

NetWitness[®] Platform XDR

Version 12.0

Storage Guide

Contact Information

NetWitness Community at <https://community.netwitness.com> contains a knowledge base that answers common questions and provides solutions to known problems, product documentation, community discussions, and case management.

Trademarks

RSA and other trademarks are trademarks of RSA Security LLC or its affiliates ("RSA"). For a list of RSA trademarks, go to <https://www.rsa.com/en-us/company/rsa-trademarks>. Other trademarks are trademarks of their respective owners.

License Agreement

This software and the associated documentation are proprietary and confidential to RSA Security LLC or its affiliates and are furnished under license, and may be used and copied only in accordance with the terms of such license and with the inclusion of the copyright notice below. This software and the documentation, and any copies thereof, may not be provided or otherwise made available to any other person.

No title to or ownership of the software or documentation or any intellectual property rights thereto is hereby transferred. Any unauthorized use or reproduction of this software and the documentation may be subject to civil and/or criminal liability.

This software is subject to change without notice and should not be construed as a commitment by RSA.

Third-Party Licenses

This product may include software developed by parties other than RSA. The text of the license agreements applicable to third-party software in this product may be viewed on the product documentation page on NetWitness Community. By using this product, a user of this product agrees to be fully bound by terms of the license agreements.

Note on Encryption Technologies

This product may contain encryption technology. Many countries prohibit or restrict the use, import, or export of encryption technologies, and current use, import, and export regulations should be followed when using, importing or exporting this product.

Distribution

Use, copying, and distribution of any RSA Security LLC or its affiliates ("RSA") software described in this publication requires an applicable software license.

RSA believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED "AS IS." RSA MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

© 2020 RSA Security LLC or its affiliates. All Rights Reserved.

July, 2022

Contents

| | |
|---|-----------|
| Storage Overview | 6 |
| Storage Requirements | 7 |
| Drive Specifications | 7 |
| Required NetWitness Platform Storage Volumes | 7 |
| Performance Recommendations | 9 |
| Input/Output Operations Per Second | 9 |
| General Description of How NetWitness Platform Hosts Store Data | 9 |
| Prepare Physical Storage | 10 |
| Configure Block devices for Drive pack | 10 |
| Benefits of Series 6/6E Drive Pack | 10 |
| Decoder Meta Use Cases | 10 |
| Concentrator Index Use Cases | 11 |
| Configure Block Devices for PowerVaults | 11 |
| Configure Block Device for Decoder / Log Decoder | 12 |
| Configure Block Device for Concentrator | 14 |
| Configure Storage | 14 |
| Configure Storage for Decoder / LogDecoder | 14 |
| Configure Storage for Concentrator | 15 |
| Enable Security on SED Capable Drives | 15 |
| Prepare Virtual or Cloud Storage | 16 |
| Decoder, Log Decoder, Concentrator, Archiver | 16 |
| NW Server, ESA Primary, ESA Secondary and Malware Analysis | 16 |
| Log Collector | 17 |
| Endpoint Log Hybrid | 17 |
| Additional Endpoint Log Hybrid Partitions | 21 |
| UEBA | 22 |
| Configure Storage Using the REST API | 23 |
| REST API Storage Configuration Commands | 23 |
| Storage Configuration Tasks | 24 |
| Task 1 - Attach Storage to the Host and Access the REST API Storage Commands | 24 |
| Task 2 - (Conditional) RAID Configuration for PowerVault and DACs | 26 |
| Task 3 - Allocate Block Devices to Partitions, Volume Groups, and Logical Volumes | 29 |
| Task 4 - Allocate Volume Groups to NetWitness Services - srvAlloc | 31 |
| Task 5 - (Optional) Reconfigure Storage Configuration for 10G Capture | 31 |

| | |
|--|------------|
| Prepare Unity Storage | 34 |
| Task 1 - Access Unisphere User Interface (UI) | 35 |
| Task 2 - Create Pools | 36 |
| Task 3 - Create LUNS | 39 |
| Task 4 - Register Hosts | 41 |
| Task 5 - Assign LUNS to Hosts | 43 |
| Task 6 - Install PowerPath | 45 |
| Migrate Data to Another Storage Type | 47 |
| Migrate Data Using the Warm and Hot Tier Option | 47 |
| Stop the Service | 47 |
| Set Up PowerVault | 47 |
| Configure The Mount Points | 48 |
| Set up Warm and Hot Tiers | 49 |
| Decommission the DAC | 51 |
| Move Data From DAC to PowerVault | 52 |
| Data on PowerVault After Move from DAC | 55 |
| Appendix A. How NetWitness Platform Hosts Store Data | 56 |
| Decoder Hosts | 56 |
| Concentrator Host | 56 |
| Archiver Host | 57 |
| Hybrid Hosts | 57 |
| Options for SAN Configurations | 57 |
| Performance Recommendations | 57 |
| Enable Security on SED Capable Drive groups on Host with a mix of SED and NON SED Drives | 57 |
| Appendix B. Encrypt a Series 6E Core or Hybrid Host (encryptSedVd.py) | 61 |
| Enable SED on configured Drive Groups | 63 |
| Enable Virtual Drives / Drive Groups - PERC H740 (Mini) Adaptors (Internal storage) | 66 |
| Enable SED on configured Virtual Drives/ Drive Groups on Power Vault (PERC 840) | 68 |
| Enable Virtual Drives / Drive Groups - PERC H840 Adaptors | 68 |
| Appendix C. Troubleshooting | 76 |
| Reconfigure Pre-Configured DAC Attached to Decoder Using REST API | 76 |
| Appendix D. Sample Storage Configuration Scenarios for 15-Drive DACs | 77 |
| Configure Storage for Archiver | 77 |
| Configure Storage for Network (Packet) Decoder | 80 |
| Configure Storage for Network Concentrator | 90 |
| Configure Storage for Log Decoder Hybrid | 96 |
| Appendix E. Sample Storage Configuration Scenarios for 8 or 12-Drive PowerVault | 101 |
| Configure Storage for Archiver using NW-PV-A/NW-PV-A-N | 101 |

| | |
|---|------------|
| Configure Storage for Decoder using NW-PV-B/NW-PV-B-N | 103 |
| Configure Storage for Concentrator using NW-PV-C/NW-PV-C-N | 109 |
| Configure Storage for Concentrator using NW-PV-D/NW-PV-D-N | 112 |
| Appendix F. Sample Storage Configuration Scenarios for Meta Disk Kits | 116 |
| Meta-Only (One kit – 3 SED drives configured as RAID 5) | 116 |
| Maximize PowerVault Storage Capacity (One Kit – 3 SED Drives Configured as RAID 5) | 121 |
| Appendix G. Sample Storage Configuration for Concentrator Index with One Meta Disk Kit | 130 |
| Revision History | 137 |

Storage Overview

This guide provides you with storage requirements and the instructions on how to allocate storage for physical (DACs, PowerVaults, Unity) and virtual storage devices for NetWitness Platform. It also includes the following topics.

- Detect Encryption on Existing PowerVault
- Migrate Data to Another Device

Refer to the following Hardware Setup Guides for information on how to connect these device to NetWitness Platform Core and Hybrid physical hosts:

- PowerVault (MD 1400) Setup Guide (see the "Enclosure Options" section of "Hardware Description")
- [NetWitness Community](#).
- 60-Drive DAC Setup Guide - [NetWitness Community](#).
- 15-Drive DAC Setup Guide - [NetWitness Community](#).

Storage Requirements

This section contains all the storage requirements needed to successfully attach storage to your NetWitness Platform deployment host systems. It contains the required drive types, appropriate volumes, and performance IOPS that are needed.

Drive Specifications

General specifications for core NetWitness Platform Hosts are:

- IO size 490/Dec
- Response/Latency < 20ms
- Decoder 10/90 read/write (low random I/O)
- Concentrator 50/50 read/write (high random I/O)

| RAID Group | Suitable Volumes |
|-------------------|---|
| NL-SAS or 10K SAS | All Packet Decoder volumes All Log Decoder volumes All Archiver volumes Concentrator meta volume |
| SSD | Concentrator index volume |

Required NetWitness Platform Storage Volumes

Service Volume Names

| Service | Volume Name | File Systems Created |
|-----------------|-----------------|---|
| Network Decoder | decoder | packetdb |
| Network Decoder | decodersmall | decoder root, index, sessiondb, metadb |
| Log Decoder | logdecoder | packetdb |
| Log Decoder | logdecodersmall | logdecoder root, index, sessiondb, metadb |
| Concentrator | concentrator | concentrator root, metadb, sessiondb |
| Concentrator | index | index |
| Archiver | archiver | database |

Volume Sizing

The volume sizes below are automatically created when using the NetWitness Platform storage tool, described in [Configure Storage Using the REST API](#).

| Volume | Filesystem | Mount Point | Size |
|-----------------|------------|--|--|
| decodersmall | decoroot | /var/netwitness/decoder | 10 GB |
| decodersmall | index | /var/netwitness/decoder/index | 30 GB |
| decodersmall | sessiondb | /var/netwitness/decoder/sessiondb | 600 GB |
| decodersmall | metadb | /var/netwitness/decoder/metadb | 100% of free space on decodersmall volume |
| decoder | packetdb | /var/netwitness/decoder/packetdb | 100% of free space on decoder volume |
| logdecodersmall | decoroot | /var/netwitness/logdecoder | 10 GB |
| logdecodersmall | index | /var/netwitness/logdecoder/index | 30 GB |
| logdecodersmall | sessiondb | /var/netwitness/logdecoder/sessiondb | 600 GB |
| logdecodersmall | metadb | /var/netwitness/logdecoder/metadb | 100% of free space on logdecodersmall volume |
| logdecoder | packetdb | /var/netwitness/logdecoder/packetdb | 100% of free space on logdecoder volume |
| concentrator | root | /var/netwitness/concentrator | 30 GB |
| concentrator | sessiondb | /var/netwitness/concentrator/sessiondb | 10% of free space on concentrator volume |
| concentrator | metadb | /var/netwitness/concentrator/metadb | 100% of free space on concentrator volume |
| index | index | /var/netwitness/concentrator/index | 100% of free space on index volume |
| archiver | database | /var/netwitness/archiver/database | 100% of free space on archiver volume |

Performance Recommendations

NetWitness recommends that Packet and Log Decoders receive two LUNs or Block Devices, one for Packet data, the other for all other databases. This allows you to segregate the high-bandwidth Packet Database from the other databases so they do not compete for I/O bandwidth with other activity.

Concentrators require a separate SSD-based index volume for best performance. You must house this index volume on a different RAID group than the Concentrator Meta database volume, which you can store on NL-SAS. Archivers can use a single large NL-SAS storage volume per appliance.

Input/Output Operations Per Second

The following table lists the IOPS requirements for the Decoder and Concentrator hosts.

| Logs | Log Decoder | Concentrator |
|---------|-------------|--------------|
| 10K EPS | 400 | 8,000 |
| 20K EPS | 550 | 10,300 |
| 25K EPS | 1,200 | 10,800 |

| Packets | Network Decoder | Concentrator |
|---------|-----------------|--------------|
| 1Gbps | 600 | 6,050 |
| 2 Gbps | 950 | 8,300 |
| 4 Gbps | 1,650 | 12,800 |
| 6 Gbps | 2,400 | 17,300 |
| 8 Gbps | 3,200 | 21,800 |

General Description of How NetWitness Platform Hosts Store Data

For information about how NetWitness Platform hosts store data, see [Appendix A. How NetWitness Platform Hosts Store Data](#).

Prepare Physical Storage

IMPORTANT: NetWitness recommends you to create a block device for NetWitness Storage.

This section provides two options to configure block device:

- [Configure Block devices for Drive pack](#)
- [Configure Block Devices for PowerVaults](#)

Note: Block device is also referred to as Virtual Drive or Drive Group

Configure Block devices for Drive pack

You can add additional drives to the Series 6 or 6E appliances to accommodate various use cases. These drives provide the capability for the decoder meta or concentrator index volumes to reside on the appliance. A minimum of 2 drives and a maximum of 6 drives are possible. The number of drives will depend on how much meta cache or index is needed.



Benefits of Series 6/6E Drive Pack

- **Maximize PowerVault Storage Capacity** - Traditionally, PowerVault storage allocates a volume for the Decoder metadata. This reduces the usable storage on the PowerVault. Drive Packs reduce this issue by providing 20TB of extra usable PV storage.
- **Reduces Cost for Meta Only Use Case** - For metadata-only deployments, drive pack fits for a customer who want to purchase hardware from NetWitness. This provides more cost-effective solution, because a drive pack can substitute a PowerVault.
- **Enable existing deployments to utilize compression options.**
- **Provides capability for expanding meta keys and associated indexing.**

Decoder Meta Use Cases

- Meta-Only
- Maximize Power Vault Storage

Two or more 2.4TB 10K SAS SED drives can be added to a Decoder for the decodersmall or logdecodersmall volumes. These volumes are used to store the meta cache on the Decoders.

Both the Log Decoders and Network Decoders parse out meta data from the raw captured traffic. The meta data is then aggregated to a Concentrator for indexing.

The host requires storage to store a cache for the meta extracted during the data capture for Concentrator aggregation. The meta cache on a Decoder is generally fixed in size, but you can expand it to support additional cache to avoid possible connectivity loss between the Decoder and the corresponding Concentrator.

Typically, the decodersmall or logdecodersmall volumes are stored on the first three drives of the first and second (10G config only) PowerVault enclosures. By utilizing the drive pack option, these three drives can instead be used for the packetdb (maximizing Power Vault storage).



For meta-only scenarios, the decodersmall volume would be stored on the drive pack, therefore eliminating the need for a Power Vault.

Concentrator Index Use Cases

- Support Additional Meta-Key Indexing
- Capability to Enable compression for Existing Deployments

Two or more 3.84 TB SSD SED drives can be added to a Concentrator to increase the index volume. The index storage needs are scaled based on the NetWitness Platform deployment retention requirements. If additional meta keys are enabled and indexed, it may impact index retention.

For existing deployments, an SSD index drive pack is required if you need to enable compression. When compressing the packetdb and metadb, additional index is needed to support compression of those databases.

See [Appendix F. Sample Storage Configuration Scenarios for Meta Disk Kits](#) for sample storage configuration scenarios for Meta Disk Kits.

Configure Block Devices for PowerVaults

The Physical, Virtual, or Cloud NetWitness hosts for Decoders, Log Decoders, Concentrators, and Archivers need block storage attached. Make sure that the allocated storage meets all of the storage requirements. Specifically, make sure that the required storage volumes are created (for more information, see Required NetWitness Platform Storage Volumes in [Storage Requirements](#)), and:

- At least two block devices are created for Decoders (Meta, Session and Packet volumes)

Note: The larger block devices hold the packet volume, and the smaller block devices hold the meta and session.

- At least two block devices are created for Concentrators (Index and Meta volumes).
- Ensure that the block device meets the minimum IOPS for expected ingestion rates.

Configure Block Device for Decoder / Log Decoder

While creating the block device RAID configuration, the best practice is to configure a RAID 6 for the larger NL-SAS drives and RAID 5 or 1 for any 10k SAS or SSD type drives.

1. Identify the controller ID (Ctl) for 'PERC H840P Adaptor'.

```
/opt/MegaRAID/perccli/perccli64 show
```

In the below figure the controller ID is '1' corresponds to 'PERCH840PAdaptor'.

```
[root@116S6Core1 ~]# /opt/MegaRAID/perccli/perccli64 show
Status Code = 0
Status = Success
Description = None

Number of Controllers = 2
Host Name = 116S6Core1
Operating System = Linux 3.10.0-1160.21.1.el7.x86_64
StoreLib IT Version = 07.0400.0200.0400
StoreLib IR3 Version = 15.50-0

System Overview :
=====

-----
Ctl Model          Ports PDs DGs DNOpt VDs VN0pt BBU sPR DS EHS ASOs Hlth
-----
 0 PERCH740PMini    8  10  2   0   2   0 Opt On -  N   0 Opt
 1 PERCH840Adapter  8  12  0   0   0   0 Opt On -  N   0 Opt
-----

Ctl=Controller Index|DGs=Drive groups|VDs=Virtual drives|Fld=Failed
PDs=Physical drives|DNOpt=DG NotOptimal|VN0pt=VD NotOptimal|Opt=Optimal
Msng=Missing|Dgd=Degraded|NdAtn=Need Attention|Unkwn=Unknown
sPR=Scheduled Patrol Read|DS=DimmerSwitch|EHS=Emergency Hot Spare
Y=Yes|N=No|ASOs=Advanced Software Options|BBU=Battery backup unit
Hlth=Health|Safe=Safe-mode boot
```

2. Identify the Enclosure ID (EID) for controller '1'. In this case the EID is '247'.

```
/opt/MegaRAID/perccli/perccli64 /c1 /eall show
```

```
[root@116S6Core1 ~]# /opt/MegaRAID/perccli/perccli64 /c1 /eall show
Controller = 1
Status = Success
Description = None

Properties :
=====

-----
EID State Slots PD PS Fans TSs Alms SIM Port# ProdID VendorSpecific
-----
247 OK          12 12  2   4   7   0   2 01 x4 MD1400
-----

EID-Enclosure Device ID |PD-Physical drive count |PS-Power Supply count|
TSs-Temperature sensor count |Alms-Alarm count |SIM-SIM Count
```


- Identify the existing block devices on the host. The block device name is identified under **NAME** column. The block devices names shown below are **sda,sdb** and **sdc**. Use **'lsblk'** to list the block devices.

```
[root@116S6Core1 ~]# lsblk
NAME                                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                                  8:0    0  1.1T  0 disk
├─sda1                               8:1    0    1G  0 part /boot
├─sda2                               8:2    0  1.1T  0 part
│   ├─netwitness_vg00-root          253:0    0   30G  0 lvm /
│   ├─netwitness_vg00-swap         253:1    0    4G  0 lvm [SWAP]
│   ├─netwitness_vg00-nwhome       253:2    0  3.2T  0 lvm /var/netwitness
│   ├─netwitness_vg00-varlog       253:3    0   10G  0 lvm /var/log
│   └─netwitness_vg00-usrhome      253:4    0   10G  0 lvm /home
sdb                                  8:16   0  2.2T  0 disk
├─sdb1                               8:17   0  2.2T  0 part
│   └─netwitness_vg00-nwhome       253:2    0  3.2T  0 lvm /var/netwitness
sdc                                  8:32   0  8.7T  0 disk
[root@116S6Core1 ~]#
```

- Create the Virtual Drive or Drive Group (**DG**) on PERCH840PAdaptor using disks in slot 0 through 9 (for example, all the drives) using below command.

Warning: Every decoder needs a logdecodersmall or decodersmall volume for meta. This example assumes the meta volume already exists on another PowerVault or Drive Pack. If this enclosure will account for the meta volume, the first two or 3 drives would need to be allocated for the meta volume block device. Another block device would need to be created with the remaining drives for the packetdb volume.

```
/opt/MegaRAID/perccli/perccli64 /c1 add vd type=raid6 drives=247:0-11
strip=128 force
[root@116S6Core1 ~]# /opt/MegaRAID/perccli/perccli64 /c1 add vd type=raid6 drives=247:0-11 strip=128 force
Controller = 1
Status = Success
Description = Add VD Succeeded
[root@116S6Core1 ~]#
```

- To view the Virtual Drive created in the above step:

```
/opt/MegaRAID/perccli/perccli64 /c1 /vall show
[root@116S6Core1 ~]# /opt/MegaRAID/perccli/perccli64 /c1 /vall show
Controller = 1
Status = Success
Description = None

Virtual Drives :
=====
-----
DG/VD TYPE  State Access Consist Cache Cac sCC          Size Name
-----
0/0  RAID6 Optl RW      No      RWBD -   OFF 106.918 TB
-----

Cac=CacheCade|Rec=Recovery|OfLn=OffLine|Pdgd=Partially Degraded|Dgrd=Degraded
Optl=Optimal|R0=Read Only|RW=Read Write|HD=Hidden|TRANS=TransportReady|B=Blocked|
Consist=Consistent|R=Read Ahead Always|NR=No Read Ahead|WB=WriteBack|
FWB=Force WriteBack|WT=WriteThrough|C=Cached IO|D=Direct IO|sCC=Scheduled
Check Consistency
```

- Identify the new block device on the host. The block device name is identified under **NAME** column. The new block device corresponding to the above virtual drive is **sdd**. This block device name is required when configuring storage. Use **'lsblk'** to list the block devices.

```
lsblk
```

```
[root@116Decoder perccli]# lsblk
NAME                                MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
sda                                  8:0      0  931G  0 disk
├─sda1                               8:1      0    1G  0 part /boot
└─sda2                               8:2      0  930G  0 part
   ├─netwitness_vg00-root            253:0    0    30G  0 lvm  /
   ├─netwitness_vg00-swap            253:1    0     4G  0 lvm  [SWAP]
   ├─netwitness_vg00-nwhome          253:2    0   2.7T  0 lvm  /var/netwitness
   ├─netwitness_vg00-varlog          253:3    0    10G  0 lvm  /var/log
   └─netwitness_vg00-usrhome          253:4    0    10G  0 lvm  /home
sdb                                  8:16     0   1.8T  0 disk
├─sdb1                               8:17     0   1.8T  0 part
└─netwitness_vg00-nwhome            253:2    0   2.7T  0 lvm  /var/netwitness
sdc                                  8:32     0   8.7T  0 disk
sdd                                  8:48     0  106.9T 0 disk
[root@116Decoder perccli]#
```

- You must Configure Storage for Decoder / LogDecoder and Concentrator [Configure Storage](#) to complete the configuration.

Configure Block Device for Concentrator

Block Devices must be configured on PowerVaults before configuring the PowerVaults as storage to Concentrator. The Block Devices can be configured using the steps similar to [Configure Block Device for Decoder / Log Decoder](#) using percli64 utility. Use the SSD drives for index and the remaining drives for the Meta or Session DB.

Configure Storage

Configure Storage for Decoder / LogDecoder

Use REST API tool to configure the above block devices or virtual drives as Storage for Decoder / Log Decoder or Concentrator. For more information, see *Storage Configuration Tasks* (Task 3 and Task 4) for Decoder or Log Decoder and Task 1 to Task 5 for Concentrator in [Configure Storage Using the REST API](#) topic.

| Service | Controller | Volume | Block Device |
|-----------------------|------------------------|--------------|--|
| Decoder / Log Decoder | PERC H740 Mini Adaptor | decodersmall | Refer to step 7 in Prepare Physical Storage (Decoder / Log Decoder) in this example the block device is 'sdc' . |
| Decoder / Log Decoder | PERC H840 Adaptor | decoder | Refer to Step 6 in Configure Block Device for Decoder / Log Decoder in this example the block device is 'sdd' . |

Configure Storage for Concentrator

Use REST API tool to configure the block devices created on Drive Pack and/or Power Vaults. The block devices created on SSD is allocated to Index database and the one created on HDD to Meta/Session Database. See Storage Configuration Tasks (Task 3 and Task 4) for Concentrator in [Configure Storage Using the REST API](#) topic.

Enable Security on SED Capable Drives

To enable Security on the SED Capable Drive Group on PERC H740 Mini and PERC H840 Adaptors, see [Appendix B. Encrypt a Series 6E Core or Hybrid Host \(encryptSedVd.py\)](#) .

Prepare Virtual or Cloud Storage

This section describes how to set up virtual or cloud storage for the following types of component hosts:

- [Decoder, Log Decoder, Concentrator, Archiver](#)
- [NW Server, ESA Primary, ESA Secondary and Malware Analysis](#)
- [Log Collector](#)
- [Endpoint Log Hybrid](#)
- [Additional Endpoint Log Hybrid Partitions](#)
- [UEBA](#)

Decoder, Log Decoder, Concentrator, Archiver

Virtual or Cloud NetWitness hosts for Decoders, Log Decoders, Concentrators, and Archivers need block storage attached. Make sure that the allocated storage meets all of the storage requirements. Specifically, make sure that the required storage volumes are created (see "Required NetWitness Platform Storage Volumes" in [Storage Requirements](#)), and:

- At least two Block Devices are created for Decoders (meta /session and packet volumes)
- At least two block devices are created for Concentrators (index and meta volumes)
- Ensure that block devices can meet the minimum IOPS for expected ingestion rates

Attach the allocated storage to the NetWitness host by following the hosting platforms native procedure.

- VmWare – Vsphere Console (add disk to VM)
- Hyper-V – Manager Console (add disk to VM)
- Azure – Add Managed Disks to virtual instance
- AWS – Add EBS Storage to virtual instance
- Google Cloud Platform (GCP) - Add storage to virtual instance

Once the disk is attached to the virtual host, configure block devices for services. For more information, see [Virtual Hosts Additional Storage Playbook](#).

NW Server, ESA Primary, ESA Secondary and Malware Analysis

For an extension of `/var/netwitness/` partition, attach an external volume.

Run `lsblk` to get the physical volume name and run the following commands:

1. `pvcreate <pv_name>` (for example, `pv_name` is `/dev/sdc`)
2. `vgextend netwitness_vg00 /dev/sdc`

3. `lvresize --resizefs --extents +100%FREE /dev/netwitness_vg00/nwhome`
4. `xfs_growfs /dev/netwitness_vg00/nwhome`

NetWitness recommends the following partition definitions. However, you can change these values based on the retention days.

| LVM | Folder | Block Storage |
|--|-------------------------------|---|
| <code>/dev/netwitness_vg00/nwhome</code> | <code>/var/netwitness/</code> | Refer to the Cloud Provider Block Storage setup (storage) tables. |

Log Collector

For an extension of `/var/netwitness/` partition, attach an external volume

Run `lsblk` to get the physical volume name and run the following commands:

1. `pvcreate <pv_name>` (for example, `pv_name` is `dev/sdc`)
2. `vgextend netwitness_vg00 /dev/sdc`
3. `lvresize --resizefs --extents +100%FREE /dev/netwitness_vg00/nwhome`
4. `xfs_growfs /dev/netwitness_vg00/nwhome`

NetWitness recommends the following partition definitions. However, you can change these values based on the retention days.

| LVM | Folder | Block Storage |
|--|-------------------------------|---|
| <code>/dev/netwitness_vg00/nwhome</code> | <code>/var/netwitness/</code> | Refer to the Cloud Provider Block Storage setup (storage) tables. |

Endpoint Log Hybrid

The total disk size required depends on the data retention period. You can use the below per day disk usage indicative values to calculate the required disk size for your deployment. For example, to retain 30 days of data, multiply the below per day disk usage values with 30.

The following table provides disk usage for one full scan. The full scan disk usage values are based on the below event count:

- Files count -1100
- Processes count -100
- DLLs count - 500
- Drivers count -150
- Services count - 500
- Tasks count -100

Endpoint Log Hybrid(50K Advance Agents - Disk usage per full scan)

| | MetaDB | PacketDB | SessionDB | Index | Total |
|--------------|--------|----------|-----------|-------|---|
| Log Decoder | 220 GB | 12 GB | 5 GB | NA | 237 GB |
| Concentrator | 230 GB | NA | 5 GB | 6 GB | 241 GB |
| MongoDB | NA | NA | NA | NA | 35 GB (First full scan) 30 GB (Subsequent per scan increase) |

The following tables provide per day disk usage for tracking data. The total tracking events per agent per day is 29000.

Endpoint Log Hybrid (50K Advance Agents - Tracking data without Expanded Network Visibility)

| | MetaDB | PacketDB | SessionDB | Index | Total |
|--------------|---------|----------|-----------|-------|--|
| Log Decoder | 1500 GB | 140 GB | 46 GB | NA | 1,686 GB |
| Concentrator | 1600 GB | NA | 46 GB | 30 GB | 1,676 GB |
| MongoDB | NA | NA | NA | NA | 35 GB (First full scan) 1.5 GB (Tracking data per day increase) |

The following tables provide per day disk usage for tracking data. Total tracking events per agent per day is 33000

Endpoint Log Hybrid (50K Advance Agents - Tracking data with Expanded Network Visibility)

| | MetaDB | PacketDB | SessionDB | Index | Total |
|--------------|---------|----------|-----------|-------|--|
| Log Decoder | 1800 GB | 152 GB | 55 GB | NA | 2007 GB |
| Concentrator | 1900 GB | NA | 55 GB | 36 GB | 1991 GB |
| MongoDB | NA | NA | NA | NA | 35 GB (First full scan) 1.5 GB (Tracking data per day increase) |

The following table provides per day disk usage for insight agents. The total tracking data per agent per day is 10800 plus 1 full scan daily.

Endpoint Log Hybrid (50K Insights Agents with Expanded Network Visibility)

| | MetaDB | PacketDB | SessionDB | Index | Total |
|-------------|--------|----------|-----------|-------|--------|
| Log Decoder | 500 GB | 52 GB | 18 GB | NA | 570 GB |

Endpoint Log Hybrid (50K Insights Agents with Expanded Network Visibility)

| | | | | | |
|--------------|--------|----|-------|-------|---|
| Concentrator | 600 GB | NA | 18 GB | 13 GB | 631 GB |
| MongoDB | NA | NA | NA | NA | 35 GB (First full scan) 30 GB (Subsequent per scan increase) |

The following table provides Endpoint Agents sizing based on the feature.

| Feature | Description | Agent or Endpoint Server |
|---------------------------|---|---|
| Endpoint Only | Only scan and tracking data | Maximum 50K Endpoint Agents only |
| Windows Logs Only | Only Windows Logs from agents. Assuming 20K events per second supported by Hybrid. | Maximum 20K Agents: <ul style="list-style-type: none"> Generates 20K log events per second |
| File Collection Only | Only File Collection from agents. Assuming 20K events per second supported by Hybrid | Maximum 20K Agents : <ul style="list-style-type: none"> Generates 20K log events per second |
| Endpoint and Windows Logs | Event per second per agent <ul style="list-style-type: none"> (For Windows Logs) 1 event sent by 1 agent every second (For Tracking Events) 0.4 event sent by 1 agent every second 20K events per second supported by Hybrid <div style="border: 1px solid green; padding: 5px; margin-top: 10px;"> <p>Note: Total agents should be calculated as below: Hybrid events per second/ (Windows Logs Endpoint Server of 1 agent + Tracking Event Endpoint Server for 1 agent) For example, 20000 / (1.0 + 0.4)</p> </div> | Maximum 15K (approximately) Agents: <ul style="list-style-type: none"> Generates 15K (approximately) Windows log events Plus <ul style="list-style-type: none"> Generates 15K (approximately) Agents EDR data |

| Feature | Description | Agent or Endpoint Server |
|--|---|---|
| Endpoint, Windows Logs and File Collection | Event per second per agent: <ul style="list-style-type: none"> • (For Windows Logs) 1 event sent by 1 agent every second • (For Tracking Events) 0.4 event sent by 1 agent every second • (For File Collection) 1 event sent by 1 agent every second • 20,000 events per second supported by Hybrid | Maximum 10K (approximately) Agents: <ul style="list-style-type: none"> • Generates 10K (approximately) Windows log events Plus <ul style="list-style-type: none"> • Generates 10K (approximately) Endpoint Agents data Plus <ul style="list-style-type: none"> • Generates 10K (approximately) Agents File Collection data |
| <div style="border: 1px solid green; padding: 5px;"> <p>Note: Total agents should be calculated as below: Hybrid events per second/ (Windows Logs Endpoint Server of 1 agent + Tracking Event Endpoint Server for 1 agent + File Collection) For example, 20000 / (1.0 + 1.0 + 0.4)</p> </div> | | |

Extending File Systems

For Endpoint Server, attach external disk for extension of `/var/netwitness/` partition, create an external disk with suffix as `nwhome`.

Follow these steps:

1. Ensure you have added a new disk. For more information, see "Task 1. Add New Disk" in the *Virtual Hosts Installation Guide for NetWitness Platform*. Go to the [NetWitness All Versions Documents](#) page and find NetWitness Platform guides to troubleshoot issues.
2. Execute `lsblk` and get the physical volume name
3. `pvcreate <pv_name>` suppose the PV name is `/dev/sdc`
4. `vgextend netwitness_vg00 /dev/sdc`
5. `lvresize --resizefs --extents +100%FREE /dev/netwitness_vg00/nwhome`
6. `xfs_growfs /dev/mapper/netwitness_vg00-nwhome`

NetWitness recommended partition for Endpoint Server (can be changed based on the retention days).

| LVM | Folder | Size | Disk Type |
|--|-------------------------------|------|-----------|
| <code>/dev/netwitness_vg00/nwhome</code> | <code>/var/netwitness/</code> | 6TB | HDD |

For Mongo DB, attach external disk for extension of `/var/netwitness/mongo` partition, create an external disk with suffix as `nwhome`.

Follow these steps:

1. Ensure you have added a new disk. For more information, see "Task 1. Add New Disk" in the *Virtual Hosts Installation Guide*. Go to the [NetWitness All Versions Documents](#) page and find NetWitness Platform guides to troubleshoot issues.
2. Execute `lsblk` and get the physical volume name
3. `pvcreate <pv_name>` suppose the PV name is `/dev/sdc1`
4. `vgextend hybrid /dev/sdc1`
5. `lvresize --resizefs --extents +100%FREE /dev/hybrid-vmng`
6. `xfs_growfs /dev/mapper/hybrid-vmng`

NetWitness recommended partition for Mongo DB (Can be changed based on the retention days). Minimum recommended size for `var/netwitness` is 500 GB.

| LVM | Folder | Size | Disk Type |
|-------------------------------|------------------------------------|------|-----------|
| <code>/dev/hybrid-vmng</code> | <code>/var/netwitness/mongo</code> | 6TB | HDD |

Additional Endpoint Log Hybrid Partitions

The following partition should be on the volume group endpoint and should be in a single RAID 0 array.

| Folder | LVM | Volume Group |
|---|-----------------------|--------------|
| <code>/var/netwitness/mongo</code> | hybrid-mongo | endpoint |
| <code>/var/netwitness/concentrator</code> | concentrator-concroot | endpoint |
| <code>/var/netwitness/concentrator/index</code> | hybrid-concindex | endpoint |
| <code>/var/netwitness/logdecoder</code> | hybrid-ldecroot | endpoint |

Run `lsblk` to get the physical volume name and run the following commands:

1. `pvcreate /dev/md0`
2. `vgcreate -s 32 endpoint /dev/md0`
3. `lvcreate -L <disk_size> -n <lv_name> endpoint`
4. `mkfs.xfs /dev/ endpoint /<lv_name>`
5. Repeat the above steps for all the LVMs mentioned.

NetWitness recommends the following partitions. However, you can change these values based on the retention days.

| LVM | Folder | Block Storage |
|--|-------------------------------|---|
| <code>/dev/netwitness_vg00/nwhome</code> | <code>/var/netwitness/</code> | Refer to the Cloud Provider Block Storage setup (storage) tables. |

| LVM | Folder | Block Storage |
|-----------------------------------|------------------------------------|---|
| /dev/endpoint/hybridmongo | /var/netwitness/mongo | Refer to the Cloud Provider Block Storage setup (storage) tables. |
| /dev/endpoint/concentratorconroot | /var/netwitness/concentrator | Refer to the Cloud Provider Block Storage setup (storage) tables. |
| /dev/endpoint/hybridconcinde | /var/netwitness/concentrator/index | Refer to the Cloud Provider Block Storage setup (storage) tables. |
| /dev/endpoint/hybridldeecroot | /var/netwitness/logdecoder | Refer to the Cloud Provider Block Storage setup (storage) tables. |

UEBA

The following procedure attaches an external disk and extends the `/var/netwitness/` partition. You must use `nwhome` as the external disk suffix. This procedure illustrates how to add a 2TB disk.

Note: `/var/netwitness` is the only partition that can reside on this volume.

- List the physical volume name.
`lsblk (for example, dev/mapper/sdc)`
- Extend the `/var/netwitness/` partition.

```
pvcreate <pv_name>where pv_name is dev/mapper/sdc
vgextend netwitness_vg00 /dev/mapper/sdc
lvresize --resizefs --extents +100%FREE /dev/mapper/netwitness_vg00/nwhome
xfs_growfs /dev/mapper/netwitness_vg00-nwhome
```

This partition is the NetWitness recommended partition for UEBA. You can change it based on retention days.

Configure Storage Using the REST API

In NetWitness Platform 11.3 and later releases, you use the REST API for all storage configuration operations. For information about how to use the REST API, see the *RESTful API User Guide*. Go to the [NetWitness All Versions Documents](#) page and find NetWitness Platform guides to troubleshoot issues.

REST API Storage Configuration Commands

Each of the commands listed below has built-in help that describes their function and usage. If you are using the REST interface, select the command from the drop-down menu to see the help text. For examples of REST API storage configuration commands, see [Appendix D. Sample Storage Configuration Scenarios for 15-Drive DACs](#).

Commands for Direct-Attached RAID Volumes

- `raidList` : List the RAID controllers and direct-attach enclosures that are present on this host.
- `raidNew` : Allocate direct-attached enclosures to block devices.

Commands for Allocating Block Devices as Storage

- `devlist` : List available block devices on the host.
- `partNew` : Allocate partitions on a block device and create volume groups.
- `vgs` : Summarize how block devices are organized into volume groups.

Commands for Allocating Storage to Services

- `srvList` :List services on the host and their allocated storage paths.
- `srvAlloc` : Allocate a volume group to a service.
- `srvFree` : Remove a volume group from a service.
- `multipath-II` : To verify if SAN devices are attached.

Command to Reconfigure Services to Detect and Use All of the New Storage

- `reconfig` - After configuring new storage, detect and use new storage on the associated service and database.

Storage Configuration Tasks

Task 1 - Attach storage to the host and access the REST API storage configuration commands.

Task 2 - (Conditional) Configure RAID if necessary.

Task 3 - Allocate block devices to partitions, volume groups, and logical volumes.

Task 4 - Allocate volume groups to NetWitness services.

Task 5 - Reconfigure services and databases to detect and appropriately use new storage.

Task 1 - Attach Storage to the Host and Access the REST API Storage Commands

IMPORTANT: Task 1 is not applicable for NetWitness version 11.5.0.0 and 11.5.0.1.

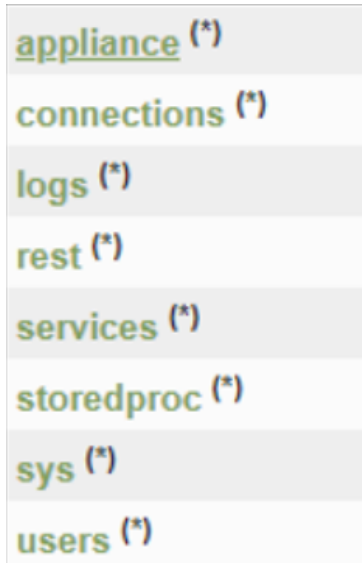
Complete the following steps to attach an external storage device to a host and access the storage configuration commands available through the REST API.

1. Attach the storage and make it available to this host.
 - To attach PV storage, refer to the *PowerVault (Dell MD 1400) Setup Guide*.
 - For third-party storage, create the RAID groups to match the volumes listed in [Storage Requirements](#)
2. There are two ways that you can access the REST API storage commands: from a Browser, or from the **Services > Explore** view from the User Interface.

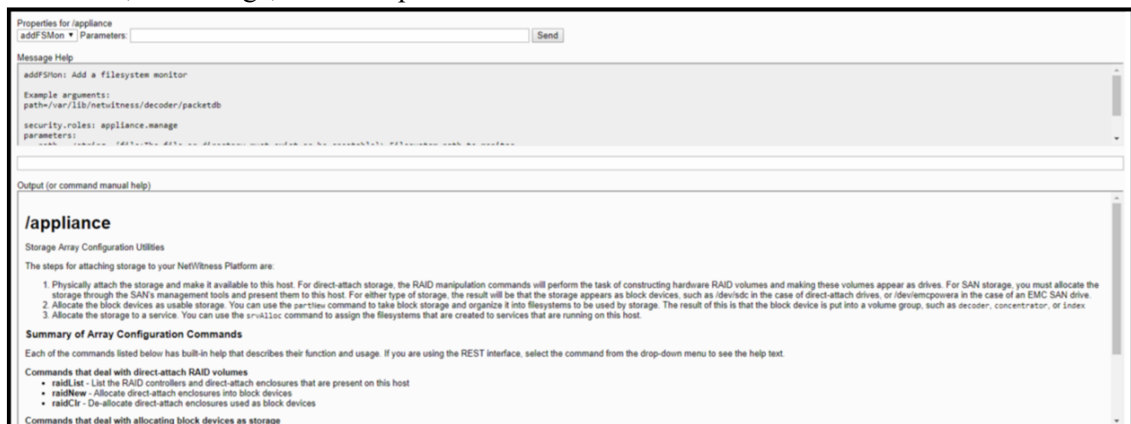
Note: Once you have accessed the REST API, the steps that you perform are the same, no matter which method you used to access it.



- From a Browser.
 - a. Open a Browser and specify the ip-address of the host with port **50106**.
 The following example is the Decoder, but you need to use port 50106 for any host hardware for which you are configuring storage using the REST API.
 https://<decoder-ip-address>:50106

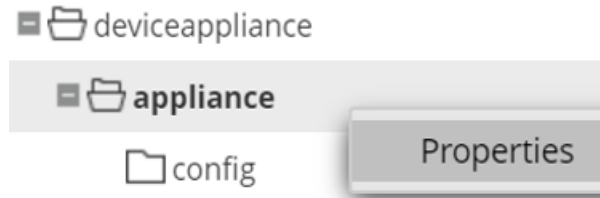
- b. Log in with the admin account credentials.
 The following REST API menu is displayed.



- c. Click on the (*) next to **appliance** to access the REST command set.
 The **Properties for /appliance** dialog is displayed under the initial REST menu. The **Output (or command manual help)** section describes the commands that the REST API can send to the device, their usage, and their parameters.



- From the User Interface.
 - a. In the **NetWitness** menu, go to  (Admin) > **SERVICES**.
 - b. Select the service (for example, a Concentrator).
 - c. Under  (actions), select **View > Explore**.
 - d. Navigate to **deviceappliance/appliance**, right click, and click **Properties**.



Note: If you are on NetWitness version 11.5.0.0 or 11.5.0.1, you must navigate to **System > Host Tasks > Task**.

You can now access the storage commands from the **Properties** dialog.

3. Proceed to:
 - [Task 2](#) if you need to configure RAID for PowerVault or DACs.
 - [Task 3](#) if you do not need to configure RAID and already have a block device available.

Task 2 - (Conditional) RAID Configuration for PowerVault and DACs

IMPORTANT: Task 2 is mandatory if you are on NetWitness version 11.5.0.0 or 11.5.0.1.

NetWitness Platform hardware uses direct-attached SAS drives for storage. These drives are housed in a SAS enclosure. SAS enclosures are shelves of drives attached to the NetWitness node by a cable connected to the SAS host bus adapter.

SAS enclosures are also known as other names, such as "DAC" (Direct-Attached Capacity), or "JBOD" (Jumbo Box of Disks), or "Dell PowerVault".

NetWitness Platform utilizes Dell PERC SAS host bus adapters. NetWitness Platform devices typically include two SAS host bus adapters. One is used for controller drives that are internal to the NetWitness Node, and another is used for controlling drives attached to the SAS enclosures. The internal controller and drives are configured when the node is built, but the external SAS enclosures are not. You execute the `raidList` and `raidNew` commands to identify and configure the external SAS enclosures.

These commands work with the following SAS enclosure types:

- EMC ESAS 15-drive enclosures
- EMC ESAS 60-drive enclosures
- Dell PowerVault 12-drive enclosures
- Dell PowerVault 8-drive enclosures

Note: EMC 60-drive enclosures are logically organized as four separate 15-drive sub-enclosures. They behave as if there are four 15-drive enclosures, each of which can be configured independently.

The `raidList` and `raidNew` commands operate on entire enclosures. Execute `raidList` to identify the enclosures. execute `raidNew` to configure an enclosure to perform one of the pre-determined roles within a NetWitness Platform node.

After you attach storage to the host and access the REST API storage commands, complete the following steps to create RAID if required.

1. Execute the `raidList` command to identify the controllers and enclosures that are attached to the system.

In the following example, Controller 1 does not display any block devices. This indicates the array is not configured.

```

Properties for /appliance
raidList Parameters:  Send

Message Help
raidList: list drive shelves attached to this appliance
security.roles: appliance.manage

/appliance?msg=raidList&force-content-type=text/plain

Output (or command manual help)
Controller 0, Enclosure 32
Vendor: DP
Model: BP13G+EXP
In Use: true
Drives: 931.511 GB x 2
        1.818 TB x 2
Devices: sda
         sdb

Controller 1, Enclosure 82
Vendor: DELL
Model: MD1400
In Use: false
Drives: 10.691 TB x 12
Devices:

Controller 1, Enclosure 13
Vendor: DELL
Model: MD1400
In Use: false
Drives: 10.691 TB x 12
Devices:

```

2. Select a RAID layout scheme for the Enclosure.

The following table lists the PV to Supported Hosts Mapping.

| Type | SKU | Specification | Supported Hosts |
|------------------|---------|---|-----------------|
| High Performance | NW-PV-C | Dell Storage MD 1400 6 x 12TB NL-SAS SED, 2 x 3.8TB SSD SED | Concentrator |

| Type | SKU | Specification | Supported Hosts |
|------------------|---------|---|--|
| High Performance | NW-PV-D | Dell Storage MD 1400 9 x 12TB NL-SAS SED, 3 x 3.8TB SSD SED | Concentrator |
| High Density | NW-PV-A | Dell Storage MD1400 12 x 12 TB NL-SAS SED | Decoder, LogDecoder, Archiver, hybrid (log & packet) |
| High Density | NW-PV-B | Dell Storage MD 1400 8 x 12TB NL-SAS SED | Decoder, LogDecoder, Archiver, hybrid (log & packet) |

The following tables show you the supported allocation schemes.

Note: For RAID configuration, when the Decoder is configured for 10G capture, use the `decoder` scheme for the **first two enclosures** and the `archiver` scheme for subsequent enclosures. When you are not configuring for 10G capture, use the `decoder` scheme for the **first enclosure** and the `archiver` scheme for subsequent enclosures. These configurations will maximize storage capacity and performance.

| Scheme | Drives Required | Allocation |
|-----------------------|--------------------------------|---|
| decoder or logdecoder | 8 or 12 or 15 HDDs | 3x drives in RAID 5 for decodersmall or logdecodersmall, all remaining drives in RAID 5 |
| archiver | 8 or 12 or 15 HDDs | All drives in RAID 6 for archiver or decoder database volume |
| networkhybrid | 8 or 12 or 15 HDDs | 3x drives in RAID 5 for meta expansion, all remaining drives in RAID 5 for packet expansion |
| loghybrid | 8 or 12 or 15 HDDs | Half of the drives in RAID 5 for meta expansion, half the drives in RAID 5 for packet expansion |
| concentrator | 2 or more SSDs, 4 or more HDDs | All SSDs in RAID 1 or RAID 5 for index, all HDDs in RAID 6 for meta |

- After the controller, enclosure, and scheme are identified, execute the `raidNew` command to create RAID Volumes. For example:

```
send /appliance raidNew controller=1 enclosure=82 scheme=decoder
preferSecure=false
```

 Add the `commit=1` parameter to actually execute this operation. Execute the `raidList` command to list the created block devices.
- (Optional) Configure SEDs (Self-Encrypting Drives). If the `raidNew` command detects self-encrypting drives and a security key has been set on the controller, the `raidNew` command will attempt to create a secure array. To set a security key on the controller, execute the `raidKey` command. For example:

```
send /appliance raidKey controller=1 key=myPassphrase keyId=1
```

 - To create a secured (that is, encrypted) array on physical devices attached to a controller with a security key set, specify `preferSecure=true` when using `raidNew`

- To create an unsecured (that is, unencrypted) array on physical devices attached to a controller with a security key set, specify `preferSecure=false` when using `raidNew`.
5. Go to [Task 3 - Allocate Block Devices to Partitions, Volume Groups, and Logical Volumes](#), after you create RAID volumes.

Task 3 - Allocate Block Devices to Partitions, Volume Groups, and Logical Volumes

The `partNew` command prepares a storage device to use in NetWitness Platform. It performs the following tasks.

- Creates the partition table on the block device.
- Creates the Linux Volume Manager physical device partition.
- Creates a volume group containing the physical device.
- Creates logical volumes in the volume group.
- Creates XFS filesystems on each logical volume.
- Creates `/etc/fstab` entries for each logical volume.
- Mounts each logical volume.

Complete the following steps to allocate block devices to partitions, volume groups, and logical volumes.

1. Run the `devlist` command to locate unused block devices. The following example shows the `devlist` command output.

Output (or command manual help)

```
sda: vendor=DELL model="PERC H730P Mini" size="931 GB" used=1
sdb: vendor=DELL model="PERC H730P Mini" size="1.81 TB" used=1
sdc: vendor=DELL model="PERC H830 Adp" size="21.38 TB" used=1
sdd: vendor=DELL model="PERC H830 Adp" size="85.53 TB" used=1
```

Also, you must provide a name for the service that will be used with the storage, for example, **decoder** for the Network Decoder service, or **concentrator** for the Concentrator service. You have the option of providing the volume type. The default volume type has the same name as the service.

Note: Run the `devlist` command to see if the multipath user-friendly names are listed correctly.

- Run the `multipath -ll` command to make sure that SAN devices are attached. The following is an example when SAN devices are attached.

```
[root@116Decoder40GBDTrans block]# multipath -ll
mpathb (36006016001e04100babaab5acb9a24e0) dm-17 DGC ,VRAID
size=20T features='2 queue_if_no_path retain_attached_hw_handler' hwhandler='1 alua' wp=rw
|+- policy='service-time 0' prio=50 status=active
| - 17:0:0:1 sdj 8:144 active ready running
|+- policy='service-time 0' prio=10 status=enabled
| - 7:0:0:1 sdh 8:112 active ready running
mpatha (36006016001e04100e5baab5a5c2c6979) dm-2 DGC ,VRAID
size=10T features='2 queue_if_no_path retain_attached_hw_handler' hwhandler='1 alua' wp=rw
|+- policy='service-time 0' prio=50 status=active
| - 17:0:0:0 sdi 8:128 active ready running
|+- policy='service-time 0' prio=10 status=enabled
| - 7:0:0:0 sdg 8:96 active ready running
[root@116Decoder40GBDTrans block]#
```

Note: Block devices should be configured with a user-friendly name such as `mpatha`, `mpathb` etc.

- Execute the `partNew` command to allocate block devices to partitions, volume groups, and logical volumes.

By default, the `partNew` command does not make changes. It displays the actions that will be taken if you commit the command string. To actually make the changes to the system, add the `commit=true` parameter to the command.

For example, to assign devices `sdd` and `sde` to Decoder:

```
send /appliance partNew name=sdm service=decoder volume=decodersmall
commit=true
send /appliance partNew name=sdd service=decoder volume=decoder commit=true
```

Caution: For the **decoder** and **concentrator** services, you must create storage volumes in a specific order.

- The **decoder** has the **decodersmall** and **decoder** volumes. Create the **decodersmall** volume before the **decoder** volume because **decodersmall** contains the small filesystem mounted at `/var/netwitness/decoder`.

- The **concentrator** has the **concentrator** and **index** volumes. Create the **concentrator** volume before **index** volume or it will fail and you receive the following message.

```
Failed to process message partNew for /appliance
com.rsa.netwitness.carlos.transport.TransportException: Volumes for index
require mount point /var/netwitness/concentrator to be created and
mounted first.
```

- Execute the `vgs` command to validate that the `partNew` command created the correct Logical Volumes.

The output of this command:

- Enumerates all the volume groups on this host.
- Displays the physical volumes that the volume group consists of, and the logical volumes within the volume group.

- Go to [Task 4 - Allocate Volume Groups to NetWitness Services- `srvAlloc`](#).

Task 4 - Allocate Volume Groups to NetWitness Services - `srvAlloc`

The `srvAlloc` command configures services on a host to use storage in a volume group. You must provide the name of the service to configure and the volume group to assign to the service (the service you provide must be installed on the host). For information about NetWitness Platform service volumes, see "NetWitness Platform Service Volume Reference" in [Storage Requirements](#).

Allocate services in the following order:

- For the Decoder, allocate `decodersmall` first then the `decoder`
- For a Concentrator, allocate `concentrator` first then `index`.



Note: By default, the `srvAlloc` command does not make changes. You must append the `commit=1` parameter to the command string to actually make the changes to the system and restart the specified service after making changes.

1. Execute the `srvList` command to see a list of services installed on this host.
The `srvList` command communicates with the service through the SSL port. You install a Category on a host. A Category can be a single service, or multiple related services, located on the same host.
2. Execute the `srvAlloc` command to configure a service on a host to use storage in a volume group.
For example:

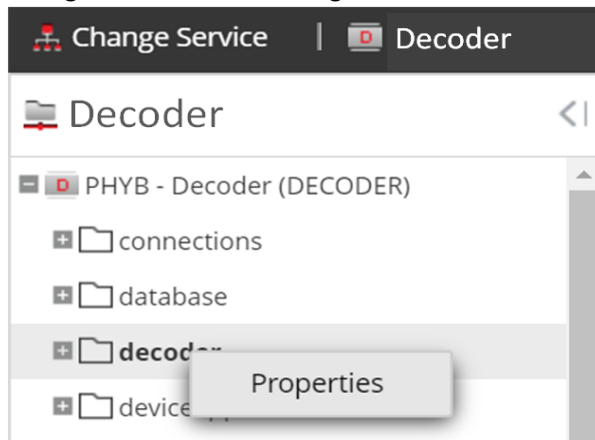
```
service=concentrator volume=concentrator commit=1
service=concentrator volume=index commit=1
```
3. Go to Task 5 - Reconfigure Services and Databases to Detect and Appropriately Use New Storage.

Task 5 - (Optional) Reconfigure Storage Configuration for 10G Capture

You need to reconfigure the Decoder service and databases for 10G capture. Complete the following steps so that the Network Decoder service and its database detect and use new free space.

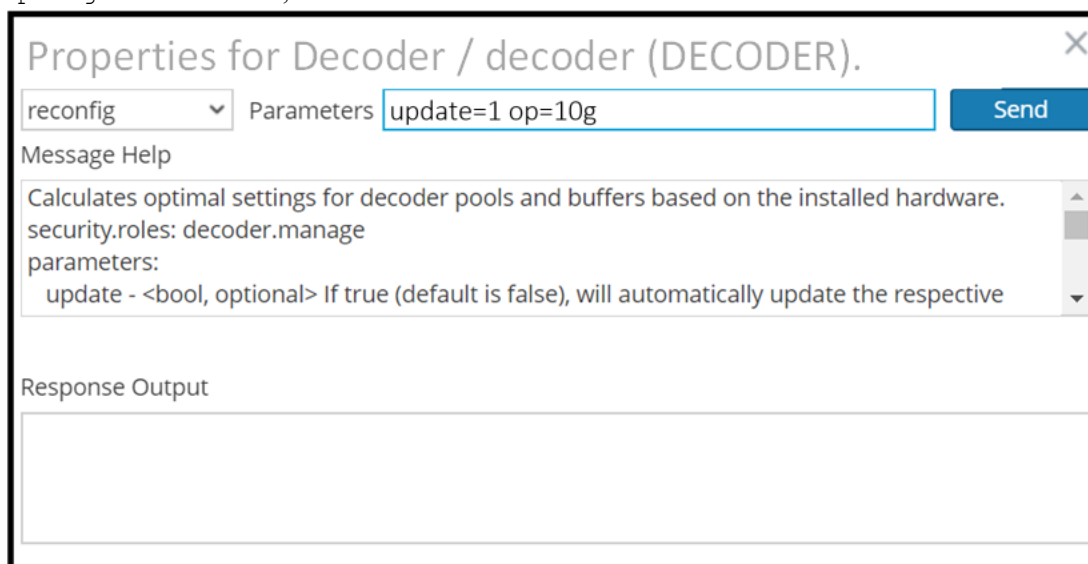
1. In the **NetWitness** menu, go to  (Admin) > **SERVICES**.
The **SERVICES** view is displayed.
2. Select the **decoder**.
3. Under  (actions), select **View** > **Explore**.
The **Explore** tree for the service is displayed.

4. Reconfigure space on the **decoder** service.
 - a. Navigate to the **decoder**, right click, and click **Properties**.

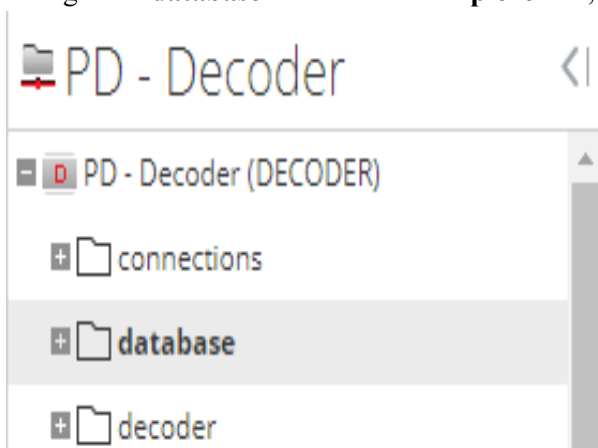


The **Properties** dialog is displayed.

- b. Execute the `reconfig` command by selecting it from the drop-down list, specify `update=1 op=10g` in **Parameters**, and click **Send**.

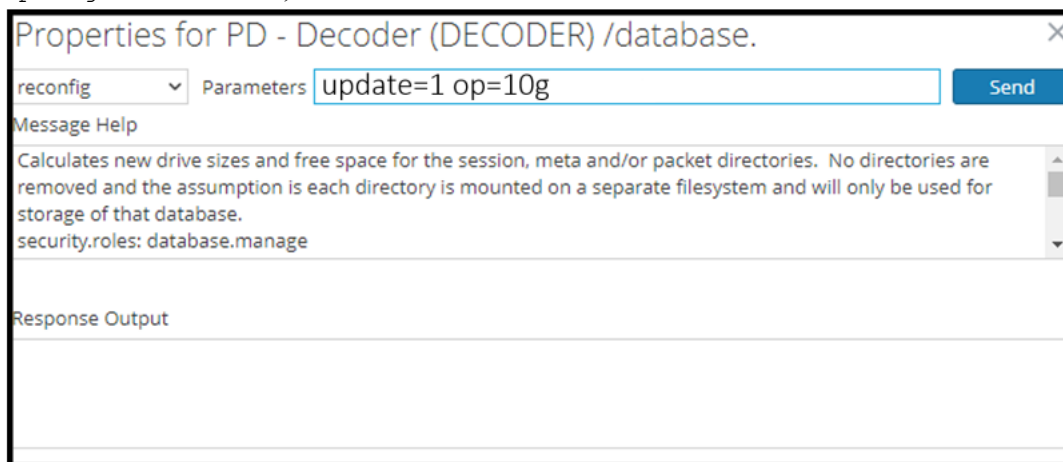


5. Reconfigure space on the database.
 - a. Navigate to **database** in the service **Explore** tree, right click, and click **Properties**.



The **Properties** dialog is displayed.

- b. Execute the `reconfig` command by selecting it from the drop-down list, specify `update=1 op=10g` in **Parameters**, and click **Send**.



Prepare Unity Storage

You must work with your Dell EMC Storage Engineer to allocate storage within your Unity environment for the NetWitness Platform and ensure the allocated storage meets all of the NetWitness Platform Storage Requirements. Specifically, make sure that:

- You have at least two LUNS created for Decoders (meta /session and packet volumes).
- You have at least two LUNS created for Concentrators (index and meta volumes).
- Ensure block devices can meet the minimum IOPS for expected ingestion rates.

You must add every NetWitness host that uses the Unity storage as a host within the Unity interface. After you create hosts and LUNs, you must assign the LUNs to the hosts. Assigning the LUNs to hosts makes the storage visible to the hosts so they can locate the storage through the host-based Dell EMC PowerPath software.

Note: A Dell EMC engineer will configure the following Unity Array.

You need to perform the following tasks to prepare Unity Storage.

[Task 1 - Access Unisphere User Interface \(UI\)](#)

[Task 2 - Create Pools](#)

[Task 3 - Create LUNS](#)

[Task 4 - Register Hosts](#)

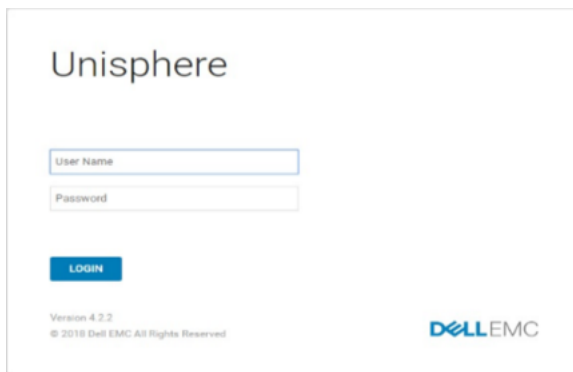
[Task 5 - Assign LUNS to Hosts](#)

[Task 6 - Install PowerPath](#)

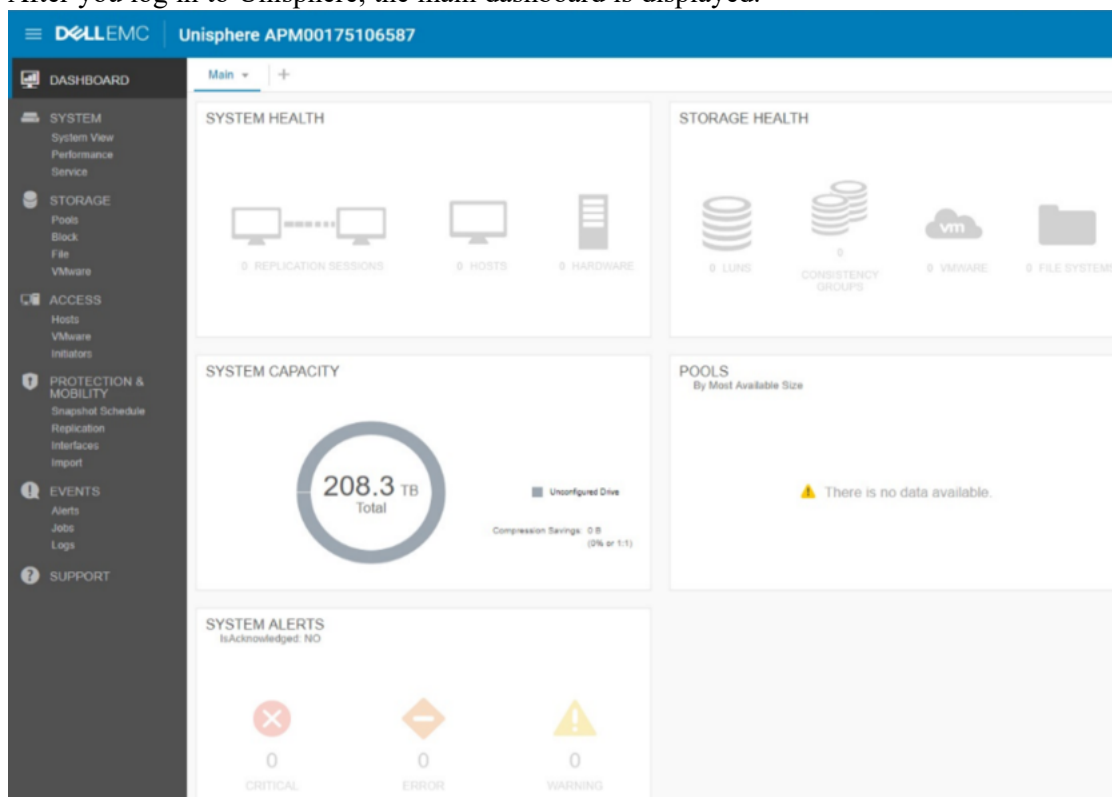
Task 1 - Access Unisphere User Interface (UI)

1. Connect your workstation on the same subnet as the UNITY.
2. Open a browser and go to **http://<unisphereIP>** to connect to the Unisphere UI.
3. Log in with the credentials provided by the DellEMC CE. The default credentials are **admin/Password123#**.

Note: Unisphere will ask you to change password the first time log in. It also asks you to install the license before you can configure array (DellEMC CE may do this for you. You must get the new admin password from them).



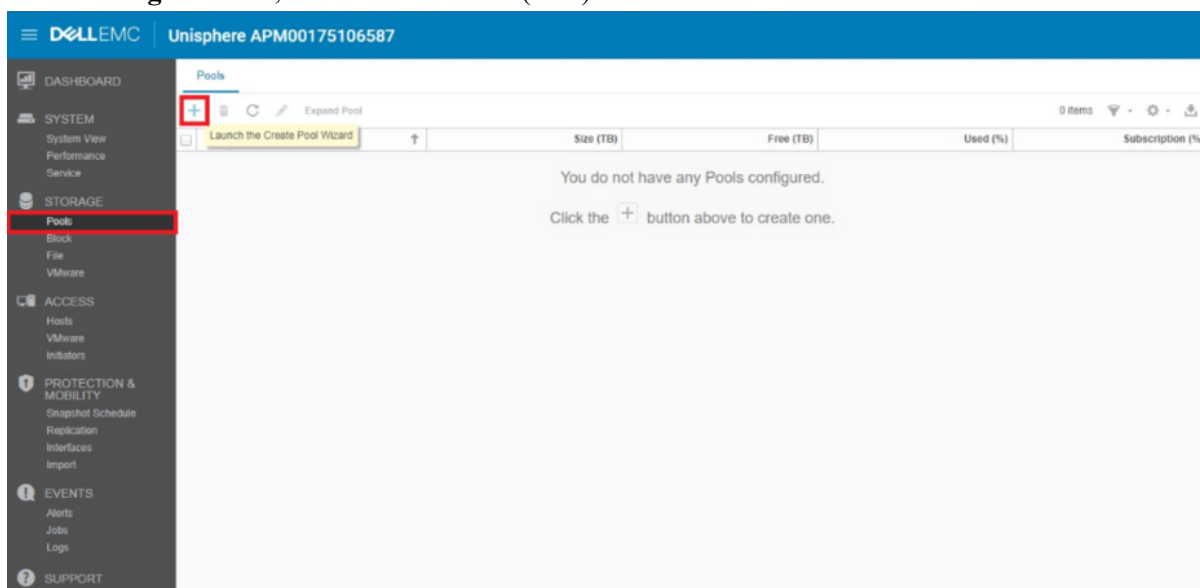
After you log in to Unisphere, the main dashboard is displayed.



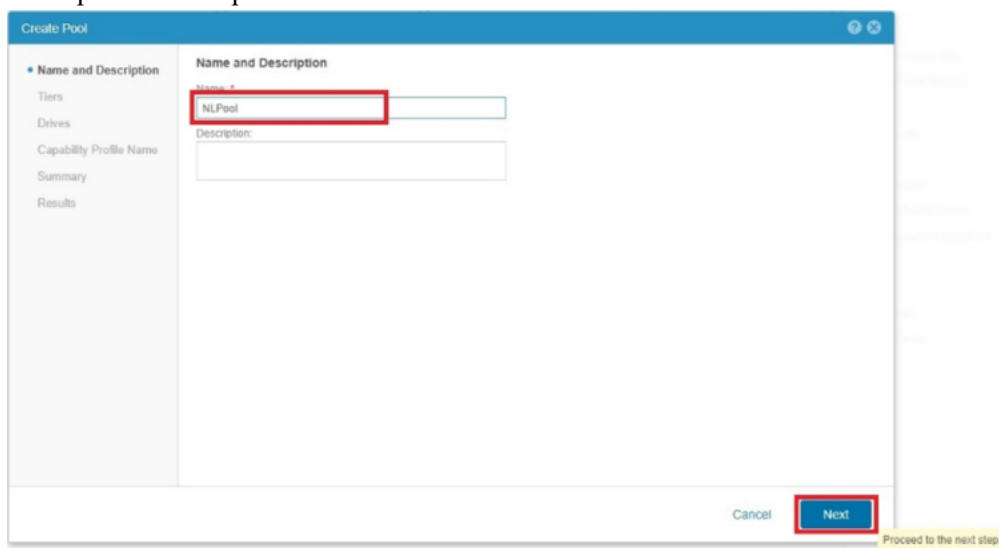
Task 2 - Create Pools

The NetWitness configuration consists of two different pools. One pool is dedicated to the NL-SAS drives and the other pool is dedicated to the SSDs.

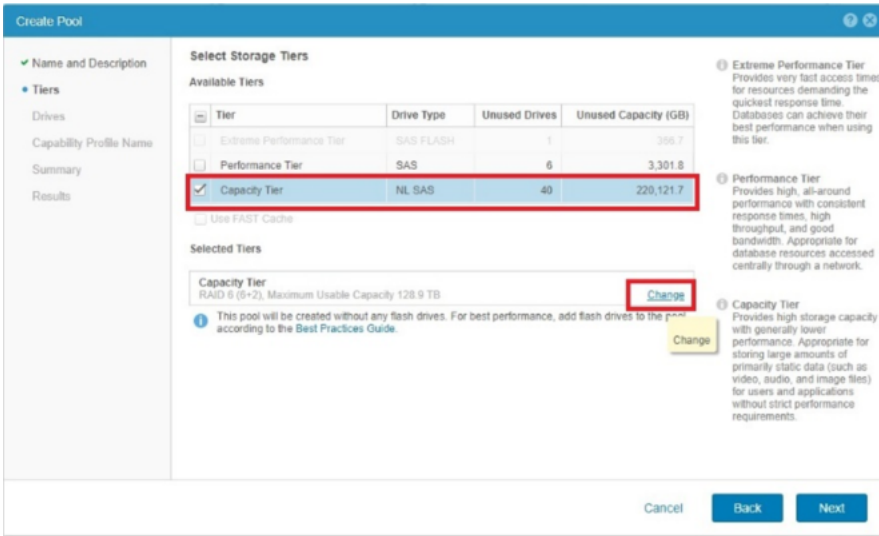
1. From **Storage Section**, click > **Pools** > **+** (Add) to launch the Create Pool Wizard.



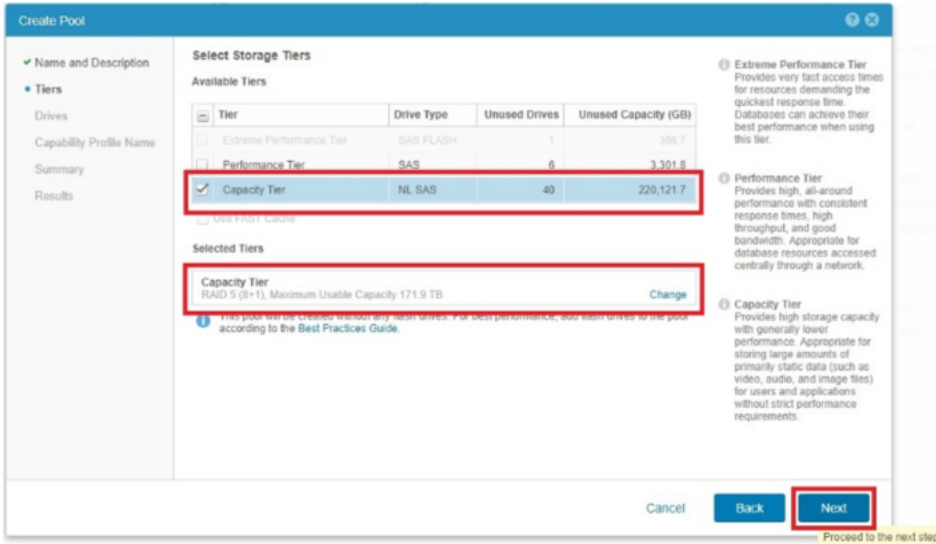
2. Enter in a name for the pool (for example, **NLPool**) and click **Next**. Optionally, you can also enter a description for the pool.



3. Select **Capacity Tier** under **Tier** for the tier type (drive type) and click **Change**.



4. Choose the RAID type and from the drop down and select the RAID size.
The RAID type and size are a customer preference. The only requirement is to make sure you have enough IOPS within the pool to accommodate the log or packet capture and queries. In the following example, a **RAID 5 (8+1)** configuration is selected, however some customers may prefer a **RAID 6 (10+2 or 12 +2)**.
5. Make sure you have the correct Raid type and size selected.



6. Choose the number of drives you want to add into the pool and click **Next**.

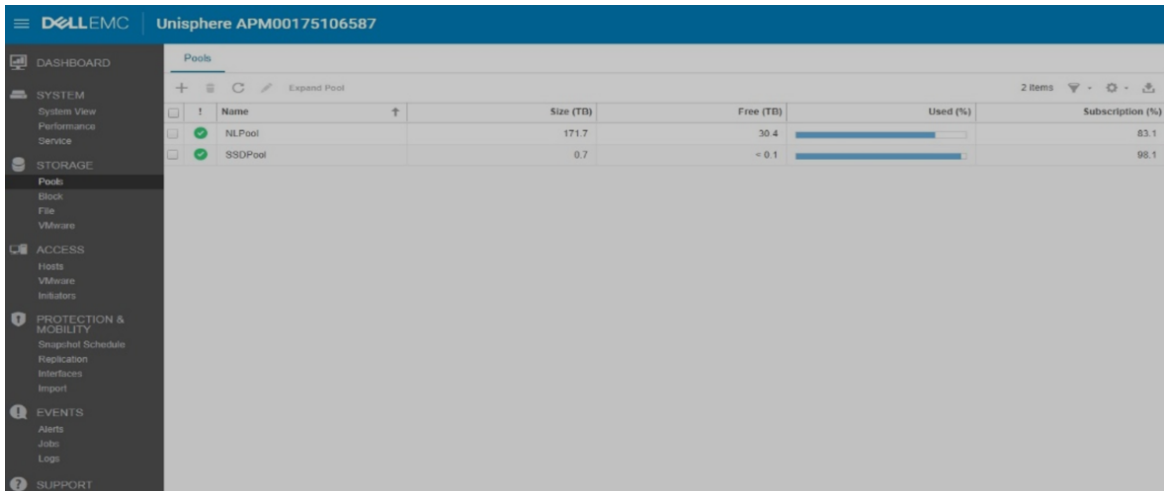
The screenshot shows the 'Create Pool' wizard in the 'Select Amount of Storage' step. The left sidebar has 'Drives' selected. The main area shows 'Capacity Tier - RAID 5 (8+1), Maximum Usable Capacity 171.9 TB' and a dropdown menu for '6.0 TB NL SAS (7.2K RPM)' with 'Add 36 of 40 Drives (Usable Capacity 171.9 TB)' selected. Below this, it says 'Totals: 36 Drives (171.9 TB)'. At the bottom right, the 'Next' button is highlighted with a red box, and a yellow tooltip says 'Proceed to the next step.'

7. Skip the **VMware Capability** section and click **Next**.

The screenshot shows the 'Create Pool' wizard in the 'VMware Capability Profile Name and Description' step. The left sidebar has 'Capability Profile Name' selected. The main area has a checkbox 'Create VMware Capability Profile for the Pool' which is unchecked. There are input fields for 'Name' and 'Description'. To the right, there is a note: 'To be able to use a pool for VMware VVols based storage provisioning it is necessary to expose a Capability Profile for the pool. Please enter name and description for the Capability Profile'. At the bottom right, the 'Next' button is highlighted with a red box.

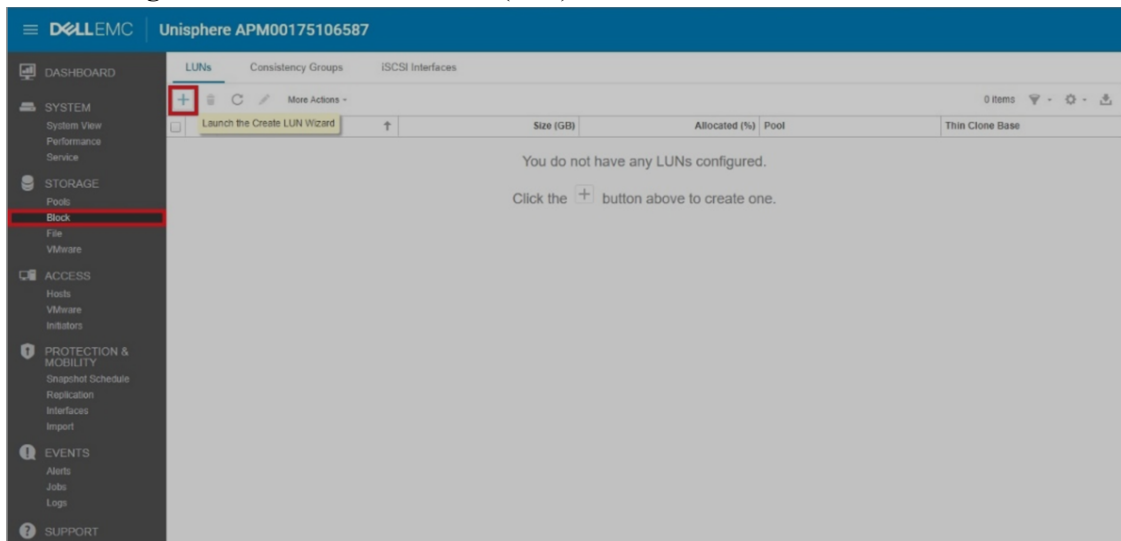
8. Make sure that everything is correct on the Summary tab, and click **Finish**.
9. Create another pool for the SSDs using steps 2 – 8.
- Enter in a name for the other pool (for example, **SDDPool**) and click **Next**. Optionally, you can also enter a description for the pool.
 - Select **Extreme Performance Tier** under **Tier** for the tier type (drive type) and click **Change**.
 - Choose the RAID type and from the drop down, select the RAID size, and click **OK**.

Note: Raid 5 (4+1) RAID Configuration is different then Capacity Tier.



Task 3 - Create LUNS

- From **Storage** section, click **Block** > **+** (Add) to launch the **Create LUN Wizard**.



The table below list all of the possible LUNS you may need to create. The ConIndex is the only LUN you need to assign to the SSD Pool. Make sure that the LUN sizes do not exceed what is listed below.

| | | | |
|----------------|---------------|-----------------|----|
| DecoderLarge01 | 75 TB orLess | NL Pool | No |
| DecoderSmall01 | 20 TB or Less | NL Pool | No |
| Concentrator01 | 15 TB or Less | NL Pool | No |
| Archiver01 | 75 TB or Less | NL Pool | No |
| ConIndex01 | 3 TB or Less | SSD Pool | No |

2. Enter the LUN Name from the list. Optionally, you can enter a description of LUN.
3. Select the appropriate pool from the list on the drop-down menu.
4. Deselect the **Thin** checkbox (These will be fully provisioned LUNs).
5. Select **Next** to proceed to the next menu.

The screenshot shows the 'Create LUNs' configuration window. The 'Configure LUN(s)' tab is active. The 'Name' field is set to 'DecoderLarge01', the 'Pool' is 'NLPool (Capacity Tier, 171.9 TB free)', the 'Size' is '20 TB', and the 'Thin' checkbox is unchecked. The 'Next' button is highlighted in red. A yellow tooltip at the bottom right says 'Proceed to the next step.'

6. Click **Next** until you get to the summary section.
7. Verify that the **Name**, **Pool**, **Size** and **Thin** selections are all correct.
8. Click **Finish** to complete LUN creation.

The screenshot shows the 'Create LUNs' configuration window in the 'Summary' tab. The 'Name' is 'DecoderLarge01', the 'Pool' is 'NLPool', the 'Size' is '20.0 TB', and 'Thin' is 'No'. The 'Finish' button is highlighted in red.

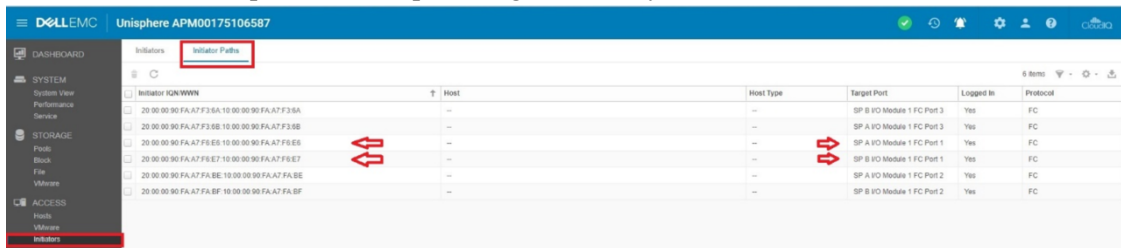
- Repeat steps 2- 8 for the remaining LUN creations.

Task 4 - Register Hosts

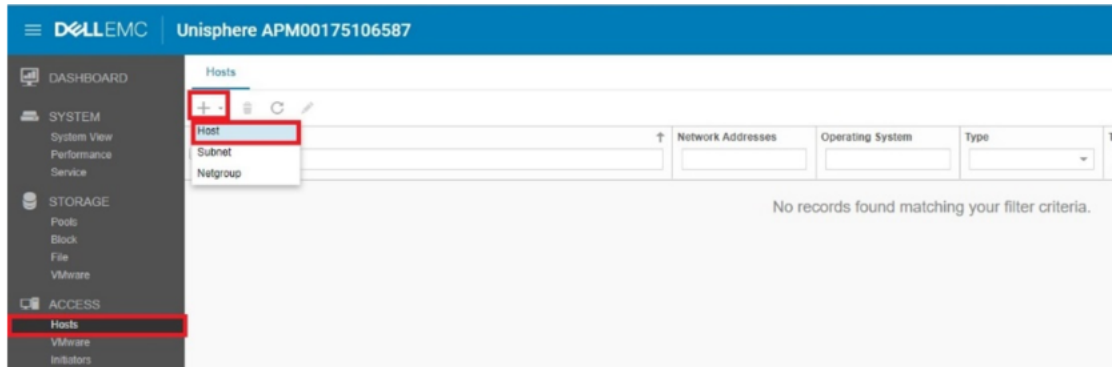
Before proceeding, record the hostname and IP address of the Head Unit and make sure that the HBAs in the head unit are properly cabled to the UNITY.

- From the Access section, click **Initiators**.
- Under the **Initiator Paths** tab, make sure that the correct HBAs are selected that you will use to register the Head Unit.

You should see two initiators per Head Unit. This represents the fiber connection from port 1 to SPA and port 1 to SPB. If you have multiple head units, the easiest method is to power each down and then power them up and register one by one.



- From the **Access** section, click **Hosts** > **+** (Add) > **Host** to add a host configuration.



- Enter the Hostname of the Head Unit.
- Under **Operating System**, select **Linux** from the drop down menu.
- Enter the IP address of the Head Unit.

7. Click **Next** to proceed to the next section.

Add a Host

Specify a Name and Additional Information

Name: * 95Decoder

Description:

Operating System: Linux

Network Address: 10.25.66.32

Tenant: Select or enter a tenant

While the host operating system information is not required, providing it will allow for more specific setup and troubleshooting instructions.

In order to customize access to NFS shares, the Network Address (name or IP address) is required. Port information is not allowed.

Network Address examples:
 IPv4 address: 192.168.1.2
 IPv6 address: FE80:3202:B3FF:FE1E:8329
 Host name: hostname

Tenant information is not required. To create a tenant, select the Tenants tab for a file system.

Cancel **Next**

Proceed to the next step.

8. In the Initiators section, select the two initiators that correspond to the correct port associated with the Head Unit and click **Next** to proceed.

Add a Host

Select Discovered Initiators or Manually Add Initiators

Automatically Discovered Initiators

| Initiator IQN/WWN | Connected To |
|---|-----------------------------|
| <input checked="" type="checkbox"/> 20:00:00:90:FA:A7:F5:E6:10:00:00:90:FA:A7:F5:E6 | SP A iVO Module 1 FC Port 1 |
| <input type="checkbox"/> 20:00:00:90:FA:A7:FA:BF:10:00:00:90:FA:A7:FA:BF | SP B iVO Module 1 FC Port 2 |
| <input type="checkbox"/> 20:00:00:90:FA:A7:F3:6A:10:00:00:90:FA:A7:F3:6A | SP B iVO Module 1 FC Port 3 |
| <input checked="" type="checkbox"/> 20:00:00:90:FA:A7:F3:6A:10:00:00:90:FA:A7:F3:6A | SP B iVO Module 1 FC Port 3 |

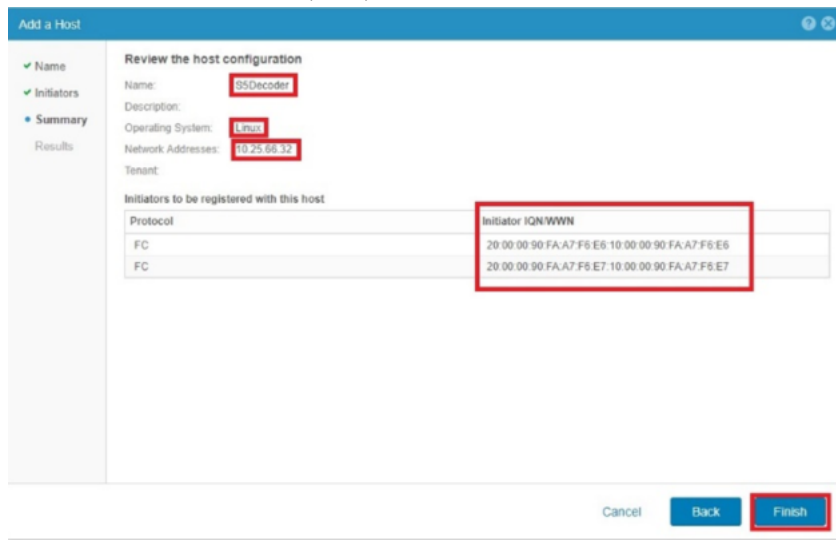
Manually Added Initiators

No initiators have been manually added yet. Click the + button to manually add an initiator.

Cancel **Back** **Next**


Proceed to the next step.

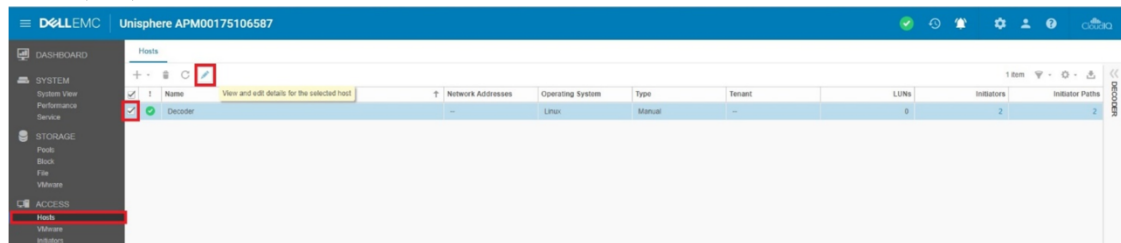
- Make sure that the **Name**, **OS**, **IP** and **WWNs** are correct and click **Finish**.



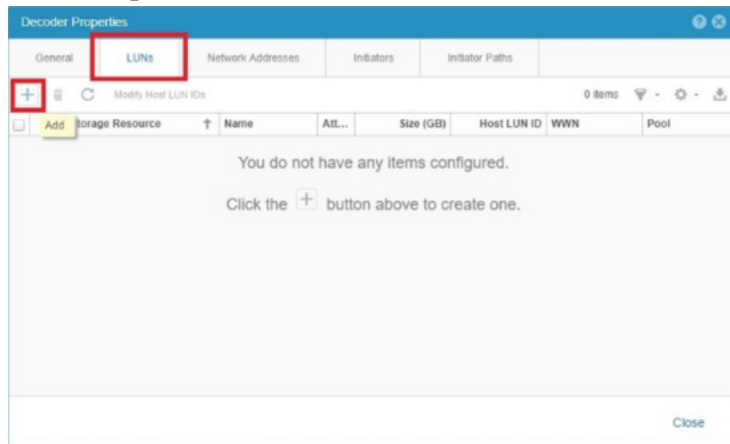
- Repeat steps 2-9 for all Head Units.
- In the Initiators section, select the two initiators that correspond to the correct port associated with the Head Unit. Then click “Next” to proceed.

Task 5 - Assign LUNS to Hosts

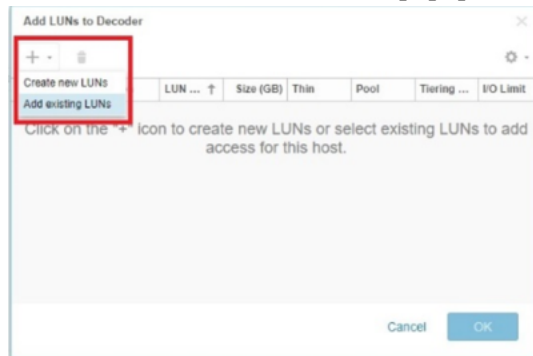
- From the **Access** section, click **Hosts**, select the head unit (for example, **Decoder**) and click  (edit) to view and edit details for the selected host.



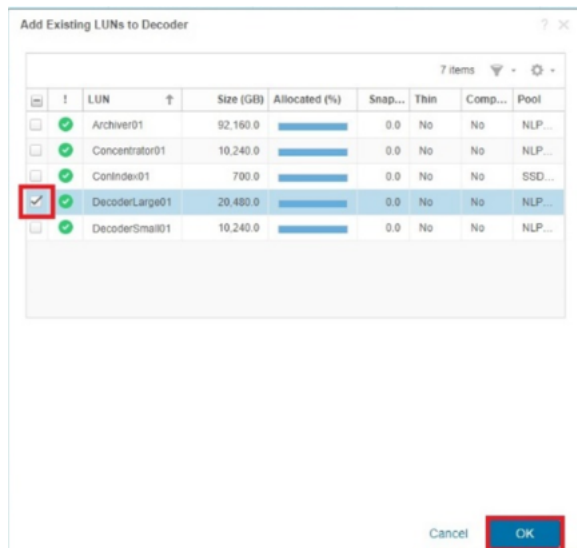
2. In the **Properties** section, select the **LUNS** tab and click **+** (Add icon).



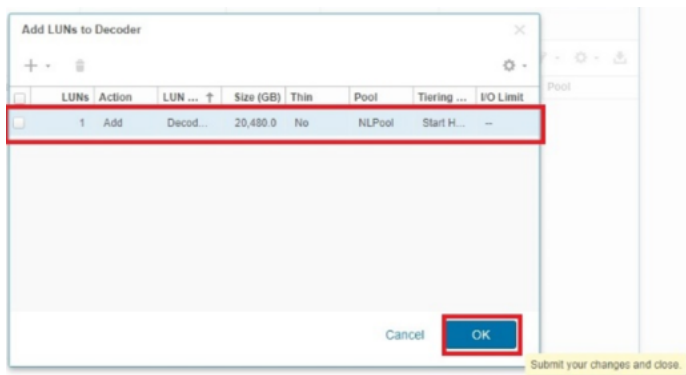
3. From the **Add LUNs to <Host>** popup, click **+** > **Add existing LUNs**.



4. Select the LUN to add to the Head Unit and **OK**.

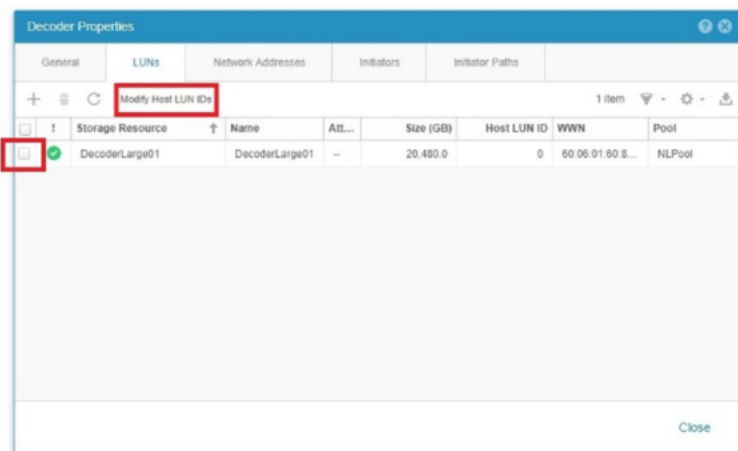



5. Make sure that the correct LUN was added to the host and click **OK**.



6. (OPTIONAL) If you need to modify the HLU (Host LUNN Unique ID):

- a. Select the LUN you want to change.
- b. Click **Modify Host LUN IDs**.



7. Click  (edit), change the HLU to the number you want, and click **OK**.

Task 6 - Install PowerPath

1. Make sure that the Emulex ports on the Decoder host are attached to the Unity.
2. Log in to root on the Decoder attached to the Unity with the admin credentials.
3. Install PowerPath and register the Dell EMC PowerPath licenses for Unity hardware.

```
yum install DelleMCPower.LINUX-6.4.0.00.00-95.RHEL7.x86_64.rpm
```

Note: When you purchase an NetWitness Provided Unity, PowerPath licenses are sent to you. You can download PowerPath at support.dell.com.

Note: It is possible that the RPM downloaded from Dell EMC is not signed with a cert that the NetWitness device has available, which can cause the installation to fail with the package not signed error. Run the yum install with the `--ngpgcheck` option to enable the software to install.

4. Make sure that all the PowerPath connections are correct.

```
powermt display dev=all
```

The following output is an example of valid PowerPath connections.

```
===== Host ----- - Stor - -- I/O Path -- -- Stats ---
### HW Path          I/O Paths   Interf.  Mode   State  Q-IOs Errors
=====
   15 lpfc            sde        SP A6   active  alive   0     0
   18 lpfc            sdg        SP B6   active  alive   0     0

Pseudo name=emcpowerb
Unity ID=APM00174407815 [Host_62]
Logical device ID=600601609D9046006996745A46B60AB6 [DecoderSmall101]
state=alive; policy=CLAROpt; queued-IOS=0
Owner: default=SP A, current=SP A      Array failover mode: 4
=====
----- Host ----- - Stor - -- I/O Path -- -- Stats ---
### HW Path          I/O Paths   Interf.  Mode   State  Q-IOs Errors
=====
   15 lpfc            sdd        SP A6   active  alive   0     0
   18 lpfc            sdf        SP B6   active  alive   0     0
```

5. Verify that the PowerPath license is installed using the `emcpreg` command.

```
[root@NWAPPLIANCE24932 ~]# emcpreg -list
Key BQPO-DB4M-VFC2-Q24R-ML9Z-EQTU
Product: PowerPath
Capabilities: Al
```

6. Add the following string to the `/etc/lvm/lvm.conf` file to filter the LVM (Logical Volume Manager) so that it ignores duplicate volumes.

```
filter = [ "a|^/dev/sda2$|", "a|^/dev/sdb1$|",
"a|^/dev/emcpower.*|", "r|.*/|" ]
```

7. Run the following commands in this order:

- a. `systemctl enable PowerPath.service`
- b. `systemctl start PowerPath.service`

8. Reboot the Decoder.

9. Complete the instructions in [Configure Storage Using the REST API](#) to complete storage configuration.

Migrate Data to Another Storage Type

This section provides two options for moving data from DACs to PowerVaults:

[Migrate Data Using the Warm and Hot Tier Option](#)

[Move Data From DAC to PowerVault](#)

Refer to the Hardware Setup Guides on [NetWitness Community](#) for detailed instructions for setting up NetWitness Platform host and storage hardware.



Migrate Data Using the Warm and Hot Tier Option

In this procedure, you configure a warm tier for the DAC's, so that they do not write any new data. The warm tier continues to be available for analyst operations. You configure the PowerVaults as a hot tier, where new data can be written and available for analysts. When the required data retention is available on the hot tier, the warm tier can be decommissioned.

To set up the warm and hot tiers, perform the following tasks:

- [Stop the Service](#)
- [Set Up PowerVault](#)
- [Configure The Mount Points](#)
- [Set up Warm and Hot Tiers](#)
- [Decommission the DAC](#)

Stop the Service

1. Log in to the NetWitness Platform user interface.
2. Go to  (Admin) > **SERVICES** and select the service (for example, Log Decoder).
3. Click  > **View** > **Config**, and under Log Decoder Configuration, clear the **Capture Autostart** checkbox, and then click **Apply**.
4. In the menu bar, click the down arrow next to **Config**, select **System**, and at the top of the panel, click **Stop Capture**.
5. From the command line interface in NwConsole, stop the service by running the following command:

```
systemctl stop nwlogdecoder
```

Set Up PowerVault

1. Go to the REST API for the service by entering the IP address of the service, in this example, the Log Decoder. For example, `172.16.0.1:50106`.
2. Click the asterisk (*) next to the service. for example, **decoder (*)**.

3. Under **Properties for /decoder**, click the down arrow, select **RaidNew** and enter the following parameters, entering the name of the service for scheme. In this example, we use `logdecoder`.
`controller=1 enclosure=75 scheme=logdecoder commit=1`
4. Click **Send**.
5. To configure the partitions, click the down arrow again, select **PartNew**, and enter the following parameters,
`name=sde service=logdecoder volume=logdecoderssmall commit=1`
6. Click **Send**.
7. With **PartNew** still selected, enter the following parameters:
`name=sdf service=logdecoder volume=logdecoder commit=1`

Note: To validate the partition definitions before committing them, you can enter these parameters without `commit=1`, and click **Send**. After you validate the parameters, add `#commit=1` and then click **Send** to commit the parameter settings.

Configure The Mount Points

1. On the NwConsole at the root level of the service (for example, the Log Decoder), run `df -h`.
A list of mounted partitions is displayed.
2. Unmount all of the old storage points of the DAC and copy all the data to the Log Decoder. At the root level, run the `umount` command and the path name of each partition. You can concatenate the path names, for example:

```
umount /var/netwitness/logdecoder/index
/var/netwitness/logdecoder/sessiondb /var/netwitness/logdecoder/metadb
/var/netwitness/logdecoder/packetdb /var/netwitness/logdecoder/index0
/var/netwitness/logdecoder/sessiondb0 /var/netwitness/logdecoder/metadb0
/var/netwitness/logdecoder/packetdb0
```
3. Temporarily mount the petitions in the `decoroot` folder in the `/mnt` directory in order to access the files. For example:

```
mount /dev/mapper/logdecoderssmall-decoroot /mnt/decoroot/
```
4. Copy the contents of `decoroot` from `/mnt` to `/var/netwitness/logdecoder`, answering Y (yes) to the prompts:

```
cp -R statdb /var/netwitness/logdecoder/
```
5. Unmount `/mnt/decoroot`.

```
umount /mnt/decoroot
```
6. Comment out `decoroot` from `/etc/fstab`, as this was on the DAC and the DAC will be decommissioned.



```
#/dev/logdecoderssmall/decoroot
/var/netwitness/logdecoder/xfs/noatime,nosuid 1 2
```
7. Mount all the remaining file systems.

```
mount -a
```
8. Start the `nwlogdecoder` service (with capture still disabled).

```
systemctl start nwlogdecoder
```

Set up Warm and Hot Tiers

Caution: Before you set up warm and hot tiers, be sure that you know the right warm and hot tier entries for each collection so that you can set them up accurately.

1. Go to  (Admin) > **SERVICES** and select the service (for example, Log Decoder).
2. For the Log Decoder service, click  > **View** > **Explore**, and go to **database** > **config**.
 - a. Copy the contents of `meta.dir` and paste them to `meta.dir.warm` as shown in the following example:

| logdecoder - Log Dec... < | /database/config | logdecoder - Log Decoder |
|--|------------------------|---|
| logdecoder - Log Decoder (LOG_DECODER) | hash.algorithm | none |
| connections | hash.databases | session,meta,packet |
| database | hash.dir | |
| config | manifest.dir | |
| stats | meta.compression | none |
| decoder | meta.compression.level | 0 |
| deviceappliance | meta.dir | /var/netwitness/logdecoder/metadb=4.58 TB |
| index | meta.dir.cold | |
| logs | meta.dir.warm | |
| rest | meta.file.size | auto |
| | meta.files | auto |

| logdecoder - Log Dec... < | /database/config | logdecoder - Log Decoder |
|--|------------------------|---|
| logdecoder - Log Decoder (LOG_DECODER) | hash.algorithm | none |
| connections | hash.databases | session,meta,packet |
| database | hash.dir | |
| config | manifest.dir | |
| stats | meta.compression | none |
| decoder | meta.compression.level | 0 |
| deviceappliance | meta.dir | /var/netwitness/logdecoder/metadb=4.58 TB |
| index | meta.dir.cold | |
| | meta.dir.warm | /var/netwitness/logdecoder/metadb=4.58 TB |

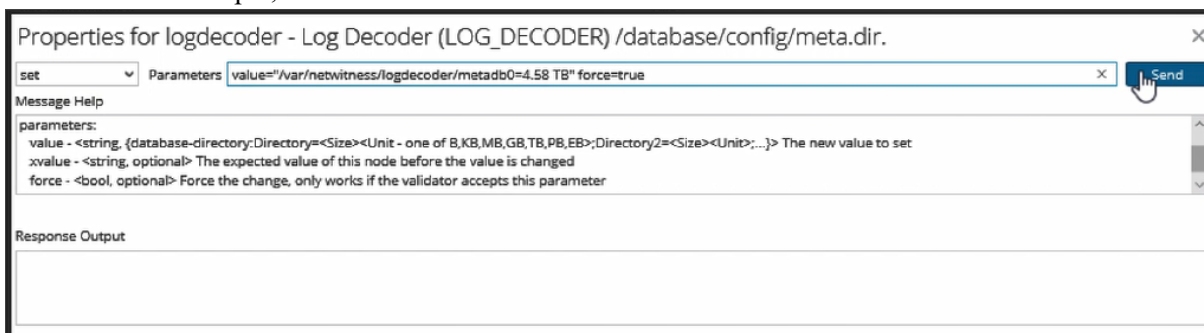
- b. In the same way, copy the packet database in `packet.dir` to `packet.dir.warm`.
 - c. Copy the session database in `session.dir` to `session.dir.warm`.
3. Go to **index** > **config** and copy `index.dir` to `index.dir.warm`.

Note that the new volumes end in 0, so PowerVault will write to the directories ending in 0, for example:

```
[root@logdecoder ~]# df -h
Filesystem                Size      Used Avail Use% Mounted on
/dev/mapper/netwitness_vg00-root 30G    3.3G    27G   11% /
devtmpfs                   63G         0    63G    0% /dev
tmpfs                       63G     12K    63G    1% /dev/shm
tmpfs                       63G     34M    63G    1% /run
tmpfs                       63G         0    63G    0% /sys/fs/cgroup
/dev/sdal                  1019M     96M    924M   10% /boot
/dev/mapper/netwitness_vg00-nwhome 3.3T    1.2G    3.3T    1% /var/netwitness
/dev/mapper/netwitness_vg00-usrhome 10G     33M    10G    1% /home
/dev/mapper/netwitness_vg00-varlog 10G    1.5G    8.6G   15% /var/log
tmpfs                       13G         0    13G    0% /run/user/0
/dev/mapper/logdecodersmall-index 30G     54M    30G    1% /var/netwitness/logdecoder/index
/dev/mapper/logdecodersmall-sessiondb 600G    733M    599G    1% /var/netwitness/logdecoder/sessiondb
/dev/mapper/logdecodersmall-metadb 4.9T    11G    4.9T    1% /var/netwitness/logdecoder/metadb
/dev/mapper/logdecoder-packetdb 31T     12G    31T    1% /var/netwitness/logdecoder/packetdb
/dev/mapper/logdecodersmall0-index 30G     33M    30G    1% /var/netwitness/logdecoder/index0
/dev/mapper/logdecodersmall0-sessiondb 600G    34M    600G    1% /var/netwitness/logdecoder/sessiondb0
/dev/mapper/logdecodersmall0-metadb 21T     34M    21T    1% /var/netwitness/logdecoder/metadb0
/dev/mapper/logdecoder0-packetdb 86T     35M    86T    1% /var/netwitness/logdecoder/packetdb0
[root@logdecoder ~]#
```

Update the Decoder configuration with the path to the PowerVault mount by adding a 0 to the path.

1. In the `/database/config` column, right-click **meta.dir** and click **Properties**.
2. In **Properties for logdecoder**, select **set**, and in **Parameters**, enter `value="/var/netwitness/logdecoder/metadb0=4.58 TB" force=true` and add `force=true`, as shown in this example, and then click **Send**.





3. Repeat step 2 for **session.dir**, **packet.dir**, and **index.dir**. Do not be concerned if the size is the same as the DAC in `"=xx GB"`. This will be updated in the next step.

Note: We are only putting the PowerVault paths into the `*.dir` values.


4. Update the sizes for the live PowerVault volumes.
 - a. In the Log Decoder Explore view, in the left panel, right-click **database** and click **Properties**.
 - b. Select **reconfig** and in **Parameters**, enter `update=1` and click **Send**.
 - c. Repeat steps a and b for **index**.
5. Restart the service.

```
systemctl restart nwlogdecoder
```



6. Go to  (Admin) > **SERVICES**, select the Log Decoder service, and click  > **View** > **System**.
7. Click **Start Capture**.
8. Go to the **Config** view, select **Capture Autostart**, and click **Apply**.
9. Reboot the host.

Decommission the DAC

When the DAC data has aged, you should go back into the Explore view and remove all of the *.dir.warm configurations for session, meta, packet and index. You can determine when the DAC data



has aged by going to the Log Decoder  > **View** Explore view. Since we have a hot and warm tier, there are two sets of configuration stats that you need to be aware of. For example, for a packet Decoder, when you look at the packet oldest time in `packet.oldest.file.time`, look at the `packet.oldest.file.time.hot` value and if you see that your DAC had storage up until 30 days ago you can take your DAC offline and decommission it.

These are the basic steps for decommissioning a DAC. NetWitness recommends that you work with your Customer Support representative when you decommission your DACs.

1. Go to  (Admin) > **SERVICES** and select the service (for example, Log Decoder).
2. Click  > **View** > **Config**, and under Log Decoder Configuration, clear the **Capture Autostart** checkbox, and then click **Apply**.
3. In the menu bar, click the down arrow next to **Config**, select **System**, and at the top of the panel, click **Stop Capture**.
4. From the commandline interface in NwConsole, stop the service by running the following command:

```
systemctl stop nwlogdecoder
```
5. Unmount the warm tier. At the root level, run the `umount` command and the path name of each partition. You can concatenate the path names, for example:

```
umount /var/netwitness/logdecoder/index  
/var/netwitness/logdecoder/sessiondb /var/netwitness/logdecoder/metadb  
/var/netwitness/logdecoder/packetdb /var/netwitness/logdecoder/index0  
/var/netwitness/logdecoder/sessiondb0 /var/netwitness/logdecoder/metadb0  
/var/netwitness/logdecoder/packetdb0
```
6. Comment out all the old DAC dbs from `/etc/fstab`, so that only the PowerVault dbs remain.
7. Start the service.

```
systemctl start nwlogdecoder
```
8. In the user interface, go to  (Admin) > **SERVICES** and select the Log Decoder service.
9. Click  > **View** > **Explore** and remove the warm tier configurations:
 - a. In **database** > **config**, delete the content for `meta.dir.warm`, `packet.dir.warm`, `session.dir.warm`.

- b. In **index > config**, delete the content for `index.dir.warm`.
 - c. Go to the **Config** view, select **Capture Autostart**, and click **Apply**
 - d. Go to the **System** view and click **Start Capture**.
10. Restart the service.
- ```
systemctl restart nwlogdecoder
```

The DAC is now unmounted, and is no longer configured in the Decoder for warm storage and is ready to be wiped clean.

1. Remove the logical volume. Run `lvscan` to get a list of the logical volumes.
2. Run `lvremove` on the old logical volumes, for example:
 

```
/dev/logdecodersmall/decroot /dev/lvremove /dev/logdecodersmall/index
/dev/logdecodersmall/sessiondb /dev/logdecodersmall/metadb
/dev/logdecodersmall/packetdb
```
3. Remove the volume groups. Run `vgscan` to get a list of volume groups.
4. Run `vgremove` on the old volume groups (be careful not to remove any volume groups that end in 0, as they are PowerVault).
5. Run `pvscan` to view block devices that are freed up.
6. When the DAC has been successfully removed, reboot the host.

## Move Data From DAC to PowerVault

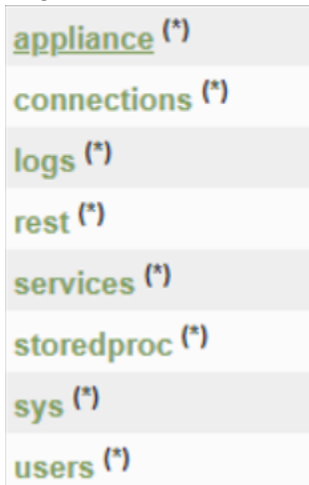
The following procedure describes how to move data from DAC to PowerVault. Before you move data from 2 DACs to 2 PowerVaults, a table, similar to the following table, is displayed if you run the `pvs` (Physical Volume Size) command from the Decoder Linux console (or SSH to the Decoder) with 2 DACs attached and configured to the Decoder. The column headings are Physical Volume (PV), Volume Group (VG), Linux Format (Fmt), Linux Attribute (Attr), Physical Volume Size (PSize), and Physical Volume Free Space (PFree).

| PV        | VG              | Fmt  | Attr | PSize    | PFree |
|-----------|-----------------|------|------|----------|-------|
| /dev/sda2 | netwitness_vg00 | lvm2 | a--  | <930.00g | 0     |
| /dev/sdb1 | netwitness_vg00 | lvm2 | a--  | <1.82t   | 0     |
| /dev/sdc  | decodersmall    | lvm2 | a--  | <5.46t   | 0     |
| /dev/sdd  | decoder         | lvm2 | a--  | <27.29t  | 0     |
| /dev/sde  | decodersmall0   | lvm2 | a--  | <5.46t   | 0     |
| /dev/sdf  | decoder0        | lvm2 | a--  | <27.29t  | 0     |

Complete the following steps to move data from a DAC to a PowerVault.



1. Attach two PowerVaults to a separate PERC controller on the Decoder.
2. Create the devices.
  - a. Open a Browser and specify the ip-address of the Network Decoder and port **50106** to access the REST tool.
  - b. Log in with the `admin` account credentials.



- c. Click on the (\*) next to **appliance** to access the REST command set.
- d. Run `raidList` to display the Controller/Enclosure combination with the new PowerVault enclosures.

In the following example, the output shows `dev/sdg` and `/dev/sdh` on **Controller 2, Enclosure 246**.

```
Controller 2, Enclosure 246
Vendor: DELL
Model: MD1400
In Use: true
Drives: 10.691 TB x 12
Devices: sdg
 sdh
```

- e. Under **Properties for /appliance**, select `raidNew`, specify `controller=<PowerVault-controller-id>` `enclosure=<PowerVault-enclosure-id>` `scheme=decoder` `preferSecure=false`, and click **Send**.

**Note:** You specify `preferSecure=false` if the PowerVault drives are not SED drives. If PowerVault drives are SED drives and you do not want to encrypt them you specify `preferSecure=false`. You must specify `preferSecure=true` if PowerVault drives are SED drives and you want to encrypt them.

3. Go to the Decoder Linux console or SSH to the Decoder and run the following commands.

```
parted -s /dev/sdg mklabel gpt
parted -s -a optimal /dev/sdg mkpart LVM 0% 100%
pvcreate -f /dev/sdg
parted -s /dev/sdh mklabel gpt
parted -s -a optimal /dev/sdh mkpart LVM 0% 100%
```

```
pvcreate -f /dev/sdh
```

If the volume is created successfully, the following message is displayed.

```
Physical volume "/dev/sdg" successfully created
```

**Note:** Repeat this step for every block device. The block device names may be different depending on how many enclosures per per card slot.

4. Run the following command strings to extend the DAC volume group (**decoder**, **decodersmall**) to the PowerVault Physical volume.

```
vgextend decoder /dev/sdg
vgextend decodersmall /dev/sdh
```

5. Run the following command strings to move the data from the DAC to the PowerVault. In this following command string, the DAC is **/dev/sdc** and the PowerVault is **/dev/sdg**.

```
pvmove /dev/sdc /dev/sdg
pvmove /dev/sdd /dev/sdh
```

**Note:** 1.) The `pvmove` command synchronizes data across volumes so that NetWitness can continue ingesting or aggregating data while the migration is executing. You can run the `pvmove` command multiple times if it fails. 2.) Depending on the amount of data on the drives, the move can take a long time complete depending on the amount of data. For example, in a test, it took four hours to move one TB of data.

6. After the move is complete, run the following commands to reduce and remove the DAC drive.

```
vgreduce decoder /dev/sdc
pvremove /dev/sdc
vgreduce decodersmall /dev/sdd
pvremove /dev/sdd
```

7. Detach the physical connections from the DACs to the host.
8. Verify that the Physical volumes are moved from the DACs to the PowerVaults.
  - a. Reboot the host.

```
reboot
```
  - b. Verify that the `/etc/fstab` file is correct.
  - c. Run the `pvs` command and make sure that the **PSize** and **PFree** values are correct on the PowerVault.

```
root@nitifer-decoder:~# pvs
PU UG Fmt Attr PSize PFree
/dev/sda2 netwitness_vg00 lvm2 a-- <930.00g 0
/dev/sdb1 netwitness_vg00 lvm2 a-- <1.82t 0
/dev/sdc1 decodersmall lvm2 a-- 21.38t <15.93t
/dev/sdd1 decoder lvm2 a-- <85.54t 58.25t
```

## Data on PowerVault After Move from DAC

After you move data from 2 DACs to 2 PowerVaults, a table, similar to the following table, is displayed if you run the `pvs` (Physical Volume Size) command from the Decoder Linux console (or SSH to the Decoder) with 2 PowerVaults attached and configured to the Decoder. The column headings are Physical Volume (PV), Volume Group(VG), Linux Format (Fmt), Linux Attribute (Attr), Physical Volume Size (PSize), and Physical Volume Free Space(PFree).

| PV        | VG              | Fmt  | Attr | PSize    | PFree   |
|-----------|-----------------|------|------|----------|---------|
| /dev/sda2 | netwitness_vg00 | lvm2 | a--  | <930.00g | 0       |
| /dev/sdb1 | netwitness_vg00 | lvm2 | a--  | <1.82t   | 0       |
| /dev/sdc1 | decodersmall    | lvm2 | a--  | 21.38t   | <15.93t |
| /dev/sdd1 | decoder         | lvm2 | a--  | <85.54t  | 58.25t  |

---

# Appendix A. How NetWitness Platform Hosts Store Data

---

In most deployments, NetWitness Platform Decoders, Log Decoders, Concentrators, Archivers, and Hybrid hosts require external storage to house their data. Each host uses the external storage in different ways and with different expectations on throughput and performance of the external storage. Some hosts have a higher occurrence of sequential writes and some hosts have a higher occurrence of random reads and writes.

## Decoder Hosts

Log Decoders and Network Decoders capture data and parse meta. The difference between these two hosts is in the type of data they capture:

- Log Decoder captures logs.
- Network Decoder captures packets.

Both Log Decoders and Network Decoders parse out meta data from the raw captured traffic. The meta data is then aggregated to a Concentrator for indexing. The host requires storage to house the raw payload data (raw packets or raw logs) and a cache for the meta extracted during data capture for Concentrator aggregation.

Your retention requirements is a key factor in determining the amount of storage you need for the raw packets or raw logs. In most deployments, you add storage over time based on increased retention requirements and increased capture rates. The storage for the raw data must support a high amount of sequential writes with random reads. Especially in the case of higher speed Network Decoder environments, it is recommended to have a minimum of two partitions exposed to the host to support the throttling between partitions for reads and writes.

The meta cache on a Decoder is generally fixed in size but you can expand it to support additional cache the possible loss of connectivity between the Decoder and a corresponding Concentrator. The meta cache must support a random IOPS rate for sustained writes from the Decoder of meta extracted and the corresponding reads from the Concentrator as meta is aggregated to a Concentrator.

## Concentrator Host

A Concentrator aggregates and indexes the meta data from a Decoder. Both the meta and index storage needs are scaled based on your NetWitness Platform deployment retention requirements. Similar to raw data stored on the Decoders, you may need to increase the storage for both meta data and index data over time to meet your retention requirements.

The meta storage houses all meta data extracted from either a Network Decoder or Log Decoder. Although the ratio of how much meta is extracted may change, the expectations for performance against meta storage is the same for both packet capture and log capture environments. The meta storage must support a sustained amount of sequential writes with random reads of meta data.

The index storage houses the live index generated from the meta data aggregated to a Concentrator. The size of the index is directly related to the size of the meta store. In addition to supporting IOPS for sustained writes, the index also needs to support a much higher rate IOPS for reads than meta based on interactive queries run through analyst interaction and reports and alerts.

## Archiver Host

The Archiver host requires a single partition for both meta and raw log storage. The storage pool deals primarily with sequential writes for long term data written from a Log Decoder or Network Decoder and random reads for reports and analysis.

## Hybrid Hosts

A Hybrid hosts two or more services on a single host. For example:

- A Network Hybrid hosts both the Decoder and Concentrator services handling packets exclusively. It captures packet data and indexes this data to the Concentrator service. Expectations for storage performance match what is outlined for a dedicated Network Decoder host and dedicated Concentrator host.
- A Log Hybrid hosts both the Log Decoder and Concentrator services handling logs exclusively. It captures log data and indexes the data to a Concentrator service. Expectations for performance match what is outlined for a dedicated Log Decoder and dedicated Concentrator.
- An Endpoint Log Hybrid hosts the Endpoint Server, Log Decoder, Concentrator, Log Collector, and Endpoint Broker services. It collects and manages endpoint (host) data from Windows, Mac, and Linux hosts, collects log files and Windows logs from Windows hosts, and generates metadata to correlate endpoint data with sessions from other events sources, such as logs and packets.

## Options for SAN Configurations

If you want to use a Storage Area Network (SAN) , use the same basic drive groups and partition organization that you use for the other NetWitness storage devices. Depending on the SAN configuration and overhead, SAN configurations may require more enclosures and drives to operate with the same performance as on PowerVault or DAC. When deciding whether to use SAN, PowerVault, or DAC, any additional overhead on the SAN will be important to determine the minimum storage required.

## Performance Recommendations

NetWitness recommends that Packet and Log Decoders receive two LUNs or Block Devices, one for Packet data, the other for all other databases. This allows you to segregate the high-bandwidth Packet Database from the other databases so they do not compete for I/O bandwidth with other activity.

Concentrators require a separate SSD-based index volume for best performance. You must house this index volume on a different RAID group than the Concentrator Meta database volume, which you can stored on NL-SAS. Archivers can use a single large NL-SAS storage volume per appliance.

## Enable Security on SED Capable Drive groups on Host with a mix of SED and NON SED Drives

The encryptSedVd.py may fail to identify the SED Capable Virtual Drives when there is mix of both SED and NON-SED drives on the appliance. The below steps are applicable when both SED and NON-SED capable virtual drives exist on the host.

- SSH to the appliance and enable security on the PERC H740 (mini) Adaptor. The controller number for this adaptor is **0**. The PERC H840 Adaptor is shown as **1**.

To list all the controllers on the appliance:

```
/opt/MegaRAID/perccli/perccli64 show | egrep -A3 'Model'
```

The first column (**Ctl**) lists out the controller index on the appliance. In this case, the controller '**0**' corresponds to '**PERC H740 Mini**' and controller '**1**' corresponds to '**PERC H840 Adaptor**'. The columns '**DGs**' and '**VDs**' displays the virtual drives and drive groups on the controller.

```
[root@116Decoder perccli]# /opt/MegaRAID/perccli/perccli64 show | egrep -A3 'Model'
Ctl Model Ports PDs DGs DN0pt VDs VN0pt BBU sPR DS EHS AS0s Hlth

0 PERCH740PMini 8 10 3 0 3 0 Opt On - N 0 Opt
1 PERCH840Adapter 8 12 1 0 1 0 Opt On - N 0 Opt
[root@116Decoder perccli]#
```

- To enable the security on the 'PERC H740 (mini) Adaptor', for example, Controller '**0**', execute the following command:

```
/opt/MegaRAID/perccli/perccli64 /c0 set securitykey='<SOME_STRING_VALUE>' !'
keyid='< SOME_STRING_VALUE >'
```

Example:

```
/opt/MegaRAID/perccli/perccli64 /c0 set securitykey='Netwitness1!' keyid=1
'Netwitness1' is the securityKey and '1' is ID. Preserve both the Key and
keyID securely.
```

```
[root@116Decoder perccli]# /opt/MegaRAID/perccli/perccli64 /c0 set securitykey='Netwitness1!' keyid='netwitness'
Controller = 0
Status = Success
Description = None

Controller Properties :
=====

Ctrl Method Result

0 set Key Success

```

- Identify the correct Drive group (DG) / Virtual Drive (VD) corresponding to the SED Capable drives that you are trying to enable security.

```
/opt/MegaRAID/perccli/perccli64 /c0 /vall show | egrep -A5 'DG/VD'
```

Refer to first two and last column to identify the correct Drive Group (DG) / Virtual Drive (VD) correspond to the 6 SED enabled drives. On Series 6 appliances, there is only one DG/VD with **RAID6**. '**NAME**' column can be used to identify the VD/DG. In this case, the DG/VD is '**2**'. Using a combination of '**Type**', '**Name**' and '**Size**' columns (these are defined by the user when the VDs are created above).

```
[root@116Decoder perccli]# /opt/MegaRAID/perccli/perccli64 /c0 /vall show | egrep -A5 'DG/VD'
DG/VD TYPE State Access Consist Cache Cac sCC Size Name

0/0 RAID1 Optl RW Yes RWBD - OFF 931.0 GB
1/1 RAID1 Optl RW Yes RWBD - OFF 1.818 TB
2/2 RAID6 Optl RW Yes RWBD - OFF 8.730 TB Virtual Disk 2

[root@116Decoder perccli]#
```

- To turn on Security on the disk group (created out of the 6 SED Capable drives), execute the below command:

```

/opt/MegaRAID/perccli/perccli64 /c0 /d2 set security=on
[root@116Decoder perccli]# /opt/MegaRAID/perccli/perccli64 /c0 /d2 set security=on
Controller = 0
Status = Success
Description = Success

```

5. Get the Enclosure ID (**EID**) using on the controller '0'. In this case, it is '64'

```

/opt/MegaRAID/perccli/perccli64 /c0 /eall show

```

```

[root@116Decoder perccli]# /opt/MegaRAID/perccli/perccli64 /c0 /eall show
Controller = 0
Status = Success
Description = None

```

```

Properties :

```

```

=====

EID State Slots PD PS Fans TSs Alms SIM Port# ProdID VendorSpecific

64 OK 10 10 0 0 0 0 1 00 & 00 x8 BP14G+EXP +

```

```

EID-Enclosure Device ID |PD-Physical drive count |PS-Power Supply count|
TSs-Temperature sensor count |Alms-Alarm count |SIM-SIM Count

```

```

[root@116Decoder perccli]# █

```

6. To confirm that the drives / Drive Groups (DG) are **SED Enabled** and **Secured**, run the below command and verify the **SED Capable**, **Secured**, **SED Enabled** flags are set as 'Yes' for drives in slots 4 (**s4**) through 9 (**s9**).

```

/opt/MegaRAID/perccli/perccli64 /c0 /e64/sall show all | egrep -i '
(Policies/Settings |SED Capable|Secured|SED Enabled) '

```

```

Drive /c0/e64/s0 Policies/Settings :

```

```

SED Capable = No

```

```

SED Enabled = No

```

```

Secured = No

```

```

Drive /c0/e64/s1 Policies/Settings :

```

```

SED Capable = No

```

```

SED Enabled = No

```

```

Secured = No

```

```

Drive /c0/e64/s2 Policies/Settings :

```

```

SED Capable = No

```

```

SED Enabled = No

```

```

Secured = No

```

```

Drive /c0/e64/s3 Policies/Settings :

```

```

SED Capable = No

```

```

SED Enabled = No

```

Secured = No  
Drive /c0/e64/s4 Policies/Settings :  
SED Capable = Yes  
SED Enabled = Yes  
Secured = Yes  
Drive /c0/e64/s5 Policies/Settings :  
SED Capable = Yes  
SED Enabled = Yes  
Secured = Yes  
Drive /c0/e64/s6 Policies/Settings :  
SED Capable = Yes  
SED Enabled = Yes  
Secured = Yes  
Drive /c0/e64/s7 Policies/Settings :  
SED Capable = Yes  
SED Enabled = Yes  
Secured = Yes  
Drive /c0/e64/s8 Policies/Settings :  
SED Capable = Yes  
SED Enabled = Yes  
Secured = Yes  
Drive /c0/e64/s9 Policies/Settings :  
SED Capable = Yes  
SED Enabled = Yes  
Secured = Yes



## Appendix B. Encrypt a Series 6E Core or Hybrid Host (encryptSedVd.py)

NetWitness Series 6E Core and Hybrid hosts have Self-Encrypting Drives (SED). The `encryptSedVd.py` script:

- Validates that the Series 6E host has the correct setup for encryption.
- Encrypts unencrypted drives.

**Note:** For external storage devices such as PowerVault, refer to "[Configure Storage Using the REST API](#)" under "Using the REST API to Configure Storage" for instructions on how to encrypt their SED drives.

The following scenarios are examples of why you would use the `encryptSedVd.py` script.

- You want to know if a physical host has encryption. In this case, if the script determines that the device does not have encryption, it gives you the opportunity to encrypt it.
- You set up a device without encryption and you want to encrypt it.

You will find this script in the `rsa-sa-tools` directory for releases 11.4.0.0 and later. The following directory is for 11.4.0.0.

```
rsa-sa-tools-11.4.0.0-xxxx.noarch.rpm
```

The following procedure illustrates how to use the script.

1. Log in as `root`.
2. Change the directory to the `rsa-sa-tools` RPM base directory:

```
cd /opt/rsa/saTools/supportScript/
```

3. Execute the following command:

```
OWB_ALLOW_NON_FIPS=1 ./encryptSedVd.py
```

The script tells you if the disks are encrypted or not encrypted.

- If the drives are encrypted, the script displays the following message.  
No unencrypted RAID virtual drives with SED physical drives found.
- If the drives are not encrypted, the script identifies the unencrypted drives as shown in the following example.

```
Detected unencrypted RAID Virtual Drives with SED Physical Disks
Please select the drives to encrypt
Navigation: <Tab><Up/Down Arrow> move vertical
<Esc> Quit, <Enter> Save, <Space> Select/Deselect, <A> Select All, <D> Deselect All

 ID VD DG RAID SIZE HBA
() 0 0 0 RAID1 1.1TB PERC H740P Mini
() 0 1 1 RAID1 2.2TB PERC H740P Mini
```

4. If the drives are not encrypted and you want to encrypt them:
  - a. Select the drives you want to encrypt with the space bar and press **Enter**.

The following prompt is displayed.

```

Please enter a passphrase for the PERC H740P Mini security key, minimum length 8 characters, maximum 32
The passphrase must contain a mix of lowercase, uppercase, numeric and non-alphanumeric characters
Optionally enter a key identifier, a default id will be created if not specified

Editing: <Backspace> clear cursor left, <Delete> clear cursor right
Navigation: <Tab><Up/Down Arrow> move vertical, <Left/Right Arrow> move horizontal
<Esc> quit without saving, <Enter> save, trailing spaces are ignored

Enter Passphrase:

Verify Passphrase:

Key ID (optional):


```

- b. In the **Enter Passphrase** text box, type the <passphrase>, for example nFreDaW\$792, and press **Tab**.
- c. In the **Verify Passphrase** text box, re-enter passphrase again for validation.
- d. In the **Key ID (optional)** text box, enter an optional ID string for the security key less than 256 characters or press Enter for none.

The following prompt is displayed.

```

The Passphrase for the security key *Must* be securely backed up in case of PERC adapter hardware
failure and/or replacement, without it the data on all encrypted disks will be unrecoverable.

Entered Passphrase('Quoted'): 'Testing$123'
Entered KeyId('Quoted'): '1'

() I understand the risks and have added the passphrase to my organization's permanent record
<Esc> Cancel, <Y> Acknowledge Backup, <D> Decline Backup, <Enter> Save

```

- e. Select <Y> and press **Enter** to confirm that you added the Passphrase.
- f. Submit the following command string to verify that the SED drives are encrypted.

```
/opt/MegaRAID/perccli/perccli64 /c0 show more
```

The following information is displayed. You can see that all four SED drives are encrypted (that is, Y is displayed for each drive in the SED column).

```

Physical Drives = 4

PD LIST :
=====

EID:SlT DID State DG Size Intf Med SED PI SeSz Model Sp

64:0 0 Onln 0 1.090 TB SAS HDD Y N 512B ST1200MM0069 U
64:1 1 Onln 0 1.090 TB SAS HDD Y N 512B ST1200MM0069 U
64:2 2 Onln 1 2.182 TB SAS HDD Y N 512B ST2400MM0149 U
64:3 3 Onln 1 2.182 TB SAS HDD Y N 512B ST2400MM0149 U

```

**Note:** The SED Enabled and Secured label values are set to Yes, if the drives are SED enabled and secured.

To check the drives on controller 0 and enclosure 247 use the below command:

```
/opt/MegaRAID/percccli/percccli64 /c1 /e247/sall show all | egrep -i '
(Policies/Settings|SED Capable|Secured|SED Enabled) '
```

You will find detailed information on percccli commands in the Dell PowerEdge RAID Controller CLI Reference Guide ([http://14u-00.jinr.ru/pub/misc/h-w/LSI/dell-sas-hba-12gbps\\_reference\\_guide\\_en-us.pdf](http://14u-00.jinr.ru/pub/misc/h-w/LSI/dell-sas-hba-12gbps_reference_guide_en-us.pdf)).

## Enable SED on configured Drive Groups

Virtual Drives configured are SED Capable but are NOT SED Enabled.

To enable virtual drives or drive groups using PERC H840 Adaptors (External storage):

1. SSH to the appliance and run the below script to encrypt the virtual drive (on external storage).

**Note:** The encryptSedVd.py script turn on the SED feature only on Virtual Drives or Drive Groups on the PERC H840 Adaptors (external storage) and NOT on PERC H740 mini. Refer to [Enable Virtual Drives / Drive Groups - PERC H740 \(Mini\) Adaptors \(Internal storage\)](#) to enable SED on PERC H740 Mini .

```
OWB_ALLOW_NON_FIPS=true /opt/rsa/saTools/supportScript/encryptSedVd.py
```

2. Select the Virtual Drive and press **Enter**.  
Passphrase screen is displayed.
3. Enter the Passphrase and press **Enter**.  
For Example,  
Passphrase : **Netwitness1!**

**keyID: netwitness**

```
Please enter a passphrase for the PERC H840 Adapter security key, minimum length 8 characters, maximum 32
The passphrase must contain a mix of lowercase, uppercase, numeric and non-alphanumeric characters
Optionally enter a key identifier, a default id will be created if not specified

Editing: <Backspace> clear cursor left, <Delete> clear cursor right
Navigation: <Tab><Up/Down Arrow> move vertical, <Left/Right Arrow> move horizontal
<Esc> quit without saving, <Enter> save, trailing spaces are ignored

Enter Passphrase:
Netwitness!

Verify Passphrase:
Netwitness!

Key ID (optional):
netwitness
```

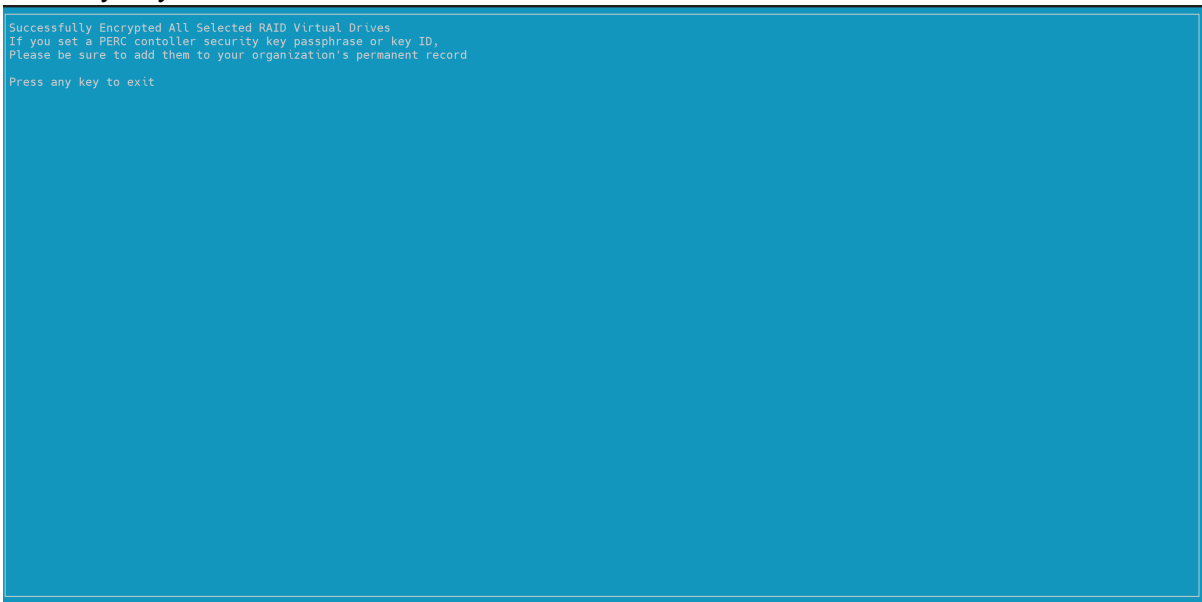
**4. Acknowledge the message and Press **Enter** to Save.**

```
The Passphrase for the security key *Must* be securely backed up in case of PERC adapter hardware
failure and/or replacement, without it the data on all encrypted disks will be unrecoverable.

Entered Passphrase('Quoted'): 'Netwitness!'
Entered KeyId('Quoted'): 'netwitness'

() I understand the risks and have added the passphrase to my organization's permanent record
<Esc> Cancel, <Y> Acknowledge Backup, <D> Decline Backup, <Enter> Save
```

### 5. Press any Key to Exit.



### 6. To confirm that the drives are SED Enabled and secured, run the following command and verify the SED Enabled and Secured returns Yes.

```
/opt/MegaRAID/perccli/perccli64 /c1 /e247/sall show all | egrep -i '
(Policies/Settings|SED Capable|Secured|SED Enabled) '
```

```
Drive /c1/e247/s0 Policies/Settings :
```

```
SED Capable = Yes
```

```
SED Enabled = Yes
```

```
Secured = Yes
```

```
Drive /c1/e247/s1 Policies/Settings :
```

```
SED Capable = Yes
```

```
SED Enabled = Yes
```

```
Secured = Yes
```

```
Drive /c1/e247/s2 Policies/Settings :
```

```
SED Capable = Yes
```

```
SED Enabled = Yes
```

```
Secured = Yes
```

```
Drive /c1/e247/s3 Policies/Settings :
```

```
SED Capable = Yes
```

```
SED Enabled = Yes
```

```
Secured = Yes
```

```
Drive /c1/e247/s4 Policies/Settings :
```

```
SED Capable = Yes
```

```
SED Enabled = Yes
```

```
Secured = Yes
```

```
Drive /c1/e247/s5 Policies/Settings :
```

```
SED Capable = Yes
```

```

SED Enabled = Yes
Secured = Yes
Drive /c1/e247/s6 Policies/Settings :
SED Capable = Yes
SED Enabled = Yes
Secured = Yes
Drive /c1/e247/s7 Policies/Settings :
SED Capable = Yes
SED Enabled = Yes
Secured = Yes
Drive /c1/e247/s8 Policies/Settings :
SED Capable = Yes
SED Enabled = Yes
Secured = Yes
Drive /c1/e247/s9 Policies/Settings :
SED Capable = Yes
SED Enabled = Yes
Secured = Yes
Drive /c1/e247/s10 Policies/Settings :
SED Capable = Yes
SED Enabled = Yes
Secured = Yes
Drive /c1/e247/s11 Policies/Settings :
SED Capable = Yes
SED Enabled = Yes
Secured = Yes

```

## Enable Virtual Drives / Drive Groups - PERC H740 (Mini) Adaptors (Internal storage)

You can enable the SED capability on the Virtual Drive or Drive Groups created out of on-board SED capable drives (in slots 4 through 9 – total of 6 drives) using the `percli64` utility. You cannot use `/opt/rsa/saTools/supportScript/encryptSedVd.py` to turn on Security on the Virtual drives on the PERC H740 (mini) Adaptor.

1. SSH to the appliance and enable security on the PERC H740 (mini) Adaptor. The controller number for this adaptor is **0**. The PERC H840 Adaptor is shown as **1**.

To list all the controllers on the appliance run the following command:

```
/opt/MegaRAID/perccli/perccli64 show | egrep -A3 'Model'
```

The first column (**Ctl**) lists out the controller index on the appliance. In this case, the controller **0** corresponds to **PERC H740 Mini** and controller **1** corresponds to **PERC H840 Adaptor**. The columns **DGs** and **VDs** displays the virtual drives and drive groups on the controller.

2. To enable the security on the **PERC H740 (mini) Adaptor**, for example, Controller **0**, run the following command:

```
/opt/MegaRAID/perccli/perccli64 /c0 set securitykey='<String>'!
keyid='<String>'
```

Example:

```
/opt/MegaRAID/perccli/perccli64 /c0 set securitykey='Netwitness1!'
keyid='netwitness'
```

'Netwitness1' is the securityKey and 'netwitness' is ID.

Make a note of both the Key and keyID securely.

3. Identify the correct Drive group (DG) or Virtual Drive (VD) corresponding to the SED Capable drives that you want to enable security.

```
/opt/MegaRAID/perccli/perccli64 /c0 /vall show | egrep -A5 'DG/VD'
```

Check the first two and last column to identify the correct Drive Group / Virtual Drive correspond to the 6 SED enabled drives that are SED Capable. On Series 6 appliances, there is only one DG or VD with **RAID6** type. Name column can be used to identify the VD or DG. In this case, the DG or VD is **2**. Using a combination of **Type**, **Name** and **Size** columns (these are defined when you created VDs above).

4. To turn on Security on the disk group (created out of the 6 SED Capable drives) for **decodersmall** volume group, run the below command:

```
/opt/MegaRAID/perccli/perccli64 /c0 /d2 set security=on
```

5. Get the Enclosure ID (**EID**) using on the controller **0**. In this case, it is **64**

```
/opt/MegaRAID/perccli/perccli64 /c0 /eall show
```

6. To confirm that the drives or drive groups are **SED Enabled** and **Secured**, run the below command and verify the **SED Capable**, **Secured**, **SED Enabled** flags are set as **Yes** for drives in slots 4 (s4) through 9 (s9).

```
/opt/MegaRAID/perccli/perccli64 /c0 /e64/sall show all | egrep -i '
(Policies/Settings |SED Capable|Secured|SED Enabled)'
```

Drive /c0/e64/s0 Policies/Settings :

SED Capable = No

SED Enabled = No

Secured = No

Drive /c0/e64/s1 Policies/Settings :

SED Capable = No

SED Enabled = No

Secured = No

Drive /c0/e64/s2 Policies/Settings :

SED Capable = No

SED Enabled = No

Secured = No

Drive /c0/e64/s3 Policies/Settings :

---

SED Capable = No  
SED Enabled = No  
Secured = No  
Drive /c0/e64/s4 Policies/Settings :  
SED Capable = Yes  
SED Enabled = Yes  
Secured = Yes  
Drive /c0/e64/s5 Policies/Settings :  
SED Capable = Yes  
SED Enabled = Yes  
Secured = Yes  
Drive /c0/e64/s6 Policies/Settings :  
SED Capable = Yes  
SED Enabled = Yes  
Secured = Yes  
Drive /c0/e64/s7 Policies/Settings :  
SED Capable = Yes  
SED Enabled = Yes  
Secured = Yes  
Drive /c0/e64/s8 Policies/Settings :  
SED Capable = Yes  
SED Enabled = Yes  
Secured = Yes  
Drive /c0/e64/s9 Policies/Settings :  
SED Capable = Yes  
SED Enabled = Yes  
Secured = Yes

## Enable SED on configured Virtual Drives/ Drive Groups on Power Vault (PERC 840)

### Enable Virtual Drives / Drive Groups - PERC H840 Adaptors

**Note:** The virtual disk created in *Configure Block Devices for PowerVaults* section in [Prepare Physical Storage](#) is SED capable but NOT SED Enabled.



1. To enable, SSH into the appliance and run the below script to encrypt the VD (on external storage).  
OWB\_ALLOW\_NON\_FIPS=true /opt/rsa/saTools/supportScript/encryptSedVd.py

**Note:** The encryptSedVd.py script turn on the SED feature only on Virtual Drives or Drive Groups on the PERC H840 Adaptors (external storage) and NOT on PERC H740 mini. Refer to **Enable Virtual Drives / Drive Groups - PERC H740 (Mini) Adaptors (Internal storage)** to enable SED on PERC H740 Mini

```
OWB_ALLOW_NON_FIPS=true /opt/rsa/saTools/supportScript/encryptSedVd.py
```

```
Detected unencrypted RAID Virtual Drives with SED Physical Disks
Please select the drives to encrypt
Navigation: <Tab><Up/Down Arrow> move vertical
<Esc> Quit, <Enter> Save, <Space> Select/Deselect, <A> Select All, <D> Deselect All

 ID VD DG RAID SIZE HBA
 (X) 1 0 0 RAID6 106.9TB PERC H840 Adapter
```

2. Select both the **Virtula Disks** and press **Enter**.  
The Passphrase screen is displayed.

```
Please enter a passphrase for the PERC H840 Adapter security key, minimum length 8 characters, maximum 32
The passphrase must contain a mix of lowercase, uppercase, numeric and non-alphanumeric characters
Optionally enter a key identifier, a default id will be created if not specified

Editing: <Backspace> clear cursor left, <Delete> clear cursor right
Navigation: <Tab><Up/Down Arrow> move vertical, <Left/Right Arrow> move horizontal
<Esc> quit without saving, <Enter> save, trailing spaces are ignored

Enter Passphrase:

Verify Passphrase:

Key ID (optional):

```

3. Enter the Passphrase and press **Enter**.

For example,

Passphrase : **Netwitness1!**

keyID: **netwitness**

```
Please enter a passphrase for the PERC H840 Adapter security key, minimum length 8 characters, maximum 32
The passphrase must contain a mix of lowercase, uppercase, numeric and non-alphanumeric characters
Optionally enter a key identifier, a default id will be created if not specified

Editing: <Backspace> clear cursor left, <Delete> clear cursor right
Navigation: <Tab><Up/Down Arrow> move vertical, <Left/Right Arrow> move horizontal
<Esc> quit without saving, <Enter> save, trailing spaces are ignored

Enter Passphrase:
Netwitness1!

Verify Passphrase:
Netwitness1!

Key ID (optional):
netwitness
```

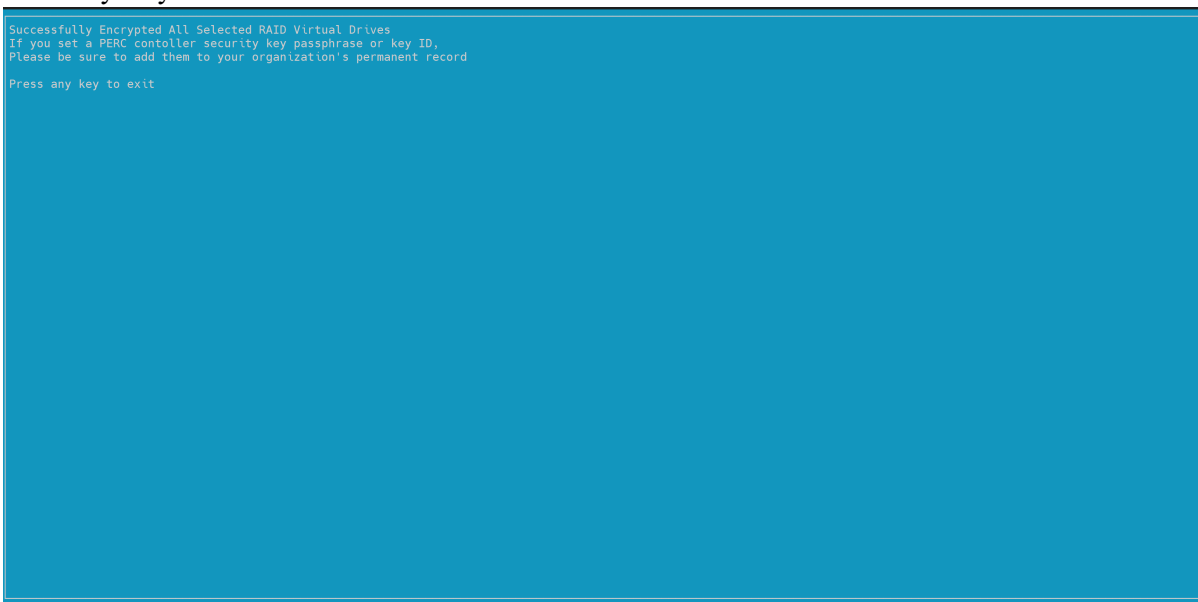
4. Acknowledge the message and Press **Enter** to Save.

```
The Passphrase for the security key *Must* be securely backed up in case of PERC adapter hardware
failure and/or replacement, without it the data on all encrypted disks will be unrecoverable.

Entered Passphrase('Quoted'): 'Netwitness1!'
Entered KeyId('Quoted'): 'netwitness'

() I understand the risks and have added the passphrase to my organization's permanent record
<Esc> Cancel, <Y> Acknowledge Backup, <D> Decline Backup, <Enter> Save
```

### 5. Press any Key to Exit.



### 6. To confirm that the drives are SED Enabled and secured, run the below command and verify the SED Enabled and Secured returns Yes.

```
/opt/MegaRAID/perccli/perccli64 /c1 /e247/sall show all | egrep -i '(Policies/Settings|SED Capable|Secured|SED Enabled)'
```

Drive /c1/e247/s0 Policies/Settings :

SED Capable = Yes

SED Enabled = Yes

Secured = Yes

Drive /c1/e247/s1 Policies/Settings :

SED Capable = Yes

SED Enabled = Yes

Secured = Yes

Drive /c1/e247/s2 Policies/Settings :

SED Capable = Yes

SED Enabled = Yes

Secured = Yes

Drive /c1/e247/s3 Policies/Settings :

SED Capable = Yes

SED Enabled = Yes

Secured = Yes

Drive /c1/e247/s4 Policies/Settings :

SED Capable = Yes

SED Enabled = Yes

```
Secured = Yes
Drive /c1/e247/s5 Policies/Settings :
SED Capable = Yes
SED Enabled = Yes
Secured = Yes
Drive /c1/e247/s6 Policies/Settings :
SED Capable = Yes
SED Enabled = Yes
Secured = Yes
Drive /c1/e247/s7 Policies/Settings :
SED Capable = Yes
SED Enabled = Yes
Secured = Yes
Drive /c1/e247/s8 Policies/Settings :
SED Capable = Yes
SED Enabled = Yes
Secured = Yes
Drive /c1/e247/s9 Policies/Settings :
SED Capable = Yes
SED Enabled = Yes
Secured = Yes
Drive /c1/e247/s10 Policies/Settings :
SED Capable = Yes
SED Enabled = Yes
Secured = Yes
Drive /c1/e247/s11 Policies/Settings :
SED Capable = Yes
SED Enabled = Yes
Secured = Yes
```

## Enable Security on SED Capable Drive groups on Host with a mix of SED and NON SED Drives

The `encryptSedVd.py` may fail to identify the SED Capable Virtual Drives when there is mix of both SED and NON-SED drives on the appliance. The below steps are applicable when both SED and NON-SED capable virtual drives exist on the host.

- SSH to the appliance and enable security on the PERC H740 (mini) Adaptor. The controller number for this adaptor is **0**. The PERC H840 Adaptor is shown as **1**.

To list all the controllers on the appliance:

```
/opt/MegaRAID/perccli/perccli64 show | egrep -A3 'Model'
```

The first column (**Ctl**) lists out the controller index on the appliance. In this case, the controller '**0**' corresponds to '**PERC H740 Mini**' and controller '**1**' corresponds to '**PERC H840 Adaptor**'. The columns '**DGs**' and '**VDs**' displays the drive groups and virtual drives on the controller.

```
[root@116Decoder perccli]# /opt/MegaRAID/perccli/perccli64 show | egrep -A3 'Model'
Ctl Model Ports Pds Dgs DNOpt Vds VNOpt BBU sPR DS EHS ASOs Hlth

0 PERCH740PMini 8 10 3 0 3 0 Opt On - N 0 Opt
1 PERCH840Adapter 8 12 1 0 1 0 Opt On - N 0 Opt
[root@116Decoder perccli]#
```

- To enable the security on the 'PERC H740 (mini) Adaptor' i.e Controller '**0**', execute the following command:

```
/opt/MegaRAID/perccli/perccli64 /c0 set securitykey='<SOME_STRING_VALUE>'!
keyid='< SOME_STRING_VALUE >'
```

Example:

```
/opt/MegaRAID/perccli/perccli64 /c0 set securitykey='Netwitness1!' keyid=1
```

'Netwitness1' is the securityKey and '1' is ID. Preserve both the Key and keyID securely.

```
[root@116Decoder perccli]# /opt/MegaRAID/perccli/perccli64 /c0 set securitykey='Netwitness1!' keyid='netwitness'
Controller = 0
Status = Success
Description = None

Controller Properties :
=====
```

| Ctrl | Method  | Result  |
|------|---------|---------|
| 0    | set Key | Success |

```

```

- Identify the correct Drive group (DG) / Virtual Drive (VD) corresponding to the SED Capable drives that we are trying to enable security.

```
/opt/MegaRAID/perccli/perccli64 /c0 /vall show | egrep -A5 'DG/VD'
```

Refer to first two and last column to identify the correct Drive Group (DG) / Virtual Drive (VD) correspond to the 6 SED enabled drives. On Series 6 appliances, there is only one DG/VD with **RAID6**. '**Name**' column can be used to identify the VD/DG. In this case, the DG/VD is '**2**'. Using a combination of '**Type**', '**Name**' and '**Size**' columns (these were defined by the user when the VDs are created above).

```
[root@116Decoder perccli]# /opt/MegaRAID/perccli/perccli64 /c0 /vall show | egrep -A5 'DG/VD'
DG/VD TYPE State Access Consist Cache Cac sCC Size Name

0/0 RAID1 Optl RW Yes RWBD - OFF 931.0 GB
1/1 RAID1 Optl RW Yes RWBD - OFF 1.818 TB
2/2 RAID6 Optl RW Yes RWBD - OFF 8.730 TB Virtual Disk 2

[root@116Decoder perccli]#
```

- To turn on Security on the disk group (created out of the 6 SED Capable drives), execute the below command:

```
/opt/MegaRAID/perccli/perccli64 /c0 /d2 set security=on
[root@116Decoder perccli]# /opt/MegaRAID/perccli/perccli64 /c0 /d2 set security=on
Controller = 0
Status = Success
Description = Success
```

- Get the Enclosure ID (EID) using on the controller '0'. In this case, it is '64'

```
/opt/MegaRAID/perccli/perccli64 /c0 /eall show
```

```
[root@116Decoder perccli]# /opt/MegaRAID/perccli/perccli64 /c0 /eall show
Controller = 0
Status = Success
Description = None
```

```
Properties :
```

```
=====
```

```

EID State Slots PD PS Fans TSs ALms SIM Port# ProdID VendorSpecific

64 OK 10 10 0 0 0 0 1 00 & 00 x8 BP14G+EXP +

```

```
EID-Enclosure Device ID |PD-Physical drive count |PS-Power Supply count|
TSs-Temperature sensor count |ALms-Alarm count |SIM-SIM Count
```

```
[root@116Decoder perccli]#
```

- To confirm that the drives / Drive Groups (DG) are **SED Enabled** and **Secured**, run the below command and verify the **SED Capable**, **Secured**, **SED Enabled** flags are set as 'Yes' for drives in slots 4 (s4) through 9 (s9).

```
/opt/MegaRAID/perccli/perccli64 /c0 /e64/sall show all | egrep -i '
(Policies/Settings |SED Capable|Secured|SED Enabled)'
```

```
Drive /c0/e64/s0 Policies/Settings :
```

```
SED Capable = No
```

```
SED Enabled = No
```

```
Secured = No
```

```
Drive /c0/e64/s1 Policies/Settings :
```

```
SED Capable = No
```

```
SED Enabled = No
```

```
Secured = No
```

```
Drive /c0/e64/s2 Policies/Settings :
```

```
SED Capable = No
```

```
SED Enabled = No
```

```
Secured = No
```

Drive /c0/e64/s3 Policies/Settings :

SED Capable = No

SED Enabled = No

Secured = No

Drive /c0/e64/s4 Policies/Settings :

SED Capable = Yes

SED Enabled = Yes

Secured = Yes

Drive /c0/e64/s5 Policies/Settings :

SED Capable = Yes

SED Enabled = Yes

Secured = Yes

Drive /c0/e64/s6 Policies/Settings :

SED Capable = Yes

SED Enabled = Yes

Secured = Yes

Drive /c0/e64/s7 Policies/Settings :

SED Capable = Yes

SED Enabled = Yes

Secured = Yes

Drive /c0/e64/s8 Policies/Settings :

SED Capable = Yes

SED Enabled = Yes

Secured = Yes

Drive /c0/e64/s9 Policies/Settings :

SED Capable = Yes

SED Enabled = Yes

Secured = Yes

---

## Appendix C. Troubleshooting

---

This section contains instructions on how to resolve various storage tasks using the REST API.

### Reconfigure Pre-Configured DAC Attached to Decoder Using REST API

This scenario covers how to reconfigure a DAC using the REST API that was configured using another tool and clear any pre-existing data (if no longer need or backed up to another storage device).

The following information describes the state of the host and storage hardware prior to the attempt to reconfigure the storage devices using the REST API.

When the DAC was added, it had old data and was configured (but not using the REST API). This prevented the REST API from executing the `raidNew` command and returned the `Physical disk does not have appropriate attributes` error message.

The following steps describe the scenario and with its resolution.

1. From the Decoder Linux console (or SSH to Decoder), submitted the following command string.  

```
/opt/MegaRAID/perccli/perccli64 /c2/fall del
```

You will find detailed information on `perccli` commands in the **Dell PowerEdge RAID Controller CLI Reference Guide** ([https://topics-cdn.dell.com/pdf/dell-sas-hba-12gbps\\_reference-guide\\_en-us.pdf](https://topics-cdn.dell.com/pdf/dell-sas-hba-12gbps_reference-guide_en-us.pdf)).

This deleted all foreign configuration from controller 2 and cleared all data from the DAC.
2. Tried to partition the DAC, but the `partNew` command failed because that information was already defined on the DAC. `partNew` displayed that you must use one an available device, but `devList` showed it in use.
3. Assuming that the partitions were defined, tried to allocate the storage devices, but this did not work because the DAC was not mounted.
4. Tried to mount the DAC from the command line, but received `mount failed: structure needs to be cleaned` error message.
5. There was no data that needed to be preserved on the DAC, so submitted the following command strings to clean the structure.  

```
mkfs.xfs -f /dev/decoder0/packetdb
mkfs.xfs -f /dev/decoder1/packetdb
```
6. Mounted devices to their appropriate locations in `/var/netwitness/decoder`.
7. Completed the remainder of the applicable steps as described in [Configure Storage Using the REST API](#) to reconfigure the DAC



## Appendix D. Sample Storage Configuration Scenarios for 15-Drive DACs

This appendix illustrates the following example of how to configure storage on two non-encrypted 15-drive DAC external storage devices.

- [Configure Storage for Archiver](#)
- [Configure Storage for Network \(Packet\) Decoder](#)
- [Configure Storage for Network Concentrator](#)
- [Configure Storage for Log Decoder Hybrid](#)

### Configure Storage for Archiver

The following scenario configures storage on one, non-encrypted, 15-Drive DAC for an Archiver physical host.

1. Execute the `raidList` command.
  - a. Record the Controller Number, Enclosure Number, In Use, Drives, and Devices.  
You should see the following information.  
In Use: FALSE  
Devices: <empty>
  - b. Verify the Drive Count, Size, and Vendor.  
The following example illustrates what you should see before you create a RAID array.

The screenshot shows a web-based configuration interface. On the left is a navigation tree with folders like 'archiver', 'connections', 'deviceappliance', and 'appliance'. The main area displays the 'raidList' command output for a host named 'NWHOST2100'. The output is as follows:

```

Controller 0, Enclosure 32
Vendor: DP
Model: BP13G+EXP
In Use: true
Drives: 931.511 GB x 2
1.818 TB x 2
Devices: sda
sdb

Controller 1, Enclosure 0
Vendor: EMC
Model: ESES Enclosure
In Use: false
Drives: 3.637 TB x 15
Devices:

```

2. Execute the `raidNew` command with the following parameters using the controller number and the enclosure number you just recorded.  
`controller=1 enclosure=0 scheme=archiver commit=1`

The following example illustrates what you should see after you create a RAID array.

Properties for NWHOST2100 – Archiver (ARCHIVER)/device/appliance/appliance.

raidNew Parameters controller=1 enclosure=0 scheme=archiver commit=1

Message Help

enclosure - <uint32, (enum-one:32,0)> Enclosure number of the shelf to clear  
 scheme - <string, (enum-one:decoder|logdecoder|concentrator|archiver|network-hybrid|log-hybrid)> Type of RAID volumes to allocate  
 preferSecure - <bool, optional, (bool:0,1,yes,no,true,false,on,off)> Prefer creation of a secure array given compatible physical drives and a controller with a security key set  
 commit - <bool, optional> commit changes

Response Output

```
/opt/MegaRAID/perccli/perccli64 /c1 add vd r6 drives=0:0,0:1,0:2,0:3,0:4,0:5,0:6,0:7,0:8,0:9,0:10,0:11,0:12,0:13,0:14 wb,ra,cached Strip=128
Controller = 1
Status = Success
Description = Add VD Succeeded
```

- Execute the `raidList` command to verify the new RAID array.

You should now see the following information.

In Use: TRUE

Devices: <device> (for example, `sdc`)

Properties for NWHOST2100 – Archiver (ARCHIVER)/device/appliance/appliance.

raidList Parameters

Message Help

list drive shelves attached to this appliance  
 security.roles: appliance.manage

Response Output

```
Controller 0, Enclosure 32
Vendor: DP
Model: BP13G+EXP
In Use: true
Drives: 931.511 GB x 2
 1.818 TB x 2
Devices: sda
 sdb

Controller 1, Enclosure 0
Vendor: EMC
Model: ESES Enclosure
In Use: true
Drives: 3.637 TB x 15
Devices: sdc
```

- Execute the `partNew` command with the following parameters to create partitions and mount points in the `etc/fstab` file.  
`name=<device>` (for example, `sdc`) `service=archiver` `volume=archiver` `commit=1`
- Execute the `srvAlloc` command with the following parameters to allocate the space to the archiver service. This adds storage to the archiver service configuration and restarts the service every time it is executed.  
`service=archiver` `volume=archiver0` `commit=1`

Properties for NWHOST2100 – Archiver (ARCHIVER)/deviceappliance/appliance.

```

srvAlloc Parameters service=archiver volume=archiver0 commit=1
Message Help
service - <string, {enum-one:archiver|concentrator|decoder|logdecoder}> service that will use storage
volume - <string, {enum-one:archiver0|netwitness_vg00}> volume group name
commit - <bool, optional> commit changes

```

Change Service | NWHOST2100 - Archiver | System

Start Aggregation Stop Aggregation Host Tasks Shutdown Service

### Archiver Service Information

Name NWHOST2100 (Archiver)  
 Version 11.3.0.0 (Rev null)  
 Memory Usage 30016 KB (0.02% of 126 GB)  
 CPU 0%  
 Running Since 2019-Jun-12 13:12:17  
 Uptime 1 minute 10 seconds  
 Current Time 2019-Jun-12 13:13:27

6. Confirm the “Hot Storage” in “Data Retention”.

Change Service | NWHOST2100 - Archiver | Config

General **Data Retention** Files Appliance Service Configuration

Configure the rollover criteria for removing database records from primary storage using an age-based threshold, and schedule the timing for checking if t

1. Configure hot, warm and cold storage
2. Configure collections
3. Define retention rules

Total Hot Storage **47.29 TB** ⚙️ Total Warm Storage Not Configured ⚙️ Cold Storage Not Configured ⚙️

1 Mount Point

#### Collections

| Collection           | Usage / Hot Storage   | Usage / Warm Storage | Cold Storage | Retention |
|----------------------|-----------------------|----------------------|--------------|-----------|
| default              | 0 B / 44.93 TB (95%)  | Disabled             | ○            | No Limit  |
| <b>Total Storage</b> | <b>0 B / 44.93 TB</b> | <b>0 B / 0 B</b>     |              |           |

#### Retention Rules

| Order ^ | Rule Name | Condition |
|---------|-----------|-----------|
|         | default   | *         |

7. Reconfigure the following Archiver service to detect and take advantage of all of the free space as described in [Task 5 - \(Optional\) Reconfigure Storage Configuration for 10G Capture](#).

## Configure Storage for Network (Packet) Decoder

The following scenario configures storage on two, non-encrypted, 15-Drive DACs for a Network Decoder for 10G Capture physical host.

1. Execute the `raidList` command.
  - a. Record the Controller Number, Enclosure Number, In Use, Drives, and Devices.

You should see the following information.

In Use: FALSE

Devices: <empty>

- b. Verify the Drive Count, Size, and Vendor.

The following example illustrates what you should see before you create a RAID array.

```
Properties for NWHOST2100 -
raidList Parameters
Message Help
list drive shelves attached to this appliance
security.roles: appliance.manage

Response Output
Drives: 931.511 GB x 2
1.818 TB x 2
Devices: sda
sdb

Controller 1, Enclosure 0
Vendor: EMC
Model: ESES Enclosure
In Use: false
Drives: 3.637 TB x 15
Devices:

Controller 1, Enclosure 2
Vendor: EMC
Model: ESES Enclosure
In Use: false
Drives: 3.637 TB x 15
Devices:
```

2. Execute the `raidNew` command with the following parameters using the controller number and the enclosure number you just recorded.
  - Parameters for the first enclosure:
 

```
controller=1 enclosure=0 scheme=decoder commit=1
```

raidNew  Parameters controller=1 enclosure=0 scheme=decoder commit=1

Message Help

parameters:  
controller - <int32, {enum-one:0,1}> Controller the shelf is attached to  
enclosure - <uint32, {enum-one:32,0,2}> Enclosure number of the shelf to clear  
scheme - <string, {enum-one:decoder|logdecoder|concentrator|archiver|network-hybrid|log-hybrid}> Type of RAID volumes to allocate

Response Output

```
/opt/MegaRAID/perccli/perccli64 /c1 add vd r5 drives=0:0:0:1,0:2 wb ra cached Strip=128
Controller = 1
Status = Success
Description = Add VD Succeeded
```

```
/opt/MegaRAID/perccli/perccli64 /c1 add vd r5 drives=0:3,0:4,0:5,0:6,0:7,0:8,0:9,0:10,0:11,0:12,0:13,0:14 wb ra cached Strip=128
Controller = 1
Status = Success
Description = Add VD Succeeded
```

- Parameters for the second enclosure:

```
controller=1 enclosure=2 scheme=decoder commit=1
```

**raidList** Parameters

**Message Help**

list drive shelves attached to this appliance  
security.roles: appliance.manage

**Response Output**

```
Devices: sda
 sdb

Controller 1, Enclosure 0
Vendor: EMC
Model: ESES Enclosure
In Use: true
Drives: 3.637 TB x 15
Devices: sdc
 sdd

Controller 1, Enclosure 2
Vendor: EMC
Model: ESES Enclosure
In Use: true
Drives: 3.637 TB x 15
Devices: sde
 sdf
```

- Use the `raidList` command to display block devices for enclosures so you can verify `In Use: TRUE`.

4. SSH to the Network Decoder and use the `lsblk` command to confirm sizes for **decodersmall**.

```
[root@NWHOST2000 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 931G 0 disk
├─sda1 8:1 0 1G 0 part /boot
├─sda2 8:2 0 930G 0 part
│ ├─netwitness_vg00-root 253:0 0 30G 0 lvm /
│ ├─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
│ ├─netwitness_vg00-nwhome 253:2 0 2.7T 0 lvm /var/netwitness
│ ├─netwitness_vg00-varlog 253:3 0 10G 0 lvm /var/log
│ └─netwitness_vg00-usrhome 253:4 0 10G 0 lvm /home
sdb 8:16 0 1.8T 0 disk
├─sdb1 8:17 0 1.8T 0 part
│ └─netwitness_vg00-nwhome 253:2 0 2.7T 0 lvm /var/netwitness
sdc 8:32 0 7.3T 0 disk
sdd 8:48 0 40T 0 disk
sde 8:64 0 7.3T 0 disk
sdf 8:80 0 40T 0 disk
```

**Note:** For RAID configuration, when you use the decoder for 10G Capture you use **decoder** for both enclosures for performance reasons. When you do not use the **decoder** for 10G Capture, you use the decoder and archiver for the enclosures to maximize storage for because the second enclosure is a single RAID under the **archiver** configuration.

5. Execute the `partNew` command to create the **decodersmall** partition first (decoder dir, index, metadb, sessiondb) (First Enclosure, SDC, SDD) with the following parameters.  
`name=sdc service=decoder volume=decodersmall commit=1`

```
partNew Parameters name=sdcsdd service=decoder volume=decodersmall commit=1
```

Message Help

```
name - <string, {enum-one:sdcsdd,sde,sdf}> block device name
service - <string, {enum-one:archiver|concentrator|decoder|logdecoder}> service that will use storage
volume - <string, optional, {enum-one:index|concentrator|decodersmall|decoder|logdecodersmall|logdecoder|archiver}> volume to create
commit - <bool, optional> commit changes
```

## Response Output

```
Logical volume "decoroot" created.
/sbin/mkfs.xfs /dev/decodersmall/decoroot
meta-data=/dev/decodersmall/decoroot isize=512 agcount=4, agsize=655360 blks
= sectsz=512 attr=2, projid32bit=1
= crc=1 finobt=0, sparse=0
data = bsize=4096 blocks=2621440, imaxpct=25
= sunit=0 swidth=0 blks
naming =version 2 bsize=4096 ascii-ci=0 ftype=1
log =internal log bsize=4096 blocks=2560, version=2
= sectsz=512 sunit=0 blks, lazy-count=1
realtime =none extsz=4096 blocks=0, rtextents=0
/bin/mkdir -p /var/netwitness/decoder
/bin/mount /var/netwitness/decoder
/sbin/lvcreate -y -n index -L 30G decodersmall
Logical volume "index" created.
/sbin/mkfs.xfs /dev/decodersmall/index
meta-data=/dev/decodersmall/index isize=512 agcount=4, agsize=1966080 blks
= sectsz=512 attr=2, projid32bit=1
= crc=1 finobt=0, sparse=0
data = bsize=4096 blocks=7864320, imaxpct=25
```

```
[root@NWHOST2000 ~]# df -h
Filesystem Size Used Avail Use% Mounted on
/dev/mapper/netwitness_vg00-root 30G 2.5G 28G 9% /
devtmpfs 63G 0 63G 0% /dev
tmpfs 63G 12K 63G 1% /dev/shm
tmpfs 63G 26M 63G 1% /run
tmpfs 63G 0 63G 0% /sys/fs/cgroup
/dev/mapper/netwitness_vg00-nwhome 2.7T 98M 2.7T 1% /var/netwitness
/dev/mapper/netwitness_vg00-varlog 10G 49M 10G 1% /var/log
/dev/mapper/netwitness_vg00-usrhome 10G 33M 10G 1% /home
/dev/sda1 1014M 88M 927M 9% /boot
tmpfs 13G 0 13G 0% /run/user/0
/dev/mapper/decodersmall-decoroot 10G 33M 10G 1% /var/netwitness/decoder
/dev/mapper/decodersmall-index 30G 33M 30G 1% /var/netwitness/decoder/index
/dev/mapper/decodersmall-sessiondb 600G 33M 600G 1% /var/netwitness/decoder/sessiondb
/dev/mapper/decodersmall-metadb 6.7T 33M 6.7T 1% /var/netwitness/decoder/metadb
[root@NWHOST2000 ~]#
```

- Execute the `partNew` command to create the decoder volume (packetdb) (First Enclosure, SDC, SDD) with the following parameters.

```
name==sdd service=decoder volume=decoder commit=1
```



partNew  Parameters name=sdd service=decoder volume=decoder commit=1

Message Help

name - <string, {enum-one:sdc,sdd,sde,sdf}> block device name  
 service - <string, {enum-one:archiver|concentrator|decoder|logdecoder}> service that will use storage  
 volume - <string, optional, {enum-one:index|concentrator|decodersmall|decoder|logdecodersmall|logdecoder|archiver}> volume to create  
 commit - <bool, optional> commit changes

## Response Output

```
/sbin/parted -s /dev/sdd mlabel gpt
/sbin/parted -s -a optimal /dev/sdd mkpart LVM 0% 100%
/sbin/pvcreate -f /dev/sdd1
Physical volume "/dev/sdd1" successfully created.
/sbin/vgcreate -f decoder /dev/sdd1
Volume group "decoder" successfully created
/sbin/lvcreate -y -n packetdb -l 100%FREE decoder
Logical volume "packetdb" created.
/sbin/mkfs.xfs /dev/decoder/packetdb
meta-data=/dev/decoder/packetdb isize=512 agcount=41, agsize=268435455 blks
= sectsz=512 attr=2, projid32bit=1
= crc=1 finobt=0, sparse=0
data = bsize=4096 blocks=10742791168, imaxpct=5
= sunit=0 swidth=0 blks
naming =version 2 bsize=4096 ascii-ci=0 ftype=1
log =internal log bsize=4096 blocks=521728, version=2
= sectsz=512 sunit=0 blks, lazy-count=1
realtime =none extsz=4096 blocks=0, rtextents=0
/bin/mkdir -p /var/netwitness/decoder/packetdb
/bin/mount /var/netwitness/decoder/packetdb
```

```
[root@NWHOST2000 ~]# df -h
Filesystem Size Used Avail Use% Mounted on
/dev/mapper/netwitness_vg00-root 30G 2.5G 28G 9% /
devtmpfs 63G 0 63G 0% /dev
tmpfs 63G 12K 63G 1% /dev/shm
tmpfs 63G 26M 63G 1% /run
tmpfs 63G 0 63G 0% /sys/fs/cgroup
/dev/mapper/netwitness_vg00-nwhome 2.7T 98M 2.7T 1% /var/netwitness
/dev/mapper/netwitness_vg00-varlog 10G 50M 10G 1% /var/log
/dev/mapper/netwitness_vg00-usrhome 10G 33M 10G 1% /home
/dev/sda1 1014M 88M 927M 9% /boot
tmpfs 13G 0 13G 0% /run/user/0
/dev/mapper/decodersmall-decoroot 10G 33M 10G 1% /var/netwitness/decoder
/dev/mapper/decodersmall-index 30G 33M 30G 1% /var/netwitness/decoder/index
/dev/mapper/decodersmall-sessiondb 600G 33M 600G 1% /var/netwitness/decoder/sessiondb
/dev/mapper/decodersmall-metadb 6.7T 33M 6.7T 1% /var/netwitness/decoder/metadb
/dev/mapper/decoder-packetdb 41T 34M 41T 1% /var/netwitness/decoder/packetdb
```

In the following example, the following partions are created for SDC, SDD (Enclosure 0).

```
[root@NWHOST2000 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 931G 0 disk
├─sda1 8:1 0 1G 0 part /boot
├─sda2 8:2 0 930G 0 part
│ ├─netwitness_vg00-root 253:0 0 30G 0 lvm /
│ ├─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
│ ├─netwitness_vg00-nwhome 253:2 0 2.7T 0 lvm /var/netwitness
│ ├─netwitness_vg00-varlog 253:3 0 10G 0 lvm /var/log
│ └─netwitness_vg00-usrhome 253:4 0 10G 0 lvm /home
sdb 8:16 0 1.8T 0 disk
├─sdb1 8:17 0 1.8T 0 part
│ └─netwitness_vg00-nwhome 253:2 0 2.7T 0 lvm /var/netwitness
sdc 8:32 0 7.3T 0 disk
├─sdc1 8:33 0 7.3T 0 part
│ ├─decodersmall-decoroot 253:5 0 10G 0 lvm /var/netwitness/decoder
│ ├─decodersmall-index 253:6 0 30G 0 lvm /var/netwitness/decoder/index
│ ├─decodersmall-sessiondb 253:7 0 600G 0 lvm /var/netwitness/decoder/sessiondb
│ └─decodersmall-metadb 253:8 0 6.7T 0 lvm /var/netwitness/decoder/metadb
sdd 8:48 0 40T 0 disk
├─sdd1 8:49 0 40T 0 part
│ └─decoder-packetdb 253:9 0 40T 0 lvm /var/netwitness/decoder/packetdb
sde 8:64 0 7.3T 0 disk
sdf 8:80 0 40T 0 disk
```

At this point, you add the second DAC enclosure.

- Execute the `partNew` command to create the `decodersmall` partition first (Second Enclosure, SDE, SDF) with the following parameters.

```
name=sde service=decoder volume=decodersmall commit=1
```

Properties for 11mtlnxnwpacket01 - Decoder (DECODER) /deviceappliance/appliance.

```
partNew Parameters name=sde service=decoder volume=decodersmall commit=1
Message Help
name - <string, {enum-one:sdc,sdd,sde,sdf}> block device name
service - <string, {enum-one:archiver|concentrator|decoder|logdecoder}> service that will use storage
volume - <string, optional, {enum-one:index|concentrator|decodersmall|decoder|logdecodersmall|logdecoder|archiver}> volume to create
commit - <bool, optional> commit changes
```

#### Response Output

```
/sbin/parted -s /dev/sde mklabel gpt
/sbin/parted -s -a optimal /dev/sde mkpart LVM 0% 100%
/sbin/pvcreate -f /dev/sde1
Physical volume "/dev/sde1" successfully created.
/sbin/vgcreate -f decodersmall0 /dev/sde1
Volume group "decodersmall0" successfully created
/sbin/lvcreate -y -n index -L 30G decodersmall0
Logical volume "index" created.
/sbin/mkfs.xfs /dev/decodersmall0/index
meta-data=/dev/decodersmall0/index isize=512 agcount=4, agsize=1966080 blks
= sectsz=512 attr=2, projid32bit=1
= crc=1 finobt=0, sparse=0
data = bsize=4096 blocks=7864320, imaxpct=25
= sunit=0 swidth=0 blks
naming =version 2 bsize=4096 ascii-ci=0 ftype=1
log =internal log bsize=4096 blocks=3840, version=2
= sectsz=512 sunit=0 blks, lazy-count=1
realtime =none extsz=4096 blocks=0, rtextents=0
/bin/mkdir -p /var/netwitness/decoder/index0
/bin/mount /var/netwitness/decoder/index0
```

```
[root@NWHOST2000 ~]# df -h
Filesystem Size Used Avail Use% Mounted on
/dev/mapper/netwitness_vg00-root 30G 2.5G 28G 9% /
devtmpfs 63G 0 63G 0% /dev
tmpfs 63G 12K 63G 1% /dev/shm
tmpfs 63G 26M 63G 1% /run
tmpfs 63G 0 63G 0% /sys/fs/cgroup
/dev/mapper/netwitness_vg00-nwhome 2.7T 98M 2.7T 1% /var/netwitness
/dev/mapper/netwitness_vg00-varlog 10G 50M 10G 1% /var/log
/dev/mapper/netwitness_vg00-usrhome 10G 33M 10G 1% /home
/dev/sda1 1014M 88M 927M 9% /boot
tmpfs 13G 0 13G 0% /run/user/0
/dev/mapper/decodersmall-decoroot 10G 33M 10G 1% /var/netwitness/decoder
/dev/mapper/decodersmall-index 30G 33M 30G 1% /var/netwitness/decoder/index
/dev/mapper/decodersmall-sessiondb 600G 33M 600G 1% /var/netwitness/decoder/sessiondb
/dev/mapper/decodersmall-metadb 6.7T 33M 6.7T 1% /var/netwitness/decoder/metadb
/dev/mapper/decoder-packetdb_ 41T 34M 41T 1% /var/netwitness/decoder/packetdb
```

8. Execute the `partNew` command to create the `packetdb` decoder volume (Second Enclosure SDE, SDF) with the following parameters.

```
name=sdf service=decoder volume=decoder commit=1
```

partNew Parameters name=sdf service=decoder volume=decoder commit=1

#### Message Help

```
name - <string, {enum-one:sdc,sdd,sde,sdf}> block device name
service - <string, {enum-one:archiver|concentrator|decoder|logdecoder}> service that will use storage
volume - <string, optional, {enum-one:index|concentrator|decodersmall|decoder|logdecodersmall|logdecoder|archiver}> volume to create
commit - <bool, optional> commit changes
```

#### Response Output

```
/sbin/parted -s /dev/sdf mklabel gpt
/sbin/parted -s -a optimal /dev/sdf mkpart LVM 0% 100%
/sbin/pvcreate -f /dev/sdf1
Physical volume "/dev/sdf1" successfully created.
/sbin/vgcreate -f decoder0 /dev/sdf1
Volume group "decoder0" successfully created
/sbin/lvcreate -y -n packetdb -l 100%FREE decoder0
Logical volume "packetdb" created.
/sbin/mkfs.xfs /dev/decoder0/packetdb
meta-data=/dev/decoder0/packetdb isize=512 agcount=41, agsize=268435455 blks
= sectsz=512 attr=2, projid32bit=1
= crc=1 finobt=0, sparse=0
data = bsize=4096 blocks=10742791168, imaxpct=5
= sunit=0 swidth=0 blks
naming =version 2 bsize=4096 ascii-ci=0 ftype=1
log =internal log bsize=4096 blocks=521728, version=2
= sectsz=512 sunit=0 blks, lazy-count=1
realtime =none extsz=4096 blocks=0, rtextents=0
/bin/mkdir -p /var/netwitness/decoder/packetdb0
/bin/mount /var/netwitness/decoder/packetdb0
```

```
[root@NWHOST2000 ~]# df -h
Filesystem Size Used Avail Use% Mounted on
/dev/mapper/netwitness_vg00-root 30G 2.5G 28G 9% /
devtmpfs 63G 0 63G 0% /dev
tmpfs 63G 12K 63G 1% /dev/shm
tmpfs 63G 27M 63G 1% /run
tmpfs 63G 0 63G 0% /sys/fs/cgroup
/dev/mapper/netwitness_vg00-nwhome 2.7T 98M 2.7T 1% /var/netwitness
/dev/mapper/netwitness_vg00-varlog 10G 50M 10G 1% /var/log
/dev/mapper/netwitness_vg00-usrhome 10G 33M 10G 1% /home
/dev/sda1 1014M 88M 927M 9% /boot
tmpfs 13G 0 13G 0% /run/user/0
/dev/mapper/decodersmall-decoroot 10G 33M 10G 1% /var/netwitness/decoder
/dev/mapper/decodersmall-index 30G 33M 30G 1% /var/netwitness/decoder/index
/dev/mapper/decodersmall-sessiondb 600G 33M 600G 1% /var/netwitness/decoder/sessiondb
/dev/mapper/decodersmall-metadb 6.7T 33M 6.7T 1% /var/netwitness/decoder/metadb
/dev/mapper/decoder-packetdb 41T 34M 41T 1% /var/netwitness/decoder/packetdb
/dev/mapper/decodersmall10-index 30G 33M 30G 1% /var/netwitness/decoder/index0
/dev/mapper/decodersmall10-sessiondb 600G 33M 600G 1% /var/netwitness/decoder/sessiondb0
/dev/mapper/decodersmall10-metadb 6.7T 33M 6.7T 1% /var/netwitness/decoder/metadb0
/dev/mapper/decoder0-packetdb 41T 34M 41T 1% /var/netwitness/decoder/packetdb0

[root@NWHOST2000 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 931G 0 disk
├─sda1 8:1 0 1G 0 part /boot
├─sda2 8:2 0 930G 0 part
│ ├─netwitness_vg00-root 253:0 0 30G 0 lvm /
│ ├─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
│ ├─netwitness_vg00-nwhome 253:2 0 2.7T 0 lvm /var/netwitness
│ ├─netwitness_vg00-varlog 253:3 0 10G 0 lvm /var/log
│ └─netwitness_vg00-usrhome 253:4 0 10G 0 lvm /home
sdb 8:16 0 1.8T 0 disk
├─sdb1 8:17 0 1.8T 0 part
└─netwitness_vg00-nwhome 253:2 0 2.7T 0 lvm /var/netwitness
sdc 8:32 0 7.3T 0 disk
├─sdc1 8:33 0 7.3T 0 part
│ ├─decodersmall-decoroot 253:5 0 10G 0 lvm /var/netwitness/decoder
│ ├─decodersmall-index 253:6 0 30G 0 lvm /var/netwitness/decoder/index
│ ├─decodersmall-sessiondb 253:7 0 600G 0 lvm /var/netwitness/decoder/sessiondb
│ └─decodersmall-metadb 253:8 0 6.7T 0 lvm /var/netwitness/decoder/metadb
sdd 8:48 0 40T 0 disk
├─sdd1 8:49 0 40T 0 part
└─decoder-packetdb 253:9 0 40T 0 lvm /var/netwitness/decoder/packetdb
sde 8:64 0 7.3T 0 disk
├─sde1 8:65 0 7.3T 0 part
│ ├─decodersmall10-index 253:10 0 30G 0 lvm /var/netwitness/decoder/index0
│ ├─decodersmall10-sessiondb 253:11 0 600G 0 lvm /var/netwitness/decoder/sessiondb0
│ └─decodersmall10-metadb 253:12 0 6.7T 0 lvm /var/netwitness/decoder/metadb0
sdf 8:80 0 40T 0 disk
├─sdf1 8:81 0 40T 0 part
└─decoder0-packetdb 253:13 0 40T 0 lvm /var/netwitness/decoder/packetdb0
```

9. Execute the `srvAlloc` command with the following parameters to add the storage information into the Service Configuration settings.

- `service=decoder volume=decodersmall commit=1`
- `service=decoder volume=decodersmall10 commit=1`
- `service=decoder volume=decoder commit=1`
- `service=decoder volume=decoder0 commit=1`

srvAlloc ▾ Parameters service=decoder commit=1 volume=decoder0

#### Message Help

service - <string, {enum-one:archiver|concentrator|decoder|logdecoder}> service that will use storage  
 volume - <string, {enum-one:decoder,decoder0,decodersmall,decodersmall0,netwitness\_vg00}> volume group name  
 commit - <bool, optional> commit changes

#### Response Output

Set /database/config/packet.dir to /var/netwitness/decoder/packetdb==38 TB;/var/netwitness/decoder/packetdb0==38,01 TB

| /database/config         | NWHOST2000 - Concentrator                                                                  |
|--------------------------|--------------------------------------------------------------------------------------------|
| meta.compression         | none                                                                                       |
| meta.compression.level   | 0                                                                                          |
| meta.dir                 | /var/netwitness/decoder/metadb==6.3 TB;/var/netwitness/decoder/metadb0==6.32 TB            |
| meta.dir.cold            |                                                                                            |
| meta.dir.warm            |                                                                                            |
| meta.file.size           | auto                                                                                       |
| meta.files               | auto                                                                                       |
| meta.free.space.min      | 23 GB                                                                                      |
| meta.index.fidelity      | 4                                                                                          |
| meta.integrity.flush     | sync                                                                                       |
| meta.write.block.size    | 64 KB                                                                                      |
| packet.compression       | none                                                                                       |
| packet.compression.level | 0                                                                                          |
| packet.dir               | /var/netwitness/decoder/packetdb==38 TB;/var/netwitness/decoder/packetdb0==38,01 TB        |
| packet.dir.cold          |                                                                                            |
| packet.dir.warm          |                                                                                            |
| packet.file.size         | auto                                                                                       |
| packet.file.type         | pcapng                                                                                     |
| packet.files             | auto                                                                                       |
| packet.free.space.min    | 23 GB                                                                                      |
| packet.index.fidelity    | 1                                                                                          |
| packet.integrity.flush   | sync                                                                                       |
| packet.write.block.size  | 64 KB                                                                                      |
| session.dir              | /var/netwitness/decoder/sessiondb==569.71 GB;/var/netwitness/decoder/sessiondb0==569.72 GB |
| session.dir.cold         |                                                                                            |

10. Reconfigure the following Network Decoder service and its database to detect and take advantage of all of the free space as described in [Task 5 - \(Optional\) Reconfigure Storage Configuration for 10G Capture](#).

## Configure Storage for Network Concentrator

The following scenario configures storage on one, non-encrypted, 15-Drive DAC for a Network Concentrator physical host.

1. Execute the `raidList` command.

`raidList`

Message Help

```
list drive shelves attached to this appliance
security.roles: appliance.manage
```

### Response Output

```
Controller 0, Enclosure 32
Vendor: DP
Model: BP13G+EXP
In Use: true
Drives: 931.511 GB x 2
 1.818 TB x 2
Devices: sda
 sdb
```

```
Controller 1, Enclosure 6
Vendor: EMC
Model: ESES Enclosure
In Use: false
Drives: 186.309 GB x 6
 3.637 TB x 9
Devices:
```



- Execute the `raidNew` command with the following parameters.

`controller=1 enclosure=6 scheme=concentrator`

raidNew Parameters `controller=1 enclosure=6 scheme=concentrator commit=1`

Message Help

parameters:  
 controller - <int32, {enum-one:0,1}> Controller the shelf is attached to  
 enclosure - <uint32, {enum-one:32,6}> Enclosure number of the shelf to clear  
 scheme - <string, {enum-one:decoder|logdecoder|concentrator|archiver|network-hybrid|log-hybrid}> Type of RAID volumes to allocate

## Response Output

```
/opt/MegaRAID/perccli/perccli64 /c1 add vd r5 drives=6:0,6:1,6:2,6:3,6:4,6:5 wb ra cached Strip=128
Controller = 1
Status = Success
Description = Add VD Succeeded

/opt/MegaRAID/perccli/perccli64 /c1 add vd r6 drives=6:6,6:7,6:8,6:9,6:10,6:11,6:12,6:13,6:14 wb ra cached Strip=128
Controller = 1
Status = Success
Description = Add VD Succeeded
```

```
[root@NWHOST1500 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 931G 0 disk
├─sda1 8:1 0 1G 0 part /boot
└─sda2 8:2 0 930G 0 part
 ├─netwitness_vg00-root 253:0 0 30G 0 lvm /
 ├─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
 ├─netwitness_vg00-nwhome 253:2 0 2.7T 0 lvm /var/netwitness
 ├─netwitness_vg00-varlog 253:3 0 10G 0 lvm /var/log
 └─netwitness_vg00-usrhome 253:4 0 10G 0 lvm /home
sdb 8:16 0 1.8T 0 disk
├─sdb1 8:17 0 1.8T 0 part
└─netwitness_vg00-nwhome 253:2 0 2.7T 0 lvm /var/netwitness
sdc 8:32 0 928.8G 0 disk
sdd 8:48 0 25.5T 0 disk
[root@NWHOST1500 ~]#
```

- Execute the `partNew` command to create the **concentrator** partition first with the following parameters. You must create the **concentrator** volume before **index** volume or it will fail.

```
name=sdd service=concentrator volume=concentrator commit=1
```

```
partNew Parameters name=sdd service=concentrator volume=concentrator commit=1
```

Message Help

```
parameters:
name - <string, {enum-one:sdc,sdd}> block device name
service - <string, {enum-one:archiver|concentrator|decoder|logdecoder}> service that will use storage
volume - <string, optional, {enum-one:index|concentrator|decodersmall|decoder|logdecodersmall|logdecoder|archiver}> volume to create
```

Response Output

```
/sbin/parted -s /dev/sdd mklabel gpt
/sbin/parted -s -a optimal /dev/sdd mkpart LVM 0% 100%
/sbin/pvcreate -f /dev/sdd1
Physical volume "/dev/sdd1" successfully created.
/sbin/vgcreate -f concentrator /dev/sdd1
Volume group "concentrator" successfully created
/sbin/lvcreate -y -n root -L 30G concentrator
Logical volume "root" created.
/sbin/mkfs.xfs /dev/concentrator/root
meta-data=/dev/concentrator/root isize=512 agcount=4, agsize=1966080 blks
= sectsz=512 attr=2, projid32bit=1
= crc=1 finobt=0, sparse=0
data = bsize=4096 blocks=7864320, imaxpct=25
= sunit=0 swidth=0 blks
naming =version 2 bsize=4096 ascii-ci=0 ftype=1
log =internal log bsize=4096 blocks=3840, version=2
= sectsz=512 sunit=0 blks, lazy-count=1
realtime =none extsz=4096 blocks=0, rtextents=0
/bin/mkdir -p /var/netwitness/concentrator
/bin/mount /var/netwitness/concentrator
```

```
[root@NWHOST1500 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 931G 0 disk
├─sda1 8:1 0 1G 0 part /boot
├─sda2 8:2 0 930G 0 part
│ ├─netwitness_vg00-root 253:0 0 30G 0 lvm /
│ ├─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
│ ├─netwitness_vg00-nwhome 253:2 0 2.7T 0 lvm /var/netwitness
│ ├─netwitness_vg00-varlog 253:3 0 10G 0 lvm /var/log
│ └─netwitness_vg00-usrhome 253:4 0 10G 0 lvm /home
sdb 8:16 0 1.8T 0 disk
├─sdb1 8:17 0 1.8T 0 part
└─netwitness_vg00-nwhome 253:2 0 2.7T 0 lvm /var/netwitness
sdc 8:32 0 928.8G 0 disk
sdd 8:48 0 25.5T 0 disk
├─sdd1 8:49 0 25.5T 0 part
│ ├─concentrator-root 253:5 0 30G 0 lvm /var/netwitness/concentrator
│ ├─concentrator-sessiondb 253:6 0 600G 0 lvm /var/netwitness/concentrator/sessiondb
│ └─concentrator-metadb 253:7 0 24.9T 0 lvm /var/netwitness/concentrator/metadb
```

- Execute the partNew command with the following parameters with the following parameters to create an index on SSDs.

```
name=sdc service=concentrator volume=index commit=1
```



```

partNew Parameters name=sdc service=concentrator volume=index commit=1
Message Help
parameters:
name - <string, {enum-one:sdc,sdd}> block device name
service - <string, {enum-one:archiver|concentrator|decoder|logdecoder}> service that will use storage
volume - <string, optional, {enum-one:index|concentrator|decodersmall|decoder|logdecodersmall|logdecoder|archiver}> volume to create

Response Output
/sbin/parted -s /dev/sdc mklabel gpt
/sbin/parted -s -a optimal /dev/sdc mkpart LVM 0% 100%
/sbin/pvcreate -f /dev/sdc1
Physical volume "/dev/sdc1" successfully created.
/sbin/vgcreate -f index /dev/sdc1
Volume group "index" successfully created
/sbin/lvcreate -y -n index -l 100%FREE index
Wiping xfs signature on /dev/index/index.
Logical volume "index" created.
/sbin/mkfs.xfs /dev/index/index
meta-data=/dev/index/index isize=512 agcount=4, agsize=60866304 blks
= sectsz=4096 attr=2, projid32bit=1
= crc=1 finobt=0, sparse=0
data = bsize=4096 blocks=243465216, imaxpct=25
= sunit=0 swidth=0 blks
naming =version 2 bsize=4096 ascii-ci=0 ftype=1
log =internal log bsize=4096 blocks=118879, version=2
= sectsz=4096 sunit=1 blks, lazy-count=1
realtime =none extsz=4096 blocks=0, rtextents=0
/bin/mkdir -p /var/netwitness/concentrator/index

[root@NWHOST1500 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 931G 0 disk
├─sda1 8:1 0 1G 0 part /boot
├─sda2 8:2 0 930G 0 part
│ ├─netwitness_vg00-root 253:0 0 30G 0 lvm /
│ ├─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
│ ├─netwitness_vg00-nwhome 253:2 0 2.7T 0 lvm /var/netwitness
│ └─netwitness_vg00-varlog 253:3 0 10G 0 lvm /var/log
└─netwitness_vg00-usrhome 253:4 0 10G 0 lvm /home
sdb 8:16 0 1.8T 0 disk
├─sdb1 8:17 0 1.8T 0 part
└─netwitness_vg00-nwhome 253:2 0 2.7T 0 lvm /var/netwitness
sdc 8:32 0 928.8G 0 disk
├─sdc1 8:33 0 928.8G 0 part
└─index-index 253:8 0 928.8G 0 lvm /var/netwitness/concentrator/index
sdd 8:48 0 25.5T 0 disk
├─sdd1 8:49 0 25.5T 0 part
│ ├─concentrator-root 253:5 0 30G 0 lvm /var/netwitness/concentrator
│ ├─concentrator-sessiondb 253:6 0 600G 0 lvm /var/netwitness/concentrator/sessiondb
│ └─concentrator-metadb 253:7 0 24.9T 0 lvm /var/netwitness/concentrator/metadb

```

```
[root@NWHOST1500 ~]# df -h
Filesystem Size Used Avail Use% Mounted on
/dev/mapper/netwitness_vg00-root 30G 2.1G 28G 7% /
devtmpfs 63G 0 63G 0% /dev
tmpfs 63G 12K 63G 1% /dev/shm
tmpfs 63G 10M 63G 1% /run
tmpfs 63G 0 63G 0% /sys/fs/cgroup
/dev/sda1 1014M 91M 924M 9% /boot
/dev/mapper/netwitness_vg00-varlog 10G 52M 10G 1% /var/log
/dev/mapper/netwitness_vg00-usrhome 10G 33M 10G 1% /home
/dev/mapper/netwitness_vg00-nwhome 2.7T 98M 2.7T 1% /var/netwitness
tmpfs 13G 0 13G 0% /run/user/0
/dev/mapper/concentrator-root 30G 33M 30G 1% /var/netwitness/concentrator
/dev/mapper/concentrator-sessiondb 600G 33M 600G 1% /var/netwitness/concentrator/sessiondb
/dev/mapper/concentrator-metadb 25T 33M 25T 1% /var/netwitness/concentrator/metadb
/dev/mapper/index-index 929G 33M 929G 1% /var/netwitness/concentrator/index
```

- Execute the `srvAlloc` command with the following parameters.

```
service=concentrator volume=index commit=1
```

Parameters

Message Help

parameters:

service - <string, {enum-one:archiver|concentrator|decoder|logdecoder}> service that will use storage

volume - <string, {enum-one:concentrator,index,netwitness\_vg00}> volume group name

commit - <bool, optional> commit changes

#### Response Output

```
Set /index/config/index.dir to /var/netwitness/concentrator/index==881.87 GB
```

| Change Service   NWHOST1500 - Concentrator   Explore |                    |
|------------------------------------------------------|--------------------|
| NWHOST1500 - Concentrator                            | /index/config      |
| NWHOST1500 - Concentrator (CONC)                     | index.dir          |
| concentrator                                         | index.dir.cold     |
| connections                                          | index.dir.warm     |
| database                                             | index.slices.open  |
| deviceappliance                                      | page.compression   |
| index                                                | reindex.enable     |
| config                                               | save.session.count |

- Execute the `srvAlloc` command with the following parameters.  

```
service=concentrator volume=concentrator commit=1
```

srvAlloc ▾ Parameters `service=concentrator volume=concentrator commit=1`

Message Help

parameters:

- service - <string, {enum-one:archiver|concentrator|decoder|logdecoder}> service that will use storage
- volume - <string, {enum-one:concentrator,index,netwitness\_vg00}> volume group name
- commit - <bool, optional> commit changes

Response Output

Set /database/config/meta.dir to /var/netwitness/concentrator/metadb==23.6 TB

Set /database/config/session.dir to /var/netwitness/concentrator/sessiondb==569.72 GB

| Parameter              | Value                                             |
|------------------------|---------------------------------------------------|
| hash.algorithm         | none                                              |
| hash.databases         | session,meta                                      |
| hash.dir               |                                                   |
| manifest.dir           |                                                   |
| meta.compression       | none                                              |
| meta.compression.level | 0                                                 |
| meta.dir               | /var/netwitness/concentrator/metadb==23.6 TB      |
| meta.dir.cold          |                                                   |
| meta.dir.warm          |                                                   |
| meta.file.size         | auto                                              |
| meta.files             | auto                                              |
| meta.free.space.min    | 23 GB                                             |
| meta.index.fidelity    | 4                                                 |
| meta.integrity.flush   | sync                                              |
| meta.write.block.size  | 64 KB                                             |
| session.dir            | /var/netwitness/concentrator/sessiondb==569.72 GB |

7. Reconfigure the following Network Concentrator service and its database to detect and take advantage of all of the free space as described in [Task 5 - \(Optional\) Reconfigure Storage Configuration for 10G Capture](#).

## Configure Storage for Log Decoder Hybrid

The following scenario configures storage on one, non-encrypted, 15-Drive DAC for a Log Decoder Hybrid physical host.

1. Execute the `raidList` command.

raidList

Message Help

```
list drive shelves attached to this appliance
security.roles: appliance.manage
```

### Response Output

```
Controller 0, Enclosure 32
Vendor: DP
Model: BP13G+EXP
In Use: true
Drives: 745.21 GB x 2
 931.511 GB x 4
 5.457 TB x 8
Devices: sda
 sdb
 sdc
 sdd
 sde

Controller 1, Enclosure 31
Vendor: EMC
Model: ESES Enclosure
In Use: false
Drives: 3.637 TB x 15
Devices:
```

2. Execute the `raidNew` command with the following parameters.  
`controller=1 enclosure=31 scheme=log-hybrid commit=1`

raidNew Parameters controller=1 enclosure=31 scheme=log-hybrid commit=1

Message Help

controller - <int32, {enum-one:0,1}> Controller the shelf is attached to  
 enclosure - <uint32, {enum-one:32,31}> Enclosure number of the shelf to clear  
 scheme - <string, {enum-one:decoder|logdecoder|concentrator|archiver|network-hybrid|log-hybrid}> Type of RAID volumes to allocate  
 preferSecure - <bool, optional, {bool:0,1,yes,no,true,false,on,off}> Prefer creation of a secure array given compatible physical drives and a controller with a security key set

## Response Output

```
/opt/MegaRAID/perccli/perccli64 /c1 add vd r5 drives=31:0,31:1,31:2,31:3,31:4,31:5,31:6 wb ra cached Strip=128
Controller = 1
Status = Success
Description = Add VD Succeeded
```

```
/opt/MegaRAID/perccli/perccli64 /c1 add vd r5 drives=31:7,31:8,31:9,31:10,31:11,31:12,31:13,31:14 wb ra cached Strip=128
Controller = 1
Status = Success
Description = Add VD Succeeded
```

```
[root@NWHOST1700 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 931G 0 disk
├─sda1 8:1 0 1G 0 part /boot
├─sda2 8:2 0 930G 0 part
│ ├─netwitness_vg00-root 253:0 0 30G 0 lvm /
│ ├─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
│ ├─netwitness_vg00-nwhome 253:11 0 876G 0 lvm /var/netwitness
│ ├─netwitness_vg00-varlog 253:12 0 10G 0 lvm /var/log
│ └─netwitness_vg00-usrhome 253:13 0 10G 0 lvm /home
sdb 8:16 0 931G 0 disk
├─sdb1 8:17 0 931G 0 part
│ └─decodermeta-vlnwdm 253:9 0 931G 0 lvm /var/netwitness/decoder/metadb
sdc 8:32 0 16.4T 0 disk
├─sdc1 8:33 0 16.4T 0 part
│ ├─decoderpacket-vlnwdp 253:2 0 16.2T 0 lvm /var/netwitness/decoder/packetdb
│ ├─decoderpacket-vlnwds 253:3 0 100G 0 lvm /var/netwitness/decoder/sessiondb
│ ├─decoderpacket-vlnwdi 253:4 0 50G 0 lvm /var/netwitness/decoder/index
│ └─decoderpacket-vlnwd 253:5 0 30G 0 lvm /var/netwitness/decoder
sdd 8:48 0 16.4T 0 disk
├─sdd1 8:49 0 16.4T 0 part
│ ├─concentrator-vlnwcm 253:6 0 14.9T 0 lvm /var/netwitness/concentrator/metadb
│ ├─concentrator-vlnwcs 253:7 0 1.5T 0 lvm /var/netwitness/concentrator/sessiondb
│ └─concentrator-vlnwc 253:8 0 30G 0 lvm /var/netwitness/concentrator
sde 8:64 0 744.6G 0 disk
├─sde1 8:65 0 744.6G 0 part
│ └─index-vlnwci 253:10 0 744.6G 0 lvm /var/netwitness/concentrator/index
sdf 8:80 0 21.8T 0 disk
sdg 8:96 0 25.5T 0 disk
```

3. Execute the `partNew` command with the following parameters with the following parameters.

- `name=sdf service=concentrator volume=concentrator commit=1`

partNew

Message Help

name - <string, {enum-one:sdf,sdg}> block device name  
 service - <string, {enum-one:archiver|concentrator|decoder|logdecoder}> service that will use storage  
 volume - <string, optional, {enum-one:index|concentrator|decodersmall|decoder|logdecodersmall|logdecoder|archiver}> volume to create  
 commit - <bool, optional> commit changes

## Response Output

```
/sbin/parted -s /dev/sdf mklabel gpt
/sbin/parted -s -a optimal /dev/sdf mkpart LVM 0% 100%
/sbin/pvcreate -f /dev/sdf1
Physical volume "/dev/sdf1" successfully created.
/sbin/vgcreate -f concentrator0 /dev/sdf1
Volume group "concentrator0" successfully created
```

```
[root@NWHOST1700 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 931G 0 disk
├─sda1 8:1 0 1G 0 part /boot
├─sda2 8:2 0 930G 0 part
│ ├─netwitness_vg00-root 253:0 0 30G 0 lvm /
│ ├─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
│ ├─netwitness_vg00-nwhome 253:11 0 876G 0 lvm /var/netwitness
│ ├─netwitness_vg00-varlog 253:12 0 10G 0 lvm /var/log
│ └─netwitness_vg00-usrhome 253:13 0 10G 0 lvm /home
sdb 8:16 0 931G 0 disk
├─sdb1 8:17 0 931G 0 part
└─decodermeta-vlnwdm 253:9 0 931G 0 lvm /var/netwitness/decoder/metadb
sdc 8:32 0 16.4T 0 disk
├─sdc1 8:33 0 16.4T 0 part
│ ├─decoderpacket-vlnwdp 253:2 0 16.2T 0 lvm /var/netwitness/decoder/packetdb
│ ├─decoderpacket-vlnwds 253:3 0 100G 0 lvm /var/netwitness/decoder/sessiondb
│ ├─decoderpacket-vlnwdi 253:4 0 50G 0 lvm /var/netwitness/decoder/index
│ └─decoderpacket-vlnwd 253:5 0 30G 0 lvm /var/netwitness/decoder
sdd 8:48 0 16.4T 0 disk
├─sdd1 8:49 0 16.4T 0 part
│ ├─concentrator-vlnwcm 253:6 0 14.9T 0 lvm /var/netwitness/concentrator/metadb
│ ├─concentrator-vlnwcs 253:7 0 1.5T 0 lvm /var/netwitness/concentrator/sessiondb
│ └─concentrator-vlnwc 253:8 0 30G 0 lvm /var/netwitness/concentrator
sde 8:64 0 744.6G 0 disk
├─sde1 8:65 0 744.6G 0 part
└─index-vlnwci 253:10 0 744.6G 0 lvm /var/netwitness/concentrator/index
sdf 8:80 0 21.8T 0 disk
├─sdf1 8:81 0 21.8T 0 part
│ ├─concentrator0-sessiondb 253:14 0 600G 0 lvm /var/netwitness/concentrator/sessiondb0
│ └─concentrator0-metadb 253:15 0 21.2T 0 lvm /var/netwitness/concentrator/metadb0
sdg 8:96 0 25.5T 0 disk
```

- name=sdg service=logdecoder volume=logdecoder commit=1

partNew Parameters name=sdg service=logdecoder volume=logdecoder commit=1

Message Help

name - <string, {enum-one:sdf,sdg}> block device name  
 service - <string, {enum-one:archiver|concentrator|decoder|logdecoder}> service that will use storage  
 volume - <string, optional, {enum-one:index|concentrator|decodersmall|decoder|logdecodersmall|logdecoder|archiver}> volume to create  
 commit - <bool, optional> commit changes

Response Output

```
/sbin/parted -s /dev/sdg mklabel gpt
/sbin/parted -s -a optimal /dev/sdg mkpart LVM 0% 100%
/sbin/pvcreate -f /dev/sdg1
Physical volume "/dev/sdg1" successfully created.
/sbin/vgcreate -f logdecoder0 /dev/sdg1
Volume group "logdecoder0" successfully created
```

```
[root@NWHOST1700 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 931G 0 disk
├─sda1 8:1 0 1G 0 part /boot
└─sda2 8:2 0 930G 0 part
 ├─netwitness_vg00-root 253:0 0 30G 0 lvm /
 ├─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
 ├─netwitness_vg00-nwhome 253:11 0 876G 0 lvm /var/netwitness
 ├─netwitness_vg00-varlog 253:12 0 10G 0 lvm /var/log
 └─netwitness_vg00-usrhome 253:13 0 10G 0 lvm /home
sdb 8:16 0 931G 0 disk
├─sdb1 8:17 0 931G 0 part
└─decodermeta-vlnwdm 253:9 0 931G 0 lvm /var/netwitness/decoder/metadb
sdc 8:32 0 16.4T 0 disk
├─sdc1 8:33 0 16.4T 0 part
│ ├─decoderpacket-vlnwdp 253:2 0 16.2T 0 lvm /var/netwitness/decoder/packetdb
│ ├─decoderpacket-vlnwds 253:3 0 100G 0 lvm /var/netwitness/decoder/sessiondb
│ ├─decoderpacket-vlnwdi 253:4 0 50G 0 lvm /var/netwitness/decoder/index
│ └─decoderpacket-vlnwd 253:5 0 30G 0 lvm /var/netwitness/decoder
sdd 8:48 0 16.4T 0 disk
├─sdd1 8:49 0 16.4T 0 part
│ ├─concentrator-vlnwcm 253:6 0 14.9T 0 lvm /var/netwitness/concentrator/metadb
│ ├─concentrator-vlnwcs 253:7 0 1.5T 0 lvm /var/netwitness/concentrator/sessiondb
│ └─concentrator-vlnwc 253:8 0 30G 0 lvm /var/netwitness/concentrator
sde 8:64 0 744.6G 0 disk
├─sde1 8:65 0 744.6G 0 part
└─index-vlnwci 253:10 0 744.6G 0 lvm /var/netwitness/concentrator/index
sdf 8:80 0 21.8T 0 disk
├─sdf1 8:81 0 21.8T 0 part
│ ├─concentrator0-sessiondb 253:14 0 600G 0 lvm /var/netwitness/concentrator/sessiondb0
│ └─concentrator0-metadb 253:15 0 21.2T 0 lvm /var/netwitness/concentrator/metadb0
sdg 8:96 0 25.5T 0 disk
├─sdg1 8:97 0 25.5T 0 part
└─logdecoder0-packetdb 253:16 0 25.5T 0 lvm /var/netwitness/decoder/packetdb0
```



4. Execute the `srvAlloc` command with the following parameters.

- `service=concentrator volume=concentrator0 commit=1`

Parameters

Message Help

```

service - <string, {enum-one:archiver|concentrator|decoder|logdecoder}> service that will use storage
volume - <string, {enum-one:concentrator,concentrator0,decodermeta,decoderpacket,index,logdecoder0,netwitness_vg00}> volume group name
commit - <bool, optional> commit changes

```

## Response Output

```

Set /database/config/meta.dir to /var/netwitness/concentrator/metadb=14.08 TB;/var/netwitness/concentrator/metadb0==20.17 TB
Set /database/config/session.dir to /var/netwitness/concentrator/sessiondb=1.41 TB;/var/netwitness/concentrator/sessiondb0==569.72 GB

```

| NWHOST1700 - Concentrator (C) |                        | Explore                                                                                           |
|-------------------------------|------------------------|---------------------------------------------------------------------------------------------------|
| NWHOST1700 - Concentrator     | /database/config       | NWHOST1700 - Concentrator                                                                         |
| NWHOST1700 - Concentrator (C) | hash.algorithm         | none                                                                                              |
| concentrator                  | hash.databases         | session_meta                                                                                      |
| connections                   | hash.dir               |                                                                                                   |
| database                      | manifest.dir           |                                                                                                   |
| config                        | meta.compression       | none                                                                                              |
| stats                         | meta.compression.level | 0                                                                                                 |
| deviceappliance               | meta.dir               | /var/netwitness/concentrator/metadb=14.08 TB;/var/netwitness/concentrator/metadb0==20.17 TB       |
| index                         | meta.dir.cold          |                                                                                                   |
| logs                          | meta.dir.warm          |                                                                                                   |
| rest                          | meta.file.size         | auto                                                                                              |
| sdk                           | meta.files             | auto                                                                                              |
| services                      | meta.free.space.min    | 132 GB                                                                                            |
| storedproc                    | meta.index.fidelity    | 4                                                                                                 |
| sys                           | meta.integrity.flush   | sync                                                                                              |
| users                         | meta.write.block.size  | 64 KB                                                                                             |
|                               | session.dir            | /var/netwitness/concentrator/sessiondb=1.41 TB;/var/netwitness/concentrator/sessiondb0==569.72 GB |

- `service=logdecoder volume=logdecoder0 commit=1`

5. Reconfigure the following Log Decoder service and its database to detect and take advantage of all of the free space as described in [Task 5 - \(Optional\) Reconfigure Storage Configuration for 10G Capture](#).



## Appendix E. Sample Storage Configuration Scenarios for 8 or 12-Drive PowerVault

This appendix illustrates the following example of how to configure storage on one non-encrypted 8 or 12-drive PowerVault external storage devices.

- [Configure Storage for Archiver using NW-PV-A/NW-PV-A-N](#)
- [Configure Storage for Decoder using NW-PV-B/NW-PV-B-N](#)
- [Configure Storage for Concentrator using NW-PV-C/NW-PV-C-N](#)
- [Configure Storage for Concentrator using NW-PV-D/NW-PV-D-N](#)

### Configure Storage for Archiver using NW-PV-A/NW-PV-A-N

The following scenario configures storage on one, non-encrypted, 12-Drive PowerVault for an Archiver physical host.

1. Execute the `raidList` command.
  - a. Record the Controller Number, Enclosure Number, In Use, Drives, and Devices.

You should see the following information.

```
In Use: FALSE
Devices: <empty>
```

- b. Verify the Drive Count, Size, and Vendor.

The following example illustrates what you should see before you create a RAID array.

Properties for /appliance

raidList  Parameters:

---

Message Help

```
raidList: list drive shelves attached to this appliance
security.roles: appliance.manage
```

---

</appliance?msg=raidList&force-content-type=text/plain>

---

Output (or command manual help)

```
Controller 0 at PCI Address 18:00.0, Enclosure 64, SCSI Channel 2
Vendor: DP
Model: BP14G+EXP
In Use: true
Drives: 1.09 TB HDD x 2
 2.182 TB HDD x 2
Devices: sda /dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:0:0
 sdb /dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:1:0

Controller 1 at PCI Address 3b:00.0, Enclosure 251, SCSI Channel 2
Vendor: DELL
Model: MD1400
In Use: false
Drives: 7.277 TB HDD x 12
Devices:
```

2. Execute the `raidNew` command with the following parameters using the controller number and the enclosure number you just recorded.

```
controller=1 enclosure=251 scheme=archiver commit=1
```

The following example illustrates what you should see after you create a RAID array.

Properties for /appliance  
 raidNew Parameters:

Message Help

```
raidNew: allocate RAID devices in a drive shelf
security.roles: appliance.manage
parameters:
 controller - <uint32, {enum-one:The value must be one of the following: 0,1}> Controller the shelf is attached to
 enclosure - <uint32, optional, {enum-one:The value must be one of the following: 64|251}> Enclosure number of the shelf to clear. Required if the contr
 scheme - <string, {enum-one:The value must be one of the following: decoder|logdecoder|concentrator|archiver|network-hybrid|log-hybrid|log-indexed-decod
 allocate
```

/appliance?msg=raidNew&force-content-type=text/plain&controller=1&enclosure=251&scheme=archiver&commit=1

Output (or command manual help)

```
/opt/MegaRAID/perccli/perccli64 /c1 add vd r6 drives=251:0,251:1,251:2,251:3,251:4,251:5,251:6,251:7,251:8,251:9,251:10,251:11 ra Strip=128
CLI Version = 007.1623.0000.0000 May 17, 2021
Operating system = Linux 3.10.0-1160.66.1.el7.x86_64
Controller = 1
Status = Success
Description = Add VD Succeeded.
```

- Execute the `raidList` command to verify the new RAID array.

You should now see the following information.

In Use: TRUE

Devices: <device> (for example, `sdc`)

Properties for /appliance  
 raidList Parameters:

Message Help

```
raidList: list drive shelves attached to this appliance
security.roles: appliance.manage
```

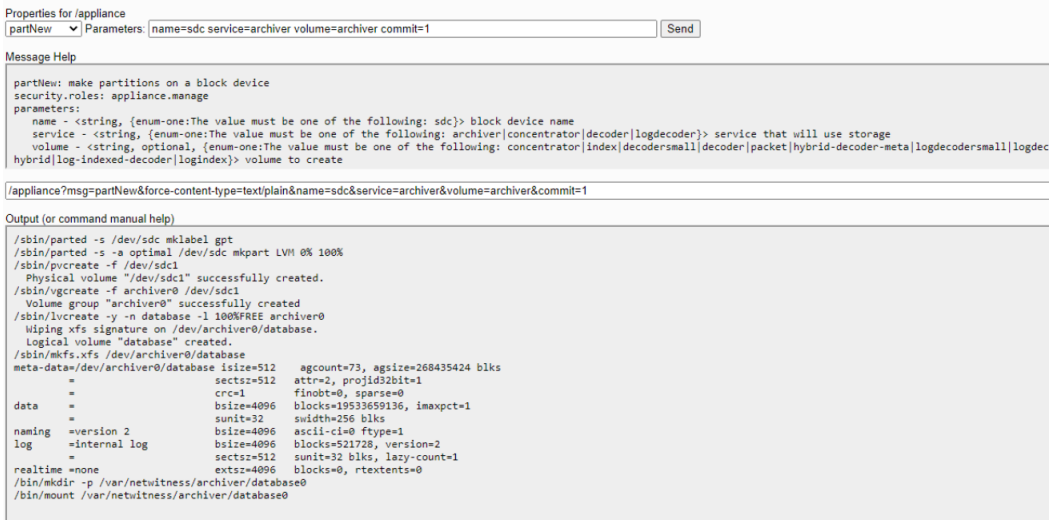
/appliance?msg=raidList&force-content-type=text/plain

Output (or command manual help)

```
Controller 0 at PCI Address 18:00.0, Enclosure 64, SCSI Channel 2
Vendor: DP
Model: BP14G+EXP
In Use: true
Drives: 1.09 TB HDD x 2
 2.182 TB HDD x 2
Devices: sda /dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:0:0
 sdb /dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:1:0

Controller 1 at PCI Address 3b:00.0, Enclosure 251, SCSI Channel 2
Vendor: DELL
Model: MD1400
In Use: true
Drives: 7.277 TB HDD x 12
Devices: sdc /dev/disk/by-path/pci-0000:3b:00.0-scsi-0:2:0:0
```

- Execute the `partNew` command with the following parameters to create partitions and mount points in the `etc/fstab` file.  
`name=sdc service=archiver volume=archiver commit=1`



5. Execute the `srvAlloc` command with the following parameters to allocate the space to the archiver service. This adds storage to the archiver service configuration and restarts the service every time it is executed.

```
service=archiver volume=archiver0 commit=1
```

## Configure Storage for Decoder using NW-PV-B/NW-PV-B-N

The following scenario configures storage on one, non-encrypted, 8-Drive PowerVault for a Network Decoder physical host.

1. Execute the `raidList` command.
  - a. Record the Controller Number, Enclosure Number, In Use, Drives, and Devices. You should see the following information.

```
In Use: FALSE
Devices: <empty>
```

b. Verify the Drive Count, Size, and Vendor.

The following example illustrates what you should see before you create a RAID array.

Properties for /appliance  
 raidList  Parameters:

Message Help

```
raidlist: list drive shelves attached to this appliance
security.roles: appliance.manage
```

---

/appliance?msg=raidList&force-content-type=text/plain

Output (or command manual help)

```
Controller 0 at PCI Address 18:00.0, Enclosure 64, SCSI Channel 2
Vendor: DP
Model: BP14G+EXP
In Use: true
Drives: 1.09 TB HDD x 2
 2.182 TB HDD x 2
Devices: sda /dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:0:0
 sdb /dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:1:0

Controller 1 at PCI Address 3b:00.0, Enclosure 251, SCSI Channel 2
Vendor: DELL
Model: MD1400
In Use: false
Drives: 7.277 TB HDD x 8
Devices:
```

- Execute the `raidNew` command with the following parameters using the controller number and the enclosure number you just recorded. Parameters for the first enclosure:  
`controller=1 enclosure=0 scheme=decoder commit=1`

| Scheme                | Enclosure Type | Drives Required        | Allocation                                                                                                                                  |
|-----------------------|----------------|------------------------|---------------------------------------------------------------------------------------------------------------------------------------------|
| decoder               | External       | 8 or 12 or 15 HDDs     | 3x drives in RAID 5 for decodersmall, all remaining drives in RAID 5 for decoder                                                            |
| log decoder           | External       | 8 or 12 or 15 HDDs     | Same as decoder configuration                                                                                                               |
| decoder hot spare     | External       | 8 or 12 or 15 HDDs     | 2x drives in RAID 1 for decodersmall, 1 drive as hot spare, all remaining drives in RAID 5 for decoder                                      |
| log decoder hot spare | External       | 8 or 12 or 15 HDDs     | Same as decoder hot spare configuration                                                                                                     |
| archiver              | External       | 8 or 12 or 15 HDDs     | All drives in RAID 6 for archiver database volume                                                                                           |
| packet expansion      | External       | 8 or 12 or 15 HDDs     | All drives in RAID 6 for decoder volume. No drives allocated for decodersmall.                                                              |
| network hybrid        | External       | 8 or 12 or 15 HDDs     | 3x drives in RAID 5 for meta expansion, all remaining drives in RAID 5 for packet expansion                                                 |
| network hybrid        | Internal       | S5/S6 hybrid drive set | 2x small HDD RAID 1 for decoder meta, 5x large HDD decoder, 3x large HDD concentrator, 2x SSD index                                         |
| log hybrid            | External       | 8 or 12 or 15 HDDs     | Half of the drives in RAID 5 for meta expansion, half the drives in RAID 5 for packet expansion                                             |
| log hybrid            | Internal       | S5/S6 hybrid drive set | 2x small HDD RAID 1 for decoder meta, 4x large HDD decoder, 4x large HDD concentrator, 2x SSD index                                         |
| endpoint hybrid       | Internal       | S5/S6 hybrid drive set | 2x small HDD RAID 1 for decoder meta, 4x large HDD RAID 10 for log decoder and endpoint, 4x large HDD RAID 5 for concentrator, 2x SSD index |
| log indexed decoder   | Internal       | S6E hybrid drive set   | 10x HDD RAID 6 for log decoder meta and packet, 2x SSD index                                                                                |

Properties for /appliance  
 raidNew Parameters:

Message Help

```
raidNew: allocate RAID devices in a drive shelf
security.roles: appliance.manage
parameters:
 controller - <uint32, {enum-one:The value must be one of the following: 0,1}> Controller the shelf is attached to
 enclosure - <uint32, optional, {enum-one:The value must be one of the following: 64|251}> Enclosure number of the shelf to clear. Required if
 scheme - <string, {enum-one:The value must be one of the following: decoder|logdecoder|concentrator|archiver|network-hybrid|log-hybrid|log-1-
 expansion|decoder-hotspare|logdecoder-hotspare}> Type of RAID volumes to allocate
```

/appliance?msg=raidNew&force-content-type=text/plain&controller=1&enclosure=251&scheme=decoder&commit=1

```
Output (or command manual help)

/opt/MegaRAID/perccli/perccli64 /c1 add vd r5 drives=251:0,251:1,251:2 ra Strip=128
CLI Version = 007.1623.0000.0000 May 17, 2021
Operating system = Linux 3.10.0-1160.66.1.el7.x86_64
Controller = 1
Status = Success
Description = Add VD Succeeded.

/opt/MegaRAID/perccli/perccli64 /c1 add vd r5 drives=251:3,251:4,251:5,251:6,251:7 ra Strip=128
CLI Version = 007.1623.0000.0000 May 17, 2021
Operating system = Linux 3.10.0-1160.66.1.el7.x86_64
Controller = 1
Status = Success
Description = Add VD Succeeded.
```

```
[root@decoder94 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 1.1T 0 disk
├─sda1 8:1 0 1G 0 part /boot
├─sda2 8:2 0 1.1T 0 part
│ └─netwitness_vg00-root 253:0 0 30G 0 lvm /
│ └─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
│ └─netwitness_vg00-nwhome 253:2 0 3.2T 0 lvm /var/netwitness
│ └─netwitness_vg00-varlog 253:3 0 10G 0 lvm /var/log
│ └─netwitness_vg00-usrhome 253:4 0 10G 0 lvm /home
sdb 8:16 0 2.2T 0 disk
├─sdb1 8:17 0 2.2T 0 part
│ └─netwitness_vg00-nwhome 253:2 0 3.2T 0 lvm /var/netwitness
sdc 8:32 0 14.6T 0 disk
sdd 8:48 0 29.1T 0 disk

[root@decoder94 ~]#
```

- Execute the `partNew` command to create the **decodersmall** partition first (decoder dir, index, metadb, sessiondb) (First Enclosure, SDC, SDD) with the following parameters.

name=sdcc service=decoder volume=decodersmall commit=1

Properties for /appliance

partNew Parameters: name=sdcc service=decoder volume=decodersmall commit=1

Message Help

```
partNew: make partitions on a block device
security.roles: appliance.manage
parameters:
 name - <string, {enum-one:The value must be one of the following: sdc|sdd}> block device name
 service - <string, {enum-one:The value must be one of the following: archiver|concentrator|decoder|logdecoder}> service that will use storage
 volume - <string, optional, {enum-one:The value must be one of the following: concentrator|index|decodersmall|decoder|packet|hybrid-decoder-meta|logdecodersma
logdecoder-meta|archiver|hybrid-concentrator|endpoint-log-hybrid|log-indexed-decoder|logindex}> volume to create
```

/appliance?msg=partNew&force-content-type=text/plain&name=sdcc&service=decoder&volume=decodersmall&commit=1

Output (or command manual help)

```
/sbin/parted -s /dev/sdc mklabel gpt
/sbin/parted -s -a optimal /dev/sdc mkpart LVM 0% 100%
/sbin/pvcreate -f /dev/sdc1
Physical volume "/dev/sdc1" successfully created.
/sbin/vgcreate -f decodersmall /dev/sdc1
Volume group "decodersmall" successfully created
/sbin/lvcreate -y -n decoroot -L 10G decodersmall
Logical volume "decoroot" created.
/sbin/mkfs.xfs /dev/decodersmall/decoroot
meta-data=/dev/decodersmall/decoroot isize=512 agcount=16, agsize=163808 blks
 = sectsz=512 attr=2, projid32bit=1
 = crc=1 finobt=0, sparse=0
data = bsize=4096 blocks=2620928, imaxpct=25
 = sunit=32 swidth=64 blks
naming =version 2 bsize=4096 ascii-ci=0 ftype=1
log =internal log bsize=4096 blocks=2560, version=2
 = sectsz=512 sunit=32 blks, lazy-count=1
realtime =none extsz=4096 blocks=0, rtextents=0
/sbin/mkdir -p /var/netwitness/decoder
/sbin/mount /var/netwitness/decoder
/sbin/lvcreate -y -n index -L 30G decodersmall
Wiping xfs signature on /dev/decodersmall/index.
```

```
[root@decoder94 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 1.1T 0 disk
├─sda1 8:1 0 1G 0 part /boot
├─sda2 8:2 0 1.1T 0 part
│ ├─netwitness_vg00-root 253:0 0 30G 0 lvm /
│ ├─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
│ ├─netwitness_vg00-nwhome 253:2 0 3.2T 0 lvm /var/netwitness
│ ├─netwitness_vg00-varlog 253:3 0 10G 0 lvm /var/log
│ └─netwitness_vg00-usrhome 253:4 0 10G 0 lvm /home
├─sdb 8:16 0 2.2T 0 disk
├─sdb1 8:17 0 2.2T 0 part
│ └─netwitness_vg00-nwhome 253:2 0 3.2T 0 lvm /var/netwitness
├─sdc 8:32 0 14.6T 0 disk
├─sdc1 8:33 0 14.6T 0 part
│ ├─decodersmall-decoroot 253:5 0 10G 0 lvm /var/netwitness/decoder
│ ├─decodersmall-index 253:6 0 30G 0 lvm /var/netwitness/decoder/index
│ ├─decodersmall-sessiondb 253:7 0 600G 0 lvm /var/netwitness/decoder/sessiondb
│ └─decodersmall-metadb 253:8 0 14T 0 lvm /var/netwitness/decoder/metadb
└─sdd 8:48 0 29.1T 0 disk
```

- Execute the `partNew` command to create the decoder volume (packetdb) (First Enclosure, SDC, SDD) with the following parameters.

name=sdd service=decoder volume=decoder commit=1

Properties for /appliance

partNew  Parameters: name=sdd service=decoder volume=decoder commit=1

Message Help

```
partNew: make partitions on a block device
security.roles: appliance.manage
parameters:
 name - <string, {enum-one:The value must be one of the following: sdc|sdd}> block device name
 service - <string, {enum-one:The value must be one of the following: archiver|concentrator|decoder|logdecoder}> service that will use storage
 volume - <string, optional, {enum-one:The value must be one of the following: concentrator|index|decodersmall|decoder|packet|hybrid-decoder-m
logdecoder-meta|archiver|hybrid-concentrator|endpoint-log-hybrid|log-indexed-decoder|logindex}> volume to create
```

/appliance?msg=partNew&force-content-type=text/plain&name=sdd&service=decoder&volume=decoder&commit=1

Output (or command manual help)

```
/sbin/parted -s /dev/sdd mklabel gpt
/sbin/parted -s -a optimal /dev/sdd mkpart LVM 0% 100%
/sbin/pvcreate -f /dev/sdd1
Wiping xfs signature on /dev/sdd1.
Physical volume "/dev/sdd1" successfully created.
/sbin/vgcreate -f decoder /dev/sdd1
Volume group "decoder" successfully created
/sbin/lvcreate -y -n packetdb -l 100%FREE decoder
Logical volume "packetdb" created.
/sbin/mkfs.xfs /dev/decoder/packetdb
meta-data=/dev/decoder/packetdb isize=512 agcount=32, agsize=244170720 blks
= sectsz=512 attr=2, projid32bit=1
= crc=1 finobt=0, sparse=0
data = bsize=4096 blocks=7813463040, imaxpct=5
= sunit=32 swidth=128 blks
naming =version 2 bsize=4096 ascii-ci=0 ftype=1
log =internal log bsize=4096 blocks=521728, version=2
= sectsz=512 sunit=32 blks, lazy-count=1
realtime=none extsz=4096 blocks=0, rtextents=0
/bin/mkdir -p /var/netwitness/decoder/packetdb
/bin/mount /var/netwitness/decoder/packetdb
```

```
[root@decoder94 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 1.1T 0 disk
├─sda1 8:1 0 1G 0 part /boot
├─sda2 8:2 0 1.1T 0 part
│ ├─netwitness_vg00-root 253:0 0 30G 0 lvm /
│ ├─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
│ ├─netwitness_vg00-nwhome 253:2 0 3.2T 0 lvm /var/netwitness
│ ├─netwitness_vg00-varlog 253:3 0 10G 0 lvm /var/log
│ └─netwitness_vg00-usrhome 253:4 0 10G 0 lvm /home
└─sdb 8:16 0 2.2T 0 disk
 ├─sdb1 8:17 0 2.2T 0 part
 └─netwitness_vg00-nwhome 253:2 0 3.2T 0 lvm /var/netwitness
sdc 8:32 0 14.6T 0 disk
├─sdc1 8:33 0 14.6T 0 part
│ ├─decodersmall-decoroot 253:5 0 10G 0 lvm /var/netwitness/decoder
│ ├─decodersmall-index 253:6 0 30G 0 lvm /var/netwitness/decoder/index
│ ├─decodersmall-sessiondb 253:7 0 600G 0 lvm /var/netwitness/decoder/sessiondb
│ └─decodersmall-metadb 253:8 0 14T 0 lvm /var/netwitness/decoder/metadb
└─sdd 8:48 0 29.1T 0 disk
 └─sdd1 8:49 0 29.1T 0 part
 └─decoder-packetdb 253:9 0 29.1T 0 lvm /var/netwitness/decoder/packetdb
[root@decoder94 ~]#
```

```
[root@decoder94 ~]# df -kh
Filesystem Size Used Avail Use% Mounted on
devtmpfs 63G 0 63G 0% /dev
tmpfs 63G 40K 63G 1% /dev/shm
tmpfs 63G 11M 63G 1% /run
tmpfs 63G 0 63G 0% /sys/fs/cgroup
/dev/mapper/netwitness_vg00-root 30G 3.7G 27G 13% /
/dev/sdal 1014M 122M 893M 12% /boot
/dev/mapper/netwitness_vg00-nwhome 3.3T 543M 3.3T 1% /var/netwitness
/dev/mapper/netwitness_vg00-varlog 10G 138M 9.9G 2% /var/log
/dev/mapper/netwitness_vg00-usrhome 10G 33M 10G 1% /home
tmpfs 13G 0 13G 0% /run/user/0
/dev/mapper/decodersmall-decoroot 10G 33M 10G 1% /var/netwitness/decoder
/dev/mapper/decodersmall-index 30G 33M 30G 1% /var/netwitness/decoder/index
/dev/mapper/decodersmall-sessiondb 600G 34M 600G 1% /var/netwitness/decoder/sessiondb
/dev/mapper/decodersmall-metadb 14T 34M 14T 1% /var/netwitness/decoder/metadb
/dev/mapper/decoder-packetdb 30T 34M 30T 1% /var/netwitness/decoder/packetdb
[root@decoder94 ~]#
```

- Execute the `srvAlloc` command with the following parameters to add the storage information into the Service Configuration settings.

- `service=decoder volume=decodersmall commit=1`

Properties for /appliance

srvAlloc Parameters:

Message Help

```
srvAlloc: apply volume group storage to a service on this appliance
security.roles: appliance.manage
parameters:
 service - <string, {enum-one:The value must be one of the following: archiver|concentrator|decoder|logdecoder}> service that will use storage
 volume - <string, {enum-one:The value must be one of the following: decoder|decodersmall|netwitness_vg00}> volume group name
 commit - <bool, optional> commit changes
```

/appliance?msg=srvAlloc&force-content-type=text/plain&service=decoder&volume=decodersmall&commit=1

Output (or command manual help)

```
Set /database/config/meta.dir to /var/netwitness/decoder/metadb==13.23 TB
Set /database/config/session.dir to /var/netwitness/decoder/sessiondb==569.72 GB
Set /index/config/index.dir to /var/netwitness/decoder/index==28.48 GB
```

- `service=decoder volume=decoder commit=1`

Properties for /appliance

srvAlloc Parameters:

Message Help

```
srvAlloc: apply volume group storage to a service on this appliance
security.roles: appliance.manage
parameters:
 service - <string, {enum-one:The value must be one of the following: archiver|concentrator|decoder|logdecoder}> service that will use storage
 volume - <string, {enum-one:The value must be one of the following: decoder|decodersmall|netwitness_vg00}> volume group name
 commit - <bool, optional> commit changes
```

/appliance?msg=srvAlloc&force-content-type=text/plain&service=decoder&volume=decoder&commit=1

Output (or command manual help)

```
Set /database/config/packet.dir to /var/netwitness/decoder/packetdb==27.65 TB
```



## Configure Storage for Concentrator using NW-PV-C/NW-PV-C-N

The following scenario configures storage on one, non-encrypted, 8-Drive PowerVault for a Network Concentrator physical host.

1. Execute the `raidList` command.
  - a. Record the Controller Number, Enclosure Number, In Use, Drives, and Devices.  
You should see the following information.

```
In Use: FALSE
Devices: <empty>
```

- b. Verify the Drive Count, Size, and Vendor.

The following example illustrates what you should see before you create a RAID array.

Output (or command manual help)

```
Controller 0 at PCI Address 18:00.0, Enclosure 64, SCSI Channel 2
Vendor: DP
Model: BP14G+EXP
In Use: true
Drives: 1.09 TB HDD x 2
 2.182 TB HDD x 2
Devices: sda /dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:0:0
 sdb /dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:1:0

Controller 1 at PCI Address 3b:00.0, Enclosure 251, SCSI Channel 2
Vendor: DELL
Model: MD1400
In Use: false
Drives: 1.746 TB SSD x 2
 7.277 TB HDD x 6
Devices:
```

2. Execute the `raidNew` command with the following parameters using the controller number and the enclosure number you just recorded.

```
controller=1 enclosure=251 scheme=concentrator commit=1
```

The following example illustrates what you should see after you create a RAID array.

Properties for /appliance  
raidNew Parameters: controller=1 enclosure=251 scheme=concentrator commit=1

Message Help  
scheme - <string, (enum-one:The value must be one of the following: decoder|logdecoder|concentrator|archiver|network-hybrid|log-hy  
expansion|decoder-hotspare|logdecoder-hotspare)> Type of RAID volumes to allocate  
preferSecure - <bool, optional, {bool:The value must be one of the following acceptable boolean values: 0,1,yes,no,true,false,on,o  
drives and a controller with a security key set  
commit - <bool, optional> commit changes

/appliance?msg=raidNew&force-content-type=text/plain&controller=1&enclosure=251&scheme=concentrator&commit=1

Output (or command manual help)

```
/opt/MegaRAID/percc11/percc1164 /c1 add vd r1 drives=251:0,251:1 ra Strip=128
CLI Version = 007.1623.0000.0000 May 17, 2021
Operating system = Linux 3.10.0-1160.66.1.el7.x86_64
Controller = 1
Status = Success
Description = Add VD Succeeded.

/opt/MegaRAID/percc11/percc1164 /c1 add vd r6 drives=251:2,251:3,251:4,251:5,251:6,251:7 ra Strip=128
CLI Version = 007.1623.0000.0000 May 17, 2021
Operating system = Linux 3.10.0-1160.66.1.el7.x86_64
Controller = 1
Status = Success
Description = Add VD Succeeded.
```

```
[root@conc95 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 1.1T 0 disk
├─sda1 8:1 0 1G 0 part /boot
├─sda2 8:2 0 1.1T 0 part
│ ├─netwitness_vg00-root 253:0 0 30G 0 lvm /
│ ├─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
│ ├─netwitness_vg00-nwhome 253:2 0 3.2T 0 lvm /var/netwitness
│ └─netwitness_vg00-varlog 253:3 0 10G 0 lvm /var/log
└─netwitness_vg00-usrhome 253:4 0 10G 0 lvm /home
sdb 8:16 0 2.2T 0 disk
├─sdb1 8:17 0 2.2T 0 part
│ └─netwitness_vg00-nwhome 253:2 0 3.2T 0 lvm /var/netwitness
sdc 8:32 0 1.8T 0 disk
sdd 8:48 0 29.1T 0 disk
[root@conc95 ~]#
```

- Execute the `partNew` command to create the **concentrator** partition first with the following parameters. You must create the **concentrator** volume before **index** volume or it will fail.

```
name=sdd service=concentrator volume=concentrator commit=1
```

Properties for /appliance  
partNew Parameters: name=sdd service=concentrator volume=concentrator commit=1

Message Help

```
partNew: make partitions on a block device
security.roles: appliance.manage
parameters:
 name - <string, {enum-one:The value must be one of the following: sdc|sdd}> block device name
 service - <string, {enum-one:The value must be one of the following: archiver|concentrator|decoder|logdecoder}> service that will use storage
 volume - <string, optional, {enum-one:The value must be one of the following: concentrator|index|decodersmall|decoder|packet|hybrid-decoder-meta|logdecodersma
logdecoder-meta|archiver|hybrid-concentrator|endpoint-log-hybrid|log-indexed-decoder|logindex}> volume to create
```

/appliance?msg=partNew&force-content-type=text/plain&name=sdd&service=concentrator&volume=concentrator&commit=1

Output (or command manual help)

```
/sbin/parted -s /dev/sdd mklabel gpt
/sbin/parted -s -a optimal /dev/sdd mkpart LVM 0% 100%
/sbin/pvcreate -f /dev/sdd1
Physical volume "/dev/sdd1" successfully created.
/sbin/vgcreate -f concentrator /dev/sdd1
Volume group "concentrator" successfully created
/sbin/lvcreate -y -n root -L 30G concentrator
Wiping xfs signature on /dev/concentrator/root.
Logical volume "root" created.
/sbin/mkfs.xfs /dev/concentrator/root
meta-data=/dev/concentrator/root isize=512 agcount=16, agsize=491488 blks
=
= sectsz=512 attr=2, projid32bit=1
=
= crc=1 finobt=0, sparse=0
data =
= bsize=4096 blocks=7863808, imaxpct=25
=
= sunit=32 swidth=128 blks
naming =version 2
=
= bsize=4096 ascii-ci=0 ftype=1
```

#### 4. Execute the partNew command with the following parameters to create an **index** on SSDs.

```
name=sdc service=concentrator volume=index commit=1
```

Properties for /appliance  
partNew Parameters: name=sdc service=concentrator volume=index commit=1

Message Help

```
partNew: make partitions on a block device
security.roles: appliance.manage
parameters:
 name - <string, {enum-one:The value must be one of the following: sdc|sdd}> block device name
 service - <string, {enum-one:The value must be one of the following: archiver|concentrator|decoder|logdecoder}> service that will use storage
 volume - <string, optional, {enum-one:The value must be one of the following: concentrator|index|decodersmall|decoder|packet|hybrid-decoder-meta|logdecod
logdecoder-meta|archiver|hybrid-concentrator|endpoint-log-hybrid|log-indexed-decoder|logindex}> volume to create
```

/appliance?msg=partNew&force-content-type=text/plain&name=sdc&service=concentrator&volume=index&commit=1

Output (or command manual help)

```
/sbin/parted -s /dev/sdc mklabel gpt
/sbin/parted -s -a optimal /dev/sdc mkpart LVM 0% 100%
/sbin/pvcreate -f /dev/sdc1
Physical volume "/dev/sdc1" successfully created.
/sbin/vgcreate -f index /dev/sdc1
Volume group "index" successfully created
/sbin/lvcreate -y -n index -l 100%RES index
Wiping xfs signature on /dev/index/index.
Logical volume "index" created.
/sbin/mkfs.xfs /dev/index/index
meta-data=/dev/index/index isize=512 agcount=32, agsize=14646240 blks
=
= sectsz=512 attr=2, projid32bit=1
=
= crc=1 finobt=0, sparse=0
data =
= bsize=4096 blocks=468679680, imaxpct=5
=
= sunit=32 swidth=32 blks
naming =version 2
=
= bsize=4096 ascii-ci=0 ftype=1
log =internal log
=
= sectsz=512 sunit=32 blks, lazy-count=1
realtime =none
=
= extsz=4096 blocks=0, rtextents=0
/sbin/mkdir -p /var/netwitness/concentrator/index
/sbin/mount /var/netwitness/concentrator/index
```

```
[root@con95 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 1.1T 0 disk
├─sda1 8:1 0 1G 0 part /boot
├─sda2 8:2 0 1.1T 0 part
│ └─netwitness_vg00-root 253:0 0 30G 0 lvm /
│ └─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
│ └─netwitness_vg00-nwhome 253:2 0 3.2T 0 lvm /var/netwitness
│ └─netwitness_vg00-varlog 253:3 0 10G 0 lvm /var/log
│ └─netwitness_vg00-usrhome 253:4 0 10G 0 lvm /home
sdb 8:16 0 2.2T 0 disk
├─sdb1 8:17 0 2.2T 0 part
│ └─netwitness_vg00-nwhome 253:2 0 3.2T 0 lvm /var/netwitness
sdc 8:32 0 1.8T 0 disk
├─sdc1 8:33 0 1.8T 0 part
│ └─index-index 253:8 0 1.8T 0 lvm /var/netwitness/concentrator/index
sdd 8:48 0 29.1T 0 disk
├─sdd1 8:49 0 29.1T 0 part
│ └─concentrator-root 253:5 0 30G 0 lvm /var/netwitness/concentrator
│ └─concentrator-sessiondb 253:6 0 2.9T 0 lvm /var/netwitness/concentrator/sessiondb
│ └─concentrator-metadb 253:7 0 26.2T 0 lvm
```

#### 5. Execute the srvAlloc command with the following parameters.

```
service=concentrator volume=concentrator commit=1
```

## Configure Storage for Concentrator using NW-PV-D/NW-PV-D-N

The following scenario configures storage on one, non-encrypted, 12-Drive PowerVault for a Log Decoder Hybrid physical host.

1. Execute the `raidList` command.
  - a. Record the Controller Number, Enclosure Number, In Use, Drives, and Devices.  
You should see the following information.

In Use: FALSE

Devices: <empty>

- b. Verify the Drive Count, Size, and Vendor.

The following example illustrates what you should see before you create a RAID array.

Properties for /appliance  
 Parameters:

Message Help

```
raidList: list drive shelves attached to this appliance
security.roles: appliance.manage
```

</appliance?msg=raidList&force-content-type=text/plain>

Output (or command manual help)

```
Controller 0 at PCI Address 02:00.0, Enclosure 32
Vendor: DP
Model: BP13G+EXP
In Use: true
Drives: 931.512 GB HDD x 2
 1.819 TB HDD x 2
Devices: sda /dev/disk/by-path/pci-0000:02:00.0-scsi-0:2:0:0
 sdb /dev/disk/by-path/pci-0000:02:00.0-scsi-0:2:1:0

Controller 1 at PCI Address 03:00.0, Enclosure 108
Vendor: DELL
Model: MD1400
In Use: false
Drives: 1.455 TB SSD x 3
 10.692 TB HDD x 9
Devices:
```

- Execute the `raidNew` command with the following parameters using the controller number and the enclosure number you just recorded.

```
controller=1 enclosure=108 scheme=concentrator commit=1
```

The following example illustrates what you should see after you create a RAID array.

Properties for /appliance  
raidNew Parameters: controller=1 enclosure=108 scheme=concentrator commit=1 Send

Message Help

```
raidNew: allocate RAID devices in a drive shelf
security.roles: appliance.manage
parameters:
 controller - <uint32, {enum-one:The value must be one of the following: 0,1}> Controller the shelf is attached to
 enclosure - <uint32, optional, {enum-one:The value must be one of the following: 32|108}> Enclosure number of the shelf to clear. Required if
 scheme - <string, {enum-one:The value must be one of the following: decoder|logdecoder|concentrator|archiver|network-hybrid|log-hybrid|log-in
 expansion|decoder-hotspare|logdecoder-hotspare}> Type of RAID volumes to allocate
```

/appliance?msg=raidNew&force-content-type=text/plain&controller=1&enclosure=108&scheme=concentrator&commit=1

Output (or command manual help)

```
/opt/MegaRAID/perccli/perccli64 /c1 add vd r5 drives=108:0,108:1,108:2 ra Strip=128
Controller = 1
Status = Success
Description = Add VD Succeeded

/opt/MegaRAID/perccli/perccli64 /c1 add vd r6 drives=108:3,108:4,108:5,108:6,108:7,108:8,108:9,108:10,108:11 ra Strip=128
Controller = 1
Status = Success
Description = Add VD Succeeded
```

```
[root@Concentrator132 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 931G 0 disk
├─sda1 8:1 0 1G 0 part /boot
├─sda2 8:2 0 930G 0 part
│ └─netwitness_vg00-root 253:0 0 30G 0 lvm /
│ └─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
│ └─netwitness_vg00-nwhome 253:2 0 2.7T 0 lvm /var/netwitness
│ └─netwitness_vg00-varlog 253:3 0 10G 0 lvm /var/log
│ └─netwitness_vg00-usrhome 253:4 0 10G 0 lvm /home
sdb 8:16 0 1.8T 0 disk
├─sdb1 8:17 0 1.8T 0 part
│ └─netwitness_vg00-nwhome 253:2 0 2.7T 0 lvm /var/netwitness
sdc 8:32 0 2.9T 0 disk
sdd 8:48 0 74.9T 0 disk
[root@Concentrator132 ~]#
```

- Execute the `partNew` command to create the **concentrator** partition first with the following parameters. You must create the concentrator volume before index volume or it will fail.

```
name=sdd service=concentrator volume=concentrator commit=1
```

```

Properties for /appliance
partNew Parameters: name=sdd service=concentrator volume=concentrator commit=1 Send

Message Help
For help with the parameters of the partNew service
security.roles: appliance.manage
parameters:
 name - <string, {enum-one:The value must be one of the following: sdc|sdd}> block device name
 service - <string, {enum-one:The value must be one of the following: archiver|concentrator|decoder|logdecoder}> service that will use storage
 volume - <string, optional, {enum-one:The value must be one of the following: concentrator|index|decodersmall|decoder|packet|hybrid-decoder-meta|logdecodersmall|logdecoder-meta|archiver|hybrid-concentrator|endpoint-log-hybrid|log-indexed-decoder|logindex}> volume to create
 commit - <bool, optional> commit changes

/appliance?msg=partNew&force-content-type=text/plain&name=sdd&service=concentrator&volume=concentrator&commit=1

Output (or command manual help)

/sbin/parted -s /dev/sdd mklabel gpt
/sbin/parted -s -a optimal /dev/sdd mkpart LVM 0% 100%
/sbin/pvcreate -f /dev/sdd1
Physical volume "/dev/sdd1" successfully created.
/sbin/vgcreate -f concentrator /dev/sdd1
Volume group "concentrator" successfully created
/sbin/lvcreate -y -n root -L 30G concentrator
Wiping xfs signature on /dev/concentrator/root.
Logical volume "root" created.
/sbin/mkfs.xfs /dev/concentrator/root
meta-data=/dev/concentrator/root isize=512 agcount=4, agsize=1966080 blks
 = sectsz=4096 attr=2, projid32bit=1
 = crc=1 finobt=0, sparse=0
data = bsize=4096 blocks=7864320, imaxpct=25
 = sunit=0 swidth=0 blks
naming =version 2 bsize=4096 ascii-ci=0 ftype=1
log =internal log bsize=4096 blocks=3840, version=2
 = sectsz=4096 sunit=1 blks, lazy-count=1
realtime =none extsz=4096 blocks=0, rtextents=0
/bin/mkdir -p /var/netwitness/concentrator
/bin/mount /var/netwitness/concentrator
/sbin/lvcreate -y -n sessiondb -l 10%FREE concentrator

[root@Concentrator132 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 931G 0 disk
├─sda1 8:1 0 1G 0 part /boot
└─sda2 8:2 0 930G 0 part
 ├─netwitness_vg00-root 253:0 0 30G 0 lvm /
 ├─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
 ├─netwitness_vg00-nwhome 253:2 0 2.7T 0 lvm /var/netwitness
 ├─netwitness_vg00-varlog 253:3 0 10G 0 lvm /var/log
 └─netwitness_vg00-usrhome 253:4 0 10G 0 lvm /home
sdb 8:16 0 1.8T 0 disk
├─sdb1 8:17 0 1.8T 0 part
└─netwitness_vg00-nwhome 253:2 0 2.7T 0 lvm /var/netwitness
sdc 8:32 0 2.9T 0 disk
sdd 8:48 0 74.9T 0 disk
├─sdd1 8:49 0 74.9T 0 part
│ ├─concentrator-root 253:5 0 30G 0 lvm /var/netwitness/concentrator
│ ├─concentrator-sessiondb 253:6 0 7.5T 0 lvm /var/netwitness/concentrator/sessiondb
│ └─concentrator-metadb 253:7 0 67.3T 0 lvm /var/netwitness/concentrator/metadb

```

- Execute the `partNew` command with the following parameters to create an **index** on SSDs.  
`name=sdc service=concentrator volume=index commit=1`

```

Properties for /appliance
partNew Parameters: name=sdsc service=concentrator volume=index commit=1 Send

Message Help
partNew: name: partitions on a server device
security.roles: appliance.manage
parameters:
 name - <string, {enum-one:The value must be one of the following: sdc|sdd}> block device name
 service - <string, {enum-one:The value must be one of the following: archiver|concentrator|decoder|logdecoder}> service that will use storage
 volume - <string, optional, {enum-one:The value must be one of the following: concentrator|index|decodersmall|decoder|packet|hybrid-decoder-meta|logdecoder-meta|archiver|hybrid-concentrator|endpoint-log-hybrid|log-indexed-decoder|logindex}> volume to create
 commit - <bool, optional> commit changes

/appliance?msg=partNew&force-content-type=text/plain&name=sdsc&service=concentrator&volume=index&commit=1

Output (or command manual help)
/sbin/parted -s /dev/sdc mklabel gpt
/sbin/parted -s -a optimal /dev/sdc mkpart LVM 0% 100%
/sbin/pvcreate -f /dev/sdc1
Physical volume "/dev/sdc1" successfully created.
/sbin/vgcreate -f index /dev/sdc1
Volume group "index" successfully created
/sbin/lvcreate -y -n index -l 100%FREE index
Wiping xfs signature on /dev/index/index.
Logical volume "index" created.
/sbin/mkfs.xfs /dev/index/index
meta-data=/dev/index/index isize=512 agcount=4, agsize=195280640 blks
= sectsz=4096 attr=2, projid32bit=1
= crc=1 finobt=0, sparse=0
data = bsize=4096 blocks=781122560, imaxpct=5
= sunit=0 swidth=0 blks
naming =version 2 bsize=4096 ascii-ci=0 ftype=1
log =internal log bsize=4096 blocks=381407, version=2
= sectsz=4096 sunit=1 blks, lazy-count=1
realtime=none extsz=4096 blocks=0, rtxents=0
/bin/mkdir -p /var/netwitness/concentrator/index
/bin/mount /var/netwitness/concentrator/index

[root@Concentrator132 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 931G 0 disk
├─sda1 8:1 0 1G 0 part /boot
├─sda2 8:2 0 930G 0 part
│ └─netwitness_vg00-root 253:0 0 30G 0 lvm /
│ └─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
│ └─netwitness_vg00-nwhome 253:2 0 2.7T 0 lvm /var/netwitness
│ └─netwitness_vg00-varlog 253:3 0 10G 0 lvm /var/log
│ └─netwitness_vg00-usrhome 253:4 0 10G 0 lvm /home
sdb 8:16 0 1.8T 0 disk
├─sdb1 8:17 0 1.8T 0 part
└─netwitness_vg00-nwhome 253:2 0 2.7T 0 lvm /var/netwitness
sdc 8:32 0 2.9T 0 disk
├─sdc1 8:33 0 2.9T 0 part
└─index-index 253:8 0 2.9T 0 lvm /var/netwitness/concentrator/index
sdd 8:48 0 74.9T 0 disk
├─sdd1 8:49 0 74.9T 0 part
│ └─concentrator-root 253:5 0 30G 0 lvm /var/netwitness/concentrator
│ └─concentrator-sessiondb 253:6 0 7.5T 0 lvm /var/netwitness/concentrator/sessiondb
│ └─concentrator-metadb 253:7 0 67.3T 0 lvm /var/netwitness/concentrator/metadb

[root@Concentrator132 ~]#

[root@conc95 ~]# df -kh
Filesystem Size Used Avail Use% Mounted on
devtmpfs 63G 0 63G 0% /dev
tmpfs 63G 420K 63G 1% /dev/shm
tmpfs 63G 43M 63G 1% /run
tmpfs 63G 0 63G 0% /sys/fs/cgroup
/dev/mapper/netwitness_vg00-root 30G 3.6G 27G 12% /
/dev/sdal 1014M 122M 893M 12% /boot
/dev/mapper/netwitness_vg00-varlog 10G 156M 9.9G 2% /var/log
/dev/mapper/netwitness_vg00-usrhome 10G 33M 10G 1% /home
/dev/mapper/netwitness_vg00-nwhome 3.3T 494M 3.3T 1% /var/netwitness
/dev/mapper/concentrator-root 30G 61M 30G 1% /var/netwitness/concentrator
/dev/mapper/concentrator-sessiondb 3.0T 34M 3.0T 1% /var/netwitness/concentrator/sessiondb
/dev/mapper/concentrator-metadb 27T 34M 27T 1% /var/netwitness/concentrator/metadb
tmpfs 13G 0 13G 0% /run/user/0

[root@conc95 ~]#

```

- Execute the `srvAlloc` command with the following parameters.  
`service=concentrator volume=concentrator commit=1`

## Appendix F. Sample Storage Configuration Scenarios for Meta Disk Kits

This appendix illustrates the examples of how to configure a Meta Disk Kit described in Configure Block devices for Drive pack section for Network Decoder in the below scenarios.

- [Meta-Only: No Externally Attached Storage](#)
- [Maximize PowerVault Storage capacity for Network Decoder](#)

### Meta-Only (One kit – 3 SED drives configured as RAID 5)

**Configure one Meta Disk kit as decodersmall volume for Network Decoder:**

**Note:**

1. The configuration for Log Decoder is the similar as that of a Network Decoder. Substitute the service and volume names that correspond to Log Decoder.
2. A single Meta Disk pack is configured as RAID5 (3 drives) and two Meta Disk packs (6 drives) are configured as RAID6.
3. When configuring two Meta Disk Packs or adding the second Meta Disk pack, use appropriate disk slot numbers when creating the virtual drive (step 5 below).

On the Series 6 (Dell R640) appliance, the Meta Drive Pack disks are installed in slots 4, 5 and 6. The virtual drive configuration requires identifying the controller ID and Enclosure ID (EID). On Series 6 appliance, the controller ID and Enclosure IDs are 0 and 64. However, the `nwraidtool.py` script that is installed on every server can help to confirm these ID numbers, or the `PercCLI` commands as shown in the below steps.

**Identify controller/enclosures:**

1. Identify the controller ID (Ctl) for the internal controller (PERC H740P Mini/ PERC H750). In the below figure the controller ID is 0 and highlighted in yellow. The drive count is displayed under PDs.

```
/opt/MegaRAID/perccli/perccli64 show
```



```
[root@117Decoder ~]# /opt/MegaRAID/perccli/perccli64 show
Status Code = 0
Status = Success
Description = None

Number of Controllers = 2
Host Name = 117Decoder
Operating System = Linux 3.10.0-1160.36.2.el7.x86_64
StoreLib IT Version = 07.0400.0200.0400
StoreLib IR3 Version = 15.50-0

System Overview :
=====

Ctl Model Ports PDS DGs DNOpt VDs VNOpt BBU sPR DS EHS ASOs Hlth

0 PERCH740PMini 8 - 1 0 1 0 Opt On - N 0 Opt
1 PERCH840Adapter 8 24 1 0 1 0 Opt On - N 0 Opt

Ctl=Controller Index|DGs=Drive groups|VDs=Virtual drives|Fld=Failed
PDS=Physical drives|DNOpt=DG NotOptimal|VNOpt=VD NotOptimal|Opt=Optimal
Msng=Missing|Dgd=Degraded|NdAtn=Need Attention|Unkwn=Unknown
sPR=Scheduled Patrol Read|DS=DimmerSwitch|EHS=Emergency Hot Spare
Y=Yes|N=No|ASOs=Advanced Software Options|BBU=Battery backup unit
Hlth=Health|Safe=Safe-mode boot
```

2. Identify the Enclosure ID (EID) for controller ‘0’. In this case the EID is ‘64’ and highlighted in yellow.

```
/opt/MegaRAID/perccli/perccli64 /c0 /eall show
```

```
[root@117Decoder ~]# /opt/MegaRAID/perccli/perccli64 /c0 /eall show
Controller = 0
Status = Success
Description = None

Properties :
=====

EID State Slots PD PS Fans TSs Alms SIM Port# ProdID VendorSpecific

64 OK 10 - 0 0 0 0 1 00 & 00 x8 BP14G+EXP +

EID-Enclosure Device ID |PD-Physical drive count |PS-Power Supply count|
TSs-Temperature sensor count |Alms-Alarm count |SIM-SIM Count
```

3. Identify the new Meta Disk kit disk slot numbers (Slots 4 through 9) on the controller PERC H740P Mini. These drives do not belong to any Drive Group (DG). The DG column for these drives display a dash ‘-’, the State column shows ‘UGood’, and the SED value as ‘Y’. See the entries highlighted in yellow.

```
[root@116S6Core1 ~]# /opt/MegaRAID/perccli/perccli64 /c0 /eall /sall show
Controller = 0
Status = Success
Description = Show Drive Information Succeeded.
```

```
Drive Information :
```

```
=====
```

```

EID:Slt DID State DG Size Intf Med SED PI SeSz Model Sp

64:0 0 Onln 0 1.090 TB SAS HDD Y N 512B ST1200MM0069 U
64:1 1 Onln 0 1.090 TB SAS HDD Y N 512B ST1200MM0069 U
64:2 2 Onln 1 2.182 TB SAS HDD Y N 512B ST2400MM0149 U
64:3 3 Onln 1 2.182 TB SAS HDD Y N 512B ST2400MM0149 U
64:4 4 UGood - 2.182 TB SAS HDD Y N 512B ST2400MM0149 U
64:5 5 UGood - 2.182 TB SAS HDD Y N 512B ST2400MM0149 U
64:6 6 UGood - 2.182 TB SAS HDD Y N 512B ST2400MM0149 U
64:7 7 UGood - 2.182 TB SAS HDD Y N 512B ST2400MM0149 U
64:8 8 UGood - 2.182 TB SAS HDD Y N 512B ST2400MM0149 U
64:9 9 UGood - 2.182 TB SAS HDD Y N 512B ST2400MM0149 U

```

```
EID-Enclosure Device ID|Slt-Slot No.|DID-Device ID|DG-DriveGroup
DHS-Dedicated Hot Spare|UGood-Unconfigured Good|GHS-Global Hotspare
UBad-Unconfigured Bad|Onln-Online|Ofln-Offline|Intf-Interface
Med-Media Type|SED-Self Encryptive Drive|PI-Protection Info
SeSz-Sector Size|Sp-Spun|U-Up|D-Down/PowerSave|T-Transition|F-Foreign
UGUnsp-Unsupported|UGShld-UnConfigured shielded|HSPShld-Hotspare shielded
CFShld-Configured shielded|Cpybck-CopyBack|CBSHld-Copyback Shielded
```

- Identify the existing block devices (sda and sdb) on the host used by the OS and NetWitness. Use the command 'lsblk' to list out the block devices.

```
lsblk
```

```
[root@116S6Core1 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 1.1T 0 disk
├─sda1 8:1 0 1G 0 part /boot
├─sda2 8:2 0 1.1T 0 part
│ ├─netwitness_vg00-root 253:0 0 30G 0 lvm /
│ ├─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
│ ├─netwitness_vg00-nwhome 253:2 0 3.2T 0 lvm /var/netwitness
│ ├─netwitness_vg00-varlog 253:3 0 10G 0 lvm /var/log
│ └─netwitness_vg00-usrhome 253:4 0 10G 0 lvm /home
sdb 8:16 0 2.2T 0 disk
├─sdb1 8:17 0 2.2T 0 part
└─netwitness_vg00-nwhome 253:2 0 3.2T 0 lvm /var/netwitness
```

#### Configure the virtual drive(s) on the installed Meta Disk kit (slots 4,5, and 6):

- Create the Virtual Drive/Drive Group (DG) on the internal controller using the disks in slot 4 through 6 using the below command.

```
/opt/MegaRAID/perccli/perccli64 /c0 add vd type=raid5 drives=64:4-6
strip=128
```

```
[root@117Decoder ~]# /opt/MegaRAID/percccli/percccli64 /c0 add vd type=raid5 drives=64:4-6 strip=128
Controller = 0
Status = Success
Description = Add VD Succeeded
```

- 6. The new virtual drive shows up as '2/2' under DG/VD column. See the entry highlighted in yellow.  
/opt/MegaRAID/percccli/percccli64 /c0 /vall show

```
[root@117Decoder percccli]# /opt/MegaRAID/percccli/percccli64 /c0 /vall show
Controller = 0
Status = Success
Description = None

Virtual Drives :
=====

DG/VD TYPE State Access Consist Cache Cac sCC Size Name

0/0 RAID1 Optl RW Yes RWBD - OFF 1.090 TB
1/1 RAID1 Optl RW Yes RWBD - OFF 2.182 TB
2/2 RAID5 Optl RW Yes RWBD - OFF 4.365 TB

Cac=CacheCade|Rec=Recovery|OfLn=OffLine|Pdgd=Partially Degraded|Dgrd=Degraded
Optl=Optimal|R0=Read Only|RW=Read Write|HD=Hidden|TRANS=TransportReady|B=Blocked|
Consist=Consistent|R=Read Ahead Always|NR=No Read Ahead|WB=WriteBack|
FWB=Force WriteBack|WT=WriteThrough|C=Cached IO|D=Direct IO|sCC=Scheduled
Check Consistency
```

- 7. Identify the new block device on the host. The block device name is identified under the NAME column. In this example the new block device is 'sdc'. This block device name is required when configuring storage. Use 'lsblk' to list the block devices. See entry highlighted in yellow.

lsblk

```
[root@117Decoder percccli]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 1.1T 0 disk
├─sda1 8:1 0 1G 0 part /boot
├─sda2 8:2 0 256M 0 part /boot/efi
├─sda3 8:3 0 1.1T 0 part
│ └─netwitness_vg00-root 253:0 0 30G 0 lvm /
│ └─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
│ └─netwitness_vg00-nwhome 253:2 0 3.2T 0 lvm /var/netwitness
│ └─netwitness_vg00-varlog 253:3 0 10G 0 lvm /var/log
│ └─netwitness_vg00-usrhome 253:4 0 10G 0 lvm /home
sdb 8:16 0 2.2T 0 disk
├─sdb1 8:17 0 2.2T 0 part
└─netwitness_vg00-nwhome 253:2 0 3.2T 0 lvm /var/netwitness
sdc 8:32 0 4.4T 0 disk
```

- 8. Now, you must Configure the above block device as decodersmall for Network Decoder. Use the REST API (Ex: <https://<HostIP>:50106>, HostIP is the ip address of the decoder host) to retrieve the existing raid list. Note the block device name corresponding to the Meta Disk kit. In this case it is 'sdc'.

**Note:** If a second Meta Disk kit is being configured, the block device would be 'sdd'.

Properties for /appliance  
raidList  Parameters:

Message Help

```
raidList: list drive shelves attached to this appliance
security.roles: appliance.manage
```

/appliance?msg=raidList&force-content-type=text/plain

Output (or command manual help)

```
Controller 0 at PCI Address 18:00.0, Enclosure 64
Vendor: DP
Model: BP14G+EXP
In Use: true
Drives: 1.09 TB HDD x 2
 2.182 TB HDD x 8
Devices: sda /dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:0:0
 sdb /dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:1:0
 sdc /dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:2:0
```

9. Make partitions on the block device ('sdc') for decoder service with the following parameters:

```
name=sdc service=decoder volume=decoderssmall commit=1
```

For logdecoder use: name=sdc service=logdecoder volume=logdecoderssmall  
commit=1

**Note:** If a second Meta Disk kit is being configured, the volume for decoder would be 'decoderssmall0'. For logdecoder, it is 'logdecoderssmall0'.

Properties for /appliance  
partNew  Parameters: name=sdc service=decoder volume=decoderssmall commit=1

Message Help

```
partNew: make partitions on a block device
security.roles: appliance.manage
parameters:
 name - <string, {enum-one:The value must be one of the following: sdc}> block device name
 service - <string, {enum-one:The value must be one of the following: archiver|concentrator|decoder|logdecoder}> service that will use storage
 volume - <string, optional, {enum-one:The value must be one of the following: concentrator|index|decoderssmall|decoder|packet|hybrid-decoder-meta|logdecoderssmall|logdecoder|logpacket|hybrid-logdecoder-meta|archiver|hybrid-concentrator|endpoint-log-hybrid|log-indexed-decoder|logindex}> volume to create
```

/appliance?msg=partNew&force-content-type=text/plain&name=sdc&service=decoder&volume=decoderssmall&commit=1

Output (or command manual help)

```
/sbin/parted -s /dev/sdc mklabel gpt
/sbin/parted -s -a optimal /dev/sdc mkpart LWN 0% 100%
/sbin/pvcreate -f /dev/sdc1
Physical volume "/dev/sdc1" successfully created.
/sbin/vgcreate -f decoderssmall /dev/sdc1
Volume group "decoderssmall" successfully created
/sbin/lvcreate -y -n decoroot -l 10% decoderssmall
Wiping LWN2_member signature on /dev/decoderssmall/decoroot.
Logical volume "decoroot" created.
/sbin/mkfs.xfs /dev/decoderssmall/decoroot
meta-data=/dev/decoderssmall/decoroot isize=512 agcount=16, agsize=163808 blks
 = sectsz=512 attr=2, projid32bit=1
 = crc=1 finobt=0, sparse=0
 = bsize=4096 blocks=2620928, imaxpct=25
 = sunit=32 swidth=64 blks
naming =version 2 bsize=4096 ascii-ci=0 ftype=1
log =internal log bsize=4096 blocks=2560, version=2
 = sectsz=512 sunit=32 blks, lazy-count=1
realtime =none extsz=4096 blocks=0, rtextents=0
/sbin/mkdir -p /var/netwitness/decoder
/bin/mount /var/netwitness/decoder
/sbin/lvcreate -v -n index -L 30% decoderssmall
```

10. Allocate the decoderssmall to Decoder service using 'srvAlloc'.

```
service=decoder volume=decoderssmall commit=1
```

**Note:** For logdecoder: service=logdecoder volume=logdecoderssmall commit=1

Properties for /appliance

srvAlloc Parameters:

Message Help

srvAlloc: apply volume group storage to a service on this appliance  
 security.roles: appliance.manage  
 parameters:  
 service - <string, {enum-one:The value must be one of the following: archiver|concentrator|decoder|logdecoder}> service that will use storage  
 volume - <string, {enum-one:The value must be one of the following: decoderssmall|netwitness\_vg00}> volume group name  
 commit - <bool, optional> commit changes

/appliance?msg=srvAlloc&force-content-type=text/plain&service=decoder&volume=decoderssmall&commit=1

Output (or command manual help)

Set /database/config/meta.dir to /var/netwitness/decoder/metadb==3.55 TB  
 Set /database/config/session.dir to /var/netwitness/decoder/sessiondb==569.72 GB  
 Set /index/config/index.dir to /var/netwitness/decoder/index==28.48 GB

```
[root@117Decoder ~]# df -hP
Filesystem Size Used Avail Use% Mounted on
devtmpfs 63G 0 63G 0% /dev
tmpfs 63G 320K 63G 1% /dev/shm
tmpfs 63G 27M 63G 1% /run
tmpfs 63G 0 63G 0% /sys/fs/cgroup
/ Files apper/netwitness_vg00-root 30G 3.4G 27G 12% /
/udev/_da1 1019M 88M 932M 9% /boot
/dev/sda2 256M 8.0K 256M 1% /boot/efl
/dev/mapper/netwitness_vg00-nwhome 3.3T 655M 3.3T 1% /var/netwitness
/dev/mapper/netwitness_vg00-varlog 10G 53M 10G 1% /var/log
/dev/mapper/netwitness_vg00-usrhome 10G 33M 10G 1% /home
/dev/mapper/decoderssmall-decoroot 10G 35M 10G 1% /var/netwitness/decoder
/dev/mapper/decoderssmall-index 30G 33M 30G 1% /var/netwitness/decoder/index
/dev/mapper/decoderssmall-metadb 3.8T 34M 3.8T 1% /var/netwitness/decoder/metadb
/dev/mapper/decoderssmall-sessiondb 600G 34M 600G 1% /var/netwitness/decoder/sessiondb
tmpfs 13G 0 13G 0% /run/user/0
```

## Maximize PowerVault Storage Capacity (One Kit – 3 SED Drives Configured as RAID 5)

Configure one Meta Disk kit as decoderssmall volume for Network Decoder maximizing the PowerVault Storage.

### Note:

1. Best practice – when 1-4 PowerVaults are configured, a single Meta Disk kit is recommended and when 5-8 PowerVaults are configured two (2) Meta Disk kits are recommended (optional).
2. The configuration for Log Decoder is the similar as that of a Network Decoder. Substitute the service and volume names that correspond to Log Decoder.
3. A single Meta Disk kit is configured as RAID5 (3 drives) and two Meta Disk kits (6 drives) are configured as RAID6.
4. When configuring two Meta Disk kits or adding the second Meta Disk kit, use appropriate disk slot numbers when creating the virtual drive (step 5 below).

On the Series 6 (Dell R640) appliance, the Meta Drive Pack disks are installed in slots 4, 5 and 6. The virtual drive configuration requires identifying the controller ID and Enclosure ID (EID). On Series 6 appliance, the controller ID and Enclosure IDs are 0 and 64. However, the nwraidtool.py script that is installed on every server can help to confirm these ID numbers, or the PercCLI commands as shown in the below steps.

### Identify controller/enclosures:



1. Identify the controller ID (Ctl) for the internal controller (PERC H740P Mini/ PERC H750). In the below figure the controller ID is 0 and highlighted in yellow. The drive count is displayed under PDs.

```
/opt/MegaRAID/perccli/perccli64 show
```

```
[root@117Decoder ~]# /opt/MegaRAID/perccli/perccli64 show
Status Code = 0
Status = Success
Description = None

Number of Controllers = 2
Host Name = 117Decoder
Operating System = Linux 3.10.0-1160.36.2.el7.x86_64
StoreLib IT Version = 07.0400.0200.0400
StoreLib IR3 Version = 15.50-0

System Overview :
=====

Ctl Model Ports PDs DGs DNOpt VDs VNOpt BBU sPR DS EHS ASOs Hlth

0 PERCH740PMini 8 1 - 0 1 0 Opt On - N 0 Opt
1 PERCH840Adapter 8 24 1 0 1 0 Opt On - N 0 Opt

Ctl=Controller Index|DGs=Drive groups|VDs=Virtual drives|Fld=Failed
PDs=Physical drives|DNOpt=DG NotOptimal|VNOpt=VD NotOptimal|Opt=Optimal
Msg=Missing|Dgd=Degraded|NdAtn=Need Attention|Unkwn=Unknown
sPR=Scheduled Patrol Read|DS=DimmerSwitch|EHS=Emergency Hot Spare
Y=Yes|N=No|ASOs=Advanced Software Options|BBU=Battery backup unit
Hlth=Health|Safe=Safe-mode boot
```

2. Identify the Enclosure ID (EID) for controller '0'. In this case the EID is '64' (highlighted in yellow).

```
/opt/MegaRAID/perccli/perccli64 /c0 /eall show
```

```
[root@117Decoder ~]# /opt/MegaRAID/perccli/perccli64 /c0 /eall show
Controller = 0
Status = Success
Description = None

Properties :
=====

EID State Slots PD PS Fans TSs Alms SIM Port# ProdID VendorSpecific

64 OK 10 1 0 0 0 0 1 00 & 00 x8 BP14G+EXP +

EID-Enclosure Device ID |PD-Physical drive count |PS-Power Supply count|
TSs-Temperature sensor count |Alms-Alarm count |SIM-SIM Count
```

3. Identify the new Meta Disk kit disk slot numbers (Slots 4 through 9) on the controller PERC H740P Mini. These drives do not belong to any Drive Group (DG). The DG column for these drives display a dash '-', the State column shows 'UGood', and the SED value as 'Y'. See the entries highlighted in yellow.

```
[root@116S6Core1 ~]# /opt/MegaRAID/perccli/perccli64 /c0 /eall /sall show
Controller = 0
Status = Success
Description = Show Drive Information Succeeded.

Drive Information :
=====
```

| EID:SlT | DID | State | DG | Size     | Intf | Med | SED | PI | SeSz | Model        | Sp |
|---------|-----|-------|----|----------|------|-----|-----|----|------|--------------|----|
| 64:0    | 0   | Onln  | 0  | 1.090 TB | SAS  | HDD | Y   | N  | 512B | ST1200MM0069 | U  |
| 64:1    | 1   | Onln  | 0  | 1.090 TB | SAS  | HDD | Y   | N  | 512B | ST1200MM0069 | U  |
| 64:2    | 2   | Onln  | 1  | 2.182 TB | SAS  | HDD | Y   | N  | 512B | ST2400MM0149 | U  |
| 64:3    | 3   | Onln  | 1  | 2.182 TB | SAS  | HDD | Y   | N  | 512B | ST2400MM0149 | U  |
| 64:4    | 4   | UGood | -  | 2.182 TB | SAS  | HDD | Y   | N  | 512B | ST2400MM0149 | U  |
| 64:5    | 5   | UGood | -  | 2.182 TB | SAS  | HDD | Y   | N  | 512B | ST2400MM0149 | U  |
| 64:6    | 6   | UGood | -  | 2.182 TB | SAS  | HDD | Y   | N  | 512B | ST2400MM0149 | U  |
| 64:7    | 7   | UGood | -  | 2.182 TB | SAS  | HDD | Y   | N  | 512B | ST2400MM0149 | U  |
| 64:8    | 8   | UGood | -  | 2.182 TB | SAS  | HDD | Y   | N  | 512B | ST2400MM0149 | U  |
| 64:9    | 9   | UGood | -  | 2.182 TB | SAS  | HDD | Y   | N  | 512B | ST2400MM0149 | U  |

```

EID-Enclosure Device ID|SlT-Slot No.|DID-Device ID|DG-DriveGroup
DHS-Dedicated Hot Spare|UGood-Unconfigured Good|GHS-Global Hotspare
UBad-Unconfigured Bad|Onln-Online|Offln-Offline|Intf-Interface
Med-Media Type|SED-Self Encryptive Drive|PI-Protection Info
SeSz-Sector Size|Sp-Spun|U-Up|D-Down/PowerSave|T-Transition|F-Foreign
UGUnsp-Unsupported|UGShld-UnConfigured shielded|HSPShld-Hotspare shielded
CFShld-Configured shielded|Cpybck-CopyBack|CBSHld-Copyback Shielded
```

- Identify the existing block devices (sda and sdb) on the host used by the OS and NetWitness. Use the command 'lsblk' to list out the block devices.

```
lsblk
```

```
[root@116S6Core1 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 1.1T 0 disk
├─sda1 8:1 0 1G 0 part /boot
├─sda2 8:2 0 1.1T 0 part
│ ├─netwitness_vg00-root 253:0 0 30G 0 lvm /
│ ├─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
│ ├─netwitness_vg00-nwhome 253:2 0 3.2T 0 lvm /var/netwitness
│ └─netwitness_vg00-varlog 253:3 0 10G 0 lvm /var/log
└─netwitness_vg00-usrhome 253:4 0 10G 0 lvm /home
sdb 8:16 0 2.2T 0 disk
├─sdb1 8:17 0 2.2T 0 part
└─netwitness_vg00-nwhome 253:2 0 3.2T 0 lvm /var/netwitness
```

#### Configure virtual drive(s) on installed meta pack disks (slots 4,5 and 6):

- Create the Virtual Drive/Drive Group (DG) on the internal controller using the disks in slot 4 through 6 using the below command.

```
/opt/MegaRAID/perccli/perccli64 /c0 add vd type=raid5 drives=64:4-6
strip=128
```

```
[root@117Decoder ~]# /opt/MegaRAID/perccli/perccli64 /c0 add vd type=raid5 drives=64:4-6 strip=128
Controller = 0
Status = Success
Description = Add VD Succeeded
```

- The new virtual drive shows up as '2/2' (highlighted in yellow) under DG/VD column.

```
/opt/MegaRAID/perccli/perccli64 /c0 /vall show
```

```
[root@117Decoder ~]# /opt/MegaRAID/perccli/perccli64 /c0 /vall show
Controller = 0
Status = Success
Description = None

Virtual Drives :
=====

DG/VD TYPE State Access Consist Cache Cac sCC Size Name

0/0 RAID1 Optl RW Yes RWBD - OFF 1.090 TB
1/1 RAID1 Optl RW Yes RWBD - OFF 2.182 TB
2/2 RAID5 Optl RW Yes RWBD - OFF 4.365 TB

Cac=CacheCade|Rec=Recovery|OfLn=OffLine|Pdgd=Partially Degraded|Dgrd=Degraded
Optl=Optimal|RO=Read Only|RW=Read Write|HD=Hidden|TRANS=TransportReady|B=Blocked|
Consist=Consistent|R=Read Ahead Always|NR=No Read Ahead|WB=WriteBack|
FWB=Force WriteBack|WT=WriteThrough|C=Cached IO|D=Direct IO|sCC=Scheduled
Check Consistency
```

- Identify the new block device on the host. The block device name is identified under the NAME column. In this example the new block device is 'sdc'. This block device name is required when configuring storage. Use 'lsblk' to list the block devices.

```
lsblk
```

```
[root@117Decoder ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 1.1T 0 disk
├─sda1 8:1 0 1G 0 part /boot
├─sda2 8:2 0 256M 0 part /boot/efi
├─sda3 8:3 0 1.1T 0 part
├─netwitness_vg00-root 253:0 0 30G 0 lvm /
├─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
├─netwitness_vg00-nwhome 253:2 0 3.2T 0 lvm /var/netwitness
├─netwitness_vg00-varlog 253:3 0 10G 0 lvm /var/log
└─netwitness_vg00-usrhome 253:4 0 10G 0 lvm /home
sdb 8:16 0 2.2T 0 disk
├─sdb1 8:17 0 2.2T 0 part
└─netwitness_vg00-nwhome 253:2 0 3.2T 0 lvm /var/netwitness
sdc 8:32 0 4.4T 0 disk
```

- Configure the above block device as decodersmall volume for Decoder (volume name for Log Decoder is logdecoderssmall). Use REST API (Ex: <https://<HostIP>:50106>, HostIP is the ip address of the decoder host) to retrieve the existing raid list. Note the block device name corresponding to the Meta Disk kit. In this case, it is 'sdc' (highlighted in yellow)

**Note:** If a second Meta Disk kit is being configured, the block device would be 'sdd'.



Properties for /appliance  
raidList Parameters:  Send

Message Help

raidList: list drive shelves attached to this appliance  
security.roles: appliance.manage

/appliance?msg=raidList&force-content-type=text/plain

Output (or command manual help)

```
Controller 0 at PCI Address 18:00.0, Enclosure 64
Vendor: DP
Model: BP14G+EXP
In Use: true
Drives: 1.09 TB HDD x 2
 2.182 TB HDD x 8
Devices: sda /dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:0:0
 sdb /dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:1:0
 sdc /dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:2:0
```

9. Make partitions on the block device ('sdc') with the following parameters:

```
name=sdc service=decoder volume=decoderssmall commit=1
```

For logdecoder service, use: name=sdc service=logdecoder  
volume=logdecoderssmall commit=1

**Note:** If a second Meta Disk kit is being configured, the volume for decoder service is 'decoderssmall' and 'logdecoderssmall' for logdecoder service.

Properties for /appliance  
partNew Parameters: name=sdc service=decoder volume=decoderssmall commit=1 Send

Message Help

partNew: make partitions on a block device  
security.roles: appliance.manage  
parameters:  
name - <string, {enum=one:The value must be one of the following: sdc}> block device name  
service - <string, {enum=one:The value must be one of the following: archiver|concentrator|decoder|logdecoder}> service that will use storage  
volume - <string, optional, {enum=one:The value must be one of the following: concentrator|index|decoderssmall|decoder|packet|hybrid-decoder-meta|logdecoderssmall|logdecoder|logpacket|hybrid-logdecoder-meta|archiver|hybrid-concentrator|endpoint-log-hybrid|log-indexed-decoder|logindex}> volume to create

/appliance?msg=partNew&force-content-type=text/plain&name=sdc&service=decoder&volume=decoderssmall&commit=1

Output (or command manual help)

```
/sbin/parted -s /dev/sdc mklabel gpt
/sbin/parted -s -a optimal /dev/sdc mkpart LVM 0% 100%
/sbin/pvcreate -f /dev/sdc1
Physical volume "/dev/sdc1" successfully created.
/sbin/vgcreate -f decoderssmall /dev/sdc1
Volume group "decoderssmall" successfully created
/sbin/lvcreate -y -n decoroot -L 10G decoderssmall
Wiping LVM2_member signature on /dev/decoderssmall/decoroot.
Logical volume "decoroot" created.
/sbin/mkfs.xfs /dev/decoderssmall/decoroot
meta-data=/dev/decoderssmall/decoroot isize=512 agcount=16, agsize=163808 blks
 = sectsz=512 attr=2, projid32bit=1
 = crc=1 finobt=0, sparse=0
data = bsize=4096 blocks=2689928, imaxpct=25
 = sunit=32 swidth=64 blks
naming =version 2 bsize=4096 ascii-ci=0 ftype=1
log =internal log bsize=4096 blocks=2560, version=2
 = sectsz=512 sunit=32 blks, lazy-count=1
realtime =none extsz=4096 blocks=0, rtextents=0
/bin/mkdir -p /var/netwitness/decoder
/bin/mount /var/netwitness/decoder
/sbin/lvcreate -y -n index -L 30G decoderssmall
```

10. Allocate the decoderssmall to Decoder service

```
service=decoder volume=decoderssmall commit=1
For logdecoder use:
service=logdecoder volume=logdecoderssmall commit=1
```

**Note:** If a second Meta Pack is being configured, the volume for decoder is 'decoderssmall0' and 'logdecoderssmall0' for logdecoder service.

Properties for /appliance  
 srvAlloc Parameters: service=decoder volume=decodersmall commit=1

Message Help

srvAlloc: apply volume group storage to a service on this appliance  
 security.roles: appliance.manage  
 parameters:  
 service - <string, {enum-one:The value must be one of the following: archiver|concentrator|decoder|logdecoder}> service that will use storage  
 volume - <string, {enum-one:The value must be one of the following: decodersmall|netwitness\_vg00}> volume group name  
 commit - <bool, optional> commit changes

/appliance?msg=srvAlloc&force-content-type=text/plain&service=decoder&volume=decodersmall&commit=1

Output (or command manual help)

```
Set /database/config/meta.dir to /var/netwitness/decoder/metadb==3.55 TB
Set /database/config/session.dir to /var/netwitness/decoder/sessiondb==569.72 GB
Set /index/config/index.dir to /var/netwitness/decoder/index==28.48 GB
```

```
[root@117Decoder ~]# df -hP
Filesystem Size Used Avail Use% Mounted on
devtmpfs 63G 0 63G 0% /dev
tmpfs 63G 320K 63G 1% /dev/shm
tmpfs 63G 267M 63G 1% /run
tmpfs 63G 0 63G 0% /sys/fs/cgroup
/dev/mapper/netwitness_vg00-root 30G 5.3G 25G 18% /
/dev/sda1 1019M 88M 932M 9% /boot
/dev/sda2 256M 8.0K 256M 1% /boot/efi
/dev/mapper/netwitness_vg00-nw home 3.3T 740M 3.3T 1% /var/netwitness
/dev/mapper/netwitness_vg00-var log 10G 219M 9.8G 3% /var/log
/dev/mapper/netwitness_vg00-user home 10G 33M 10G 1% /home
/dev/mapper/decodersmall-decoroot 10G 128M 9.9G 2% /var/netwitness/decoder
/dev/mapper/decodersmall-index 30G 33M 30G 1% /var/netwitness/decoder/index
/dev/mapper/decodersmall-metadb 3.8T 431G 3.4T 12% /var/netwitness/decoder/metadb
/dev/mapper/decodersmall-sessiondb 600G 34M 600G 1% /var/netwitness/decoder/sessiondb
```

### Configure external storage (PowerVault) as packetDB for Decoder:

11. Use REST API (Ex: <https://<HostIP>:50106>, HostIP is the ip address of the decoder host) to configure raid on the attached PowerVault. Identify the controller and enclosure information using raidList. In this example, two PowerVaults are connected to the Decoder appliance (controller 1 with enclosure 246 and 249). REST API is used in the next steps to configure the PowerVault (Enclosure 246) connected to Controller 1.

Properties for /appliance  
 raidList Parameters:

Message Help

raidList: list drive shelves attached to this appliance  
 security.roles: appliance.manage

/appliance?msg=raidList&force-content-type=text/plain

Output (or command manual help)

```
Vendor: DP
Model: BP14G+EXP
In Use: true
Drives: 1.09 TB HDD x 2
2.182 TB HDD x 8
Devices: sda /dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:0:0
sdb /dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:1:0
sdc /dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:2:0

Controller 1 at PCI Address 3b:00.0, Enclosure 246
Vendor: DELL
Model: MD1400
In Use: false
Drives: 10.692 TB HDD x 12
Devices:

Controller 1 at PCI Address 3b:00.0, Enclosure 249
Vendor: DELL
Model: MD1400
In Use: false
Drives: 10.692 TB HDD x 12
Devices:
```

12. Create the raid on the attached PowerVault. Identify the controller and enclosure from 'raidList'. Use appropriate value for preferSecure to enable encryption. To turn on encryption on SED capable drives, use preferSecure=1 after setting the security key on the controller. Encryption can also be turned on after configuring storage using the steps listed in Appendix B. Encrypt a Series 6E Core or Hybrid Host of the storage guide.

```
controller=1 enclosure=246 scheme=packet-expansion preferSecure=0
commit=1
```

Properties for /appliance  
**raidNew** Parameters:

Message Help  
 enclosure - <string, optional> (enum-one: The value must be one of the following: 04|246|249) Enclosure number of the shelf to clear. Required if the controller has more than one enclosure attached.  
 scheme - <string, (enum-one: The value must be one of the following: decoder|logdecoder|concentrator|archiver|network-hybrid|log-hybrid|log-indexed-decoder|endpoint-log-hybrid|packet-expansion|decoder-hotspare|logdecoder-hotspare)> Type of RAID volumes to allocate  
 preferSecure - <bool, optional> (bool: The value must be one of the following acceptable boolean values: 0,1,yes,no,true,false,on,off) Prefer creation of a secure array given compatible physical drives and a controller with a security key set  
 commit - <bool, optional> commit changes

/appliance?msg=raidNew&force-content-type=text/plain&controller=1&enclosure=246&scheme=packet-expansion&preferSecure=0&commit=1

Output (or command manual help)  

```
/opt/MegaRAID/perccli/perccli64 /c1 add vd r6 drives=246:0,246:1,246:2,246:3,246:4,246:5,246:6,246:7,246:8,246:9,246:10,246:11 ra Strip=128
Controller = 1
Status = Success
Description = Add VD Succeeded
```

13. Identify the block device created in the earlier step. Use 'raidList' to retrieve all the block devices. In this case, it is 'sdd'.

Properties for /appliance  
**raidList** Parameters:

Message Help  
 raidList: list drive shelves attached to this appliance  
 security.roles: appliance.manage

/appliance?msg=raidList&force-content-type=text/plain

Output (or command manual help)  

```
Controller 0 at PCI Address 18:00.0, Enclosure 64
Vendor: DP
Model: BP14G+EXP
In Use: true
Drives: 1.09 TB HDD x 2
2.182 TB HDD x 8
Devices: sda /dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:0:0
sdb /dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:1:0
sdc /dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:2:0

Controller 1 at PCI Address 3b:00.0, Enclosure 246
Vendor: DELL
Model: MD1400
In Use: true
Drives: 10.692 TB HDD x 12
Devices: sdd /dev/disk/by-path/pci-0000:3b:00.0-scsi-0:2:0:0

Controller 1 at PCI Address 3b:00.0, Enclosure 249
Vendor: DELL
Model: MD1400
In Use: false
Drives: 10.692 TB HDD x 12
```

14. Make partitions on the block device ('sdd') using 'partNew'.

```
name=sdd service=decoder volume=decoder commit=1
```

Properties for /appliance

partNew Parameters:

Message Help

```
partNew: make partitions on a block device
security.roles: appliance.manage
parameters:
 name - <string, {enum-one:The value must be one of the following: sdd}> block device name
 service - <string, {enum-one:The value must be one of the following: archiver|concentrator|decoder|logdecoder}> service that will use sto
 volume - <string, optional, {enum-one:The value must be one of the following: concentrator|index|decodersmall|decoder|packet|hybrid-decod
logdecoder-meta|archiver|hybrid-concentrator|endpoint-log-hybrid|log-indexed-decoder|logindex}> volume to create
 commit - <bool, optional> commit changes
```

/appliance?msg=partNew&force-content-type=text/plain&name=sdd&service=decoder&volume=decoder&commit=1

Output (or command manual help)

```
/sbin/parted -s /dev/sdd mklabel gpt
/sbin/parted -s -a optimal /dev/sdd mkpart LVM 0% 100%
/sbin/pvcreate -f /dev/sdd1
Physical volume "/dev/sdd1" successfully created.
/sbin/vgcreate -f decoder /dev/sdd1
Volume group "decoder" successfully created
/sbin/lvcreate -y -n packetdb -l 100%FREE decoder
Logical volume "packetdb" created.
/sbin/mkfs.xfs /dev/decoder/packetdb
meta-data=/dev/decoder/packetdb isize=512 agcount=107, agsize=268435424 blks
 = sectsz=512 attr=2, projid32bit=1
 = crc=1 finobt=0, sparse=0
data = bsize=4096 blocks=28700834816, imaxpct=1
 = sunit=32 swidth=256 blks
naming =version 2 bsize=4096 ascii-ci=0 ftype=1
log =internal log bsize=4096 blocks=521728, version=2
 = sectsz=512 sunit=32 blks, lazy-count=1
realtime =none extsz=4096 blocks=0, rtextents=0
/bin/mkdir -p /var/netwitness/decoder/packetdb
/bin/mount /var/netwitness/decoder/packetdb
```

```
[root@117Decoder ~]# df -hP
Filesystem Size Used Avail Use% Mounted on
devtmpfs 63G 0 63G 0% /dev
tmpfs 63G 160K 63G 1% /dev/shm
tmpfs 63G 27M 63G 1% /run
tmpfs 63G 0 63G 0% /sys/fs/cgroup
/dev/mapper/netwitness_vg00-root 30G 3.4G 27G 12% /
/dev/sda1 1019M 88M 932M 9% /boot
/dev/sda2 256M 8.0K 256M 1% /boot/efi
/dev/mapper/netwitness_vg00-nwhome 3.3T 656M 3.3T 1% /var/netwitness
/dev/mapper/netwitness_vg00-varlog 10G 53M 10G 1% /var/log
/dev/mapper/netwitness_vg00-usrhome 10G 33M 10G 1% /home
/dev/mapper/decodersmall-decoroot 10G 36M 10G 1% /var/netwitness/decoder
/dev/mapper/decodersmall-index 30G 33M 30G 1% /var/netwitness/decoder/index
/dev/mapper/decodersmall-metadb 3.8T 34M 3.8T 1% /var/netwitness/decoder/metadb
/dev/mapper/decodersmall-sessiondb 600G 34M 600G 1% /var/netwitness/decoder/sessiondb
tmpfs 13G 0 13G 0% /run/user/0
/dev/mapper/decoder-packetdb 107T 38M 107T 1% /var/netwitness/decoder/packetdb
```

- Allocate the decoder volume to Decoder service using 'srvAlloc'.

```
service=decoder volume=decoder commit=1
```

Properties for /appliance

srvAlloc ▾ Parameters:

Message Help

```
srvAlloc: apply volume group storage to a service on this appliance
security.roles: appliance.manage
parameters:
 service - <string, {enum-one:The value must be one of the following: archiver|concentrator|decoder|logdecoder}> service that will
 volume - <string, {enum-one:The value must be one of the following: decoder|decodersmall|netwitness_vg00}> volume group name
 commit - <bool, optional> commit changes
```

Output (or command manual help)

```
Set /database/config/packet.dir to /var/netwitness/decoder/packetdb==101.57 TB
```

**Note:** Additional PowerVaults shall be configured using REST API as outlined in the Storage Guide.

## Appendix G. Sample Storage Configuration for Concentrator Index with One Meta Disk Kit

This appendix illustrates the examples of how to configure concentrator index with one Meta Disk Kit.

### Note:

- Concentrator index meta kit supports a minimum of two to a maximum of 6 drives (SSD).
- A single Meta Disk pack is configured as RAID5 (3 drives) or two Meta Disk packs (6 drives) are configured as RAID6. Two drive configuration is also supported. The two drives are added in slots 4,5 and configured as RAID1.
- When configuring two Meta Disk Packs or adding the second Meta Disk pack, use appropriate disk slot numbers when creating the virtual drive (step 5 below).

On the Series 6 (Dell R640) appliance, the Meta Drive Pack disks are installed in slots 4, 5 and 6. The virtual drive configuration requires identifying the controller ID and Enclosure ID (EID). On Series 6 appliance, the controller ID and Enclosure IDs are 0 and 64. However, the `nwr RAIDtool.py` script that is installed on every server can help to confirm these ID numbers, or the PercCLI commands as shown in the below steps.

### Identify Controller/Enclosures

1. Identify the controller ID (Ctl) for the internal controller (PERC H740P Mini/ PERC H750). In the below figure the controller ID is 0 and highlighted in yellow. The drive count is displayed under PDs.

```
/opt/MegaRAID/perccli/perccli64 show
```

```
[root@S6Core2 ~]# /opt/MegaRAID/perccli/perccli64 show
CLI Version = 007.1623.0000.0000 May 17, 2021
Operating system = Linux 3.10.0-1160.71.1.el7.x86_64
Status Code = 0
Status = Success
Description = None

Number of Controllers = 2
Host Name = S6Core2
Operating System = Linux 3.10.0-1160.71.1.el7.x86_64

System Overview :
=====
```

| Ctl | Model           | Ports | PDs | DGs | DNOpt | VDs | VNOpt | BBU | sPR | DS | EHS | ASOs | Hlth |
|-----|-----------------|-------|-----|-----|-------|-----|-------|-----|-----|----|-----|------|------|
| 0   | PERCH740PMini   | 8     | 7   | 2   | 0     | 2   | 0     | Opt | On  | -  | N   | 0    | Opt  |
| 1   | PERCH840Adapter | 8     | 24  | 2   | 0     | 2   | 0     | Opt | On  | -  | N   | 0    | Opt  |

```

Ctl=Controller Index|DGs=Drive groups|VDs=Virtual drives|Fld=Failed
PDs=Physical drives|DNOpt=Array NotOptimal|VNOpt=VD NotOptimal|Opt=Optimal
Msg=Missing|Dgd=Degraded|NdAtn=Need Attention|Unkwn=Unknown
sPR=Scheduled Patrol Read|DS=DimmerSwitch|EHS=Emergency Spare Drive
Y=Yes|N=No|ASOs=Advanced Software Options|BBU=Battery backup unit/CV
Hlth=Health|Safe=Safe-mode boot|CertProv=Certificate Provision mode
Chrg=Charging | MsgnCbl=Cable Failure
```



- Identify the Enclosure ID (EID) for controller '0'. In this case the EID is '64' and highlighted in yellow.

```
/opt/MegaRAID/perccli/perccli64 /c0 /eall show
```

```
[root@S6Core2 ~]# /opt/MegaRAID/perccli/perccli64 /c0 /eall show
CLI Version = 007.1623.0000.0000 May 17, 2021
Operating system = Linux 3.10.0-1160.71.1.el7.x86_64
Controller = 0
Status = Success
Description = None

Properties :
=====

EID State Slots PD PS Fans TSs Alms SIM Port# ProdID VendorSpecific

64 OK 10 7 0 0 0 0 1 - BP14G+EXP +

EID=Enclosure Device ID | PD=Physical drive count | PS=Power Supply count
TSs=Temperature sensor count | Alms=Alarm count | SIM=SIM Count | ProdID=Product ID
```

- Identify the new Meta Disk kit disk slot numbers (Slots 4 through 9) on the controller PERC H740P Mini. These drives do not belong to any Drive Group (DG). The DG column for these drives display a dash '-', the State column shows 'UGood', and the SED value as 'Y'. See the entries highlighted in yellow.

```
/opt/MegaRAID/perccli/perccli64 /c0 /eall /sall show
```

```
[root@S6Core2 ~]# /opt/MegaRAID/perccli/perccli64 /c0 /eall /sall show
CLI Version = 007.1623.0000.0000 May 17, 2021
Operating system = Linux 3.10.0-1160.71.1.el7.x86_64
Controller = 0
Status = Success
Description = Show Drive Information Succeeded.

Drive Information :
=====

EID:Slt DID State DG Size Intf Med SED PI SeSz Model Sp Type

64:0 0 Onln 0 931.000 GB SAS HDD N N 512B ST1000NX0473 U -
64:1 1 Onln 0 931.000 GB SAS HDD N N 512B ST1000NX0473 U -
64:2 2 Onln 1 1.818 TB SAS HDD N N 512B ST2000NX0463 U -
64:3 3 Onln 1 1.818 TB SAS HDD N N 512B ST2000NX0463 U -
64:4 4 UGood - 3.492 TB SAS SSD Y N 512B KPM5WVUG3T84 U -
64:5 5 UGood - 3.492 TB SAS SSD Y N 512B KPM5WVUG3T84 U -
64:6 6 UGood - 3.492 TB SAS SSD Y N 512B KPM6WVUG3T84 U -

EID=Enclosure Device ID|Slt=Slot No|DID=Device ID|DG=DriveGroup
DHS=Dedicated Hot Spare|UGood=Unconfigured Good|GHS=Global Hotspare
UBad=Unconfigured Bad|Sntze=Sanitize|Onln=Online|Offln=Offline|Intf=Interface
Med=Media Type|SED=Self Encryptive Drive|PI=Protection Info
SeSz=Sector Size|Sp=Spun|U=Up|D=Down|T=Transition|F=Foreign
UGUnsp=UGood Unsupported|UGShld=UGood shielded|HSPShld=Hotspare shielded
CFShld=Configured shielded|Cpybck=CopyBack|CBSHld=Copyback shielded
UBUnsp=UBad Unsupported|Rbld=Rebuild
```

- Identify the existing block devices (sda and sdb) on the host used by the OS and NetWitness. 'sdc' and 'sdd' are block devices corresponding to configured PowerVault. Use the command `lsblk` to list out the block devices.

```
[root@S6Core2 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 931G 0 disk
├─sda1 8:1 0 1G 0 part /boot
├─sda2 8:2 0 930G 0 part
│ ├─netwitness_vg00-root 253:0 0 30G 0 lvm /
│ ├─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
│ ├─netwitness_vg00-nwhome 253:6 0 2.7T 0 lvm /var/netwitness
│ ├─netwitness_vg00-varlog 253:7 0 10G 0 lvm /var/log
│ └─netwitness_vg00-usrhome 253:8 0 10G 0 lvm /home
sdb 8:16 0 1.8T 0 disk
├─sdb1 8:17 0 1.8T 0 part
│ └─netwitness_vg00-nwhome 253:6 0 2.7T 0 lvm /var/netwitness
sdc 8:32 0 2.9T 0 disk
├─sdc1 8:33 0 2.9T 0 part
│ └─index-index 253:5 0 2.9T 0 lvm /var/netwitness/concentrator/index
sdd 8:48 0 74.9T 0 disk
├─sdd1 8:49 0 74.9T 0 part
│ ├─concentrator-root 253:2 0 30G 0 lvm /var/netwitness/concentrator
│ ├─concentrator-sessiondb 253:3 0 7.5T 0 lvm /var/netwitness/concentrator/sessiondb
│ └─concentrator-metadb 253:4 0 67.3T 0 lvm /var/netwitness/concentrator/metadb
```

#### Configure Virtual Drive(s) on the Installed Meta Disk kit (Slots 4,5, and 6):

- Create the Virtual Drive/Drive Group (DG) on the internal controller using the disks in slot 4 through 6 using the below command.

```
/opt/MegaRAID/perccli/perccli64 /c0 add vd type=raid5 drives=64:4-6
strip=128
```

```
[root@S6Core2 ~]# /opt/MegaRAID/perccli/perccli64 /c0 add vd type=raid5 drives=64:4-6 strip=128
CLI Version = 007.1623.0000.0000 May 17, 2021
Operating system = Linux 3.10.0-1160.71.1.el7.x86_64
Controller = 0
Status = Success
Description = Add VD Succeeded.
```

- The new virtual drive shows up as '2/2' under DG/VD column.

```
/opt/MegaRAID/perccli/perccli64 /c0 /vall show
```



```
[root@S6Core2 ~]# /opt/MegaRAID/perccli/perccli64 /c0 /vall show
CLI Version = 007.1623.0000.0000 May 17, 2021
Operating system = Linux 3.10.0-1160.71.1.el7.x86_64
Controller = 0
Status = Success
Description = None

Virtual Drives :
=====

DG/VD TYPE State Access Consist Cache Cac SCC Size Name

0/0 RAID1 Optl RW Yes RWBD - OFF 931.000 GB
1/1 RAID1 Optl RW Yes RWBD - OFF 1.818 TB
2/2 RAID5 Optl RW No RWBD - OFF 6.984 TB

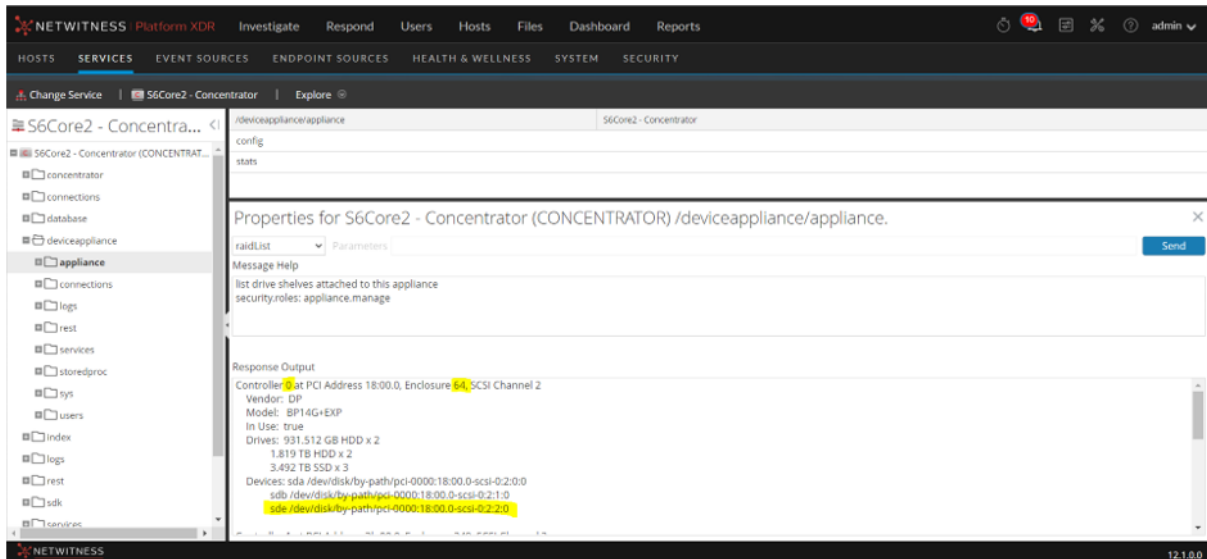
VD=Virtual Drive| DG=Drive Group|Rec=Recovery
Cac=CacheCade|Rec=Recovery|OfLn=OffLine|Pdgd=Partially Degraded|Dgrd=Degraded
Optl=Optimal|dflt=Default|RO=Read Only|RW=Read Write|HD=Hidden|TRANS=TransportReady
B=Blocked|Consist=Consistent|R=Read Ahead Always|NR=No Read Ahead|WB=WriteBack
FWB=Force WriteBack|WT=WriteThrough|C=Cached IO|D=Direct IO|sCC=Scheduled
Check Consistency
```

- Identify the new block device on the host. The block device name is identified under the NAME column. In this example the new block device is 'sdc'. This block device name is required when configuring storage. Use `lsblk` to list the block devices.

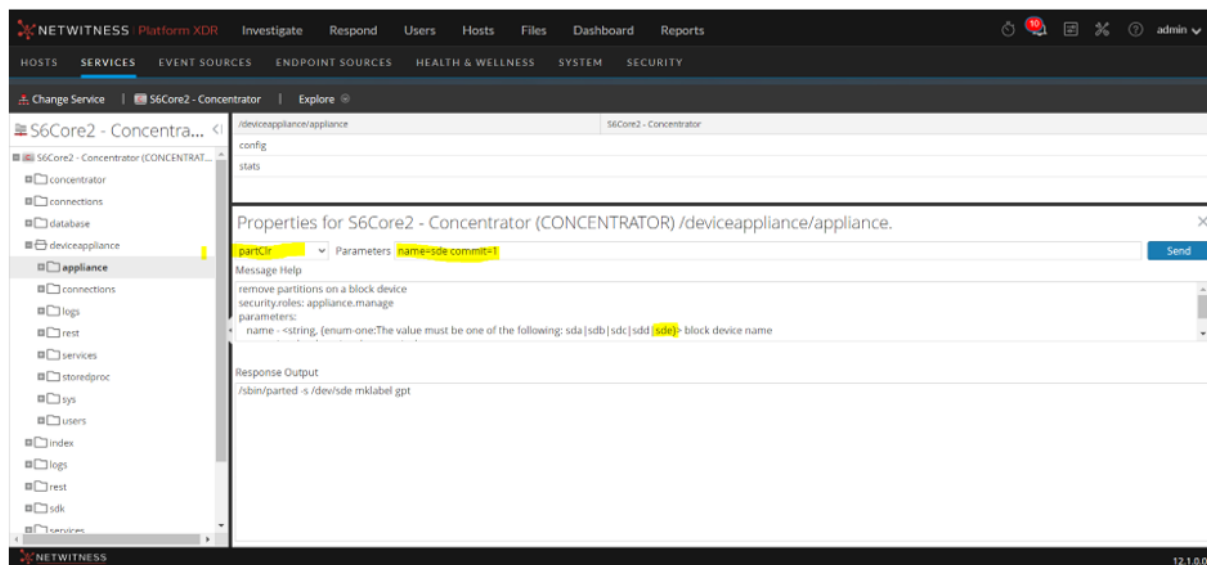
```
[root@S6Core2 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 931G 0 disk
├─sda1 8:1 0 1G 0 part /boot
├─sda2 8:2 0 930G 0 part
│ ├─netwitness_vg00-root 253:0 0 30G 0 lvm /
│ ├─netwitness_vg00-swap 253:1 0 4G 0 lvm [SWAP]
│ ├─netwitness_vg00-nwhome 253:6 0 2.7T 0 lvm /var/netwitness
│ ├─netwitness_vg00-varlog 253:7 0 10G 0 lvm /var/log
│ └─netwitness_vg00-usrhome 253:8 0 10G 0 lvm /home
sdb 8:16 0 1.8T 0 disk
├─sdb1 8:17 0 1.8T 0 part
└─netwitness_vg00-nwhome 253:6 0 2.7T 0 lvm /var/netwitness
sdc 8:32 0 2.9T 0 disk
├─sdc1 8:33 0 2.9T 0 part
└─index-index 253:5 0 2.9T 0 lvm /var/netwitness/concentrator/index
sdd 8:48 0 74.9T 0 disk
├─sdd1 8:49 0 74.9T 0 part
│ ├─concentrator-root 253:2 0 30G 0 lvm /var/netwitness/concentrator
│ ├─concentrator-sessiondb 253:3 0 7.5T 0 lvm /var/netwitness/concentrator/sessiondb
│ └─concentrator-metadb 253:4 0 67.3T 0 lvm /var/netwitness/concentrator/metadb
sde 8:64 0 7T 0 disk
└─sde1 8:65 0 7T 0 part
```

- Now, you must Configure the above block device as index for Concentrator. Use the Explorer view (Login into UI, select Hosts->Service->Action->View->Explorer->deviceAppliance->appliance->properties) to complete configuration. Use 'raidList' to retrieve the existing raid list. Note the block device name corresponding to the Meta Disk kit. In this case it is 'sde'.

**Note:** If a second Meta Disk kit is being configured, the block device would be 'sdf'.

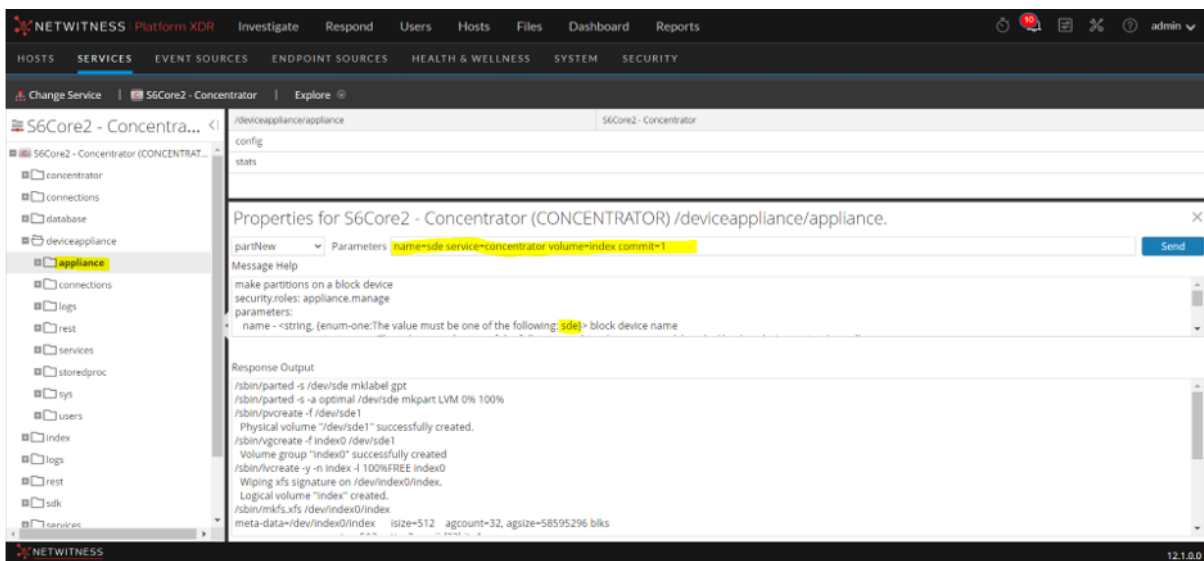


9. Clear partitions on the above block device ('sde') with the following parameters:  
name=sde commit=1

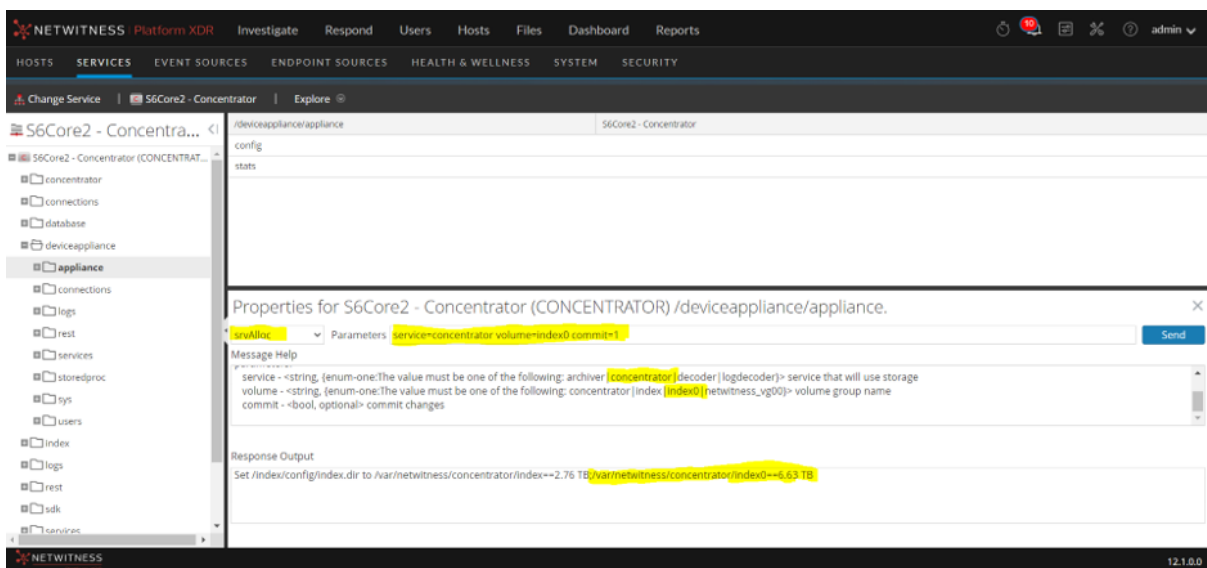


10. Make partitions on the block device ('sde') with the following parameters:  
name=sde service=concentrator volume=index commit=1

**Note:** If a second Meta Disk kit is being configured, the volume would be 'index1'.



- Allocate the index to Concentrator service using 'srvAlloc'  
`service=concentrator volume=index commit=1`



```
[root@S6Core2 ~]# df -hP
Filesystem Size Used Avail Use% Mounted on
devtmpfs 63G 0 63G 0% /dev
tmpfs 63G 20K 63G 1% /dev/shm
tmpfs 63G 11M 63G 1% /run
tmpfs 63G 0 63G 0% /sys/fs/cgroup
/dev/mapper/netwitness_vg00-root 30G 12G 19G 40% /
/dev/sda1 1014M 122M 893M 12% /boot
/dev/mapper/netwitness_vg00-nwhome 2.7T 2.7T 23G 100% /var/netwitness
/dev/mapper/concentrator-root 30G 6.8G 24G 23% /var/netwitness/concentrator
/dev/mapper/index-index 3.0T 1.9T 1.1T 65% /var/netwitness/concentrator/index
/dev/mapper/concentrator-sessiondb 7.5T 2.6T 5.0T 34% /var/netwitness/concentrator/sessiondb
/dev/mapper/netwitness_vg00-varlog 10G 212M 9.8G 3% /var/log
/dev/mapper/concentrator-metadb 68T 41T 27T 61% /var/netwitness/concentrator/metadb
/dev/mapper/netwitness_vg00-usrhome 10G 33M 10G 1% /home
tmpfs 13G 0 13G 0% /run/user/0
/dev/mapper/index0-index 7.0T 34M 7.0T 1% /var/netwitness/concentrator/index0
[root@S6Core2 ~]#
```

## Revision History

---

| Revision | Date         | Description          |
|----------|--------------|----------------------|
| 1.0      | August, 2022 | 12.0 Release Updates |