aruba
a Hewlett Packard Enterprise company

VRD

HPE FLEXFABRIC 59XX TOR SWITCH SW UPGRADE PROCEDURE

PROVIDING IRF BASED LAN/SAN ISSU SW UPGRADES AT TOR IN DC

TABLE OF CONTENTS

Introduction

This VRD provides information on performing non-disruptive software updates when using Hewlett Packard Enterprise data center Top-of-rack (ToR) 59xx Switch Series. The document will provide information on upgrades within local area network (LAN) and converged Fibre Channel (FC) and Fibre Channel over Ethernet (FCoE) storage area network (SAN) environments.

The intention of the DC ToR IRF-Based ISSU procedure is to ensure that both LAN and SAN traffic always have at least one path available for traffic during the entire upgrade.

The intended audience for this document are IT administrators and solution architects who manage and deploy FC/FCoE solutions using Hewlett Packard Enterprise Networking switches.

Background information

Non-disruptive software updates, also known as in-service software upgrade (ISSU), is a term that generally refers to techniques which allow admins to update software in a non-disruptive fashion on a network device while still ensuring that there is a path for traffic to flow.

This document will describe the types of ISSU updates that can occur with HPE DC 59xx ToR switches based on the code used and the solution that has been deployed. It will provide 'how to' procedures as well as best practices and example screen shots of ISSU updates.

Note:
Intelligent Resilient Framework (IRF) is a switch virtualization technology which allows multiple physical switches to be virtualized into a single logical switch, providing for simpler management, flatter topologies, and reducing the need for aggregation layers and protocols such as STP and VRRP. IRF should always be deployed in conjunction with Multi-Active Detection (MAD) technologies to prevent split brain scenarios. Details about IRF and MAD are beyond the scope of this document. Please refer to your product's IRF Configuration Guide for more information.

With the exception of special file or service patch upgrades, all ISSU upgrades with Comware switches must leverage IRF to avoid network disruption. Please note that with older 5900s there was a few versions of code that supported single box ISSU without IRF. However, moving forward all Comware ISSU upgrades will require IRF to be non-disruptive.

Fundamentals

When looking at ISSU upgrades with HPE DC 59xx ToR switches we need to first understand that there are two main types of upgrade types that can occur:

Compatible upgrade:

- The running software version is compatible with the upgrade software version
- This upgrade type supports the ISSU methods
- There is a code rollback possibility

Incompatible upgrade:

- The running software version is incompatible with the upgrade software version
- The two software versions cannot run concurrently
- This upgrade type supports only the called incompatible upgrade method
- This method requires a reboot of the data and control plane. It is service disruptive if hardware redundancy is not available
- There is no rollback capability

Incompatible upgrade

The Incompatible upgrade ISSU method usually needs to be performed when upgrading from one major code release to another major code release (i.e., R23xx to R24xx). However, though this can be described as a worst case scenario, it is possible to perform an Incompatible upgrade using an IRF-based solution without seriously impacting the network users.

Compatible reboot upgrade

As shown in the table below, the compatible upgrade type has a few types of ISSU methods which can be used. The most common upgrade type within major code releases (i.e. code within R23xx) will be the Compatible reboot method. This method requires that both the control and data plane of the DC ToR switch be rebooted for the upgrade to complete. However, when used in conjunction with HPE IRF, this type of upgrade can ensure that the network users will not be affected whether they are in a L2 or L3 environment.

This document will focus mainly on the Incompatible upgrade and the Compatible reboot upgrade since these are the most common types of upgrades.

Service and file incremental upgrades

Service and file incremental upgrades are special upgrades which are upgrading either hidden system files or upgrading specific service features. A hidden file upgrade can provide for no impact in standalone as well as IRF-based solutions, and the service upgrade will only affect the specific services that need to be updated. For example, if the OSPF service is getting an update then it will only affect the OSPF process of the switch. Please refer to the Fundamentals Configuration Guide if needing to perform one of these upgrades.

Table 1. ISSU upgrade types, methods, and expected network impact

ISSU methods		Upgrade type	Description
Upgrades segments that contain differences between new and old versions. **Sub-second upgrade**	Service upgrade	Compatible	• Upgrades service features. • The upgrade does not affect the operation of features that are not being upgraded.
	File upgrade	Compatible	• Upgrades hidden system program files. • No service interruption.
Reboot		Compatible	New and old software versions can run concurrently on IRF members. This method reboots both the control and data planes and is service disruptive if the device stands alone. IRF is required to avoid disruption.
Incompatible upgrade **MAD&reboot**		Incompatible	Software versions cannot run concurrently on IRF members. This method reboots both the control and data planes and is service disruptive if the device stands alone. IRF is required to avoid disruption.

SAN design

The critical nature of SAN traffic requires that the SAN network be able to provide a resilient and redundant fabric. To provide this level of high availability no-single-point-of-failure (NSPOF) configurations, SAN designs where developed which consist of at least two separate fabrics providing multiple paths between the initiator and the target. These traditional NSPOF networks deploy what is known as a “physical air-gap” between the SAN fabrics. This “physical air-gap” essentially means that there are two side by side SAN fabrics purely for redundancy. A switch or a link in one fabric can fail or go down, and the other fabric will still be able to ensure there is an available path in between the initiator and target.

Hewlett Packard Enterprise Networking solutions can provide “physical air-gap” solutions, but they are also able to provide the same level of redundancy while also leveraging IRF. When using IRF within SAN solutions a “logical air-gap” can be created which simulates the “physical air-gap” within traditional SAN solutions. This logical separation is created by using VSANs and port assignments to separate traffic. For example, SAN isolation can be achieved by ensuring that only VSAN 1 has been enabled for ports located on one of the physical switches, while VSAN 2 has been enabled for ports only located on the other physical switch.

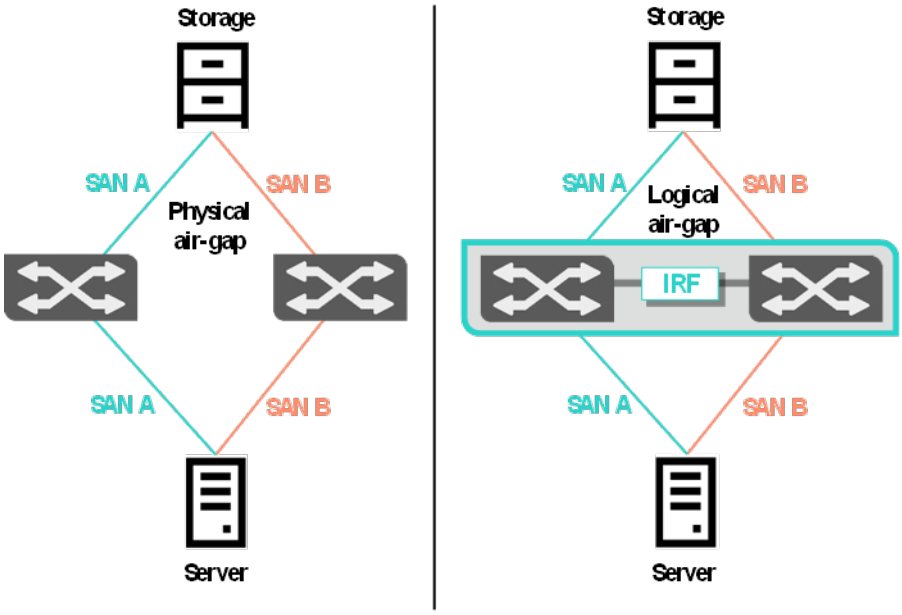


Figure 1. Physical vs logical “air-gap” between SANs

SAN software upgrades

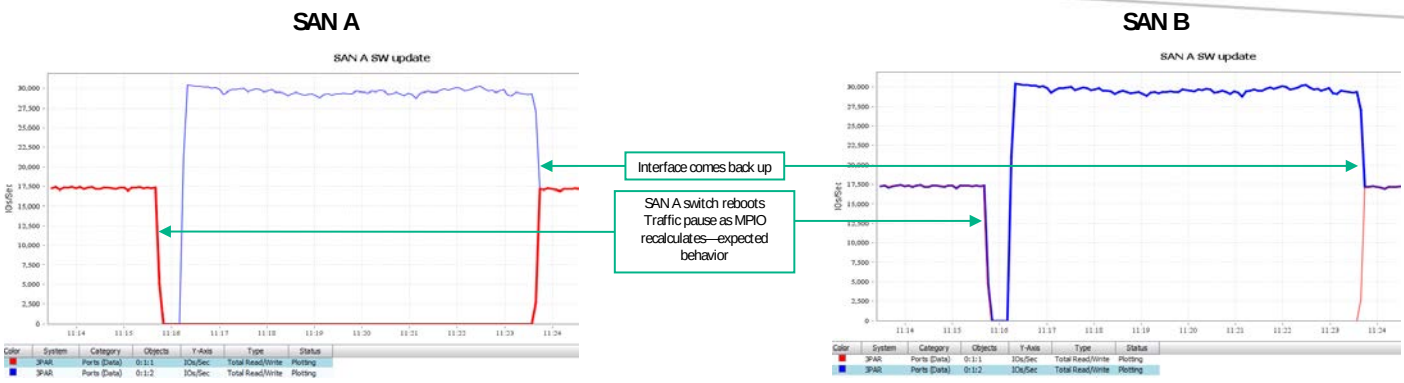
Performing a SW upgrade to switches in a SAN fabric that utilize a “physical air-gap” is straight forward. Admins can upgrade SW on the switches one fabric at a time (i.e., first SAN A and then SAN B). This ensures that there is always at least one path available between hosts and targets.

When using IRF, admins can also ensure a single path is always available when performing SW upgrades.

SAN upgrade expectations

The below figure details what occurs to SAN traffic when using a “physical air-gap”.

As shown in the image, before the upgrade starts traffic to the SAN device is balanced across both physical paths to the storage unit, thanks to MPIO software on the server. As the admin reboots the SAN A switch to upgrade the software, the MPIO software sees a path fail. The MPIO software will then pause all SAN traffic across both paths as it recalculates which stable path still exists in which it can send traffic. After it has determined that the SAN B path is still up and stable, it then resumes traffic across the SAN B path while Switch A reboots. After Switch A boots up with the new software we can see that traffic then gets balanced across both paths again.



Upgrade code compatibility

The below two tables detail the code compatibility level seen with example versions of the R23xx and R24xx code streams.

Table 2. R23xx code compatibility stream

From To	R2307	R2308P01	R2310	R2311P01	R2311P05	R2311P06	R24xx
R2307	N/A	Compatible: ISSU reboot	Compatible: ISSU reboot	Compatible: ISSU reboot	Compatible: ISSU reboot	Compatible: ISSU reboot	Incompatible: MAD&reboot
R2308P01	Compatible: ISSU reboot	N/A	Compatible: ISSU reboot	Compatible: ISSU reboot	Compatible: ISSU reboot	Compatible: ISSU reboot	Incompatible: MAD&reboot
R2310	Compatible: ISSU reboot	Compatible: ISSU reboot	N/A	Compatible: ISSU reboot	Compatible: ISSU reboot	Compatible: ISSU reboot	Incompatible: MAD&reboot
R2311P01	Compatible: ISSU reboot	Compatible: ISSU reboot	Compatible: ISSU reboot	N/A	Compatible: ISSU reboot	Compatible: ISSU reboot	Incompatible: MAD&reboot
R2311P05	Compatible: ISSU reboot	Compatible: ISSU reboot	Compatible: ISSU reboot	Compatible: ISSU reboot	N/A	Compatible: ISSU reboot	Incompatible: MAD&reboot
R2311P06	Compatible: ISSU reboot	Compatible: ISSU reboot	Compatible: ISSU reboot	Compatible: SSU reboot	Compatible: ISSU reboot	N/A	Incompatible: MAD&reboot
R24xx	Incompatible: MAD&reboot	Incompatible: MAD&reboot	Incompatible: MAD&reboot	Incompatible: MAD&reboot	Incompatible: MAD&reboot	Incompatible: MAD&reboot	N/A

Table 3. R24xx code compatibility stream

From To	R23xx	R2416	R2418P06	R2422P01
R23xx	N/A	Incompatible: MAD&reboot	Incompatible: MAD&reboot	Incompatible: MAD&reboot
R2416	Incompatible: MAD&reboot	N/A	Compatible: reboot	Compatible: reboot
R2418P06	Incompatible: MAD&reboot	Compatible: reboot	N/A	Compatible: reboot
R2422P01	Incompatible: MAD&reboot	Compatible: reboot	Compatible: reboot	N/A

ToR compatible reboot upgrade/incompatible upgrade process overview

The table below details the upgrade procedures when using two switch IRF-based solution. Two switch IRF-based solutions are detailed here because they are the most common type of DC ToR solution used when High Availability is a critical feature.

For Compatible reboot upgrades, the process requires that you use the 'issu' commands to perform the procedure. The process should always start on a Slave switch, and in a two stack IRF solution the process will require three total reboots. Note that the 'issu accept' command is optional. This command ends the issu rollback timer, which is defaulted to 45 minutes. The rollback timer ensures that the software will roll back to the original software if the issu accept command or the issu commit slot <x> command are not triggered within 45 minutes.

For Incompatible upgrades, the process requires that you use the 'boot-loader' commands to perform the procedure. In this case, the process should always start the switch with the lowest numbered IRF switch ID, and in a two stack IRF solution the process will require two total reboots.

Table 4. Upgrade steps—two switch IRF fabric

Compatible reboot	Incompatible (ex: slot 1 being master)
4 steps: 1. **issu load file … slot <slave>** 2. **issu run switchover** 3. **issu accept (optional)** 4. **issu commit slot <x>**	4 steps: 1. boot-loader … slot 1 2. reboot slot 1 3. boot-loader … slot 2 4. reboot slot 2
3 reboots: • Original IRF master: 2 reboots • Original IRF slave: 1 reboot	2 reboots: • Original IRF master: 1 reboot • Original IRF slave: 1 reboot

Compatible reboot upgrade details

The Compatible reboot upgrade procedures can be broken down into four steps with four phases. Since this is a Compatible upgrade the recommended commands to use are the ISSU commands, and we should always start with the Slave switch first.

- Step 0: The starting phase where the network is up running and passing traffic. However, the admin is tasked with upgrading the software on the networking devices.
- Step 1: The first step of the Compatible reboot upgrade consists of initializing the ISSU update by running the `issu load file …. Slot <slave>` command. This command starts the ISSU procedure by upgrading the SW of the Slave Slot 2 switch. As the Slave reboots and updates the SW, the Master Slot 1 switch will forward traffic for the network.
- Step 2: After the Slave Slot 2 boots back up with the new code, it will merge with the Master Slot 1 in the IRF stack. This is expected behavior as this is a Compatible upgrade in which the two software versions are allowed to exist in the same IRF fabric. To continue the upgrade, this step consists of running the `issu run switchover` command. This command tells the Slot 1 Master switch to reboot.
- Step 3: After Slot 1 reboots, it will now join the IRF fabric but now Slot 2 is the new Master and Slot 1 is now the Slave which is still running the old code. Step 3 consists of running the `issu commit slot 1` command. This command informs Slot 1 Slave to reboot and now upgrade the SW. If the solution consisted of more than two switches in the IRF stack you could upgrade them one at a time continuing with the `issu commit slot <number>` command after each switch comes back on line.
- Step 4: This is the final step in which the new Slave Slot 1 switch merges with the IRF fabric. Both switches are now running the same SW and IRF has reformed and traffic is passing normally.

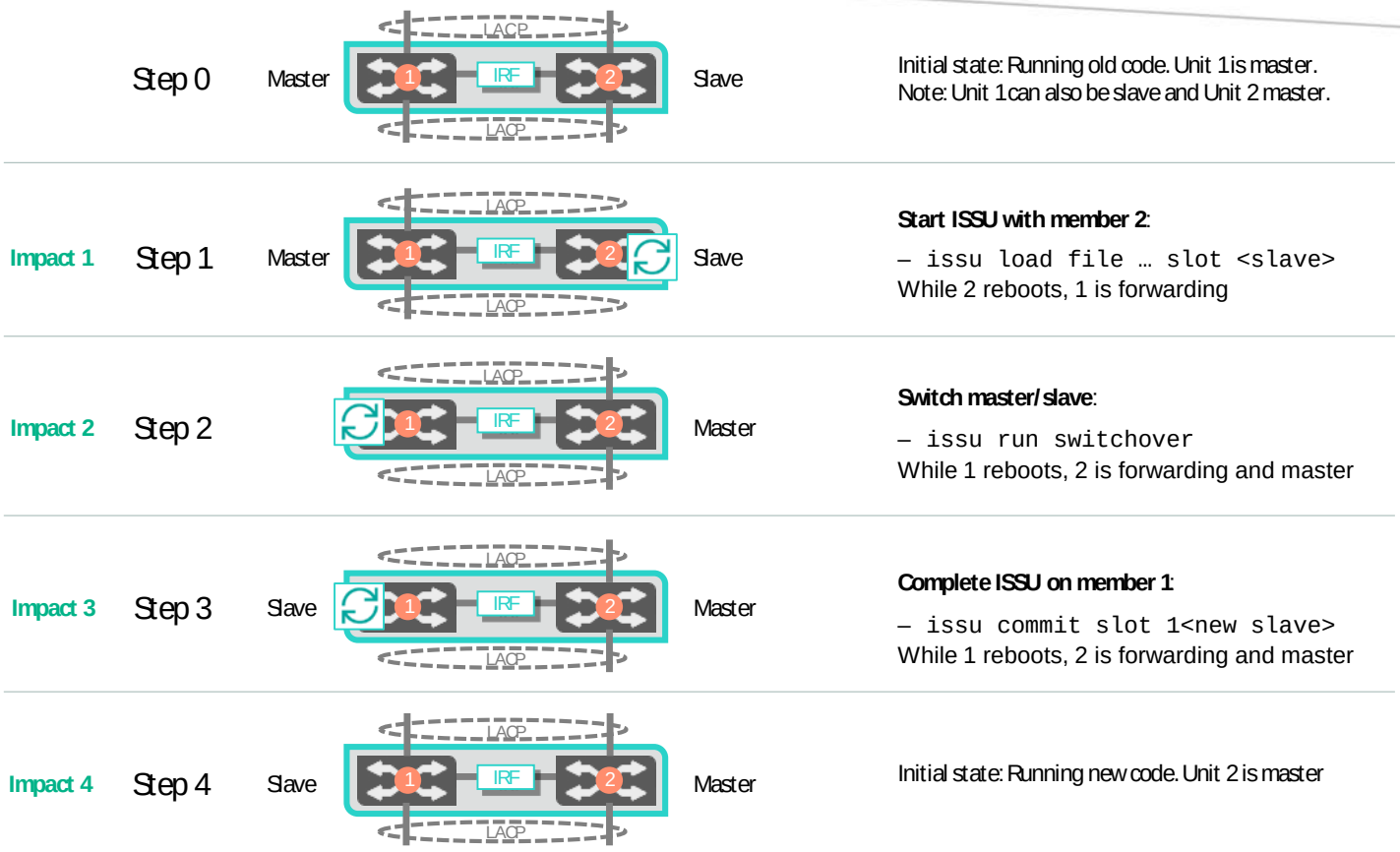


Figure 2. Compatible upgrade steps

Incompatible upgrade details

The incompatible upgrade procedures can also be broken down into four steps with four phases. Since this is an Incompatible upgrade the recommended commands to use are the boot-loader commands and always start with the lowest numbered switch first.

- Step 0: The starting phase where the network is up running and passing traffic. However, the admin is tasked with upgrading the software on the networking devices.
- Step 1: The first step of the Incompatible upgrade consists of using the `boot-loader … slot 1` command to upgrade the SW on the lowest numbered switch. At this phase the boot-loader command tells the Master Slot 1 switch to boot to the new code and then of course we need to reboot the switch so that it can actually update the SW.
- Step 2: After the Master Slot 1 switch boots back up with the new code, it will not be able to merge into the IRF stack. This is expected behavior since this is Incompatible code. Since IRF cannot reform between the two running switches, MAD takes over because it now sees a split brain scenario. Because in this discussion there are only two switches, MAD will make the decision to shut down all the ports on the higher numbered switch (Slot 2). MAD will ensure that the lowered numbered switch, Slot 1, will continue to pass traffic.
  - If the solution consisted of more than two switches then how MAD behaves will be based on which MAD mechanism is being used. The ISSU procedure in both cases is the same, but users should be aware of where and why traffic is flowing:
    - LACP MAD first looks at the number of switches in a fabric. It then looks for the largest fabric with the lowest numbered switch. If, for example, the IRF fabric had three total switches, LACP MAD will see Slots 2 and 3 forming an IRF Fabric. Slot 1 will not be in the fabric because it is running the new code which is incompatible. Traffic would be passing across Slots 2 and 3 while all the ports on Slot 1 would be down. You can then continue to upgrade Slot 2 as normal. After Slot 2 boots back up with the new code it will form an IRF fabric with Slot 1 since both of them are now running the new code. Of course this means that now MAD has shut down all the ports on Slot 3. Continue upgrading Slot 3 as normal.

- BFD MAD, ARP MAD, ND MAD all simply look for the lowest numbered Slot switch. They will all allow traffic to flow over the lowest numbered switch while shutting down all the ports on the higher numbered switches.

- Step 3: Now that MAD has shut down all the ports on Slot 2 and Slot 1 Master is now up and running with the new code, we are free to upgrade the higher numbered slots next.
- Step 4: This is the final step in which both switches are now running the same SW and IRF has reformed and traffic is passing normally.

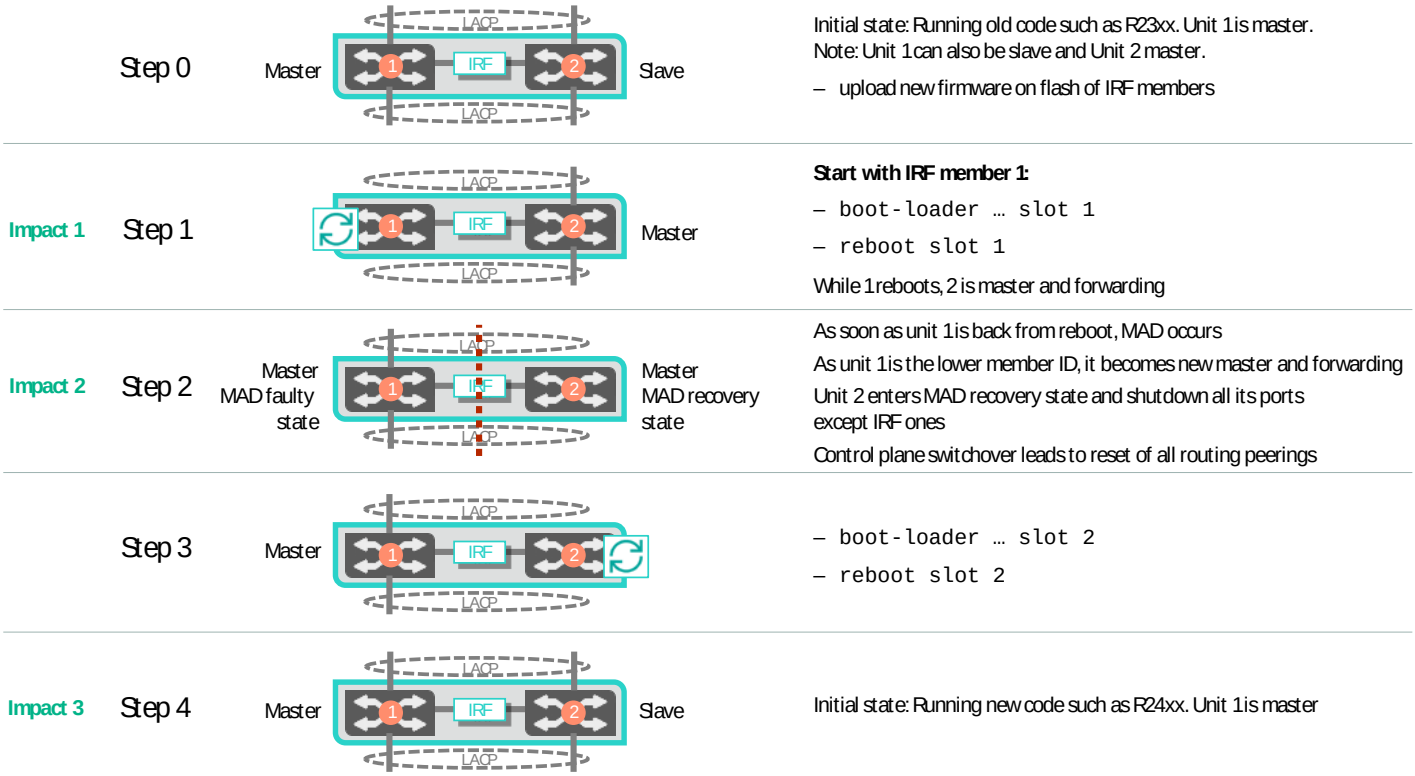


Figure 3. Incompatible upgrade steps

ISSU best practices

Follow the below best practices when performing an ISSU upgrade. Many of the steps can be seen as obvious, but the point is to not overlook any.

- Always review the release notes. Many times the release notes will contain specific information that shouldn't be overlooked.
- For critical environments always test ISSU process in sandbox environment.
- When using MAD to detect split brain scenarios, consider that BFD MAD is faster than LACP MAD. However, BFD MAD requires the use of front facing ports. Some customers may not want to burn front facing 10GbE ports for this, but that will vary with the situation.
- If connecting to other Comware switches then always use LACP and the "`link-aggregation lacp traffic-redirect-notification enable`" (enabled at system level) feature. When you restart either an IRF member device or the device's card that contains Selected ports, this feature redirects traffic of the IRF member device or card to other IRF member devices or cards.
- Use the long LACP timeout interval. The long timeout is enabled by default with Comware, so make sure that the `lacp period short` command has not been configured on an interface.
- Ensure that link utilization is less than 50% of the total LAGG bandwidth in a two switch IRF scenario.
- Choose the time of the week with the lowest traffic.
- When using dynamic routing, and if static routing is not an option, use NSR or GR for routing protocols including LDP, RSVP, OSPF, ISIS, BGP and FSPF.

- Disable BFD for protocols including LDP, RSVP, OSPF, ISIS, RIP, BGP, VRRP and NQA
- Always start with a slave switch.

Incompatible reboot scenarios

- Always use the `boot-loader` and `reboot` commands
- Always start the upgrade with the lower numbered slots first
- Temporarily use static routes in L3 scenarios (turn off BFD, GR)

ISSU pre-check list

The below list is a simple list that shows many of the commands that a user should perform to ensure that they are ready for the upgrade.

- Save the configuration to all switches:

```
<IRF Switch>save startup.cfg all
The current configuration will be saved to flash:/startup.cfg. Continue? [Y/N]:y
flash:/startup.cfg exists, overwrite? [Y/N]:y
Now saving current configuration to the device.
Saving configuration flash:/startup.cfg.Please wait...
Configuration is saved to device successfully.
Slot 2:
Configuration is saved to device successfully.
```

- Verify master slave status:

```
<IRF Switch>display irf
MemberID    Role        Priority  CPU-Mac         Description
 *+1        Master      32        cc3e-5f94-e77b  ---
   2        Standby     20        cc3e-5f94-eb2c  ---
--------------------------------------------------
* indicates the device is the master.
+ indicates the device through which the user logs in.
```

- Verify interfaces are stable and up:

```
<IRF Switch>display interface brief | include UP
Fc1/0/18   10   F     auto  F    8G     UP
Fc2/0/17   20   F     auto  F    8G     UP
Vfc11      F     on    TF   UP     XGE1/0/1
Vfc21      F     on    TF   UP     XGE2/0/1
InLoop0             UP   UP(s)    --
M-GE0/0/0           UP   UP       10.10.10.66
NULL0               UP   UP(s)    --
Vlan1               UP   UP       10.1.1.233
BAGG11              UP   20G(a)  F(a)   T    1    to Pod1_Win2012
BAGG100             UP   20G(a)  F(a)   T    1    to LAN
BAGG111             UP   20G(a)  F(a)   T    1
BAGG112             UP   20G(a)  F(a)   T    1
```

……

- Verify MAD is enabled:

```
<IRF Switch>display mad verbose
Current MAD status: Detect
Excluded ports(configurable):
Excluded ports(can not be configured):
  FortyGigE1/0/49
  FortyGigE1/0/50
```

```
  FortyGigE2/0/49
  FortyGigE2/0/50
MAD ARP disabled.
MAD ND disabled.
MAD enabled aggregation port:
  Bridge-Aggregation100
```

- Verify installed software version:

```
<IRF Switch>display version
HPE Comware Software, Version 7.1.045, Release 2311P06
Copyright (c) 2010-2015 Hewlett-Packard Development Company, L.P.
HPE FF 5900CP-48XG-4QSFP+ Switch uptime is 1 week, 0 days, 3 hours, 35 minutes
Last reboot reason : Other
Boot image: flash:/5900_5920-cmw710-boot-r2311p06.bin
Boot image version: 7.1.045P23, Release 2311P06
  Compiled Apr 02 2015 11:36:38
System image: flash:/5900_5920-cmw710-system-r2311p06.bin
System image version: 7.1.045, Release 2311P06
Compiled Apr 02 2015 11:36:54


<IRF Switch>display install active
Active packages on slot 1:
  flash:/5900_5920-cmw710-boot-r2311p06.bin
  flash:/5900_5920-cmw710-system-r2311p06.bin
Active packages on slot 2:
  flash:/5900_5920-cmw710-boot-r2311p06.bin
  flash:/5900_5920-cmw710-system-r2311p06.bin
```

- Verify all modules are working normally-if Fault state exists ISSU procedure should not start:

```
<IRF Switch>display device
Slot 1
SubSNo PortNum PCBVer  FPGAVer  CPLDVer  BootRomVer AddrLM Type       State
0      52      Ver.B   NULL     002 002  132        IVL    MAIN       Normal

Slot 2
SubSNo PortNum PCBVer  FPGAVer  CPLDVer  BootRomVer AddrLM Type       State
0      52      Ver.B   NULL     002 002  132        IVL    MAIN       Normal
```

- Verify all ports in every LAGG is Selected (S) state:

```
<IRF Switch>display link-aggregation verbose Bridge-Aggregation 100
Loadsharing Type: Shar -- Loadsharing, NonS -- Non-Loadsharing
Port Status: S -- Selected, U -- Unselected, I -- Individual
Flags:  A -- LACP_Activity, B -- LACP_Timeout, C -- Aggregation,
        D -- Synchronization, E -- Collecting, F -- Distributing,
        G -- Defaulted, H -- Expired

Aggregate Interface: Bridge-Aggregation100
Aggregation Mode: Dynamic
Loadsharing Type: Shar
System ID: 0x8000, cc3e-5f94-eb2b
Local:
  Port            Status  Priority Oper-Key  Flag
--------------------------------------------------------------------------------
  XGE1/0/48       S       32768    1         {ACDEF}
  XGE2/0/48       S       32768    1         {ACDEF}
```

- Verify current settings for boot-loader:

```
<IRF Switch>display boot-loader
Software images on slot 1:
Current software images:
  flash:/5900_5920-cmw710-boot-r2311p06.bin
  flash:/5900_5920-cmw710-system-r2311p06.bin
Main startup software images:
  flash:/5900_5920-cmw710-boot-r2311p06.bin
  flash:/5900_5920-cmw710-system-r2311p06.bin
Backup startup software images:
  None
Software images on slot 2:
Current software images:
  flash:/5900_5920-cmw710-boot-r2311p06.bin
  flash:/5900_5920-cmw710-system-r2311p06.bin
Main startup software images:
  flash:/5900_5920-cmw710-boot-r2311p06.bin
  flash:/5900_5920-cmw710-system-r2311p06.bin
Backup startup software images:
  None
```

- Verify the new code has been installed on all members:

```
<IRF Switch>dir slot1#flash:/
Directory of flash:
   0 -rw-     8150016 Jan 27 2012 00:10:19   5900_5920-cmw710-boot-r2307.bin
   1 -rw-     8225792 Jan 27 2012 00:19:18   5900_5920-cmw710-boot-r2308p01.bin
   6 -rw-    52889600 Jan 27 2012 00:12:55   5900_5920-cmw710-system-r2307.bin
   7 -rw-    52940800 Jan 27 2012 00:21:54   5900_5920-cmw710-system-r2308p01.bin
```

- Verify the state of ISSU – ensure process has not been started:

```
<IRF Switch>display issu state
ISSU state: Init
Compatibility: Unknown
Work state: Normal
Upgrade method: Card by card
Upgraded slot: None
Current upgrading slot: None
Current version list:
  boot: 7.1.045P23, Release 2311P06
  system: 7.1.045, Release 2311P06
Current software images:
  flash:/5900_5920-cmw710-boot-r2311p06.bin
  flash:/5900_5920-cmw710-system-r2311p06.bin
```

- Verify code compatibility:
  - `display version comp-matrix file boot flash:/5900_5920-cmw710-boot-r2416.bin system flash:/5900_5920-cmw710-system-r2416.bin`

Upgrade examples and results

Below is an image which shows the test environment used to perform each of these upgrades.

The test environment consisted of the following equipment:

- HPE FlexFabric 5900CP Switch—quantity 2
- HPE ProLiant BL460c G9—quantity 1
  - Windows Server® 2012
- HPE 3PAR 7200—quantity 1
  - HPE 3PAR OS version 3.1.3
- Ixia test machine
- HPE FlexFabric 5900AF switches—quantity 1
- HPE FlexFabric 12900 switch—quantity 1

The test environment is a L2 ToR solution including a SAN connection and will be shown in the results section. The Ixia test machine in these tests was connected to an upstream switch (HPE FF 12900) which connects to the pair of HPE FlexFabric 5900CPs which need upgrading. This upstream switch connects to the HPE FlexFabric 5900CPs using LACP with the traffic redirect feature enabled.

The Ixia test machine was also connected to a downstream switch (HPE FF 5900AF) which connects to the pair of HPE FlexFabric 5900CPs. This downstream switch connects to the HPE FlexFabric 5900CPs using LACP but the traffic redirect feature was not enabled, since servers are devices which will not support this feature.

The Ixia device flooded frames between these upstream and downstream switches throughout the tests.

During the upgrade the Windows Server was also sending pings to the LAN switch (HPE FF 12900).

For SAN traffic, a large 500GbE file transfer from the Windows Server to the HPE 3Par storage unit was started just before the upgrade procedure started.

During the upgrade process the Ixia results, pings and SAN traffic data was gathered at each phase of possible network impact and the results will be shown in the results section.

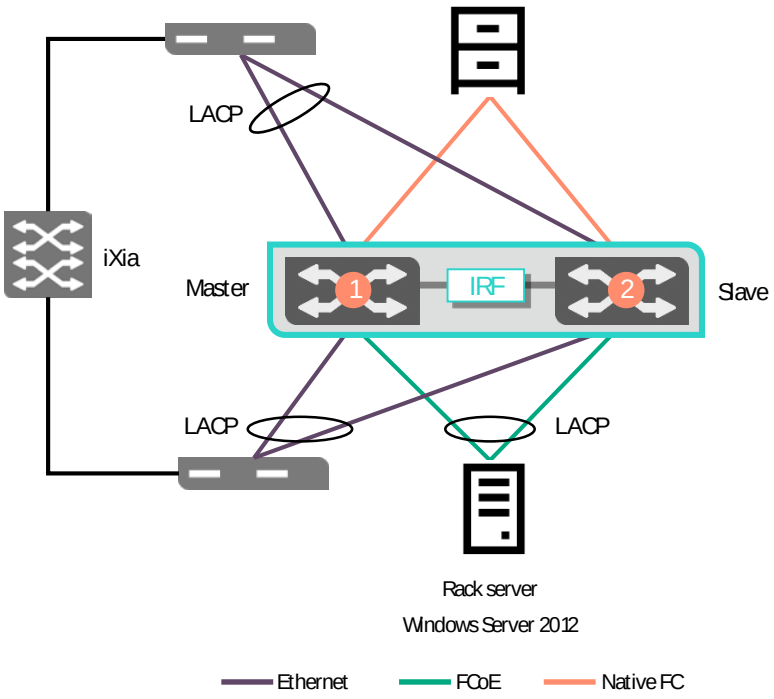


Figure 4. Test environment

Compatible reboot upgrade (R2418p06 to R2418p01)

As shown in table 4, the Compatible reboot process consist of four steps, one of which is optional.

Going from R2418p06 to R2418p01 was chosen because the SAN LLDP issue does not exist when going from R2418p06 to R2418p01.

As already mentioned, the Compatible upgrade should always start with a Slave switch. The four steps are as follows:

1. `issu load file … slot <slave>`
2. `issu run switchover`
3. `issu accept (optional)`
4. `issu commit slot <x>`

Compatible reboot upgrade results (R2418p06 to R2418p01)

The Compatible reboot upgrade results are shown in the below image.

At Step 1, the first potential impact step showed zero pings lost with **5.874 ms** of frame loss from the Ixia device.

At Step 2, the second potential impact step again showed zero pings lost with **11.366 ms** of frame loss from the Ixia device.

At Step 3, the third potential impact step showed zero pings lost with **14.917 ms** of frame loss from the Ixia device.

At Step 4, the fourth potential impact step showed zero pings lost with **6.142 ms** of frame loss from the Ixia device.

As shown with these results the total time that the network saw any disruption consists of a total of **38.299 ms.**

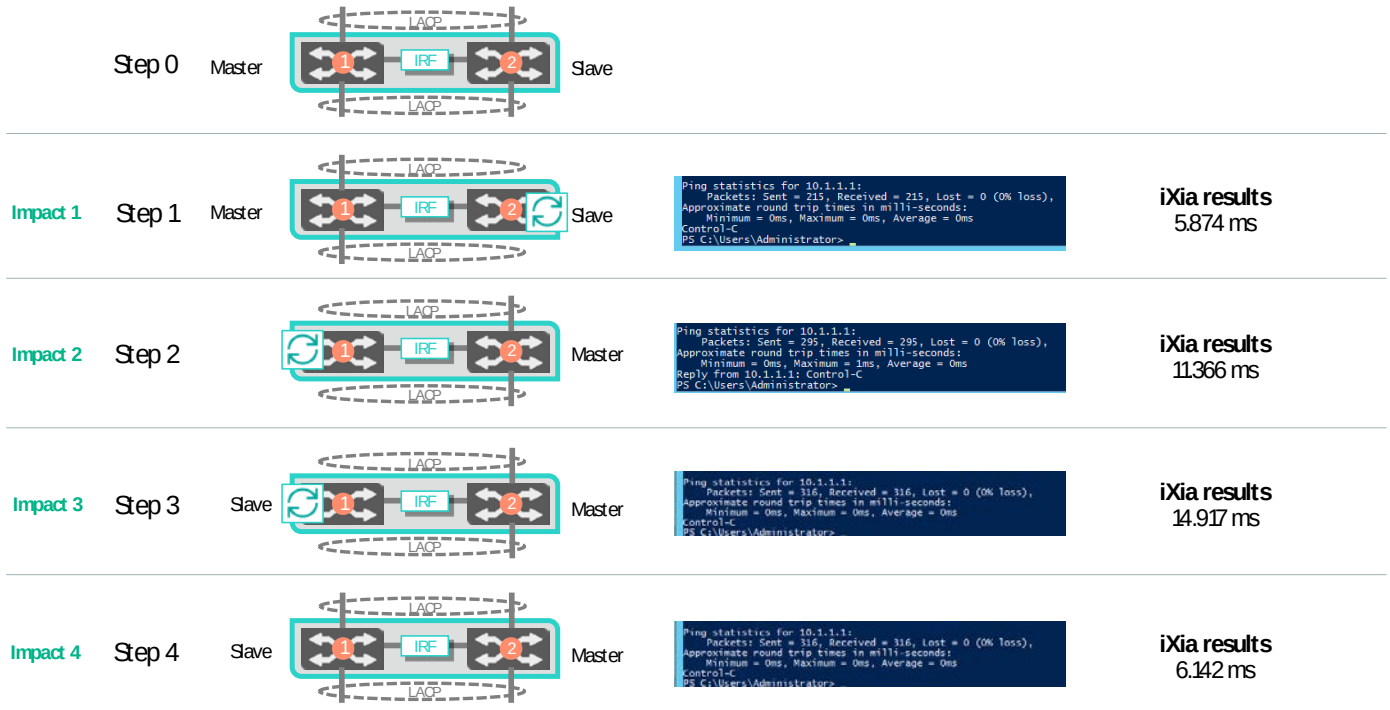


Figure 5. Compatible upgrade LAN results

With the SAN environment the 500GB file is transferred throughout the whole upgrade. At each stage when a switch is rebooted, we saw the expected behavior with the MPIO software pausing all traffic on both SAN A and SAN B. Once it recalculated the paths and saw a stable path, the MPIO SW resumed traffic on the remaining path. The file finished transferring right at the last IRF merge.

A hash was checked of the file before and after transferring and the hash matched on both ends.

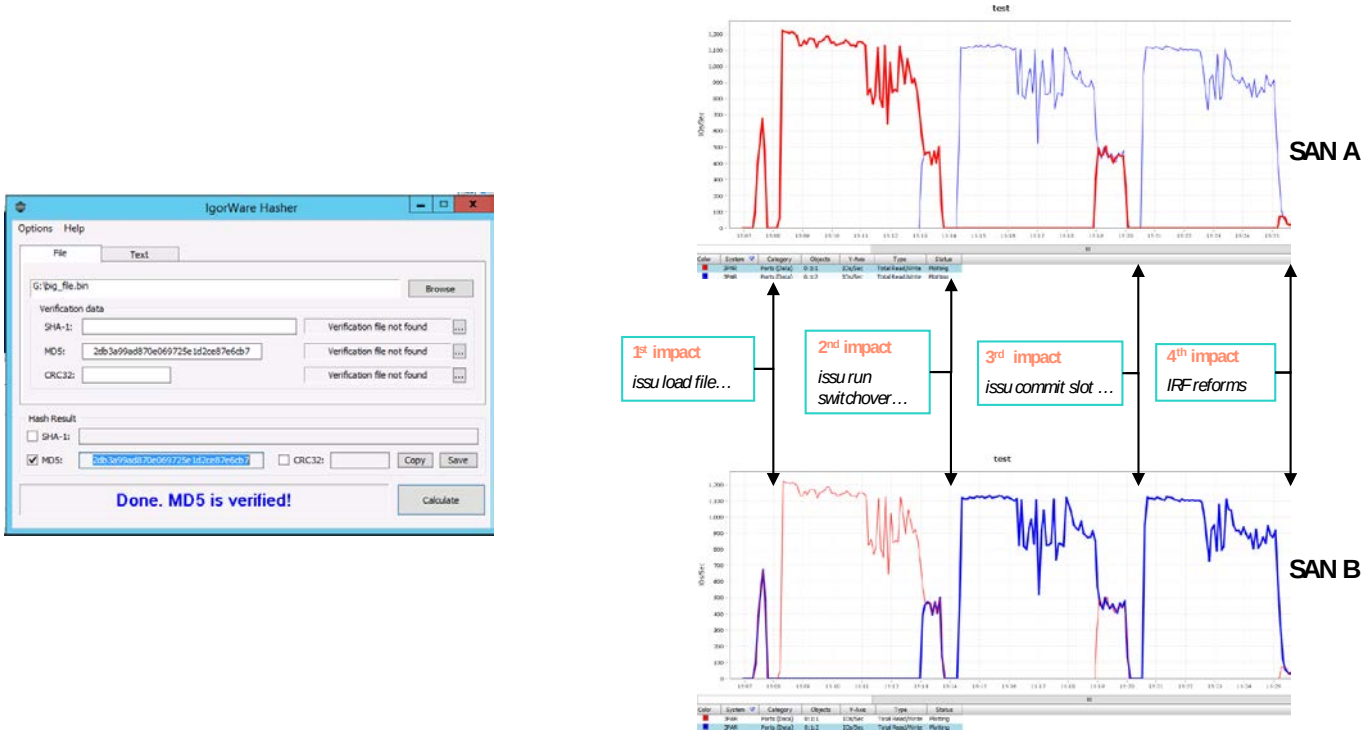


Figure 6. Compatible upgrade SAN results

Incompatible upgrade results (R2311p05 to R2422p01)

The Incompatible upgrade results are shown in the below image.

At Step 1, the first potential impact step showed zero pings lost with **5.735ms** of frame loss from the Ixia device.

At Step 2, the second potential impact step showed one ping drop and **253.939 ms** of frame loss from the Ixia device.

Step 3 does not present a potential impact because MAD already has all the ports shut on Slot 2.

At Step 4, the third potential impact step showed zero pings lost with **4.454 ms** of frame loss from the Ixia device.

As shown with these results the total time that the network saw a disruption consisted of a total of **264.128 ms.** The increased time here comes at Step 2 in which IRF tried to reform, but it couldn't because the code is incompatible code. It then ran the MAD negation shutting down all the ports on Slot 2 and allowing Slot 1 to forward traffic.

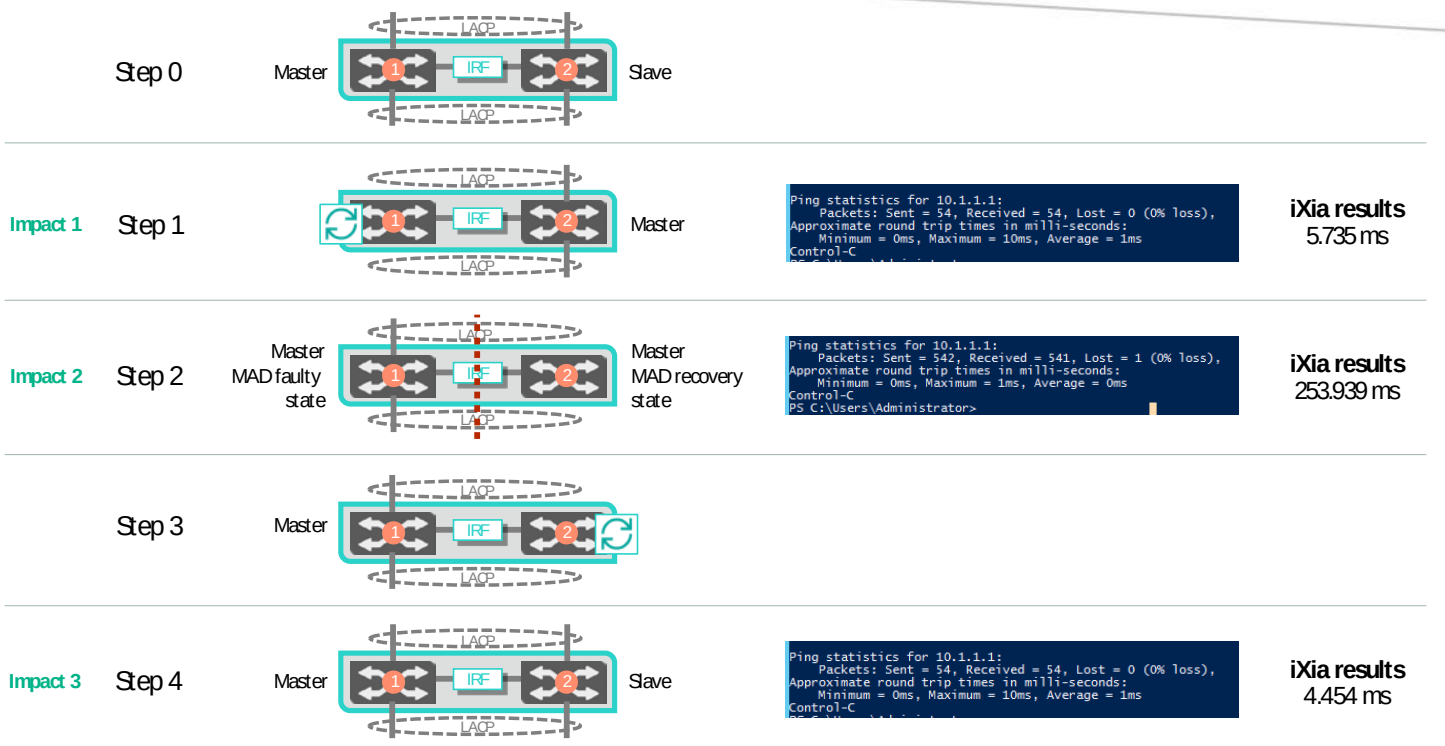


Figure 7. Incompatible upgrade LAN results

With the SAN environment the 500GB file is transferred throughout the whole upgrade. At each stage when a switch is rebooted, we saw the expected behavior with the MPIO software pausing all traffic on both SAN A and SAN B. Once it recalculated the paths and saw a stable path the MPIO SW resumed traffic on the remaining path. The file finished transferring right at the last IRF merge.

A hash was checked of the file before and after transferring and the hash matched on both ends.

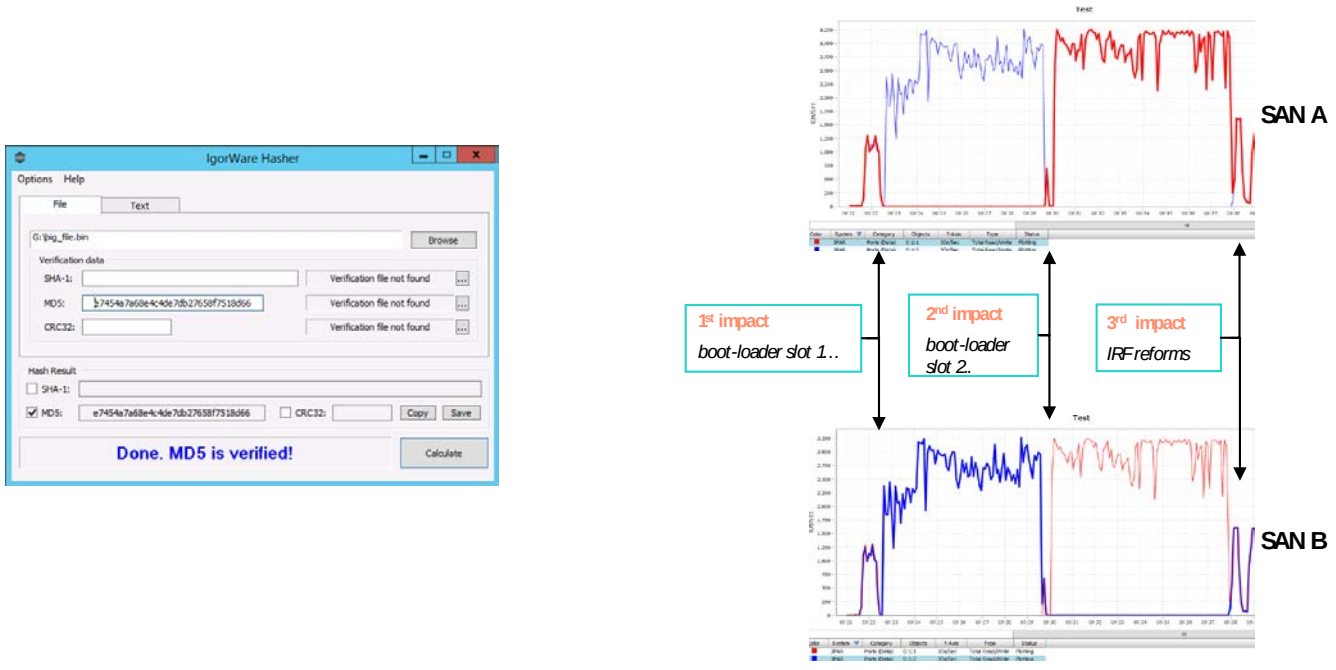


Figure 8. Incompatible upgrade SAN results

Summary

Most network admins dread having to do a SW upgrade on critical components within the DC. However, upgrading SW on networking devices is something that cannot be avoided.

As shown in the above two test results, the Compatible reboot upgrade had the smallest impact on the network with a total of 38.299 ms of network disruption. The Incompatible upgrade 'worst-case' method saw a total of 264.128 ms of network disruption.

In both cases the SAN file transfer proceeded throughout the whole upgrade process and even after the file transfer the hash check matched.

The HPE FlexFabric 59xx switch series can leverage IRF to perform either a compatible or incompatible upgrade while ensuring the network users are not aware the upgrade just occurred.

Resources, contacts, or additional links

HPE Data Center Networking

HPE FlexFabric Networking

HPE Storage

HPE Servers

3333 Scott Blvd. | Santa Clara, CA 95054
1.844.472.2782 | T: 1.408.227.4500 | FAX: 1.408.227.4550 | info@arubanetworks.com