Aruba Networks

AirWave MOP v2.2

Adding new physical hard disks to an ESXi server and leveraging Multi-Disks for AirWave deployed in VM.



Aruba Networks 10/7/2013



Revision Number	Date	Author(s)	Change Description
R1.0	October 1, 2013	Adam Kennedy	Document Creation
R1.1	October 4, 2013	Adam Kennedy	Updates from TAC
R1.2	October 6, 2013	Adam Kennedy	Misc edits, added content
R2.0	October 7, 2013	Adam Kennedy	Edits, TAC/PLM Signoff.
R2.1	October 8, 2013	Adam Kennedy	Typos etc
R2.2	October 9, 2013	Adam Kennedy	AirWave Backup, restoral



Contents

1.	Introduction	3
N	lotes on IAP operation and AirWave	3
2.	Airwave backup and restoral	3
3.	VMWare Hardware Platform	4
4.	Adding the physical disk drive to the ESXi Server	8
	Adding a new Hard Disk and data store in the AirWave Virtual Machine	. 12
5.	Confirming the additional drive in AirWave	. 15
6.	Creating the new disk partition from /dev/sdb	.16
	Command Summary	.16
	Output of partition Creation	. 17
7.	Mapping new partition into a Single Logical Volume for AirWave	. 19
	Command Summary	. 19
	Command output detail	. 19
	Logical Volume prior to changes	. 20
	Logical Volume post changes	. 22
С	Confirmation in the AirWave Management UI	.23
7.	Resizing an AirWave data store on a single hard disk	.24
	Command summary for creating new partition	.24



1. Introduction

The purpose of this MOP is to document the process for adding more physical hard drives to an ESXi Server which will be added to an existing AirWave VMWare Virtual Machine. The expected result of this MOP is that AirWave will be able to utilize additional drive space on physically separate hard drive volumes. Total operational time required to perform these steps including installation of the physical drive is under 20minutes.

Notes on IAP operation and AirWave

Aruba Instant Access Points (IAP's) post information via HTTPS on state every minute to AirWave and update statistics every 5 minutes. In that, AirWave does not poll IAP devices directly. To this end, during the course of the execution of this MOP – the suggested reboots of the AirWave VM will have no impact to clients connected to the IAP or the IAP themselves. In that, it is not necessary to port IAP's to another instance of AirWave to perform these tasks.

Should an IAP not be able to post status to the production AirWave server at the specified interval, it stores the information in memory until connection to the AirWave server (post reboot) is reestablished.

2. Airwave backup and restoral

Prior to any changes being made to the production AirWave platform, it's important to confirm the backup of the AirWave client data. By default, AirWave takes a backup nightly at 04:15 and stores it locally on the AirWave Appliance in the following location:

@ Airway	/e		+		1.	00
← ▲ https://192.168.175.7/amp_backups						
🎦 Junipe	r 🙏 Aruba	3200 🙏 Aruba 3	500 🗍 VIV	lware 不 Aiı	Wave 🙏 (CPPM [
ARUE	k s					
Home	Groups	APs/Devices	Clients	Reports	System	Devi
Status	Syslog 8	Traps Event	t Log Tr	iggers A	lerts Ba	ckups

Backups are run nightly.

nightly_data001.tar.gz Backup of 6491767 bytes made 7 hrs 41 mins ago. nightly_data002.tar.gz Backup of 6159893 bytes made 1 day 7 hrs 41 mins ago. nightly_data003.tar.gz Backup of 5833564 bytes made 2 days 7 hrs 41 mins ago. nightly_data004.tar.gz Backup of 5504387 bytes made 3 days 7 hrs 41 mins ago.

Figure 1 - AirWave Backup files



The Airwave backup and restoral procedure is fully documented in the Airwave 7.7 User Guide on page 223 or click the 'Help' icon in the top right hand corner of the above Airwave page.

It should also be noted that the steps denoted in this MOP are not committed to the CentOS platform until the partition command is issues to resize the partitions. Up until this point, no hard changes have been made to the system. The command specifically is 'resize2fs /dev/VolGroup00/LogVol00' which is contained in section 6 of this document.

In addition to this, it's possible to also take a 'snapshot' of the Airwave Virtual Machine installation form the vShpere console and later use this to restore Airwave to this known point in time. This procedure would be covered in your VMware operational procedures and falls outside of scope of this MOP, but we have cited it for reference.



Figure 2 - Taking a VMWare Snapshot

3. VMWare Hardware Platform

For the purposes of this MOP, an ESXi server has been installed running version 5.1.0 as per the below screen capture. In addition, the VMWare server is running version 7.7.3 (64bit) of the AirWave Management Platform. The present installation of AirWave 7.7.3 has been allocated an 80GB partition of the 2TB HDD.



Resources			
Consumed Host CPU:472 MHzConsumed Host Memory:3894.00 MEActive Guest Memory:655.00 ME			MHz MB MB
Provisioned Storage: 88.09 Not-shared Storage: 88.09 Used Storage: 88.09) GB) GB) GB	
Storage 🗠	Drive Type	Capacity	
datastore1	Non-SSD	1.81 TB	
۰ III			
	-		
Network	Type		

Figure 3 - Existing AirWave HDD allocation

login as: root Access denied root@192.168.175.7's password: Last login: Fri Oct 4 16:47:29 2013 from 192.168.175.62 SVN Root: http://svn.corp.AirWave.com/usr/local/svnroot/mercury/tags/RELEASE_7_7_3 Current Version: RELEASE_7_7_3 [root@localhost mercury]#

Installed in the VMWare server is 24GB of DDR3 RAM and a single 2TB SATA hard drive. As part of this MOP, we'll be adding a secondary physical hard drive into the ESXi. The required vSphere client is installed granting remote management from the operational workstation.





Figure 4 - vSphere Client Information



By logging into the vSphere console, we can validate the existing installation as well as the available resources to the VM Platform:



Figure 5 - Existing VMWare Summary

If we login into the AirWave console via SSH credentials, we can validate the existing drive volumes that are allocated to AirWave.

[root@localhost mercury]# fdisk -l Disk /dev/sda: 85.9 GB, 85899345920 bytes 255 heads, 63 sectors/track, 10443 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes



Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00068cec

Device Boot Start End Blocks Id System /dev/sda1 * 1 13 102400 83 Linux Partition 1 does not end on cylinder boundary. /dev/sda2 13 10444 83782656 8e Linux LVM

Disk /dev/mapper/VolGroup00-LogVol01: 4294 MB, 4294967296 bytes 255 heads, 63 sectors/track, 522 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x0000000

Disk /dev/mapper/VolGroup00-LogVol00: 81.5 GB, 81470160896 bytes

255 heads, 63 sectors/track, 9904 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000 [root@localhost mercury]#

The **BOLDED** sections validate the existing installation of AirWave has been allocated 80GB of storage on the 2TB HDD.

4. Adding the physical disk drive to the ESXi Server

- 1. Power down the ESXi Server.
- 2. Open the case and validate the spare slot capacity of the server to accommodate the new disk drive.
- 3. Insert the new drive into the spare slot ensuring an anti-static strap is used to ensure proper protection for the drive. Properly secure the drive using the provided screws while ensuring that no other cables have been knocked loose during the drive installation.





Figure 6 - New HDD Install

4. Connect the communications cable to the proper port on the server board, in this case a SATA communications port.





Figure 7 - Cabling new HDD in ESXi Server

- 5. Upon confirmation of all cables and connectors being properly attached, boot the ESXi Server.
- 6. Login to the vSphere client application



🕜 VMware vSphere Client
Vmware VMware vSphere" Client
In vSphere 5.5, all new vSphere features are available only through the vSphere Web Client. The traditional vSphere Clent will continue to operate, supporting the same feature set as vSphere 5.0, but not exposing any of the new features in vSphere 5.5. The vSphere Client is still used for the vSphere Update Manager (VUM) and Host Client, along with a few solutions (e.g. Site Recovery Manager).
To directly manage a single host, enter the IP address or host name. To manage multiple hosts, enter the IP address or name of a vCenter Server.
IP address / Name: 192.168.175.6
User name: root
Password:
Use Windows session credentials

Figure 8 - vSphere login post HDD Install

7. Click on the top level IP for the VM, select the 'Summary' tab and we now see that there are two physical disks installed in the ESXi Server as per below

Resources				
CPU usage: 23 MHz Capacity 4 x 3.092 GHz				
Memory usage: 1159.00 M	18	Capacity 24424.55 M	В	
Storage 🛆	Drive Type	Ca	apacity	
datastore1	Non-SSD	1	1.81 TB	
datastore2	Non-SSD	460	.75 GB	208
•				- F.
Network	Туре			
🧕 VM Network	Standard po	rt group		
•	1			. •

Figure 9 - New HDD Visible in vSphere



Adding a new Hard Disk and data store in the AirWave Virtual Machine

Next we need to create a new virtual hard disk for the AirWave 7.7.3 Virtual Machine to leverage.

1. Navigate to the AirWave 7.7.3 VM, click on Summary and then Edit Settings.

AirWave 7.7.3	
Getting Started Summary Resource Allocation Performan	nce Events Console Permissions
General	Resources
Guest OS: CentOS 4/5/6 (64-bit) VM Version: 7 CPU: 4 vCPU Memory: 8192 MB Memory Overhead: 80.01 MB VMware Tools: \blacklozenge Not running (Not installed)	Consumed Host CPU: 223 MHz Consumed Host Memory: 3919.00 MB Active Guest Memory: 491.00 MB Refresh Storage Usage Provisioned Storage: 88.09 GB Not-shared Storage: 88.09 GB
IP Addresses:	Used Storage: 88.09 GB
DNS Name:	datastore1 Non-SSD 1.81 TB
State: Powered On Host: localhost.xplornet.com	III Network Type
vSphere HA Protection: ② N/A 모	VM Network Standard port group
Commands Power Off Suspend Reset Edit Settings Open Console	

Figure 10 - Adding to the AirWave 7.7.3 Datastore

- 2. Click on 'ADD', then select 'Hard Disk' and click NEXT
- 3. Click 'Create a new Virtual Disk' and clock NEXT



AirWave 7.7.3 - Virtual Machine Properties	
Hardware Options Resources	Virtual Machine Version: 7
Show All Devices Add	Remove 255 GB Memory Size: 8 GB GB
Hardware Summar	
CPUS Video c VMCI d CPUS	
SLS1 co Hard di CD/DVI Select a Disk Floppy Advanced Options Ready to Complete	A virtual disk is composed of one or more files on the host file system. Together these files appear as a single hard disk to the guest operating system.
ready to complete	Create a new virtual disk Use an existing virtual disk Reuse a previously configured virtual disk.
	Raw Device Mappings Give your virtual machine direct access to SAN. This option allows you to use existing SAN commands to manage the storage and continue to access it using a datastore.
Help Help	Sack Next ≥ Cancel

Figure 11 - Creating a new VM Virtual Disk for AirWave 7.7.3

4. Specify the size of the desired disk capacity (capacity must exist) and then specify the LOCATION of the data store for this new disk. For our example, this is the newly added datastore2 which is the 500GB SATA drive and we have specified an additional 80GB allocation.

NOTE: By default, installation of AirWave in VM suggests and 80GB allocation for thick provisioning in the VM environment. That is not to say that more hard drive space cannot be allocated at the time of install, this is merely the default suggestion. An AirWave installation on a single Hard disk can have the provisioned data store reallocated at any time assuming that capacity exists (read: other VM's installed are not allocated drive space effectively making it unavailable to AirWave). To resize an existing data store allocation on a single hard disk with sufficient capacity, please refer to the section 7 entitled "Resizing an AirWave data store on a single Hard disk."



evice Typ	<u>e</u>	Capacity —					9
<u>elect a Dis</u> reate a	<u>sk</u> Disk	Disk Size:	80 🛨 GB	-			9
dvanced eady to (Options Complete	Disk Provisio	ning				
💋 Sele	ect a datastore o	or datastore clust	er				
Select	t a datastore or d	atastore cluster:					
TNOT	ine -	Diffe type	copucity 110	visioned rice	турс	mintrovisioning	
	datastam2	Non-SSD	460.75 GB 252	.34 GB 208.41 GB	VMF53	Supported	
8	UdidStorez			4 74 70	VMESS	European dead	
8	UdidStorez			A 74 70	VMECE	Cup p a shad	

5. Click NEXT on the 'Advance Options' screen and we are then presented with confirmation of the disk that is to be added to the AirWave Virtual Machine as per below. Click 'Finish' to complete.

Add Hardware Ready to Complete Review the selected options	s and click Finish to add	the hardware.	-	×
Device Type	Options:			
Create a Disk Advanced Options Ready to Complete	Hardware type: Create disk: Disk capacity: Disk provisioning: Datastore: Virtual Device Node: Disk mode:	Hard Disk New virtual disk 80 GB Thick Provision Lazy Zeroed datastore2 SCSI (0:1) Persistent		
Help		<1	Back Finish	Cancel

Figure 13 - Confirmation of Hard Disk to add



6. Our operation is confirmed by the new summary screen of AirWave 7.7.3 Virtual Machine. Capacity has now been doubled to 160GB from the previous 80GB.

AirWave 7.7.3						
Getting Started Summ	ary Resource Allocation Perform	ance Events Con	sole Permissior	15		
General		Resource	es			
Guest OS: VM Version: CPU: Memory:	CentOS 4/5/6 (64-bit) 7 4 vCPU 8192 MB	Consume Consume Active Gu	ed Host CPU: ed Host Memory: uest Memory:		225 3921.00 655.00	MHz MB MB
Memory Overhead: VMware Tools: IP Addresses:	 80.01 MB Not running (Not installed) 	Provision Not-shar Used Sto	ed Storage: ed Storage: rrage:	Rei	168.09 168.09 168.09 168.09	GB GB GB GB
DNS Name:		Storage	e 🛆	Drive Type Non-SSD	Capacity 1.81 TB	
State: Host:	Powered On localhost.xplornet.com	da	atastore2	Non-SSD	460.75 GB	12{ ▶
vSphere HA Protection:	② N/A ₽	Network	k M Network	Type Standard port group	1	
Commands						

Figure 14 - Validation of new Disk created on datastore2 (500GB SATA Drive)

5. Confirming the additional drive in AirWave

At this point we need to reboot the AirWave Server via the vSphere client dashboard. Rebooting the server takes a couple of minutes.

Upon re-logging into the AirWave root console, we can see that a second disk is now visible within the CentOS operating system as denoted by the presence of **/dev/sdb** below:

login as: root root@192.168.175.7's password: Last login: Fri Oct 4 18:01:06 2013 from 192.168.175.62 [root@localhost mercury]# **fdisk** -I

Disk /dev/sda: 85.9 GB, 85899345920 bytes

255 heads, 63 sectors/track, 10443 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00068cec



Device Boot Start End Blocks Id System /dev/sda1 * 1 13 102400 83 Linux Partition 1 does not end on cylinder boundary. /dev/sda2 13 10444 83782656 8e Linux LVM

Disk /dev/sdb: 85.9 GB, 85899345920 bytes

255 heads, 63 sectors/track, 10443 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000

Disk /dev/mapper/VolGroup00-LogVol01: 4294 MB, 4294967296 bytes 255 heads, 63 sectors/track, 522 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x0000000

Disk /dev/mapper/VolGroup00-LogVol00: 81.5 GB, 81470160896 bytes 255 heads, 63 sectors/track, 9904 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x0000000

[root@localhost mercury]#

6. Creating the new disk partition from /dev/sdb

Next we need to create a partition on the new hard disk. This is done at the CentOS/AirWave console prompt by entering a few simple commands.

Command Summary

fdisk /dev/sdb
Command (m for help): n <----- n to create a new Partition
p primary partition (1-4)
p <----- p to create a primary partition
Partition number (1-4): 1 <-----1 to identity partition 1
First cylinder (1-10443, default 1): 1 <-----1 to identify the first cylinder in the new disk</pre>



Last cylinder, +cylinders or +size{K,M,G} (1-10443, default 10443): <-----enter to use the default last cynlinder Hex code (type L to list codes): 8e <------selection 8e to create a Linux LVM partition Command (m for help): p <------ to print the environment of the new partition

Output of partition Creation

[root@localhost mercury]# fdisk /dev/sdb

Device contains neither a valid DOS partition table, nor Sun, SGI or OSF disklabel Building a new DOS disklabel with disk identifier 0xb5aacf15. Changes will remain in memory only, until you decide to write them. After that, of course, the previous content won't be recoverable. Warning: invalid flag 0x0000 of partition table 4 will be corrected by w(rite)

WARNING: DOS-compatible mode is deprecated. It's strongly recommended to switch off the mode (command 'c') and change display units to sectors (command 'u').

Command (m for help): n <----- n to create a new Partition Command action

- e extended
- p primary partition (1-4)

p <----- p to create a primary partition

Partition number (1-4): 1 <-----1 to identity partition 1

First cylinder (1-10443, default 1): **1** <------ **1 to identify the first cylinder in the new disk** Last cylinder, +cylinders or +size{K,M,G} (1-10443, default 10443): <-----enter to use the default last cylinder in the new disk

Using default value 10443

Command (m for help): t <------ to specify the type of partition to create Selected partition 1 Hex code (type L to list codes): 8e <-----selection 8e to create a Linux LVM partition Changed system type of partition 1 to 8e (Linux LVM)

Command (m for help): p <----- to print the environment of the new partition

Disk /dev/sdb: 85.9 GB, 85899345920 bytes 255 heads, 63 sectors/track, 10443 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes



Disk identifier: 0xb5aacf15

Device BootStartEndBlocksIdSystem/dev/sdb1110443838833668eLinux LVM <-----confirmation of the new LVM partition</td>

Command (m for help): **w** The partition table has been altered! Calling ioctl() to re-read partition table. Syncing disks.

[root@localhost mercury]# fdisk -l Disk /dev/sda: 85.9 GB, 85899345920 bytes 255 heads, 63 sectors/track, 10443 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00068cec

Device Boot Start End Blocks Id System /dev/sda1 * 1 13 102400 83 Linux Partition 1 does not end on cylinder boundary. /dev/sda2 13 10444 83782656 8e Linux LVM

Disk /dev/sdb: 85.9 GB, 85899345920 bytes 255 heads, 63 sectors/track, 10443 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0xb5aacf15

Device BootStartEndBlocksIdSystem/dev/sdb1110443838833668eLinux LVM <-----confirmation of the new Linux LVM</td>

Disk /dev/mapper/VolGroup00-LogVol01: 4294 MB, 4294967296 bytes 255 heads, 63 sectors/track, 522 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x0000000



Disk /dev/mapper/VolGroup00-LogVol00: 81.5 GB, 81470160896 bytes 255 heads, 63 sectors/track, 9904 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x0000000

At this point, we reboot AirWave again to ensure it is recognizing the newly created partition confirming the above output as before.

7. Mapping new partition into a Single Logical Volume for AirWave

We now need to map the new partition so that AirWave utilizes it as one logical volume. We can confirm the existing volume by using the **pvdisplay** command:

[root@localhost mercury]# pvdisplay

Physical vol	ume
PV Name	/dev/sda2
VG Name	VolGroup00
PV Size	79.90 GiB / not usable 27.00 MiB
Allocatable	yes (but full)
PE Size	32.00 MiB
Total PE	2556
Free PE	0
Allocated PE	2556
PV UUID	jc5vEC-k3Wt-TTOu-f9OT-CizN-qEtL-cMDLN

Command Summary

pvcreate /dev/sdb1 vgdisplay lvdisplay lvextend -L 155G /dev/VolGroup00/LogVol00 /dev/sdb1 resize2fs /dev/VolGroup00/LogVol00

Command output detail

1. First we need to use the **pvcreate** command to create a new partition:

[root@localhost mercury]# pvcreate /dev/sdb1



Writing physical volume data to disk "/dev/sdb1" Physical volume "/dev/sdb1" successfully created

2. We can confirm the existing volume information by using the vgdisplay command

[root@localhost mercury]# vgdisplay

Volume group				
VG Name	VolGroup00			
System ID				
Format	lvm2			
Metadata Are	as 1			
Metadata Sequence No 3				
VG Access	read/write			
VG Status	resizable			
MAX LV	0			
Cur LV	2			
Open LV	2			
Max PV	0			
Cur PV	1			
Act PV	1			
VG Size	79.88 GiB			
PE Size	32.00 MiB			
Total PE	2556			
Alloc PE / Size	2556 / 79.88 GiB			
Free PE / Size	0/0			
VG UUID	nwawpQ-07bH-QE04-lsqQ-bWao-ooxW-2sgRvp			

3. We are able to validate the logical volume detail prior to the change by using the **lvdisplay** command

Logical Volume prior to changes

[root@localhost mercury]# lvdisplay

Logical volume					
LV Name	/dev/VolGroup00/LogVol01				
VG Name	VolGroup00				
LV UUID	GLRzkk-1usA-JS17-QDa5-LWKE-taaB-fJp4Sl				
LV Write Access	s read/write				
LV Status	available				
# open	1				
LV Size	4.00 GiB				
Current LE	128				



Segments1AllocationinheritRead ahead sectorsauto- currently set to256Block device253:0

--- Logical volume ---

LV Name	/dev/VolGroup00/LogVol00				
VG Name	VolGroup00				
LV UUID	D6yzCX-6Hoi-dbml-Mmkd-PQ5J-rdPV-spDAtw				
LV Write Access	read/write				
LV Status	available				
# open	1				
LV Size	75.88 GiB				
Current LE	2428				
Segments	1				
Allocation	inherit				
Read ahead sectors auto					
- currently set to	256				
Block device	253:1				

4. We then use the **lvextend** command to extend the volume to the new desired size leveraging the new partition on **/dev/sdb1**. Note that the size of the extension will depend on the size of the hard disk you have installed as well as the size of the partition you have created on the disk.

[root@localhost mercury]# lvextend -L 155G /dev/VolGroup00/LogVol00 /dev/sdb1 Extending logical volume LogVol00 to 155.00 GiB Logical volume LogVol00 successfully resized [root@localhost mercury]#

5. We finally resize the logical volume group and volume to reflect our new partition by using the **resize2fs** command below:

[root@localhost mercury]# resize2fs /dev/VolGroup00/LogVol00
resize2fs 1.41.12 (17-May-2010)
Filesystem at /dev/VolGroup00/LogVol00 is mounted on /; on-line resizing required
old desc_blocks = 5, new_desc_blocks = 10
Performing an on-line resize of /dev/VolGroup00/LogVol00 to 40632320 (4k) blocks.
The filesystem on /dev/VolGroup00/LogVol00 is now 40632320 blocks long.



**Note this command will take some time to execute pending the size of the new disk as well as the size of the new partition.

6. We can then confirm our changes and success by issuing the **lvdisplay** command again post the previous commands.

Logical Volume post changes

[root@localhost	mercury]# lvdisplay			
Logical volur	ne			
LV Name	/dev/VolGroup00/LogVol01			
VG Name	VolGroup00			
LV UUID	GLRzkk-1usA-JS17-QDa5-LWKE-taaB-fJp4Sl			
LV Write Access read/write				
LV Status	available			
# open	1			
LV Size	4.00 GiB			
Current LE	128			
Segments	1			
Allocation	Allocation inherit			
Read ahead sectors auto				
- currently set to 256				
Block device 253:0				
Logical volume				
LV Name	/dev/VolGroup00/LogVol00			
VG Name	VolGroup00			
LV UUID	D6yzCX-6Hoi-dbml-Mmkd-PQ5J-rdPV-spDAtw			
LV Write Access read/write				
LV Status	available			
# open	1			
LV Size	155.00 GiB			
Current LE	4960			
Segments	2			
Allocation	inherit			
Read ahead sectors auto				
- currently set to 256				
Block device 253:1				
[root@localhost mercury]#				

With the above sequence of commands executed, we have successfully created a new 80GB partition on the /dev/sdb disk and logically mapped it to the volume creating a contagious 155GB Disk for AirWave to use for operation.



Confirmation in the AirWave Management UI

We can further validate this by logging into the AirWave UI interface:

Welcome to AirWave Management Platform				
APUPA AirWave				
Username admin				
Password •••••				
Log In				
Deutsch English Español Français Italiano 日本語 한국어 Türkçe 简体中文 祭體中文				
Figure 15 - AirWave Management console login				

Then navigate to 'System' and 'Performance' and scroll to the bottom where the disk space details are



Figure 16 - Hard Disk Performance Summary Post changes



7. Resizing an AirWave data store on a single hard disk

As was noted in section 3 of this document, the default installation size of AirWave in VM is 80GB for thick provisioning. If drive capacity exists on the hard drive containing the AirWave installation, the datastore on the drive can be easily resized to provide additional capacity for AirWave operation. As the drive already exists along with an active partition, we need only resize the installation in vSphere and then perform the same commands articulated previously for creating a new partition and resizing at the AirWave root console prompt.

i. First login to the vSphere console, navigate to the AirWave VM and click on 'Summary'. Next click on 'Edit Settings' and increase the 'Provisioned Size' of the installation as desired ensuring that the allocation falls within what is available on the disk, then click OK. For this example, we'll increase to 200GB.

AirWave 7.7.3 Getting Starter Summary esource Allocation Performance Events Console Permissions								
General		🕢 AirWave 7.7.3 - Virtual Machine	Properties					
Guest OS: VM Version: CPU: Memory:	CentOS 4/5/6 (64-bit) 7 4 vCPU 8100 MB	Hardware Options Resources	Add Remove	Virtual Machine Version: 7 Disk File [datastore 1] AirWave 7.7.3/AirWave 7.7.3.vmdk				
Memory Overhead: VMware Tools: IP Addresses:	80.01 MB • Not running (Not installed)	Hardware Memory CPUs	Summary 8192 MB 4	Disk Provisioning Type: Thick Provision Lazy Zeroed				
DNS Name: State: Host:	Powered On localhost.xplornet.com	Video card VMCI device Control device Hard disk 1	Video card Restricted Still gir Puellel Virtual Disk	Virtual Device Node				
Active Tasks: vSphere HA Protection: Commands	: ② N/A 🖓	CD/DVD drive 1 Network adapter 1 Floppy drive 1	CD-ROM 1 VM Network Floppy 1	SCSI (0:0) Hard disk 1				
Power Off Suspend Reset Edit Settings Open Console]			Independent Independent disks are not affected by snapshots. C Persistent Changes are immediately and permanently written to the disk. C Nonpersistent Changes to this disk are discarded when you power off or revert to the snapshot.				

Figure 17 - Resize Existing HDD data store in vSphere

Command summary for creating new partition

ii. Now create the new partition

[root@localhost mercury]# **fdisk** /dev/sda [root@localhost mercury]# **n** {new partition}



[root@localhost mercury]# p {primary partition} [root@localhost mercury]# 3 {select partition number, by default 3 is the next available} [root@localhost mercury]# t {select partition id we just made (3)} [root@localhost mercury]# 8e {Linux LVM partition} [root@localhost mercury]# p {print. the new device should be described as Linux LVM} [root@localhost mercury]# w {write to memory}

Note: You will need to reboot if fdisk updated kernel tables, just follow the recommendation message that will show up.

[root@localhost mercury]# reboot

iii. Now create a new volume name:[root@localhost mercury]# pvcreate /dev/sda3

iv. Then extend the existing volume group. **Vgdisplay** can be used to list and identify the volume groups you have.

[root@localhost mercury]# vgextend VolGroup00 /dev/sda3

Now, extend the logical volume, again, use **lvdisplay** to list and identify the logical volumes you have.

[root@localhost mercury]# lvextend -L 200G /dev/VolGroup00/LogVol00 /dev/sda3 Extending logical volume LogVol00 to 200.00 GiB Logical volume LogVol00 successfully resized [root@localhost mercury]#

v. Finally, we need to resize the logical volume group and volume to reflect our new partition by using the **resize2fs** command below:

[root@localhost mercury]# resize2fs /dev/VolGroup00/LogVol00
resize2fs 1.41.12 (17-May-2010)
Filesystem at /dev/VolGroup00/LogVol00 is mounted on /; on-line resizing required
old desc_blocks = 5, new_desc_blocks = 10
Performing an on-line resize of /dev/VolGroup00/LogVol00 to 40632320 (4k) blocks.
The filesystem on /dev/VolGroup00/LogVol00 is now 40632320 blocks long.

With the above completed, you can issue the **lvdisplay** command to verify the new volume size.