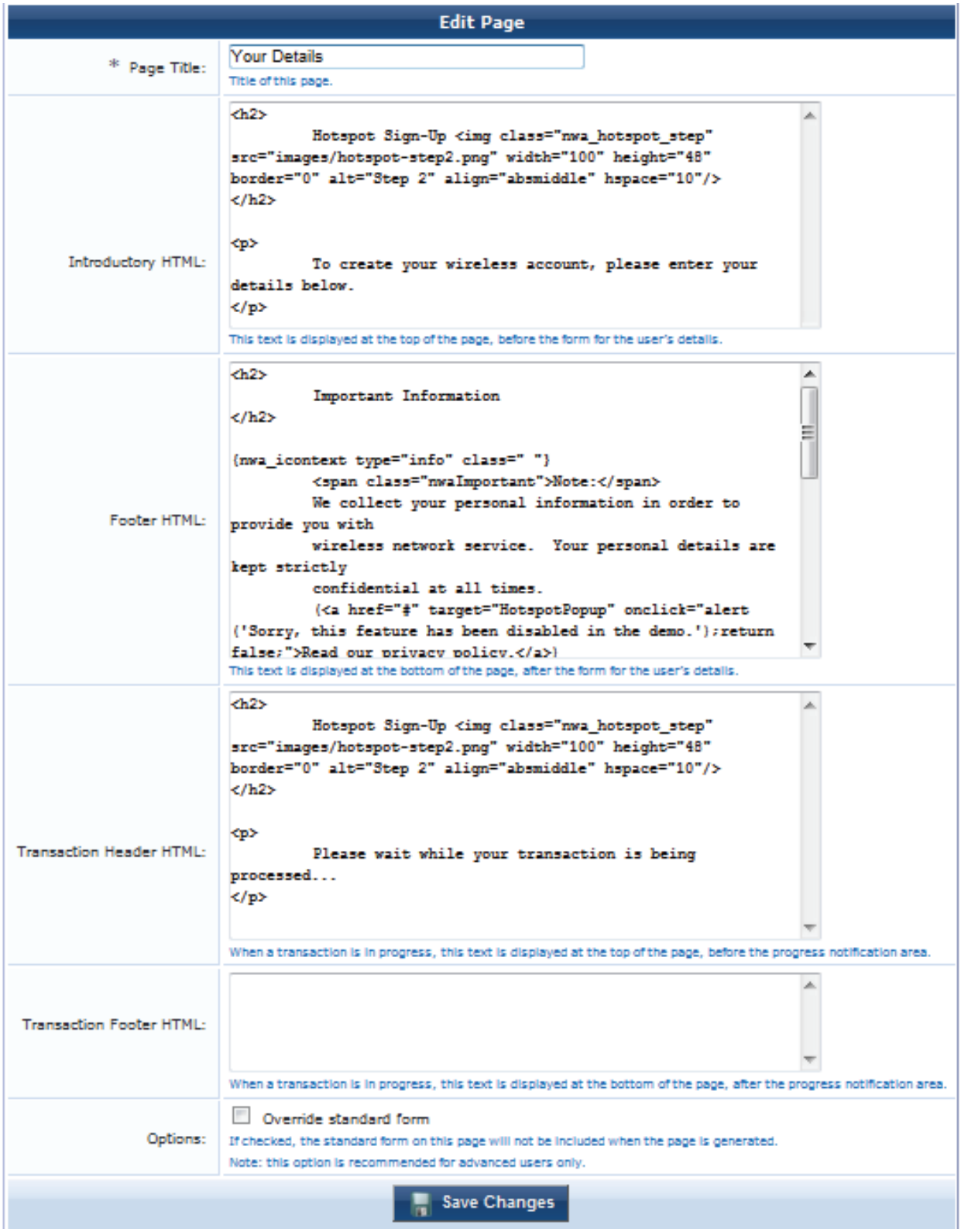
Although it is not shown in this illustration, the default page also includes footer text providing information about privacy policies and security pertaining to the data collected by this page.

The example below shows the default “Your Details” page for a customer who chooses the Free Access plan.



To customize how the “Your Details” page is displayed to the guest, go to **Configuration > Hotspot Manager > Manage Hotspot Sign-Up**, then click the **Customize page 2 (Customer Details)** link in the upper-right corner.

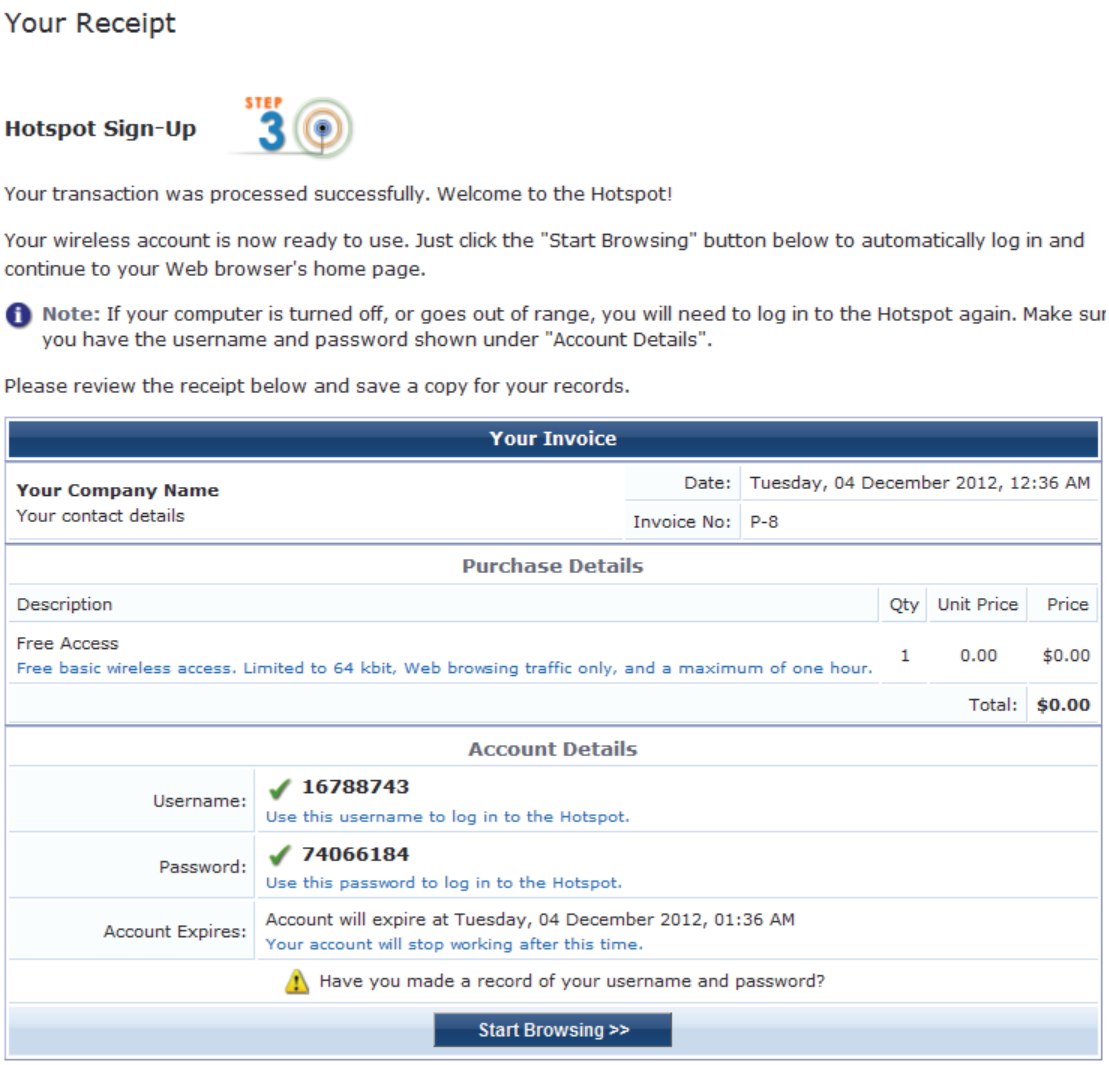
The Edit Hotspot User Details Page form opens. You can use this form to edit the content displayed when the customer enters their personal details, including credit card information if purchasing access. The progress of the user’s transaction is also shown on this page.



See "Smarty Template Syntax" in the Reference chapter for details about the template syntax you may use to format the content on this page.

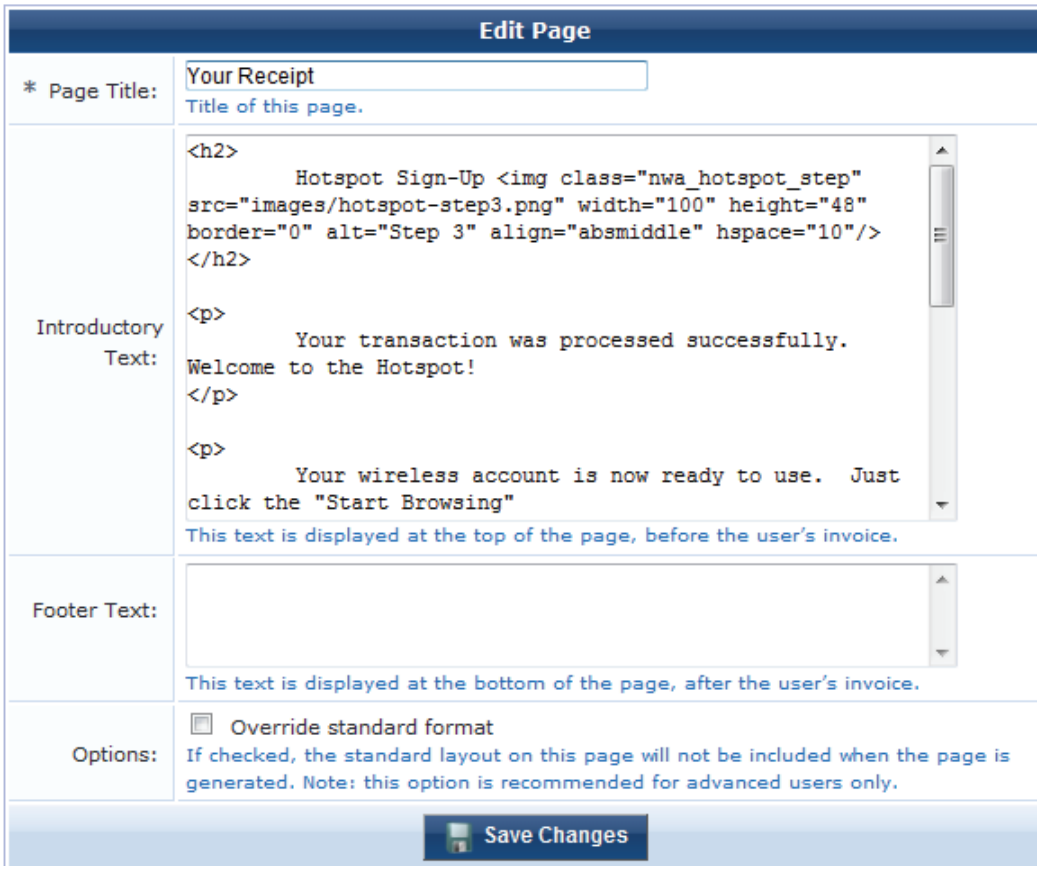
Customizing Visitor Sign-Up Page Three

Page three of the guest self-provisioning process provides the customer an invoice containing confirmation of their transaction and the details of their newly created wireless account. An example of the default “Your Receipt” page is shown below.



To customize how the “Your Receipt” page is displayed to the guest, go to **Configuration > Hotspot Manager > Manage Hotspot Sign-Up**, then click the **Customize page 3 (Invoice or Receipt)** link in the upper-right corner.

The Edit Hotspot User Receipt Page form opens. You can use this form to edit the title, introductory text, and footer text of the receipt page.



See "Smarty Template Syntax" in the Reference chapter for details about the template syntax you may use to format the content on this page.

Viewing the Hotspot User Interface

The Hotspot Manager allows you to view and test Hotspot self-provisioning pages, as well as log in to and view the Hotspot self-service portal that allows customers to view their current account expiration date, purchase time extensions, log out of the Hotspot, or change their user password.

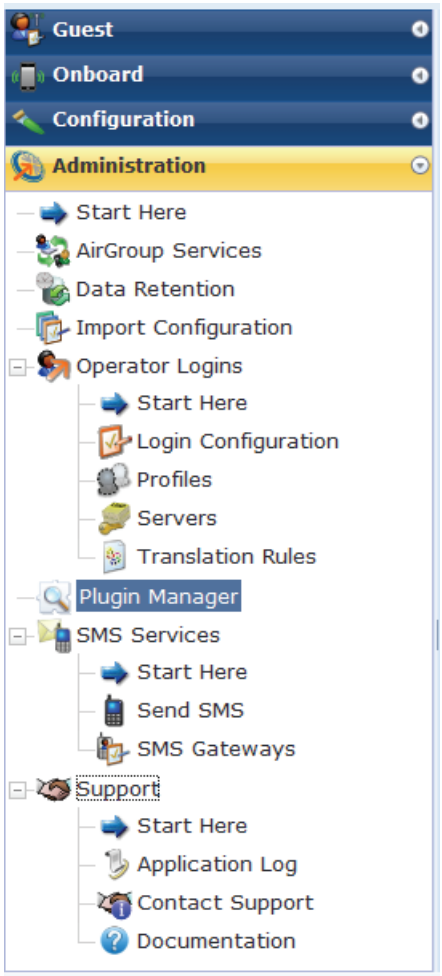
To access either of these user pages, navigate to **Configuration > Hotspot manager** and select the **Self-Provisioning** or **Self-Service** links in the left navigation menu.

The Administration module provides tools used by a network administrator to perform both the initial configuration and ongoing maintenance of ClearPass Guest.

Accessing Administration

To access ClearPass Guest's administration features, click the **Administration** link in the left navigation.

Figure 31 *The Administration Module's Left Navigation*



AirGroup Services

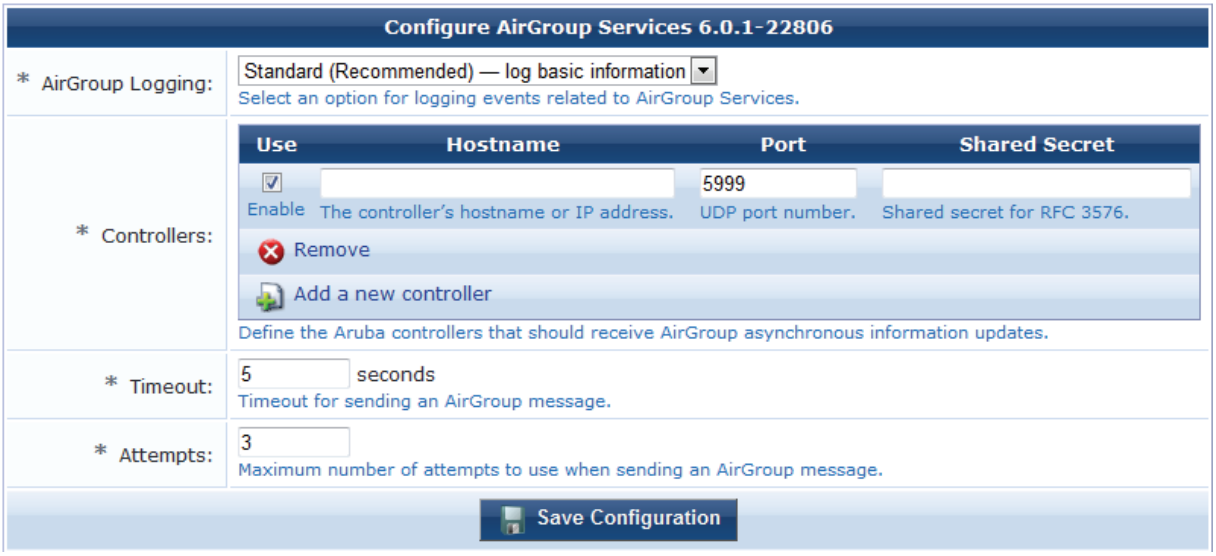
This section describes configuration options for the AirGroup Services plugin, and provides links to other AirGroup steps performed in ClearPass Guest. For an overview of AirGroup functionality, see "AirGroup Deployment Process" in the "ClearPass Guest Overview" chapter. For complete AirGroup deployment information, refer to the AirGroup Deployment Guide and the ClearPass Policy Manager documentation.

Configuring the AirGroup Services Plugin

To enable support for dynamic notification of AirGroup events when new devices are added, and to configure AirGroup logging, each AirGroup-enabled Aruba controller must also be defined in ClearPass Guest.

To configure the AirGroup Services plugin:

1. Go to **Administration > AirGroup Services** and click the **Configure AirGroup Services** command link. The Configure AirGroup Services form opens.



2. In the **AirGroup Logging** drop-down list, choose one of the following options:
   - **Disabled—Do not log AirGroup related events**
   - **Standard (Recommended)—Log basic information**
   - **Extended—Log additional information**
   - **Debug—Log debug information**
   - **Trace—Log all debug information**
3. In the **Controllers** row, to add a new AirGroup controller and enable it to receive dynamic notifications of AirGroup events, click the **Add a new controller** link. The row expands to include fields for entering the controller's properties.
4. Specify the following properties for each AirGroup-enabled controller:
   a. **Hostname or IP address**
   b. **Port number** – Note that this should be airgroup cppm-server aaa rfc3576-server, the UDP port number of the AirGroup process on the controller. This is the same port number that was defined when the CPPM interface was configured. The default in ClearPass Guest is 5999.
   c. **Shared secret** – This is the rfc-3576_udp_port shared secret used for AirGroup
5. In the **Timeout** row, enter the number of seconds after which an attempt to send an AirGroup message will time out.

6. In the **Attempts** row, enter the maximum number of times the system should attempt to send an AirGroup message.
7. Click **Save Configuration**.

Creating AirGroup Administrators

AirGroup Administrators are users of ClearPass Guest who can define and manage their organization's shared devices. Devices can be shared globally, or shared with restrictions based on the username, role, or location of the user trying to access the device.

The AirGroup Administrator profile is automatically created in ClearPass Guest when the AirGroup Services plugin is installed. This profile is used to define the AirGroup Administrator role. To create an AirGroup Administrator, see "Creating a New Operator" in the "Operator Logins" chapter.

Creating AirGroup Operators

AirGroup Operators are users of ClearPass Guest who can provision a limited number of their own personal devices. Each device provisioned by an operator is automatically shared with all of that operator's provisioned devices. The operator can also define a group of other users who are allowed to share the operator's devices.

The AirGroup Operator profile is automatically created in ClearPass Guest when the AirGroup Services plugin is installed. This profile is used to define the AirGroup Operator role. To create an AirGroup Operator, see "Creating a New Operator" in the "Operator Logins" chapter.

Authenticating AirGroup Users via LDAP

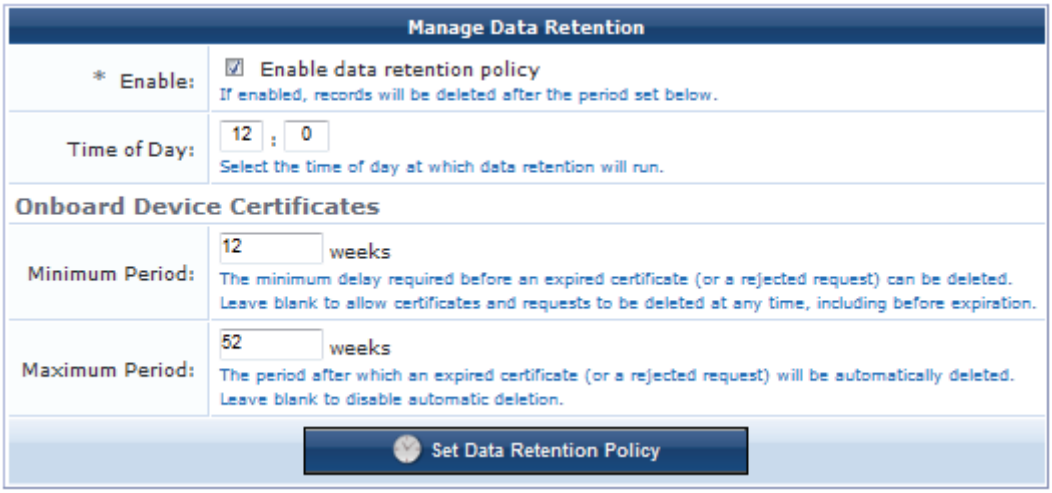
ClearPass Guest supports LDAP authentication for administrators and operators. To provide AirGroup Services to LDAP-authenticated users:

1. Define the LDAP server for AirGroup. See "External Operator Authentication" in the "Operator Logins" chapter.
2. Define the appropriate translation rules to categorize the LDAP users. See "Custom LDAP Translation Processing" in the "Operator Logins" chapter.

Data Retention

The Data Retention Policy page (**Administration > Data Retention**) lets you manage historical data by archiving or deleting it. For a data retention policy to take effect, you must schedule and enable database maintenance. To do so, refer to the ClearPass Policy Manager documentation.

Figure 32 *Data Retention Policy page*



Select **Enable** to enable the data retention policy option and enter how many weeks in the **Log Rotation** field to indicated how many weeks you want log files kept before they are deleted.

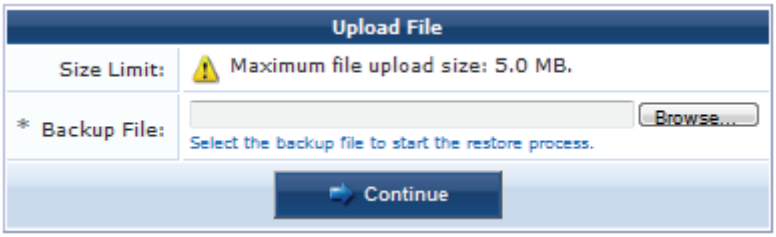
For mobile device certificates, select the minimum delay, in weeks, required before an expired certificate or rejected request can be deleted. The maximum period is the number of weeks after which an expired certificate is automatically deleted.

Import Configuration

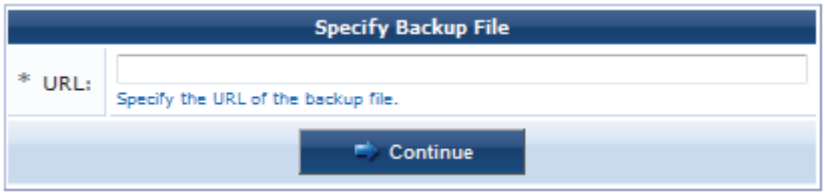
The Import Configuration screen lets you import selected items from a ClearPass Guest 3.9 configuration.

To import configuration settings from a standalone ClearPass Guest 3.9 backup file:

1. Go to **Administration > Import Configuration**. The Import Configuration: Step 1 page opens with the Upload File form displayed.

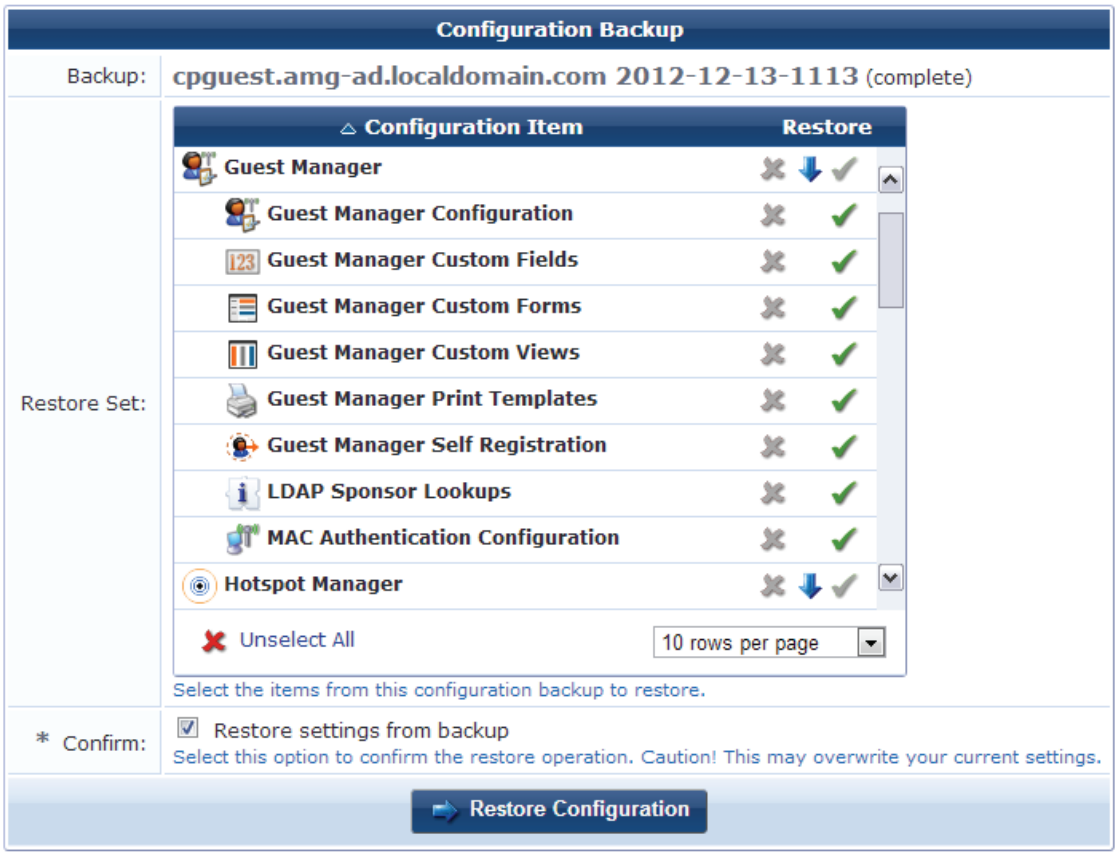


2. If your file does not exceed the 5.0 MB size limit, use this form to upload your file. If your file is larger than the maximum file upload size of 5.0 MB, you must specify a URL instead. Click the **Restore a backup from a URL** link above the **Upload File** form. The Specify Backup File form is displayed.



To use the Upload File form, click the **Browse** button in the **Backup File** row to navigate to and select the backup file you want to restore. To use the Specify Backup File form, enter the URL for the backup file.

Click **Continue**. The Import Configuration: Step 2 page opens.



- The red X icon means the item is not available.
- The blue arrow icon means part of the item's configuration will restored.
- The green check mark means the item's full configuration will be restored.

3. Select the items in the list that you want to restore, then mark the **Restore settings from backup** check box to confirm. Click **Restore Configuration**.

System progress is displayed while the changes are made. When the backup is complete, the Administration module's Start Here page displays a list of any errors that occurred during the backup operation. This might include such things as items not found or plugin missing.

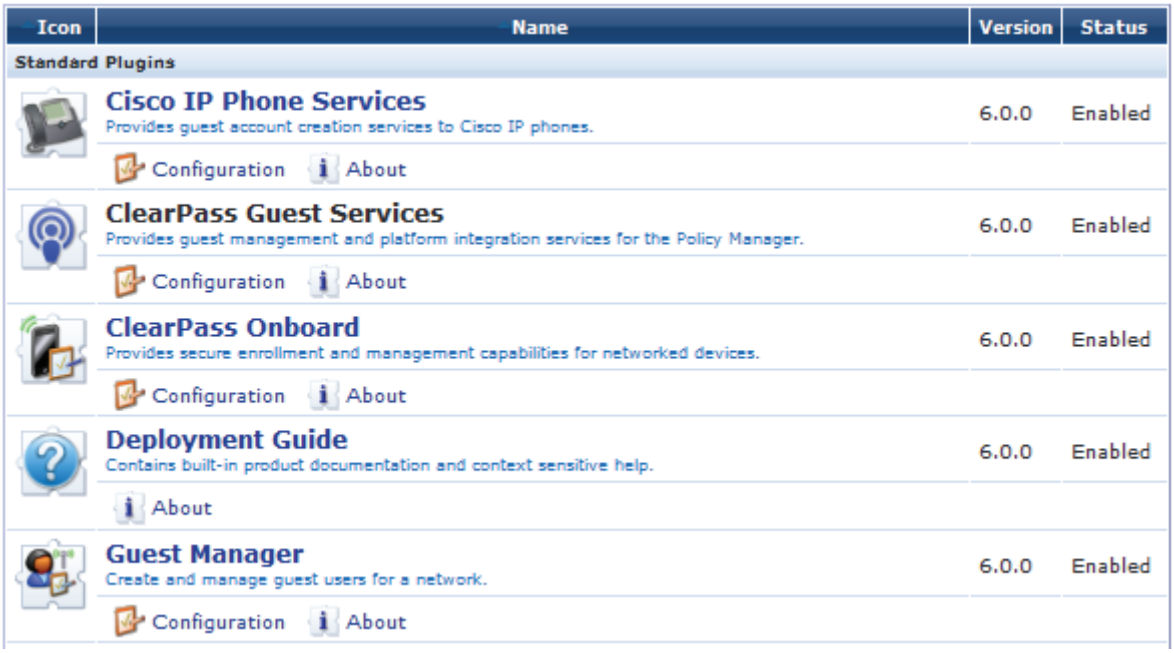
Plugin Manager

Plugins are the software components that fit together to make your Web application. The Available Plugins list shows all the plugins currently included in your application. It lets you view information about each plugin and configure some aspects of most plugins. You can click a plugin's name to go directly to that area of the application—for example, clicking the name of the **SMTP Services** plugin opens the Customize Email Receipt page in the Configuration module.

Viewing Available Plugins

To access the Available Plugins list, navigate to **Administration > Plugin Manager.** The Available Plugins page opens.Plugins are listed by category and include:

- Standard application plugins—Provide corresponding functionality for interactive use by operators
- Kernel plugins—Provide the basic framework for the application
- Operator Login plugins—Control access to the Web application
- Skin plugins—Provide the style for the application's visual appearance
- Translation plugins—Provide translated user interface text and messages in various languages

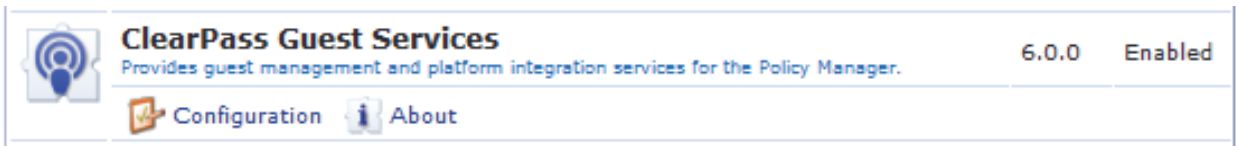


The **About** link displays information about the plugin, including the installation date and update date. The About page for the Kernel plugin also includes links to verify the integrity of all plugin files or perform an application check.

Click a plugin's **Configuration** link to view or modify its settings. See "Configuring Plugins" for details about the configuration settings.

Configuring Plugins

You can configure most standard, kernel, and skin plugins. Skin plugins can also be enabled or disabled, letting you choose which skin to use. To view or change a plugin's configuration, go to the **Administration > Plugin Manager** page and click the **List Available Plugins** command.



To view or change the configuration settings for a plugin, click the plugin's **Configuration** link. The **Configure Plugin** form shows the current configuration settings for a plugin, and allows you to make changes to these settings.

To undo any changes to the plugin's configuration, click the plugin's **Restore default configuration** link. The plugin's configuration is restored to the factory default settings.

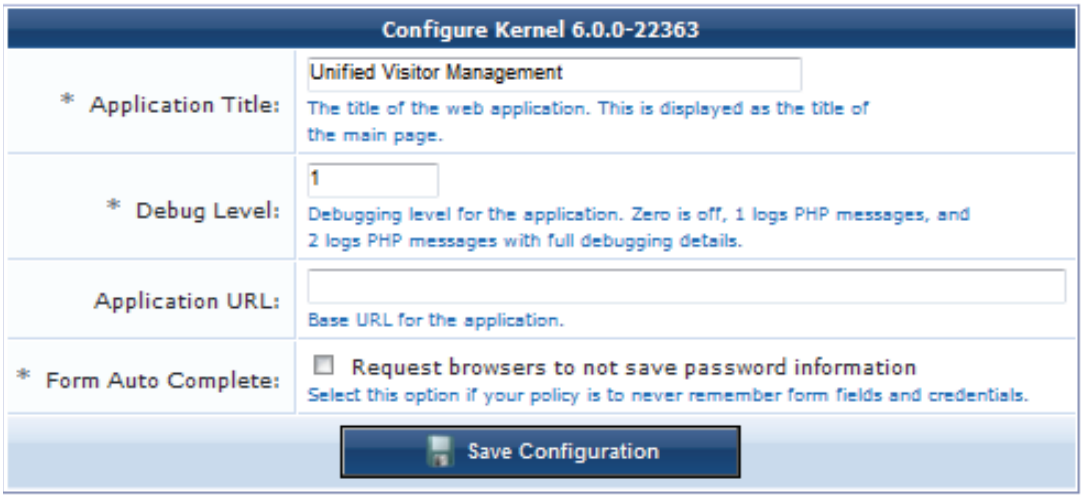
In most cases, plugin configuration settings do not need to be modified directly. Use the customization options available elsewhere in the application to make configuration changes.

For more information about plugin configuration:

- **AirGroup Services**—See "Creating AirGroup Administrators"
- **Kernel**—See "Configuring the Kernel Plugin"
- **Aruba ClearPass Skin—**See "Configuring the Aruba ClearPass Skin Plugin"
- **Guest Manager**—See "Default Settings for Account Creation" in the "Configuration"chapter
- **SMS Services**—See "Sending an SMS"
- **SMTP Services**—See "Email Receipts and SMTP Services" in the "Configuration"chapter
- **MAC Authentication**—See "MAC Authentication in ClearPass Guest" in the "Guest Manager"chapter

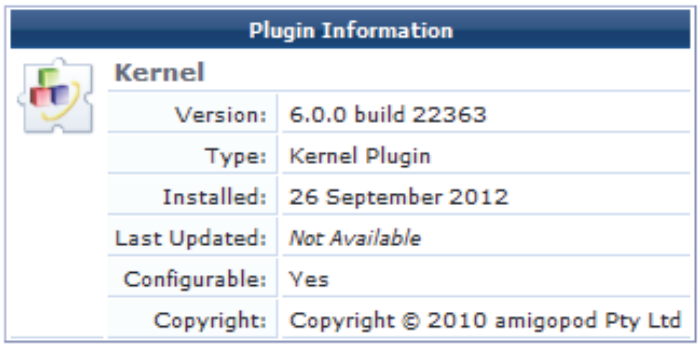
Configuring the Kernel Plugin

The Kernel Plugin provides the basic framework for the application. Settings you can configure for this plugin include the application title, the debugging level, the base URL, and the application URL, and autocomplete.



1. To change the application's title, enter the new name in the **Application Title** field (for example, your company name) to display that text as the title of your Web application. Click **Save Configuration**.

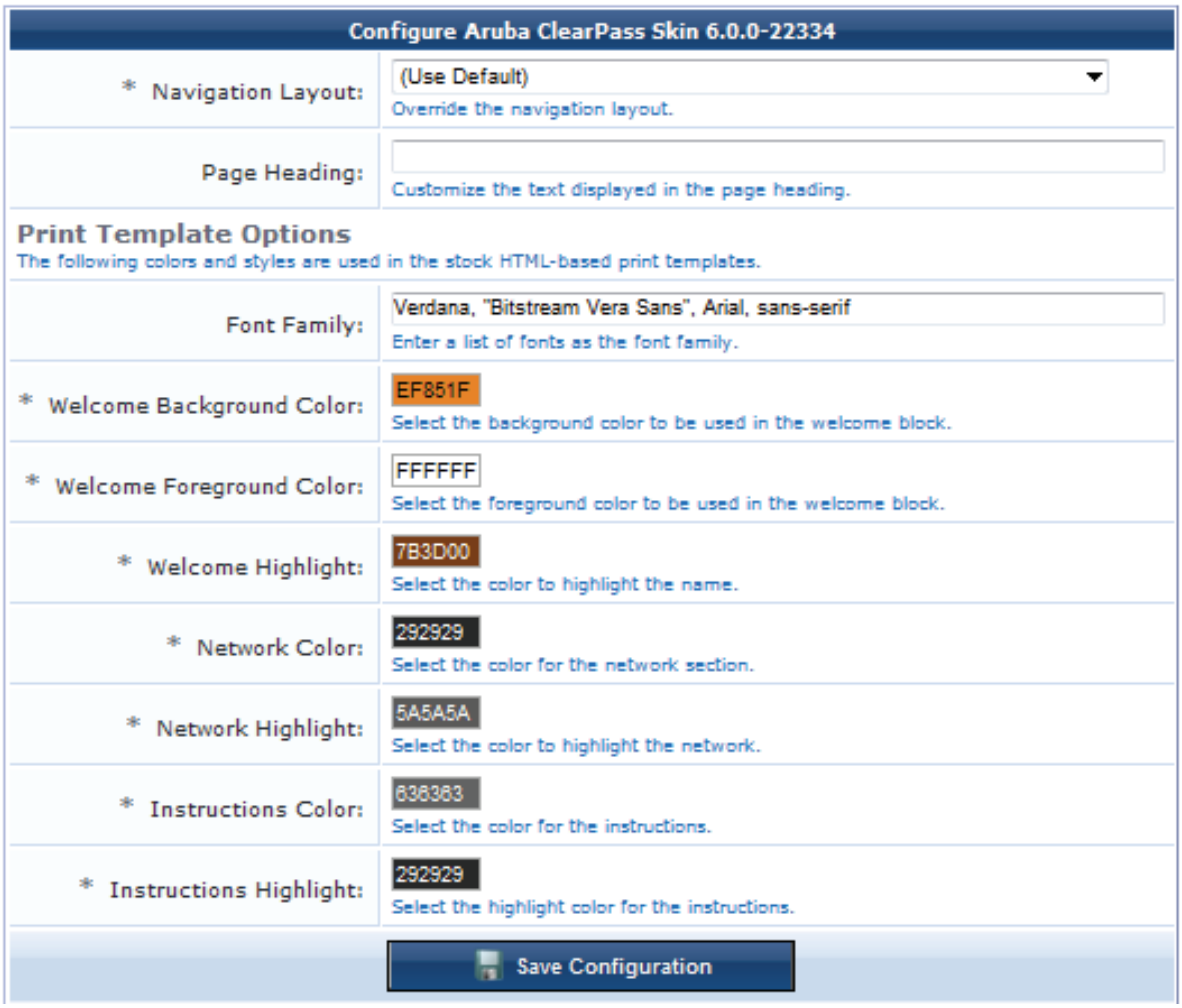
2. The Kernel plugin's **Debug Level** and **Application URL** options should not be modified unless you are instructed to do so by Aruba support.
3. To turn off autocomplete on forms, mark the check box in the **Form Auto Complete** row. This disables credentials caching.
4. To restore the plugin's configuration to the original settings, click the **Restore default configuration** link below the form. A message alerts you that the change cannot be undone, and a comparison of the current and default settings highlights the changes that will be made.
5. Review the differences between the current settings and the default configuration. To commit the change to the default settings. click the **Restore Default Configuration** link.



Configuring the Aruba ClearPass Skin Plugin

A Web application's skin determines its visual style—the colors, menus, and graphics. You can use either the standard Aruba ClearPass skin plugin, a blank plugin if you are providing your own complete HTML page, or custom skin plugins that let you configure the colors, navigation, logo, and icons.

1. To modify the standard Aruba ClearPass skin plugin, click its **Configuration** link on the Available Plugins page.



2. The default navigation layout is “expanded.” To change the behavior of the navigation menu, click the **Navigation Layout** drop-down list and select a different expansion level for menu items.
3. The **Page Heading** field allows you to enter additional heading text to be displayed at the very top of the page.
4. In the **Font Family** row, to change the font, delete the current selection and enter the list of fonts to use.
5. To change a color in any of the color fields, click the color sample box to open the color picker. Set a color, then click **Select** in the color picker for that item. Repeat for each color you want to change.
6. Click Save Configuration.

The default skin used by the ClearPass Guest application is the one that is enabled in the Plugin Manager. To change the default skin globally, navigate to the plugin list and click the **Enable** link for the skin you would like to use as the default. When you install a new custom skin, it is automatically enabled and becomes the default skin. If your application’s appearance does not automatically change, find the custom plugin in the list, click **Configure**, and click its **Enable** link. If you prefer to use the standard Aruba ClearPass skin, navigate to it in the Available Plugins list and click its **Enable** link.

The default skin is displayed on all visitor pages, and on the login page if no other skin is specified for it. However; you can override this for a particular operator profile, an individual operator, or give the login page a different appearance than the rest of the application. You can also specify a skin for guest self-registration pages.

- To use a different skin for a particular operator profile, see “Creating an Operator Profile” in the “Operator Logins” chapter.

- To use a different skin for an individual operator login, see "Local Operator Authentication" in the "Operator Logins"chapter.
- To have the login page use a different skin than the rest of the application, see "Operator Logins Configuration" in the "Operator Logins"chapter.
- To specify a skin for a customized guest self-registration page, see "Configuring Basic Properties for Self-Registration" in the "Configuration"chapter.

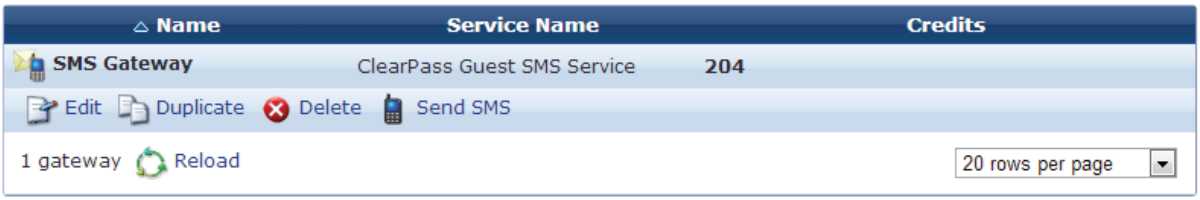
SMS Services

With SMS Services, you can configure ClearPass Guest to send SMS messages to guests. You can use SMS to send a customized guest account receipt to your guest's mobile phone. You can also use SMS Services to send an SMS from your Web browser. To use the SMS features, you must have the SMS Services plugin installed.

Viewing SMS Gateways

To view the list of SMS gateways:

1. Go to **Administration > SMS Services > SMS Gateways**. The SMS Gateways list view opens. This list displays the name and available credits for any currently defined SMS gateways.



2. To work with a gateway, click its row in the list. The gateway's row expands to include the **Edit, Duplicate**, **Delete**, **Make Default**, and **Send SMS** options.
   - **Edit**—To make changes to the gateway in this row, click its Edit link. The Edit SMS Gateway form opens. See "Editing an SMS Gateway" on page 253.
   - **Duplicate**—To make a copy of the gateway to use as a base for a new gateway, click the Duplicate link. A new gateway is added to the list with the name "Copy of <original gateway>".
   - **Delete**—To remove the gateway from the list, click this link. You are asked to confirm the deletion. Click **OK** to delete the gateway.
   - **Make Default**—Click this link in a gateway's row to make it the default gateway for SMS messages.
   - **Send SMS**—Click this link in a gateway's row to send an SMS message via that gateway. The row expands to include the New SMS Message form, where you can enter the recipient's mobile phone number and the message text, then send the message.
3. To add a carrier to the list, click the **Create** tab above the form. The SMS SMTP Carrier Editor form is added at the top of the list. See "Creating a New SMS Gateway" on page 251.

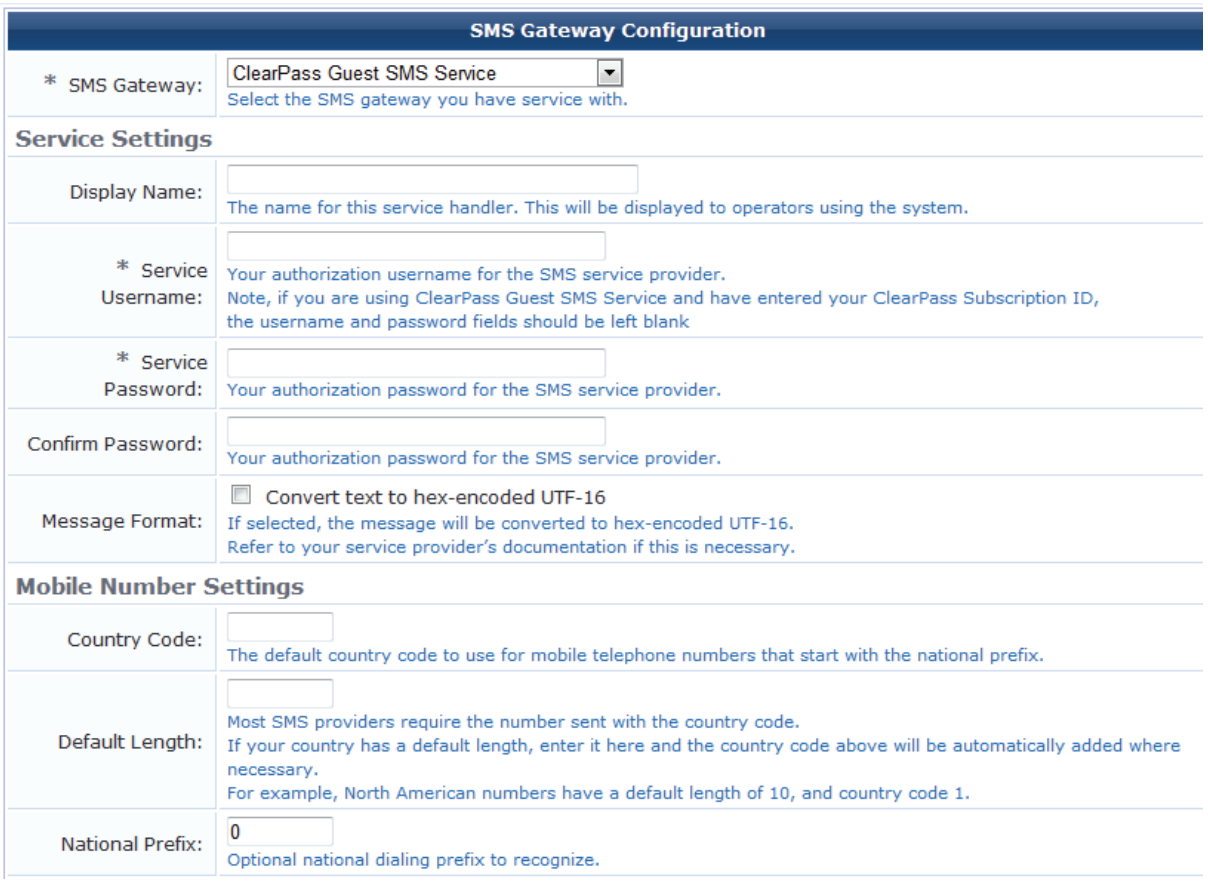
Creating a New SMS Gateway

An SMS gateway is automatically created and added to the SMS Gateways list when you enter your subscription ID in ClearPass Policy Manager at **Administration > Agents and Software Updates > Software Updates**. You can also use ClearPass Guest to create an SMS gateway.

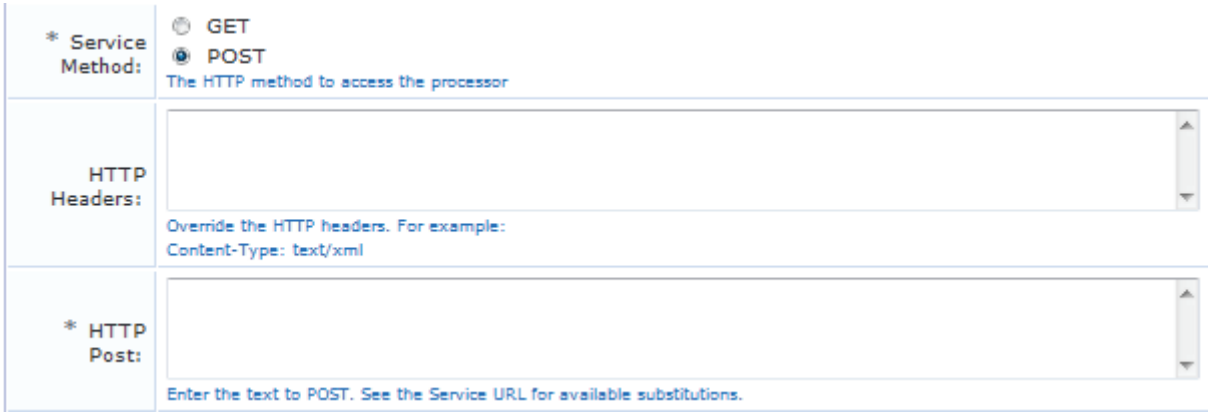
To create a new SMS gateway:

1. Go to **Administration > SMS Services > SMS Gateways**. The SMS Gateways list view opens.
2. Click the **Create new SMS gateway** link in the upper right corner. The **SMS Gateway Configuration** form opens.

   The first part of the form includes the Service Settings and Mobile Number Settings areas.



3. In the **SMS Gateway** field, if you choose **Custom HTTP Handler** from the drop-down list, you may specify the HTTP method to use. The form expands to include options for configuring that gateway type, and the **Service Method** row includes the GET and POST options. When you select the **POST** option, the HTTP Headers and HTTP Post rows are added. You can use the text fields in these rows to override HTTP headers and enter the text to post.



4. In the **Service Username** and **Service Password** fields, you may enter your authorization username and password for your SMS service provider.

   If you are using ClearPass Guest SMS Service and have entered your ClearPass subscription ID in the Software Updates page of ClearPass Policy Manager's Administration module, leave these fields blank. The subscription ID is automatically used as the username and password for the ClearPass SMS Service.
5. In the **Message Format** row, if needed for custom SMS handlers, you can specify that the message format should be converted to hex-encoded UTF-16 (Unicode).
6. In the **Mobile Settings** area, if your country uses a national dialing prefix such as "0", you may enter this in the **National Prefix** row. When sending an SMS to a number that starts with the national dialing prefix, the prefix is removed and replaced with the country code instead.

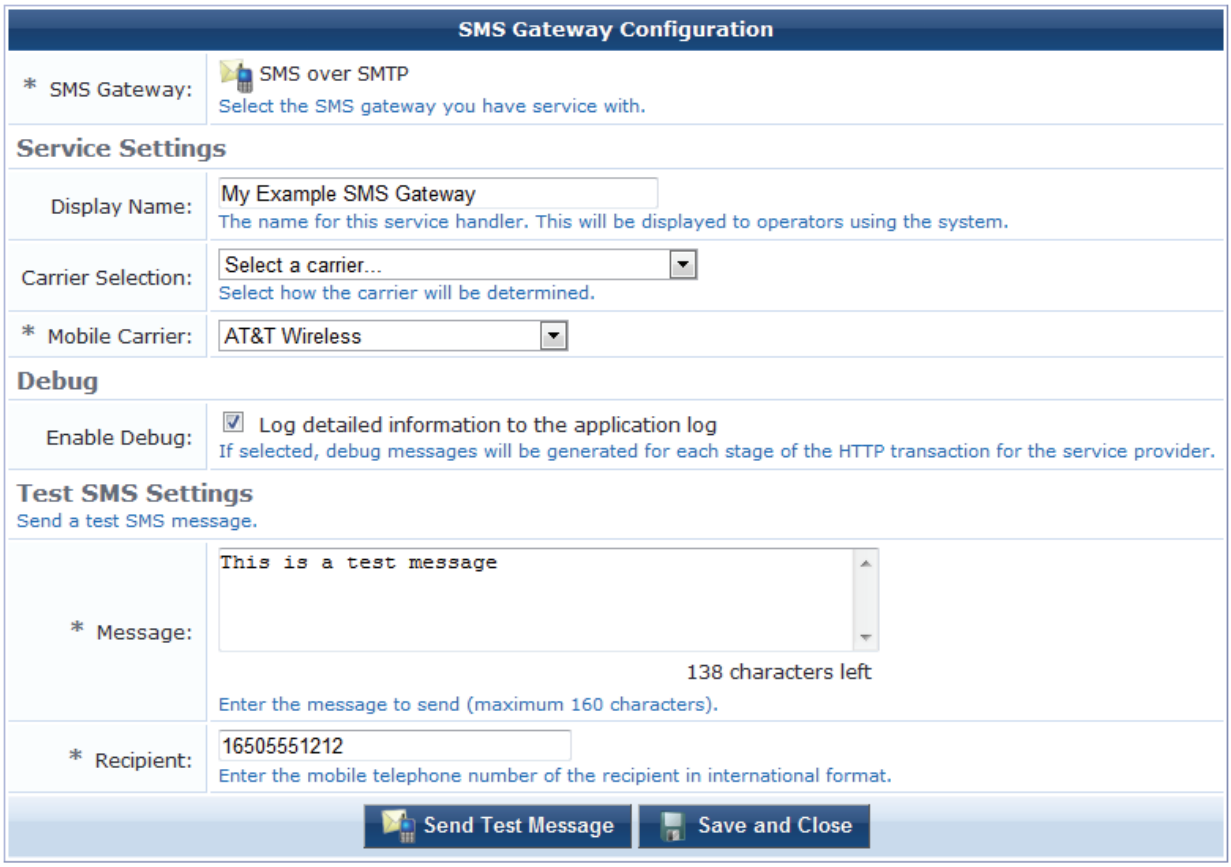
The second part of the form includes the **Connection Settings**, **Debug**, **Credits**, and **Test SMS Settings** areas.

Complete the fields with the appropriate information, then click either **Send Test Message** or **Save and Close**. The new configuration settings will take effect immediately.

Editing an SMS Gateway

To edit an SMS gateway:

1. Go to **Administration > SMS Services > SMS Gateways**. The SMS Gateways list view opens.
2. Click the gateway's row in the list. The row expands to include the Edit SMS Gateway form for the existing gateway.



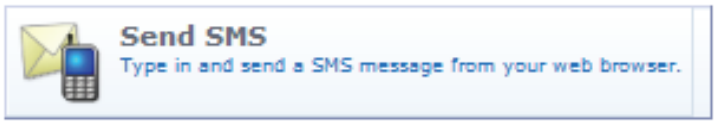
3. The **SMS Gateway** field displays the gateway service that was selected when the gateway was created. This cannot be edited after creation.
4. In the **Service Settings** area, you may edit the **Display Name.**
5. When you duplicate a gateway, the Carrier Selection configuration options are included. In the **Carrier Selection** drop-down list, choose one of the following options:
   - **Registration form will have the visitor_carrier field**—The visitor will supply the carrier information when they register.
   - **Select a carrier**—The form includes the Mobile Carrier field. Choose the carrier from the **Mobile Carrier** drop-down list.
   - **Configure Carrier Settings**—The form expands to include configuration options for the carrier:
     - **SMS Address**—You may choose to use a template to determine the email address, or to use a fixed address.
     - **Address Template or Address**—If you chose to use a template to determine the address, the next field is **Address Template**. Enter an example email address that will be used as the pattern for the address format. If you chose to use a fixed email address, the next field is **Address**. Enter the email address to which all messages will be sent.

- **Number Format**—Choose a country code requirement option from this drop-down list. The available options are **Use the visitor's value**, **Always include the country code**, or **Never include the country code**.
- **Subject Line**—You may enter text for the message's subject line. This field supports Smarty template syntax. For a Smarty template syntax description, See "Smarty Template Syntax" in the "Reference" chapter.

6. To log detailed information in the application log for each stage of the HTTP transaction, mark the check box in the **Enable Debug** row.
7. To verify the configuration, enter a test message in the **Message** field and enter the test recipient's mobile phone number in the **Recipient** field, then click **Send Test Message**.
8. When all fields are completed appropriately, click Save and Close. The SMS Gateways list is updated with the changes.

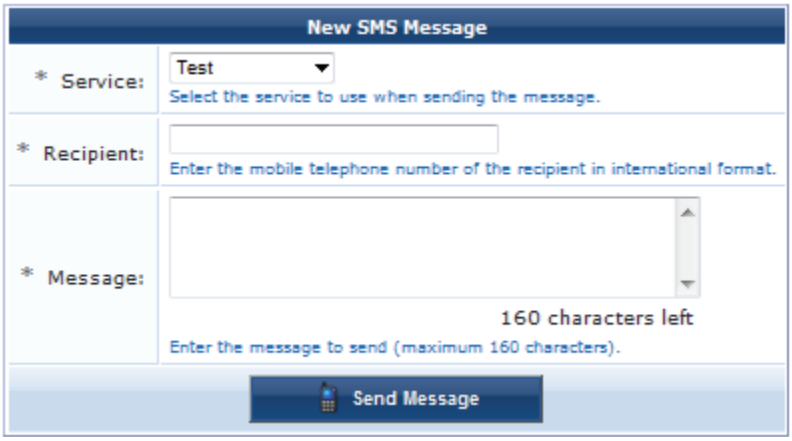
Sending an SMS

You are able to send an SMS message if the system has been configured to allow this.



To send an SMS message:

1. Go to **Administration > SMS Services > Send SMS**. The **New SMS Message** form opens.



2. Complete the form by typing in the SMS message and entering the mobile phone number that you are sending the SMS to. The maximum length for the message is 160 characters. If multiple services are available, you may also choose the service to use when sending the message.
3. Click **Send Message**.

About SMS Credits

Each SMS message sent consumes one credit.

To determine the number of remaining SMS credits, navigate to the **Administration > SMS Gateways** window. The **Credits Available** field indicates the number of remaining SMS credits for your account. This value is determined once the first message has been sent, and is updated after sending each message.

When credits are running low, a warning message is emailed to the administrator group. The email address is determined by looking up all local operators with the special IT Administrators operator profile, and using any configured email address for those operators.

Up to three messages will be sent:

- A low-credit warning is sent once the “Credits Available” value reaches the warning threshold (the default value is 50).
- A second low-credit warning is sent once the “Credits Available” value reaches half the warning threshold.
- A final message is sent once the “Credits Available” value reaches zero.

NOTE

To adjust the warning threshold, set the Credit Warning value in the configuration for the SMS Services Plugin.

About SMS Guest Account Receipts

You can send SMS receipts for guest accounts that are created using either sponsored guest access or self-provisioned guest access. This is convenient in situations where the visitor may not be physically present to receive a printed receipt.

ClearPass Guest may be configured to automatically send SMS receipts to visitors, or to send receipts only on demand.

To manually send an SMS receipt:

1. Navigate to the **Guest > List Accounts** and click to expand the row of the guest to whom you want to send a receipt.
2. Click **Print** to display the Account Details view, then click the **Send SMS receipt** link. The SMS Receipt form opens. Use the fields on this form to enter the service to use, the recipient’s mobile phone number, the mobile carrier, and the message text.

When using guest self-registration, SMS Delivery options are available for the receipt page actions; See **“Editing SMS Delivery of Guest Receipts”** in the “Configuration”chapter for full details.

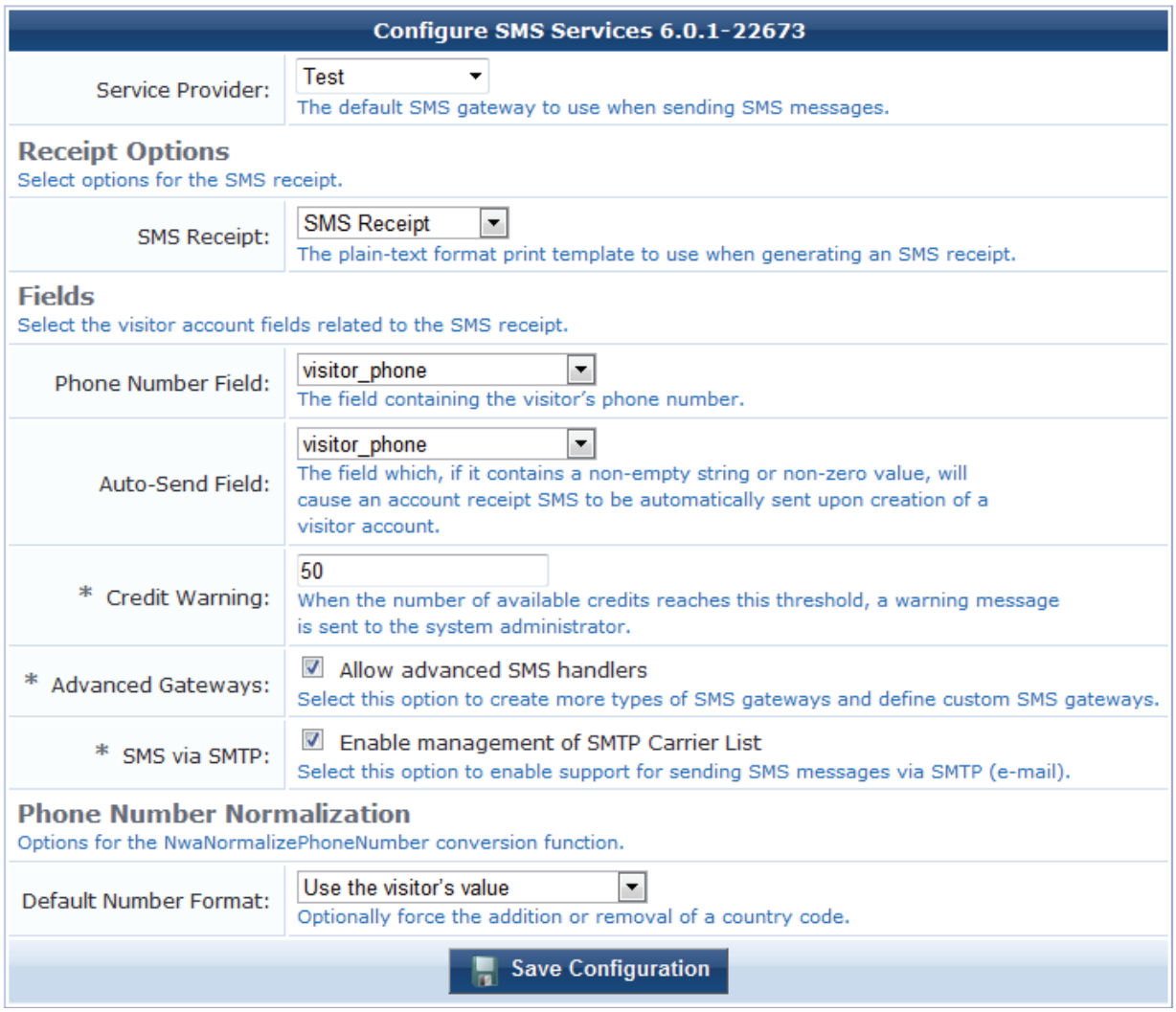
SMS Receipt Options

The SMS Services plugin configuration allows you to configure options related to SMS receipts. These settings can be viewed and modified using the Plugin Manager.

To view or configure SMS services and receipt options:

1. Go to **Administration > Plugin Manager**. The Available Plugins list opens.
2. Scroll to the **SMS Services** row and click its **Configuration** link. The Configure SMS Services form opens.

Figure 33 *Configure SMS Services Plugin*



SMS Receipt – Select the print template to be used when an SMS receipt is created. The print template used for the receipt must be in plain text format.

- **Phone Number Field** – Select which guest account field contains the guest's mobile telephone number. This field is used to determine the SMS recipient address.
- **Auto-Send Field** – Select a guest account field which, if set to a non-empty string or non-zero value, will trigger an automatic SMS when the guest account is created or updated. The **auto-send** field can be used to create an "opt-in" facility for guests. Use a check box for the **auto_send_sms** field and add it to the **create_user** form, or a guest self-registration instance, and SMS messages will be sent to the specified phone number only if the check box has been selected.
- **Credit Warning** – When SMS credits get below this threshold, the system will send a warning to the system administrator.
- **Advanced Gateways** – Select this option to configure SMS gateways from multiple SMS providers. ClearPass Guest SMS services support SMS USA, SMS Worldwide, AQL, Sirocco, Tempos 21 and Upside Wireless SMS gateways.
- **SMS via SMTP** – Select this option to allow visitor account receipt messages to be sent in an email using the defined SMTP server.

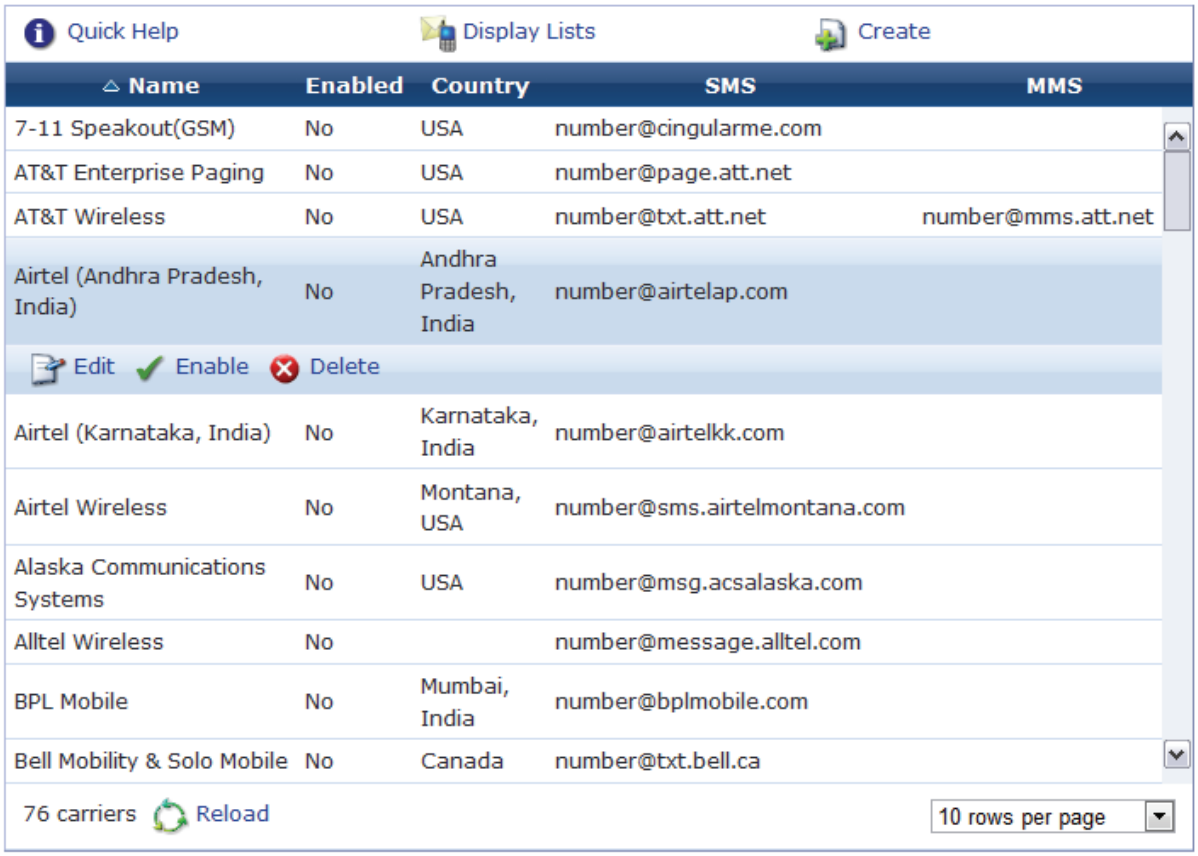
- **Phone Number Normalization** – The phone number normalization process translates phone strings that are entered in various formats into a single standard format. Click this drop-down list and select one of the following options:
  - **Use the visitors value**: When you select this option, the SMS gateway will always send the SMS message using the phone number and country code entered by the visitor.
  - **Always include the country code**: When you select this option, the SMS gateway will always send the SMS message using the global country code and default phone number length specified in the **Default Country Code** and **Default Phone Length** fields. For example, consider an Australian mobile phone number with a default number length of 9 plus a leading zero, and a country code of 61. If you selected the **Always include the country code** option, the Australian mobile number *0412345678* would normalize to *+61412345678* in the internationalized format.
  - **Never include the country code**: When you select this option, any country code specified by the visitor is removed before the SMS message is sent.

Working with the SMTP Carrier List

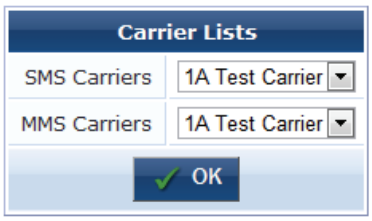
You can manage the list of SMTP carriers that are included in the Mobile Carrier drop-down list on the SMS Services > SMS Gateways > Edit SMS Gateway form.

To view or work with the SMTP carrier list:

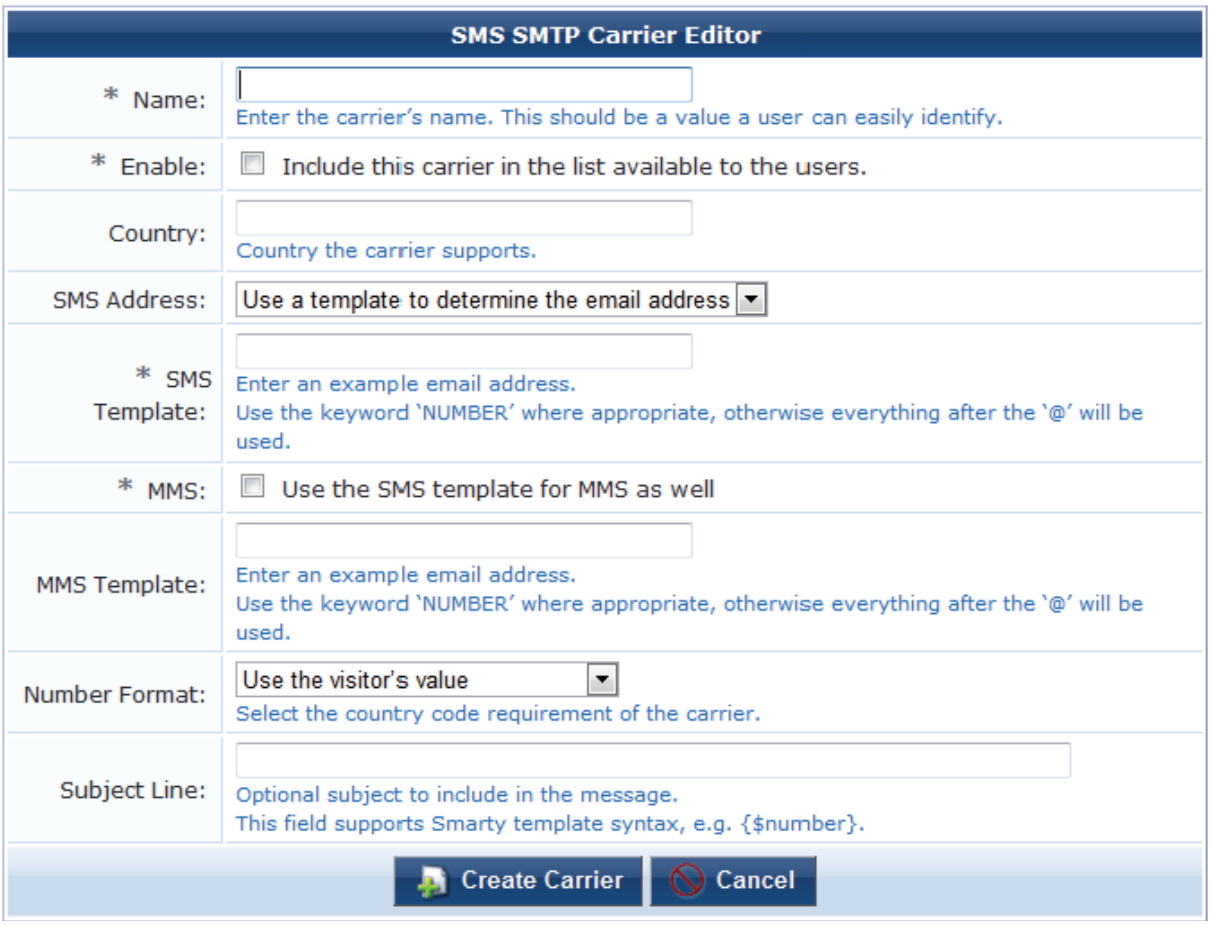
1. Go to **Administration > SMS Services > SMTP Carriers**. The SMS SMTP Carrier List view opens. The carriers in this list are the ones that are included in the Mobile Carrier drop-down list on the SMS Services > SMS Gateways > Edit SMS Gateway form.



2. To filter the list, click the **Display Lists** tab above the form. The form expands to include the Carrier Lists options. You may specify an SMS or MMS carrier.



3. To enable, disable, or delete a carrier, click the carrier in the list. The carrier's row expands to include the **Edit**, **Enable** or **Disable**, and **Delete** options.
4. The procedures for adding and for editing a carrier are the same.
   - To add a carrier to the list, click the **Create** tab above the form. The SMS SMTP Carrier Editor form is added at the top of the list.
   - To edit an existing carrier, click the carrier's row in the list. The row expands to include the SMS SMTP Carrier Editor form for that carrier.



5. In the **Name** field, enter the carrier's name. If there is more than one format of the carrier company's name, use the format the public most readily identifies with the carrier service.
6. To include the carrier in the list of choices for users, mark the **Enable** check box.
7. (Optional) In the **Country** field, enter the country where the carrier's service is offered. If appropriate, you may also indicate an area within the country, such as a city, county, or state.

8. In the **SMS Address** drop-down list, choose one of the following options:
   - **Use a template to determine the email address**— When this option is chosen, the next field's name becomes **SMS Template**.
   - **Use a fixed email address**—Use this option if all SMS messages are to be sent to the same address. When this option is chosen, the next field's name becomes **Address**.
9. Configure the option you chose in the previous step:
   - If you chose **Use a template**... in the **SMS Address** field, enter an example email address in the **SMS Template** field. This provides the pattern for the address format.
     - The default is to substitute the number for all characters preceding the @ sign, producing the pattern **number@address**.
     - Some carriers require additional characters before or after the phone number. In this case, use the keyword string **NUMBER** in the pattern to limit the substitution to just the phone number portion of the address—for example, NUMBER.msg@carrier.example.com, or username+NUMBER@mymail.com
   - If you chose **Use a fixed email address** n the **SMS Template** field, use the **Address** field to enter the email address to which all SMS messages will be sent.
10. In the **MMS** row:
    - To use the SMS template for MMS messages, mark the check box in this row. The SMS Address configuration will be applied to MMS messages, and the MMS Template row is removed from the form.
    - To use an MMS template for MMS messages, leave this check box unmarked.
11. If you will use an MMS template for MMS messages, enter an example email address in the **MMS Template** field. This provides the pattern for the address format.
12. In the **Number Format** row, choose a country code requirement option from the drop-down list. The available options are **Use the visitor's value**, **Always include the country code**, or **Never include the country code**.
13. (Optional) In the **Subject Line** field, you may enter text for the message's subject line. This field supports Smarty template syntax, and the number is available as {$number}.

    For example:

    ```
    Sent to: {$number} in the year {'Y'|date}
    ```

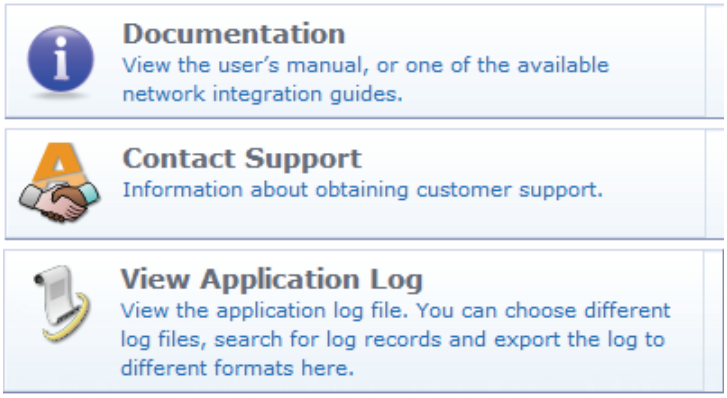
    ...would produce:

    Sent to: 15555551234 in the year 2012

    For a Smarty template syntax description, See "Smarty Template Syntax" in the "Reference" chapter.
14. When all fields are completed appropriately, click **Edit Carrier** or **Create Carrier**. The SMS SMTP Carrier List is updated with the changes.

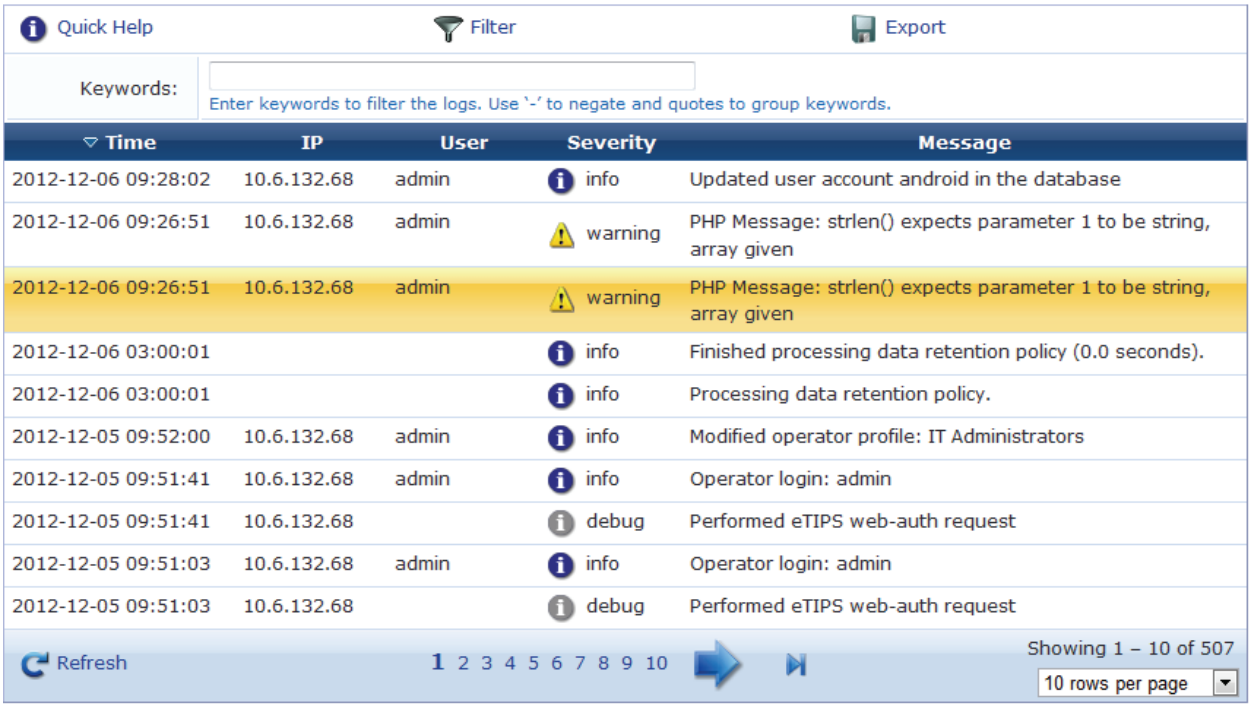
Support Services

The **Administration > Support Services** page provides links to ClearPass Guest documentation, the application log, and Aruba Customer Support contact information.



Viewing the Application Log

To view events and messages generated by the application, go to **Administration > Support > Application Log**. The Application Log view opens.

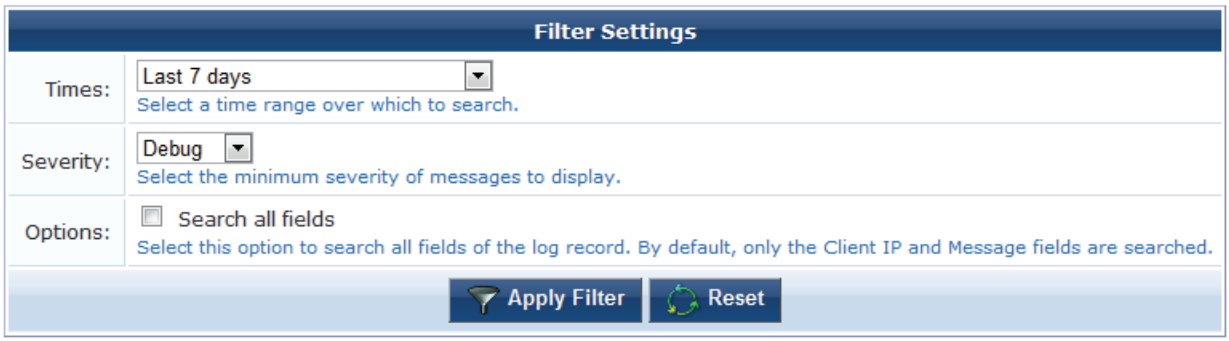


To view in-depth information about an event, click the event's row. The form expands to show details. Click the event's row again to close it.

To search for a particular log record, use the **Keywords** field above the table to enter search terms. You can use the hyphen character (-) in front of a keyword to exclude items, and you can use quotes (“ “) to group words as a key phrase.

The Application Log lists the events, messages, and configuration changes for the past seven days. To view events and messages for a different period, or to limit the search items:

1. Click the **Filter** tab. The Filter Settings form opens.



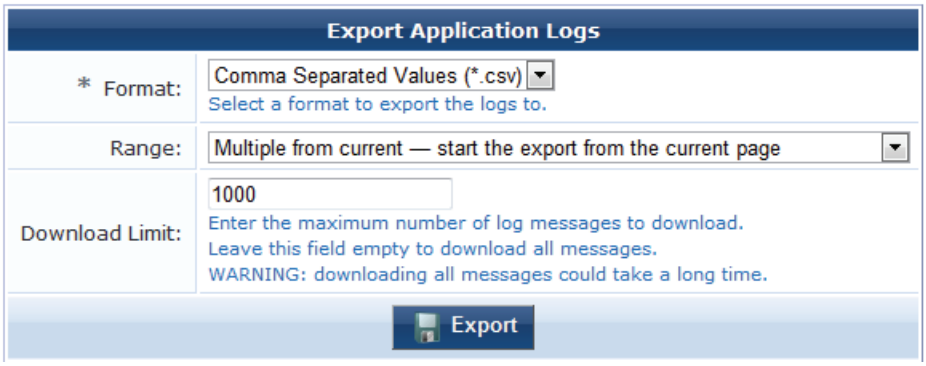
2. You can use the **Times** drop-down list to specify a time period to filter for.
3. The **Severity** drop-down list lets you limit the range of severity to search for:
   - **Error**—Returns Error items
   - **Warning**—Returns Error and Warning items
   - **Info**—Returns Error, Warning, and Info items
   - **Debug**—Returns Error, Warning, Info, and Debug items
4. By default, only the Client IP and Message fields are searched. To search all fields, mark the check box in the **Options** row.

Events are stored in the Application Log for seven days by default. To review a record of significant runtime events prior to the last seven days, you can use the Audit Viewer in ClearPass Policy Manager’s Monitoring module.

Exporting the Application Log

To save the log in other formats:

1. Click the **Export** tab. The Export Application Logs form opens.



2. In the **Format** drop-down list, choose the format you want the file saved as. The available formats are Comma-Separated Values (.CSV), HTML document (.html), Tab-Separated Values (.tsv), Text file (.txt), and XML document (.xml).
3. In the **Range** drop-down list, select the range of pages to save. Options include the current page only, all pages starting from the current page, or all pages starting from the first page that matched any keyword or filter criteria you entered.
4. If you entered a range of pages in the Range drop-down list, the form expands to include the **Download Limit** row.
5. Click **Export**. You are given the option to open the file, save it to your Downloads folder (the default), or save it to another location.

Contacting Support

To view contact information for Aruba Support, go to **Administration > Support > Contact Support**. The Contact Support page opens. This page provides the following information:

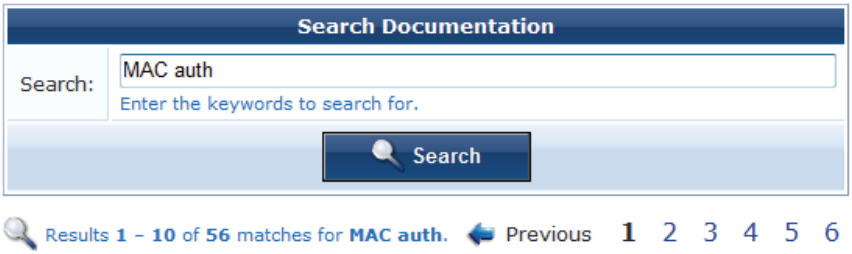
- Toll-free telephone numbers for North American support
- A link to contact Aruba Support by email
- A link to Aruba's online Contact Support page, which includes telephone numbers and other contact information for over 30 countries

Viewing Documentation

To view ClearPass Guest documentation:

1. Go to **Administration > Support > Documentation**. The Documentation page opens.

2. To view this Deployment Guide in your browser, click **Browse Documentation**. The document opens in a separate browser tab.
3. To search the Deployment Guide, click **Search Documentation**. The Search Documentation form opens.
4. In the **Search** field, enter keywords for the subject. You can enter a string of keywords, phrases enclosed in quotes (“my phrase”), and you can exclude a term by preceding it with a minus sign (-).
5. Click **Search**. The search engine returns a list of results.



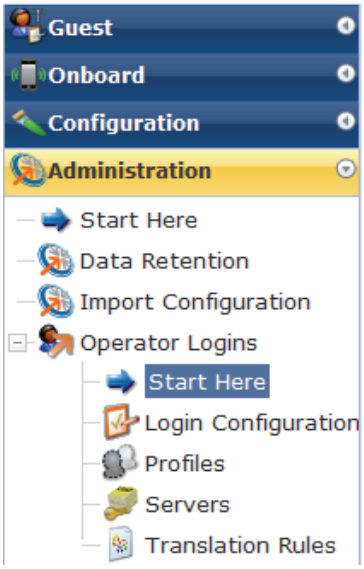
6. Click a result link. The online help opens in a separate browser tab with the destination displayed.

An operator is a company's staff member who is able to log in to ClearPass Guest. Different operators may have different roles that can be specified with an operator profile. These profiles might be to administer the ClearPass Guest network, manage guests, or run reports.

Operators may be defined locally in ClearPass Guest, or externally in an LDAP directory server.

Accessing Operator Logins

To access ClearPass Guest's operator login features, click the **Administration** link in the left navigation, then click **Operator Logins**.



About Operator Logins

ClearPass Guest supports role-based access control through the use of operator profiles. Each operator using the application is assigned a profile which determines the actions that the operator may perform, as well as global settings such as the look and feel of the user interface.

Your profile may only allow you to create guest accounts, or your profile might allow you to create guest accounts as well as print reports. What your profile permits is determined by the network administrator.

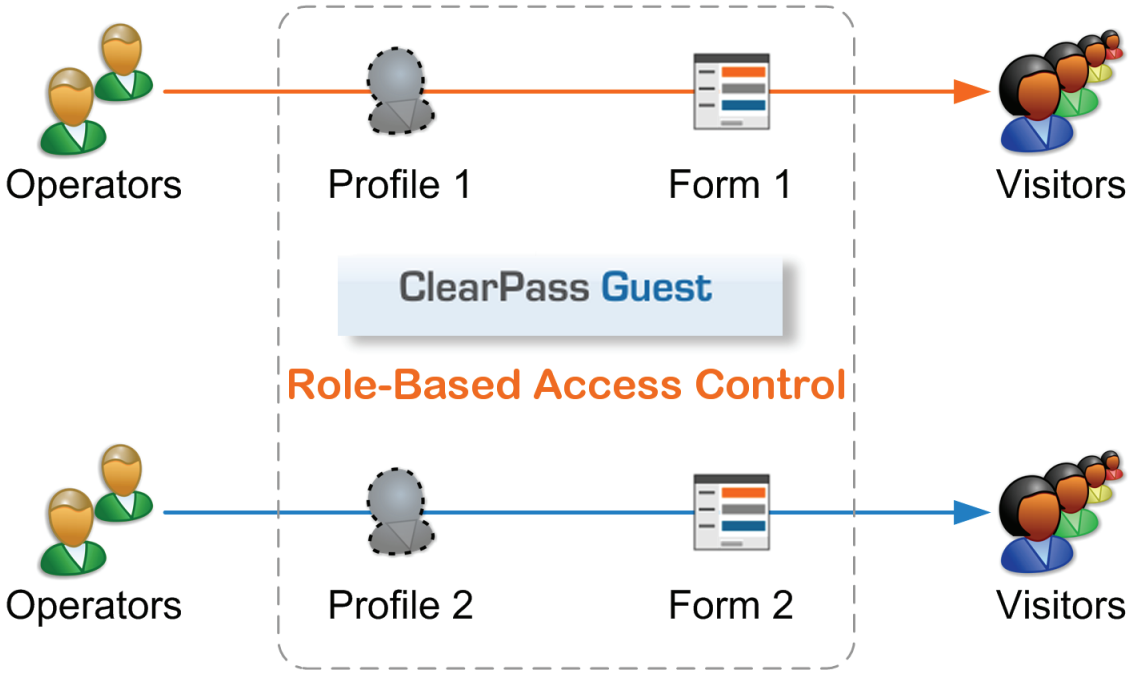
Two types of operator logins are supported: local operators and operators who are defined externally in your company's directory server. Both types of operators use the same login screen.

Role-Based Access Control for Multiple Operator Profiles

Using the operator profile editor, the forms and views used in the application may be customized for a specific operator profile, which enables advanced behaviors to be implemented as part of the role-based access control model.

This process is shown in the following diagram.

Figure 34 *Operator profiles and visitor access control*



See "About Operator Logins" for details on configuring different forms and views for operator profiles.

Operator Profiles

An operator profile determines what actions an operator is permitted to take when using ClearPass Guest.

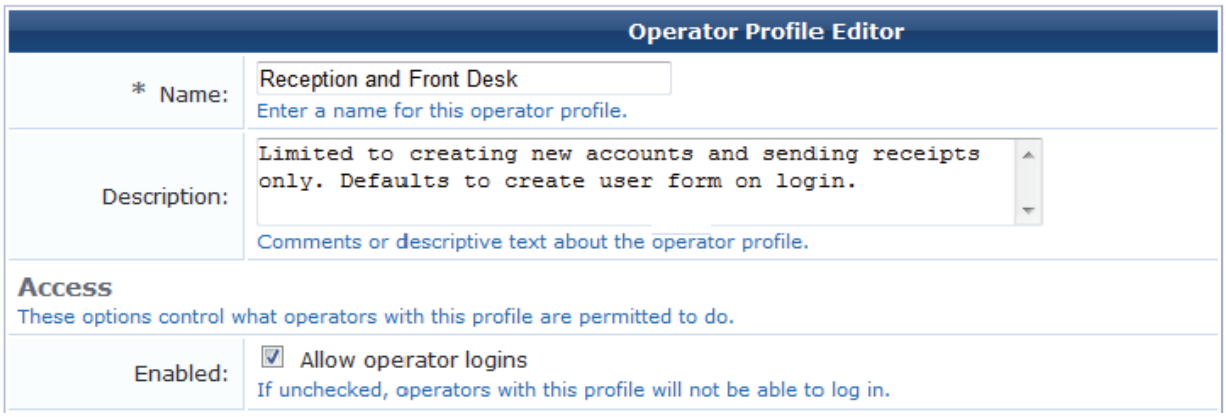
Some of the settings in an operator profile may be overridden in a specific operator's account settings. These customized settings will take precedence over the default values defined in the operator profile.

To define new operator profiles and to make changes to existing operator profiles, go to **Administrator > Operator Logins > Profiles**. The Operator Profiles page opens with the profiles list displayed.

Creating an Operator Profile

Click the **Create Operator Profile** link to create a new operator profile.

The **Operator Profile Editor** form is displayed. This form has several sections, which are described in more detail below.

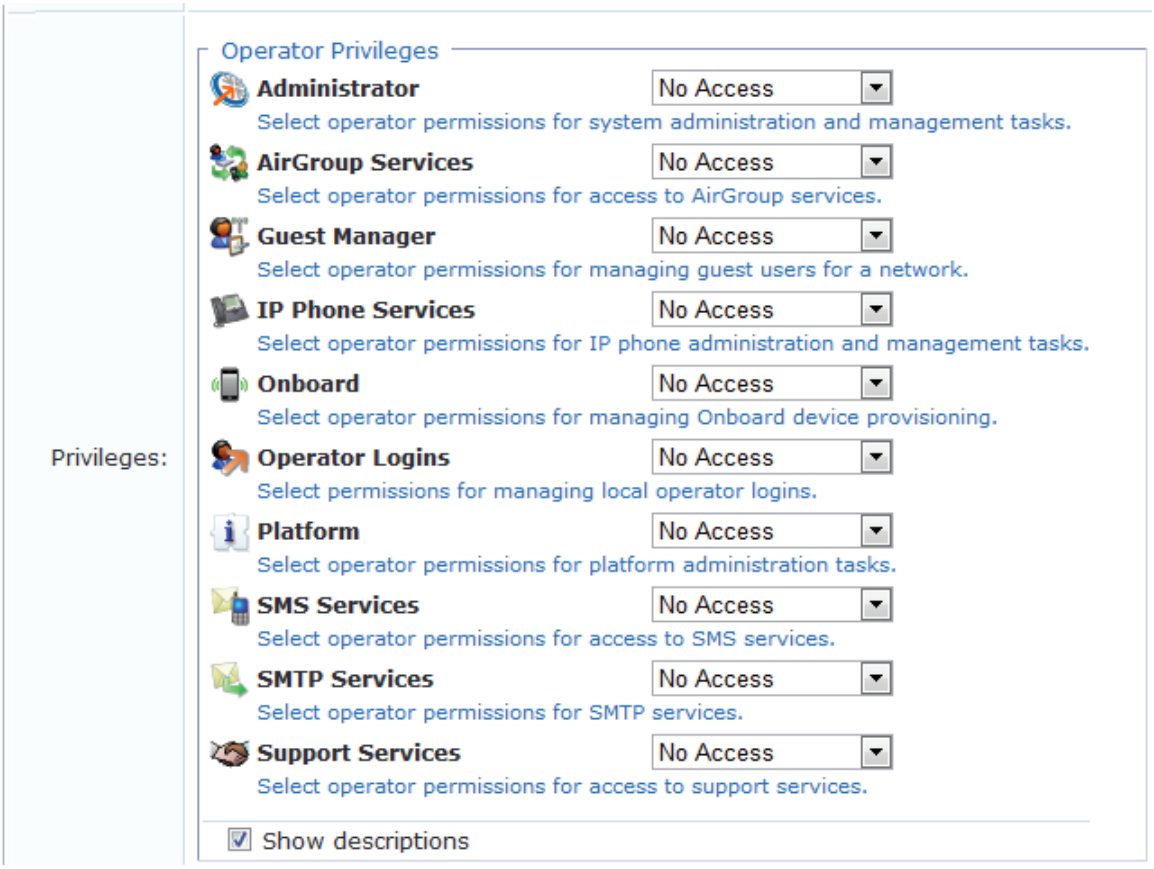


The fields in the first area of the form identify the operator profile and capture any optional information:

1. You must enter a name for this profile in the **Name** field.
2. (Optional) You may enter additional information about the profile in the **Description** field.

The fields in the **Access** area of the form define permissions for the operator profile:

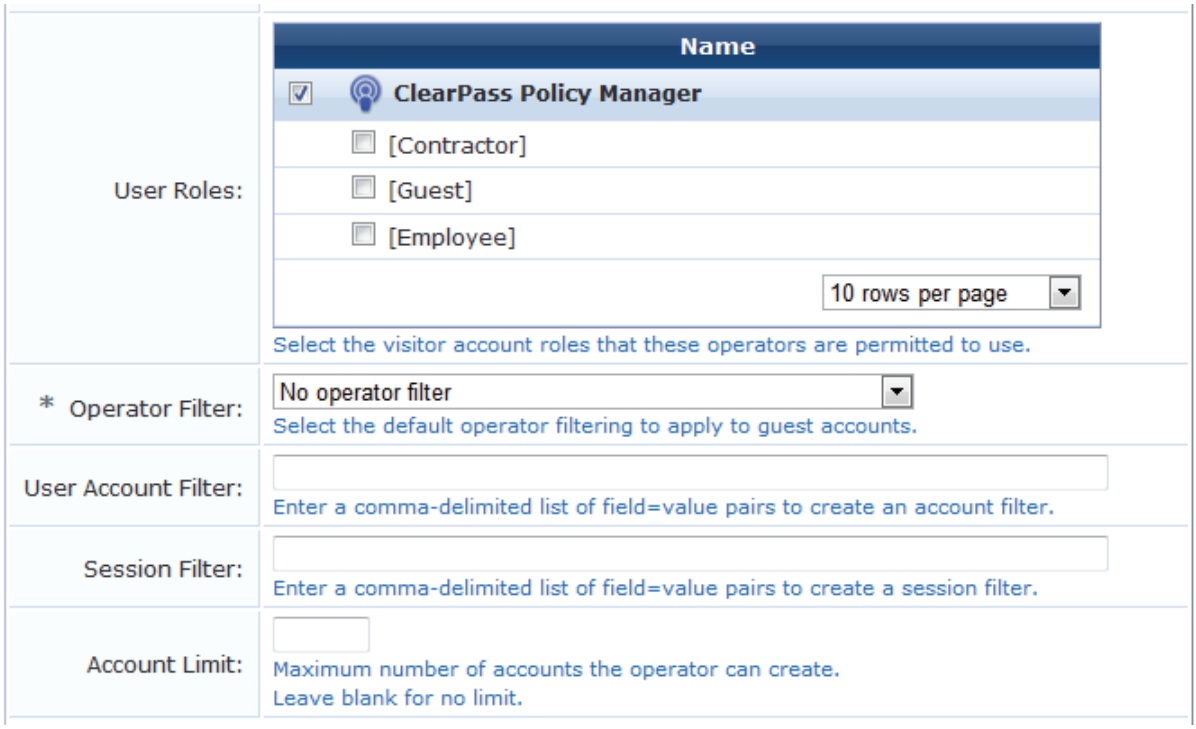
1. In the **Enabled** row, the **Allow Operator Logins** check box is selected by default. To disable a profile, unmark the **Allow Operator Logins** check box. If a profile is disabled, any operators with that profile will be unable to log in to the system. This may be useful when performing system maintenance tasks.
2. In the **Operator Privileges** area, use the drop-down lists to select the appropriate permissions for this operator profile.



For each permission, you may grant **No Access**, **Read Only Access**, **Full Access**, or **Custom** access. The default in all cases is No Access. This means that you must select the appropriate privileges in order for the profile to work. See "Operator Profile Privileges" for details about the available access levels for each privilege.

If you choose the **Custom** setting for an item, the form expands to include additional privileges specific to that item.

3. The **User Roles** list allows you to specify which user databases and roles the operator will be able to access.



   If one or more roles are selected, then only those roles will be available for the operator to select from when creating a new guest account. The guest account list is also filtered to show only guest accounts with these roles.

   If a database is selected in the User Roles list, but no roles within that database are selected, then all roles defined in the database will be available. This is the default option.

4. The **Operator Filter** may be set to limit the types of accounts that can be viewed by operators. Options include: default, no operator filter, only show accounts created by the operator, and only show accounts created by operators within their profile.
5. The **User Account Filter** and **Session Filter** fields are optional, and allow you to create and configure these filtering options:
   - The **User Account Filter** field lets you create a persistent filter applied to the user account list. For example, this feature is useful in large deployments where an operator only wants to have a filtered view of some accounts. To create an account filter, enter a comma-delimited list of field-value pairs. Supported operators are described below.
   - The **Session Filter** field lets you create a filter for only that session. To create a session filter, enter a comma-delimited list of field-value pairs. Supported operators are described below.

The user can enter a simple substring to match a portion of the username or any other fields that are configured for search, and may include the following operators:

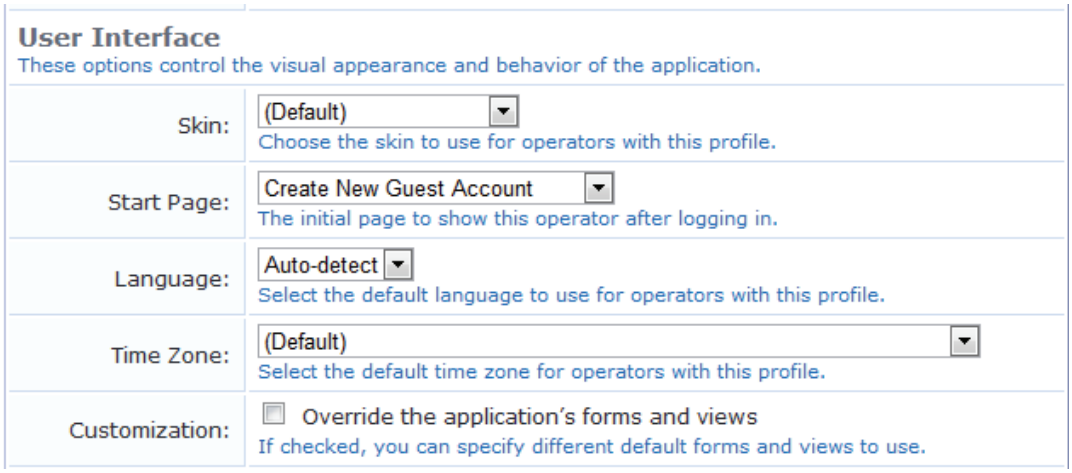
Table 20 *Operators supported in filters*

Operator	Meaning	Additional Information
=	is equal to	You may search for multiple values when using the equality (=) or inequality !=) operators. To specify multiple values, list them separated by the pipe character (|). For example, specifying the filter "role_id=2|3, custom_field=Value" restricts the user accounts displayed to those with role IDs 2 and 3 (Guest and Employee), and with the field named "custom_field" set to "Value".
!=	is not equal to	
>	is greater than	
>=	is greater than or equal to	
<	is less than	
<=	is less than or equal to	
~	matches the regular expression	
!~	does not match the regular expression	

6. In the **Account Limit** row, you can enter a number to specify the maximum number of accounts an operator can create. Note that disabled accounts are included in the account limit. To set no limit, leave the Account Limit field blank.

   When you create or edit an AirGroup operator, the value you enter in the Account Limit field specifies the maximum number of devices an AirGroup operator with this profile can create.

Configuring the User Interface



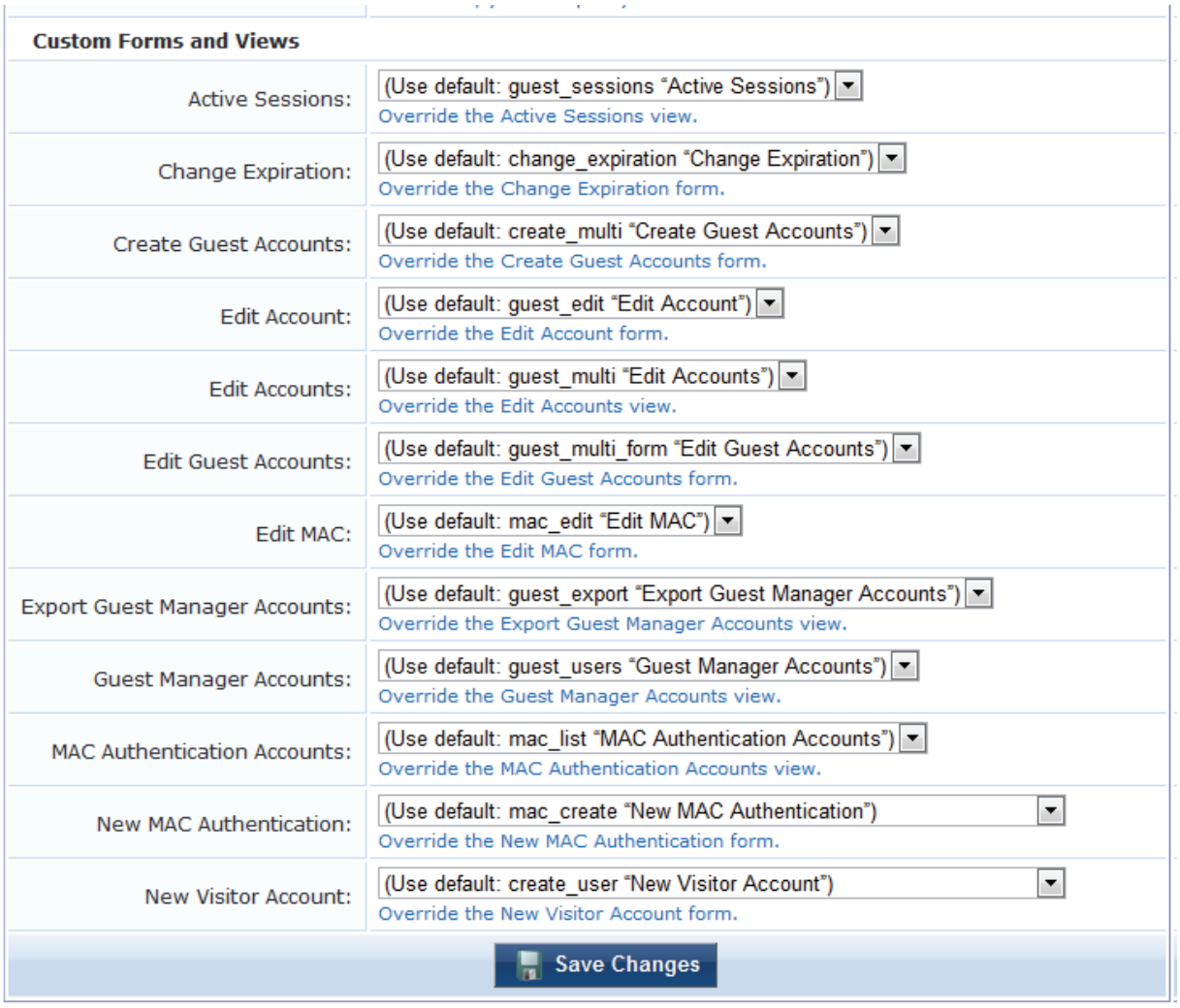
The fields in the **User Interface** area of the form determine elements of the application's visual appearance and behavior that operators with this profile will see. The **Skin**, **Start Page**, **Language**, and **Time Zone** options specify the defaults to use for operators with this profile. Individual operator logins may have different settings, which will be used instead of the values specified in the operator profile. For information on specifying options at the individual operator level, see "Local Operator Authentication".

1. (Optional) In the **Skin** row, the **Default** setting indicates that the skin plugin currently marked as enabled in the Plugin Manager will be used. To have a different skin displayed for users with this operator profile, choose one of the available skins from the drop-down list. For more information on skins, see "Configuring the Aruba ClearPass Skin Plugin" in the "Administration"chapter.

2. (Optional) In the **Start Page** row, the **Default** setting indicates that the application's standard Home page will be the first page displayed after login. To have a different start page displayed to users with this operator profile, choose a page from the drop-down list. For example, if a profile is designed for users who do only certain tasks, you might want the application to open at the module where those tasks are performed.
3. (Optional) In the **Language** row, the default setting is **Auto-detect**. This lets the application determine the operator's language preference from their local system settings. To specify a particular language to use in the application, choose the language from the drop-down list.
4. (Optional) In the **Time Zone** row, the **Default** setting indicates that the operator's time zone will default to the system's currently configured time zone. You can use the drop-down list to specify a particular time zone.
5. (Optional) In the **Customization** row, you can choose to override the application's default forms and views. For more information, see the next section, "Customizing Forms and Views".

Customizing Forms and Views

You can use the **Customization** option in the Operator Profile Editor to override default forms and views and specify different ones to be used for the operator profile.



To specify that an operator profile should use a different form when creating a new visitor account:

1. (Optional) In the **Customization** row, select the **Override the application's forms and views** check box. The form expands to show the forms and views that can be modified. If alternative forms or views have been created, you may use the drop-down lists to specify which ones to use.
2. When you have selected the custom forms and views to use, click **Save Changes** to complete the creation of the operator profile.

Operator Profile Privileges

The privilege selections available for an operator profile provide you with control over the functionality that is available to operators.

No Access means that the operator will have no access to the particular area of functionality. Options for that functionality will not appear for that operator in the menus.

Read Only Access means that the operator can see the options available but is unable to make any changes to them.

Full Access means that all the options are available to be used by the operator.

Custom access allows you to choose individual permissions within each group. For example, Guest Manager allows you to control access to the following areas:

- Active sessions management
- Viewing historical data for active sessions
- Changing expiration time of guest accounts
- Creating multiple guest accounts
- Creating new guest accounts
- Editing multiple guest accounts
- Exporting guest account data
- Full user control of guest accounts
- Importing guest accounts
- Listing guest accounts
- Managing customization of guest accounts
- Managing print templates
- Removing or disabling guest accounts
- Resetting guest passwords

Refer to the description of each individual operator privilege to determine what the effects of granting that permission will be.

Managing Operator Profiles

Once a profile has been created you are able to view, to edit and to create new profiles. When you click an operator profile entry in the Operator Profiles list, a menu appears that allows you to perform any of the following operations:

- **View/Hide Details** – displays or hides configuration details for the selected operator profile, including the profile name, description, operator login access, and the settings for the defined skin, start page, language and time zone.
- **Edit** – changes the properties of the specified operator profile
- **Delete** – removes the operator profile from the Operator Profiles list
- **Duplicate** – creates a copy of an operator profile

- **Create Operator** – opens the **Create Operator Login** form, allowing you to create a new operator login associated with the selected operator profile.
- **Show Operators** – shows a list of operator login names associated with that operator profile
- **Show Usage** – opens a window in the Operator Profiles list that shows if the profile is in use, and lists any LDAP authentication servers, LDAP translation rules and operator logins associated with that profile. Each entry in this window appears as a link to the form that lets you edit that LDAP or operator login setting.

Configuring AirGroup Operator Device Limit

By default, an AirGroup operator can create up to five personal devices. To change this default:

1. Go to **Administration > Operator Logins > Profiles**, then select the **AirGroup Operator** profile in the list.
2. Click the **Edit** link. The Edit Operator Profile form opens.
3. In the **Account Limit** field, specify an appropriate value. This is the maximum number of personal devices that an operator with this profile can create.
4. Click **Save Changes**.

You can create a set of operator profiles and configure each profile with a different account limit. This makes it easy to assign operator profiles appropriately for small groups, larger groups, or events. To create each profile in the set, duplicate the built-in AirGroup Operator profile, and update the Account Limit field in the new profile.

Local Operator Authentication

ClearPass Policy Manager profiles and ClearPass Guest profiles are different. To create a ClearPass Guest operator login, local users are first defined in ClearPass Policy Manager with a role that matches an operator profile in Guest, then rules are used to map the role to the Guest operator profile.

Creating a New Operator

To create a new operator or administrator for ClearPass Guest or AirGroup, some steps are performed in ClearPass Policy Manager (CPPM), and some steps are performed in ClearPass Guest, as described below:

1. Create an operator profile in ClearPass Guest, or use an existing one. See "Operator Profiles".

   To create AirGroup users, choose either the AirGroup Administrator or AirGroup Operator profile, as appropriate. These profiles are automatically included in ClearPass Guest when the AirGroup Services plugin is installed.
2. Create a CPPM role for the operator: In ClearPass Policy Manager (CPPM), go to **Configuration > Identity > Roles** and create a role that matches the operator profile. Refer to the ClearPass Policy Manager documentation for information on creating the role.
3. Create a local user for the operator: In CPPM, go to **Configuration > Identity > Local Users**. Select the CPPM role defined for the user. Refer to the ClearPass Policy Manager documentation for information on creating the local user.
4. Create a translation rule to map the CPPM role name to the ClearPass Guest operator profile: In ClearPass Guest, go to **Administration > Operator Logins > Translation Rules**.
5. In the **Translation Rules** list, choose the profile, then click its **Edit** link.
6. Edit the fields appropriately to match the CPPM role name to the ClearPass Guest operator profile. See "LDAP Translation Rules".
7. Click **Save Changes**.

External Operator Authentication

Operators defined externally in your company's directory server form the second type of operator. Authentication of the operator is performed using LDAP directory server operations. The attributes stored for an authenticated operator are used to determine what operator profile should be used for that user.

The **Manage Operator Servers** and the **Translation Rules** commands allow you to set up operator logins integrated with a Microsoft Active Directory domain or another LDAP server.

NOTE

The operator management features, such as creating and editing operator logins, apply only to local operator logins defined in ClearPass Guest. You cannot create or edit operator logins using LDAP. Only authentication is supported.

Manage LDAP Operator Authentication Servers

ClearPass Guest supports a flexible authentication mechanism that can be readily adapted to any LDAP server's method of authenticating users by name. There are built-in defaults for Microsoft Active Directory servers, POSIX-compliant directory servers, and RADIUS servers.

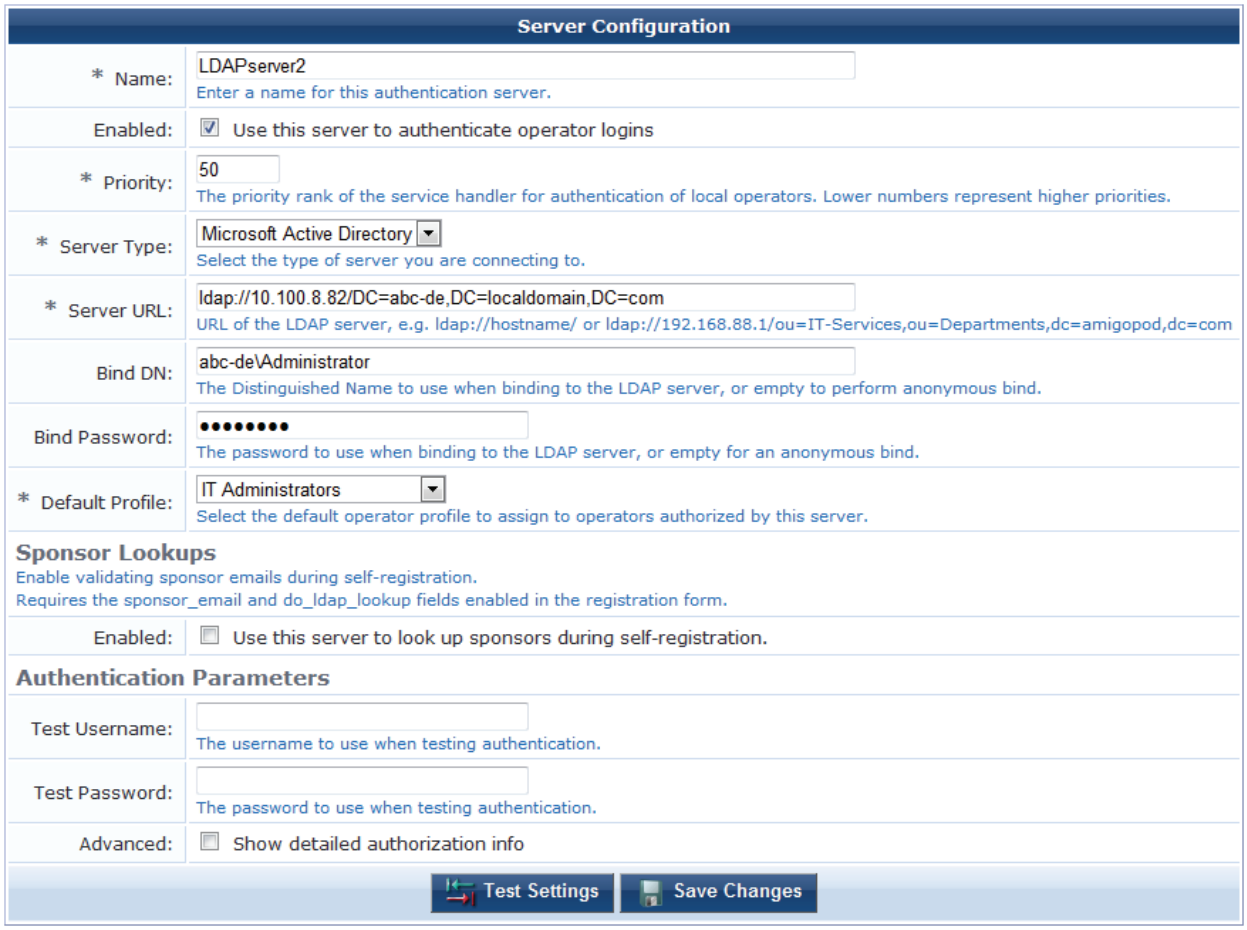
When an operator attempts to log in, each LDAP server that is enabled for authentication is checked, in order of priority from lowest to highest.

Once a server is found that can authenticate the operator's identity (typically with a username and password), the LDAP server is queried for the attributes associated with the user account.

These LDAP attributes are then translated to operator attributes using the rules defined in the LDAP translation rules. In particular, an operator profile will be assigned to the authenticated user with this process, which controls what that user is permitted to do.

Creating an LDAP Server

To create an LDAP server, go to **Administration > Operator Logins > Servers**, then click the **Create new LDAP server** link in the upper-right corner. The **Server Configuration** form opens.



To specify a basic LDAP server connection (hostname and optional port number), use a Server URL of the form **ldap://hostname/** or **ldap://hostname:port/**. See "Advanced LDAP URL Syntax" for more details about the types of LDAP URL you may specify.

In the top area of the form, select the **Enabled** option (below the **Name** field) if you want this server to authenticate operator logins.

This form allows you to specify the type of LDAP server your system will use. Click the **Server Type** drop-down list and select one of the following options:

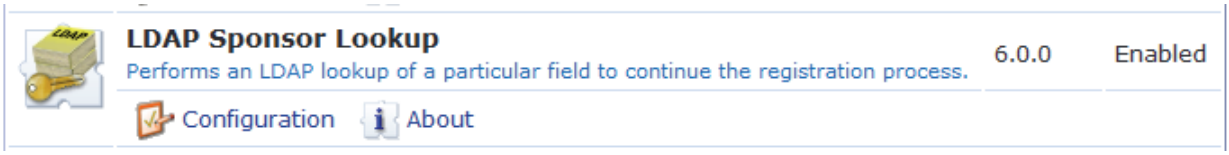
Table 21 *Server Type Parameters*

Server Type	Required Configuration Parameters
Microsoft Active Directory	• **Server URL**: The URL of the LDAP server • **Bind DN**: The password to use when binding to the LDAP server, or empty for an anonymous bind. • **Bind Password**: If your LDAP server does not use anonymous bind, you must supply the required credentials to bind to the directory. (Leave this field blank to use an anonymous bind.) • **Default Profile**: The default operator profile to assign to operators authorized by this LDAP server.

Table 21 *Server Type Parameters (Continued)*

POSIX Compliant:	• **Server URL**: The URL of the LDAP server • **Bind DN**: The password to use when binding to the LDAP server, or empty for an anonymous bind. • **Bind Password**: The password to use when binding to the LDAP server. Leave this field blank to use an anonymous bind. • **Base DN**: The Distinguished Name to use for the LDAP search. • **Default Profile**: The default operator profile to assign to operators authorized by this LDAP server.
Custom	• **Server URL**: The URL of the LDAP server • **Bind DN**: The password to use when binding to the LDAP server, or empty for an anonymous bind. • **Bind Password**: The password to use when binding to the LDAP server. Leave this field blank to use an anonymous bind. • **Base DN**: The Distinguished Name to use for the LDAP search. • **Unique ID**: The name of an LDAP attribute used to match the username. • **Filter**: Additional LDAP filters to use to search for the server. • **Attributes**: List of LDAP attributes to retrieve. Or leave bland to retrieve all attributes (default). • **Default Profile**: The default operator profile to assign to operators authorized by this LDAP server.
RADIUS	• **RADIUS Server**: The hostname or IP address of the RADIUS server. • **Port Number**: The port number of the RADIUS authentication service. • **Shared Secret**: The shared secret for the RADIUS server. • **Authentication Method**: The authentication method that supplies the credentials. • **Default Profile**: The default operator profile to assign to operators authorized by this server.

Select the **Enabled** check box under **Sponsor Lookups** if you want to enable the validation of sponsor emails during self-registration. When this option is selected, this server will look up sponsors during self-registration and double-check the attribute used for emails on the LDAP server. This option requires that the **sponsor_email** and **do_ldap_lookup** fields are enabled in the registration form. This feature requires you to have the LDAP Sponsor Lookup plugin installed. Use the Plugin Manager to verify that this plugin is available.



When you have completed the form, you can check your settings. Use the **Test Username** and **Test Password** fields to supply a username and password for the authentication check, then click the **Test Settings** button. If the authentication is successful, the operator profile assigned to the username will be displayed. If the authentication fails, an error message will be displayed. See "LDAP Operator Server Troubleshooting" for information about common error messages and troubleshooting steps to diagnose the problem.

Click the **Save Changes** button to save this LDAP Server. If the server is marked as enabled, subsequent operator login attempts will use this server for authentication immediately.

Advanced LDAP URL Syntax

For Microsoft Active Directory, the LDAP server connection will use a default distinguished name of the form **dc=domain,dc=com**, where the domain name components are taken from the bind username.

To specify a different organizational unit within the directory, include a distinguished name in the LDAP server URL, using a format such as:

```
ldap://192.168.88.1/ou=IT%20Services,ou=Departments,dc=server,dc=com
```

To specify a secure connection over SSL/TLS, use the prefix **ldaps://**.

To specify the use of LDAP v3, use the prefix **ldap3://**, or **ldap3s://** if you are using LDAP v3 over SSL/TLS.

When Microsoft Active Directory is selected as the Server Type, LDAP v3 is automatically used.

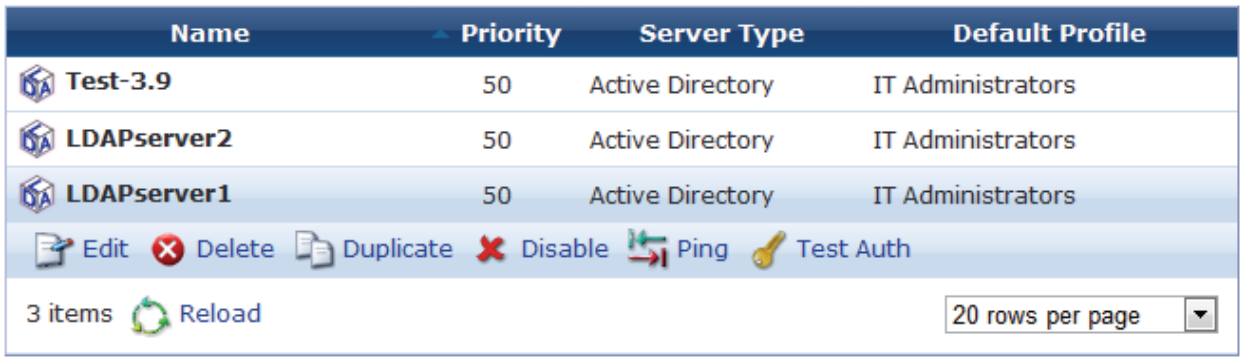
An LDAP v3 URL has the format **ldap://host:port/dn?attributes?scope?filter?extensions**.

- **dn** is the base X.500 distinguished name to use for the search.
- **attributes** is often left empty.
- **scope** may be 'base', 'one' or 'sub'.
- **filter** is an LDAP filter string, for example, (objectclass=*)
- **extensions** is an optional list of name=value pairs.

Refer to RFC 2255 for further details.

Viewing the LDAP Server List

Once you have defined one or more LDAP servers, those servers will appear in the LDAP server list on the **Administration > Operator Logins > Servers** page.



Select any of the LDAP servers in the list to display options to perform the following actions on the selected server:

- **Edit**—Opens the Server Configuration form, where you can make changes to the properties of the LDAP server.
- **Delete**—Removes the server from the LDAP server list.
- **Duplicate**—Creates a copy of an LDAP server. You can then click the Edit link to open the Server Configuration form and use original server's properties as a template for creating a new server.
- **Disable**—Temporarily disables a server while retaining its entry the server list.
- **Enable**—Reenables a disabled LDAP server.
- **Ping**—Sends a ping message (echo request) to the LDAP server to verify connectivity between the LDAP server and the ClearPass Guest server.

- **Test Auth**—Adds a **Test Operator Login** area in the LDAP servers form that allows you to test authentication of operator login values.
- **Test Lookup**—Adds a **Test Operator Lookup** form in the LDAP servers list that allows you to look up sponsor names. This option is only available if sponsor lookup has been enabled for the server on the Edit Authentication Server page.

LDAP Operator Server Troubleshooting

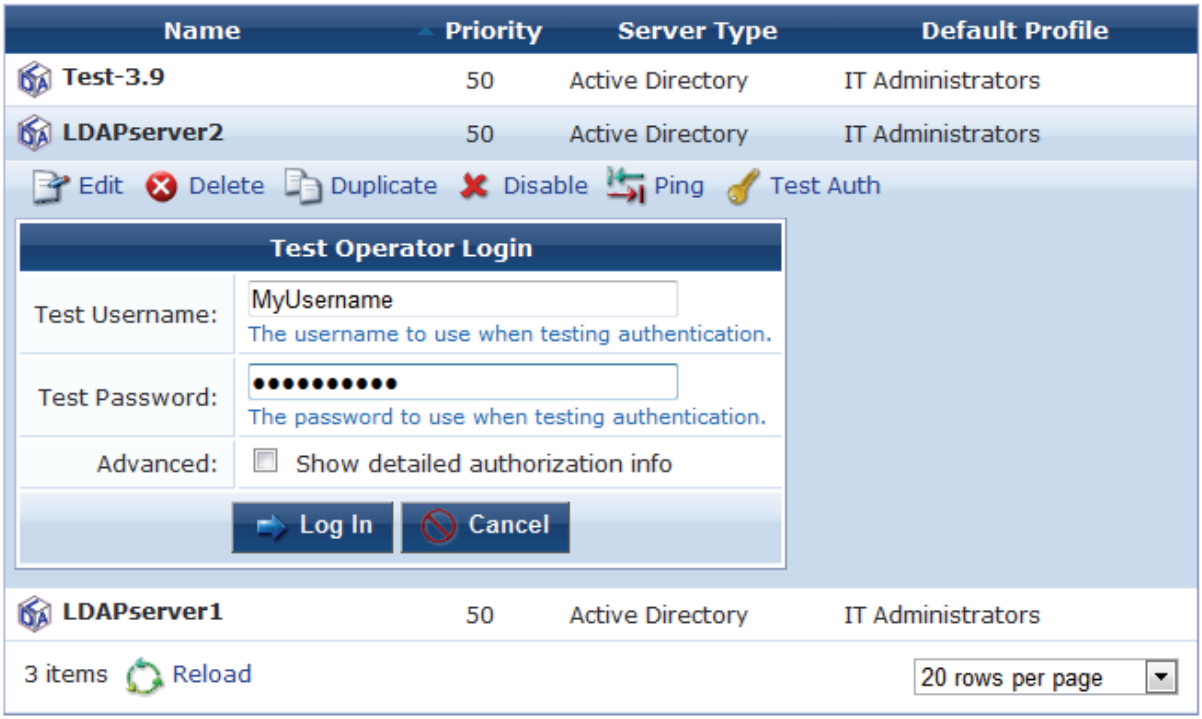
You can use the LDAP Operator Servers list to troubleshoot network connectivity, operator authentication, and to look up operator usernames.

Testing Connectivity

To test network connectivity between an LDAP server and the ClearPass Guest server, click the **Ping** link in the server's row. The results of the test appear below the server entry in the LDAP server table.

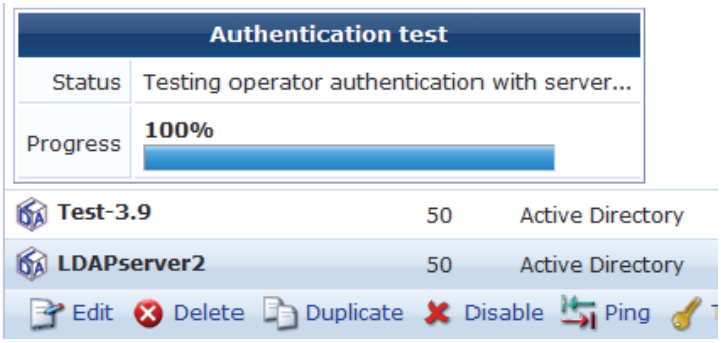
Testing Operator Login Authentication

1. To test authentication of operator login values, select a server name in the LDAP Server table, then click the **Test Auth** link. The Test Operator Login area is added to the page.



2. Enter an operator username and password for the LDAP Server.
3. (Optional) Click the **Advanced** check box to display detailed authorization information for the specified operator.

4. Click **Log In** to attempt to authenticate the LDAP server, or click **Cancel** to cancel the test. The Authentication Test area is added above the server names to indicate the test's progress.



You can also verify operator authentication when you create a new LDAP server configuration using the **Test Settings** button on the **LDAP Configuration** form (See "Creating an LDAP Server" for a description).

Looking Up Sponsor Names

This option is only available if sponsor lookup has been enabled for the server on the Edit Authentication Server page.

1. To look up a sponsor, select a server name in the LDAP Server table, then click the **Test Lookup** link. The Test Operator Lookup area is added to the LDAP servers list.
2. In the **Lookup** field, enter a lookup value. This can be an exact username, or you can include wildcards.If you use wildcards, the search might return multiple values.
3. In the **Search Mode** field, use the drop-down list to specify whether to search for an exact match or use wildcard values.
4. (Optional) Click the **Advanced** check box to display detailed authorization information for the specified sponsor.
5. Click **Search Directory** to attempt to find sponsor names that match the lookup values, or click **Cancel** to cancel the test. The Authentication Test area is added above the server names to indicate the search's progress.

Troubleshooting Error Messages

The error messages in the following table can be used to diagnose error messages such as: "LDAP Bind failed: Invalid credentials (80090308: LdapErr: DSID-0C090334, comment: AcceptSecurityContext error, data **525**, vece), bind DN was: …"

Table 22 *LDAP Error Messages*

Error Data	Reason
525	User not found
52e	Invalid credentials (password is incorrect)
530	Not permitted to log on at this time
531	Not permitted to log on at this workstation
532	Password has expired

Table 22 *LDAP Error Messages (Continued)*

533	Account is disabled
701	Account has expired
773	User must reset password
775	User account is locked

Other items to consider when troubleshooting LDAP connection problems:

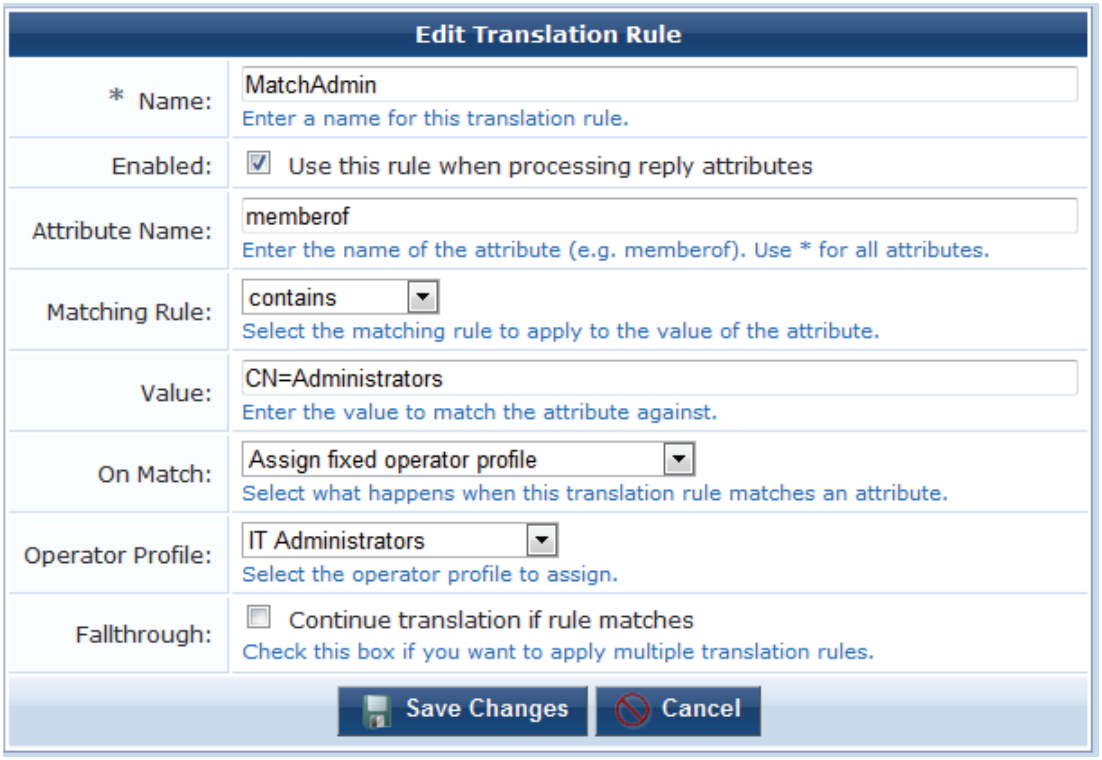
- **Verify that you are using the correct LDAP version** – use ldap:// for version 2 and ldap3:// to specify LDAP version 3.
- **Verify that you are using an SSL/TLS connection** – use ldaps:// or ldap3s:// as the prefix of the Server URL.
- **Verify that the Bind DN is correct** – the correct DN will depend on the structure of your directory, and is only required if the directory does not permit anonymous bind.
- **Verify that the Base DN is correct** – the Base DN for user searches is fixed and must be specified as part of the Server URL. If you need to search in different Base DNs to match different kinds of operators, then you should define multiple LDAP Servers and use the priority of each to control the order in which the directory searches are done.

LDAP Translation Rules

LDAP translation rules specify how to determine operator profiles based on LDAP attributes for an authenticated operator.

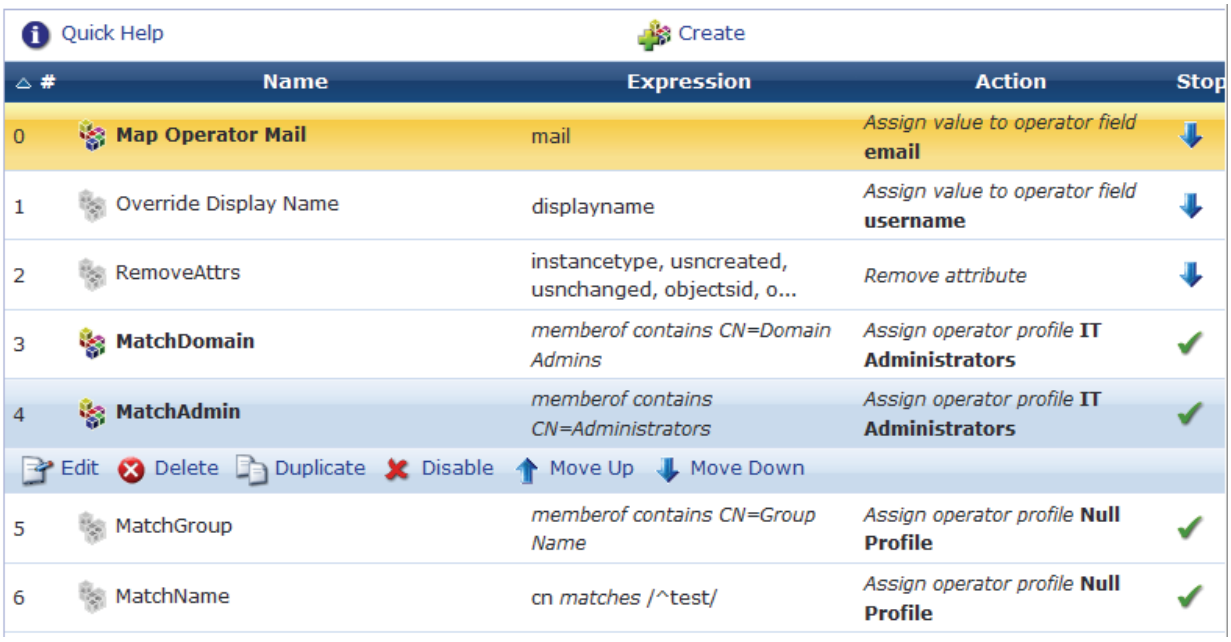
To create a new LDAP translation rule:

1. Go to **Administration > Operator Logins > Translation Rules**, then click the **Create new translation rule** link. The Edit Translation Rule form opens.



2. In the **Name** field, enter a self-explanatory name for the translation rule. In the example above, the translation rule is to check that the user is an administrator, hence the name **MatchAdmin**.
3. Select the **Enabled** check box to enable this rule once you have created it. If you do not select this check box, the rule you create will appear in the rules list, but will not be active until you enable it.
4. Click the **Matching rule** drop-down list and select a rule. The Matching Rule field can be one of:
   - (blank) – always matches
   - **contains** – case-insensitive substring match anywhere in string
   - **matches** – regular expression match, where the value is a Perl-compatible regular expression including delimiters (for example, to match the regular expression "admin" case-insensitively, use the value "**/admin/i**"; See "Regular Expressions" in the "Reference" chapter for more details about regular expressions)
   - **equals** – case-insensitive string comparison, matches on equality
   - **does not equal** – case-insensitive string comparison, matches on inequality
   - **less than** – numerical value is less than the match value
   - **greater than** – numerical value is greater than the match value
   - **starts with** – case-insensitive substring match at start of string
   - **ends with** – case-insensitive substring match at end of string
5. Select a Value. The **Value** field states what is to be matched, in this case **CN=Administrators** to look for a specific group of which the user is a member.
6. Click the **On Match** drop-down list and select the action the system should take when there is a match. Your options here are to:
   - **Do nothing** – makes no changes.
   - **Assign fixed operator profile** – assigns the selected Operator Profile to the operator
   - **Assign attribute's value to operator field** – uses the value of the attribute as the value for an operator field. This option can be used to store operator configuration details in the directory.
   - **Assign custom value to operator field** – uses a template to assign a value to a specific operator field. If you choose this option, the form expands to include the Custom text box for you to enter your custom template code. See "Custom LDAP Translation Processing".
   - **Apply custom processing** – evaluates a template that may perform custom processing on the LDAP operator. If you choose this option, the form expands to include the Custom text box for you to enter your custom template code. See "Custom LDAP Translation Processing".
   - **Remove attribute from operator** – removes the selected LDAP attribute from the operator.
7. Click the **Operator Profile** drop-down list and select the profile to be assigned if there is a rule match. In the example shown above, if the Administrator group is matched, the **Administrator** profile is to be assigned.
8. Select the **Fallthrough** check box if you want to use multiple translation rules. When you create multiple rules, you can build a complete logical structure to perform any type of processing on the LDAP attributes available in your directory.
9. Click **Save Changes** to save your rule settings.

The **Administration > Operator Logins > Translation Rules** window shows a list of all configured translation rules.



Translation rules are processed in order, until a matching rule is found that does not have the Fallthrough field set.

To edit the matching rule list, select an entry in the table to display a menu that lets you perform the following actions:

- **Edit** – changes the configuration of matching rule
- **Delete** – removes matching rule from the list
- **Duplicate** – creates a duplicate copy of an existing rule
- **Disable** – temporarily disables the rule without deleting it from the rule list
- **Enable** – reenables a disabled operator login
- **Move Up** – moves the rule up to a higher priority on the rule list
- **Move Down** – moves the rule down to a lower priority on the rule list

Custom LDAP Translation Processing

When matching an LDAP translation rule, custom processing may be performed using a template.

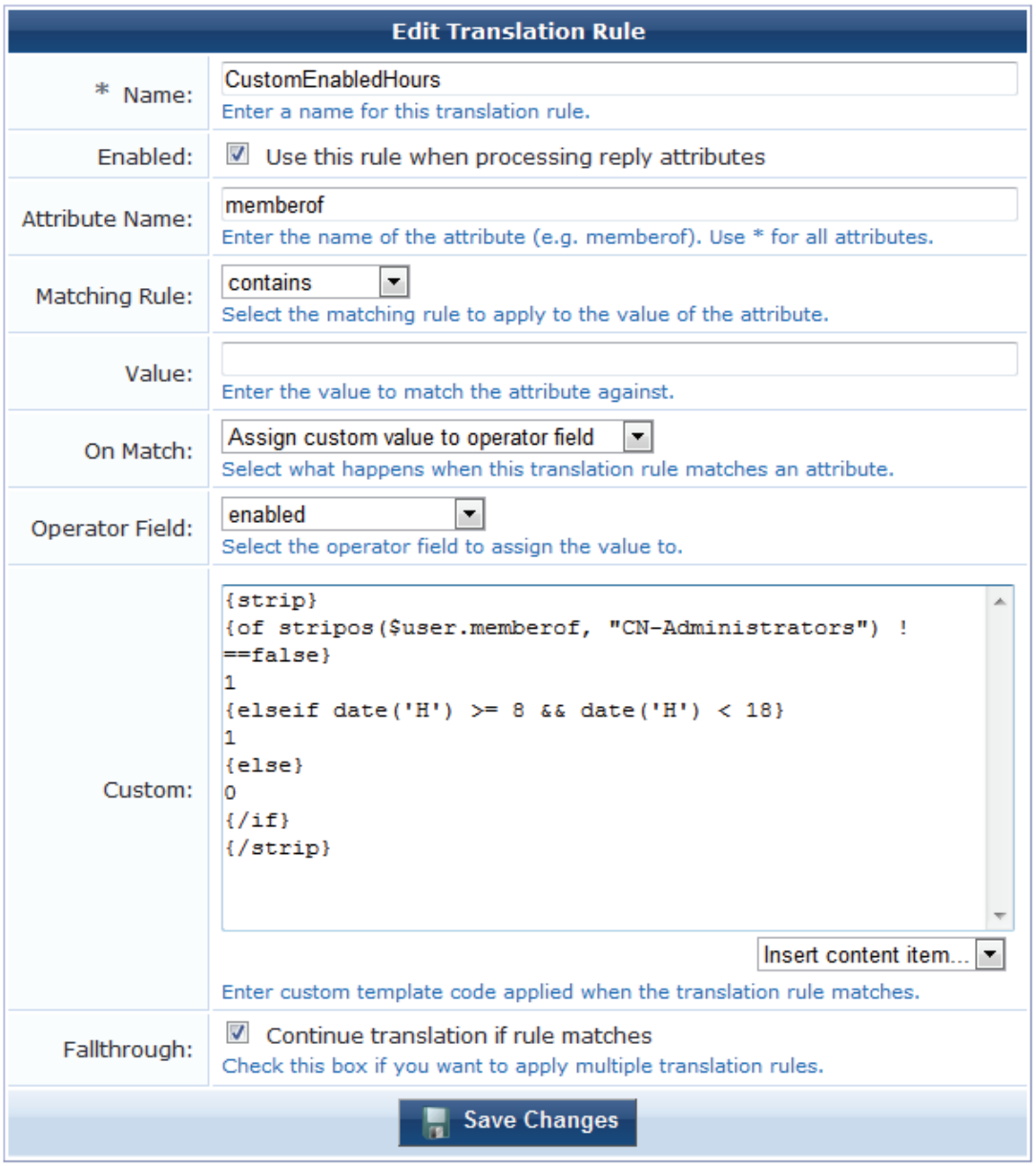
The template variables available are listed in the table below.

Table 23 *Template Variables*

Variable	Description
$attr	The name of the LDAP attribute that was matched.
$user	Contains settings for the operator, including all LDAP attributes returned from the server.

For a Smarty template syntax description, See "Smarty Template Syntax" in the "Reference" chapter. These may be used to make programmatic decisions based on the LDAP attribute values available at login time.

For example, to permit non-administrator users to access the system only between the hours of 8:00 am and 6:00 pm, you could define the following LDAP translation rule:



The Custom rule is:

```
{strip}
{if stripos($user.memberof, "CN=Administrators")!==false}
1
{elseif date('H') >= 8 && date('H') < 18}
1
{else}
0
{/if}
{/strip}
```

Explanation: The rule will always match on the “memberof” attribute that contains the user’s list of groups. The operator field “enabled” will determine if the user is permitted to log in or not. The custom template uses the {strip} block function to remove any whitespace, which makes the contents of the template easier to understand. The {if} statement first checks for membership of the Administrators group using the PHP stripos() function for case-insensitive substring matching; if matched, the operator will be enabled. Otherwise, the server’s current time is checked to see if it is after 8am and before 6pm; if so, the operator

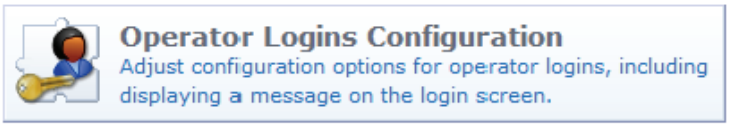
will be enabled. If neither condition has matched, the “enabled” field will be set to 0 and login will not be permitted.

Operator Logins Configuration

You are able to configure a message on the login screen that will be displayed to all operators. This must be written in HTML. You may also use template code to further customize the appearance and behavior of the login screen.

Options related to operator passwords may also be specified, including the complexity requirements to enforce for operator passwords.

Navigate to **Administration > Operator Logins** and click the **Operator Logins Configuration** command link to modify these configuration parameters.



Custom Login Message

If you are deploying ClearPass Guest in a multi-lingual environment, you can specify different login messages depending on the currently selected language.

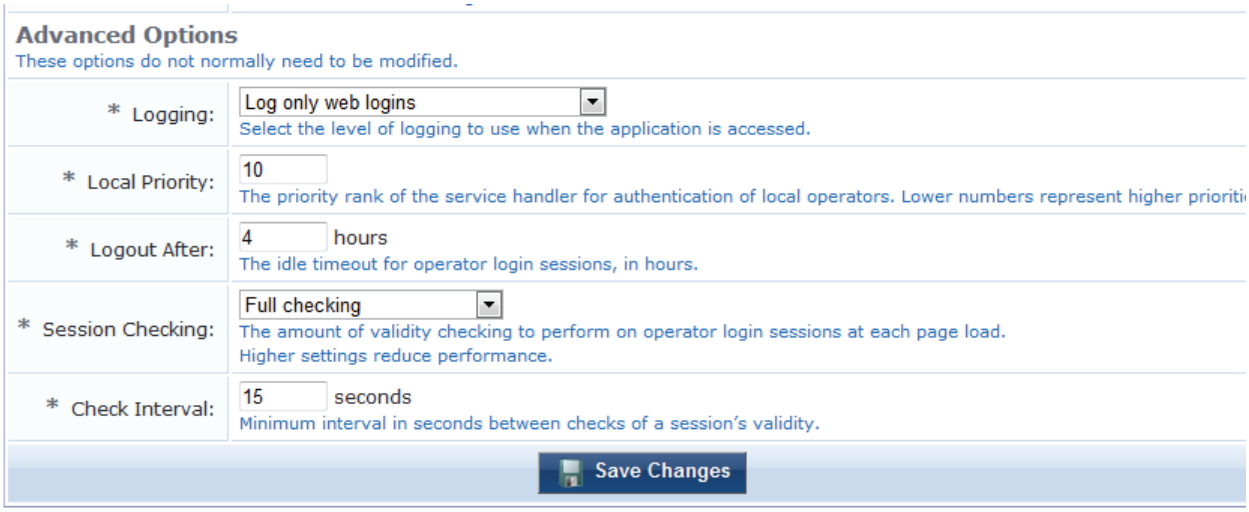
The following example from the demonstration site uses Danish (da), Spanish (es) and the default language English, as highlighted in bold:

```
{if $current_language == 'da'}
<p>
 Indtast brugernavn og password for at <br>
 få adgang til ClearPass Guest
</p>
<p>
 Kontakt <a href="http://www.airwire.dk/">Airwire</a> (Norden) for at få demoadgang
</p>
{elseif $current_language == 'es'}
<p>
```

```
Para entrar en el web demo de ClearPass Guest,<br>
 necesitas un nombre y contraseña.
</p>
<p>
 Si no tienes un login, puedes obtener uno<br>
 <a href="http://www.arubanetworks.com/">contactando con Aruba Networks</a>.
</p>
{else}
<p>
 The ClearPass Guest demo site <br>
 requires a username and password.
</p>
<p>
 If you don't have a login, <br>
 <a href="http://www.arubanetworks.com/">contact Aruba Networks</a> to obtain one.
</p>
{/if}
<br clear="all">
```

In the **Login Footer** field, enter any HTML information that you want displayed in the Operator Login form. Select the login skin from the **Login Skin** drop-down menu. Options include the default skin or a customized skin.

Advanced Operator Login Options



The following options are available in the Logging drop-down list:

- No logging
- Log only failed operator login attempts
- Log only Web logins
- Log only XMLRPC access
- Log all access

Log messages for operator logins, whether successful or unsuccessful, are shown in the application log.

Automatic Logout

The Logout After option in the Advanced Options section lets you configure an amount of idle time after which an operator's session will be ended.

The value for Logout After should be specified in hours. You can use fractional numbers for values less than an hour; for example, use 0.25 to specify a 15 minute idle timeout.

Basic HTML Syntax

ClearPass Guest allows different parts of the user interface to be customized using the Hypertext Markup Language (HTML).

Most customization tasks only require basic HTML knowledge, which is covered in this section.

HTML is a markup language that consists primarily of *tags* that are enclosed inside angle brackets, for example, ** **. Most tags are paired to indicate the start and end of the text being marked up; an end tag is formed by including the tag inside the angle brackets with a forward slash, for example, ** **.

Use the following standard HTML tags in customization:

Table 24 *Standard HTML Tags*

Item	HTML Syntax
Basic Content	
Heading level 1	`<h1>Main Heading</h1>`
Heading level 2	`<h2>Subheading</h2>`
Heading level 3	`<h3>Section heading</h3>`
Regular paragraph text	`<p>Paragraph text</p>`
Line break	` ` ` ` – equivalent syntax (XHTML)

Bullet list	`<ul>`
	`  <li>List item text</li>`
	`</ul>`

Numbered list	`<ol>`
	`  <li>List item text</li>`
	`</ol>`

Text Formatting	
Bold text	`<b>words to be made bold</b>` `<strong>equivalent syntax</strong>`
Italic	`<i>words to be made italic</i>` `<em>equivalent syntax</em>`
Underline	`<u>words to underline</u>`
Typewriter text	`<tt>Shown in fixed-width font</tt>`

Table 24 *Standard HTML Tags (Continued)*

Styled text (inline)	`<span style="…">Uses CSS formatting</span>` `<span class="…">Uses predefined style</span>`
Styled text (block)	`<div style="…">Uses CSS formatting</div>` `<div class="…">Uses predefined style</div>`
Hypertext	
Hyperlink	`<a href="url">Link text to click on</a>`
Inline image	`<img src="url">` `<img src="url" />` – XHTML equivalent
Floating image	`<img src="url" align="left">`

For more details about HTML syntax and detailed examples of its use, consult a HTML tutorial or reference guide.

Standard HTML Styles

ClearPass Guest defines standard CSS classes you can use to provide consistent formatting within the user interface.

Examples of these styles are given below.

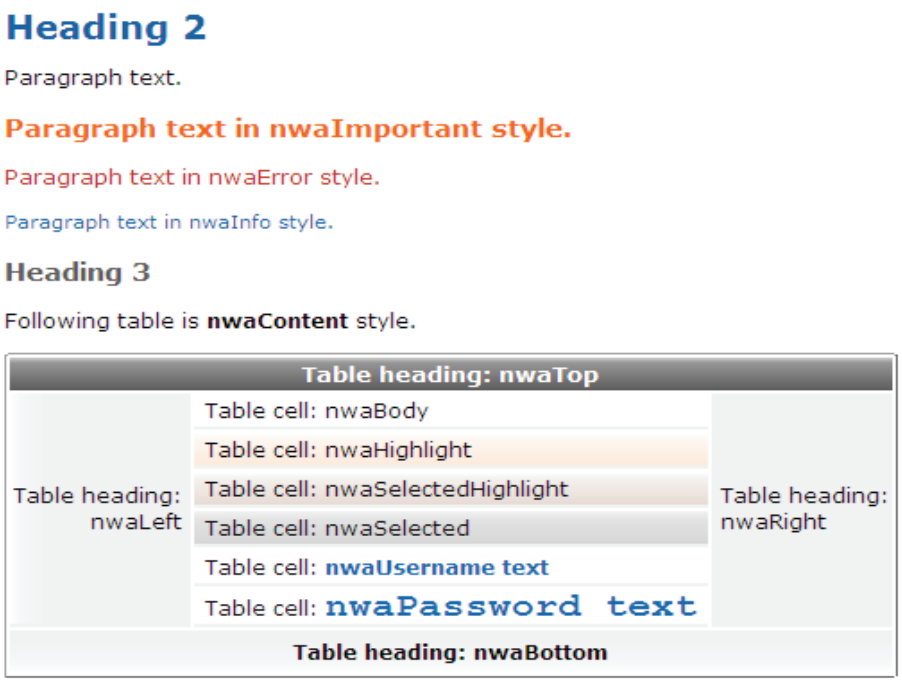


Table 25 *Formatting Classes*

Class Name	Applies To	Description
nwaIndent	Tables	Indent style used in tables
nwaLayout	Tables	Used when you want to lay out material in a table without the material looking as if it is in a table; in other words, without borders

Table 25 *Formatting Classes (Continued)*

nwaContent	Tables	Class used for a standard table with borders
nwaTop	Table Header	Table heading at top
nwaLeft	Table Header	Left column of table
nwaRight	Table Header	Right column of table
nwaBottom	Table Header	Table heading at bottom
nwaBody	Table Cell	Style to apply to table cell containing data
nwaHighlight	Table Cell	Highlighted text (used for mouseover)
nwaSelected	Table Cell	Selected text (table row after mouse click)
nwaSelectedHighlight	Table Cell	Selected text with mouseover highlight
nwaInfo	All	Informational text message
nwaError	All	Error text message
nwaImportant	All	Text that should be prominently displayed Table subheadings
nwaUsername	All	Text used to display a username
nwaPassword	All	Text used to display a password

Smarty Template Syntax

ClearPass Guest's user interface is built using the Smarty template engine. This template system separates the program logic and visual elements, enabling powerful yet flexible applications to be built.

When customizing template code that is used within the user interface, you have the option of using Smarty template syntax within the template. Using the programming features built into Smarty, you can add your own logic to the template. You can also use predefined template functions and block functions to ensure a consistent user interface.

Basic Template Syntax

Following is a brief introduction to the usage of the Smarty template engine. For more information, please refer to the Smarty documentation at http://www.smarty.net/docs.php, or the Smarty Crash Course at http://www.smarty.net/crashcourse.php.

Text Substitution

Simple text substitution in the templates may be done with the syntax **{$variable}**, as shown below:

```
The current page's title is: {$title}
```

Template File Inclusion

To include the contents of another file, this can be done with the following syntax:

```
{include file="public/included_file.html"}
```

Note that Smarty template syntax found in these files is also processed, as if the file existed in place of the **{include}** tag itself.

Comments

To remove text entirely from the template, comment it out with the Smarty syntax **{*** *commented text* ***}**. Note that this is different from a HTML comment, in that the Smarty template comment will never be included in the page sent to the Web browser.

Variable Assignment

To assign a value to a page variable, use the following syntax:

```
{assign var=name value=value}
```

The "value" can be a text value (string), number, or Smarty expression to be evaluated, as shown in the examples below:

```
{assign var=question value="forty plus two"}
The question is: {$question}
{assign var=answer value=42}
The answer is: {$answer}
{assign var=question_uppercase value=$question|strtoupper}
THE QUESTION IS: {$question_uppercase}
```

Conditional Text Blocks

To include a block of text only if a particular condition is true, use the following syntax:

```
{if $username != ""}
  <tr>
   <td class="nwaBody">Username:</td>
   <td class="nwaBody">{$username}</td>
  </tr>
{else}
  <!-- No user name, no table row -->
{/if}
```

The condition tested in the **{if} … {/if}** block should be a valid PHP expression. Note that the **{else}** tag does not require a closing tag.

Script Blocks

The brace characters **{** and **}** are specially handled by the Smarty template engine. Using text that contains these characters, such as CSS and JavaScript blocks, requires a Smarty block **{literal} … {/literal}**:

```
<script type="text/javascript" language="JavaScript">
{literal}
<!--
function my_function() {
    // some Javascript code here
}
// -->
{/literal}
</script>
```

Failing to include the **{literal}** tag will result in a Smarty syntax error when using your template. Single instances of a **{** or **}** character can be replaced with the Smarty syntax **{ldelim}** and **{rdelim}** respectively.

Repeated Text Blocks

To repeat a block of text for each item in a collection, use the **{section} ... {/section}** tag:

```
{section loop=$collection name=i}
<tr>
  <td class="nwaBody">
   {$collection[i].name}
  </td>
</tr>
{sectionelse}
  <!-- included if $collection is empty -->
{/section}
```

Note that the content after a **{sectionelse}** tag is included only if the **{section}** block would otherwise be empty.

Foreach Text Blocks

An easier to use alternative to the **{section} ... {/section}** tag is to use the **{foreach} ... {/foreach}** block:

```
{foreach key=key_var item=item_var from=$collection}
  {$key_var} = {$item_var}
{foreachelse}
  <!—included if $collection is empty -->
{/foreach}
```

The advantage of this syntax is that each item in the collection is immediately available as the named item variable, in this example **{$item_var}**. This construct is also useful when iterating through associative arrays indexed by key, as the key is immediately available with each item.

A `name=` attribute may be supplied with the opening **{foreach}** tag. When a name is supplied, the following additional Smarty variables are available for use inside the **{foreach} ... {/foreach}** block:

- **{$smarty.foreach.*name*.first}** – true if the item being processed is the first item in the collection
- **{$smarty.foreach.*name*.last}** – true if the item being processed is the last item in the collection
- **{$smarty.foreach.*name*.index}** – counter for the current item, starting at 0 for the first item
- **{$smarty.foreach.*name*.iteration}** – counter for the current item, starting at 1 for the first item
- **{$smarty.foreach.*name*.total}** – value indicating the total number of items in the collection

Note that the content after a **{foreachelse}** tag is included only if the **{foreach}** block would otherwise be empty.

Modifiers

Smarty provides *modifiers* that can be used to gain greater control over the formatting of data. Modifiers can be included by following a variable with a vertical bar **|** and the name of the modifier. Any arguments to the modifier can be specified using a colon **:** followed by the arguments.

The following example prints a date using the YYYY-MM-DD syntax:

```
{$expire_time|nwadateformat:"%Y-%m-%d"}
```

See **"Date/Time Format Syntax"** for detailed information on the date/time format modifiers, and see Table 26.

Table 26 *Smarty Modifiers*

Modifier	Description
htmlspecialchars	Escapes characters used in HTML syntax with the equivalent HTML entities (& for &, < for < and > for >)
nl2br	Replaces newline characters in the value with HTML line breaks (** **)
number_format	Formats a numerical value for display; an optional modifier argument may be used to specify the number of decimal places to display (default is 0)
nwadateformat	Date/time formatting; see**"nwadateformat Modifier"** for details about this modifier function
nwatimeformat	Date/time formatting; see**"Date/Time Format String Reference"** for details about this modifier function
nwamoneyformat	Formats a monetary amount for display purposes; an optional modifier argument may be used to specify the format string. This modifier is equivalent to the NwaMoneyFormat() function; see"NwaMoneyFormat" for details.
strtolower	Converts the value to lowercase
strtoupper	Converts the value to uppercase
ucfirst	Converts the first character of the value to uppercase
ucwords	Converts the first character of each word in the value to uppercase

Predefined Template Functions

Template functions are used to perform different kinds of processing when the template is used. The result of a template function takes the place of the function in the output of the template.

Functions are of two kinds: *block functions*, which have a beginning and ending tag enclosing the text operated on by the function, and *template functions*, which have just a single tag and do not enclose text.

To use a function, enclose the function name in curly braces **{ }** and provide any attributes that may be required for the function. Block functions also require a closing tag.

dump

`{dump var=$value}`

Smarty registered template function. Displays the value of a variable.

Use the following Smarty syntax to print a variable's contents:

```
{dump var=$var_to_dump export=html}
```

The contents of the variable are printed in a **<pre>** block. Use the attribute "export=1" to use PHP's `var_export()` format, or omit this attribute to get the default behavior – PHP's `var_dump()` format.

Use the attribute "html=1" to escape any HTML special characters in the content. This can also be done with attribute "export=html", and is recommended for use in most situations (so that any embedded HTML is not interpreted by the browser).

nwa_commandlink

`{nwa_commandlink} … {/nwa_commandlink}`

Smarty registered block function. Generates a “command link” consisting of an icon, main text and explanatory text.

Command links are block elements and are roughly the equivalent of a form button. A command link is typically used to represent a choice the user should make to proceed. The command link contains an icon, command text (that sums up the action taken by the command link), and any explanatory text needed for the command.

Usage example:

```
{nwa_commandlink icon="images" command="Command Link" linkwidth="400"
commandclass="nwaImportant" text="This is a sentence explaining the command."
textclass="nwaInfo"}link_here.php{/nwa_commandlink}
```

- The “icon” parameter is the SRC to the image of the icon. This should normally be a relative path.
- The “command” parameter is the main text of the command link.
- The “text” parameter is the explanatory text describing the action that lies behind the command link. (This is optional.)
- The “linkwidth” parameter, if specified, indicates the width of the command link in pixels. This should be at least 250; the recommended value is 400.
- The “width” and “height” parameters, if specified, provide the dimensions of the icon to display. If not specified, this is automatically determined from the image.
- The “onclick” parameter, if specified, provides the contents for the onclick attribute of the link.
- The “commandclass” parameter, if specified, sets the class attribute of the DIV element enclosing the command text. The default class is “nwaImportant”.
- The “textclass” parameter, if specified, sets the class attribute of the P element enclosing the command link’s descriptive text. The default class is “nwaInfo”.
- The “alt” parameter, if specified, sets the ALT attribute of the command link’s icon. If not specified, the default alt text used is the command text.
- The “target” parameter, if specified, sets the TARGET attribute of the hyperlink. If not specified, no TARGET attribute is provided.

The body of the element is the HREF of the command link. The “icon” and “command” parameters are required. All other parameters are optional.

nwa_iconlink

`{nwa_iconlink} … {/nwa_iconlink}`

Smarty registered block function. Generates a combined icon and text link to a specified URL.

Usage example:

```
{nwa_iconlink icon="images/icon-info22.png" text="More
Information"}more_information.php{/nwa_iconlink}
```

- The “icon” parameter is the SRC to the image of the icon. This should normally be a relative path.
- The “text” parameter is the text to display next to the icon. This will also be used as the alternate text (that is, a tooltip) for the icon image.
- The “width” and “height” parameters, if specified, provide the dimensions of the icon to display. If not specified, this is automatically determined from the image.
- The “onclick” parameter, if specified, provides the contents for the onclick attribute of the link.
- The “target” parameter, if specified, provides the contents for the target attribute of the link.

- The “alt” parameter, if specified, sets the ALT attribute of the icon. If not specified, the default alt text used is the icon text.
- The “style” parameter, if specified, provides CSS for the SPAN element used to implement the icon link.

The body of the element is the HREF of the link. This HREF will be added to both the icon and the text. If the content of the link is empty, no link will be inserted. This can be used to insert an icon and text as an inline group. Note that no HTML entity escaping is performed when inserting content using this function.

nwa_icontext

```
{nwa_icontext} … {/nwa_icontext}
```

Smarty registered block function. Generates a block of text with a marker icon displayed in the top left.

Usage examples:

```
{nwa_icontext icon="images/icon-info22.png"}Text to display{/nwa_icontext}
{nwa_icontext type="info"}Information block{/nwa_icontext}
```

- The “icon” parameter, if specified, is the SRC to the image of the icon. This should normally be a relative path.
- The “width” and “height” parameters, if specified, provide the dimensions of the icon to display. If not specified, this is automatically determined from the image.
- The “alt” parameter, if specified, provides the alternate text for the icon.
- The “class” parameter, if specified, is the style name to apply to a containing DIV element wrapped around the content. If this is empty, and a default is not provided through the “type” parameter, no wrapper DIV is added.
- The “style” parameter, if specified, is the CSS inline style to apply to a containing DIV element, as for the “class” parameter.
- The “type” parameter, if specified, indicates a predefined style to apply; this may be one of the following:
  - **error** – red cross symbol
  - **fatal** – skull symbol
  - **info** – information symbol
  - **note** (or **arrow**) – right-pointing arrow
  - **ClearPass Guest** – ClearPass Guest logo
  - **ok** (or **tick**) – green tick mark
  - **warn** (or **warning**) – warning symbol
  - **wait** – animated spinner

If “noindent=1” is specified, the block is not indented using the ‘nwaIndent’ style. If “novspace=1” is specified, the block uses a ‘DIV’ element, rather than a ‘P’ element. If neither “icon” nor “type” is supplied, the default behavior is to insert an “info” type image. Specifying a “type” is equivalent to specifying an “icon", “width", “height” and “alt” parameter, and may also include a “class” depending on the type selected.

Usage example:

```
{nwa_icontext struct=$error}{/nwa_icontext}
```

The “struct” parameter, if specified, uses a standard result type. If the “error” key is set and non-zero, the “type” parameter is set to the value error, and the “message” key is converted to a HTML formatted error message for display.

nwa_quotejs

`{nwa_quotejs} … {/nwa_quotejs}`

Smarty registered block function. Quotes its content in a string format suitable for use in JavaScript. This function also translates UTF-8 sequences into the corresponding JavaScript Unicode escape sequence (\uXXXX)

Usage example:

```
{nwa_quotejs}String with ' and "{/nwaquote_js}
```

The output of this will be:

```
'String with \' and \"'
```

The “body” parameter, if set, indicates that the string quotes are already supplied; in this case the beginning and ending quotes are not included in the output.

nwa_radius_query

`{nwa_radius_query _method=`*`MethodName`* `_assign=`*`var`* `…}`

Smarty registered template function. Performs accounting-based queries on the RADIUS server and returns the result for use in a template.

Usage example:

```
{nwa_radius_query _method=GetCallingStationTraffic
  callingstationid=$dhcp_lease.mac_address
  from_time=86400 in_out=out _assign=total_traffic}
```

This example uses the `GetCallingStationTraffic` query function and passes the “callingstationid”, “from_time” and “in_out” parameters. The result is assigned to a template variable called total_traffic, and will not generate any output.

This template function accepts the following parameters to select a RADIUS database and other connection options:

- **_db** – ID of the RADIUS database service handler (this parameter is optional, the default service handler will be used if it not set)
- **_debug** – Set to a nonzero value to enable debugging
- **_quiet** – Set to a nonzero value to inhibit warning/error messages

The following parameters control the query to be executed:

- **_method** (required) – Name of the query function to execute. A brief listing of the available methods is provided below.
- **_arg0, _arg1, …, _argN** (optional) – Positional arguments for the query function.
- Named arguments may also be supplied; the arguments must be named identically to the function arguments listed in the documentation for the query function.

The following parameters control how the result should be processed:

- **_assign** – Name of a page variable to store the output; if not set, output is sent to the browser as the result of evaluating the template function.
- **_output** – Index of item to return from the RPC result; if not set, the complete result is returned. This may be of use when an array containing multiple values is returned and only one of these values is required.
- **_default** – Default value to display or return if an error occurs or the _output field is not available in the result.

For ease of use, “assign” is also supported as a synonym for “_assign”.

This template function does not generate any output if the **_assign** parameter is set.

The methods that are available for use with this function are listed below. The $criteria array consists of one or more criteria on which to perform a database search. The array is used for advanced cases where pre-defined helper functions do not provide required flexibility.

ChangeToRole()

```
ChangeToRole($username, $role_name)
```

Changes the RADIUS role assigned to the user. If the user currently has active sessions, this function will trigger an RFC 3576 Change-of-Authorization (CoA) Request to the network access server.

The $username parameter specifies the user account to modify; use the expression GetAttr('User-Name') to use the value from the RADIUS User-Name attribute.

The $role_name parameter specifies the name of the RADIUS User Role to apply to the user.

Example:

Use the following as a conditional expression for an attribute. If the user's traffic in the past 24 hours exceeds 50 MB, the user is changed to the "Over-Quota" role.

```
return GetUserTraffic(86400) > 50e6 && ChangeToRole("Over-Quota");
```

GetCallingStationCurrentSession()

```
GetCallingStationCurrentSession($callingstationid, $mac_format = null)
```

Looks up the current (most recent) active session for the specified calling station ID.

Because different NAS equipment can send differently-formatted MAC addresses in the Calling-Station-Id attribute, the `$mac_format` argument may be specified. This should be a sprintf-style format string that accepts 6 arguments (the octets of the MAC address). The default if not specified is the IEEE 802 standard format, **%02X-%02X-%02X-%02X-%02X-%02X** – that is, uppercase hexadecimal with each octet separated with a hyphen.

See "GetCurrentSession()" for details of the return value.

GetCallingStationSessions()

```
GetCallingStationSessions($callingstationid, $from_time, $to_time = null,
$mac_format = null)
```

Calculate the number of sessions for accounting records matching a specific calling-station-id. The calling station id address is looked up automatically from the RADIUS Access-Request (Calling-Station-ID attribute).

Because different NAS equipment can send differently-formatted MAC addresses in the Calling-Station-Id attribute, the `$mac_format` argument may be specified. This should be a sprintf-style format string that accepts 6 arguments (the octets of the MAC address). The default if not specified is the IEEE 802 standard format, **%02X-%02X-%02X-%02X-%02X-%02X** – that is, uppercase hexadecimal with each octet separated with a hyphen.

See "GetTraffic()" for details on how to specify the time interval.

GetCallingStationTime()

```
GetCallingStationTime($callingstationid, $from_time, $to_time = null, $mac_format =
null)
```

Calculate sum of session times in a specified time interval.

Because different NAS equipment can send differently-formatted MAC addresses in the Calling-Station-Id attribute, the `$mac_format` argument may be specified. This should be a sprintf-style format string that accepts 6 arguments (the octets of the MAC address). The default if not specified is the IEEE 802 standard format, **%02X-%02X-%02X-%02X-%02X-%02X** – that is, uppercase hexadecimal with each octet separated with a hyphen.

The calling station ID is looked up automatically from the RADIUS Access-Request (Calling-Station-ID attribute).

See "GetTraffic()" for details on how to specify the time interval.

GetCallingStationTraffic()

```
GetCallingStationTraffic($callingstationid, $from_time, $to_time = null,
$in_out = null, $mac_format = null)
```

Calculate sum of traffic counters in a time interval. Sessions are summed if they have the same Calling-Station-Id attribute as that specified in the RADIUS Access-Request.

If no Calling-Station-Id attribute was included in the request, returns zero.

Because different NAS equipment can send differently-formatted MAC addresses in the Calling-Station-Id attribute, the `$mac_format` argument may be specified. This should be a sprintf-style format string that accepts 6 arguments (the octets of the MAC address). The default if not specified is the IEEE 802 standard format, **%02X-%02X-%02X-%02X-%02X-%02X** – that is, uppercase hexadecimal with each octet separated with a hyphen. This string matches what ClearPass Guest sees from the NAS.

The time interval specified by `$from_time` and optionally `$to_time` is also used to narrow the search.

If `$to_time` is not specified, `$from_time` is a "look back" time, that is, the time interval in seconds before the current time.

If `$to_time` is specified, the interval considered is between `$from_time` and `$to_time`.

`$in_out` may be "in" to count only input octets, "out" to count only output octets, or any other value to count both input and output octets towards the traffic total.

Examples:

- Use the following as the condition expression for a RADIUS role attribute. Authorizes a user only if their total traffic (in + out) in the past day does not exceed 10 MB. Note that the attribute with this condition expression will never be included in the response!
- return GetUserTraffic(86400) > 10485760 && AccessReject()
- Like the above, but only considers output (that is, user downloads):
- return GetUserTraffic(86400,'out') > 10485760 && AccessReject()
- Another way to limit the past 30 days downloads to 100 MB:
- return GetUserTraffic($now - 86400*30, $now, 'out') > 100*1024*1024 && AccessReject()
- Limit by MAC address, 50 MB download in past 24 hours:

```
return GetCallingStationTraffic(86400, 'out') > 50000000 && AccessReject()
```

GetCurrentSession()

```
GetCurrentSession($criteria)
```

Looks up the details for an active session, based on the specified criteria.

This is a multi-purpose function that has a very flexible query interface; for ease of use, consider using one of the related functions **GetCallingStationCurrentSession()**, **GetIpAddressCurrentSession()**, or **GetUserCurrentSession()**.

Returns null if there is no matching session, otherwise returns a single session array – a typical result follows:

```
array (
  'id' => '2073',
  'acctsessionid' => '4a762dbf00000002',
  'acctuniqueid' => 'c199b5a94ebf5184',
```

```
  'username' => 'demo@example.com',
  'realm' => '',
  'role_name' => 'Guest',
  'nasipaddress' => '192.168.2.20',
  'nasportid' => '',
  'nasporttype' => '',
  'calledstationid' => '',
  'callingstationid' => '',
  'acctstarttime' => '1249258943',
  'connectinfo_start' => '',
  'acctstoptime' => NULL,
  'connectinfo_stop' => NULL,
  'acctsessiontime' => 0,
  'acctinputoctets' => 0,
  'acctoutputoctets' => 0,
  'acctterminatecause' => NULL,
  'servicetype' => '',
  'framedipaddress' => '192.168.2.3',
  'framedprotocol' => '',
  'acctauthentic' => '',
  'nastype' => 'cisco_3576',
  'nas_name' => 'centos',
  'total_traffic' => 0,
  'state' => 'stale',
  'traffic_input' => 0,
  'traffic_output' => 0,
  'traffic_usage' => 0,
  'session_time' => 29641260,
)
```

GetIpAddressCurrentSession()

```
GetIpAddressCurrentSession($ip_addr = null)
```

Looks up the current (most recent) active session for the specified client IP address. If ip_addr is not specified, it defaults to the current value of $smarty.server.REMOTE_ADDR, which may not be the same value as the IP address of the session if there is a NAT.

See "GetCurrentSession()" for details of the return value.

GetIpAddressSessions()

```
GetIpAddressSessions($ip_addr, $from_time = null, $to_time = null)
```

Calculate the number of sessions for accounting records matching a specific IP address. The IP address attribute is looked up automatically from the RADIUS Access-Request (Framed-IP-Address attribute).

See "GetTraffic()" for details on how to specify the time interval.

See "GetIpAddressTraffic()" for additional details on the `$ip_addr` argument.

GetIpAddressTime()

```
GetIpAddressTime($ip_addr, $from_time = null, $to_time = null)
```

Calculate sum of session times in a specified time interval. The IP address is looked up automatically from the RADIUS Access-Request (Framed-IP-Address attribute).

See "GetTraffic()" for details on how to specify the time interval.

See "GetIpAddressTraffic()" for additional details on the `$ip_addr` argument.

GetIpAddressTraffic()

```
GetIpAddressTraffic($ip_addr, $from_time = null, $to_time = null, $in_out = null)
```

Calculate sum of traffic counters in a time interval. The IP address used is determined based on the context. If processing a RADIUS Access-Request, the IP address is determined using the Framed-IP-Address attribute. If processing a HTTP request, the current client IP address is assumed (from $_SERVER['REMOTE_ADDR']).

Specifying an empty value for the IP address (such as null, false, or empty string) also causes the current client IP address to be used.

See "GetTraffic()" for details on how to specify the time interval.

GetSessions()

```
GetSessions($criteria, $from_time, $to_time = null)
```

Calculate the number of sessions from accounting records in the database.

NOTE

This is a multi-purpose function that has a very flexible query interface; for ease of use, consider using one of the related functions **GetCallingStationSessions()**, **GetIpAddressSessions()**, **GetUserActiveSessions()**, or **GetUserSessions()**.

`$criteria` is the criteria on which to search for matching accounting records.

As well as the criteria specified, the time interval specified by `$from_time` and optionally `$to_time` is also used to narrow the search.

If `$to_time` is not specified, `$from_time` is a "look back" time, that is, the time interval in seconds before the current time.

If `$to_time` is specified, the interval considered is between `$from_time` and `$to_time`.

Returns the total number of sessions for matching accounting records in the time interval specified.

GetSessionTimeRemaining()

```
GetSessionTimeRemaining($username, $format = "relative")
```

Calculates the session time remaining for a given user account, if the user account was to be authenticated at the moment of the call.

The $username parameter is required. This is the username for the authentication.

The $format parameter is optional, and defaults to "relative" if not otherwise specified. This parameter may be one of the following values:

- "relative" or "session_time": Calculates the session timeout as for the Session-Timeout RADIUS attribute, that is, the number of seconds before the session should end. If the session does not have a session timeout, the value returned is 0.
- "time": Calculates the session end time, as the UNIX time at which the session should end. If the session does not have an expiration time, the value returned is 0.
- Other values: These are interpreted as a date format (see "NwaDateFormat") and the session end time is returned in this format. (Examples: "iso8601", "longdate", "recent", '%Y-%m-%d %H:%M', etc.). If the session does not have an expiration time, the value returned is a blank string.

GetTime()

```
GetTime($criteria, $from_time, $to_time = null)
```

Calculate the sum of session times for accounting records in the database.

This is a multi-purpose function that has a very flexible query interface; for ease of use, consider using one of the related functions See "GetCallingStationTime()" , GetIpAddressTime(), or GetUserTime().

`$criteria` is the criteria on which to search for matching accounting records.

As well as the criteria specified, the time interval specified by `$from_time` and optionally `$to_time` is also used to narrow the search.

If `$to_time` is not specified, `$from_time` is a "look back" time, that is, the time interval in seconds before the current time.

If `$to_time` is specified, the interval considered is between `$from_time` and `$to_time`.

Returns the total session time for all matching accounting records in the time interval specified.

GetTraffic()

```
GetTraffic($criteria, $from_time, $to_time = null, $in_out = null)
```

Calculate the sum of traffic counters for accounting records in the database.

This is a multi-purpose function that has a very flexible query interface; for ease of use, consider using one of the related functions **GetCallingStationTraffic()**, **GetIpAddressTraffic()**, or **GetUserTraffic()**.

`$criteria` is the criteria on which to search for matching accounting records. The time interval specified by `$from_time` and optionally `$to_time` is used with the criteria to narrow the search.

If `$to_time` is not specified, `$from_time` is a "look back" time, that is, the time interval in seconds before the current time. If `$to_time` is specified, the interval considered is between `$from_time` and `$to_time`.

`$in_out` may be "in" to count only input octets, "out" to count only output octets, or any other value to count both input and output octets towards the traffic total. This argument returns the computed total of traffic for all matching accounting records.

GetUserActiveSessions()

```
GetUserActiveSessions($username, $callingstationid = null)
```

Looks up the list of all sessions for the specified username.

The username attribute is looked up automatically from the RADIUS Access-Request (User-Name attribute). If a `$callingstationid` argument is supplied, sessions that match that Calling-Station-Id are excluded from the count of active sessions.

GetUserActiveSessionCount()

```
GetUserActiveSessionCount($username)
```

Counts the number of currently active sessions for the current username.

The username attribute is looked up automatically from the RADIUS Access-Request (User-Name attribute).

GetUserCumulativeUsage()

```
GetUserCumulativeUsage($username)
```

Looks up the total cumulative time for the username.

The username attribute is looked up automatically from the RADIUS Access-Request (User-Name attribute).

GetUserCurrentSession()

```
GetUserCurrentSession($username)
```

Looks up the current (most recent) active session for the specified username.

See "GetCurrentSession()" for details of the return value.

GetUserFirstLoginTime()

```
GetUserFirstLoginTime($username)
```

Looks up the first login time for the specified username.

The username attribute is looked up automatically from the RADIUS Access-Request (User-Name attribute).

GetUserSessions()

```
GetUserSessions($username, $from_time, $to_time = null)
```

Calculate the number of sessions for accounting records matching a specific user-name. The username attribute is looked up automatically from the RADIUS Access-Request (User-Name attribute).

See "GetTraffic()" for details on how to specify the time interval.

GetUserTime()

```
GetUserTime($username, $from_time, $to_time = null)
```

Calculate sum of session times in a specified time interval.

See "GetTraffic()" for details on how to specify the time interval.

GetUserTraffic()

```
GetUserTraffic($username, $from_time, $to_time = null, $in_out = null)
```

Calculate sum of traffic counters in a time interval. Sessions are summed if they have the same User-Name attribute as that specified in the RADIUS Access-Request.

See "GetCallingStationTraffic()" for details on how to specify the time interval.

Advanced Developer Reference

The reference documentation in this section is intended for advanced usage by developers.

nwa_assign

```
{nwa_assign …}
```

Smarty registered template function. Assigns a page variable based on the output of a generator function.

Simple usage example:

```
{nwa_assign var=my_variable value=my_value}
```

- The "var" parameter specifies the page variable that will receive the output.
- The "value" parameter specifies the value to assign to "var".

The various request variables may also be accessed using one of two supported methods:

- `{nwa_assign var=_GET.get_variable value=...}`
- `{nwa_assign var=smarty.get.get_variable value=...}`

The variables that can be accessed this way are _GET (smarty.get), _POST (smarty.post), _REQUEST (smarty.request), _SESSION (smarty.session), _COOKIE (smarty.cookies), and _ENV (smarty.env).

Assigning to values in _SESSION will persist the value for the next page load in the session.

Alternative usage example:

```
{nwa_assign var=userskin_plugin generator=NwaGetPluginDetails arg=$u.userskin}
```

- The "generator" parameter specifies the generator function to be called.
- A single "arg" parameter, if specified, provides a 1-argument form of calling the function; alternatively, "arg1", "arg2", ... may be specified to form an array of arguments to pass to the generator.

nwa_bling

`{nwa_bling ...}`

Smarty registered template function. Adds various kinds of visual effects to the page.

Usage example:

```
{nwa_bling id=$some_id type=fade}
```

The "id" parameter is the ID of the HTML element to which you will add 'bling' effects The "type" parameter is the kind of bling desired:

- "fade": element smoothly fades in and out
- "blink": element blinks slowly

nwa_makeid

`{nwa_makeid ...}`

Smarty registered template function. Creates a unique identifier and assigns it to a named page variable. Identifiers are unique for a given page instantiation.

Usage example:

```
{nwa_makeid var=some_id}
```

The "var" parameter specifies the page variable that will be assigned.

Alternative usage:

```
{nwa_makeid var=some_id file=filename}
```

The "file" parameter specifies a file which contains a unique ID. This allows issued IDs to be unique across different page loads. To return the value rather than assign it to a variable, use the syntax:

```
{nwa_makeid [file=filename] output=1}
```

Otherwise, this template function does not generate any output.

nwa_nav

`{nwa_nav} ... {/nwa_nav}`

Smarty registered block function. Defines a block area for navigation, a control, or generates navigation control HTML of a particular type.

Blocks are individual components of the navigation area, which basically consist of HTML. Blocks for actual navigation items have substitution tags in the form **@tagname@**.

The recognized tags are described in the table below.

Table 27 *Navigation Tags*

Tag	Description
@a@	navigation name
@name@	navigation item name (HTML safe)
@jsname@	navigation item name (JavaScript quoted)
@href@	navigation item hyperlink
@jshref@	navigation item hyperlink (JavaScript quoted)
@icon@	navigation item icon, if specified

When used with the “block” parameter, the {nwa_nav} control does not generate any HTML. When used with the “type” parameter, the {nwa_nav} control uses the previously defined blocks to generate the HTML navigation area. The following types are recognized:

- **simple** – Only the current L1 item has L2 items, L3 only when L2 active
- **all-l1** – All current L1 items are shown to L3, otherwise L1 only
- **expanded** – All L1 items have L2 items, L3 only when L2 active
- **all-expanded** – All items shown to L3

The “reset” parameter may be specified to clear any existing navigation settings.
Usage example:

```
{nwa_nav block=level1_active}<li class="active">@a@</li>{/nwa_nav}
{nwa_nav block=level1_inactive}<li>@a@</li>{/nwa_nav}
...
{nwa_nav type=simple}{/nwa_nav} {* this generates the HTML *}
```

Block types can be one of the following types:

- enter_level1_item
- enter_level2_item
- enter_level3_item
- exit_level1_item
- exit_level2_item
- exit_level3_item
- between_level1_items
- between_level2_items
- between_level3_items
- level1_active
- level1_inactive
- level2_active
- level2_inactive
- level2_parent_active
- level2_parent_inactive
- level3_active
- level3_inactive

- enter_level1
- enter_level2
- enter_level3
- exit_level1
- exit_level2
- exit_level3

nwa_plugin

`{nwa_plugin ...}`

Smarty registered template function. Generates plugin information based on the parameters specified. Specifying which plugin:

- The 'id' parameter specifies a plugin ID.
- The 'name' parameter specifies a plugin name, or plugin filename.
- The 'page' parameter specifies a page name provided by the plugin.
- The 'privilege' parameter specifies a privilege defined by the plugin.

If none of the above is specified, the default is the same as specifying the 'page' parameter with the current script name as argument (that is, the current page).

Specifying the output:

- The 'notfound' parameter specifies the return value, if the plugin was not found (default is the empty string).
- The 'output' parameter specifies the metadata field to return

If 'output' is not specified, the default is 'output=id'; that is, the plugin ID is returned.

nwa_privilege

`{nwa_privilege} ... {/nwa_privilege}`

Smarty registered block function. Includes output only if a certain kind of privilege has been granted.

Usage examples:

```
{nwa_privilege access=create_user} .. content .. {/nwa_privilege}
```

The "access" parameter specifies the name of a privilege to check for any access.

```
{nwa_privilege readonly=create_user} .. content .. {/nwa_privilege}
```

The "readonly" (synonym "ro") parameter specifies the name of a privilege to check for read-only access. Note that an operator with read-write access also has read-only access. To include content if the user ONLY has read access, that is, not if the user has full access, prefix the privilege name with a **#** character and use the parameter name "readonly" (or "ro").

```
{nwa_privilege full=create_user} .. content .. {/nwa_privilege}
```

The "full" (synonym "rw") parameter specifies the name of a privilege to check for full read-write access. The "name" parameter is the name of the privilege to check. If "name" is prefixed with a "!", the output is included only if that privilege is NOT granted (inverts the sense of the test). An optional "level" parameter may be specified, which is the level of access to the privilege required (default is 0, or any access).

nwa_replace

`{nwa_replace 1=... 2=...} ... {/nwa_replace}`

Smarty registered block function. Replace %1, %2, etc with the passed parameters 1=, 2=, etc.

Usage example:

```
{nwa_replace 1=$param1 2=$param2 ...}
This is the text resource to be replaced, where %1 and %2
are the arguments, etc.
{/nwa_replace}
```

The numbered parameters are expanded in the translated string with the positional arguments **%1**, **%2** and so forth.

nwa_text

`{nwa_text} … {/nwa_text}`

Smarty registered block function. Translates the block's content, if a language pack is available.

Usage example:

```
{nwa_text id=TEXT_ID 1=$param1 2=$param2 ...}
This is the text resource to be translated, where %1 and %2 are the arguments, etc.
{/nwa_text}
```

- The "id" parameter is the text ID of the resource.
- The numbered parameters are expanded in the translated string with the positional arguments **%1**, **%2** and so forth.

nwa_userpref

`{nwa_userpref …}`

Smarty template function. Returns the current setting of a user preference (stored with the Web application user account)

Usage examples:

```
{nwa_userpref name=prefName}
{nwa_userpref name=prefName default=10}
{nwa_userpref has=prefName}
```

- "name": return the named user preference
- "default": supply a value to be returned if the preference is not set
- "has": return 1 if the named preference exists for the current user, 0 if the preference does not exist

nwa_youtube

`{nwa_youtube video=`*`ID`* `width=`*`cx`* `height=`*`cy`* `…} … {/nwa_youtube}`

Smarty registered block function. Provides simple support for embedding a YouTube video in the body of a page. The content of this block is the initial "alternate content" that will be presented until the YouTube player can be embedded (if it can be embedded).

Not all devices are capable of playing back YouTube video content.

Usage example:

```
{nwa_youtube video=Y7dpJ0oseIA width=320 height=240}
YouTube is the world's most popular online video community.
{/nwa_youtube}
```

The supported parameters for this block function are:

- **video** (required) – the YouTube video ID to embed.
- **width** (required) – the width in pixels of the video.
- **height** (required) – the height in pixels of the video.

- **autoplay** (optional) – if true, auto-play the video.
- **chrome** (optional) – if true, use the chromed player; that is, provide a user experience with playback controls.
- **version** (optional) – the minimum version required to play the video.
- **onended** (optional) – the name of a global function (that is, a member of the JavaScript "window" object) that is to be called at the end of video playback.

Date/Time Format Syntax

There are two basic modifiers available for you to use in ClearPass Guest: nwadateformat and nwatimeformat.

nwadateformat Modifier

The date format takes one or two arguments – the format description and an optional default value (used if there is no time/date to display). UTF-8 is the character encoding used throughout the application, as this covers languages such as Spanish that use non-ASCII characters.

The full list of special formats is:

Table 28 *Date and Time Formats*

Preset Name	Date/Time Format	Example
hhmmss	%H%M%S	141345
hh:mm:ss	%H:%M:%S	14:13:45
iso8601	%Y%m%d	20080407
iso8601t	%Y%m%d%H%M%S	20080407141345
iso-8601	%Y-%m-%d	2008-04-07
iso-8601t	%Y-%m-%d %H:%M:%S	2008-04-07 14:13:45
longdate	%A, %d %B %Y, %I:%M %p	Monday, 07 April 2008, 2:13 PM
rfc822	%a, %d %b %Y %H:%M:%S %Z	Mon, 07 Apr 2008 14:13:45 EST
displaytime	%I:%M %p	2:13 PM
recent	–	2 minutes ago

The % items on the right hand side are the same as those supported by the php function strftime().

The string "**?:**", if present will return the string following the "?:" if the time value is 0. Otherwise, the format string up to the "?:" is used.

See **"Date/Time Format String Reference"** in this chapter for a full list of the supported date/time format string arguments.

Examples of date formatting using the nwadateformat Smarty modifier are as follows:

{$u.expire_time|nwadateformat:"longdate"}
Monday, 07 April 2008, 2:13 PM

{$u.expire_time|nwadateformat:"iso8601"}
20080407

{$u.expire_time|nwadateformat:"iso-8601t"}
2008-04-07 14:13:45

{$u.expire_time|nwadateformat:"iso8601?:N/A"}
20080407 (or **N/A** if no time specified)

{$u.expire_time|nwadateformat:"%m/%d/%Y"}
04/07/2008

nwatimeformat Modifier

The nwatimeformat modifier takes one argument – the format description. The "minutes_to_natural" argument converts an argument specified in minutes to a text string describing an equivalent but more natural measurement for the time interval (hours, days or minutes depending on the value). An example of this usage is for the expire_postlogin field which has a value measured in minutes:

```
{$u.expire_postlogin|nwatimeformat:"minutes_to_natural"}
```

The other formats accepted for this modifier are the same as those described for the nwadateformat modifier. See "nwadateformat Modifier".

Date/Time Format String Reference

Table 29 *Date and Time Format Strings*

Format	Result
%a	Abbreviated weekday name for the current locale
%A	Full weekday name for the current locale
%b	Abbreviated month name for the current locale
%B	Full month name for the current locale
%c	Preferred date and time representation for the current locale
%C	Century number (2-digit number, 00 to 99)
%d	Day of the month as a decimal number (01 to 31)
%D	Same as %m/%d/%y
%e	Day of the month as a decimal number; a single digit is preceded by a space (‘ 1’ to ‘31’)
%h	Same as %b
%H	Hour as a decimal number (00 to 23)
%I	Hour as a decimal number (01 to 12)
%m	Month as a decimal number (01 to 12)
%M	Minute as a decimal number (00 to 59)
%p	“AM” or “PM”

Table 29 *Date and Time Format Strings (Continued)*

%r	Local time using 12-hour clock (%I:%M %p)
%R	Local time using 24-hour clock (%H:%M)
%S	Second as a decimal number (00 to 60)
%T	Current time (%H:%M:%S)
%u	Weekday as a decimal number (1=Monday…7=Sunday)
%w	Weekday as a decimal number (0=Sunday…6=Saturday)
%x	Preferred date representation for the current locale, without the time
%X	Preferred time representation for the current locale, without the date
%y	Year as a decimal number without the century (00 to 99)
%Y	Year as a decimal number
%%	A literal % character

Programmer's Reference

NwaAlnumPassword

```
NwaAlnumPassword($len)
```

Generates an alpha-numeric password (mixed case) of length `$len` characters.

NwaBoolFormat

```
NwaBoolFormat($value, $options = null)
```

Formats a boolean value as a string. If 3 function arguments are supplied, the 2nd and 3rd arguments are the values to return for false and true, respectively. Otherwise, the `$options` parameter specifies how to do the conversion:

- If an integer 0 or 1, the string values "**0**" and "**1**" are returned.
- If a string containing a "|" character, the string is split at this separator and used as the values for false and true respectively.
- If an array, the 0 and 1 index values are used for false and true values.
- Otherwise, the string values "true" and "false" are returned.

NwaByteFormat

```
NwaByteFormat($bytes, $unknown = null)
```

Formats a non-negative size in bytes as a human readable number (bytes, KB, MB, GB, etc.) Assumes that 1 KB = 1024 bytes, 1 MB = 1024 KB, etc. If a negative value is supplied, returns the `$unknown` string. If a non-numeric value is supplied, that value is returned directly.

NwaByteFormatBase10

```
NwaByteFormatBase10($bytes, $unknown = null)
```

Formats a non-negative size in bytes as a human readable number (bytes, KB, MB, GB, etc.) Assumes "base 10" rules in measurement; that is, 1 KB = 1000 bytes, 1 MB = 1000 KB, etc. If a negative value is supplied, returns the `$unknown` string. If a non-numeric value is supplied, that value is returned directly.

NwaComplexPassword

```
NwaComplexPassword($len = 8)
```

Generates complex passwords of at least `$len` characters in length, where `$len` must be at least 4. A complex password includes at least 1 each of a lower case character, upper case character, digit, and punctuation (symbol).

NwaCsvCache

```
NwaCsvCache($csv_file, $use_cache = true, $options = null)
```

Loads and parses the contents of a CSV file, using a built-in cache. The cache may be cleaned for a specific file by setting `$use_cache` to false. The cache may be cleaned for ALL files by setting `$csv_file` to the empty string and `$use_cache` to false.

CSV parsing options (see **"NwaParseCsv"**) may be specified in `$options`. Additionally, a 2-argument form of this function may be used by passing an array of `$options` as the second argument; in this case, `$use_cache` is assumed to be true. This function returns false if the file does not exist; otherwise, returns an array of arrays containing each of the parsed records from the file.

NwaDigitsPassword($len)

```
NwaDigitsPassword($len)
```

Generates digit-only passwords of at least $len characters in length.

NwaDynamicLoad

```
NwaDynamicLoad($func)
```

Loads the PHP function `$func` for use in the current expression or code block. Returns true if the function exists (that is, the function is already present or was loaded successfully), or false if the function does not exist.

Attempting to use an undefined function will result in a PHP Fatal Error. Use this function before using any of the standard Nwa…() functions.

NwaGeneratePictureString

```
NwaGeneratePictureString($string)
```

Creates a password based on a format string. For details on the special characters recognized in $string, see **"Format Picture String Symbols"**.

NwaGenerateRandomPasswordMix

```
NwaGenerateRandomPasswordMix($password_len, $lower = 1, $upper = 1, $digit = 1,
$symbol = -1)
```

Generates a random password that meets a certain minimum complexity requirement.

- `$password_len` specifies the total length in characters of the generated password. The password returned will be at least `$upper + $lower + $digit + $symbol` characters in length. Any length beyond the required minimum will be made up of any allowed characters.

- `$lower` specifies the minimum number of lowercase characters to include, or -1 to not use any lowercase characters.
- `$upper` specifies the minimum number of uppercase characters to include, or -1 to not use any uppercase characters.
- `$digit` specifies the minimum number of digits to include, or -1 to not use any digits.
- `$symbol` specifies the minimum number of symbol characters to include, or -1 to not use any symbol or punctuation characters.

NwaLettersDigitsPassword

```
NwaLettersDigitsPassword($len)
```

Generates an alpha-numeric password of `$len` characters in length consisting of lowercase letters and digits.

NwaLettersPassword

```
NwaLettersPassword($len)
```

Generates a password of `$len` characters in length consisting of lowercase letters.

NwaMoneyFormat

```
NwaMoneyFormat($amount, $format = null)
```

Formats a monetary amount for display purposes. The current page language is used to adjust formatting to the country specified. Returns a result that is guaranteed to be in UTF-8.

The `$format` argument may be null, to specify the default behavior (U.S. English format), or it may be a pattern string containing the following:

- currency symbol (prefix)
- thousands separator
- decimal point
- number of decimal places

The format **"€1.000,00"** uses the Euro sign as the currency symbol, "**.**" as the thousands separator, "**,**" as the decimal point, and 2 decimal places.

If not specified explicitly, the default format is "**$1,000.00**".

NwaParseCsv

```
NwaParseCsv($text, $options = null)
```

Parses text containing comma-separated values and returns the result as a list of records, where each record contains a list of fields. Supports CSV escaping using double quotes.

`$options` may be specified to control additional parsing options described in the table below.

Table 30 *Parsing Options*

Function	Description
fs	The field separator character (default is comma ",")
rs	The record separator character (default is newline "\n")
quo	The quote character (default is double quote ")

Table 30 *Parsing Options (Continued)*

Function	Description
excel_compatible	If true, recognize ="..." syntax as well as "..." (default true)
dos_compatible	If true, convert \r\n line endings to \n (default true)
encoding	If set, specifies the input character set to convert from (default not set)
out_charset	If set, specifies the desired character set to convert to using the iconv() function . (default is "UTF-8//TRANSLIT")
max_records	maximum number of records to return
max_fields	maximum number of fields per record
skip_records	number of records to skip at start of input
skip_fields	number of fields to skip at start of each record
sort	post-processing option; order string for NwaCreateUsortFunc to sort the records by the specified column(s)
slice_offset	post-processing option: starting offset of slice to return; see array_slice() function
slice_length	post-processing option: length of slice to return; see array_slice() function

See **"NwaParseCsv"** and **"NwaVLookup"**.

NwaParseXml

```
NwaParseXml($xml_text)
```

Parses a string as an XML document and returns the corresponding document structure as an associative array. Returns an array containing the following elements:

- **error** – set if there was a problem parsing the XML
- **message** – describes the parse error

Otherwise, the return is an array with these elements:

- **name** – name of the document element
- **attributes** – attributes of the document element
- **children** – array containing any child elements
- **content** – element content text

NwaPasswordByComplexity

```
NwaPasswordByComplexity($len, $mode = false)
```

Generates a random password of at least `$len` characters in length, based on one of the standard complexity requirements specified in `$mode`. If `$mode` is false or the empty string, the default password complexity is taken from the Guest Manager plugin configuration.

Otherwise, $mode should be one of the following values:

- **none** – No password complexity requirement
- **case** – At least one uppercase and one lowercase letter
- **number** – At least one digit

- **punctuation** – At least one symbol
- **complex** – At least one of each: uppercase letter, lowercase letter, digit, and symbol

NwaSmsIsValidPhoneNumber

```
NwaSmsIsValidPhoneNumber($phone_number)
```

Validates a phone number supplied in E.164 international dialing format, including country code.

- Any spaces and non-alphanumeric characters are removed.
- If the first character is a plus sign (+), the phone number is assumed to be in E.164 format already and the plus sign is removed; otherwise, if the SMS service handler national prefix is set and the phone number starts with that prefix, then the prefix is replaced with the country code.
- The phone number must contain no fewer than 5 and no more than 15 digits.
- The phone number is validated for a valid country code prefix.
- If all the foregoing conditions are met, the validator returns TRUE; otherwise, the validator returns FALSE.

NwaStrongPassword

```
NwaStrongPassword($len)
```

Generate strong passwords of `$len` characters in length.

A strong password may contain uppercase letters, lowercase letters, digits and certain symbols. The strong password does not contain commonly-confused characters such as “O” and “0” (capital O and zero), “I” and “l” (capital I and lowercase L), “2” and “Z” (two and capital Z), or “8” and “B” (eight and capital B).

NwaVLookup

```
NwaVLookup($value, $table, $column_index, $range_lookup = true, $value_column = 0,
$cmp_fn = null)
```

Table lookup function, similar to the Excel function VLOOKUP(). This function searches for a value in the first column of a table and returns a value in the same row from another column in the table. This function supports the values described in the table below.

Table 31 *NwaVLookup Options*

Option	Description
$value	The value to look for
$table	A 2D array of data to search; for example, a data table returned by NwaCsvCache() or NwaParseCsv()
$column_index	The desired index of the data
$range_lookup	Specifies whether to find an exact or approximate match. If true (default), assumes the table is sorted and returns either an exact match, or the match from the row with the next largest value that is less than `$value`. If false, only an exact match is returned; NULL is returned on no match
value_column	Specifies the column index in the table that contains the values; the default is 0; in other words, the first column.

Table 31 *NwaVLookup Options (Continued)*

Option	Description
$cmp_fn	Specifies a comparison function to use for values; if null, the default is used (simple equality operator ==, or the == and > operators if using binary search). The comparison function should take 2 arguments and return a value < 0, == 0, > 0 depending on the sort ordering of the arguments.

Note the following differences from Excel VLOOKUP:

- Column indexes are 0-based.
- Column indexes can also be strings.

See **"NwaParseCsv"** and **"NwaCsvCache"**.

NwaWordsPassword

```
NwaWordsPassword($len)
```

Generates a password consisting of two randomly-chosen words, separated by a small number (1 or 2 digits); that is, in the format **word1***XX***word2**. The random words selected will have a maximum length of `$len` characters, and a minimum length of 3 characters. `$len` must be at least 3.

Field, Form, and View Reference

GuestManager Standard Fields

The table below describes standard fields available for the GuestManager form.

Table 32 *GuestManager Standard Fields*

Field	Description
account_activation	String. The current account activation time in long form. This field is available on the change_expiration and guest_enable forms. The value is generated from the **do_schedule** and **schedule_time** fields, and may be one of the following: • Account will be enabled at *date and time* • Account is currently active • No account activation
auto_update_account	Boolean flag indicating that an already existing account should be updated, rather than failing to create the account. This field should normally be enabled for guest self-registration forms, to ensure that a visitor that registers again with the same email address has their existing account automatically updated. Set this field to a non-zero value or a non-empty string to enable automatic update of an existing account. This field controls account creation behavior; it is not stored with created visitor accounts.
auto_update_account	Boolean flag indicating that an already existing account should be updated, rather than failing to create the account. This field should normally be enabled for guest self-registration forms, to ensure that a visitor that registers again with the same email address has their existing account automatically updated. Set this field to a non-zero value or a non-empty string to enable automatic update of an existing account. This field controls account creation behavior; it is not stored with created visitor accounts.

Table 32 *GuestManager Standard Fields (Continued)*

Field	Description
captcha	Special field used to enable the use of a CAPTCHA security code on a form. This field should be used with the user interface type “CAPTCHA security code” and the standard validator NwaCaptchaIsValid in order to provide the standard security code functionality.
change_of_ authorization	Boolean flag indicating that any existing sessions for a visitor account should be disconnected or modified using RFC 3576. If this field is not specified on a form that modifies the visitor account, the default value is taken from the configuration for the RADIUS Services plugin. Set this field to a non-zero value or a non-empty string to enable RFC 3576 updates for active sessions. Set this field to a zero value or the empty string to disable RFC 3576 updates for active sessions.
create_time	Integer. Time at which the account was created. The creation time is specified as a UNIX timestamp. This field is automatically configured with the current time when the Initial Value is set to: `array('generator' => 'time')`
creator_accept_terms	Boolean flag indicating that the creator has accepted the terms and conditions of use. When creating an account, this field must be present, and must be set to the value **1**. If this field is unset, or has any other value, account creation will fail with an error message. To set the correct value for this field, use a check box (to require confirmation from the creator) or a hidden field (if use of the form is considered acceptance of the terms and conditions). This field controls account creation behavior; it is not stored with created visitor accounts.
creator_name	String. Name of the creator of the account. This field does not have a default value. See “sponsor_name”.
do_expire	Integer that specifies the action to take when the expire time of the account is reached. See “expire_time”. • 0—Account will not expire • 1—Disable • 2—Disable and logout • 3—Delete • 4—Delete and logout “Disable” indicates that the enabled field will be set to 0, which will prevent further authorizations using this account. “Logout” indicates that a RADIUS Disconnect-Request will be used for all active sessions that have a username matching the account username. This option requires the NAS to support RFC 3576 dynamic authorization. See “RFC 3576 Dynamic Authorization” in the “Guest Manager” chapter for more information.
do_schedule	Boolean flag indicating if the account should be enabled at schedule_time. Set this field to 0 to disable automatic activation of the account at the activation time. Set this field to 1, and provide a valid time in the schedule_time field, to automatically enable the account at the specified activation time. See “schedule_time”.
dynamic_expire_time	Integer. Time at which the account will expire, calculated according to the account’s expiration timers. The value of this field is a UNIX timestamp. This field is available when modifying an account using the change_expiration or guest_edit forms.
dynamic_is_authorized	Boolean flag indicating if the user account is authorized to log in. This field is available when modifying an account using the change_expiration or guest_edit forms.
dynamic_is_expired	Boolean flag indicating if the user account has already expired. This field is available when modifying an account using the change_expiration or guest_edit forms.

Table 32 *GuestManager Standard Fields (Continued)*

Field	Description
dynamic_session_time	Integer. The maximum session time that would be allowed for the account, if an authorization request was to be performed immediately. Measured in seconds. Set to 0 if the account is either unlimited (dynamic_is_expired is false), or if the account has expired (dynamic_is_expired is true). This field is available when modifying an account using the change_expiration or guest_edit forms.
email	String. Email address for the account. This field may be up to 100 characters in length. When creating an account, if the **username** field is not set then the **email** field is used as the username of the account.
enabled	Boolean flag indicating if the account is enabled. Set this field to **0** to disable the account. If an account is disabled, authorization requests for the account will always fail. Set this field to **1** to enable the account.
expiration_time	String. Description of the account's expiration time. This field is set when modifying an account. This field is available on the change_expiration and guest_enable forms. The value is generated from the **do_expire**, **expire_time**, **expire_postlogin** and **expire_usage** fields, and may be one of the following: • Account will expire at *date and time*, or *interval* after first login, or after *interval* total usage • Account will expire at *date and time* or *interval* after first login • Account will expire at *date and time* or after *interval* total usage • Account will expire at *date and time* • Expires *interval* after first login or after *interval* total usage • Expires *interval* after first login • Expires after *interval* total usage • No expiration time set
expire_time	Integer. Time at which the account will expire. The expiration time should be specified as a UNIX timestamp. Setting an **expire_time** value also requires a non-zero value to be set for the **do_expire** field; otherwise, the account expiration time will not be used. Set this field to **0** to disable this account expiration timer.
expire_usage	Integer. The total time period in seconds for which the account may be used. Usage is calculated across all accounting sessions with the same username. Set this field to **0** to disable this account expiration timer.
http_user_agent	String. Identifies the Web browser that you are using. This tracks user's browsers when they are registering. This is stored with the user's account.
id	String. Internal user ID used to identify the guest account to the system.
ip_address	String. The IP address to assign to stations authenticating with this account. This field may be up to 20 characters in length. The value of this field is not currently used by the system. However, a RADIUS user role may be configured to assign IP addresses using this field by adding the Framed-IP-Address attribute, and setting the value for the attribute to: `<?= $user["ip_address"]`

Table 32 *GuestManager Standard Fields (Continued)*

Field	Description
modify_expire_postlogin	String Value indicating how to modify the **expire_postlogin** field. This field is only of use when editing a visitor account. It may be set to one of the following values: • "expire_postlogin" to set the post-login expiration time to the value in the expire_postlogin field; • "plus *X*" or "minus *X*", where *X* is a time measurement, to extend or reduce the post-login expiration timer by *X* (minutes, but may have a "ywdhms" suffix to indicate years, weeks, days, hours, minutes, seconds respectively); • A number, to set the post-login expiration time to the value specified; • Any other value to leave expire_postlogin unmodified. This field controls account modifications; it is not stored with the visitor account.
modify_expire_time	String. Value indicating how to modify the **expire_time** field. This field may be provided when creating or editing a visitor account. It may be set to one of the following values: • "none" to disable the account expiration timer (**do_expire** and **expire_time** will both be set to 0); • "now" to disable the account immediately; • "expire_time" to use the expiration time specified in the **expire_time** field; • "expire_after" to set the expiration time to the current time, plus the number of hours in the **expire_after** field; • "plus *X*" or "minus *X*", where *X* is a time measurement, to extend or reduce the expiration time by *X* (hours, but may have a "ywdhms" suffix to indicate years, weeks, days, hours, minutes, seconds respectively); • A time measurement "*X*", to set the expiration time to the current time plus *X*; • Any other value to leave expire_time unmodified. This field controls account creation and modification behavior; it is not stored with created or modified visitor accounts.
modify_expire_usage	String. Value indicating how to modify the **expire_usage** field. This field is only of use when editing a visitor account. It may be set to one of the following values: • "expire_usage" to set the cumulative usage expiration timer to the value in the expire_usage field; • "plus ***X***" or "minus ***X***", where ***X*** is a time measurement, to extend or reduce the cumulative usage expiration timer by ***X*** (seconds, but may have a "ywdhms" suffix to indicate years, weeks, days, hours, minutes, seconds respectively); • A number, to set the cumulative usage expiration time to the value specified; • Any other value to leave expire_usage unmodified. This field controls account modifications; it is not stored with the visitor account.
modify_password	String. Value indicating how to modify the account password. • It may be one of the following values: • "random_password" to use the password specified in the **random_password** field; • "reset" to create a new password, using the method specified in the **random_password_method** field (or the global defaults, if no value is available in this field); • "password" to use the value from the **password** field; • Any other value leaves the password unmodified. This field controls account creation and modification behavior; it is not stored with created or modified visitor accounts.

Table 32 *GuestManager Standard Fields (Continued)*

Field	Description
modify_schedule_time	String. Value indicating how to modify the **schedule_time** field. It may be one of the following values: • "none" to disable the account activation time; • "now" to activate the account immediately; • "schedule_time" to use the activation time specified in the **schedule_time** form field (normally a UNIX time, but may be 0 to disable activation time); • "schedule_after" to set the activation time to the current time plus the number of hours in the **schedule_after** field; • "plus *X*", where *X* is a time measurement, to extend the activation time by *X*. The time measurement is normally hours, but may have a "ywdhms" suffix to indicate years, weeks, days, hours, minutes, or seconds, respectively. Alternatively, this operation may be written equivalently as '+X', 'pX', 'plusX', 'add X', 'addX', or 'aX'. Example: to delay activation time by 2 days, use the value **+2d**. • "minus *X*", where *X* is a time measurement, to reduce the activation time by *X*. See above for details about specifying a time measurement. Alternatively, this operation may be written equivalently as '-X', 'mX', 'minusX', 'sub X', 'subX', or 'sX'. Example: to bring forward activation time by 12 hours, use the value **-12h**. • A time measurement "*X*", to set the activation time to the current time plus *X*. • A time and date specification, to set the activation time to that time and date. Many different formats are specified; for clarity it is recommended that a standard format such as ISO-8601 is used ("YYYY-MM-DD hh:mm:ss" format). • Any other value to leave **schedule_time** unmodified. This field controls account creation and modification behavior; it is not stored with created or modified visitor accounts.
multi_initial_sequence	Integer. Initial sequence number. This field is used when creating guest accounts and the **random_username_method** field is set to "nwa_sequence". If this field is not set, the next available sequence number for the given **multi_prefix** is used. Sequence numbering will start with 0 if no initial sequence number has been set.
multi_prefix	String. The prefix of each username generated when creating guest accounts and the **random_username_method** field is set to "nwa_sequence".
netmask	String. Network address mask to use for stations using the account. This field may be up to 20 characters in length. The value of this field is not currently used by the system. However, a RADIUS user role may be configured to assign network masks using this field by adding the Framed-IP-Netmask attribute, and setting the value for the attribute to: `<?= $user["netmask"]`
no_password	Boolean. If set, prevents a user from changing their own password using the guest self-service portal. Set this field to a non-zero value or a non-empty string to disable guest-initiated password changes. The default is to allow guest-initiated password changes, unless this field is set.
no_portal	Boolean. If set, prevents a user from logging into the guest service portal. Set this field to a non-zero value or a non-empty string to disable guest access to the self-service portal. The default is to allow guest access to the self-service portal, unless this field is set.
no_warn_before	Boolean. User does not receive a logout expiration warning. The admin or user can opt out of this option by setting the field to 1.
notes	String. Comments or notes stored with the account. This field may be up to 255 characters in length.
num_accounts	Integer. The number of accounts to create when using the **create_multi** form. This field controls account creation behavior; it is not stored with created visitor accounts.
password	String. Password for the account. This field may be up to 64 characters in length.

Table 32 *GuestManager Standard Fields (Continued)*

Field	Description
password2	String. Password for the account. If this field is set, its value must match the value of the **password** field for the account to be created or updated. This can be used to verify that a password has been typed correctly. This field controls account creation and modification behavior; it is not stored with created or modified visitor accounts.
password_action	String. Controls the password changing behavior for a guest account. This field may be set to one of the following values: • *empty string* – Default behavior; that is, guests are not required to change their password • **deny** – Prevents the guest from changing their password • **first** – Requires the guest to change their password on their first login • **next** – Requires the guest to change their password on their next login • **recur** – Require the guest to change their password on a regular schedule (as specified by the password_action_recur field) • **recur_next** – Require the guest to change their password on their next (or first) login, and then on a regular schedule (as specified by the password_action_recur field) If the guest is required to change their password, this will take place during a network login, before the guest is redirected to the NAS for login. Guest password changes are only supported for Web login pages and guest self-registration pages that have the “Perform a local authentication check” option enabled. The default behavior is to leave guest passwords under the control of the guest. With the default behavior, guests are not prevented from changing their password, but are also not required to change it on any particular schedule.
password_action_recur	String. Specifies a date or relative time, after which a guest will be required to change their password. Using this field also requires the **password_action** field to be set to the value ‘recur’. The value of this field should be a relative time measurement, indicated with a plus sign; for example “+15 days” or “+2 months”.
password_last_change	Integer. The time that the guest’s password was last changed. The password change time is specified as a UNIX timestamp. This field is automatically updated with the current time when the guest changes their password using the self-service portal.
random_password	String. This field contains a randomly-generated password. This field is set when modifying an account (**guest_edit** form).
random_password_length	String. The length, in characters, of randomly generated account passwords. • For **nwa_words_password,** the **random_password_length** is the maximum length of the random words to use. Two random words will be used to create the password, joined together with a small number (up to 2 digits). • For **nwa_picture_password**, the **random_password_length** is ignored.

Table 32 *GuestManager Standard Fields (Continued)*

Field	Description
random_password_method	String. Identifier specifying how passwords are to be created. It may be one of the following identifiers: • **nwa_digits_password** to create a password using random digits. The length of the password is specified by the random_password_length field. • **nwa_letters_password** to create a password using random lowercase letters (a through z). The length of the password is specified by the random_password_length field. • **nwa_lettersdigits_password** to create a password using random lowercase letters and digits (a through z and 0 through 9). The length of the password is specified by the random_password_length field. • **nwa_alnum_password** to create a password using a combination of random digits, uppercase letters and lowercase letters (a-z, A-Z and 0-9). The length of the password is specified by the random_password_length field. • **nwa_strong_password** to create a password using a combination of digits, uppercase letters, lowercase letters, and some punctuation. Certain characters are omitted from the password. The length of the password is specified by the random_password_length field. • **nwa_complex_password** to create a complex password string which contains uppercase letters, lowercase letters, digits and symbol characters. • **nwa_complexity_password** is dynamic and matches your complexity setting for password generation. For example, if you require your passwords to have both letters and digits, then this validator will confirm that the password has at least one of each. • **nwa_words_password** to create a random password using a combination of two randomly-selected words and a number between 1 and 99. The maximum length of each of the randomly-selected words is specified by the **random_password_length** field. • **nwa_picture_password** to create a password using the format string specified by the random_password_picture field.
random_password_picture	String. The format string to use when creating a random password, if **random_password_method** is set to “nwa_picture_password”.
random_username_length	The length, in characters, of randomly generated account usernames. • For nwa_words_password, the **random_username_length** is the maximum length of the random words to use. Two random words will be used to create the username, joined together with a small number (up to 2 digits). • For nwa_picture_password, the **random_username_length** is ignored. • For nwa_sequence, the **random_username_length** is the length of the sequence number in the username; the sequence number will be zero-padded. For example, specifying a length of 4 will result in sequence numbers 0001, 0002, etc.

Table 32 *GuestManager Standard Fields (Continued)*

Field	Description
random_username_method	String. Identifier specifying how usernames are to be created. It may be one of the following identifiers: • **nwa_sequence** to assign sequential usernames. In this case, the **multi_prefix** field is used as the prefix for the username, followed by a sequential number; the number of digits is specified by the **random_username_length** field. • **nwa_picture_password** to create a random username using the format string specified by the **random_username_picture** field. • **nwa_digits_password** to create a username using random digits. The length of the username is specified by the **random_username_length** field. • **nwa_letters_password** to create a username using random lowercase letters. The length of the username is specified by the **random_username_length** field. • **nwa_lettersdigits_password** to create a username using random lowercase letters and digits. The length of the username is specified by the **random_username_length** field. • **nwa_alnum_password** to create a username using a combination of random digits, uppercase letters and lowercase letters. The length of the username is specified by the **random_username_length** field. • **nwa_strong_password** to create a username using a combination of digits, uppercase letters, lowercase letters, and some punctuation. Certain characters are omitted from the generated username to ensure its readability (for example, “o”, “O” and “0”). The length of the username is specified by the **random_username_length** field. • **nwa_words_password** to create a username using a combination of two randomly-selected words and a number between 1 and 99. The maximum length of each of the randomly-selected words is specified by the **random_username_length** field.
random_username_picture	String. The format string to use when creating a username, if the **random_username_method** field is set to **nwa_picture_password.** See **“Format Picture String Symbols”** for a list of the special characters that may be used in the format string.
remote_addr	String. The IP address of the guest at the time the guest account was registered.This field may be up to 20 characters in length. The value of this field is not currently used by the system.
role_id	Integer. Role to assign to the account.The value of this field must be the integer ID of a valid RADIUS user role.
role_name	String. Name of the role assigned to the account.
schedule_after	Integer. Time period, in hours, after which the account will be enabled. This field is used when the **modify_schedule_time** field is set to **schedule_after**. The value is specified in hours and is relative to the current time. This field controls account creation behavior; it is not stored with created visitor accounts.
schedule_time	Integer. Time at which the account will be enabled. The time should be specified as a UNIX timestamp.
secret_answer	String. The guest’s answer to the secret question that is stored in the secret_question field. To use this field, first add both the **secret_question** and **secret_answer** fields to a guest self-registration form. Then, in the self-service portal for a guest self-registration page, select the “Secret Question” as the Required Field. This configuration requires that guests provide the correct answer in order to reset their account password. Answers must match with regards to case in order to be considered as correct.
secret_question	String. The guest’s secret question used to confirm the identity of a guest during a reset password operation.

Table 32 *GuestManager Standard Fields (Continued)*

Field	Description
simultaneous_use	Integer. Maximum number of simultaneous sessions allowed for the account.
sponsor_email	Email address of the sponsor of the account. If the **sponsor_email** field can be inserted into an email receipt and used future emails, the “Reply-To” email address will always be the email address of the original sponsor, not the current operator.
sponsor_name	String. Name of the sponsor of the account. The default value of this field is the username of the current operator.
submit	No Type. Field attached to submit buttons. This field controls account creation behavior; it is not stored with created visitor accounts.
user_activity	Integer. Login activity of the guest account. This field is available in views and may be used to determine the most recent start and stop time of visitor account sessions.
username	String. Username of the account. This field may be up to 64 characters in length.
visitor_company	String. The visitor’s company name.
visitor_name	String. The visitor’s full name.
vvisitor_phone	String. The visitor’s contact telephone number.

Hotspot Standard Fields

The table below describes standard fields available for the Hotspot form.

Table 33 *Hotspot Standard Fields*

Field	Description
address	String. The visitor’s street address.
card_code	String. The 3 or 4 digit cardholder verification code printed on the credit card. This field is only used during transaction processing.
card_expiry	String. Credit card expiry date. This field is only used during transaction processing.
card_name	String. Name shown on the credit card. This field is only used during transaction processing.
card_number	String. Credit card number. This field is only used during transaction processing.
city	String. The visitor’s city or town name.
country	String. The visitor’s country name.
first_name	String. The visitor’s first name.
hotspot_plan_id	No Type. The ID of the plan (visitor access settings) selected by the visitor.
hotspot_plan_name	No Type. The name of the plan (visitor access settings) selected by the visitor.
last_name	String. The visitor’s last name.

Table 33 *Hotspot Standard Fields (Continued)*

Field	Description
password2	String. Password for the account (used to confirm a manually typed password).
personal_details	No Type. Field attached to a form label.
purchase_amount	No Type. Total amount of the transaction. This field is only used during transaction processing.
purchase_details	No Type. Field attached to a form label.
state	String. The visitor's state or locality name.
submit_free	No Type. Field attached to a form submit button.
visitor_accept_terms	Boolean. Flag indicating that the visitor has accepted the terms and conditions of use.
visitor_fax	String. The visitor's fax telephone number.
zip	String. The visitor's zip or postal code.

SMS Services Standard Fields

The table below describes standard fields available for the SMS Services form.

Table 34 *SMS Services Standard Fields*

Field	Description
auto_send_sms	Boolean. Flag indicating that a SMS receipt should be automatically sent upon creation of the account.
sms_auto_send_field	String. This field specifies the name of the field that contains the auto-send flag. If blank or unset, the default value from the SMS plugin configuration is used. Additionally, the special values "_Disabled" and "_Enabled" may be used to never send an SMS or always send an SMS, respectively.
sms_enabled	Boolean. This field may be set to a non-zero value to enable sending an SMS receipt. If unset, the default value is true.
sms_handler_id	String. This field specifies the handler ID for the SMS service provider. If blank or unset, the default value from the SMS plugin configuration is used.
sms_phone_field	String. This field specifies the name of the field that contains the visitor's phone number. If blank or unset, the default value from the SMS plugin configuration is used.
sms_template_id	String. This field specifies the print template ID for the SMS receipt. If blank or unset, the default value from the SMS plugin configuration is used.
sms_warn_before_message	String. This field overrides the logout warning message. If blank or unset, the default value from the Customize SMS Receipt page is used
visitor_carrier	String. The visitor's mobile phone carrier.

SMTP Services Standard Fields

The table below describes standard fields available for the SMTP Services.

Table 35 *SMPT Services Standard Fields*

Field	Description
auto_send_smtp	Boolean. Flag indicating that an email receipt should be automatically sent upon creation of the guest account. Set this field to a non-zero value or a non-empty string to enable an automatic email receipt to be sent. This field can be used to create an *opt-in* facility for guests. Use a check box for the auto_send_smtp field and add it to the create_user form, or a guest self-registration instance, and email receipts will be sent to the visitor only if the check box has been selected. Alternatively, to always send an SMTP receipt, this field can be set to a value of 1 using a hidden field.
smtp_auto_send_field	String. This field specifies the name of the field that contains the auto-send flag. If blank or unset, the default value from the email receipt configuration is used. Additionally, the special values **_Disabled** and **_Enabled** may be used to never send email or always send email, respectively.
smtp_cc_action	String. This field specifies how to send copies of email receipts. It may be one of **never**, **always_cc**, **always_bcc**, **conditional_cc**, or **conditional_bcc**. If blank or unset, the default value from the email receipt configuration is used.
smtp_cc_list	String. This field specifies a list of additional email addresses that will receive a copy of the visitor account receipt. If the value is **default**, the default carbon-copy list from the email receipt configuration is used.
smtp_email_field	String. This field specifies the name of the field that contains the visitor's email address. If blank or unset, the default value from the email receipt configuration is used. Additionally, the special value **_None** indicates that the visitor should not be sent any email.
smtp_enabled	String. This field may be set to a non-zero value to enable sending an email receipt. If unset, the default value from the email receipt configuration is used. The special values **_Auto** (Always auto-send guest receipts by email), **_AutoField** (Auto-send guest receipts by email with a special field set), **_Click** (Display a link enabling a guest receipt via email), and **_Cc** (Send an email to a list of fixed addresses) may also be used.
smtp_receipt_format	String. This field specifies the email format to use for the receipt. It may be one of **plaintext** (No skin – plain text only), **html_embedded** (No skin – HTML only), **receipt** (No skin – Native receipt format), **default** (Use the default skin), or the plugin ID of a skin plugin to specify that skin. If blank or unset, the default value from the email receipt configuration is used.
smtp_subject	String. This field specifies the subject line for the email message. Template variables appearing in the value will be expanded. If the value is **default**, the default subject line from the email receipt configuration is used.
smtp_template_id	String. This field specifies the print template ID to use for the email receipt. If blank or unset, the default value from the email receipt configuration is used.
smtp_warn_before_subject	String. This field overrides what is specified in the subject line under Logout Warnings on the email receipt. If the value is "default", the default subject line under the Logout Warnings section on the email receipt configuration is used.

Table 35 *SMPT Services Standard Fields (Continued)*

Field	Description
smtp_warn_before_template_id	String. This field overrides the print template ID specified under Logout Warnings on the email receipt. If the value is “default”, the default template ID under the Logout Warnings section on the email receipt configuration is used.
smtp_warn_before_receipt_format	String. This field overrides the format in the **Email Receipt** field under Logout Warnings. It may be one of “plaintext” (No skin – plain text only), “html_embedded” (No skin – HTML only), “receipt” (No skin – Native receipt format), “default” (Use the default skin), or the plugin ID of a skin plugin to specify that skin. If blank or unset, the default value in the **Email Receipt** Field under the Logout Warnings on the email receipt configuration is used.
smtp_warn_before_cc_list	String. This overrides the list of additional email addresses that receive a copy of the visitor account under Logout Warnings on the email receipt. If the value is “default”, the default carbon-copy list under Logout Warnings from the email receipt configuration is used.
smtp_warn_before_cc_action	String. This field overrides how copies are sent as indicated under Logout Warnings on the email receipt. to send copies of email receipts. It may be one of “never”, “always_cc”, “always_bcc”, “conditional_cc”, or “conditional_bcc”. If blank or unset, the default value from the email receipt configuration is used.
warn_before_from_sponsor	String. This field overrides the **Reply To** field (that is, the sponsor_email field of a user, or the admin's email) under the Logout Warnings on the email receipt. If the value is “default”, the **Reply To** field under Logout Warnings from the email receipt configuration is used.i
warn_before_from	String. This field overrides the Override From field under the Logout Warnings on the email receipt. If the value is “default”, the Override From field under Logout Warnings from the email receipt configuration is used.

Format Picture String Symbols

When generating a username or password using the `nwa_picture_password` method, a “picture string” should be provided to specify the format of generated username or password in the **random_username_picture** or **random_password_picture** field.

The picture string is used as the username or password, with the following symbols replaced with a random character:

Table 36 *Picture String Symbols*

Symbol	Replacement
#	Random digit (0-9)
$ or ?	Random letter (A-Z, a-z)
_	Random lowercase letter (a-z)
^	Random uppercase letter (A-Z)
*	Random letter or digit (A-Z, a-z, 0-9)
!	Random punctuation symbol, excluding apostrophe and quotation marks

Table 36 *Picture String Symbols (Continued)*

&	Random character (letter, digit or punctuation excluding apostrophe and quotation marks)
@	Random letter or digit, excluding vowels

Any other alphanumeric characters in the picture string will be used in the resulting username or password. Some examples of the picture string are shown below:

Table 37 *Picture String Example Passwords*

Picture String	Sample Password
####	3728
user####	user3728
v^^#__	vQU3nj
@@@@@	Bh7Pm

Form Field Validation Functions

See "Form Validation Properties" in the "Configuration" chapter, and "Examples of Form field Validation" in the "Configuration" chapter for details about using validation functions for form fields.

The built-in validator functions are:

- **IsArrayKey** – Checks that the value is one of the keys in the array supplied as the argument to the validator.
- **IsArrayValue** – Checks that the value is one of the values in the array supplied as the argument to the validator.
- **IsEqual** – Checks that the value is equal to the value supplied as the argument to the validator, allowing for standard type conversion rules.
- **IsGreaterThan** – Checks that the value is strictly greater than a specified minimum value supplied as the argument to the validator.
- **IsIdentical** – Checks that the value is equal to the value supplied as the argument to the validator, and has the same type.
- **IsInRange** – Checks that the value is in a specified range between a minimum and maximum value. The minimum and maximum values are specified as a 2-element array as the argument to the validator.
- **IsInOptionsList**—Checks against a list of options in the policy definition.
- **IsNonEmpty** – Checks that the value is a non-empty string (length non-zero and not all whitespace), or a non-empty array.
- **IsNonNegative** – Checks that the value is numeric and non-negative.
- **IsRegexMatch** – Checks that the value matches a regular expression supplied as the argument the validator. The regular expression should be a Perl-compatible regular expression with delimiters. For example, the validator argument **/^a/i** will match any value that starts with an "a", case-insensitively. See "Regular Expressions" in this chapter for more information about regular expression syntax.
- **IsValidBool** – Checks that the value is a standard Boolean truth value. Valid Boolean values are the integers **0** and **1** and the PHP values **false** and **true**.

- **IsValidDateTime** – Checks that the value appears to be a valid time specification string according to the rules of the PHP function strtotime(). Valid date/time syntax includes ISO 8601 standard times (**YYYY-MM-DD hh:mm:ss**) with and without time zone specifications, as well as many other formats.
- **IsValidEmail** – Checks that the value appears to be a valid RFC 822-compliant email address. When using the IsValidEmail validator, the validator argument may be specified wieth a whitelist/blacklist of domain names. Use the syntax:

```
array(
  'allow' => array(
    'corp-domain.com',
    'other-domain.com',
  ),
  'deny' => array(
    'blocked-domain.com',
    'other-blocked-domain.com',
  ),
)
```

  - The keys 'whitelist' and 'blacklist' may also be used for 'allow' and 'deny', respectively.
  - An 'allow' or 'deny' value that is a string is converted to a single element array.
  - Wildcard matching may be used on domain names: the prefix '*.' means match any domain that ends with the given suffix. A '*' component can also be used inside the hostname, and will match zero or more domain name components.
  - If the 'allow' list is empty or unset, the default behavior is to accept ALL domains other than those listed in the 'deny' list.
  - If the 'deny' list is empty or unset, the default behavior is to deny ALL domains other than those listed in the 'allow' list.
  - If both 'allow' and 'deny' lists are provided, the default behavior is to accept a domain name that does not match any of the patterns provided. The 'allow' list is checked first, followed by 'deny'. To obtain the opposite behavior, specify the wildcard '*' as the last entry in the 'deny' list.
- **IsValidFileUpload** – Checks that the value is a file upload.
- **IsValidFutureDateTime** – Checks that the value is a valid time specification string according to the rules of the PHP function strtotime(), and that the time specification refers to a point in the future.
- **IsValidFutureTimestamp** – Checks that the value is a valid UNIX time referring to a point in the future.
- **IsValidHostname** – Checks that the value is a valid IP address or a hostname that resolves to an IP address.
- **IsValidHostnameCidr** – Checks that the value is a valid IP address or hostname, which may also have an optional /N suffix indicating the network prefix length in bits (CIDR notation).
- **IsValidHostnamePort** – Checks that the value is a valid IP address or hostname, which may optionally include a port number specified with the syntax **hostname:port**.
- **IsValidIpAddr** – Checks that the value is a valid IP address.
- **IsValidLdapAttribute** – Checks that the value is a valid LDAP attribute name; that is, a string that starts with a letter, and which contains only letters, numbers, underscore (_) and hyphen (-).
- **IsValidNetmask** – Checks that the value is a valid network mask in dotted-quad notation; that is, an IP address such as 255.255.255.128 that contains a single string of *N* 1 bits followed by (32 – *N*) 0 bits.
- **IsValidNumber** – Checks that the value is numeric; that is, an integer or a decimal value. The validator argument may be an array containing one or more of the following additional options:
  - **no_negative** – if set to true, negative numbers are not accepted as a valid value.
  - **no_zero** – if set to true, zero is not accepted as a valid value.
  - **only_integer** – if set to true, decimal numbers are not accepted and only integer values are valid.

- **IsValidPassword2** – Checks that the value is a valid password that satisfies certain requirements. The validator argument must be an array describing which of the following requirements to check. To perform any password checking, the “minimum_length” and “complexity_mode” fields must be specified.
  - **password2** – specifies the name of the field containing the duplicate password entry (optional, for password validation). Defaults to “password2” if not specified.
  - **password2_required** – if nonzero, indicates that the “password2” entry must be supplied.
  - **username** – specifies the name of the field containing the username. If empty or unset, the password is not checked against this field for a match.
  - **minimum_length** – specifies the minimum length of the password in characters.
  - **disallowed_chars** – if set, specifies characters that are not allowed in the password.
  - **complexity_mode** – specifies the set of rules to use when checking the password.
  - **complexity** – if set, specifies rules for checking the composition of the password. If unset, defaults to a preset value for password complexity with modes “none”, “basic”, “number”, “punctuation” and “complex”. These rules check that passwords obey certain requirements according to the following table:

Table 38 *Complexity Requirements*

Rule Set	Min. Length	Description
none	–	No special requirements
basic	8	Non-space characters
number	8	At least 1 digit
punctuation	8	At least 1 punctuation character (non-alphanumeric)
complex	8	At least 1 digit, 1 non-alphanumeric, 1 uppercase and 1 lowercase letter

- **IsValidSentence** – Checks that the value is considered to be a ‘sentence”; that is, a string which starts with an upper-case letter and ends in a full stop.
- **IsValidTimestamp** – Checks that the value is a numeric UNIX timestamp (which measures the time in seconds since January 1, 1970 at midnight UTC).
- **IsValidTimeZone** – Checks that the value is a valid string describing a recognized time zone.
- **IsValidUrl** – Checks that the value appears to be a valid URL that includes a scheme, hostname and path. For example, in the URL http://www.example.com/, the scheme is **http**, the hostname is **www.example.com** and the path is **/**. The validator argument may optionally be an array containing a ‘scheme’ key that specifies an array of acceptable URL protocols.
- **IsValidUsername** – Checks that the value is a valid username. Usernames cannot be blank or contain spaces.
- **NwaCaptchaIsValid** – Checks that the value matches the security code generated in the CAPTCHA image. This validator should only be used with the standard **captcha** field.
- **NwaGuestManagerIsValidRoleId** – Checks that the value is a valid role ID for the current operator and user database.

- **NwaIsValidExpireAfter** – Checks that the value is one of the account expiration time options specified in the Guest Manager configuration.
- **NwaIsValidLifetime** – Checks that the value is one of the account lifetime options specified in the Guest Manager configuration.

Form Field Conversion Functions

The Conversion and Value Format functions that are available are listed below:

- **NwaConvertOptionalDateTime** – Converts a string representation of a time to the UNIX time representation (integer value). The conversion leaves blank values unmodified.
- **NwaConvertOptionalInt** – Converts a string representation of an integer to the equivalent integer value. The conversion leaves blank values unmodified.
- **NwaConvertStringToOptions** – Converts a multi-line string representation of the form

```
key1 | value1
key2 | value2
```

to the array representation

```
array (
  'key1' => 'value1',
  'key2' => 'value2',
)
```

- **NwaImplodeComma** – Converts an array to a string by joining all of the array values with a comma.
- **NwaTrim** – Removes leading and trailing whitespace from a string value.
- **NwaTrimAll** – Removes all whitespace from a string (including embedded spaces, newlines, carriage returns, tabs, etc).
- **NwaStrToUpper** – Formats the text string to all uppercase letters.
- **NwaStrToLower** – Formats the text string to all lowercase letters.
- **NwaNormalizePhoneNumber** – Removes all spaces, dashes, parenthesis and non-numerical characters from the phone number.

Form Field Display Formatting Functions

The Display Functions that are available are listed below:

Table 39 *Form Field Display Functions*

Function	Description
NwaBoolFormat	Formats a Boolean value as a string. • If the argument is 0 or 1, a 0 or 1 is returned for false and true, respectively. • If the argument is a string containing a "|" character, the string is split at the | separator and used for false and true values. • If the argument is an array, the 0 and 1 index values are used for false and true values. Otherwise, the string values "false" and "true" are returned.
NwaByteFormat	Formats a non-negative size in bytes as a human readable number (bytes, KB, MB, GB, etc). 1 KB is defined as 1,024 bytes, 1 MB as 1,024 KB (1,048,576 bytes), and 1 GB as 1,024 MB (1,073,741,824 bytes). • If a negative value is supplied, returns the argument (or null if no argument was supplied). • If a non-numeric value is supplied, that value is returned directly.

Table 39 *Form Field Display Functions (Continued)*

Function	Description
NwaCurrencyFormat	Formats a numeric value that indicates a monetary amount as a string. If the argument is null or not supplied, the current locale's settings are used to format the monetary value. • The argument may be an array, which will override the current locale's settings (see **NwaNumberFormat** for the list of settings that are used). • The argument may be a numeric value, which is used as the number of fractional digits to use when formatting the monetary amount (other locale settings will remain unchanged in this case).
NwaDateFormat	Format a date like the PHP function strftime(), using the argument as the date format string. Returns a result guaranteed to be in UTF-8 and correct for the current page language. See "Date/Time Format Syntax" for a list of available date/time formats, or use one of the following special format strings: • **hhmmss**, **hh:mm:ss** – time of day • **iso8601**, **iso8601t**, **iso-8601**, **iso-8601t** – various ISO 8601 date formats with and without hyphen separators and the time of day • **longdate** – date and time in long form • **displaytime** – time of day • **?:** – returns the string following the ?: if the time value is 0, or uses the format string before the ?: otherwise • **recent** – for example, "2 minutes ago", "3 months ago"
NwaDurationFormat	Converts a time measurement into a description of the corresponding duration. • Format parameters: seconds, minutes, hours, days, weeks. • Any format can be converted to another. • By default, this function converts an elapsed time value specified in seconds to a value that is displayed in weeks, days, hours, minutes and seconds. Up to four additional arguments may be supplied to control the conversion: • in_format – The current units of the value being converted (seconds, minutes, hours, days, weeks) • max_format – Controls the max increment you want displayed. • min_format – Controls the min increment you want displayed. Only whole numbers are printed. • default – If set, this value will be returned when the resulting duration (after min_format is taken into account) is 0.
NwaExplodeComma	Converts a string to an array by splitting the string at each comma and forming an array of all the substrings created in this way.

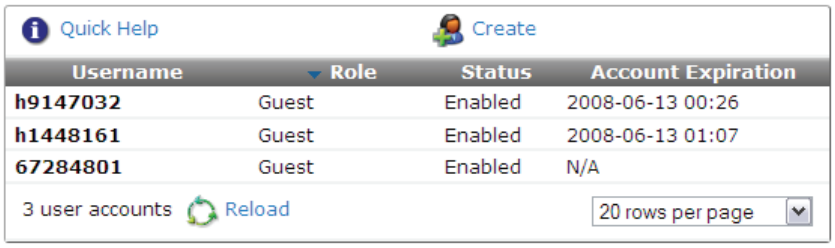
Table 39 *Form Field Display Functions (Continued)*

Function	Description
NwaNumberFormat	Formats a numeric value as a string. If the argument is null or not supplied, the current locale's settings are used to format the numeric value. The argument may be an array or a numerica value. If the argument is an array, it will override the current locale's settings (see below for the list of settings that are used). If the argument is a numeric value, it is used as the number of fractional digits to use when formatting the string (other locale settings will remain unchanged in this case). The specific locale settings used are from **localeconv(),** and are listed below. For *general numeric formatting* : • frac_digits – number of decimal places to display • decimal_point – character to use for decimal point • thousands_sep – character to use for thousands separator For *signs for positive/negative values:* • positive_sign – sign for positive values • p_sign_posn – position of sign for positive values (0..4) • negative_sign – sign for negative values • n_sign_posn – position of sign for negative values (0..4) For *formatting for monetary amounts*: • mon_decimal_point – decimal point character for monetary values • mon_thousands_sep – thousands separator for monetary values • p_sep_by_space – true if a space separates currency symbol from a positive value • p_cs_precedes – true if currency symbol precedes positive value • n_sep_by_space – true if a space separates currency symbol from a negative value • n_cs_precedes – true if currency symbol precedes negative value Additionally, the special value monetary, if true, indicates that a currency value should be formatted, rather than a regular numeric value.

View Display Expression Technical Reference

A page that contains a view is displayed in an operator's Web browser. The view con
tains data that is loaded from the server dynamically. Because of this, both data formatting and display operations for the view are implemented with JavaScript in the Web browser.

For each item displayed in the view, a JavaScript object is constructed. Each field of the item is defined as a property of this object. When evaluating the JavaScript Display Expression, the **data** variable is used to refer to this object. Thus, the expression **data.my_field** would return the value of the field named "my_field".



In the above view (the **guest_users** view), the four columns displayed correspond to the **username**, **role_name**, **enabled**, and **expire_time** fields.

Table 40 *Display Expressions for Data Formatting*

Value	Description
Display Expressions	
data.username.bold()	Displays the username string as bold text.
data.role_name	Displays the name of the role.
Nwa_BooleanText(data.enabled, "Enabled", "Disabled")	Displays either “Enabled” or “Disabled” depending on the value of the **enabled** field.
(parseInt(data.do_expire) != 0) ? Nwa_DateFormat(data.expire_time, "%Y-%m-%d %H:%M") : "N/A"	Displays “N/A” if the account has no expiration time, or a date and time string if an expiration time has been set.
JavaScript functions	
Nwa_BooleanText (*value***,** *if_true***,** *if_false***[,** *if_undefined***])**	Returns the value of *if_true* or *if_false* depending on whether the *value* evaluates to a Boolean true or false, respectively. If the value has an undefined type (in other words. has not been set), and the *if_undefined* parameter was provided, returns *if_undefined*.
Nwa_DateFormat(*value***,** *format***)**	Converts a numerical *value* (UNIX time) to a string using the date and time format string *format*. The format string uses similar syntax to the NwaDateFormat() function. See “Date/Time Format String Reference” for a full list of the supported format strings.
Nwa_FloatFormat(*value***,** *decimals***)**	Converts a numerical *value* to a string, with the number of decimal places specified in *decimals*.
Nwa_MinutesToNatural(*value***)**	Converts a numeric *value* measuring a time in minutes to a natural time representation (such as “2 minutes”, “3 hours”, “11 days”).
Nwa_NumberFormat(*value***[,** *if_undefined***])** **Nwa_NumberFormat(***value***,** *decimals***)** **Nwa_NumberFormat(***value***,** *decimals***,** *dec_point***,** *thousands_sep***[,** *if_undefined***])**	Converts a numerical value to a string. If the value has an undefined type (in other words, has not been set), and the if_undefined parameter was provided, returns if_undefined. Otherwise, the number is converted to a string using the number of decimal places specified in decimals (default 0), the decimal point character in dec_point (default “.”), and the thousands separator character in thousands_sep (default “,”).
Nwa_TrimText(*value***,** *length***)**	Trims excessively long strings to a maximum of *length* characters, appending an ellipsis (“…”) if the string was trimmed.
Nwa_ValueText(*value***[,** *if_undefined***])**	If the *value* has an undefined type (in other words, has not been set), and the *if_undefined* parameter was provided, returns *if_undefined*, or a HTML non-breaking space (“ ”) otherwise. Otherwise, the *value* is converted to a string for display.

LDAP Standard Attributes for User Class

The following list provides some of the attributes for the LDAP User class. For a complete list you should consult http://msdn2.microsoft.com/en-us/library/ms683980(VS.85).aspx#windows_2000_server_attributes.

- **userPrincipalName**: The userPrincipalName is a single-valued and indexed property that is a string that specifies the user principal name (UPN) of the user. The UPN is an Internet-style login name for the user based on the Internet standard RFC 822. The sAMAccountName property is a single-valued property that is the logon name The objectSid property is a single-valued property that specifies the security identifier (SID) of the user.
- **accountExpires**: The accountExpires property specifies when the account will expire.
- **badPasswordTime**: The badPasswordTime property specifies when the last time the user tried to log onto the account using an incorrect password.
- **badPwdCount**: The badPwdCount property specifies the number of times the user tried to log on to the account using an incorrect password.
- **codePage**: The codePage property specifies the code page for the user's language of choice. This value is not used by Windows 2000.
- **countryCode**: The countryCode property specifies the country code for the user's language of choice. This value is not used by Windows 2000.
- **lastLogoff**: The lastLogoff property specifies when the last logoff occurred.
- **lastLogon**: The lastLogon property specifies when the last logon occurred.
- logonCount: The logonCount property counts the number of successful times the user tried to log on to this account.
- **mail**: The mail property is a single-valued property that contains the SMTP address for the user (such as demo@example.com).
- **memberOf**: The memberOf property is a multi-valued property that contains groups of which the user is a direct member.
- **primaryGroupID**: The primaryGroupID property is a single-valued property containing the relative identifier (RID) for the primary group of the user.
- **sAMAccountType**: The sAMAccountType property specifies an integer that represents the account type.
- **unicodePwd**: The unicodePwd property is the password for the user.

Regular Expressions

The characters shown in Table 41 can be used to perform pattern matching tasks using regular expressions.

Table 41 *Regular Expressions for Pattern Matching*

Regex	Matches
a	Any string containing the letter “a”
^a	Any string starting with “a”
^a$	Only the string “a”
a$	Any string ending with “a”
.	Any single character

Table 41 *Regular Expressions for Pattern Matching*

\.	A literal "."
[abc]	Any of the characters a, b, or c
[a-z0-9A-Z]	Any alphanumeric character
[^a-z]	Any character not in the set a through z
a?	Matches zero or one "a"
a+	Matches one or more: a, aa, aaa, …
a*	Matches zero or more: empty string, a, aa, aaa…
a|b	Alternate matches: Matches an "a" or "b"
(a.*z)	Grouping: matches sequentially within parentheses
a*?	"Non-greedy" zero or more matches
\ooo	The character with octal code ooo
\040	A space
\d	Any decimal digit
\D	Any character that is not a decimal digit

The regular expression syntax used is Perl-compatible. For further details on writing regular expressions, consult a tutorial or programming manual.

Chapter 10

Glossary

802.1X	IEEE standard for port-based network access control.
Access-Accept	Response from RADIUS server indicating successful authentication, and containing authorization information.
Access-Reject	Response from RADIUS server indicating a user is not authorized.
Access-Request	RADIUS packet sent to a RADIUS server requesting authorization.
Accounting-Request	RADIUS packet type sent to a RADIUS server containing accounting summary information.
Accounting-Response	RADIUS packet sent by the RADIUS server to acknowledge receipt of an Accounting-Request.
accounting session time	Length of time the guest has been using the network.
accounting	Process of recording summary information about network access by users and devices.
authentication	Verification of a user's credentials, typically a username and password.
authorization	Authorization controls the type of access that an authenticated user is permitted to have.
BYOD	Bring your own device. Refers to the trend of personal mobile devices being used with enterprise network infrastructure.
CA	See *Certificate Authority*.
captive portal	Implemented by NAS. Provides access to network only to authorized users.
certificate authority	Entity in a public key infrastructure system that issues certificates to clients. A certificate signing request received by the CA is converted into a certificate when the CA adds a signature that is generated with the CA's private key. See *digital certificate*, *private key*, and *public key infrastructure*.
common name (CN)	See *distinguished name*.
$criteria	Array that consists of one or more criteria on which to perform a data based search. This array is used for advanced cases where pre-defined helper functions do not provide required flexibility.
CRL	Certificate revocation list. List of revoked certificates maintained by a certificate authority and regularly updated.
CSV	Comma-separated values.
device provisioning	Process of preparing a device for use on an enterprise network, by creating the appropriate access credentials and setting up the network connection parameters.
digital certificate	Contains identification data (see *distinguished name*) and the public key portion of a public/private key pair, and a signature that is generated by a certificate authority. The signature ensures the integrity of the data

	in the certificate (only the certificate authority can create valid certificates).
Disconnect-Ack	NAS response packet to a Disconnect-Request, indicating that the session was disconnected.
Disconnect-Nak	NAS response packet to a Disconnect-Request, indicating that the session could not be disconnected.
Disconnect-Request	RADIUS packet type sent to a NAS requesting that a user or session be disconnected.
distinguished name	Series of fields in a digital certificate that, taken together, constitute the unique identity of the person or device that owns the digital certificate. Common fields in a distinguished name include country, state, locality, organization, organizational unit, and the "common name", which is the primary name used to identify the certificate.
DN	See *distinguished name*.
EAP	Extensible Authentication Protocol (RFC 3748). An authentication framework that supports multiple authentication methods.
EAP-PEAP	Protected EAP. A widely-used protocol for securely transporting authentication data across a network.
EAP-TLS	Extensible Authentication Protocol – Transport LayerSecurity (RFC 5216). A certificate-based authentication method supporting mutual authentication, integrity-protected ciphersuite negotiation and key exchange between two endpoints.
form	Screen that collects data using fields.
field	Single item of information about a visitor account.
guest	See *Visitor*.
intermediate CA	Certificate authority with a certificate that was issued by another certificate authority. See *trust chain*.
iOS	Operating system from Apple, Inc. for mobile devices, including the iPhone, iPad, and iPod Touch.
landing page	See *Web login*.
LDAP	Lightweight Directory Access Protocol; communications protocol used to store and retrieve information about users and other objects in a directory.
Network Access Server (NAS)	Device that provides network access to users, such as a wireless access point, network switch, or dial-in terminal server. When a user connects to the NAS device, a RADIUS user authentication request (Access-Request) is generated by the NAS.
OCSP	Online certificate status protocol (RFC 2560). Protocol used to determine the current status of a digital certificate without requiring CRLs.
onboarding	See *device provisioning*.
onboard-capable device	Device supported by the QuickConnect application.
onboard provisioning	Process used to securely provision a device and configure it with network settings.

operator profile	Characteristics assigned to a class of operators, such as the permissions granted to those operators.
operator/operator login	Person who uses ClearPass Guest to create guest accounts or perform system administration.
OS X	Operating system from Apple, Inc. for desktop and laptop computers.
over-the-air provisioning	Process used to securely provision a device and configure it with network settings; applies to iOS and OS X 10.7+ only.
PEAP	Protected EAP. See *EAP-PEAP*.
ping	Test network connectivity using an ICMP echo request ("ping").
PKCS#*n*	Public-key cryptography standard N. Refers to a numbered standard related to topics in cryptography, including private keys (PKCS#1), digital certificates (PKCS#7), certificate signing requests (PKCS#10), and secure storage of keys and certificates (PKCS#12).
PKI	Public-key infrastructure. Security technology based on digital certificates and the assurances provided by strong cryptography. See also *certificate authority*, *digital certificate*, *public key*, *private key*.
print template	Formatted template used to generate guest account receipts.
private key	The part of a public/private key pair that is always kept private. The private key is used to encrypt a message's signature to authenticate the sender (only the sender knows the private key). The private key is also used to decrypt a message that was encrypted with the sender's public key (only the sender can decrypt it).
public key	The part of a public/private key pair that is made public. The public key is used to encrypt a message; the recipient's private key is required to decrypt the message. A large part of a digital certificate is the certificate owner's public key.
QuickConnect App	Application used to securely provision an Android, Windows, or OS X device and configure it with network settings.
RFC	Request For Comments; a commonly-used format for Internet standards documents.
role	Type of access being granted. You can define multiple roles. Such roles could include employee, guest, team member, or press. Roles are used for both guest access (user role) and operator access to ClearPass Guest. See *operator profile*.
root CA	Certificate authority that signs its own certificate (a self-signed certificate), and must be explicitly trusted by users of the CA.
SCEP	Simple certificate enrollment protocol. Protocol for requesting and managing digital certificates.
self-signed certificate	See *root CA*.
session	Service provided by a NAS to an authorized user.
skin	Web site's external appearance, or "look and feel." It can be thought of as a container that holds the application, its style sheet (font size and color for example), its header and footer, and so forth.
SMS	Short Message System; a method for delivering short messages (up to 140 characters) to mobile phones.

sponsor	See *operator*.
TLS	See *EAP-TLS*.
trust chain	Sequence of certificates, starting at a trusted root certificate, that establishes the identity of each certificate in the chain.
trusted root	See *root CA*.
unique device credentials	Network authentication credentials that uniquely identify the device and user and enable management of provisioned devices. May be a username and password or a TLS client certificate, depending on the type of device.
user database	Database of the guests on the system.
view	Table containing data. Used to interactively display data such as visitor accounts to operators.
visitor/guest	Someone who is permitted to access the Internet through your Network Access Server.
VPN	Virtual private network. Enables secure access to a corporate network when located remotely.
VSA	Vendor-specific attribute.
walled garden	Network resources that can be accessed by unauthorized users through the captive portal.
Web login	Login page displayed to a visitor.
X.509	Standard defining the format and contents of digital certificates.

Index

Numerics

1024-bit RSA 119
2048-bit RSA 119
4096-bit RSA 119

A

AAA 21
access control, print templates 218
account filters, creating 268
accounting 20, 21, 23
accounts
  passwords, multiple 197
  visitor account 24
Active Directory
  LDAP authentication 273
active sessions 67, 69
administration 241, 260
  plugin management 245
Administration module 241
AirGroup
  authenticating users via LDAP 243
  configuration summary 26
  configuring fields 162
  configuring operator device limit 272
  creating groups 60
  creating users 272
  defining controller 242
  enabling dynamic notifications 242
  personal devices 62
  registering devices 60
  shared locations 60
  shared roles 61
  tag=value pair 60
alerts, SMS 71
application log 260
  filtering 261
  searching 261
  viewing 260
applications, installing 89
authentication 20, 21, 23, 33, 50
authorization 21, 23, 33
  access, role-based 20
  dynamic 69

B

Base-64 encoded 108
binary certificate 108

C

caching, CSV 307
CAPTCHA security code 169
captive portal 23, 191
  hotspot 226
certificate
  formats 108
  signing requests 110
certificates
  code-signing 112
  deleting 109
  exporting 107
  importing 114
  requesting 116
  revoking 109
character set encoding 46
closed session 69
closing session 71
code-signing certificate 112
Configuration module 147
configuring
  Android provisioning 127
  device limit in AirGroup 272
  device provisioning 90
  iOS and OS X provisioning 122
  Kernel plugin 247
  legacy OS X provisioning 125
  network settings 131
  operator logins 283
  plugins 246
  provisioning settings 118
  receipts 255
  revocation checks 121
  self-service portal, display functions 326
  shared_location field 162
  shared_role field 162
  skin 248
  skin plugin 248
  SMS services 250
  Windows provisioning 126
contacting support 262

content
deleting 150
downloading 149, 150
management 148
quick view 150
renaming 150
uploading 149
viewing 150
creating
account filter 268
AirGroup administrator 272
AirGroup groups 60
AirGroup operator 272
device accounts 56
field 160
guest account 33
hotspot plan 229
LDAP server 274
LDAP translation rule 279
multiple guest accounts 35, 48
operator 272
operator profile 266
operator profiles 266
print template 216
self registration 192
session filter 268
CSV
caching 307
parsing 308
customer support 262
customizing
content 148
email receipt 211, 212
fields 159
Guest Manager 151
hotspot invoice 233
hotspot receipt 239
hotspot selection interface 234, 237, 239
login message 206
login page 204
receipt actions 198
receipt page 197
Register Shared Device 162
registration form 196
registration page 195
self-service portal 207
view fields 188

D

data retention 74, 244
databases, user 24
default skin 249
deleting
certificate 109
content 150
field 161
deployment
network provisioning 25
operational issues 25
overview 24
security policy 24
site checklist 25
device type 106
devices 50
creating accounts 56
editing 63
filtering 51
importing 65
personal, AirGroup 62
provisioning configuration 118
shared 60
viewing 63
disconnecting session 68, 69
documentation, viewing 262
downloading content 149, 150
duplicating
fields 161
forms and views 167
dynamic authorization 67, 69

E

editing
base field 167, 189
devices 63
expiration time, guest account 41
fields 161
form fields 168
forms 166, 167
guest accounts 42, 277
guest self-registration 192
hotspot plans 229
multiple guest accounts 43
print templates 218
receipt pages 197
self-registration 196
views 166, 188
email
guest self-registration receipts 202
receipts 35, 210
receipts, customizing 211
SMTP services 210
encoding 46
encryption key, in guest receipt 152
expiration
guest accounts, editing 41
exporting
certificates 107
guest accounts 48

F

Fields
- card_expiry ... 319
- card_name ... 319
- card_number ... 319
- city ... 319
- country ... 319
- Delete ... 161
- do_expire ... 156
- do_schedule ... 156
- dynamic_expire_time ... 312
- dynamic_is_expired ... 312
- Edit ... 161
- email ... 155, 313
- enabled ... 156, 313
- expiration_time ... 313
- expire_after ... 156
- expire_postlogin ... 157
- expire_time ... 156, 313
- expire_usage ... 157, 313
- first_name ... 319
- hotspot_plan_id ... 319
- hotspot_plan_name ... 319
- id ... 313
- ip_address ... 313
- last_name ... 319
- modify_expire_postlogin ... 314
- modify_password ... 155, 314
- modify_schedule_time ... 315
- multi_initial_sequence ... 315
- multi_prefix ... 154, 315
- netmask ... 315
- no_password ... 315
- no_portal ... 315
- no_warn_before ... 315
- notes ... 315
- num_accounts ... 315
- password ... 155, 217, 315
- password_action ... 316
- password_action_recur ... 316
- password_last_change ... 316
- password2 ... 155, 316, 320
- personal_details ... 320
- purchase_amount ... 320
- purchase_details ... 320
- random_password ... 155, 316
- random_password_length ... 155, 156, 316
- random_password_method ... 155, 156, 317
- random_password_picture ... 322
- random_username_length ... 155, 156, 317
- random_username_method ... 152, 155, 156
- random_username_picture ... 322
- role_id ... 156
- role_name ... 156, 217
- schedule_after ... 156
- schedule_time ... 156
- secret_answer ... 209
- secret_question ... 209
- Show forms ... 161
- simultaneous_use ... 156
- sms_auto_send_field ... 220, 320
- sms_enabled ... 220, 320
- sms_handler_id ... 220, 320
- sms_phone_field ... 220, 320
- sms_template_id ... 220, 320
- sms_warn_before_message ... 320
- smtp_auto_send_field ... 214
- smtp_cc_action ... 214
- smtp_email_field ... 214
- smtp_enabled ... 214
- smtp_receipt_format ... 214
- smtp_subject ... 214, 321
- smtp_template_id ... 214, 321
- smtp_warn_before_cc_action ... 215, 322
- smtp_warn_before_cc_list ... 215, 322
- smtp_warn_before_receipt_format ... 215
- smtp_warn_before_subject ... 214, 321
- smtp_warn_before_template_id ... 215, 322
- state ... 320
- submit_free ... 320
- username ... 155, 217
- visitor_accept_terms ... 320
- visitor_carrier ... 320
- visitor_fax ... 320
- visitor_name ... 209
- warn_before_from ... 215, 322
- warn_before_from_sponsor ... 215, 322
- zip ... 320

fields ... 23, 155
- account_activation ... 311
- address ... 319
- auto_send_sms ... 320
- auto_update_account ... 155
- card_code ... 319
- creating ... 160
- creator_accept_terms ... 155
- customizing ... 159
- deleting ... 161
- duplicating ... 161
- importing matching ... 47
- modify_expire_time ... 156
- modify_schedule_time ... 156
- multi_initial_sequence ... 154
- password ... 154
- random_username_length ... 152
- random_username_picture ... 152
- rank ordering ... 167
- show views ... 162
- simultaneous_use ... 154
- smtp_cc_list ... 214

filtering
- application log ... 261
- devices ... 51
- guest accounts ... 40, 44
- sessions ... 70

Form field
Display properties ... 168
Drop-down list ... 172
Enable If ... 187
Hidden ... 173
Password ... 174
Radio Buttons ... 174
Static text ... 175
Static text (Options lookup) ... 177
Static text (Raw value) ... 176
Submit button ... 179
Text area ... 179
Text field ... 180
Validation errors ... 181
Validation properties ... 180
Value conversion ... 185
Value formatter ... 186
Visible If ... 187

form fields
advanced properties ... 184
CAPTCHA ... 169
check box ... 169
checklist ... 170
conversion functions ... 326
Date/time picker ... 172
display functions ... 167, 326
group heading ... 178
initial value ... 180
validator functions ... 323
value format functions ... 326

formats, certificate ... 108

forms ... 23, 155, 159
change_expiration ... 159
create_multi ... 159
create_user ... 159
customizing ... 165
duplicating ... 167
editing ... 166, 167
form field editor ... 168
guest_edit ... 159
guest_multi_form ... 45, 159
guest_register ... 159
guest_register_receipt ... 159
previewing ... 168
reset_password ... 159

G

guest ... 24

guest access
business rules ... 155
click to print ... 155
email receipt ... 210
NAS login ... 190
receipt page ... 190
registration page ... 190
roles ... 20

guest access, self-provisioned ... 32

guest accounts
activate ... 42
change expiration ... 41
creating ... 33
creating multiple ... 35, 48
delete ... 41
disable ... 41
edit ... 42
editing expiration ... 41
email receipt ... 35
export ... 48
exporting ... 48
filtering ... 40, 44
importing ... 45
list ... 39
manage multiple ... 43
paging ... 40
print ... 42
receipts ... 34
reset password ... 40
selection row ... 44
SMS receipt ... 35
view passwords ... 154
XML export ... 49

guest management ... 31
custom fields ... 159
customizing ... 151
email receipts ... 210
print template wizard ... 217
print templates ... 215
self provisioned ... 190
sessions ... 67
SMS receipts ... 72, 255

Guest module ... 31

guest self-registration
download receipt ... 201
email receipts ... 202
login page ... 204
print receipt ... 201
self-service portal ... 207
SMS receipt ... 203

H

help
context-sensitive ... 27
field help ... 28
quick help ... 28
searching ... 27

hotspot management ... 225
captive portal ... 228
creating plan ... 229
customer information ... 232
customizing invoice ... 233
customizing receipt ... 239
customizing selection interface ... 234, 237, 239

editing plan 229
invoice 233
plans 229
Hotspot Manager 225
HTML
Smarty templates 287
standard styles 286
syntax 285

I

importing
certificate, code-signing 112
devices 65
guest accounts 45
matching fields 47
trusted certificate 114
installing applications 89

K

key 152
key type 119

L

LDAP
authenticating AirGroup users 243
creating translation rule 279
custom rules 281
matching actions 280
matching rules 280
operator logins 273
POSIX-compliant servers 273
server, creating 274
standard attributes 330
translation rules 273
translation rules, creating 279
URL syntax 276
local operators 272
locations, AirGroup 60
log files 260
logging
passwords 154

M

MAC
address formats 50
advanced features 65
authentication 50
registering devices 64
multiple guest accounts, creating 35

N

NAS 33
login 24
login, guest self-registration 203
Network Access Server 23
network settings
configuring 131
ntication 50
nwa_radius_query 293

O

Onboard
date retention 74
Smarty template functions 91
Onboard module 73
Open SSL text format 108
operator
creating 272
Operator logins
LDAP 273
operator logins 265
advanced options 284
configuration 283
LDAP server, creating 274
password options 267
user roles 268
operator profiles 23, 265, 266
automatic logout 284
creating 266
privileges 271
operators 24
creating 272
local 272
login message 283

P

passcode policy 143
passwords
generating 152
logging 154
multiple accounts 197
recovery 130
resetting 40
picture string 322
PKCS#12 108
PKCS#7 108
plugin management 245
plugins
configuring 246, 248
configuring, Kernel 247
configuring, skin 248
restoring default configuration 247
viewing 245
POSIX, LDAP 273
previewing
forms 168

print templates 24, 215
creating 216
creating using wizard 217
custom fields 217
editing 218
permissions 218
SMS receipts 216
programmer's reference 285
provisioning settings
configuring 118

Q

quick start, Smarty template syntax 287
quick view, content 150

R

RADIUS server 20
accounting query 293
active sessions 67
disconnecting session 68, 69
reauthorizing session 68, 69
rauthorizing
session 69
reauthorizing
session 68
receipt page 190
editing 197
receipts 255
configuring 255
email 210
SMS 72
reference 285
Register page 190
registering MAC devices 64
regular expressions 330
renaming
content 150
resetting
certificates 145
passwords 40, 208
revocation checks 121
revoking certificate 109
RFC 2255 278
RFC 3576 69
role-based access 20
Role-based access control 265
roles 24
shared 61
RSA 119

S

searching
application log 261
documentation 262
security policy checklist 25
self registration
creating page 192
self-registration
creating device 58
editing 196
self-service portal 207
auto login 208
password generation 208
resetting passwords 208
secret question 209
sending
SMS alert 71
SMS message 254
sequence diagram
AAA 21
guest self-registration 191
servers
LDAP, creating 274
session filters, creating 268
sessions
active 67, 69
closed 69
closing 71
device 55
disconnecting 68, 69
filtering 70
reauthorizing 68, 69
SMS alert 71
stale 69
shared locations 60
shared roles 61
site SSID 152
skin
configuring 248
email receipt 212
Smarty syntax
subject line 212
Smarty template functions 287
assign function 288
comments 288
foreach block 289
if block 288
include 288
literal block 288
modifiers 289
Onboard 91

section block 289
variables 287
SMS
alert for session 71
alerts 71
character limit 216
credits 254
guest account receipts 35
guest self-registration receipts 203
receipts 72
subject line 212
SMS services 250
configuring 250
credits available 254
guest receipts 72, 255
low credit warning 255
receipt options 255
send 254
sending message 254
SMS gateways 250
SMTP services 210
customizing receipt 214
sponsors 24
SSID 152
stale session 69
subject line
email receipt 210
support 262
support services 260
T
tab-separated values 48
tag=value pair 60
template
predefined template functions 290
translation rules 279
troubleshooting
application integrity check 246
Onboard 146
TSV 48
U
uploading
code-signing certificate 112
content 149
user database 24
V
viewing
application log 260
content 150
devices 63
documentation 262
plugins 245
sessions, device 55
views 24, 155, 159
column format 190
customization 165
duplicating 167
editing 166, 188
field editor 189
guest_export 48, 159
guest_multi 43, 159
guest_sessions 68, 159
guest_users 39, 159
visitors 24
account 24
VPN settings 140
W
Web logins 24
WiFi network 152
wizards
print template 217
WPA key 152
X
XML
guest account list 49
parsing 309