

Prime Cable Provisioning 6.1.5 RDU High Availability Deployment with Geo Mode Redundancy

목차

[소개](#)

[사전 요구 사항](#)

[요구 사항](#)

[구성 요소](#)

[설치](#)

[네트워크 다이어그램](#)

[1. 두 서버에서 LVBPRHOME, LVBPRDATA 및 LVBPRDBLOG용 LVM 볼륨 생성](#)

[2. 두 서버에서 RDU HA 구축을 위해 Linux 7.4 서버를 준비합니다.](#)

[3. Geo-Redundancy 모드에서 RDU 서버 설치](#)

[4. Geo-Redundancy 구축을 위한 레이어-3 라우팅 전제 조건](#)

[RDU Geo-Redundancy](#)

[PCP Geo-Redundancy 요건](#)

[사후 확인 HA](#)

소개

이 문서에서는 HA(High Availability)에 Prime Cable Provisioning 6.1.5을 Geo 모드 이중화와 함께 설치하는 방법에 대해 설명합니다.

사전 요구 사항

요구 사항

다음 주제에 대한 지식을 보유하고 있으면 유용합니다.

- Linux에 대한 지식 및 파일 시스템 및 파티셔닝 이해
- 새 기본 및 보조 가상/물리적 시스템에 6.1.5 RHEL 7.4/Kernel 3.10.0-693.11.6.x86_64를 설치합니다. 지역 모드의 RDU HA는 이 RHEL OS 및 커널 버전 및 해당 rpm 패키지만 호환됩니다.
- Linux DRBD 파일 스토리지 복제 방법 및 Corosync-pacemaker 클러스터 개념에 대한 지식
- 네트워크 구성 파일은 FQDN(Fully Qualified Domain Name)이 아니라 시스템 호스트 이름만 포함해야 합니다.

구성 요소

이 문서의 정보는 다음 소프트웨어 및 하드웨어 버전을 기반으로 합니다.

- 플랫폼: Red Hat Linux 7.4

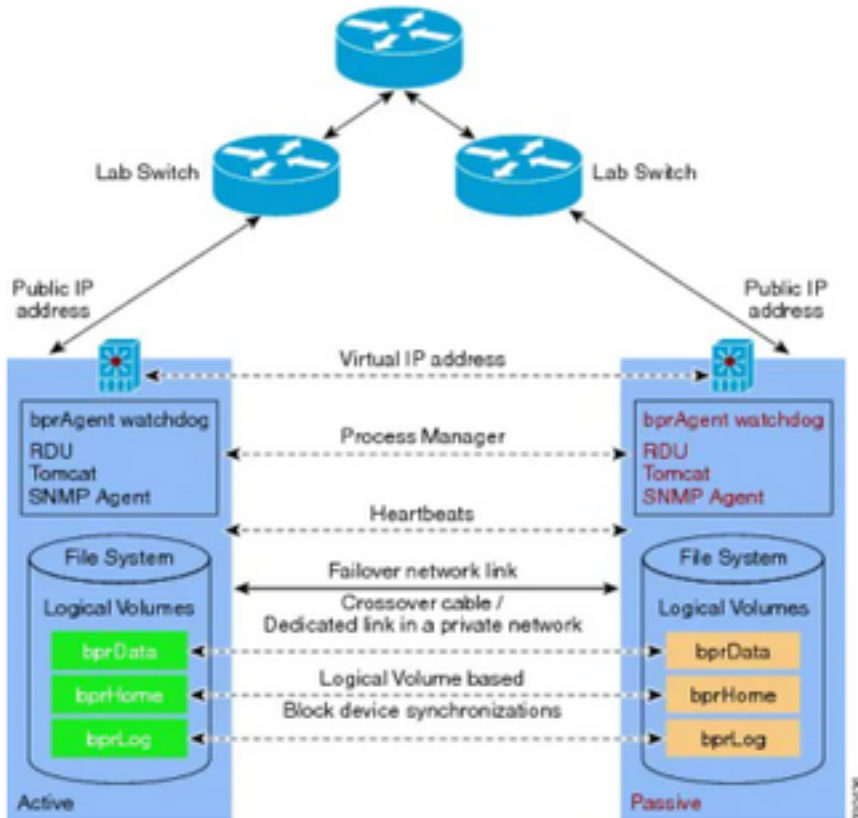
- 소프트웨어: Prime Cable 프로비저닝 6.1.5 이미지

이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다. 이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다. 네트워크가 작동 중인 경우 모든 명령의 잠재적인 영향을 이해해야 합니다.

설치

네트워크 다이어그램

Figure 1. RDU Redundancy



1. LVM은 두 서버 모두에서 LVBPRHOME, LVBPRDATA 및 LVBPRDBLOG용 볼륨을 생성합니다.
2. 두 서버에서 RDU HA 구축을 위해 Linux 7.4 서버를 준비하는 중입니다.
3. 지리적 이중화 모드에서 RDU 서버 설치
 - Geo 이중화 모드에서 RDU 서버 설치
 - HA 사전 확인 - 기본-보조 모드에서 RDU HA 설정
 - HA 설치 - 6.1.5 PCP 인스턴스를 설치합니다.
 - HA 사후 확인
4. 지역 이중화 구축을 위한 레이어 3 라우팅 전제 조건

1. 두 서버에서 LVBPRHOME, LVBPRDATA 및 LVBPRDBLOG용 LVM 볼륨 생성

이 그림은 보조 서버에 대해 수행됩니다. 주 서버에서도 동일한 절차를 완료해야 합니다.

- sda3로 새 파티션을 추가하고 fdisk 명령을 사용하여 디스크를 할당합니다.

```
[root@pcprdusecondary ~]# fdisk -l
```

```
Disk /dev/sda: 107.4 GB, 107374182400 bytes, 209715200 sectors
```

```
Units = sectors of 1 * 512 = 512 bytes
```

```
Sector size (logical/physical): 512 bytes / 512 bytes
```

```
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

```
Disk label type: dos
```

```
Disk identifier: 0x00025a26
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	2048	2099199	1048576	83	Linux
/dev/sda2		2099200	31211519	14556160	8e	Linux LVM

```
Disk /dev/mapper/rhel-root: 4294 MB, 4294967296 bytes, 8388608 sectors
```

```
Units = sectors of 1 * 512 = 512 bytes
```

```
Sector size (logical/physical): 512 bytes / 512 bytes
```

```
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

```
Disk /dev/mapper/rhel-swap: 8455 MB, 8455716864 bytes, 16515072 sectors
```

```
Units = sectors of 1 * 512 = 512 bytes
```

```
Sector size (logical/physical): 512 bytes / 512 bytes
```

```
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

```
Disk /dev/mapper/rhel-home: 2147 MB, 2147483648 bytes, 4194304 sectors
```

```
Units = sectors of 1 * 512 = 512 bytes
```

```
Sector size (logical/physical): 512 bytes / 512 bytes
```

```
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

```
[root@pcprdusecondary ~]# fdisk /dev/sda
```

```
Welcome to fdisk (util-linux 2.23.2).
```

```
Changes will remain in memory only, until you decide to write them.
```

Be careful before using the write command.

Command (m for help): **m**

Command action

- a toggle a bootable flag
- b edit bsd disklabel
- c toggle the dos compatibility flag
- d delete a partition
- g create a new empty GPT partition table
- G create an IRIX (SGI) partition table
- l list known partition types
- m print this menu
- n add a new partition
- o create a new empty DOS partition table
- p print the partition table
- q quit without saving changes
- s create a new empty Sun disklabel
- t change a partition's system id
- u change display/entry units
- v verify the partition table
- w write table to disk and exit
- x extra functionality (experts only)

Command (m for help): **p**

Disk /dev/sda: 107.4 GB, 107374182400 bytes, 209715200 sectors

Units = sectors of 1 * 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk label type: dos

Disk identifier: 0x00025a26

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

```
/dev/sda1 *          2048      2099199      1048576    83  Linux
/dev/sda2          2099200      31211519     14556160    8e  Linux LVM
```

Command (m for help): **n**

Partition type:

```
p  primary (2 primary, 0 extended, 2 free)
e  extended
```

Select (default p): **p**

Partition number (3,4, default 3): **3**

First sector (31211520-209715199, default 31211520):

Using default value 31211520

Last sector, +sectors or +size{K,M,G} (31211520-209715199, default 209715199):

Using default value 209715199

Partition 3 of type Linux and of size 85.1 GiB is set

Command (m for help): **p**

Disk /dev/sda: 107.4 GB, 107374182400 bytes, 209715200 sectors

Units = sectors of 1 * 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk label type: dos

Disk identifier: 0x00025a26

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	2048	2099199	1048576	83	Linux
/dev/sda2		2099200	31211519	14556160	8e	Linux LVM
/dev/sda3		31211520	209715199	89251840	83	Linux

Command (m for help): **t**

Partition number (1-3, default 3): **3**

Hex code (type L to list all codes): **L**

```
0  Empty          24  NEC DOS          81  Minix / old Lin bf  Solaris
```

1	FAT12	27	Hidden NTFS Win	82	Linux swap / So	c1	DRDOS/sec (FAT-
2	XENIX root	39	Plan 9	83	Linux	c4	DRDOS/sec (FAT-
3	XENIX usr	3c	PartitionMagic	84	OS/2 hidden C:	c6	DRDOS/sec (FAT-
4	FAT16 <32M	40	Venix 80286	85	Linux extended	c7	Syrinx
5	Extended	41	PPC PReP Boot	86	NTFS volume set	da	Non-FS data
6	FAT16	42	SFS	87	NTFS volume set	db	CP/M / CTOS / .
7	HPFS/NTFS/exFAT	4d	QNX4.x	88	Linux plaintext	de	Dell Utility
8	AIX	4e	QNX4.x 2nd part	8e	Linux LVM	df	BootIt
9	AIX bootable	4f	QNX4.x 3rd part	93	Amoeba	e1	DOS access
a	OS/2 Boot Manag	50	OnTrack DM	94	Amoeba BBT	e3	DOS R/O
b	W95 FAT32	51	OnTrack DM6 Aux	9f	BSD/OS	e4	SpeedStor
c	W95 FAT32 (LBA)	52	CP/M	a0	IBM Thinkpad	hi	eb BeOS fs
e	W95 FAT16 (LBA)	53	OnTrack DM6 Aux	a5	FreeBSD	ee	GPT
f	W95 Ext'd (LBA)	54	OnTrackDM6	a6	OpenBSD	ef	EFI (FAT-12/16/
10	OPUS	55	EZ-Drive	a7	NeXTSTEP	f0	Linux/PA-RISC b
11	Hidden FAT12	56	Golden Bow	a8	Darwin UFS	f1	SpeedStor
12	Compaq diagnost	5c	Priam Edisk	a9	NetBSD	f4	SpeedStor
14	Hidden FAT16	61	SpeedStor	ab	Darwin boot	f2	DOS secondary
16	Hidden FAT16	63	GNU HURD or Sys	af	HFS / HFS+	fb	VMware VMFS
17	Hidden HPFS/NTF	64	Novell Netware	b7	BSDI fs	fc	VMware VMKCORE
18	AST SmartSleep	65	Novell Netware	b8	BSDI swap	fd	Linux raid auto
1b	Hidden W95 FAT3	70	DiskSecure Mult	bb	Boot Wizard hid	fe	LANstep
1c	Hidden W95 FAT3	75	PC/IX	be	Solaris boot	ff	BBT
1e	Hidden W95 FAT1	80	Old Minix				

Hex code (type L to list all codes): 8e

Changed type of partition 'Linux' to 'Linux LVM'

Command (m for help): w

The partition table has been altered!

Calling ioctl() to re-read partition table.

WARNING: Re-reading the partition table failed with error 16: Device or resource busy.

The kernel still uses the old table. The new table will be used at

the next reboot or after you run partprobe(8) or kpartx(8)

Syncing disks.

이 오류 메시지가 필요합니다. 새 변경 사항을 적용하려면 Linux 시스템을 다시 로드해야 합니다.

```
[root@pcprdusecondary ~]# df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/mapper/rhel-root	4.0G	946M	3.1G	24%	/
devtmpfs	3.9G	0	3.9G	0%	/dev
tmpfs	3.9G	0	3.9G	0%	/dev/shm
tmpfs	3.9G	8.6M	3.9G	1%	/run
tmpfs	3.9G	0	3.9G	0%	/sys/fs/cgroup
/dev/sda1	1014M	143M	872M	15%	/boot
/dev/mapper/rhel-home	2.0G	33M	2.0G	2%	/home
tmpfs	781M	0	781M	0%	/run/user/0

```
[root@pcprdusecondary ~]# fdisk -l
```

Disk /dev/sda: 107.4 GB, 107374182400 bytes, 209715200 sectors

Units = sectors of 1 * 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk label type: dos

Disk identifier: 0x00025a26

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	2048	2099199	1048576	83	Linux
/dev/sda2		2099200	31211519	14556160	8e	Linux LVM
/dev/sda3		31211520	209715199	89251840	8e	Linux LVM

Disk /dev/mapper/rhel-root: 4294 MB, 4294967296 bytes, 8388608 sectors

Units = sectors of 1 * 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/mapper/rhel-swap: 8455 MB, 8455716864 bytes, 16515072 sectors

Units = sectors of 1 * 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/mapper/rhel-home: 2147 MB, 2147483648 bytes, 4194304 sectors

Units = sectors of 1 * 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

- sda3의 물리적 볼륨을 생성합니다.

```
[root@pcprdusecondary ~]# pvcreate /dev/sda3
[root@pcprdusecondary ~]# pvcreate /dev/sda3
Physical volume "/dev/sda3" successfully created.
[root@pcprdusecondary ~]#
```

- pvscan - 물리적 볼륨 그룹을 검사하고 나열합니다.
- vgscan - 논리 볼륨 그룹을 스캔하고 나열합니다.
- lvscan - 볼륨 그룹에 생성된 논리 볼륨을 스캔하고 나열합니다.

이 Linux LVM 생성이 RDU 서버 설치에 필요한 필수 사항입니다.

- 기본 및 보조 RDU 노드 모두에서 논리 볼륨 그룹을 3개의 논리 볼륨으로 생성해야 합니다. 논리 볼륨은 다음 사양을 기반으로 생성됩니다.

1.<Prime 케이블 프로비전 설치 디렉터리의 논리 볼륨> - /bprHome 디렉터리에 탑재되었습니다. 예: LVBPRHOME.

2.<Prime 케이블 프로비전 데이터 디렉터리의 논리 볼륨> - /bprData 디렉터리에 탑재되었습니다. 예: LVBPRDATA

3.<Prime Cable Provisioning 로그 디렉터리의 논리 볼륨> - /bprLog 디렉터리에 탑재되었습니다. 예: LVBPRDBLOG

- 요구 사항에 따라 볼륨 그룹 및 논리 볼륨을 만들고 /bprData, bprHome 및 /bprLog 디렉터리에 마운트합니다.

예: 이 절차는 3GB 디스크 공간이 있는 BPRHOME, 15GB 디스크 공간이 있는 BPRDATA, 5GB 디스크 공간 할당이 있는 BPRDBLOG용 논리 볼륨을 만드는 것입니다. 할당에 따라 확장할 디스크 공간을 선택해야 합니다.

- 볼륨 그룹을 만듭니다.


```

[root@pcprdusecondary ~]# pvscan
  PU /dev/sda2   UG rhel          lvm2 [ <13.88 GiB / 4.00 MiB free
  PU /dev/sda3           lvm2 [ <85.12 GiB]
  Total: 2 [ <99.00 GiB] / in use: 1 [ <13.88 GiB] / in no UG: 1 [ <85.
[root@pcprdusecondary ~]# vgcreate rdusesecondary /dev/sda3
  Volume group "rdusesecondary" successfully created
[root@pcprdusecondary ~]#
[root@pcprdusecondary ~]#
[root@pcprdusecondary ~]# vgs
  Reading volume groups from cache.
  Found volume group "rhel" using metadata type lvm2
  Found volume group "rdusesecondary" using metadata type lvm2
[root@pcprdusecondary ~]# pvscan
  PU /dev/sda2   UG rhel          lvm2 [ <13.88 GiB / 4.00 MiB free
  PU /dev/sda3   UG rdusesecondary lvm2 [ 85.11 GiB / 85.11 GiB free
  Total: 2 [ 98.99 GiB] / in use: 2 [ 98.99 GiB] / in no UG: 0 [ 0 ]
[root@pcprdusecondary ~]#

```

vgcreate <vg_name> <pvname>

```

[root@pcprdusecondary ~]# vgcreate rdusesecondary /dev/sda3

```

- 논리 볼륨을 만듭니다.

lvcreate -L <값 GB> -n <logicalvolumename> <볼륨 그룹 이름>

```

[root@pcprdusecondary ~]# lvcreate -L +3GB -n LVBPRHOME rdusesecondary
[root@pcprdusecondary ~]# lvcreate -L +15GB -n LVBPRDATA rdusesecondary
[root@pcprdusecondary ~]# lvcreate -L +5GB -n LVBPRDBLOG rdusesecondary

```

```

[root@pcprdusecondary ~]#
[root@pcprdusecondary ~]# lvcreate -L +3GB -n LVBPRHOME rdusesecondary
  Logical volume "LVBPRHOME" created.
[root@pcprdusecondary ~]# lvcreate -L +15GB -n LVBPRDATA rdusesecondary
  Logical volume "LVBPRDATA" created.
[root@pcprdusecondary ~]# lvcreate -L +5GB -n LVBPRDBLOG rdusesecondary
  Logical volume "LVBPRDBLOG" created.
[root@pcprdusecondary ~]#
[root@pcprdusecondary ~]# LUSCAN
-bash: LUSCAN: command not found
[root@pcprdusecondary ~]# lvs
  ACTIVE          /dev/rhel/root' [4.00 GiB] inherit
  ACTIVE          /dev/rhel/home' [2.00 GiB] inherit
  ACTIVE          /dev/rhel/swap' [7.00 GiB] inherit
  ACTIVE          /dev/rdusesecondary/LVBPRHOME' [3.00 GiB] inherit
  ACTIVE          /dev/rdusesecondary/LVBPRDATA' [15.00 GiB] inherit
  ACTIVE          /dev/rdusesecondary/LVBPRDBLOG' [5.00 GiB] inherit
[root@pcprdusecondary ~]# vgs
  Reading volume groups from cache.
  Found volume group "rhel" using metadata type lvm2
  Found volume group "rdusesecondary" using metadata type lvm2
[root@pcprdusecondary ~]# pvscan
  PU /dev/sda2   UG rhel          lvm2 [ <13.88 GiB / 4.00 MiB free]
  PU /dev/sda3   UG rdusesecondary lvm2 [ 85.11 GiB / 62.11 GiB free]
  Total: 2 [ 98.99 GiB] / in use: 2 [ 98.99 GiB] / in no UG: 0 [ 0 ]

```

bprHome - 설치 응용 프로그램 경로(기본 디렉터리 - /opt/CSCObac)

bprData - 설치 데이터 경로입니다(기본 디렉터리 - /var/CSCObac).

bprLog - 설치 로그 경로.(기본 디렉토리 - /var/CSCObac)

- lvm 파티션에 XFS 파일 시스템을 생성합니다.

mkfs.xfs /dev/<볼륨 그룹 이름>/<논리 볼륨>

```
[root@pcprdusecondary ~]# mkfs.xfs /dev/rdusecondary/LVBPRHOME
[root@pcprdusecondary ~]# mkfs.xfs /dev/rdusecondary/LVBPRDATA
[root@pcprdusecondary ~]# mkfs.xfs /dev/rdusecondary/LVBPRDBLOG
```

```
--- Logical volume ---
LU Path                /dev/rdusecondary/LVBPRDATA
LU Name                LVBPRDATA
UG Name                rdusecondary
LU UUID                d10MKX-lzuX-NzsY-zSAH-8s8T-qzq6-JM7bn
LU Write Access        read/write
LU Creation host, time pcprdusecondary.cisco.com, 2020-12-02 06:32:25 +053
LU Status               available
# open                 0
LU Size                15.00 GiB
Current LE             3840
Segments               1
Allocation              inherit
Read ahead sectors     auto
- currently set to     8192
Block device           253:4

--- Logical volume ---
LU Path                /dev/rdusecondary/LVBPRDBLOG
LU Name                LVBPRDBLOG
UG Name                rdusecondary
LU UUID                Hd1xm8-jSsf-m6Ax-tUdW-FWz-6k3G-x6zChT
LU Write Access        read/write
LU Creation host, time pcprdusecondary.cisco.com, 2020-12-02 06:34:05 +053
LU Status               available
# open                 0
LU Size                5.00 GiB
Current LE             1200
Segments               1
Allocation              inherit
Read ahead sectors     auto
- currently set to     8192
Block device           253:5
```

- 이러한 디렉토리에 bprHome, bprData, bprLog 및 마운트 논리 볼륨을 만듭니다.

```
[root@pcprdusecondary ~]# mkdir bprHome
[root@pcprdusecondary ~]# mkdir bprData
[root@pcprdusecondary ~]# mkdir bprLog
```

- 이러한 디렉토리에 논리 볼륨을 마운트합니다.

```
[root@pcprdusecondary ~]# mount /dev/RDUPRIMARY/LVBPRHOME /bprHome/
[root@pcprdusecondary ~]# mount /dev/RDUPRIMARY/LVBPRDATA /bprData/
[root@pcprdusecondary ~]# mount /dev/RDUPRIMARY/LVBPRDBLOG /bprLog
```

```
[root@pcprdusecondary ~]# df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/mapper/rhel-root	4.8G	947M	3.1G	24%	/
devtmpfs	3.9G	0	3.9G	0%	/dev
tmpfs	3.9G	0	3.9G	0%	/dev/shm
tmpfs	3.9G	8.6M	3.9G	1%	/run
tmpfs	3.9G	0	3.9G	0%	/sys/fs/cgroup
/dev/mapper/rhel-home	2.8G	33M	2.8G	2%	/home
/dev/sdal	1014M	143M	872M	15%	/boot
tmpfs	781M	0	781M	0%	/run/user/0
/dev/mapper/rdusecondary-LVBPRHOME	3.8G	33M	3.8G	2%	/bprHome
/dev/mapper/rdusecondary-LVBPRDATA	15G	33M	15G	1%	/bprData
/dev/mapper/rdusecondary-LVBPRDBLOG	5.8G	33M	5.8G	1%	/bprLog

- 이러한 명령을 사용하여 새 파티션 상태, 새로운 물리적 및 논리적 볼륨 상태, 파일 시스템 유형, 할당 블록을 등록 및 확인할 수 있습니다.

```
[root@pcprdusecondary ~]# fdisk -l
[root@pcprdusecondary ~]# pvdisplay
[root@pcprdusecondary ~]# vgdisplay
[root@pcprdusecondary ~]# lvdisplay
```

참고:

- 논리 볼륨의 fstab 항목을 추가할 필요가 없습니다. Corosync 클러스터는 볼륨 마운트를 처리합니다. 과거에는 이러한 항목 때문에 문제가 발생한 고객이 있었습니다. 시스템 재부팅 중에 타이밍 문제로 인해 기본 및 보조 모두 볼륨을 마운트하려고 시도합니다.
- 볼륨 그룹 이름과 논리 볼륨(LVBPRHOME, LVBPRDATA, LVBPRDBLOG)은 두 서버 모두에서 동일해야 합니다. 두 서버에서 동일한 디스크 공간을 공유해야 합니다.
- DRBD 블록 디바이스 파일 시스템 동기화는 두 서버에서 모두 디스크 크기만 동일하게 작동합니다.
- CentOS Linux 버전은 7.4여야 하며 커널은 3.10.0-693.11.6.el7.x86_64여야 합니다.
- 두 서버 모두 VIP가 광고되는 공용 IP 주소에 동일한 인터페이스를 사용하는지 확인합니다 (ens192).

2. 두 서버에서 RDU HA 구축을 위해 Linux 7.4 서버를 준비합니다

- [RDU HA 설치 모드](#)
- [RDU HA 노드 구성을 위한 일반적인 초기 단계](#)
- [기본-보조 모드에서 RDU HA 설정](#)
- [기본-보조 모드에서 HA 설정을 위한 RDU 노드 준비](#)

3. Geo-Redundancy 모드에서 RDU 서버 설치

- [RDU 2노드 장애 조치 쌍 설정](#)
- [기본 전용 및 보조 전용 모드에서 RDU HA 설정](#)
- [복구 모드를 사용하여 영향을 받는 RDU 노드 복구](#)

자세한 내용은 빠른 시작 설명서를 참조하십시오.

https://www.cisco.com/c/en/us/td/docs/net_mgmt/prime/cable_provisioning/6-1-3/quick/start/guide/CiscoPrimeCableProvisioning-6_1_3-QuickStartGuide/CiscoPrimeCableProvisioning-6_1_3-QuickStartGuide_chapter_0101.html#task_1DBF800D2FF84D73BD972A0C6C7B92E6

4. Geo-Redundancy 구축을 위한 레이어-3 라우팅 전제 조건

RDU Geo-Redundancy

RDU Geo Redundancy는 RHEL 7.4 또는 CentOS 7.4(둘 다 64비트)에서 지원되는 RDU HA의 향

상된 기능입니다. 이 경우 RDU 기본 노드와 보조 노드가 서로 다른 지리적 위치에 있거나 두 노드가 모두 다른 서브넷에 있을 수 있습니다.

- Geo 이중화 모드에서는 VIP가 모든 서브넷에 있을 수 있습니다. 두 노드에 공통된 서브넷 범위에 포함될 필요는 없습니다.
- Geo 이중화 모드에서 VIP의 CIDR 값은 32여야 합니다.
- VIP는 활성 서버에서 RIP 광고로 알려지므로 두 노드의 인그레스 라우터에서 경로 삽입을 수행해야 합니다.
- Geo 이중화 모드에서는 리소스 에이전트(res_VIPrip)를 사용하여 VIP를 모니터링합니다.

PCP Geo-Redundancy 요건

기본 및 보조 서버가 연결된 인그레스 라우터에서 VIP(가상 IP)에 대한 경로 삽입을 수행해야 합니다.

VIP는 활성 서버에서 RIP2 알림으로 알려지므로 RIP2를 위해 사용자 환경에서 실행되는 동적 라우팅 프로토콜로 경로 재배포를 수행해야 합니다.

RIP2 경로를 OSPF IGRP에 재배포하고 광고하는 방법.동일한 재배포는 EIGRP/IBGP와 같은 다른 프로토콜에 사용할 수 있습니다.

PCP Geo 이중화 솔루션의 경우 VIP의 CIDR 값은 32여야 합니다.

- quagga를 통한 VIP 광고가 활성화된 경우 VIP를 알릴 인터페이스를 입력합니다. 기본적으로 eth0이며, 기본 서버와 보조 서버 모두에서 이 인터페이스 이름이 동일한지 확인하고, 경로 삽입을 수행하는 인그레스 라우터에 이 인터페이스가 연결되었는지 확인합니다.
- 과가를 통한 VIP 광고가 비활성화된 경우 VIP에 대한 CIDR 값을 입력합니다.
- /etc/quagga/ripd.conf- Geo 모드에서 RIP2 conf가 추가된 경로 <https://www.nongnu.org/quagga/docs/quagga.html#RIP>
- RIP 인접성은 기본 및 보조 서버 모두에 연결된 인접 라우터에 삽입해야 합니다.다음과 같은 샘플 컨피그레이션:

```
Router#show run | sec rip
router rip
version 2
network 10.0.0.0
no auto-summary
Router#_
```

- 네이버 피어에 대한 인접성 컨피그레이션입니다.두 라우터에서 모두 구현해야 합니다.인터페이스 광고를 위해 VIP 및 공용 IP 네트워크를 추가해야 합니다.
- VIP 주소로 라우팅합니다.
- 외부 세계에 알리기 위해 활성화된 경로에 따라 ospf/eigrp/static을 통해 이 RIP 네트워크를 알립니다.

Example: Here OSPF is the dynamic protocol
router ospf <processed>

redistribute rip metric-type 1 subnets. For RIP2, it uses metric as hop count.

Example: Here ISIS is the dynamic protocol

```
router isis
```

```
redistribute rip metric
```

사후 확인 HA

- 다음 명령을 사용하여 RDU HA 클러스터 상태를 **확인합니다**.
/bprHome/CSCObac/agent/HA/bin/monitor_ha_cluster.sh
- Geo-Redundancy 모드의 문제 없이 RDU HA가 작동하는지 확인합니다. 기본 및 보조 DRBD 디스크가 동기화되고 최신 상태(cat /proc/drbd)가 표시될 때까지 기다립니다.