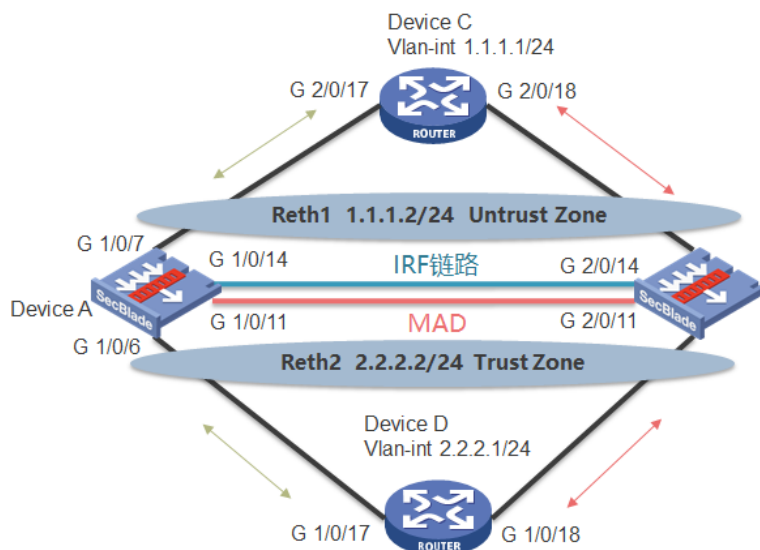


知 防火墙拆堆叠升级

IRF 冗余组 NQA Track BFD 陈启敏 2021-03-30 发表

组网及说明



两台防火墙F1070组成IRF，做冗余主备，并配置BFD MAD 进行分裂检测。业务在deviceA上跑，若deviceA上下行链路故障，则切换到deviceB

先要求在业务中断最短的情况下，堆叠主备防火墙从域间策略的软件版本升级至安全策略的软件版本

现有版本为 Release R9345P14

升级版本为 Release R9333P35

配置步骤

1 收集与备份：

```
<F1070>dir
<F1070>display version
<F1070>display irf
<F1070>display mad verbose
<F1070>save //ftp备份startup.cfg
```

2 查看剩余空间，上传升级版本，复制升级版本（主备），指定为下次启动文件：

```
<F1070>dir
<F1070>dir slot1#flash:/
<F1070>dir slot2#flash:/
<F1070>delete //删除多余文件
<F1070>reset recycle-bin //清空回收站
ftp上传版本文件
<F1070>copy flash:/ xxx.ipe slot1#flash:/
<F1070>copy flash:/ xxx.ipe slot2#flash:/
<F1070>boot-load file flash:/xxx.ipe slot 1 main //指定重启加载版本文件
<F1070>boot-load file flash:/xxx.ipe slot 2 main
<F1070>display boot-loader
```

3 关闭MAD检测口，shutdown主框的业务口，确认业务切换到备框，业务测试正常后，保存配置（之后割接过程中不再保存配置）：

```
[F1070]interface Route-Aggregation 3
[F1070-Route-Aggregation3]undo mad bfd enable
[F1070]int range g 1/0/6 to g 1/0/7
[F1070-if-range]shutdown
[F1070]save
[F1070]display redundancy group aaa
[F1070]display reth int reth 2
[F1070]dis reth int reth 2
```

4 断开堆叠线路，建议通过拔线方式，使堆叠分裂；

```
<F1070>display irf
<F1070>display irf link
<F1070>display irf configuration
```

5 确认业务正常后，重启主框：

```
[F1070]display redundancy group aaa
[F1070]display reth int reth 2
[F1070]display reth int reth 2
<F1070>reboot slot 1 （主框）
```

6 通过display system stable state确认主框板卡正常后，通过 security-policy switch-from object-policy startup.cfg xx.cfg实现对象策略转换为安全策略，再次重启设备；

```
<F1070>display device
<F1070>display sys stable state
[F1070]security-policy switch-from object-policy startup.cfg security-startup.cfg
<F1070>reboot slot 1
```

7 通过display system stable state确认主框板卡正常后，shutdown 备框业务口，同时undo shutdown 主框业务口，测试业务；

```
<F1070>display version
<F1070>display sys stable state
<F1070>display security-policy ip
[F1070]interface range g 2/0/6 to g 2/0/7 （备框）
[F1070-if-range]shutdown （备框）
[F1070]interface range g 1/0/6 to g 1/0/7 （主框）
```

[F1070-if-range]undo shutdown (主框)

[F1070]display redundancy group aaa

[F1070]display reth int reth 1

[F1070]display reth int reth 2

配置关键点

升级前准备:

1. 备份配置并收集设备运行信息, 查看flash空间是否充足同时上传和复制版本文件

2. 业务正常后, 重启备框。重启过程中, 恢复堆叠链路, 备框重启完成后, 自动加入堆叠。

件到设备

<F1070>reboot slot 2

升级过程中: 注意: 不要保存配置

1. 指定版本文件;

2. 关闭MAD检测和主框业务口并保存配置;

3. 再启主框并在主框启动完成前拔掉堆叠线;

4. 先关闭备框业务口再打开主框业务口 (间隔越短业务中断时间越短);

5. 确认业务正常运行后重启备框并在重启完成之前插入堆叠线 (注意重启时会提示是否保存当前配置, 此时一定不能保存配置) (假如备框重启完毕未能加入主框的堆叠, 则可能导致转发不通等异常。)

6. 测试业务正常后, 恢复MAD线路, 保存查看配置

7. 确认业务恢复情况并恢复MAD功能

[F1070-Route-Aggregation3]mad bfd enable

[F1070-Route-Aggregation3]mad ip add 192.168.2.1 24 member 1

序号	操作步骤	是否影响转发	业务中断时间
1	收集与备份	否	0
2	查看剩余空间, 上传升级版本, 复制升级版本 (主备), 指定为下次启动文件	否	0
3	关闭MAD检测口, shutdown主框的业务口, 确认业务切换到备框, 业务测试正常后, 保存配置 (之后割接过程中不再保存配置)	是	<3s
4	断开堆叠线路, 建议通过拔线方式, 使堆叠分裂;	否	0
5	确认业务正常后, 重启主框	否	0
6	通过display system stable state确认主框板卡正常后, 通过 security-policy switch-from object-policy startup.cfg xx.cfg实现对象策略转换为安全策略, 再次重启设备	否	0
7	通过display system stable state确认主框板卡正常后, shutdown备框业务口, 同时undo shutdown主框业务口, 测试业务;	是	<60s
8	业务正常后, 重启备框。重启过程中, 恢复堆叠链路, 备框重启完成后, 自动加入堆叠	否	0
9	测试业务正常后, 恢复MAD线路, 保存查看配置	否	0

10485/6 KB total (745584 KB free)

<F1070>dir slot1#flash:/

10485/6 KB total (745584 KB free)

<F1070>dir slot2#flash:/

10485/6 KB total (745584 KB free)

堆叠升级前不要关闭MAD检测

BFD来检测网络中是否存在多个IRF, IRF分裂后, 通过分裂检测机制IRF会检测到网络中存在其它处于正常工作状态的IRF。对于LACP MAD和BFD MAD检测, 如果两个IRF中成员设备的数量, 数量多的IRF继续工作, 数量少的IRF会进入Recovery状态, 如果成员数量相等, 则主设备成员编号小的IRF会进入Recovery状态。这样会使得备框迁移到Recovery状态, 主备框都无业务口运行, 造成业务中断

2. 为什么备框重启前不要保存配置

堆叠分裂状态下所有单框重启时都不要保存配置: 单框下重启保存时都会丢失堆叠状态下单独配置在其他框的配置, 再堆叠同步时就会丢失。堆叠断裂前, 保存配置, 主备框配置都会保存在starup文件中, 接着堆叠分裂状态下单框重启时, 是不是保存配置, 那样最后堆叠重新启动调用starup的时候就是正常的配置。

3. Reth口mac地址是否相同, 为什么要相同

相同。在上、下行设备看来, 与其连接的是以太网冗余接口, 学习到的是以太网冗余接口MAC地址。成员接口的激活状态发生变化, 不会影响到上、下行设备

Preempt delay time remained :0 sec

Preempt delay time remained :0 sec

Preempt delay time remained :0 sec

Preempt delay time remained :0 sec

Preempt delay time remained :0 sec

Preempt delay time remained :0 sec

Preempt delay time remained :0 sec

Preempt delay time remained :0 sec

Preempt delay time remained :0 sec

Preempt delay time remained :0 sec

Preempt delay time remained :0 sec

Preempt delay time remained :0 sec

Preempt delay time remained :0 sec

Preempt delay time remained :0 sec

Preempt delay time remained :0 sec

Preempt delay time remained :0 sec

Preempt delay time remained :0 sec

Preempt delay time remained :0 sec

Preempt delay time remained :0 sec