Arista Analytics Deployment Guide

Arista Networks

www.arista.com

Arista Analytics Deployment Guide

DOC-06774-01

Headquarters	Support	Sales
5453 Great America Parkway Santa Clara, CA 95054 USA +1-408-547-5500	+1-408-547-5502 +1-866-476-0000	+1-408-547-5501 +1-866-497-0000
www.arista.com/en/	support@arista.com	sales@arista.com

Contents

Chapter 1

Installing the Arista Analytics Node

This chapter describes the installation procedures for the Arista Analytics node on a Dell R440 server. It includes the following sections.

1.1 Arista Analytics

Arista Analytics provides single-pane-of-glass monitoring for production visibility, with historical analysis capability based on production network traffic metadata combined with information available on the DANZ Monitoring Fabric (DMF) Controller. Arista Analytics provides a collection of dashboards with visualizations on each dashboard that simplify analysis of production networks.

The Analytics server runs separately from the DMF Controller to allow allocation of adequate disk space and CPU memory without affecting the performance of the DANZ Monitoring Fabric.

1.2 Arista Analytics Node Hardware

The Arista Analytics Node is an appliance based on a Dell R440 server, running the Arista Analytics server. Running Arista Analytics on a dedicated hardware appliance ensures that sufficient hardware resources are allocated for good performance and prevents the Analytics service from affecting other applications running on the same device.

- Two management interfaces (10/100/1000Mb/s)
- One serial interface (db9)
- One VGA interface
- Two USB ports
- Two 10Gb SFP ports
- Two 10Gb Copper ports
- One dedicated IDRAC port

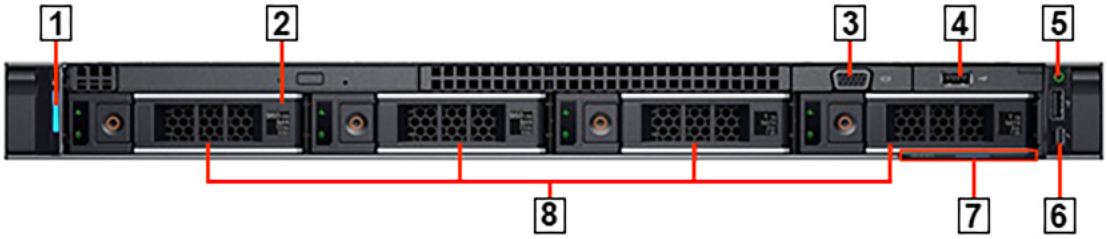
The Arista Analytics Node is available as an enterprise-class, 2-socket, 1-RU rack-mounted hardware appliance designed to deliver the right combination of performance, redundancy, and value in a high-density chassis. (HWA/HWA2).

Figure 1-1: Arista Analytics Node (HWA/HWA2) Bezel

1	Analytics Node Security Bezel	3	LCD panel
2	LCD menu buttons		

The following figure illustrates the front panel of the Arista Analytics Node.

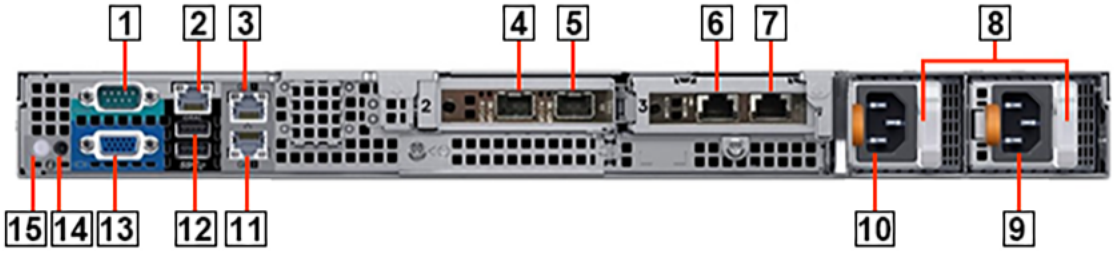
Figure 1-2: Arista Analytics Node (HWA/HWA2) Front Panel



1	System identification button /indicator	6	USB (not supported)
2	Optical drive	7	Information tag
3	Video connector	8	Hard drives Micro
4	USB ports		
5	Power-on indicator / Power button		

The following figure illustrates the rear panel of the Arista Analytics Node.

Figure 1-3: Arista Analytics Node (HWA/HWA2) Rear Panel



1	Serial connector (Default Baud Rate 115200)	9	Power Supply 2
2	iDRAC Ethernet interface	10	Power Supply 1
3	Ethernet connector 1 – Analytics Node Management port 1, Active (10/100/1000Mb/s)	11	Ethernet connector 2 – Analytics Node Management port 2, backup (10/100/1000Mb/s)
4	Ethernet connector 3 – Analytics Node 10-GbE SFP+ Collector Interface 1, Active	12	USB ports
5	Ethernet connector 4 – Analytics Node 10-GbE SFP+ Collector Interface 2, Backup	13	Video connector
6	Ethernet connector 5 – Not supported	14	System identification button
7	Ethernet connector 6 – Not supported	15	System identification indicator
8	PSU status indicators		

1.3 Installation Procedure

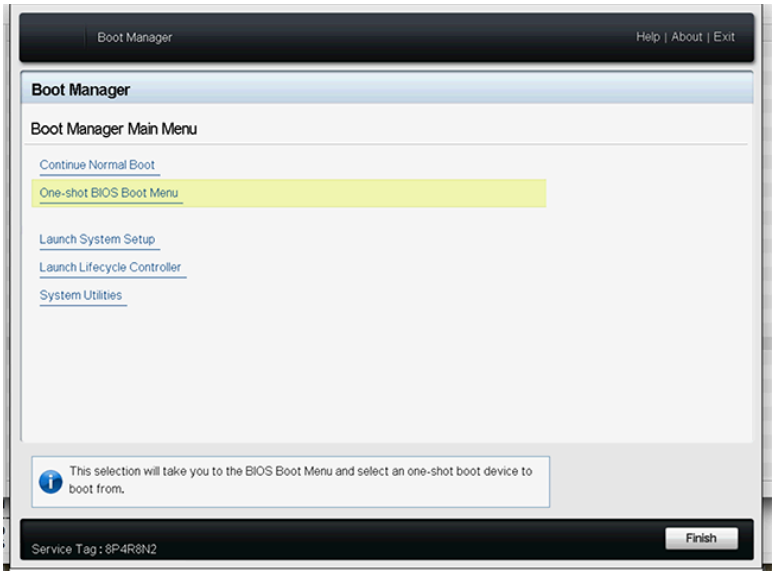
To install the Arista Analytics on the Dell R440 appliance, complete the following steps.

1. Rack the Arista Analytics Appliance.
   The appliance interfaces are on the rear of the appliance, where the power cord is connected.
2. Connect the upper leftmost analytics management interface (Named ***Gb 1***) to the management network.
3. Log in via the serial port using the admin account name. The baud rate is ***115200***.
4. Insert the USB drive with the current software image into the USB port of the Arista Analytics Node Appliance.
5. Power-cycle the appliance.
   The Boot Manager screen is displayed as shown below.

```
F2 = System Setup
F10 = Lifecycle Controller
F11 = Boot Manager
F12 = PXE Boot
Initializing Serial ATA devices . . .
```

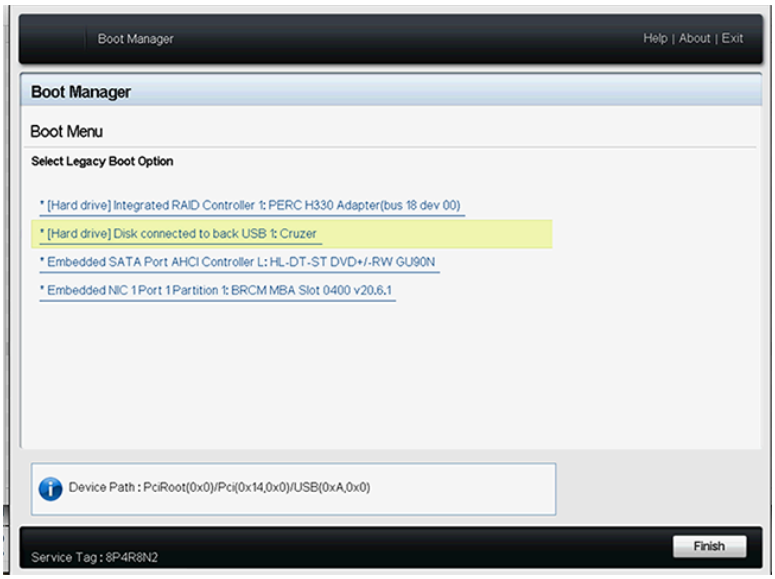
6. Press **F11** to select Boot Manager to allow booting from USB.
   The Boot Manager main menu is displayed.

Figure 1-4: Boot Manager Main Menu



7. Select One-shot BIOS Boot Menu.

Figure 1-5: Boot Menu



8. Select Disk connected to back USB 2.
9. When prompted on the system console, type yes to start the installation.
10. Complete the initial configuration of Arista Analytics.

1.4 Upgrading Arista Analytics Node

Prior to the upgrade, Arista recommends to take the back up of all custom objects. The procedure to import/export the custom object(s) is documented in ***Arista Analytics User guide*** (refer chapter **Backup and Restore**).

Single Node Upgrade can be achieved using the following steps.

1. Copy the ISO image to **`image://`**

```
analytics-1(config)# copy <HTTP_Link_to_analytics.iso> image://
Copying image from <HTTP_Link_to_analytics.iso>
Validating Image Contents: check for expected contents
Verifying image signature
Verifying image checksums
Validating Image Details
00:01:20: Completed
Image added: b4ffe
```

2. Stage Image:

```
analytics-1(config)# upgrade stage
Upgrade stage will overwrite alternate partition, proceed ("y" or "yes" to
 continue): yes
Verifying the integrity of the installation media
Staging the upgrade to DMF Analytics Node 8.1.0-alpha (analytics/master
 #935)
00:04:47: progress: 63% |*********************************->
```

3. Launch Upgrade:

```
analytics-1(config)# upgrade launch
Upgrade launch: DMF Analytics Node 8.1.0-alpha (analytics/master #935)
```

```
Upgrade launch: Various cluster members may be rebooted by automation
Upgrade launch: proceed? ("y" or "yes" to continue): yes
Upgrade launch: *WARNING* single-controller: upgrade will be non-redundant
Upgrade launch: non-redundant upgrade ("y" or "yes" to continue): yes
Upgrade launch: Various cluster members may be rebooted by automation
Upgrade launch: 07:52:00: Starting Upgrade
Upgrade launch: 07:52:00: origin version: DMF Analytics Node 8.1.0
Upgrade launch: 07:52:00: config updates are frozen: upgrade state: begin-
upgrade-old-active
Upgrade launch: 07:52:00: Completed; Ready for reboot
Upgrade state: current upgrade state: None
Upgrade launch: Moving boot partition to alternate
Upgrade launch: Successfully prepared for launch
None
00:00:06: Completed
```

Cluster Upgrades

Cluster upgrade can be achieved using the following steps on all 3 or 5 nodes simultaneously. The upgrade cluster commands need to be executed from the Active Analytics Node in the cluster.

1. Copy image:

```
analytics-1# copy <HTTP_Link_to_analytics.iso> image://cluster
analytics-1: Copying image from <HTTP_Link_to_analytics.iso>
analytics-2: Copying image from <HTTP_Link_to_analytics.iso>
analytics-3: Copying image from <HTTP_Link_to_analytics.iso>
analytics-1: Validating Image Contents: check for expected contents
analytics-1: Verifying image signature
analytics-1: Verifying image checksums
analytics-2: Validating Image Contents: check for expected contents
analytics-2: Verifying image signature
analytics-2: Verifying image checksums
analytics-3: Validating Image Contents: check for expected contents
analytics-3: Verifying image signature
analytics-3: Verifying image checksums
analytics-1: Validating Image Details
analytics-2: Validating Image Details
analytics-3: Validating Image Details
00:02:32: Completed
Image added: b4ffe
```

2. Stage Image:

```
analytics-1# upgrade cluster stage
Upgrade stage will overwrite alternate partition, proceed ("y" or "yes" to
 continue): yes
analytics-1: Verifying the integrity of the installation media
analytics-2: Verifying the integrity of the installation media
analytics-3: Verifying the integrity of the installation media
analytics-1: Staging the upgrade to DMF Analytics Node 8.2.0 (analytics/dm
f-8.2.0 #6)
analytics-2: Staging the upgrade to DMF Analytics Node 8.2.0 (analytics/dm
f-8.2.0 #6)
analytics-3: Staging the upgrade to DMF Analytics Node 8.2.0 (analytics/dm
f-8.2.0 #6)
00:06:31: Completed
00:10:44: Completed
Upgrade stage: info: *analytics-1: This release is: DMF Analytics Node 8.2.0
 (analytics/dmf-8.
‹!2.0 #6)
Upgrade stage: info: *analytics-1: Alternate partition Release: DMF
 Analytics Node 8.2.0
```

```
‹!(analytics/dmf-8.2.0 #6)
Upgrade stage: info: *analytics-1: Alternate Partition Formatted
Upgrade stage: info: *analytics-1: Alternate Partition is: /dev/flvg/root1
Upgrade stage: info: *analytics-1: All node(s) connected
Upgrade stage: info: *analytics-1: Alternate partition staged
Upgrade stage: validation: *analytics-1: Sync interface: bond1/bond0 is up
Upgrade stage: info: *analytics-1: All Application Validation Checks
 completed
Upgrade stage: info: *analytics-1: Ready for upgrade
Upgrade stage: info: analytics-3: This release is: DMF Analytics Node 8.2.0
 (analytics/dmf-8.
‹!2.0 #6)
Upgrade stage: info: analytics-3: Alternate partition Release: DMF Analytics
 Node 8.2.0
‹!(analytics/dmf-8.2.0 #6)
Upgrade stage: info: analytics-3: Alternate Partition Formatted
Upgrade stage: info: analytics-3: Alternate Partition is: /dev/flvg/root2
Upgrade stage: info: analytics-3: All node(s) connected
Upgrade stage: info: analytics-3: Alternate partition staged
Upgrade stage: validation: analytics-3: Sync interface: bond1/bond0 is up
Upgrade stage: info: analytics-3: All Application Validation Checks
 completed
Upgrade stage: info: analytics-3: Ready for upgrade
Upgrade stage: info: analytics-2: This release is: DMF Analytics Node 8.2.0
 (analytics/dmf-8.
‹!2.0 #6)
Upgrade stage: info: analytics-2: Alternate partition Release: DMF Analytics
 Node 8.2.0
‹!(analytics/dmf-8.2.0 #6)
Upgrade stage: info: analytics-2: Alternate Partition Formatted
Upgrade stage: info: analytics-2: Alternate Partition is: /dev/flvg/root1
Upgrade stage: info: analytics-2: All node(s) connected
Upgrade stage: info: analytics-2: Alternate partition staged
Upgrade stage: validation: analytics-2: Sync interface: bond1/bond0 is up
Upgrade stage: info: analytics-2: All Application Validation Checks
 completed
Upgrade stage: info: analytics-2: Ready for upgrade
```

3. Verify image has been staged successfully.

```
analytics-1# show cluster boot partition
# Node name Vol State Upgrade Product Version Build
-|----------------|-----|------------|---------|------------------|-----
--|-----|
1 analytics-1 root1 staged DMF Analytics Node 8.3.0 12
2 analytics-1 root2 Active, Boot completed DMF Analytics Node 8.2.0 6
3 analytics-3 root1 Active, Boot completed DMF Analytics Node 8.2.0 6
4 analytics-3 root2 staged DMF Analytics Node 8.3.0 12
5 analytics-2 root1 staged DMF Analytics Node 8.3.0 12
6 analytics-2 root2 Active, Boot completed DMF Analytics Node 8.2.0 6
analytics-1#
```

4. Verify all pre-upgrade launch checks will pass.

```
analytics-1# upgrade cluster pre-launch-check
info: *analytics-1: This release is: DMF Analytics Node 8.2.0 (analytics/dm
f-8.2.0 #6)
info: *analytics-1: Alternate partition Release: DMF Analytics Node 8.2.0
 (analytics/dmf-8.2.0 #6)
info: *analytics-1: Alternate Partition Formatted
info: *analytics-1: Alternate Partition is: /dev/flvg/root1
info: *analytics-1: All node(s) connected
info: *analytics-1: Alternate partition staged
```

```
validation: *analytics-1: Sync interface: bond1/bond0 is up
info: *analytics-1: All Application Validation Checks completed
info: *analytics-1: Ready for upgrade
info: analytics-3: This release is: DMF Analytics Node 8.2.0 (analytics/dm
f-8.2.0 #6)
info: analytics-3: Alternate partition Release: DMF Analytics Node 8.2.0
 (analytics/dmf-8.2.0 #6)
info: analytics-3: Alternate Partition Formatted
info: analytics-3: Alternate Partition is: /dev/flvg/root2
info: analytics-3: All node(s) connected
info: analytics-3: Alternate partition staged
validation: analytics-3: Sync interface: bond1/bond0 is up
info: analytics-3: All Application Validation Checks completed
info: analytics-3: Ready for upgrade
info: analytics-2: This release is: DMF Analytics Node 8.2.0 (analytics/dm
f-8.2.0 #6)
info: analytics-2: Alternate partition Release: DMF Analytics Node 8.2.0
 (analytics/dmf-8.2.0 #6)
info: analytics-2: Alternate Partition Formatted
info: analytics-2: Alternate Partition is: /dev/flvg/root1
info: analytics-2: All node(s) connected
info: analytics-2: Alternate partition staged
validation: analytics-2: Sync interface: bond1/bond0 is up
info: analytics-2: All Application Validation Checks completed
info: analytics-2: Ready for upgrade
analytics-1#
```

5. Launch upgrade.

```
analytics-1# upgrade cluster launch
Upgrade launch: DMF Analytics Node 8.2.0 (analytics/dmf-8.2.0 #6)
Upgrade launch: Various cluster members and managed devices may be rebooted
 by automation
Upgrade launch: proceed? ("y" or "yes" to continue): yes
UpgradeProgress: 0 analytics-1: This release is: DMF Analytics Node 8.2.0
 (analytics/dmf-8.2.0 #6)
UpgradeProgress: 1 analytics-1: Alternate partition Release: DMF Analytics
 Node 8.2.0 (analytics/dmf-8.2.0 #6)
UpgradeProgress: 2 analytics-1: Alternate Partition Formatted
UpgradeProgress: 3 analytics-1: Alternate Partition is: /dev/flvg/root1
UpgradeProgress: 4 analytics-1: All node(s) connected
UpgradeProgress: 5 analytics-1: Alternate partition staged
UpgradeProgress: 6 analytics-1: All Application Validation Checks completed
UpgradeProgress: 7 analytics-1: Upgrade launch: Various cluster members may
 be rebooted by automation
UpgradeProgress: 8 analytics-3: This release is: DMF Analytics Node 8.2.0
 (analytics/dmf-8.2.0 #6)
UpgradeProgress: 9 analytics-3: Alternate partition Release: DMF Analytics
 Node 8.2.0 (analytics/dmf-8.2.0 #6)
UpgradeProgress: 10 analytics-3: Alternate Partition Formatted
UpgradeProgress: 11 analytics-3: Alternate Partition is: /dev/flvg/root2
UpgradeProgress: 12 analytics-3: All node(s) connected
UpgradeProgress: 13 analytics-3: Alternate partition staged
UpgradeProgress: 14 analytics-3: All Application Validation Checks completed
UpgradeProgress: 15 analytics-3: Upgrade launch: Various cluster members may
 be rebooted by automation
UpgradeProgress: 16 analytics-3: Upgrade launch: saving running-config as:
 upgrade-snapshot
UpgradeProgress: 17 analytics-3: Upgrade launch: saving running-config to
 file: upgrade-rc
UpgradeProgress: 18 analytics-2: This release is: DMF Analytics Node 8.2.0
 (analytics/dmf-8.2.0 #6)
UpgradeProgress: 19 analytics-2: Alternate partition Release: DMF Analytics
 Node 8.2.0 (analytics/dmf-8.2.0 #6)
```

```
UpgradeProgress: 20 analytics-2: Alternate Partition Formatted
UpgradeProgress: 21 analytics-2: Alternate Partition is: /dev/flvg/root1
UpgradeProgress: 22 analytics-2: All node(s) connected
UpgradeProgress: 23 analytics-2: Alternate partition staged
UpgradeProgress: 24 analytics-1: Upgrade launch: saving running-config as:
 upgrade-snapshot
UpgradeProgress: 25 analytics-1: Upgrade launch: saving running-config to
 file: upgrade-rc
UpgradeProgress: 26 analytics-2: All Application Validation Checks completed
UpgradeProgress: 27 analytics-2: Upgrade launch: Various cluster members may
 be rebooted by automation
UpgradeProgress: 28 analytics-2: Upgrade launch: saving running-config as:
 upgrade-snapshot
UpgradeProgress: 29 analytics-2: Upgrade launch: saving running-config to
 file: upgrade-rc
Upgrade launch: Starting Upgrade: async-id 82ajhKaJvWRaK0TDpZZ_MRTCi7QMIUQ6
Upgrade launch: disconnecting from launch
Upgrade launch: use: 'show upgrade progress' for progress on this controller
Upgrade launch: use: 'upgrade abort' to abort upgrade on this controller
upgrade started id:82ajhKaJvWRaK0TDpZZ_MRTCi7QMIUQ6
Upgrade launch disconnected from background task
analytics-1#
```

Note: In some cases, if the shard count is too high, upgrade will not proceed. In such a situation the following will need to run:

```
analytics-1> enable
analytics-1# debug bash
****************************** WARNING ******************************
Any/All activities within bash mode are UNSUPPORTED
This is intended ONLY for additional debugging ONLY by Arista TAC.
Please type "exit" or Ctrl-D to return to the CLI
****************************** WARNING ******************************
admin@analytics-1:~$ nohup /opt/bigswitch/reindex.sh > reindex.log &
```

Note: The containers could take 10-20 minutes to come up after upgrade.

Chapter 2

Setting Up the Arista Analytics Node

This chapter describes the installation and configuration procedures for Arista Analytics. This chapter contains the following sections:

2.1 Requirements

Arista Analytics node can be deployed with or without DANZ Monitoring Fabric (DMF). You need the following information before installing the Arista Analytics node:

- IP address and netmask to assign to the Analytics server
- Default IP gateway
- DNS server IP address (optional)
- DNS Search Domain (optional)
- Admin password for the Analytics server
- NTP server IPv4 address
- Password for Analytics GUI admin user (optional)
- TACACS+ Server IPv4 Address (optional)
- TACACS+ secret (optional)
- TACACS+ Server Service (optional)

When deploying the Arista Analytics node along with DMF, you will need the following additional information.

- IP addresses for the DMF Controllers

Note: If Arista Analytics node is deployed along with DMF, ensure that the version on the Arista Analytics node is the same as that running on the DMF Controllers. Running different versions on the Arista Analytics node and DMF Controllers are not supported.

The ports in the following table should be open on security devices between the Controller or switches and the Arista Analytics server, as noted in the table.

In addition, you need to open the ports for Redis and replicated Redis on the Analytics server after first boot configuration (see the Enabling Access Control to the Analytics Server section).

Table 1: Arista Analytics Open Port Requirements

Monitoring	Port Requirement	Explanation
NetFlow	UDP 2055	Flow data exported to the Analytics node in NetFlow v5 format, either from the production network or the DANZ Monitoring Fabric.
IPFIX	UDP 4739	Flow data exported to the Analytics node in IPFIX/NetFlow v10 format, either from the production network or the DANZ Monitoring Fabric.
sFlow	UDP 6343 between switches and Analytics server	Packets are sampled on filter interfaces and the SwitchLight OS sFlow agent constructs the sFlow header and forwards to Analytics server and other sFlow collectors for processing.
Host-tracker information	UDP 6380 between switches and Analytics server	ARP, DNS, and other control traffic is forwarded from each switch to the Analytics server. A private header is prepended with a timestamp in the process. The Analytics server processes packets and maintains the host tracking database. The Controller queries the Analytics server for the latest host table.
DMF Statistics and Events	UDP 9379 (redis) between Controller and Analytics server	Statistics gathered by the Controller from switches and service nodes are sent to the Analytics server from REDIS database.
DMF Statistics and Events (cluster)	UDP 6379 (replicated redis) between Controller and Analytics server	Replicated redis is used to gather information with a DMF Controller cluster.
Monitoring Active Directory or Open VPN	UDP 5043	Required only if you are using Analytics to monitor Active Directory or Open VPN.

2.2 Arista Analytics Node First Boot Configuration

Note: Before attempting to reinstall the ISO image on an existing analytics node, run **`sudo /opt/bigswitch/rma.sh`**.

To complete the initial configuration of Arista Analytics, complete the following steps.

1. Respond to the system prompt to login using the admin account.

```
analytics login: admin
Login: admin, on Wed 2018-10-31 18:22:24 UTC, from localhost
```

2. When prompted, accept the End User License Agreement (EULA).

```
This product is governed by an End User License Agreement (EULA).
You must accept this EULA to continue using this product.
You can view this EULA by typing 'View', or from our website at:
https://www.arista.com/en/eula
Do you accept the EULA for this product? (Yes/No/View) [Yes] > Yes
Running system pre-check
Finished system pre-check
```

```
Starting first-time setup
```

3. Enter the emergency recovery user password.

```
Local Node Configuration
------------------------
Emergency recovery user password >
Emergency recovery user password (retype to confirm) >
```

4. Assign a hostname to the Analytics Node.

```
Hostname > analytics1
```

5. Choose the management network option.

```
Management network options:
[1] IPv4 only
[2] IPv6 only
[3] IPv4 and IPv6
>1
```

6. Enter the IP address to assign to the Arista Analytics Server as in the following example.

```
Configuration IPv4 Address: 10.9.18.220
```

If you do not enter a CIDR, the system prompts you for the IPv4 subnet mask.

```
IPv4 address [0.0.0.0/0] > 10.9.40.100/24
IPv4 gateway (Optional) > 10.9.40.1
DNS server 1 (Optional) > 10.3.0.4
DNS server 2 (Optional) > 10.1.5.200
DNS search domain (Optional) > qa.bigswitch.com
```

7. Starting with ***DMF 7.3.0*** release, a three node analytics cluster is supported. This is for added performance and reliability. Select the clustering option and information.

If you have a standalone analytics node or the current node you are configuring is the first node of the analytics cluster then select option:

```
[1] Start a new cluster
```

Note: Wait for the active node to load completely (ES and Kibana) before executing the first boot script on the other cluster nodes.

```
Clustering
----------
Analytics cluster options:
[1] Start a new cluster
[2] Join an existing cluster
> 1
Cluster name > analytics-test
Cluster description (Optional) > testing
Cluster administrator password >
Cluster administrator password (retype to confirm) >
```

If you have already setup you active/master node of the analytics cluster and you want additional nodes to join the cluster then select:

[2] Join an existing cluster

```
Clustering
----------
```

```
Analytics cluster options:
[1] Start a new cluster
[2] Join an existing cluster
> 2
Existing Analytics Node address > <ip_of_active_analytics_node>
Cluster administrator password >
Cluster administrator password (retype to confirm) >
```

8. Enter the IP addresses of up to four Network Time Protocol (NTP) servers, which will use to synchronize the system time.

```
Default NTP servers:
- 0.bigswitch.pool.ntp.org
- 1.bigswitch.pool.ntp.org
- 2.bigswitch.pool.ntp.org
- 3.bigswitch.pool.ntp.org
NTP server options:
[1] Use default NTP servers
[2] Use custom NTP servers
[1] > 1
```

After completing the required configuration, the system displays the following messages and a prompt to confirm the settings to be applied.

```
Menu ----
Please choose an option:
[ 1] Apply settings
[ 2] Reset and start over
[ 3] Update Recovery Password (*****)
[ 4] Update Hostname (analytics-1)
[ 5] Update IP Option (IPv4 only)
[ 6] Update IPv4 Address (10.9.40.100/24)
[ 7] Update IPv4 Gateway (10.9.40.1)
[ 8] Update DNS Server 1 (10.3.0.4)
[ 9] Update DNS Server 2 (10.1.5.200)
[10] Update DNS Search Domain (qa.bigswitch.com)
[11] Update Cluster Option (Start a new cluster)
[12] Update Cluster Name (analytics-cluster)
[13] Update Cluster Description (testing)
[14] Update Admin Password (*****)
[15] Update NTP Option (Use default NTP servers)
[1] > 1
[Stage 1] Initializing system
[Stage 2] Configuring local node
Waiting for network configuration IP address on bond0 is 10.9.40.100
 Generating cryptographic keys
[Stage 3] Configuring system time
Initializing the system time by polling the NTP servers:
0.bigswitch.pool.ntp.org
1.bigswitch.pool.ntp.org
2.bigswitch.pool.ntp.org
3.bigswitch.pool.ntp.org
[Stage 4] Configuring cluster
Cluster configured successfully. Current node ID is 20445
All cluster nodes:
Node 20445: [fe80::d294:66ff:fe4f:5746]:6642
First-time setup is complete!
```

9. If you plan to install multiple Analytics node in cluster configuration, then go back to **Step 1**. and re-do the steps for the other nodes in the cluster.
10. After the system completes the configuration, you can establish an SSH session to the active Analytics Node the IP address configured during installation.

11. If you have an analytics cluster configured, then SSH to the active/master Analytics node and configure a Virtual IP address. Else skip to **Step 14**.

```
analytics-1 > enable
analytics-1 # configure
analytics-1(config)# cluster
analytics-1(config-cluster)# virtual ip <virtual_ip>
```

12. Next verify that the cluster has been successfully setup.

```
analytics-1 > enable
analytics-1 # show cluster
Cluster Name : analytics-cluster
Cluster Virtual IP : 10.106.4.19
Redundancy Status : redundant
Last Role Change Time : 2019-10-23 22:38:39.083000 UTC
Failover Reason : Changed connection state: cluster configuration changed
Cluster Uptime : 1 week, 5 days
# IP          @ State   Uptime
-|-----------|-|-------|--------------|
1 10.106.4.21 * active  1 week, 5 days
2 10.106.4.20   standby 1 week, 5 days
3 10.106.4.22   standby 1 week, 5 days
analytics-active-server #
```

13. In config mode on the Active DMF Controller, configure the Analytics server IP address by entering the following command from the **config-analytics** submode.

```
analytics-server address <ip>
```

For example, the following commands configure the Analytics server with the IP address ***10.9.18.220***.

```
dmf-controller1(config)# analytics
dmf-controller1(config-analytics)# analytics-server address 10.9.18.220
```

To use the Analytics GUI, click the **System > Configuration** tab at the top of the page, and click the DMF Controller link in the right panel.

14. Configure sFlow on the DMF Controller or other sFlow agents.

On the DMF Active Controller, configure the Analytics server IP address as an sFlow collector by entering the following commands.

```
dmf-controller1(config)# sflow default
dmf-controller1(config-sflow)# collector 10.106.4.19
```

This example configures the Virtual IP of the Analytics cluster with the IP address ***10.106.4.19*** and the default UDP ***port 6343*** as an sFlow collector.

The CLI enters config-sflow mode, from where you can enter the commands available for configuring sFlow on the DANZ Monitoring Fabric.

To use the DMF GUI, select the **Maintenance > sFlow** option.

2.3 Using the Arista Analytics Server CLI

Starting in the ***DMF 7.0 release***, administrative access to Arista Analytics and other server-level operations, such as configuring sFlow and creating a support bundle, are completed on the DMF Active Controller. For details, refer to the latest version of the ***DANZ Monitoring Fabric Deployment Guide***, available here: https://www.arista.com/en/support/software-download/dmf-ccf-mcd.

Operations that are specific to Arista Analytics are performed by using the Analytics server CLI after logging in to the Analytics server at the address assigned during the first boot configuration.

The Analytics CLI provides a subset of the commands available on the DMF Controller. For details about any command, enter **Help <command>** or press the **Tab** to see the options available. You can refer to the ***DANZ Fabric Command Reference Guide*** for information about the DMF Controller commands, which are similar to the Analytics commands.

The following shows the commands available from Login mode.

```
analytics-1> Tab
debug exit logout ping6 show upload
echo help no reauth support watch
enable history ping set terminal whoami
```

The following shows the additional commands available from **enable** mode.

```
analytics-1> enable
analytics-1# <Tab>
boot compare copy diagnose sync upgrade
clear configure delete reset system
```

The following shows the additional commands available from **Config** mode.

```
analytics-1# config
analytics-1(config)# <Tab>
aaa crypto local radius snmp-server version
banner end logging secure tacacs
cluster group ntp service user
```

2.4 Enabling Access Control to the Analytics Server

DANZ Monitoring Fabric (DMF) statistics and events and DMF switch/interface details (which are required for certain visualizations on the Analytical Node(AN)) are advertised through Redis and replica-Redis. The following is mandatory for DMF-AN integration:

1. Configuring AN (Virtual IP) IP on the DMF Controller.
2. Allowing DMF physical IPs under Redis/replicated ACL on the AN.

To enable access to the Analytics server for Redis and replicated Redis, complete the following steps.

1. Log in to the Analytics Server CLI.
2. Change to config-cluster-access submode.

```
analytics-1> enable
analytics-1# config
analytics-1(config)# cluster
analytics-1(config-cluster)# access-control
analytics-1(config-cluster-access)#
```

3. Define an access-list for Redis.

```
analytics-1(config-cluster-access)# access-list redis
analytics-1(config-cluster-access-list)# 1 permit from ip-address/cidr
```

Replace ***ip-address/cidr*** with the IP address or subnet ID and subnet mask where the Controller is running.

4. Define an access-list for replicated Redis.

```
analytics-1(config-cluster-access)# access-list replicated-redis
analytics-1(config-cluster-access-list)# 1 permit from ip-address/cidr
```

2.4.1 Adding Access Control to GUI

This section describes adding an access control list (ACL) command to the DANZ Monitoring Fabric (DMF) supported commands family.

1. To enable access to the Analytics Node (AN) User Interface (UI) from specific IP addresses or ranges of IP addresses, apply the new CLI command in the following manner:

```
DMF-ANALYTICS-CLUSTER> enable
DMF-ANALYTICS-CLUSTER# configure
DMF-ANALYTICS-CLUSTER(config)# cluster
DMF-ANALYTICS-CLUSTER(config-cluster)# access-control
DMF-ANALYTICS-CLUSTER(config-cluster-access)# access-list
<Access list name>     Enter an access list name: Enter an access list name
active-directory       Configure access-list for active-directory
api                    Configure access-list for api
gui                    Configure access-list for gui
ipfix                  Configure access-list for ipfix
netflow                Configure access-list for netflow
redis                  Configure access-list for redis
replicated-redis       Configure access-list for replicated-redis
snmp                   Configure access-list for snmp
ssh                    Configure access-list for ssh
DMF-ANALYTICS-CLUSTER(config-cluster-access)#
```

Refer to the ***DMF User guide*** for more information on Analytics ACL for GUI.

2.5 Configuring sFlow

sFlow is an industry-standard technology, defined by RFC 3176, for monitoring high-speed switched networks. sFlow defines methods for sampling packets and counters in the data path, and for forwarding the results to a sFlow collector for analysis and display. The DANZ Monitoring Fabric (DMF) supports sFlow to capture information about the production network and for troubleshooting the monitoring fabric.

For information about advanced search and analysis of historical sFlow messages using the Arista Analytics Graphical User Interface (GUI), refer to the latest edition of the ***Arista Analytics User Guide***.

You can either configure the DANZ Monitoring Fabric Controller with global sFlow settings that apply uniformly to all DANZ Monitoring Fabric switches or configure different sFlow settings on a per-switch basis. These settings, in general, define the following:

- IP address and port number of one or more sFlow collectors: identifies one or more sFlow collectors to which to send the sFlow packets. The default UDP port number is ***6343***.
- Sample rate: specifies the number of packets to transmit before sending a sFlow packet. Sampling is enabled on all filter interfaces and disabled on core interfaces and delivery interfaces. The default sample is ***1*** packet per ***10,000*** packets.

Note: Due to switch architecture rate limiting, the maximum effective number of sFlow packets per second is ***100***.

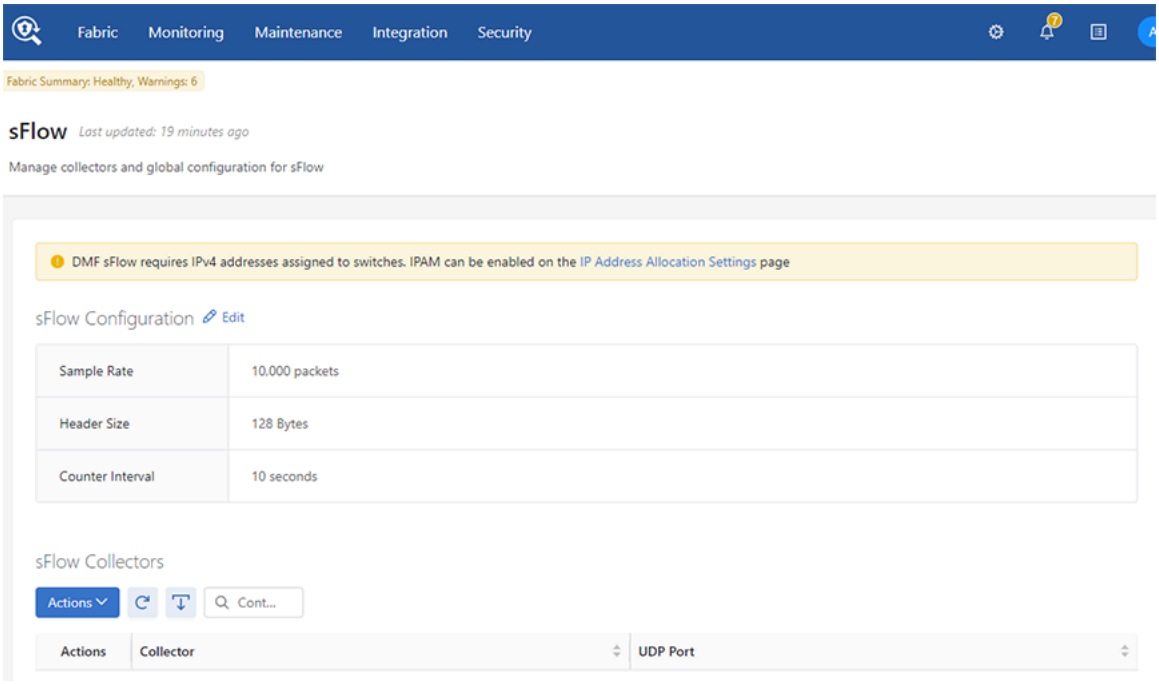
If the sFlow collector is on a device external to the DANZ Monitoring Fabric, a static route to the collector must be configured on the external tenant logical router.

2.5.1 Using the DMF Controller GUI to Configure sFlow

To enable sFlow, add Analytics or other collectors, or change the default parameters, complete the following steps.

1. To enable sFlow, select **Maintenance** > **sFlow** from the main menu.

Figure 2-1: sFlow Configuration



To view information about existing sFlow collectors, click the Expansion control to the left of the entry on the Collectors table. The system displays details about the switch counters associated with the specific collector.

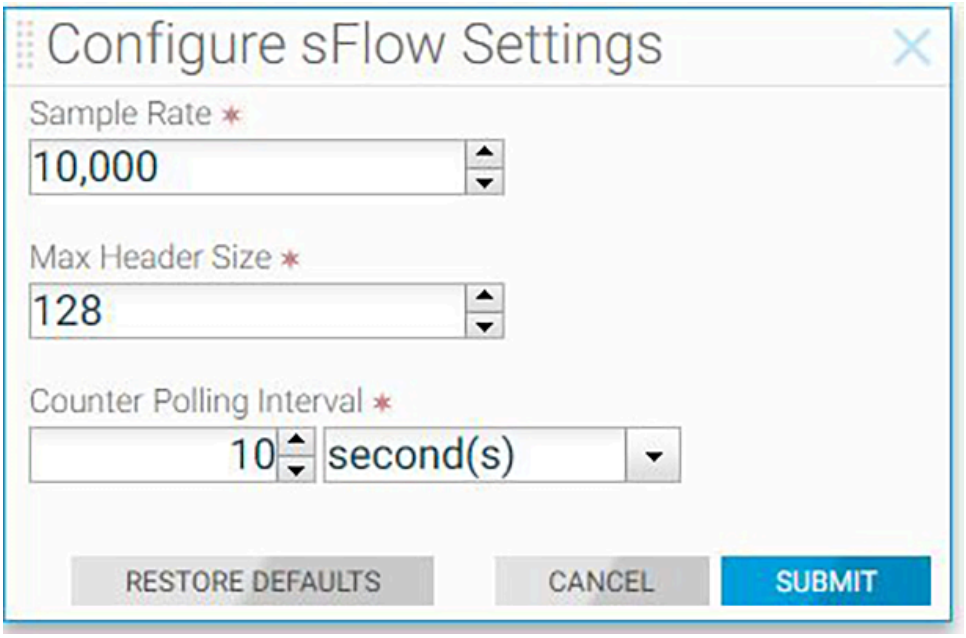
To activate or deactivate sFlow on a fabric-wide basis, click the **Settings** control to the left of the Configuration section and move the slider to Active to activate or to Inactive to deactivate.

2. You can add up to four sFlow collectors. To add a sFlow collector, first click the **Provision control (+)** in the upper left corner of the Collectors table.

Figure 2-2: Create sFlow Collector

3. Type the IP address of the sFlow collector.
4. Use the spinner to select the UDP port number used by the sFlow collector.
5. Select the tenant where the sFlow agent should collect sFlow messages.
6. Select the segment where the sFlow agent should collect sFlow messages. The default port is ***6343***.
7. Click **Save**.
8. (Optional) To view or change the default sFlow settings, select **Maintenance** > **sFlow**

Figure 2-3: Configure sFlow Settings Dialog



9. To change the sFlow global settings, click the **Settings** control to the left of the Configuration section.
10. Change the default settings for properties as required and click **Submit**.

2.5.2 Using the DMF Controller CLI to Configure sFlow

Configure the Analytics server IP address as an sFlow collector by entering the following commands.

```
dmf-Controller1(config)# sflow default
dmf-Controller1(config-sflow)# collector 10.106.1.57
```

This example configures the Analytics server with the IP address ***10.106.1.57*** and the default UDP ***port 6343*** as a sFlow collector.

The CLI enters sFlow Configuration Mode, from which you can enter the commands available for configuring sFlow on the DANZ Monitoring Fabric. For example, the following command identifies a sFlow collector at ***10.106.1.57*** using UDP ***port 6343***.

```
dmf-Controller-1(config-sflow)# collector 10.106.1.57 udp-port 6343
```

The default UDP port is ***6343***. Up to four collectors can be defined by entering the collector command for each collector.

The following command defines a header size of ***128*** bytes, a sample rate of ***1*** per ***1,000*** packets, and the counter interval of ***10*** seconds:

```
dmf-Controller-1(config)# show running-config sflow
! sflow
sflow
```

```
collector 10.106.1.57
collector 10.106.1.58
collector 10.106.1.59
counter-interval 10
header-size 128
sample-rate 100
dmf-Controller-1(config)#
```

2.6 Managing the Arista Analytics Server Software

This section describes operations for managing the Arista Analytics server.

2.6.1 Verifying the Analytics Server Version

To view the version of the Analytics server, enter the following command.

```
analytics-1# show version
Controller Version : DMF Analytics Node 8.0.0 (bigswitch/analytics/dmf-8.0.0
 #28)
```

2.6.2 Resetting to the Factory Default Configuration

To reset the Arista Analytics server to the factory default configuration, enter the following command.

```
analytics-1(config)# boot factory-default
boot factory-default: alternate partition will be overwritten
boot factory-default: proceed ("y" or "yes" to continue):
```

2.6.3 Password Reset

Resetting the Analytics Server Administrative Password

To reset the administrative password for the Analytics server, enter the following commands.

```
analytics-1# config
analytics-1(config)# reset user-password
Changing password for: admin
Current password:
New password:
Re-enter:
analytics-1(config)#
```

Resetting Password for Recovery User

To reset the password for the recovery user, please follow one of the following procedures. The steps need to be performed on both the Controllers of the cluster as resetting the password of the recovery user on one Controller won't change it for the recovery user on the other Controller.

1. Using Controller's Bash:
   a. Go to Controller Bash by executing **debugbash** command.
   b. Execute **sudo passwd recovery** command.

```
admin@Controller-1:~$ sudo passwd recovery
New password:
Retype new password:
```

```
passwd: password updated successfully
admin@Controller-1:~$
```

2. From recovery account login:

Note: For this to work, the customer needs to know the current password for the ***recovery*** user.

```
recovery@Controller-1:~$ passwd recovery
Changing password for recovery.
Current password:
New password:
Retype new password:
passwd: password updated successfully
recovery@Controller-1:~$
```

3. Using the **`API/api/v1/rpc/Controller/os/action/system-user/reset-password`**:

   The API call below will reset the ***recovery*** user's password to ***AdminAdmin***. The example given below is using **`curl`** initiated from a Linux host, but any rest client can be used to call the API.

```
curl -g -H "Cookie: session_cookie=<session_cookie>" 'https://<Controller
 IP>:8443/api/v1/
rpc/Controller/os/action/system-user/reset-password' -d '{"user-name" :
 "recovery","password" : "AdminAdmin"}' -X POST
```

Resetting Password for Admin and Other Local Users

To reset the password for ***admin*** and other local users, log in to the Controller using ***recovery*** user credentials. Use **`floodlight-reset-password`** to reset the user's password. The following example resets the ***admin*** user's password.

```
recovery@Controller-1:~$ floodlight-reset-password --user admin
Enter new admin password:
Re-enter new admin password:
Password updated for user admin
recovery@Controller-1:~$
```

The following example resets the password for a ***guest*** who is a ***read-only*** group user.

```
recovery@Controller-1:~$ floodlight-reset-password --user guest
Enter new guest password:
Re-enter new guest password:
Password updated for user guest
recovery@Controller-1:~$
```

2.6.4 Restarting the Analytics Server

If the Analytics server needs to be restarted for any reason, complete the following steps.

1. Reboot the Controller from the CLI using the following command.

```
analytics-1# system reboot controller
```

2. In the case of a three-node analytics cluster, repeat the above command on every member of the cluster.

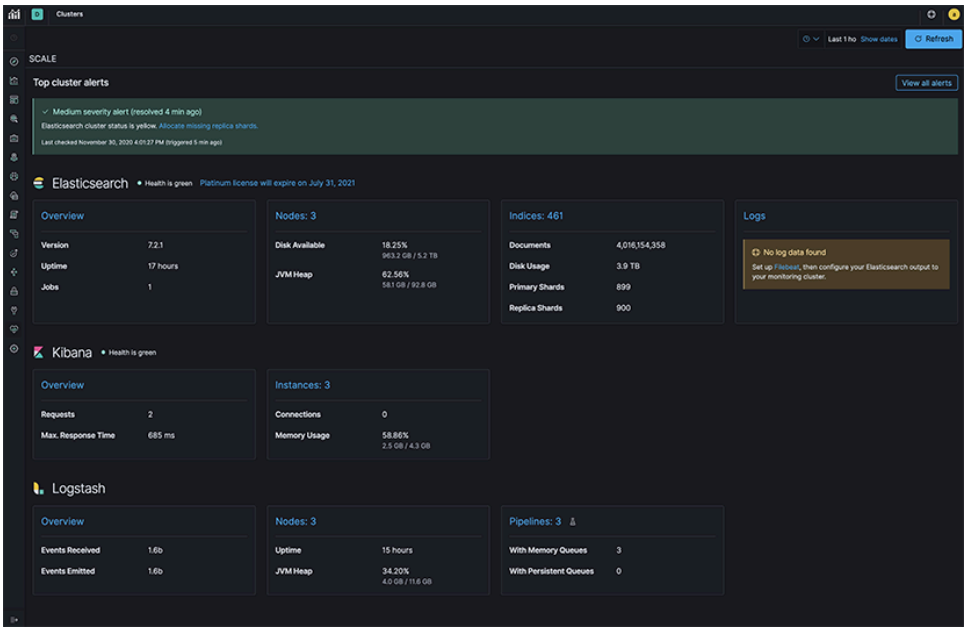
2.6.5 Checking the State of an Analytics Cluster

To check the state of the Analytics Cluster, perform the following steps.

1. Click on the heart-shaped Stack Monitoring icon in the menu bar on the left.

2. Validate that the **Elasticsearch** and **Kibana** state is green. The Graphical User Interface (GUI) should display **Health is green**.

Figure 2-4: Health Monitoring



3. Next, navigate to the CLI of the Analytics Node and run the following command.

```
analytics-2# show cluster
Cluster Name           : SCALE
Cluster Description    : Analytics in Rack 314
Cluster Virtual IP     : 10.106.1.60
Redundancy Status      : redundant
Last Role Change Time  : 2020-11-29 23:25:50.128000 PST
Failover Reason        : Changed connection state: cluster configuration
 changed
Cluster Uptime         : 2 weeks, 3 days
# IP           @ State   Uptime
-|-----------|-|-------|--------------------|
1 10.106.1.57   standby 16 hours, 24 minutes
2 10.106.1.58 * active  16 hours, 28 minutes
3 10.106.1.59   standby 16 hours, 23 minutes
analytics-2#
```

2.7 Accessing and Configuring Arista Analytics

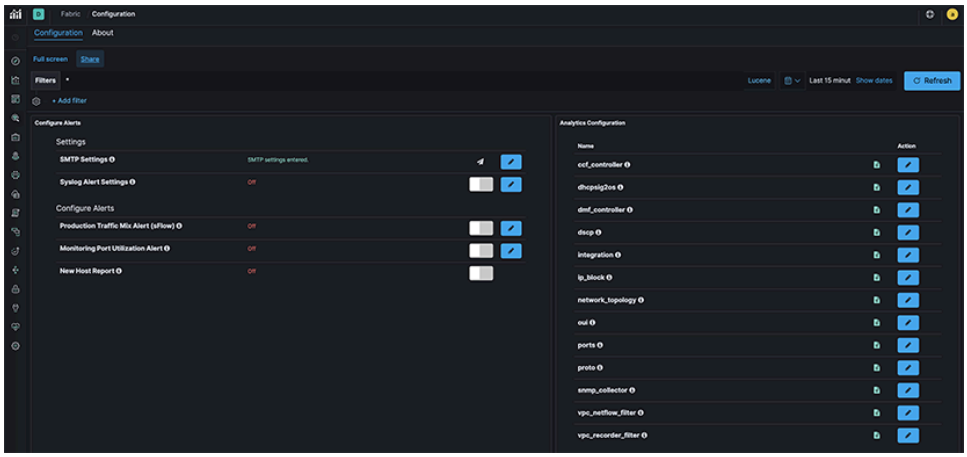
To access the Analytics GUI, point the browser to the IP address assigned to the Analytics server during first boot configuration as following:

```
http://<Analytics node IP address or domain name or Virtual IP in case of
 Analytics cluster>
```

2.7.1 Using the System Tab for Analytics Configuration

When you click the **System** > **Configuration** tab at the top of the Analytics window, the system displays the following page.

Figure 2-5: System > Configuration



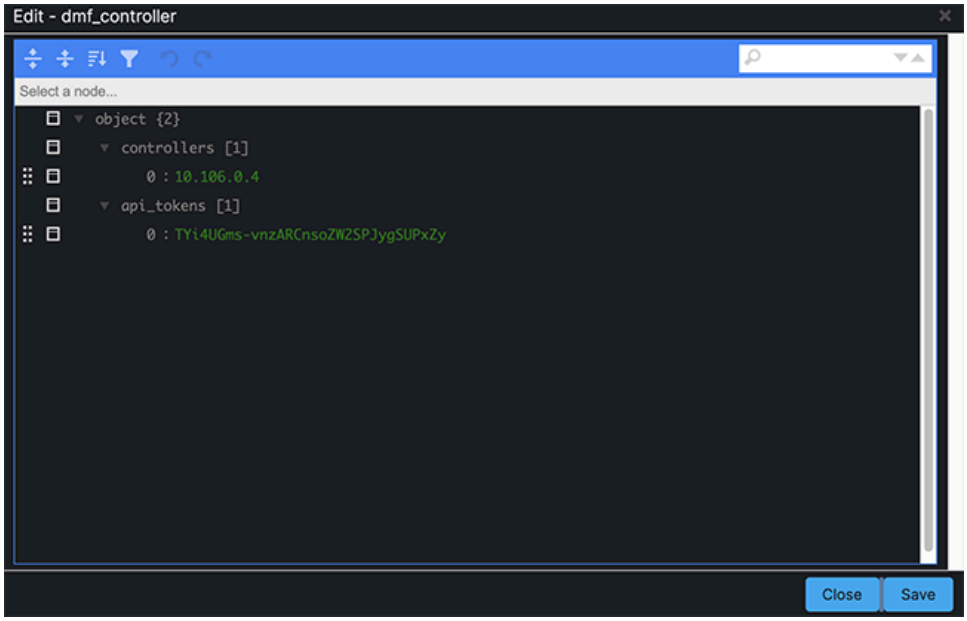
This page lets you configure the settings for sending alerts to an SMTP server, set the alert thresholds, and edit the mapping files used in the different dashboards.

2.7.2 Linking to a DMF Controller

To identify a specific DMF Controller, which is used for the Controller link in the lower left corner of the Analytics page, click the **Edit** control on the **Analytics Configuration** > **dmf_controller** option.

The system displays the following dialog.

Figure 2-6: Link Analytics Node to a DMF Controller

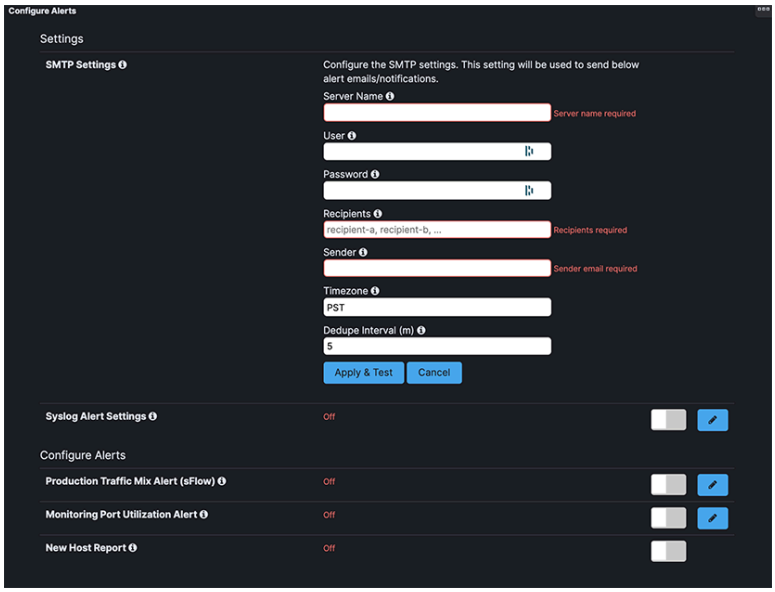


Enter the IP address of the DMF Controller and click **Save**.

2.7.3 Configuring SMTP Settings

Click the **Edit** icon to configure the settings for sending alerts to an SMTP server. The system displays the following page.

Figure 2-7: SMTP Settings



Enter the details for the SMTP server and other required information and click **Apply & Test**.

Note: Once enabled, the Server Name field cannot be left blank, even if you later decide not to use SMTP. You can enter any text string in the field to remove the SMTP server connection information.

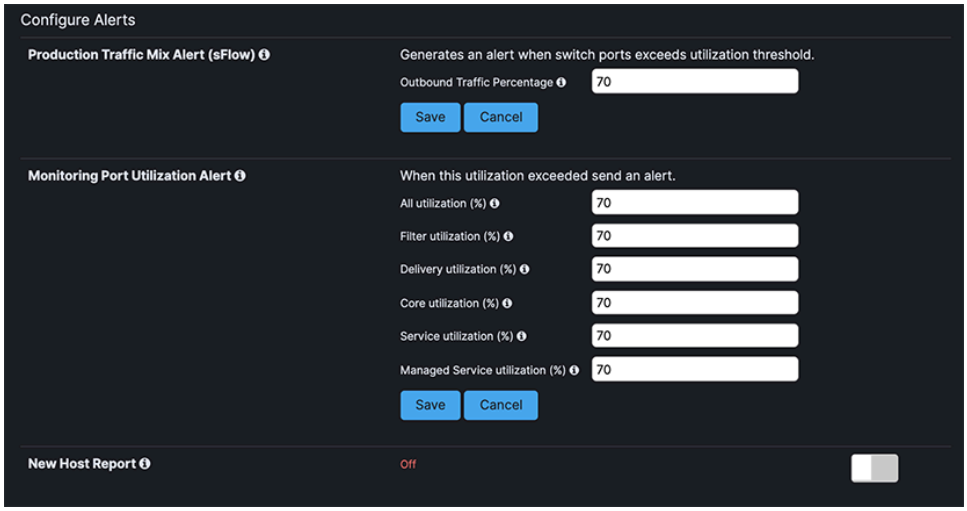
2.7.4 Configuring Alert Thresholds and Enabling Alerts

You can enable the following alerts.

- Production Traffic Mix
- Monitoring Port Utilization Report
- New Host Report

When you click the **Edit** control for the Production Traffic Mix option, the system displays the following page.

Figure 2-8: Alert Configuration



To make changes to the threshold, edit the fields provided and click **Save**. To enable the alert, move the slider to the left. When you click the **Edit** control for the Monitoring Port Utilization Report option, the system displays the following page.

To make changes to the threshold, edit the fields provided and click **Save**. To enable the alert move the slider to the left. To enable the New Host Report option, move the slider to the left.

2.7.5 Sending Analytics SMTP Alerts to a Syslog Server

To set up the Analytics Node to receive NetFlow messages from the DMF Service Node or another NetFlow generator, complete the following steps.

1. SSH to the Analytics Node to access the CLI prompt for Analytics Node configuration.
2. Enter **Config-Local** Mode on the Analytics Node CLI.

```
analytics-1> enable
analytics-1# config
analytics-1(config)# local-node
analytics-1(config-local)#
```

3. Assign an IP address to the collector interface, which should be reachable from the DMF Service Node or other NetFlow generator.

```
analytics-1(config-local)# interface collector
analytics-1(config-local-if)# ipv4
analytics-1(config-local-if-ip)# ip <collector-ip-address>
```

2.7.6 Configuring Collector Interface

Note: This feature is currently supported only on the standalone Analytics Node and is NOT supported in the Analytics Cluster.

Configure collector interface on Analytics to receive NetFlow from a service node or third-party devices by entering the following commands:

```
analytics-1(config)# local node
analytics-1(config-local)# interface collector
analytics-1(config-local-if)# ipv4
analytics-1(config-local-if-ipv4)# ip 219.1.1.10/24
analytics-1(config-local-if-ipv4)#
```

In the Arista Analytics Node, there are two 10G interfaces in bond (***bond3***) acting as a collector interface.

Note: DNS, DHCP, ARP, sFlow, and ICMP traffic coming from Analytics node management should not have the source IP address on the same subnet as the collector interface. Any traffic of those kinds with a source IP address in the same subnet as the collector interface will be dropped.

2.8 Configuring Advanced Features

This section describes the following Advanced Analytics features.

- Machine Learning
- Using Watch for Alerting
- Application Dependency Mapping
- Using RBAC with Arista Analytics
- Time-based User Lockout
- Elasticsearch RBAC examples

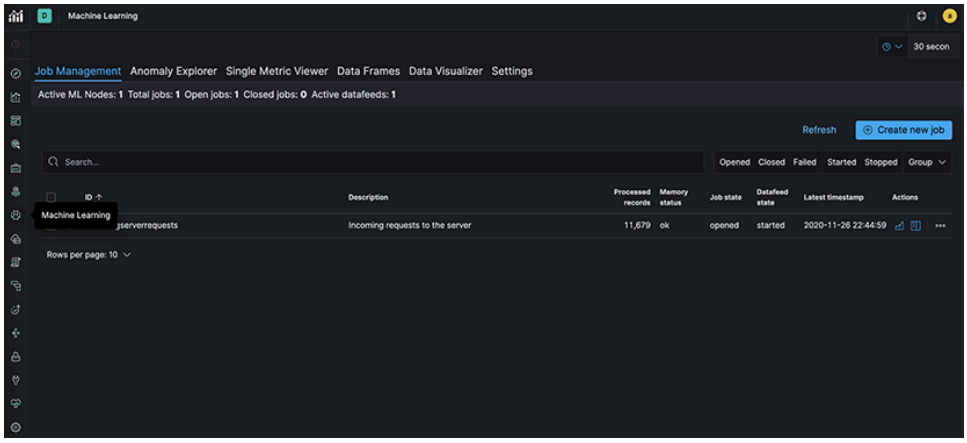
2.8.1 Machine Learning

Note: X-Pack machine learning uses pop-ups, so disable any pop-up blockers, which may create an exception for a Kibana URL.

X-Pack machine learning lets you specify activity that can be monitored over time so that changes from historical norms are flagged as discrepancies, which may indicate unauthorized network usage. For details about this feature, see the Kibana Guide: Machine learning.

To configure this feature, click the **Machine Learning** control in the left pane of the Kibana interface.

Figure 2-9: Machine Learning



The Machine Learning page provides the following tabs:

- **Job Management**: Create and manage jobs and associated data feeds.
- **Anomaly Explorer**: Display the results of machine learning jobs.
- **Single Metric Viewer**: Display the results of machine learning jobs.
- **Settings**: Add scheduled events to calendars and associate these calendars with your jobs.

2.8.2 Using Watch for Alerting

Elasticsearch alerting is a set of administrative features that enable you to watch for changes or anomalies in your data and perform the necessary actions in response. The Elasticsearch watch feature lets you generate an alert when specific network activity occurs. For details about configuring an advanced watch, refer to the Elasticsearch Reference: Alerting.

Elasticsearch provides an API for creating, managing and testing watches. A watch describes a single alert and can contain multiple notification actions.

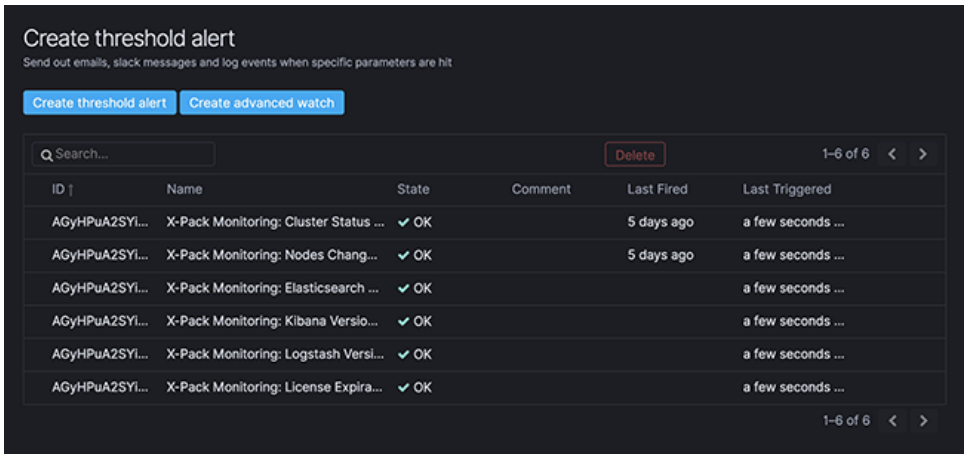
A watch is constructed from four simple building blocks:

- **Schedule**: A schedule for running a query and checking the condition.
- **Query**: The query to run as input to the condition. Watches support the full Elasticsearch query language, including aggregations.
- **Condition**: A condition that determines whether or not to execute the actions. You can use simple conditions (always true), or use scripting for more sophisticated scenarios.
- **Actions**: One or more actions, such as sending email, pushing data to 3rd party systems through a webhook, or indexing the results of the query.

A full history of all watches is maintained in an Elasticsearch index. This history keeps track of each time a watch is triggered and records the results from the query, whether the condition was met, and what actions were taken.

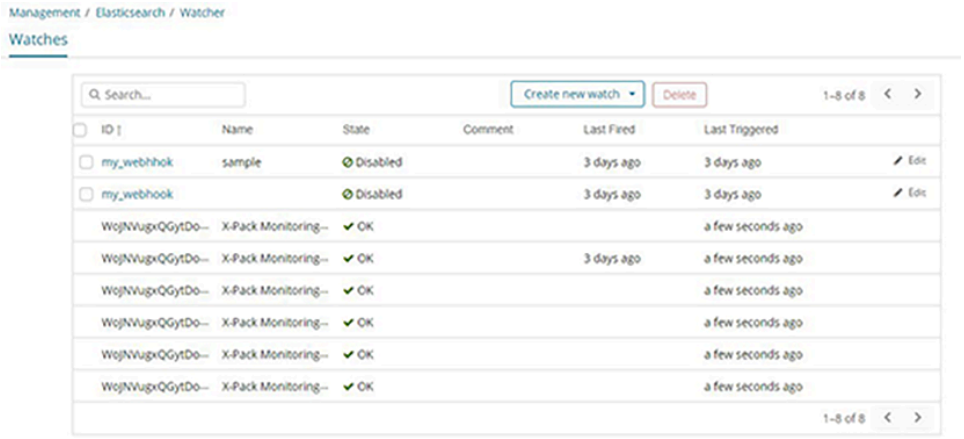
To configure an Alert, click the **Management** control in the left pane of the Kibana interface, and click **Watcher** to define a new instance.

Figure 2-10: Using a Watcher for Alerting



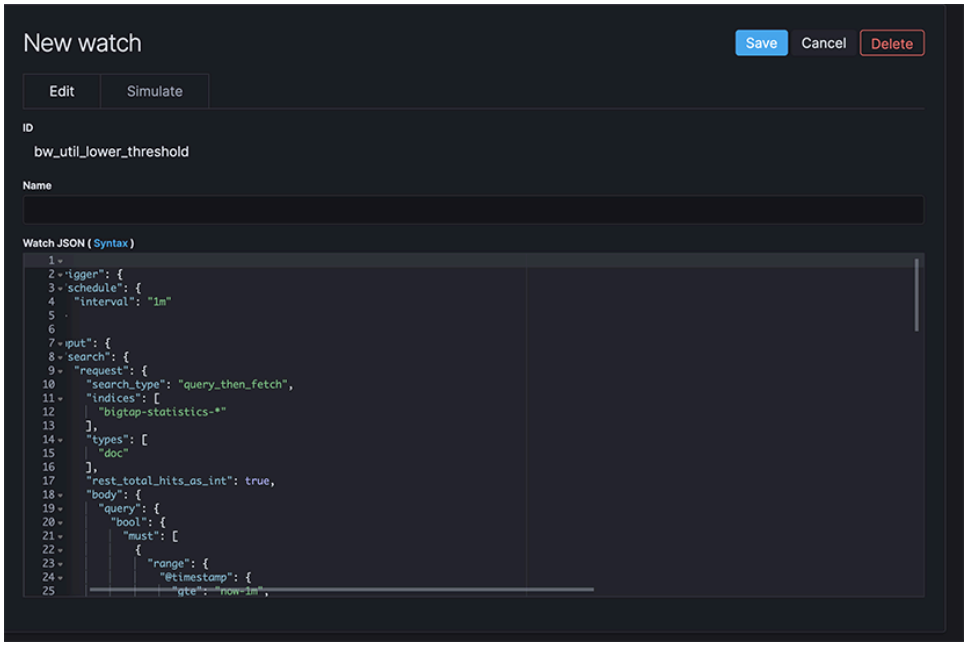
The following figure defines a new watch:

Figure 2-11: Defining a New Watch



Click **Create new watch** and select **Advanced Watch** from the menu that appears. This option lets you define a custom alert.

Figure 2-12: Example of Advanced Watch



REST script in JSON format

Compose a REST script in JSON format that includes four sections:

- **Trigger** Schedules when the watch runs. This can be an interval, which causes the watcher to run after the specified time elapses (for example, every ***10*** seconds).
- **Input** Identifies the information you want to evaluate. This can be search criteria that retrieves the required input.
- **Condition** Identifies the activity or other condition that determines if the alert should be sent.
- **Action** Identifies the text of the alert and the webhook where the alert message will be sent.

The following is an example JSON script that sends an alert whenever more than ***10*** packets containing the string ***"gte"*** are received within a ***5-second*** interval.

```
{
  "trigger": {
    "schedule": {
      "interval": "5s"
    },
    "input": {
      "search": {
        "request": {
          "search_type": "query_then_fetch",
          "indices": [
            "flow-icmp*"
          ],
          "types": [],
          "body": {
            "query": {
              "match_all": {}
            }
          }
        }
```

```
      }
    },
    "condition": {
      "compare": {
        "ctx.payload.hits.total": {
          "gte": 10
        }
      }
    },
    "actions": {
      "my_webhook": {
        "webhook": {
          "scheme": "https",
          "host": "hooks.slack.com",
          "port": 443,
          "method": "post",
          "path": "/services/T029CQ2GE/B5NBNKMGR/uZjyLgVUqrQLvGl60yM9ANUP",
          "params": {},
          "headers": {
            "Content-Type": "application/json"
          },
          "body": "{\"channel\": \"#office_bmf_test\", \"username\":
 \"webhookbot\", \"text\": \"icmp
burst detected over the set limit \", \"icon_emoji\": \":exclamation:\"}"
        }
      }
    }
  }
}
```

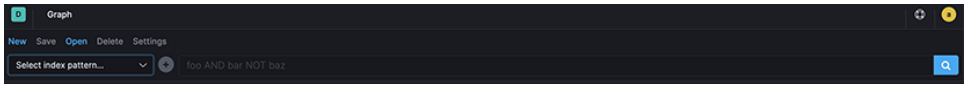
For information about configuring the SLACK webhook, refer to the following Slack documentation.

2.8.3 Application Dependency Mapping

This feature helps you identify how items in an Elasticsearch index are related, a process known as Application Dependency Mapping (ADM). You can explore the connections between indexed terms and see which connections are the most meaningful. For example, this feature lets you map the relationships between the Destination IP (DIP) and Source IP (SIP) for a specific application. For details about this feature, refer to the Kibana documentation.

Arista Analytics provides a graph exploration API and an interactive graph visualization tool that work with existing Elasticsearch indices. To configure this feature, click the **Graph** control in the left pane of the Kibana interface.

Figure 2-13: Application Dependency Mapping



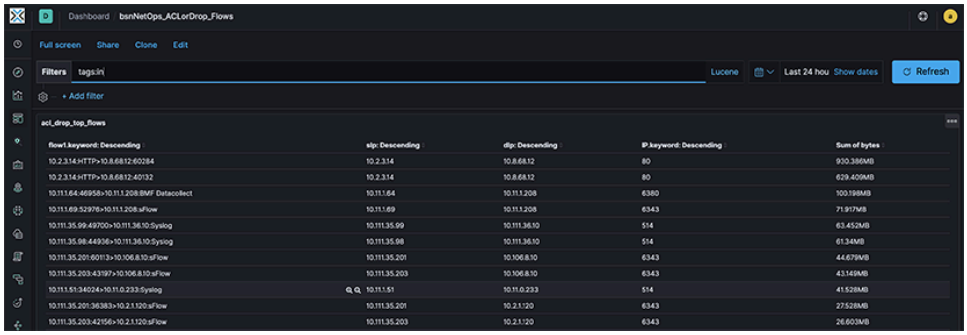
A graph is a network of related terms in the index. The terms you want to include in the graph are called vertices. The relationship between any two vertices is a connection. This feature lets you answer questions such as the following.

- Can I build a map to show different client machines accessing services identified by a Layer 4 port?
- Can I build a map to view which DNS servers are accessed by all the clients?
- Can I build a map to show how different servers interact with each other?

Advanced options let you control how your data is sampled and summarized. You can also set timeouts to prevent graph queries from adversely affecting the cluster.

Analytics also provides a dashboard that has a table with all the IPs and port numbers that are communicating with each other. To view the table, click **Dashboard** on the left panel, search for ***bsnNetOps_ACLorDrop_Flows***, and click on the link.

Figure 2-14: Active IPs and Port Numbers



2.8.4 Using RBAC with Arista Analytics

Arista Analytics supports full Role-Based Access Control (RBAC) for the web-based User Interface (UI) and CLI. Arista Analytics supports two types of users:

- **admin**: Admin user accounts have full read and write access to the CLI as well as to the Kibana UI.
- **non-admin**: Non-admin users typically have read only access. They can be defined only by an admin user.

To create and enable new user accounts, complete the following steps.

1. Create group and user in the Analytics CLI.

```
analytics-1(config)# group new-non-admin-group
analytics-1(config-group)#
analytics-1(config)# user new-non-admin-user
analytics-1(config-user)#
```

2. Verify successful creation of user.

```
analytics-1(config-group)# show user
# User name           Full name     Groups
-|------------------|-------------|-------------------|
1 admin Default       admin         admin
2 new-non-admin-user                new-non-admin-group
```

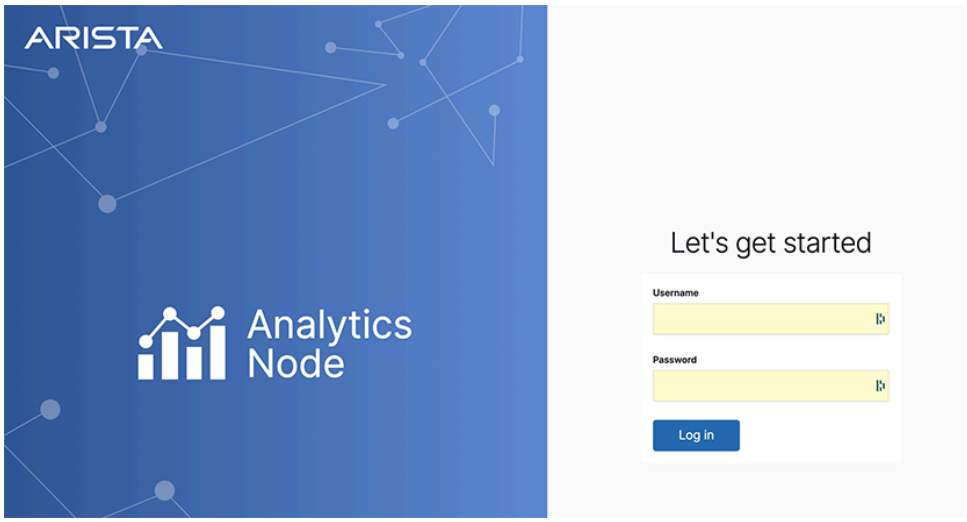
3. Verify successful creation of group.

```
analytics-1(config-group)# show group
```

4. Create role and privilege in the Kibana UI that matches the group created in Step 1. To set roles and privileges in the Kibana UI refer to the Elastic documentation

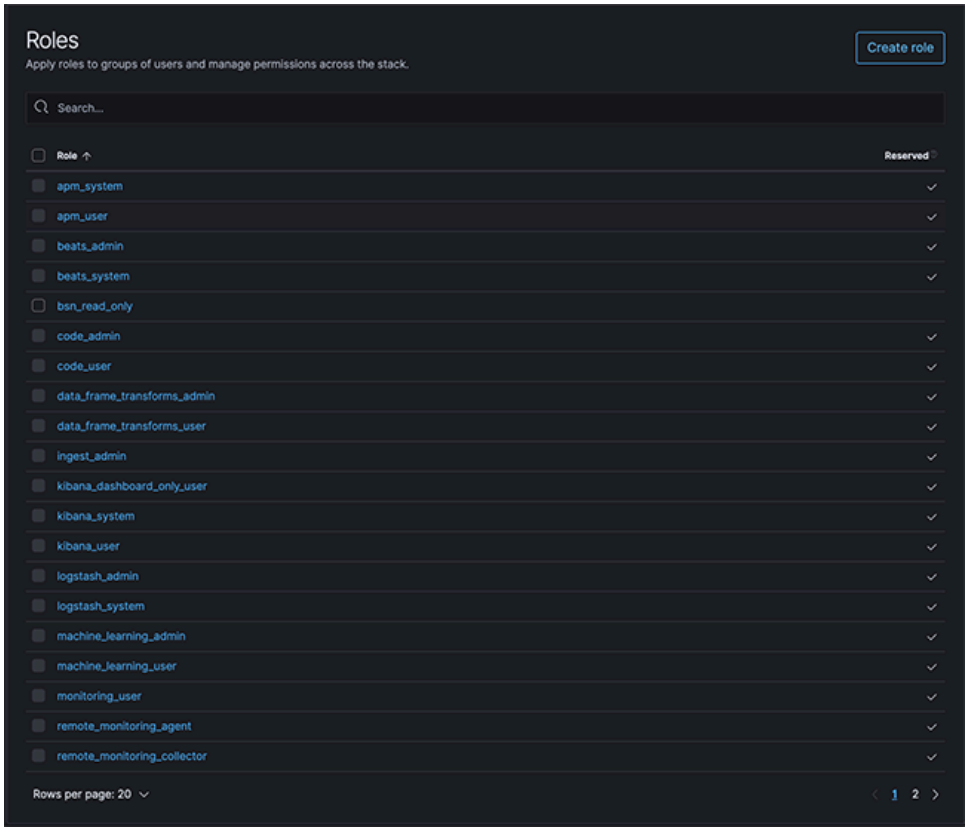
a. Log in as admin into Kibana.

Figure 2-15: Kibana UI Log In



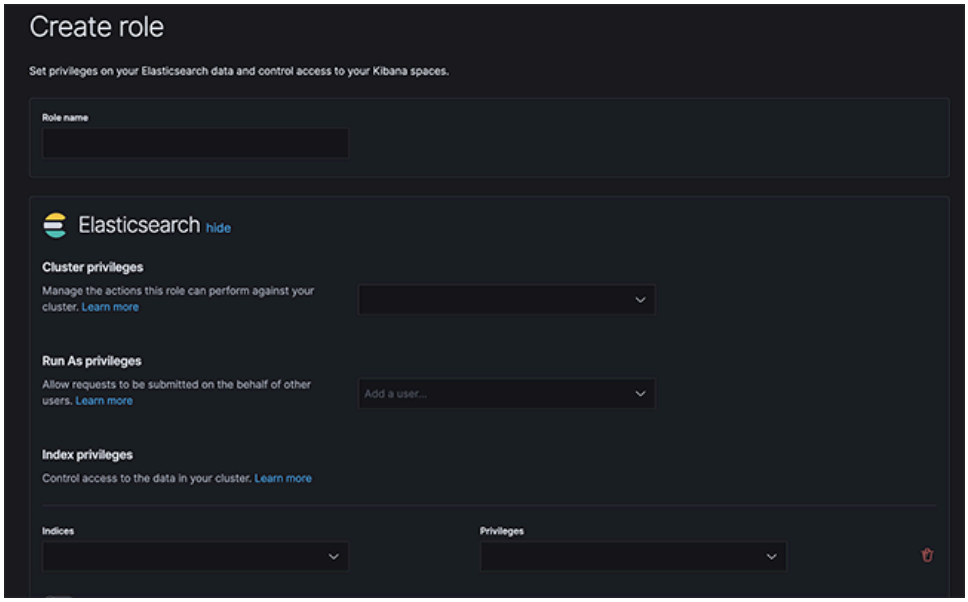
b. Go to **Management >Roles**.

Figure 2-16: Role Management



c. Click **Create Role** and populate the respective fields as shown for read-only access.

Figure 2-17: Verifying New Group



d. Add or remove indices as needed under **Index Privileges > Indices**.

e. Click **Save** and verify that the created group appears in the list shown.

Figure 2-18: Kibana Management > Roles

5. Log in as usual to Kibana, using the newly created user account.

   Click the **logout** button as you normally do for users created in Kibana.

   Log in using an incognito tab and log off by closing all tabs in incognito mode.

Note: For configuring TACACS+ and Radius refer to the ***DMF User guide*** .

2.8.5 Time-based User Lockout

Starting in the ***DMF 8.0*** release, DANZ Monitoring Fabric supports time-based user lockout functionality. Users will be locked out of login for ***t2*** time when attempting with ***n*** incorrect passwords within ***t1*** time.

Locked out users have to be cleared of lockout or they have to wait for the lockout period to expire before attempting login with the correct password. The feature is disabled by default.

To enable, use the following command:

```
Controller-1(config)# aaa authentication policy lockout failure <number of
 failed attempts> window <within t1 time>duration <lockout for t2 time>
```

- Value range for **failure** can be from 1 to 255.
- Value range for **window** and **duration** can be from ***1*** to ***4294967295*** seconds (2^32-1).

The following example locks any user out for ***15*** minutes when attempting three incorrect logins within ***3*** minutes.

```
Controller-1(config)# aaa authentication policy lockout failure 3 window 180
 duration 900
```

Note: This feature affects only remote logins such as SSH/GUI/REST API using username and password. Console-based login, password-less authentications such as SSH keys, Single Sign-on, and access-token logins are not affected. Locked-out users can still access the Controller via console or password-less authentication.

Note: The feature is node-specific with respect to the functionality. For example, if ***user1*** is locked out accessing the active Controller in the cluster, they would still be able to log in to a standby Controller with the correct password, and vice-versa. Lockout user information is also not persistent across Controller reboot or failover.

To view if a user is locked out, admin-group users can issue the following command: **`show aaa authentication lockout`**

```
Controller-1# show aaa authentication lockout
User name Host Failed   Logins        Lockout Date                  Lockout Expiration
---------|-------------|-------------|-----------------------------|-----------------------------|
admin     10.240.88.193 1             2020-09-08 16:07:36.283000 PDT 2156-10-15 22:35:51.283000 PDT
```

To clear the lockout for a user, admin-group users can issue the following command: **`clear aaa authentication lockout user <username>`**

To clear all the locked out users, admin-group users can issue the following command:

`clear aaa authentication lockout`

The following example shows how to clear the **“admin”** user who got locked out.

```
Controller-1# clear aaa authentication lockout user admin
Controller-1# show aaa authentication lockout
None.
```

The **“recovery”** user will also be locked out if attempting with incorrect passwords. To check if the user is locked out, use **pam_tally2** tool:

```
admin@Controller-1:~$ sudo pam_tally2 -u recovery
Login Failures Latest failure From
recovery 9 09/08/20 16:16:04 10.95.66.44
```

To reset the lockout for the user, use the following command:

```
admin@Controller-1:~$ sudo pam_tally2 --reset --user recovery
Login     Failures Latest failure         From
recovery     9     09/08/20 16:16:04     10.95.66.44
admin@Controller-1:~$ sudo pam_tally2 -u recovery
Login     Failures Latest failure         From
recovery
```

Note: the **window** parameter does not apply to the **“recovery”** user login as the **pam_tally2** tool does not support it.

2.8.6 Elasticsearch RBAC examples

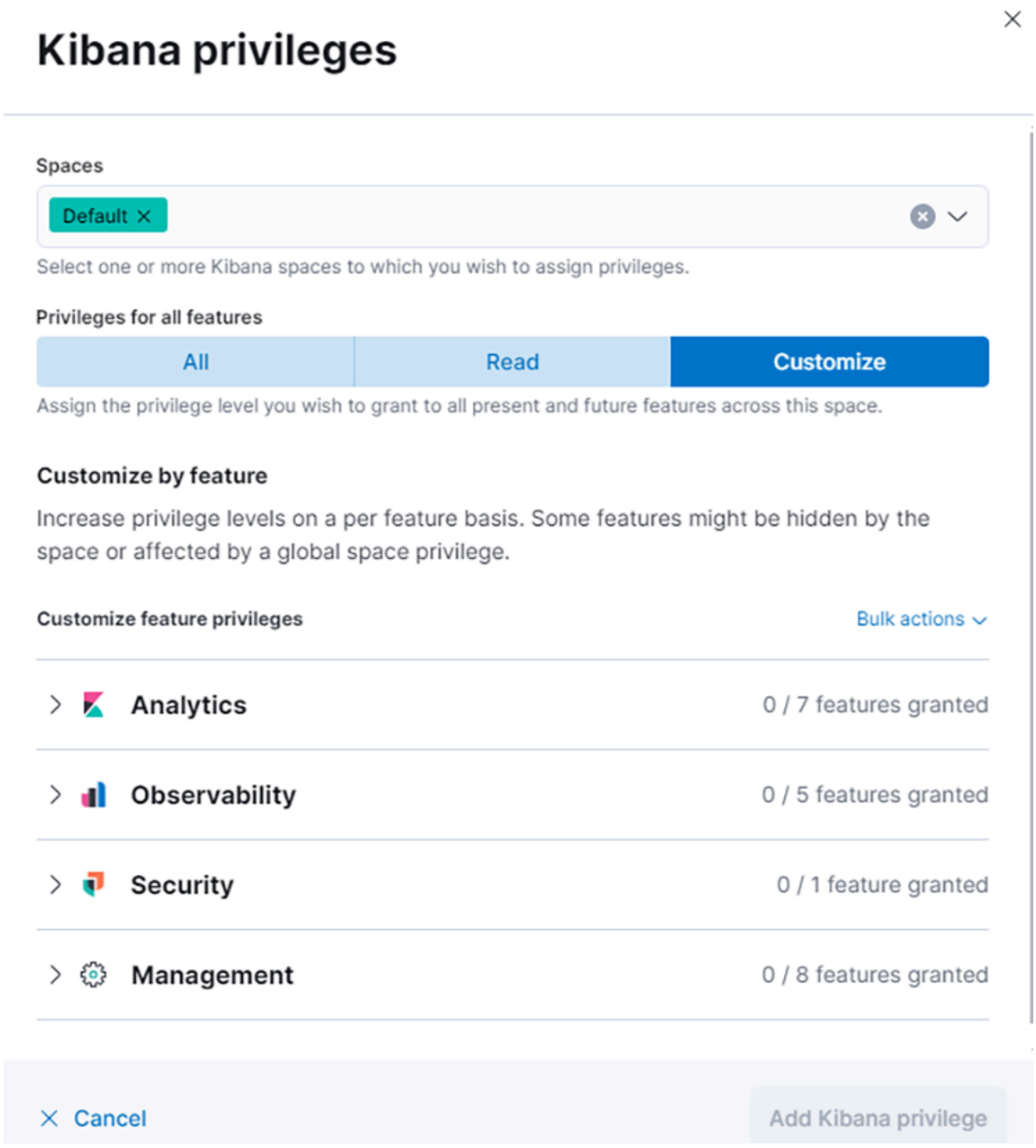
Admin User and Group: The admin user is by default added to the admin group and the superuser role in elasticsearch. No configuration is needed for it.

Read-only Access: By default, the BSN read-only role exists that maps to Floodlight as well.

Dashboard Access Only:

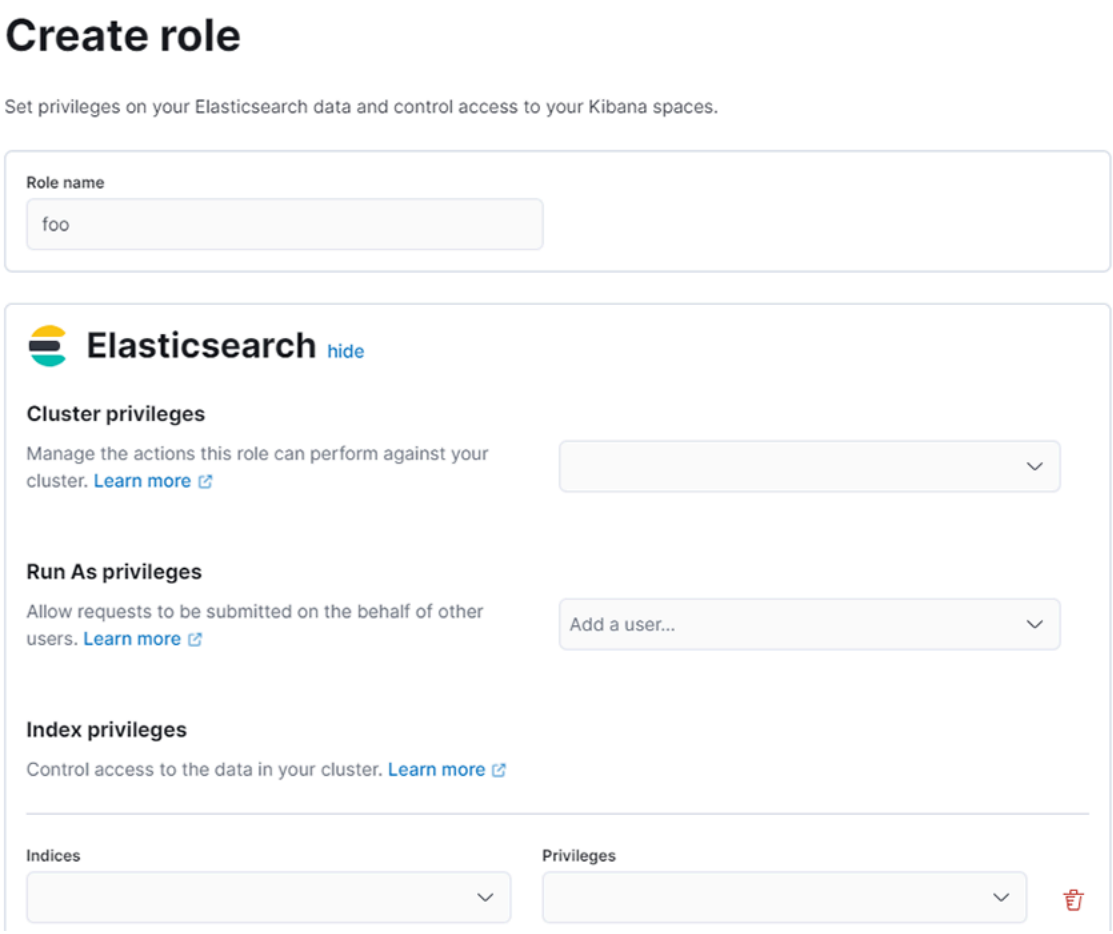
Create the role for dashboard access, by selecting **Stack** > **management** > **Roles** > **Create Role**. Here, configure the indices to ***** and set the privileges under Kibana as shown in the image below.

Figure 2-19: Kibana privileges for Dashboard access only



The following is the example for different privileges for Elasticsearch.

Figure 2-20: Elasticsearch RBAC example



2.9 Integrating Analytics with Infoblox

Infoblox provides DNS and IPAM services, which can be used to integrate with Arista Analytics. To use, associate a range of IP addresses in Infoblox with extensible attributes, then configure Analytics to map these attributes for the associated IP addresses. The attributes assigned in Infoblox then appear in place of the IP addresses in Analytics visualizations.

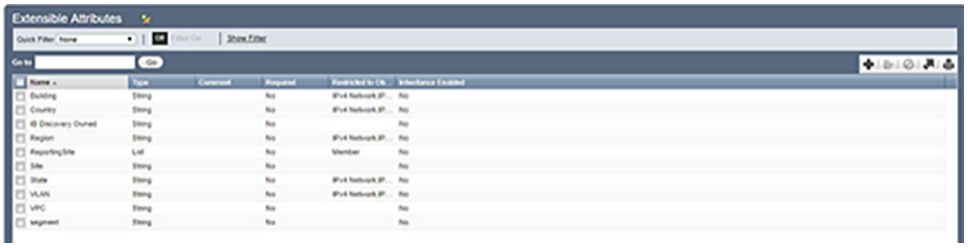
2.9.1 Configuring Infoblox for Integration

To configure Infoblox for integration with Arista analytics, complete the following steps.

1. Log in to Infoblox System Manager.

2. To set the extensible attributes in Infoblox, click the **Administration Extensible Attributes** tab.

Figure 2-21: Extensible Attributes Tab



This tab lets you define the attributes applied to a block of IP addresses. The extensible attributes you define for integrating Infoblox used with Arista Analytics are as follows:

- **EVPC**: Identifies the Enterprise Virtual Private Cloud (EVPC) assigned to a block of IP addresses in Infoblox.
- **segment**: Identifies the specific subnet interface to which an IP address is assigned.

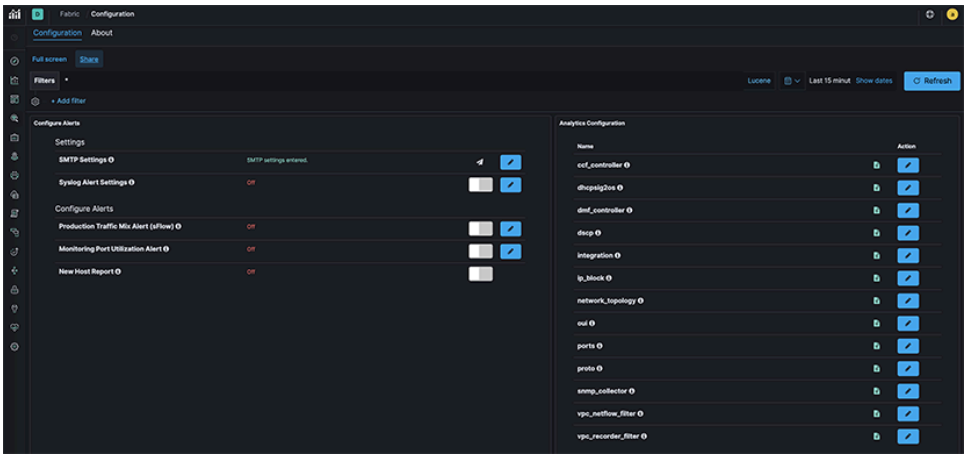
3. To assign an IP address range to the VPC extensible attribute, click **Data Management IPAM**.
4. Save the configuration.

2.9.2 Configuring Arista Analytics

After completing the configuration required to integrate Infoblox with Arista Analytics to recognize the IP address ranges assigned in Infoblox, configure Analytics by completing the following steps.

1. Log in to Arista Analytics.
2. Click **System Analytics Configuration**.

Figure 2-22: DMF Analytics Configuration



Refer the Adding Flow Enhancement via Infoblox IPAM Integration for more integration information.

2.9.3 Adding Flow Enhancement via Infoblox IPAM Integration

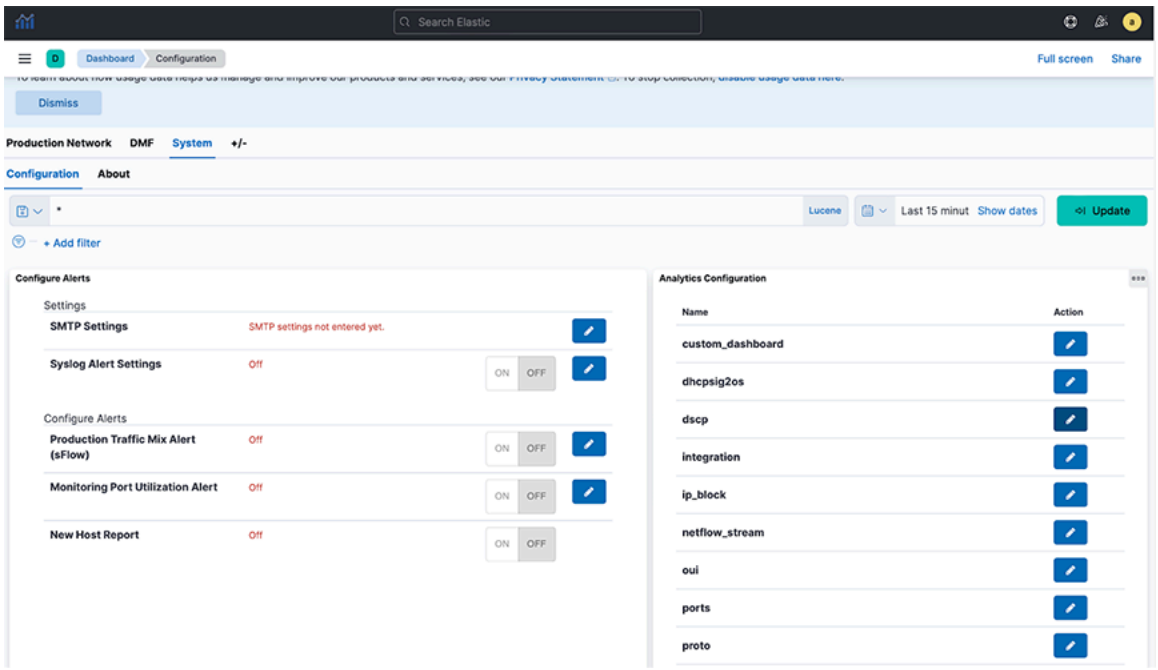
This feature integrates subnets and corresponding extensible attributes from an Infoblox application into Arista Analytics' collection of IP blocks and corresponding list of attributes.

Arista Analytics provides an enhanced display of incoming flow records using these extensible attributes from the Infoblox application.

Configuring the Flow enhancement

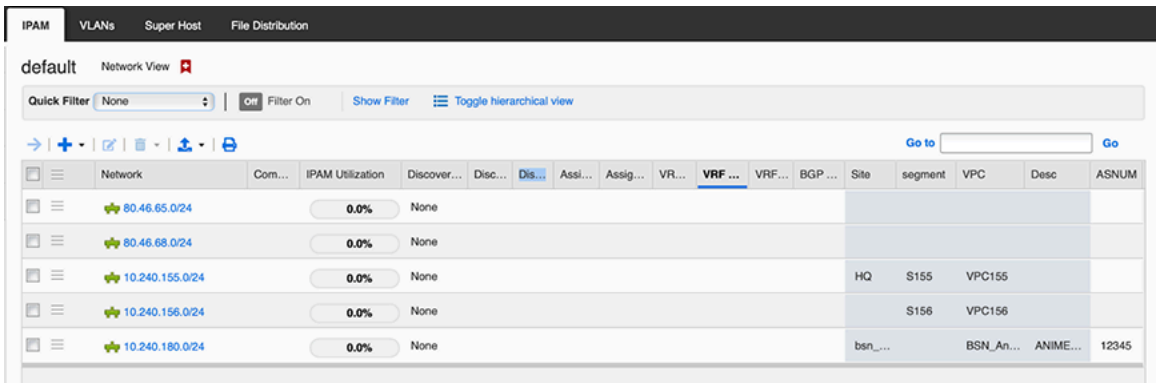
Configure the feature in Kibana by selecting the **System** > **Configuration** tab on the **Fabric** page and opening the **Analytics Configuration** integration panel.

Figure 2-23: Dashboard - Configuration



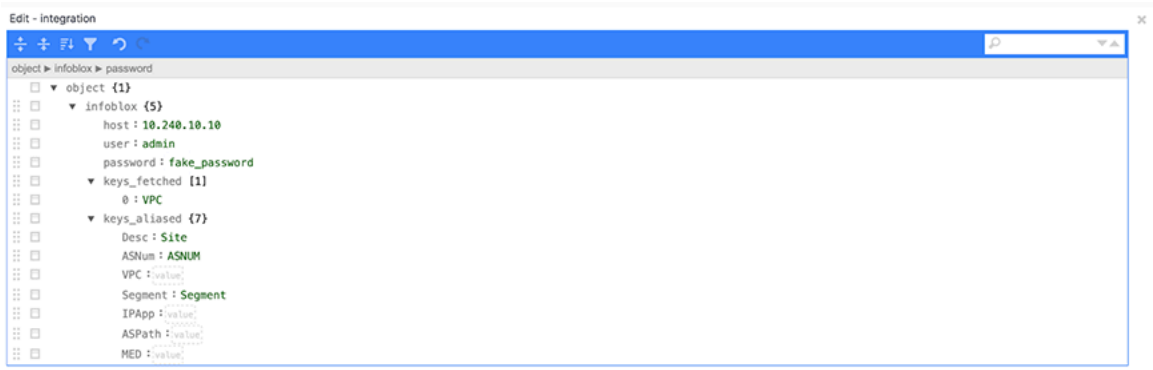
The list of IP blocks and associated external attributes appears in the Infoblox application and under the **Data Management** > **IPAM** tab. The columns shaded in gray represent the **extensible attributes** and their **values**.

Figure 2-24: Data Management > IPAM



Editing IPAM Integration

Figure 2-25: Edit - Integration



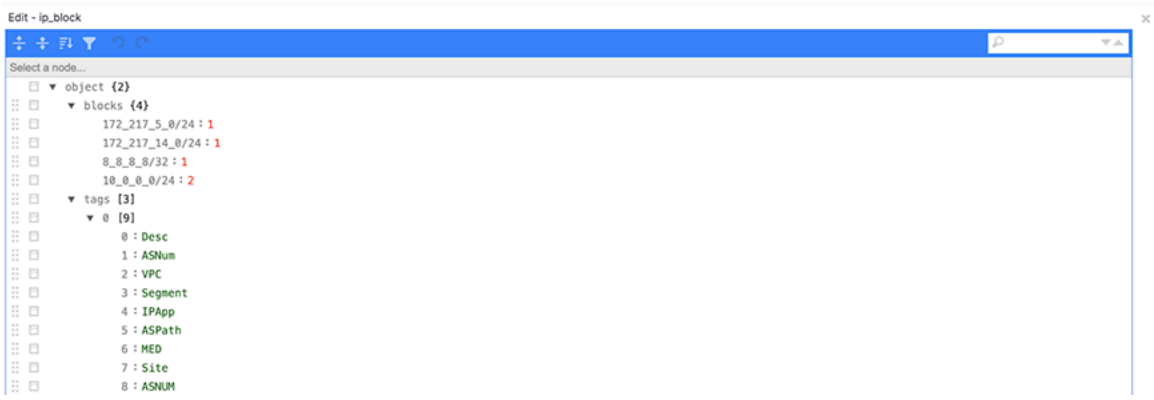
Configure the integration on Arista Analytics using the following example:

- Infoblox:
  - **host**: The IP address or DNS hostname of the Infoblox application.
  - **user**: Username for Infoblox application.
  - **password**: Password for Infoblox application.
  - **keys_fetched**:
    - The list of extensible attributes from the connected Infoblox application to be added to the Analytics Node **ip_block** tags. If an entered **extensible attributes** matches the name of an existing **ip_block** tag, it will not be added.
  - **keys_aliased**:
    - Mapping default Analytics Node **ip_block** tags to **extensible attributes** in the Infoblox application. Add additional mappings from **ip_block** tags to extensible attributes as required. Empty field values are ignored. Each mapping from the **ip_block** tag to the **extensible attributes** indicates:
      - Add the **extensible attributes** to the Analytics Node's **ip_block** tags. If an **extensible attributes** appears in both the **integration** configuration ***keys_fetched*** list and as a value in the ***keys_aliased*** mapping, it will only be added once to the Analytics Node **ip_block** tags list. It will not be added if it is already in the **ip_block** tags.
      - For IP addresses coming from the Infoblox application, the value of the **extensible attributes** should replace the value of the corresponding **ip_block** tag. The **extensible attributes** and the Analytics Node tag become aliases of each other.

For example, in the above example **integration** configuration, ***VPC*** is in ***keys_fetched***, and ***segment*** is in the values of ***keys_aliased***, but both are already in the **ip_block** tags list, so they are not added again,

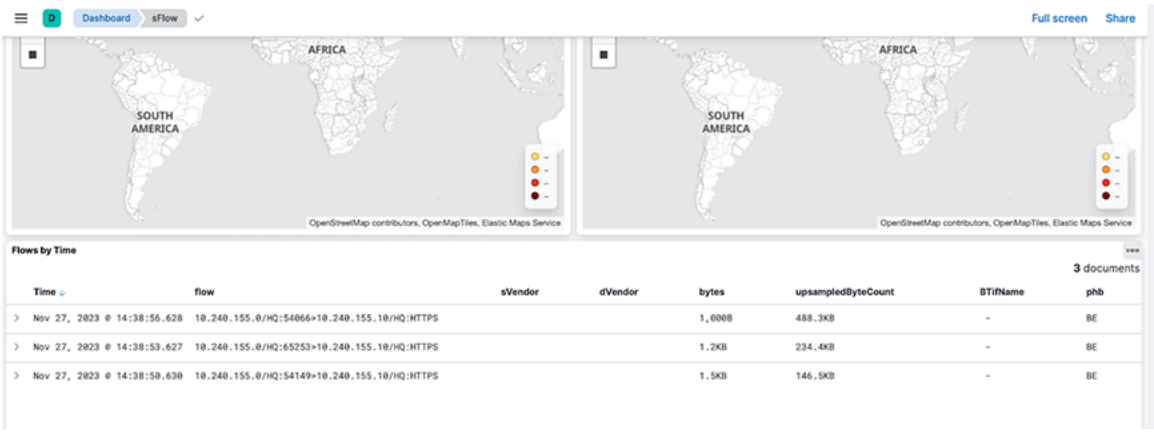
as seen below. However, ***Site*** and ***ASNUM*** are not already in the tags list and are added to the end of the tags list.

Figure 2-26: Edit - ip_block



As a result of these configuration changes, view the following enhancements to the flow records in the **Production Network** > **sFlow** tab and scroll to the **Flows by Time** chart.

Figure 2-27: Dashboard - sFlow



Suppose the sFlow packet source and/or destination IP addresses fall within the IP subnets in the Infoblox IPAM dashboard. In that case, their flow records will be augmented with the extensible attributes from Infoblox as specified in the **integration** configuration.

For example, the source and destination IP address of the ***10.240.155.0/HQ:54149 > 10.240.155.10/HQ/HTTPS*** flow fall within the ***10.240.155.0/24*** subnet in the Infoblox IPAM dashboard.

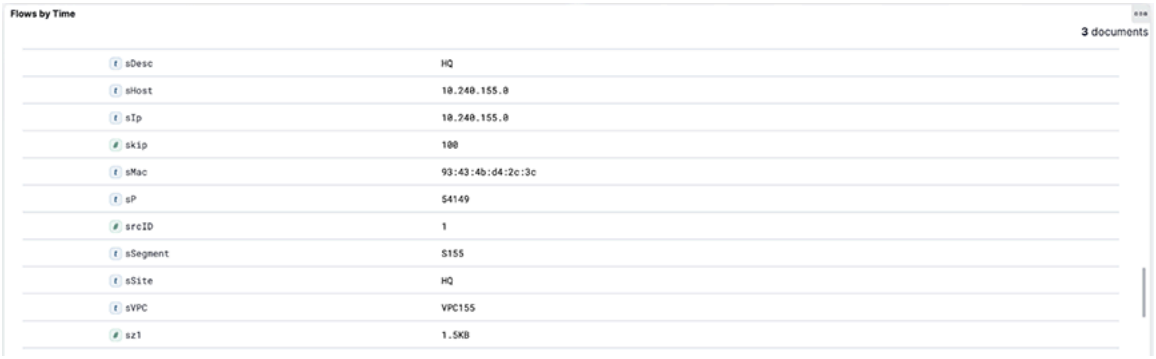
When expanding this flow in the **Flows by Time** chart, since ***VPC*** is in the **integration** ***keys_fetched***, the ***sVPC*** value is ***VPC155***.

Site is in the **integration** ***keys_aliased*** values, and a ***sSite*** value of ***HQ*** appears. Since ***Desc*** is aliased to ***Site*** (an extensible attribute), ***sDesc*** takes on the value of ***Site***. ***Segment*** is in the ***keys_aliased*** values; hence, ***sSegment*** with a value ***S155*** appears.

Observe similar attributes for the destination IP address in the flow record. All these values come from the Infoblox IPAM dashboard shown above. ***ASNUM*** does not appear as a field in the flow record below despite

being in the **integration *keys_aliased*** values because it is not configured or associated as an extensible attribute to the subnets in the Infoblox IPAM dashboard.

Figure 2-28: Flow by Time



Troubleshooting

If the flow records that you expect to be augmented with InfoBlox extensible attributes are missing these attributes, please verify that the Infoblox credentials you provided in the integration configuration are correct. After confirming the credentials and the relevant flow records are still missing the Infoblox extensible attributes, please generate a support bundle and contact Arista Networks TAC.

Limitation

Known Issue:

- When removing a tag in the middle of the **ip_block** tags list and saving the configuration, the relevant flow records may have incorrect values in their attributes during the minute following this change. After this brief period, the flow records will have the correct attributes and corresponding values.

2.10 Configuring SMTP Server to Send Email Alerts via Watcher

You can configure the email action in Watcher to send email notifications. To send an email, you must configure at least one email account in the Analytics Node. To do so, access the Analytics node via the CLI and complete the following steps.

Note: All of the following steps need to be executed on each node of the Analytics Node cluster.

1. At the Analytics Node command prompt, enter:

```
debug bash
```

2. Access the config file.

```
vi /opt/bigswitch/docker-compose.yml
```

3. Access the environment section under the Elasticsearch component.

```
version: '2'
services:
elasticsearch:
image: elasticsearch
logging:
driver: none
container_name: elasticsearch
#cpu_shares: 55
```

```
ports:
- "0.0.0.0:9201:9201"
- "0.0.0.0:9300:9300"
volumes:
- /var/lib/analytics/data:/usr/share/elasticsearch/data
- /var/log/analytics/es:/usr/share/elasticsearch/logs
- /etc/localtime:/etc/localtime:ro
- /opt/bigswitch/conf/log4j2.properties:/usr/share/elasticsearch/config/
log4j2.properties
- /var/lib/analytics/data/private.key:/usr/share/elasticsearch/config/
private.key
- /var/lib/analytics/data/cert.pem:/usr/share/elasticsearch/config/cert.pem
- /opt/bigswitch/snapshot:/usr/share/elasticsearch/snapshot
environment:
- cluster.name=${ES_CLUSTER_NAME}
- http.port=9201
```

4. Append the following lines to the environment section.

```
- xpack.notification.email.default_account=<account name>
- xpack.notification.email.account.<account name>.profile.from=<from email
 id>
- xpack.notification.email.account.<account name>.smtp.auth=true
- xpack.notification.email.account.<account name>.smtp.starttls.enable=true
- xpack.notification.email.account.<account name>.smtp.host=<SMTP server
 host name>
- xpack.notification.email.account.<account name>.smtp.port=587
- xpack.notification.email.account.<account name>.smtp.user=<SMTP user email
 id>
```

5. Use the **`keystore`** command to store the account SMTP password. Access the Elasticsearch container, run the following command, enter the password, then commit changes to the container.

```
sudo docker exec -it elasticsearch bash
bin/elasticsearch-keystore add xpack.notification.email.account.arista
.smtp.secure_password
exit
sudo docker commit elasticsearch elasticsearch
```

6. Configure the watcher action to send notifications by email.

```
"actions": {
  "send_email": {
    "email": {
      "profile": "gmail",
      "from": "<from email id>",
      "to": [
        "<To email id>"
      ],
      "subject": "<subject>",
      "body": {
        "text": "<email body>"
      }
    }
  }
```

Refer to https://www.elastic.co/guide/en/elasticsearch/reference/current/actions-email.html for more details on the Watcher email action.

Appendix A

Deployment Check List

This appendix describes how to create a bootable USB drive for installing Arista Analytics.

A.1 Analytics Deployment Checklist

Ensure that the deployment of the Arista Analytics Node is correct by verifying the following steps.

Note: All HTTP commands below should be run in the Kibana **dev_tools** console.

A.2 Checklist

1. Prior to first-boot ensure the management interface is wired and has connectivity.
2. Check if DNS configuration is correct.
3. List indices using **POST _cat/indices** to detect issues with respect to time in the flow generator (SN, Switch etc). For example, you may see indices from days in the future or past for egregious time differences.
4. Check if sFlow comes on ***port 6343***, NetFlow v5 on ***2055***, NetFlow v10 on ***4739***, using tcpdump -i ***bond0*** port ***6343*** on ***bond0*** or ***bond3*** as appropriate.
5. Check if packet comes without VLAN tag (in DMF policy).
6. Check if IPAM is enabled (all switches must have IP addresses in the Controller subnet).
7. Ping check from AN to Controller and vice versa. Ping check from AN to SNMP target.
8. Check if all containers are up on all nodes notably kibana, elasticsearch, btan, datacollect: **`docker ps -a`**
9. Check if the UI successfully loads on all nodes via physical IP of nodes.
10. Check the status of ES using **POST _cluster/health**.
11. Check if all cluster members are present in the ES and Floodlight cluster using the CLI: **`show cluster`** or the API: **`ES REST api: GET _cat/nodes`**

Appendix B

Creating A USB Drive

This appendix describes how to create a bootable USB drive for installing Analytics.

B.1 Creating the USB Boot Drive

To install the Analytics software from a USB drive, you must copy the ISO image to the USB drive to make it a bootable disk. This can be done in Windows, MacOS, or Linux.

B.1.1 Creating the USB Boot Drive with MacOS X

To create a bootable USB drive on MacOS X, complete the following steps.

1. Insert the USB drive into a USB port on the Macintosh.
   This automatically mounts the drive, but it must be unmounted to create a bootable disk.
2. Open a Mac OS terminal window.
3. Enter the **`diskutil`** command to list all the mounted disks, as in the following example:

```
diskutil list
```

MacOS Disk Utility

You can also use the MacOS Disk Utility GUI application (applications/utilities) to identify the mounted disks and unmount the USB drive.

1. Identify the **`/dev/disk<x>`** label for the inserted USB drive.
2. Unmount the USB drive (this is different than ejecting), using the following command.

```
diskutil unmountdisk /dev/disk<x>
```

   Replace **<x>** with the unique numeric identifier assigned by the system.
3. Enter the **`sudo dd`** command in the terminal window to make the USB drive bootable.

```
sudo dd if=<path to iso image> of=/dev/rdisk<x> bs=1024m
```

Note: Using the **`dd`** command with the wrong disk name can erase the installed OS or other vital information.

Use this command to copy the appliance ISO image to the USB drive. Using /dev/rdisk makes the copying faster (rdisk stands for a raw disk).

Replace **<x>** with the drive identifier for the USB drive and replace ***<path to iso image>*** with the file name and path to the location where you downloaded the ISO image.

the following command copies the file **`bmf-service-node.dmg`** to disk2:

```
sudo dd if= bmf-service-node.iso of=/dev/rdisk2 bs=1024m
```

It can take up to ten minutes to copy the image to the USB drive.

To monitor the progress of the write operation, enter the following command in a separate terminal window.

```
$ while sudo killall -INFO dd; do sleep 5; done
```

4. Eject the drive by entering the following command:

```
disk util eject
```

Alternatively, select Eject from the File menu.

B.1.2 Building the USB Boot Image with Linux

To create a bootable USB drive on using Linux, complete the following steps.

1. Insert the USB drive into a USB port on the Linux workstation.
2. In a Linux terminal window, enter the following command to identify the USB drive.

```
disk -lu
```

On Linux, the USB drive is typically **`/dev/sdb`**.

3. Verify that the USB drive is not currently mounted, or unmount it if it is. Use the **`mount`** command to list the currently mounted devices.
4. Use the **`sudo dd`** command to make the USB drive bootable by copying the ISO image.

```
# sudo dd if=<path to iso image> of=/dev/sdb bs=4096
```

Note: Using the **`dd`** command with the wrong disk name can erase the installed OS or other vital information.

Replace ***<path to iso image>*** with the file name and path to the location where you downloaded the ISO image. For example, the following command copies **`bmf-service-node.iso`** to the USB drive:

```
# sudo dd if=bmf-service-node.iso of=/dev/sdb bs=4096
```

It can take up to ten minutes to copy the image to the USB drive.

5. Eject the USB drive from your Linux workstation.

B.1.3 Creating a USB Boot Image Using Windows

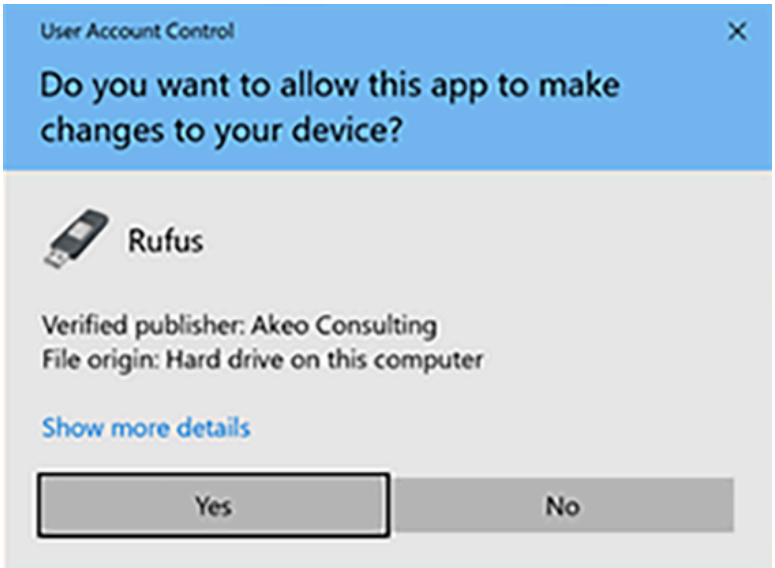
A number of Windows utilities are available for building a USB boot image from an ISO image. The following procedure uses the Rufus bootable image program.

To build a USB boot image using Windows, complete the following steps.

1. Download the Rufus utility from https://rufus.akeo.ie/.
2. After downloading the utility, double-click the **`rufus.exe`** file.

The system displays the following dialog box

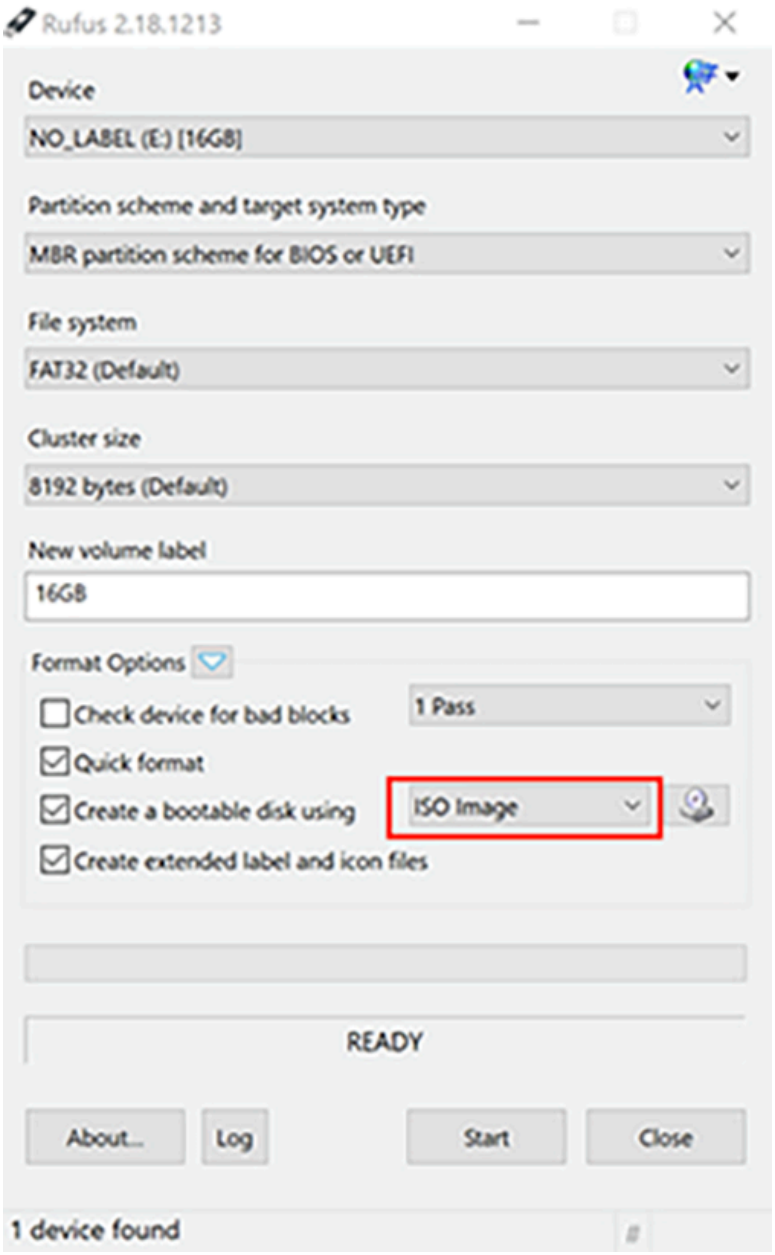
Figure B-1: User Account Control



3. Click OK to allow the changes required for installation.

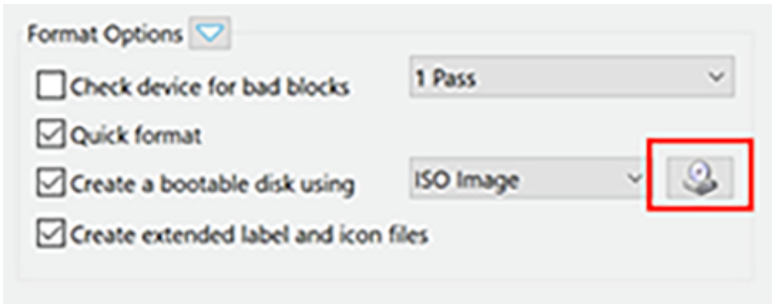
The system displays the following dialog box

Figure B-2: Rufus: Create an ISO Image Option



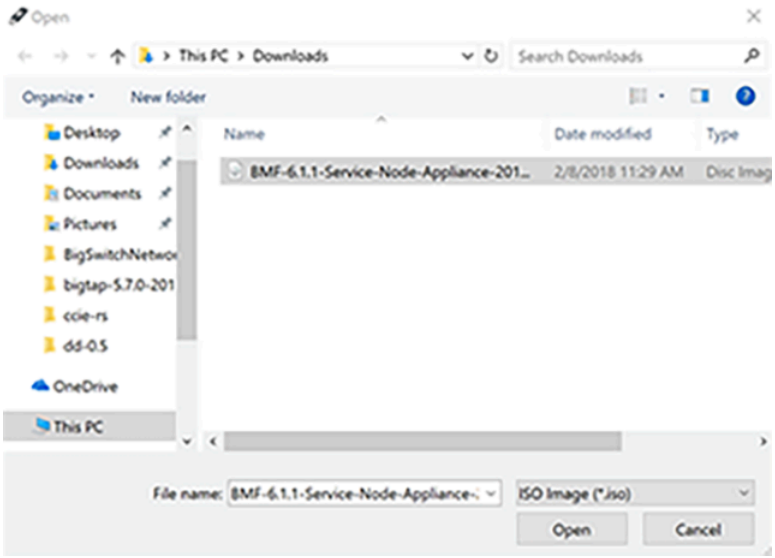
4. To create a bootable disk select ISO Image.

Figure B-3: Rufus: Select ISO Image



5. Click the CD-ROM icon.
   The system displays the following dialog box.

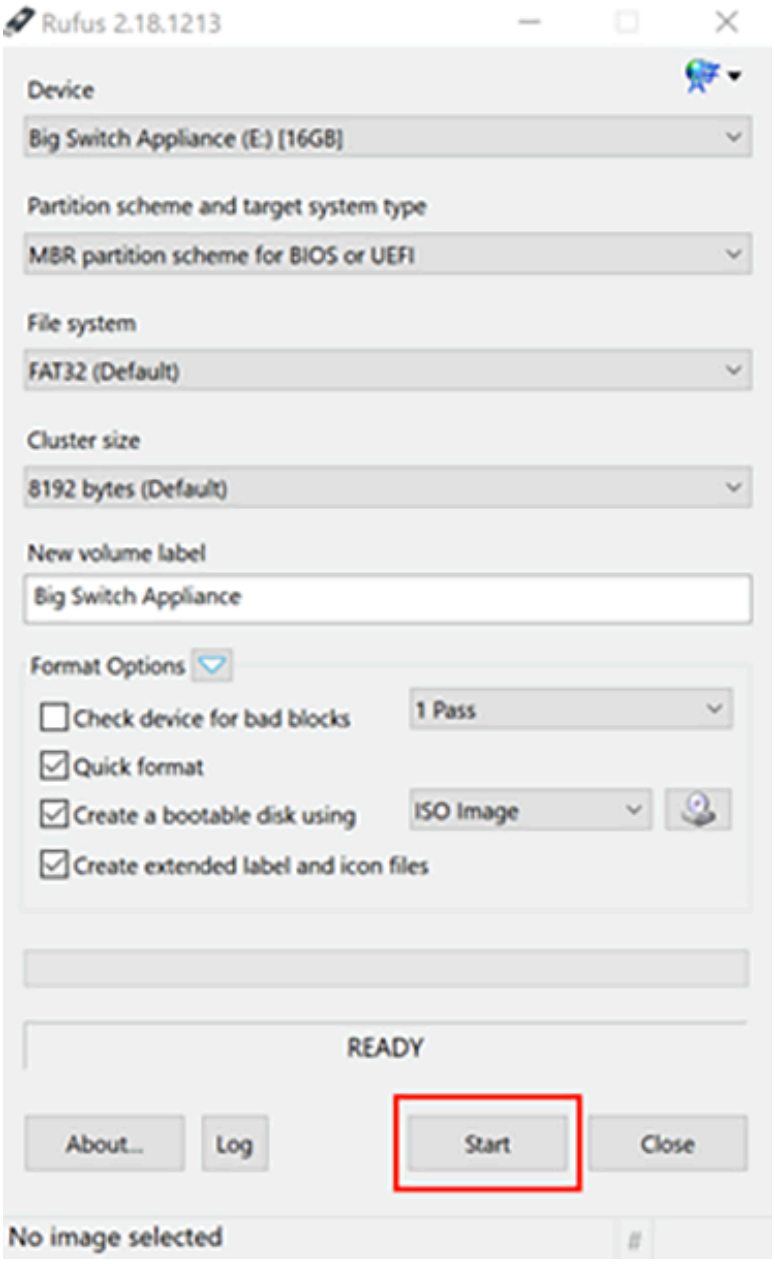
   Figure B-4: Open ISO Image File



6. Select the file to use and click **Open**.

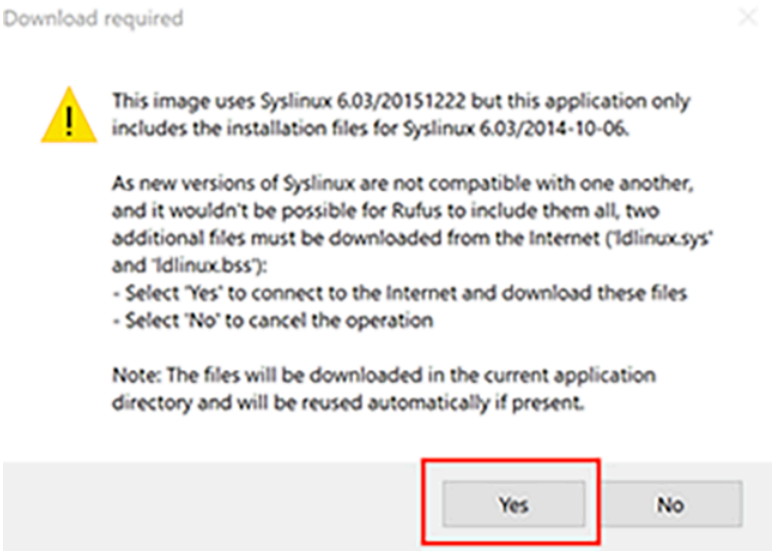
7. Click **Start** to burn the ISO image to USB.

Figure B-5: Rufus: Start



If upgrade to syslinux is required, the system displays the following dialog box.

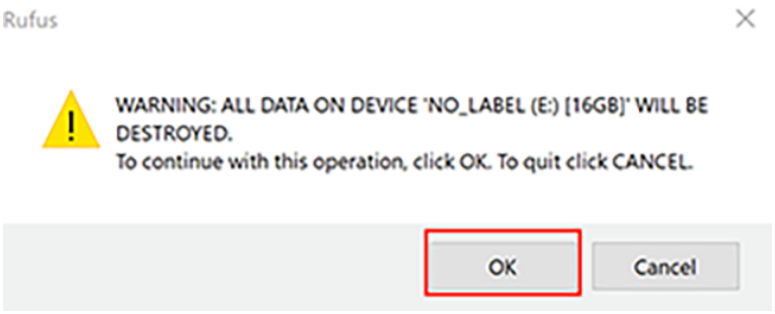
Figure B-6: User Account Control



8. If this prompt appears, click **Yes** to continue.
9. When prompted to use DD mode or ISO mode, choose ISO.

   The system displays a warning that the data on the USB drive is going to be destroyed and a new image is going to be installed

Figure B-7: Erasing Data Warning



10. Click **OK** to confirm the operation.

Appendix C

References

C.1 Related Documents

The following documentation is available for ***Arista Analytics 8.5.0***:

- ***Arista Analytics User Guide***