# RSA NETWITNESS® PLATFORM

# AWS Installation Guide

for RSA NetWitness® Platform 11.5

## Contact Information

RSA Link at https://community.rsa.com contains a knowledge base that answers common questions and provides solutions to known problems, product documentation, community discussions, and case management.

## Trademarks

RSA Conference Logo, 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 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 RSA Link. 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.

September 2020

Revised: September 2020

# Contents

# AWS Installation Overview

Before you can deploy RSA NetWitness® Platform in the Amazon Web Services (AWS) you need to:

- Review the recommended compute and memory specifications needed for each RSA NetWitness® Platform instance.

- Get familiar with the RSA NetWitness® Platform Storage Guide to understand the types of drives and volumes needed to support NetWitness instances. For more information, see the Storage Guide for RSA NetWitness® Platform 11.x.

- Make sure that you have a NetWitness Platform Throughput license.

When you are ready to begin deployment, you can purchase any of the following Third-Party solutions for packet capture in AWS. If you engage one of these third-parties, they will assign an account representative and a professional services engineer to you who will work closely with RSA Support.

- Gigamon® GigaVUE

- Ixia CloudLens™

- f5® BIG-IP

# AWS Environment Recommendations

AWS instances have the same functionality as the NetWitness Platform Azure, virtual, and hardware hosts. RSA recommends that you perform the following tasks when you set up your AWS environment.

- Based on the resource requirements of the different components, follow the best practices to use the system and the dedicated storage Elastic Block Store (EBS) Volumes appropriately.

- Make sure that the compute capacity provides a write speed of 10% greater than the required sustained capture and ingest rate for the deployment.

- Build a Concentrator directory for the index database on the Provisioned IOPS SSD.

# AWS Deployment Scenarios

Before you can deploy NetWitness Platform you need to:

- Consider the requirements of your enterprise and understand the deployment process.

- Have a high-level picture of the complexity and scope of a NetWitness Platform deployment.

## Process

The components and topology of a NetWitness Platform network can vary greatly between installations, and should be carefully planned before the process begins. Initial planning includes:

- Consideration of site requirements and safety requirements.

- Review of the network architecture and port usage.

- Support of group aggregation on Archivers and Concentrators, and virtual hosts.

When updating hosts and services, follow recommended guidelines under the "Running in Mixed Mode" topic in the *RSA NetWitness Platform Host and Services Getting Started Guide*.

You should also become familiar with Hosts, Host Types, and Services as they are used in the context of NetWitness Platform also described in the *RSA NetWitness Platform Host and Services Getting Started Guide*.

## NetWitness Platform High-Level Deployment Diagram

NetWitness Platform is inherently modular. Whether organizations are looking to deploy on-premise or in the cloud, the NetWitness components are decoupled in a way which allows flexible deployment architectures to satisfy a variety of use cases.

The following figure is an example of a hybrid cloud deployment, where the base of the components are residing within the SecOps VPC. Centralizing these components make management easier while keeping network latency to a minimum.

Network, log and endpoint traffic could then be aggregated up to the SecOps VPC. The on-premise location would function just like a normal physical deployment and would be accessible for investigations and analytics.

Cloud SaaS visibility could be captured from a Log Decoder residing in either the cloud or on-premise locations.
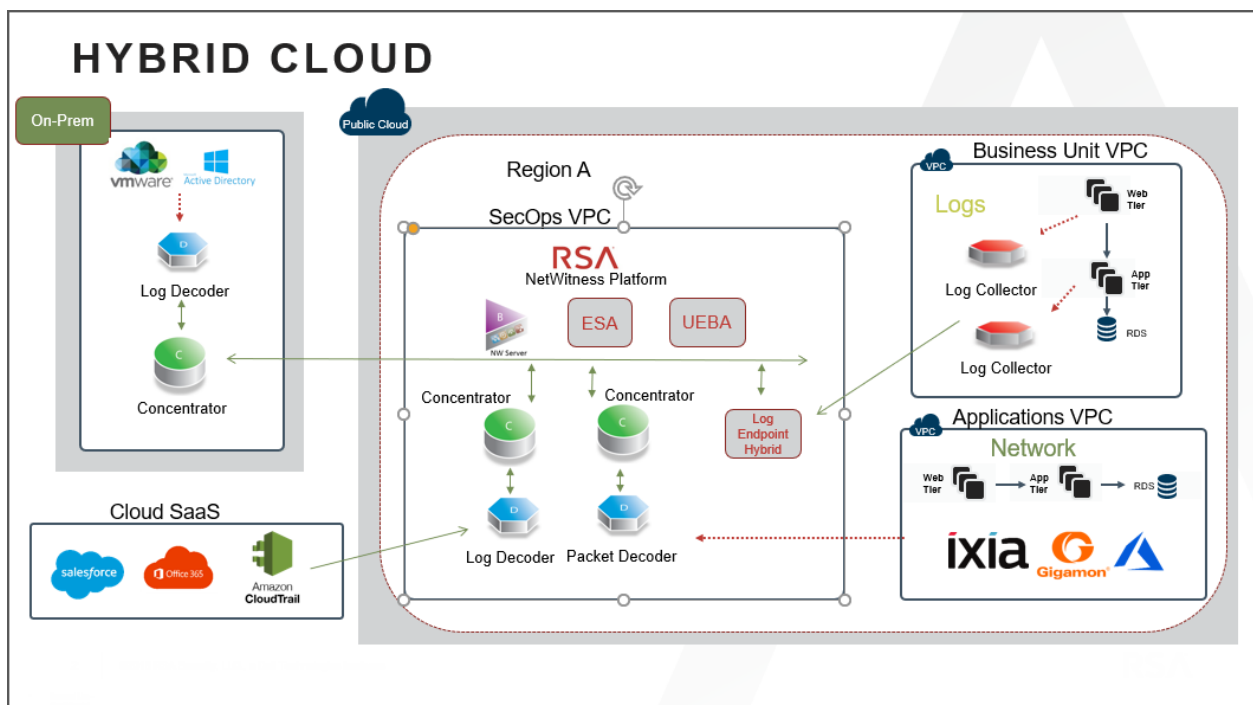


## Prerequisites

You need the following items before you begin the installation process:

- Ixia account (https://login.ixiacom.com/)
- Access to AWS console
- Network rout-able (and proper AWS Security Groups) for the containers to transfer data to the NetWitness Platform Decoder.

# AWS Deployment

This topic contains the rules and high-level tasks you must follow to deploy RSA NetWitness® Platform components in the AWS.

## Rules

You must adhere to the following rules when deploying NetWitness Platform in AWS.

- If you reboot the Network Decoder instance, the tunnel is not retained. Create the tunnel on Network Decoder again and restart the Decoder service.

- It is recommended to use private IP addresses when you provision AWS NetWitness Platform instances.

> **Note:** If you assign a public IP to the NW Server Host, update the /etc/nginx/conf.d/nginx.conf configuration file as follows:
> ```
> location /nwrpmrepo
> {
> alias /var/lib/netwitness/common/repo;
> index index.html index.htm;
> allow <Subnet-Gateway>/Subnet mask ;
> #example
> # allow 10.0.0.1/25;
> deny all;
> autoindex on;
> }
> ```

## Checklist

| Step | Description | √ |
|------|-------------|---|
| 1 | Establish AWS Environment | |
| 2 | Find NetWitness Platform  AMIs | |
| 3 | Launch an Instance and Configure a Host | |
| 4 | Configure Hosts (Instances) in NetWitness Platform | |
| 5 | Configure Packet Capture | |

## Establish AWS Environment
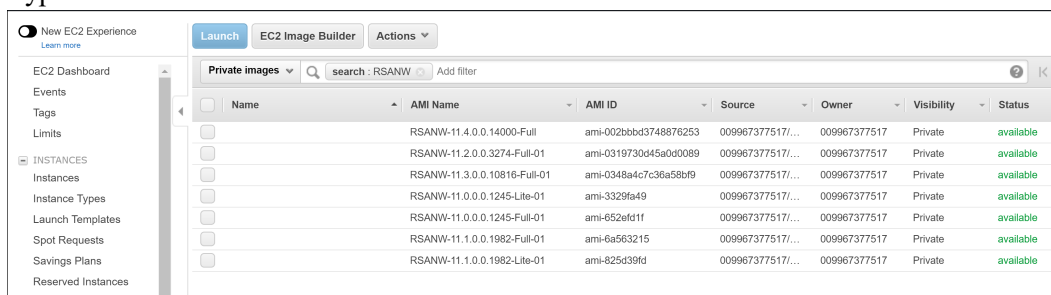
1. Make sure that you have an AWS environment with the capacity to meet or exceed the NetWitness Platform performance guidelines described in AWS Instance Configuration Recommendations.

2. Go to Find NetWitness Platform  AMIs.

# Find NetWitness Platform  AMIs

You can search for NW- AMI files within the Public/Shared/Community repository, using the keyword "RSANW".

> **Note:** For more information, see AWS **Finding Shared AMIs** documentation (http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/usingsharedamis-finding.html).

1.  Open the Amazon EC2 console (New Subscriber Account) at https://console.aws.amazon.com/ec2/.

2.  In the navigation pane, choose AMIs.

3.  In the first filter, choose Public images.

4.  Type "RSANW" in the search field to find the NetWitness Platform AMIs.



> **Note:** Contact RSA Customer Support (https://community.rsa.com/docs/DOC-1294) to obtain access to the **RSA-11.5.0.0.10816-Full-01**.

5.  Go to Launch an Instance and Configure a Host.

# Launch an Instance and Configure a Host

> **Note:** Fore more information, see AWS "Launching an Instance" documentation (http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/launching-instance.html).

1.  Select an instance from the grid (for example, **RSA-NW-Concentrator-11.5.0.0-01**) and click **Launch**.
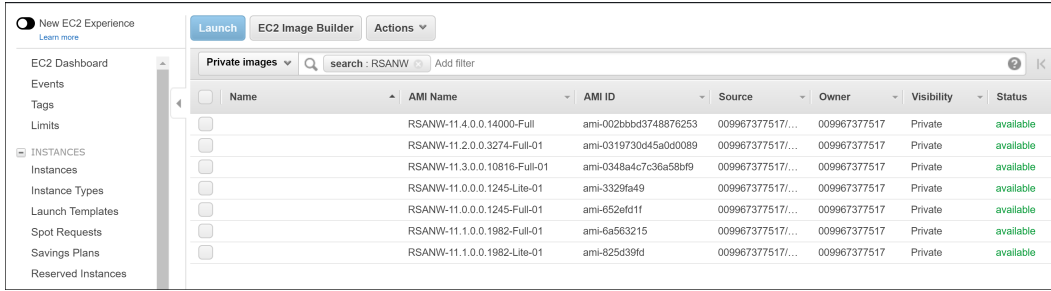
2. Choose the RAM and CPUs by selecting the instance type.
For more information, see Storage Guide for RSA NetWitness® Platform 11.x for guidelines on how to configure the EC2 Instance based on the requirements of the NetWitness Platform component (that is, service) for which you are launching an instance. The following example has the **m4.2xlarge** instance type selected with **8** CPUs and **32** GB of RAM.



3. Click **Next: Configure Instance Details** at the bottom right of the **Step 2: Choose an Instance Type** page.
The **Step 3. Configure Instance Details** page is displayed.
For NetWitness Platform, the subnet and VPC are defaulted to the values.
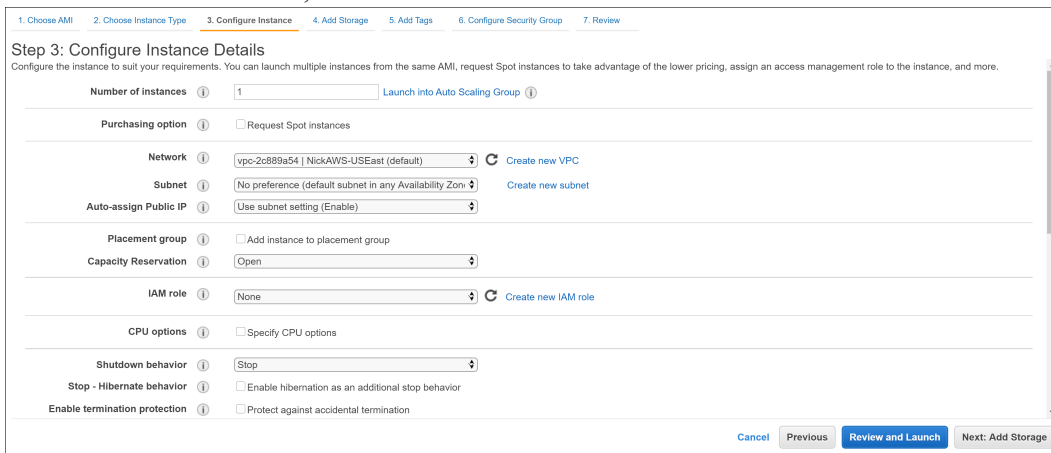


4. Click **Next: Add Storage** at the bottom right of the **Step 3: Configure Instance Details** page.
The **Step 4. Add Storage** page is displayed.

For more information, see the Storage Guide for RSA NetWitness® Platform 11.x for guidelines on how to configure storage based on based on the requirements of the NetWitness Platform component (that is, service) for which you are launching an instance.



5. Click **Next: Add Tags** at the bottom right of the **Step 4: Add Storage** page.
The **Step 5. Add Tags** page is displayed. Enter the name of your Instance.

6. Click **Next: Configure Security Group** at the bottom right of the **Step 5: Add Tags** page.
The **Step 6. Configure Security Group** page is displayed.

   a. Select the "Create a **new** security group" radio button.

   b. Create a rule that opens all the firewall ports for the NetWitness Platform component.
   You must configure the security group correctly to configure the instance (host) from the NetWitness Platform User Interface and SSH to it.

   > **Note:** Fore more information. see the "Network Architecture and Ports" documentation on RSA Link (https://community.rsa.com/docs/DOC-83050) for a comprehensive list of the ports you must set up for all NetWitness Platform components..



> **Note:** After you configure a Security Group, you can change it at any time.

7. Click **Review and Launch** at the bottom right of the **Step 6: Configure Security Group** page.
The **Step 7. Review Instance Launch** page is displayed.

8. Click **Launch** at the bottom right of the **Step 7. Review Instance Launch** page.
The **Select an existing key pair or create a new key pair** dialog is displayed.

9. Choose **Proceed without key pair**.

10. Click **Launch Instance**.

AWS displays the following information as it builds the Instance.
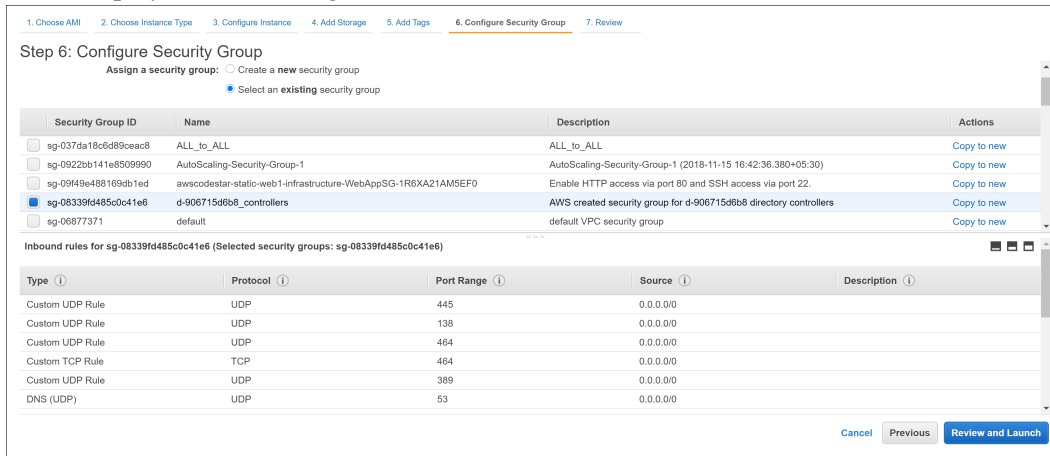


11. Click **View Instances**.

12. Select **Instances** from the left navigation panel to review all instances that AWS is initializing (for example, the **NW-Concentrator**) .
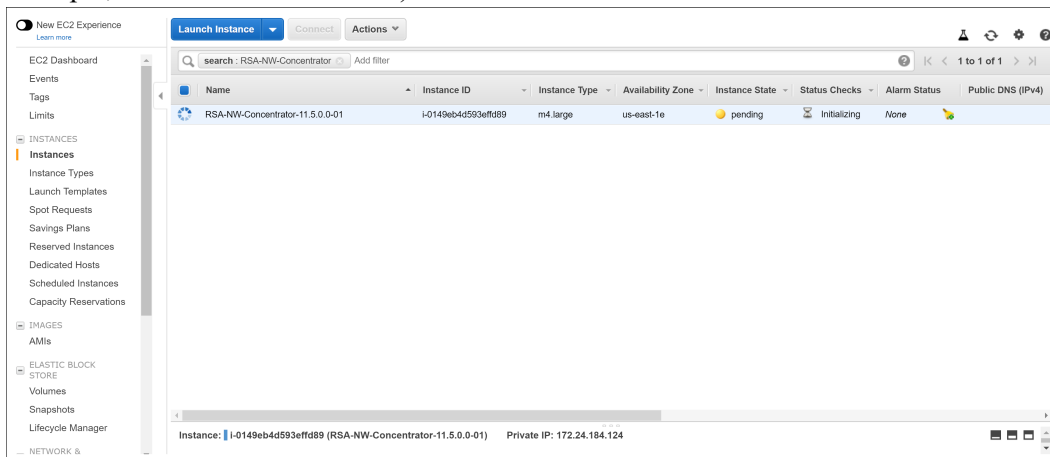


The IP Address for the new **RSA-NW-Concentrator-11.5.0.0-01** host is *sample-ip-address*.
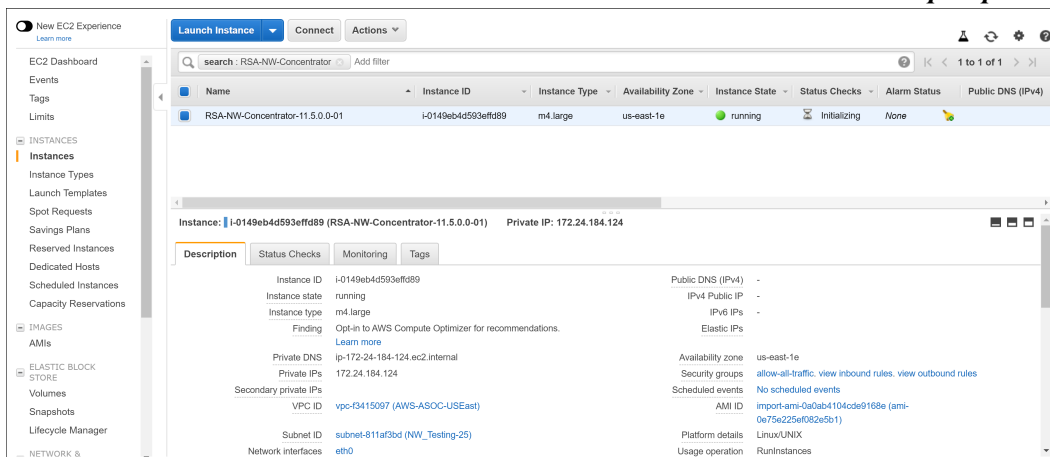


13. SSH to the newly-created instance using the default NetWitness Platform credentials.

14. Go to Configure Hosts (Instances) in NetWitness Platform .

# Storage Configurations

For storage allocations of all host types, see the Prepare Virtual or Cloud Storage topic in the *Storage Guide for RSA NetWitness® Platform 11.x*.

# Installation Tasks

Before you begin the installation tasks make sure you open the firewall ports. For more information on the lists of all the ports in a deployment, see the"Network Architecture and Ports" topic in the *Deployment Guide for RSA NetWitness Platform Guide*.

> **Caution:** Do not proceed with the installation until the ports on your firewall are configured.

## Task 1 - Install 11.5.0.0 on the NW Server Host and Component Hosts

Complete the following steps to install 11.5 on NW Server host and other component hosts. Steps that are specific to the NW Server host or to component hosts are noted.

> **Note:** You can perform this task using RSANW-11.6.0.0.17049-Full or RSANW-11.6.0.0.17049-Lite instance.

> **Caution:** If you want to install the Endpoint Relay Server, do not run the `nwsetup-tui` script. Follow the instructions in "(Optional) Installing and Configuring Relay Server" in the *NetWitness Endpoint Configuration Guide*.

1. Log in to the host with the `root` credentials and run the `nwsetup-tui` command to set up the host.

   This initiates the `nwsetup-tui` (Setup program) and the EULA is displayed.

> **Note:** Use the following options to navigate the Setup prompts.
> 1.) When you navigate through the Setup program prompts, use the down and up arrows to move among fields, and use the Tab key to move to and from commands (such as **<Yes>**, **<No>**, **<OK>**, and **<Cancel>**). Press **Enter** to register your command response and move to the next prompt.
> 2.) The Setup program adopts the color scheme of the desktop or console you use to access the host.
> 3.) If you specify DNS servers during the Setup program (`nwsetup-tui`) execution, they MUST be valid (valid in this context means valid during setup) and accessible for the `nwsetup-tui` script to proceed. Any misconfigured DNS servers cause the Setup program to fail. If you need to reach a DNS server after setup that is unreachable during setup, (for example, to relocate a host after setup that would have a different set of DNS Servers), see "(Optional) Task 1 - Re-Configure DNS Servers Post 11.5" in the "Post Installation Tasks" section in this guide.
> If you do not specify DNS Servers during setup (`nwsetup-tui`), you must select **1 The Local Repo (on the NW Server)** in the **NetWitness Platform Update Repository** prompt in step 12 (the DNS servers are not defined so the system cannot access the external repo).

```
By clicking "Accept", you (the "Customer") hereby agree, on behalf of your
company or organization, to be bound by the terms and conditions of the
End User License Agreement (the "EULA") located at
https://www.rsa.com/content/dam/rsa/PDF/shrinkwrap-license-combined.pdf
with RSA Security LLC ("RSA", or appropriate affiliate entity in the
relevant jurisdiction).  In addition, Customer hereby agrees and
acknowledges that, if Customer chooses to host its data with any third
party or in a public cloud environment, RSA has no responsibility for the
storage or protection of any Customer data or for any associated security
breach notifications. The terms herein and in the EULA shall supersede any
relevant terms in any other agreement between the Customer and RSA.  For
customers of the RSA NetWitness® products, all data analyzed in connection
herewith shall be at a cost to Customer based on RSA's then current
                                                                    92%

              <Accept >              <Decline>
```

2. Tab to **Accept** and press **Enter**.
   The **Is this the host you want for your 11.5 NW Server** prompt is displayed.

```
You must setup an NW Server before setting up
any other NetWitness Platform components.

Is this the host you want for your 11.5 NW
Server?

        < Yes >        < No  >
```
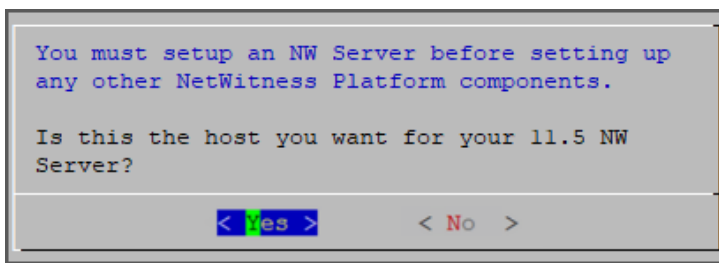
3. Tab to **Yes** and press **Enter** to install 11.5 on the NW Server.
   Tab to **No** and press **Enter** to install 11.5 on other component hosts.

> **Caution:** If you choose the wrong host for the NW Server and complete the Setup, you must restart the Setup Program (step 2) and complete steps all the subsequent steps to correct this error.

4. The **Install** prompt is displayed (**Recover** does not apply to the installation. It is for 11.5 Disaster Recovery.).

**NW Server Host prompt**:

```
NetWitness Platform 11.5 Installation
 Specify the install type: Fresh
 Install, Reinstall, or Warm Standby
 NW Server Install.

      1  Install (Fresh Install)
      2  Recover (Reinstall)
      3  Install (Warm/Standby)



         <  OK  >        < Exit >
```
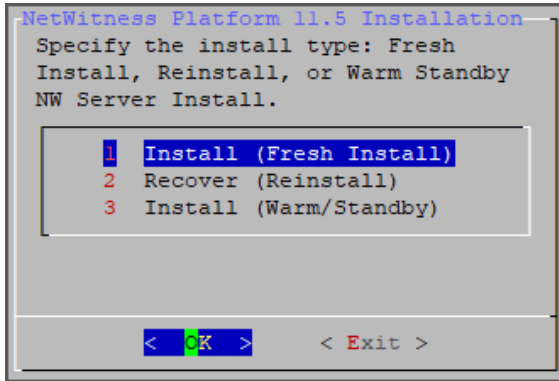
**Other Component Hosts, the prompt is the same, but does not include option 3 Install (Warm/Standby)**

5. Press **Enter**. **Install (Fresh Install)** is selected by default.
   The **System Host Name** prompt is displayed.
   **NW Server prompt**:

```
              System Host Name
 Please accept or update the system
 host name:

     <nwserver-host-name>


         <  OK  >        <Cancel>
```

**Other Component Hosts prompt says <non-nwserver-host-name>**

> **Caution:** If you include "." in a host name, the host name must also include a valid domain name.
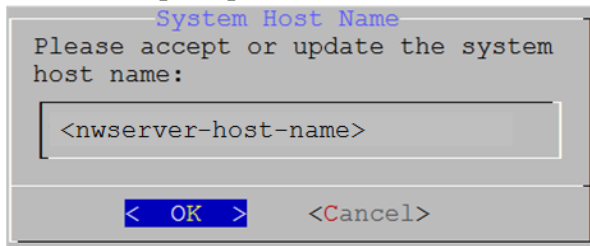
Press **Enter** if want to keep this name. If not, edit the host name, tab to **OK**, and press **Enter** to change it.

6. **This step applies only to NW Server hosts**.
   The **Master Password** prompt is displayed.

```
                   Master Password
 The master password is utilized to set the default password for both
 the system recovery account and the NetWitness UI "admin" account.
 The system recovery account password should be safely stored in case
 account recovery is needed.  The NetWitness UI "admin" account
 password can be updated upon login.

 Enter a Master Password.

 Password ************
 Verify   ************

         <  OK  >             <Cancel>
```
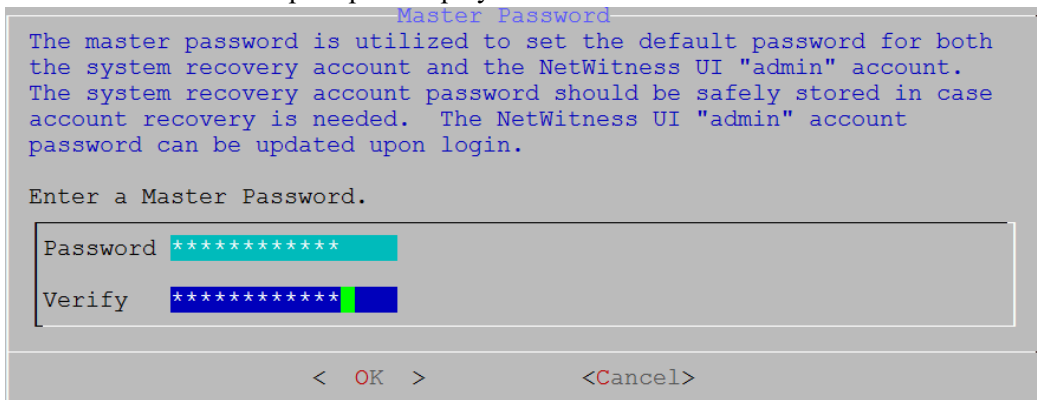
The following list of characters are supported for Master Password and Deployment Password:

---

- Symbols: **! @ # % ^ +**

- Numbers: **0-9**

- Lowercase Characters: **a-z**

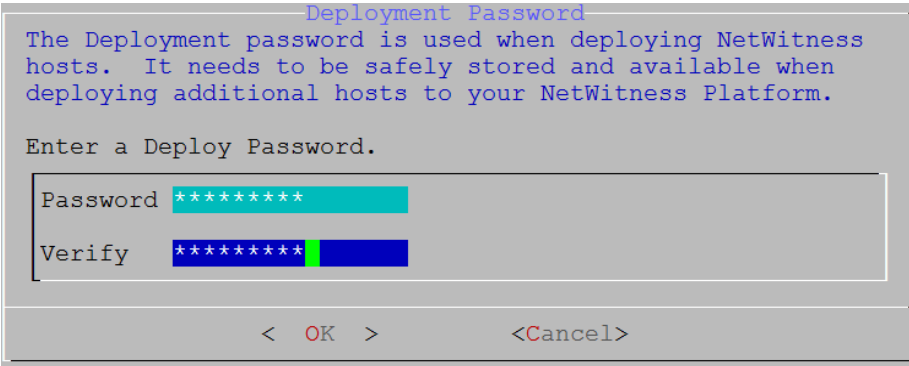- Uppercase Characters: **A-Z**

No ambiguous characters are supported for Master Password and Deployment Password. For example:
 space { } [ ] ( ) / \ ' " ` ~ ; : . < > -

Type the **Password**, down arrow to **Verify**, retype the password, tab to **OK**, and press **Enter**.

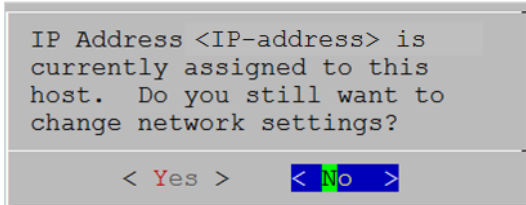7. **This step applies to both NW Server hosts and component hosts.**
   The **Deployment Password** prompt is displayed.

```
                  Deployment Password
 The Deployment password is used when deploying NetWitness
 hosts.  It needs to be safely stored and available when
 deploying additional hosts to your NetWitness Platform.

 Enter a Deploy Password.

  Password ********
  
  Verify   ********


           <  OK  >           <Cancel>
```

Type the **Password**, down arrow to **Verify**, retype the password, tab to **OK**, and press **Enter**.
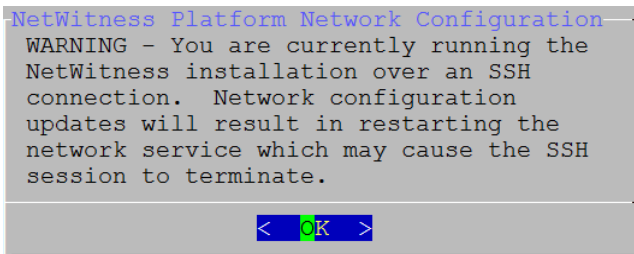
8. One of the following conditional prompts is displayed.

- If the Setup program finds a valid IP address for this host, the following prompt is displayed.

```
  IP Address <IP-address> is
  currently assigned to this
  host.  Do you still want to
  change network settings?

       < Yes >      < No  >
```

Press **Enter** if you want to use this IP and avoid changing your network settings. Tab to **Yes** and press **Enter** if you want to change the IP configuration on the host.

- If you are using an SSH connection, the following warning is displayed.

> **Note:** If you connect directly from the host console, the following warning is not displayed.

```
NetWitness Platform Network Configuration
 WARNING - You are currently running the
 NetWitness installation over an SSH
 connection.  Network configuration
 updates will result in restarting the
 network service which may cause the SSH
 session to terminate.

            <  OK  >
```
Press **Enter** to close warning prompt.

- If the Setup Program finds an IP configuration and you choose to use it, the **Update Repository** prompt is displayed. Go to step 12 to and complete the installation.

- If the Setup Program did not find an IP configuration or if you choose to change the existing IP configuration, the **Network Configuration** prompt is displayed.
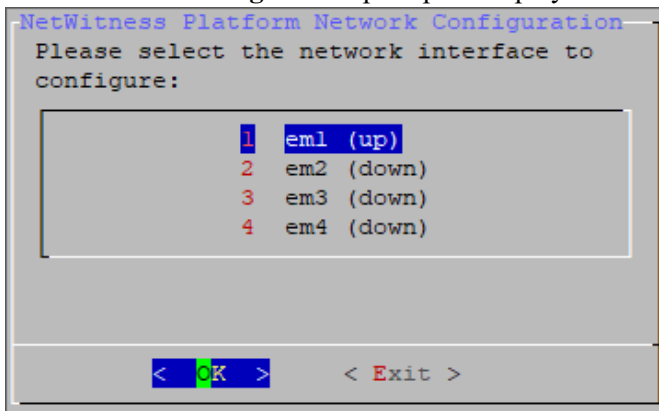
**Caution:** Only select **"Use DHCP"** as an IP address configuration for the NW Server if DHCP issues static IP addresses.

```
─────────NetWitness Platform Network Configuration─────────
 The IP address of the NW Server is used by all other NetWitness
 Platform components.  RSA recommends that you use a Static IP
 Configuration for the NW Server IP address over DHCP.  After the
 IP address is assigned, record it for future use.  You need this
 address to set up other components.

 Select an IP address configuration for the NW Server.

                 1  Static IP Configuration
                 2  Use DHCP



            <  OK  >            < Exit >
```
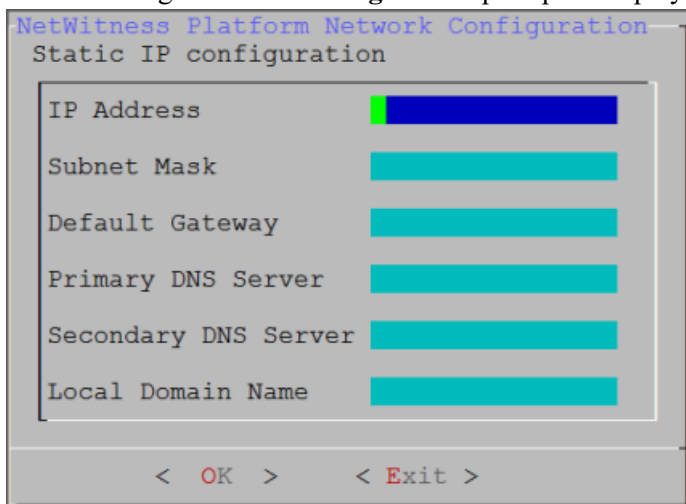
Tab to **OK** and press **Enter** to use **Static IP**.

If you want to use DHCP, down arrow to **2 Use DHCP** and press **Enter**.

The **Network Configuration** prompt is displayed.

```
─NetWitness Platform Network Configuration─
 Please select the network interface to
 configure:

        1  em1 (up)
        2  em2 (down)
        3  em3 (down)
        4  em4 (down)



    <  OK  >      < Exit >
```

9.  Down arrow to the network interface you want, tab to **OK**, and press **Enter**. If you do not want to continue, tab to **Exit.**

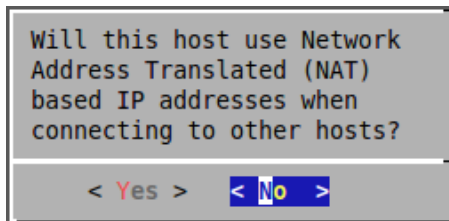The following **Static IP Configuration** prompt is displayed.

```
NetWitness Platform Network Configuration
 Static IP configuration

 IP Address            [                    ]

 Subnet Mask           [                    ]

 Default Gateway       [                    ]

 Primary DNS Server    [                    ]

 Secondary DNS Server  [                    ]

 Local Domain Name     [                    ]


        <  OK  >         < Exit >
```

10. Type the configuration values, tab to **OK**, and press **Enter**. If you do not complete all the required fields, an `All fields are required` error message is displayed (**Secondary DNS Server** and **Local Domain Name** fields are not required). If you use the wrong syntax or character length for any of the fields, an `Invalid <field-name>` error message is displayed.
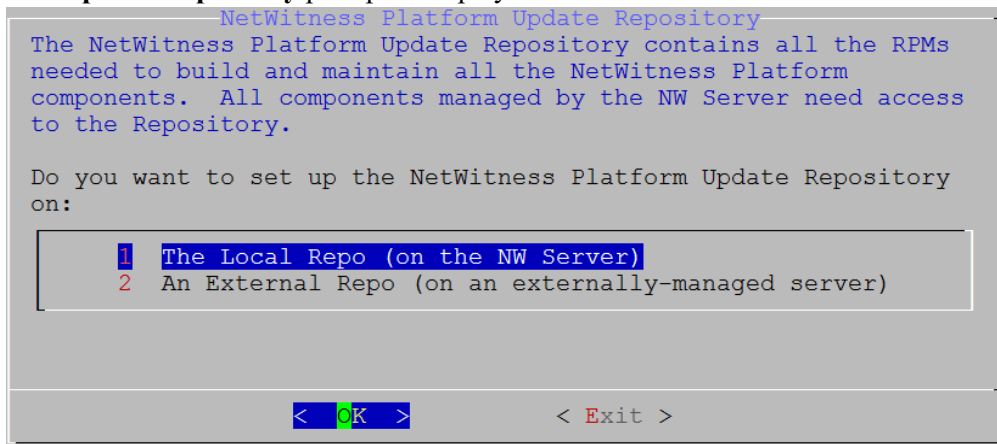
> **Caution:** If you select **DNS Server**, make sure that the DNS Server is correct and the host can access it before proceeding with the installation.

11. The Use Network Address Translation (NAT) prompt is displayed.

```
Will this host use Network
Address Translated (NAT)
based IP addresses when
connecting to other hosts?

   < Yes >    < No  >
```
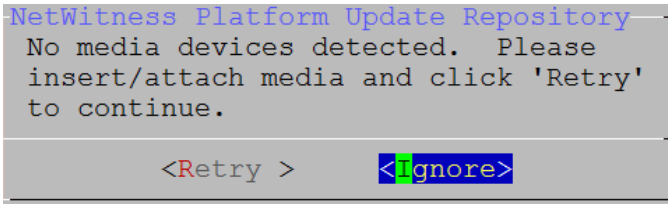
**For the NW Server**, tab to **No** and press **Enter**.
**For component hosts**, if this host requires the use of NAT-based addresses to communicate with the NW Server, tab to **Yes**. Otherwise, tab to **No** and press **Enter**.

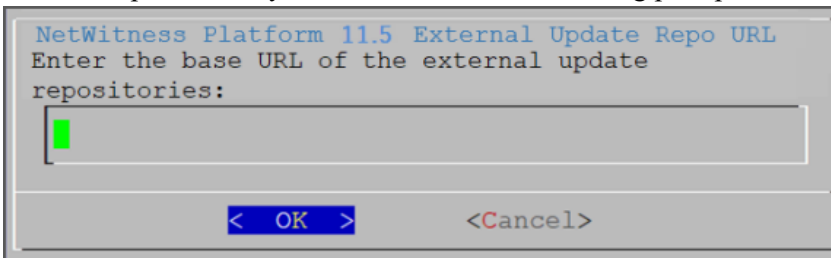12. The **Update Repository** prompt is displayed.

```
            NetWitness Platform Update Repository
 The NetWitness Platform Update Repository contains all the RPMs
 needed to build and maintain all the NetWitness Platform
 components.  All components managed by the NW Server need access
 to the Repository.

 Do you want to set up the NetWitness Platform Update Repository
 on:

      1  The Local Repo (on the NW Server)
      2  An External Repo (on an externally-managed server)



         <  OK  >              < Exit >
```

**For the NW Server**:

- Press **Enter** to choose the **Local Repo**.

- If you want to use an external repo, down arrow to **External Repo**, tab to **OK**, and press **Enter**. If you select **1 The Local Repo (on the NW Server)** in the Setup program, make sure that you have the appropriate media attached to the host (media that contains the ISO file, for example a build stick) from which it can install NetWitness Platform 11.5. If the program cannot find the attached media, you receive the following prompt.
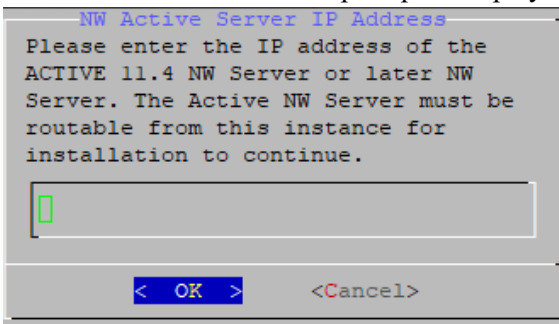
```
NetWitness Platform Update Repository
 No media devices detected.  Please
 insert/attach media and click 'Retry'
 to continue.

         <Retry >        <Ignore>
```

- If you select **2 An External Repo (on an externally-managed server)**, the UI prompts you for a URL. The repositories give you access to RSA updates and CentOS updates. Refer to "Appendix B. Create an External Repo" in this guide for instructions on how to create this repo and its external repo URL so you can enter it in the following prompt.

```
 NetWitness Platform 11.5 External Update Repo URL
 Enter the base URL of the external update
 repositories:

  [                                        ]

        <   OK   >        <Cancel>
```

Enter the base URL of the NetWitness Platform external repo and click **OK.** The **Start Install** prompt is displayed.
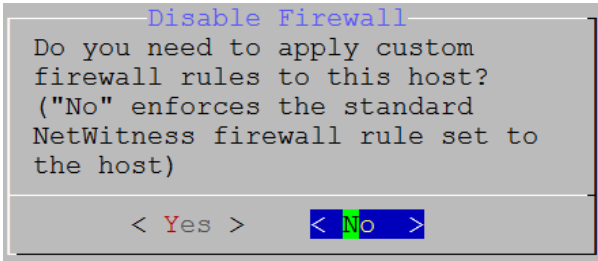
**For component hosts**:

- Select the same repo that you selected when you installed the NW Server host and follow the steps above.

- The NW Server IP Address prompt is displayed.

```
      NW Active Server IP Address
 Please enter the IP address of the
 ACTIVE 11.4 NW Server or later NW
 Server. The Active NW Server must be
 routable from this instance for
 installation to continue.

  [                        ]

        <  OK  >     <Cancel>
```

Type the NW Server IP address. Tab to **OK** and press **Enter**.
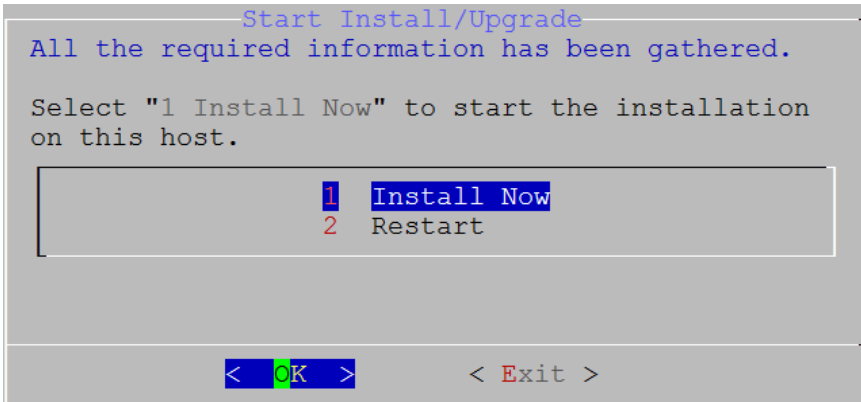
13. The Disable firewall prompt is displayed.

```
┌─────────Disable Firewall─────────┐
│ Do you need to apply custom       │
│ firewall rules to this host?      │
│ ("No" enforces the standard       │
│ NetWitness firewall rule set to   │
│ the host)                         │
│                                   │
│        < Yes >        < No >      │
└───────────────────────────────────┘
```

Tab to **No** (default), and press **Enter** to use the standard firewall configuration.

To disable the standard firewall configuration, tab to **Yes**, and press **Enter**.

If you select **Yes**, confirm your selection(select **Yes** again) or select **No** to use the standard firewall configuration.

```
┌───────────────────────────────────────────────┐
│ Warning: you chose to disable the default NetWitness │
│ firewall configuration which means you must set up │
│ firewall rules manually.                        │
│                                                 │
│ Select "Yes" to confirm that you will set up firewall │
│ rules manually.                                 │
│                                                 │
│        < Yes >              < No >              │
└─────────────────────────────────────────────────┘
```

14. The **Start Install** prompt is displayed.

```
┌─────────Start Install/Upgrade─────────┐
│ All the required information has been gathered. │
│                                                 │
│ Select "1 Install Now" to start the installation │
│ on this host.                                   │
│                                                 │
│  ┌───────────────────────────────────────────┐ │
│  │ 1   Install Now                            │ │
│  │ 2   Restart                                │ │
│  │                                            │ │
│  │                                            │ │
│  └───────────────────────────────────────────┘ │
│                                                 │
│       <  OK  >          < Exit >               │
└─────────────────────────────────────────────────┘
```
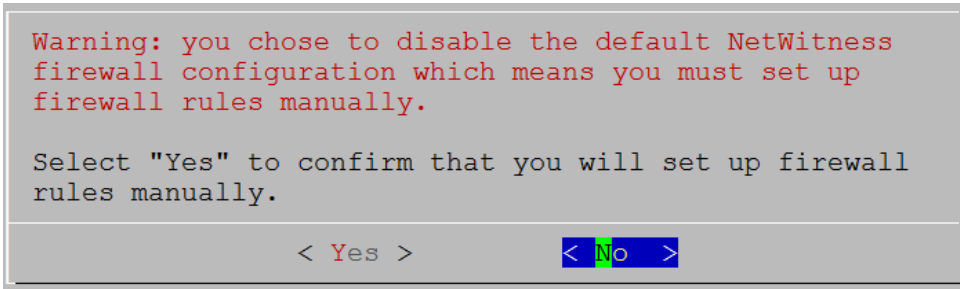
15. Press **Enter** to install 11.5.

When **Installation complete** is displayed, you have installed 11.5 on this host.

> **Note:** Ignore the hash code errors similar to the errors shown in the following figure that are displayed when you initiate the `nwsetup-tui` command. Yum does not use MD5 for any security operations so they do not affect the system security.

```
ValueError: error:3207A06D:lib(50):B_HASH_init:cr new
Checksum type 'md5' disabled
 (skipped due to only_if)
    * file[/etc/yum.repos.d/CentOS-Base.repo] action delete (up to date)
    * ruby_block[yum-cache-reload-CentOS-Base] action nothing (skipped due to action :nothing)
      (up to date)
  * yum_repository[Remove CentOS-CR repository] action delete
    * execute[yum clean all CentOS-CR] action runERROR:root:code for hash md5 was not found.
Traceback (most recent call last):
  File "/usr/lib64/python2.7/hashlib.py", line 129, in <module>
    globals()[__func_name] = __get_hash(__func_name)
  File "/usr/lib64/python2.7/hashlib.py", line 98, in __get_openssl_constructor
    f(usedforsecurity=False)
```

16. (Optional) If your system configuration requires that a component host must use a NAT IP address to reach the NW Server host, you must configure the NAT IP address of the NW Server by running the following command:

```
nw-manage --update-host --host-id <NW Server Host UUID> --ipv4-public <NAT
IP address>
```

## Set Up ESA Hosts

After you install your NW Server and component hosts, follow these steps to set up your ESA hosts.

- Install your primary ESA host following the instructions in "Install 11.5 on the NetWitness Server (NW Server) Host and Other Component Hosts" in this guide, and install the **ESA Primary** service on it after you finish the Set Up program in the UI in ⚒ (Admin) > **Hosts** > 📥 Install ⊙:
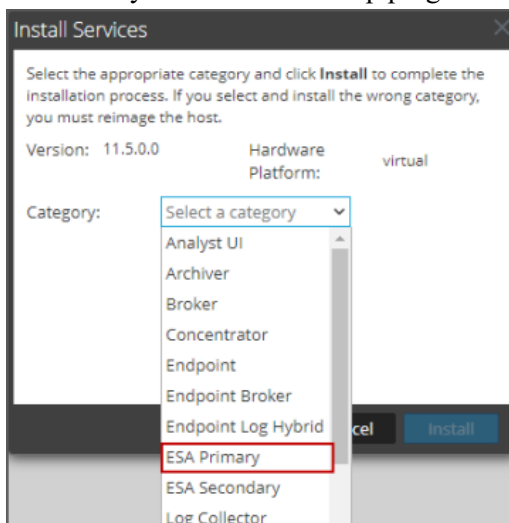
- (Conditional) If you have a secondary ESA host, install it and install the **ESA Secondary** service on it after you finish the Set Up program in the UI in ⚒ (Admin) > **Hosts** > ⊞ Install ⊙:
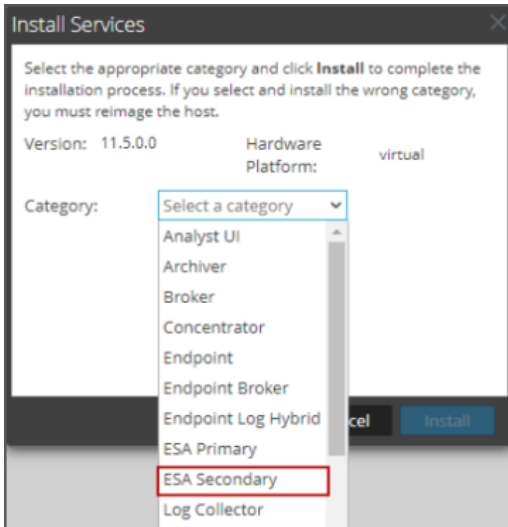


## Install Component Services on Hosts

After you have installed NW Server and component hosts, and set up your ESA hosts, follow these steps to install component services, such as Decoders and Concentrators, on your host systems.

1. Install a component service on the host.

    a. Log into NetWitness Platform and go to ⚒ (Admin) > **Hosts**.
       The **New Hosts** dialog is displayed with the **Hosts** view grayed out in the background.

       > **Note:** If the **New Hosts** dialog is not displayed, click **Discover** in the Hosts view toolbar.

    b. Select the host in the **New Hosts** dialog and click **Enable**.
       The **New Hosts** dialog closes and the host is displayed in the **Hosts** view.

    c. Select that host in the **Hosts** view and click ⊞ Install ⊙.
       The **Install Services** dialog is displayed.

d. Select the appropriate host type (for example, **Concentrator**) in **Category** and click **Install**.
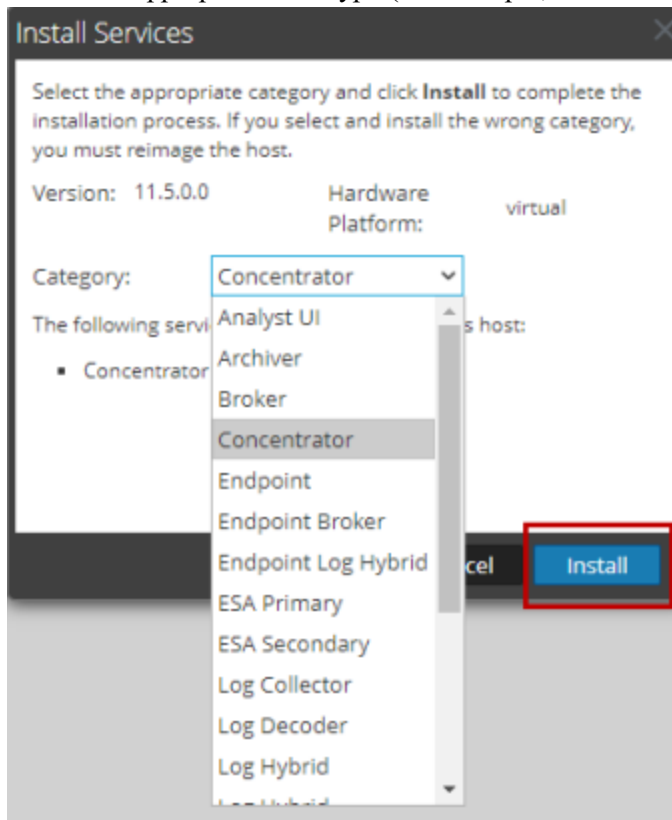


## Complete Licensing Requirements

Complete licensing requirements for installed services. See the *NetWitness Platform 11.5 Licensing Management Guide* for more information. Go to the Master Table of Contents to find all RSA NetWitness Platform 11.x documents.

## (Optional) Install Warm Standby NW Server

Refer to "Warm Standby NW Server Host" under "Deployment Option Setup Procedures" in the *Deployment Guide for RSA NetWitness Platform 11.5* for instructions on how to set up a Warm Standby NW Server.

## Configure Hosts (Instances) in NetWitness Platform

Configure individual hosts and services as described in RSA NetWitness® Platform *Host and Services Configuration Guide*. This guide also describes the procedures for applying updates and preparing for version upgrades.

**Note:** After you successfully launch an instance, AWS assigns a default hostname to it. For more information, see "Change Host Network Configuration" in the *System Maintenance Guide* for instructions on changing a hostname. Go to the Master Table of Contents to find all RSA NetWitness Platform 11.x documents.

# Configure Packet Capture

You can integrate any of the following Third-Party solutions with the Network Decoder to capture packets in the AWS cloud:

- Gigamon® GigaVUE
- Ixia CloudLens™
- f5® BIG-IP
- VPC Traffic Mirroring

## Integrate Gigamon GigaVUE with the Network Decoder

There are two main tasks to configure the Gigamon® third-party Tap vendor packet capture solution:

> Task 1. Integrate the Gigamon® solution.
>
> Task 2. Configure a tunnel on Network Decoder

### Task 1. Integrate the Gigamon Solution

Gigamon® Visibility Platform on AWS will be available through the AWS Marketplace and activated by a BYOL license. A thirty-day free trial is also available.

For more information on the Gigamon® solution, see "Gigamon® Visibility Platform for AWS Data Sheet"https://www.gigamon.com/sites/default/files/resources/datasheet/ds-gigamon-visibility-platform-for-aws-4095.pdf.

For more information on the deployment details, see "Gigamon® Visibility Platform for AWS Getting Started Guide" https://www.gigamon.com/sites/default/files/resources/deployment-guide/dg-visibility-platform-for-aws-getting-started-guide-4111.pdf.

After the "Monitoring Session" is deployed within the Gigamon GigaVUE-FM, you can configure the Network Decoder Tunnel.

### Task 2. Configure Tunnel on the Network Decoder

1. SSH to the Decoder.

2. Submit the following command strings.

```
$ sudo ip link add tun0 type gretap local any remote <ip_address_of_
VSERIES_NODE_TUNNEL_INTERFACE> ttl 255 key 0

$ sudo ip link set tun0 up mtu <MTU-SIZE>

$ sudo ifconfig (to verify if the tunnel tun0 is being listed in the list
of interfaces)

$ sudo lsmod | grep gre ( to make sure if the below kernel modules are
running:

ip_gre 18245 0

ip_tunnel 25216 1)
```

```
If they are not running then execute the below commands to enable the
modules
$ sudo modprobe act_mirred
$ sudo modprobe ip_gre
```

3. Create a firewall rule in the Network Decoder to allow traffic through the tunnel.

   a. Open the `iptables` file.
      ```
      vi /etc/sysconfig/iptables
      ```

   b. Append the line `-A INPUT -p gre -j ACCEPT` before the `commit` statement

   c. Restart `iptables` by executing the following commands.
      ```
      service iptables restart
      service ip6tables restart
      ```

4. Set the interface in the Network Decoder.

   a.  Log in NetWitness Platform, select the `decoder/config` node in Explorer view for the Network Decoder service.

   b. Set the `capture.selected = packet_mmap_,tun0`.
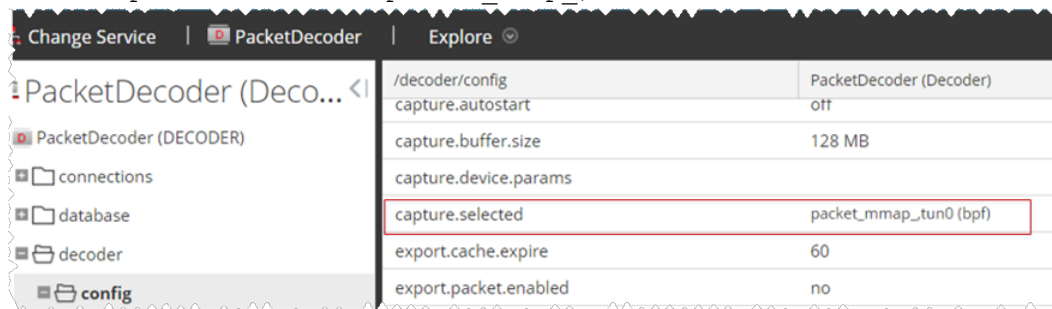


5. (Conditional) - If you have multiple tunnels on the Network Decoder.

   a. Restart Decoder service after you create the tunnel in Network Decoder.

   b. Log in to NetWitness Platform, select the `decoder/config` node in Explorer view for the Network Decoder service, and set the following parameters.

      ```
      capture.device.params = interfaces=tun0,tun1,tun2
      ```

      ```
      capture.selected = packet_mmap_,All
      ```
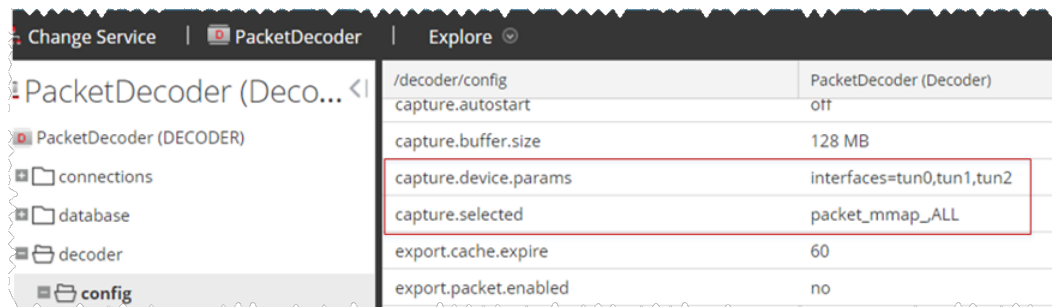


6. Restart decoder service.
   ```
   $ sudo restart nwdecoder
   ```
   The user should be all set to capture the network traffic in Decoder.

# Integrate Ixia with the Network Decoder

You must complete the following tasks to integrate the Network Decoder with Ixia CloudLens.

Task 1. Deploy Client Machines

Task 2. Create CloudLens Project

Task 3. Install Docker Container on Decoder

Task 4. Install Docker Container on Clients

Task 5. Map Network Decoder to Ixia Clients

Task 6. Validate CloudLens Packets Arriving at Decoder

Task 7. Set Interface in Network Decoder

## Task 1. Deploy Client Machines
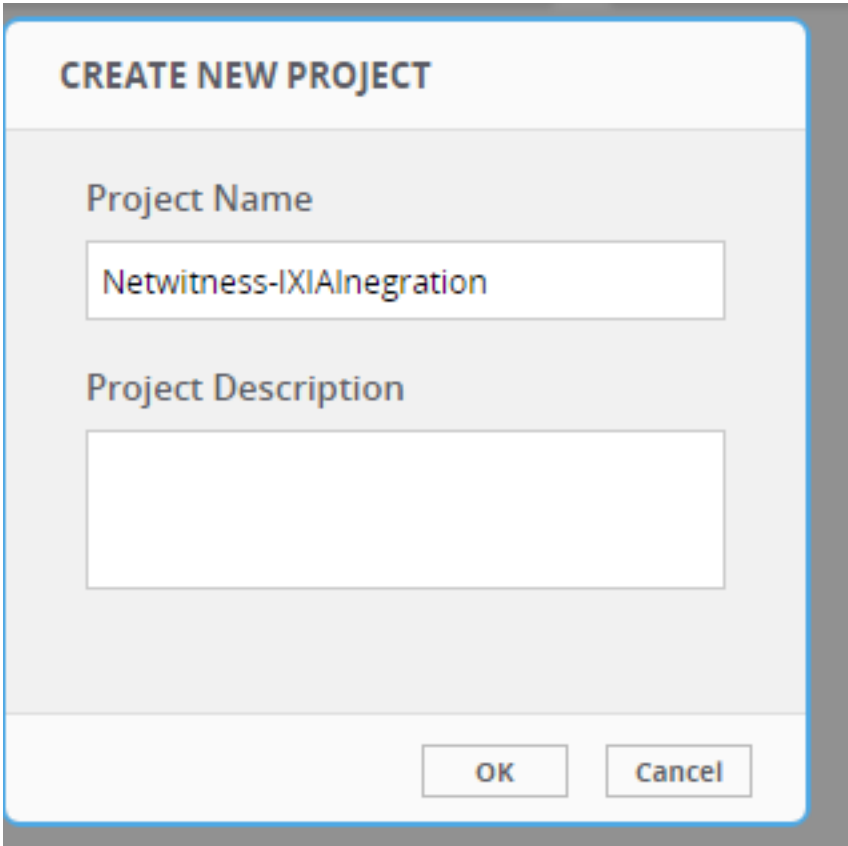
- Deploy client machines onto which you want to route the traffic to the Network Decoder. See the Ixia CloudLens documentation (https://www.ixia.cloud/help/Default.htm) for specifications needed for supported client machines.

- For Client Machines (as well as Decoder machine) the following ports must be opened on AWS Security Group Inbound Rules; UDP 19993 from all, TCP 22 from Admin IP.

## Task 2. Create CloudLens Project

Complete the following steps to create a new project and get your project key.

1. Get Cloudlens login credentials and access to a free trial.

   a. Create an Ixia login account at https://www.ixiacom.com/products/cloudlens-trial-a.

2. Go to the Cloudlens public site (https://www.ixia.cloud).

3. Click + (add) to create a new project with a name of your choosing (for example, **NetWitness-IxiaIntegration**).



4. Click on your newly created project and make note of your Project Key.
   You need the key later for the API key configured on the **Host & Tool agents**.
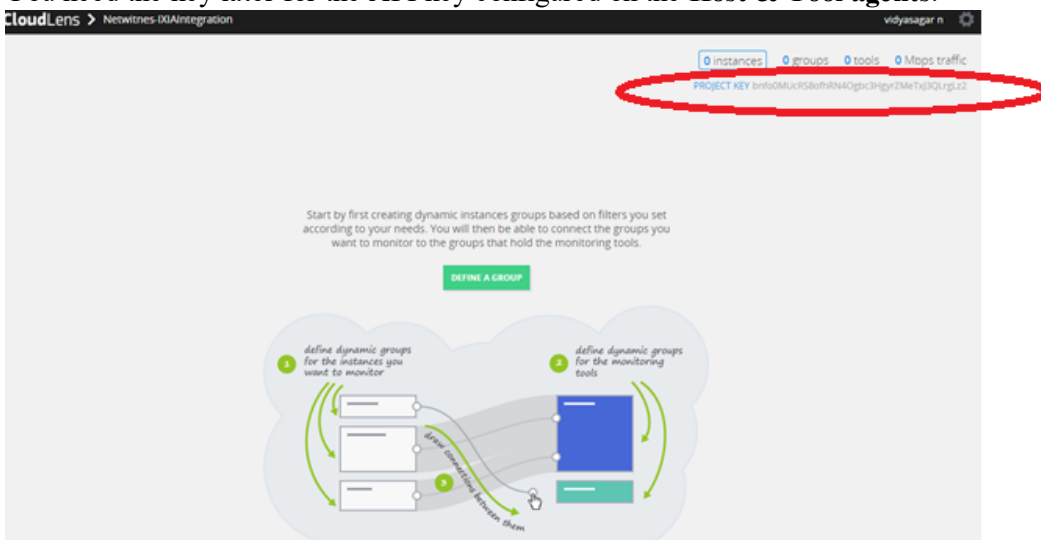


## Task 3. Install Docker Container on Decoder

Complete the following steps to install the Docker container onto the Network Decoder.

1. SSH to the Network Decoder.

2. Enter the following commands to complete the install the Docker service on the Decoder.
```
#yum clean all
# yum -y install docker
```

3. Enter the following command string to start the Docker service.
```
# service docker start
```

4. Enter the following commands to:

- Access the Ixia repository and obtain the cloudlens-agent container.

- Replace the **ProjectKeyFromIxiaProjectPortal** variable, which identifies your project key in Ixia portal, with the Project Key you created in Task 2. Create CloudLens Project.

```
sudo docker run \
--name cloudlens \
 -v /:/host \
-v /var/run/docker.sock:/var/run/docker.sock \
-d --restart=always \
--net=host \
--privileged \
ixiacom/cloudlens-agent:latest \
--server agent.ixia.cloud \
--accept_eula y \
--apikey ProjectKeyFromIxiaProjectPortal \
```

## Task 4. Install Docker Container on Clients

Complete the follow steps to Y install the Docker Container onto the client machines for which you want to route the traffic to the Network Decoder.

1. SSH to the AWS Client instance.

2. Enable root access to OS CLI (for example `sudo su -`).

3. Enter the following commands to install Docker.
```
# yum -y install docker
```

> **Caution:** The above example of the installed docker engine is for CentOS7. The instructions may vary slightly for different Linux Distributions. For more information, see the Docker docs at https://docs.docker/install.

4. Enter the following commands to start the Docker service.
```
# service docker start
```

5. Enter the following commands to:

- Access the Ixia repository and obtain the **cloudlens-agent** container.

- Replace the variable **ProjectKeyFromIxiaProjectPortal**, which identifies your project key in Ixia portal, with the Project Key you created in the previous section.
```
sudo docker run \
--name cloudlens \
 -v /:/host \
-v /var/run/docker.sock:/var/run/docker.sock \
-d --restart=always \
```

```
--net=host \
--privileged \
ixiacom/cloudlens-agent:latest \
--server agent.ixia.cloud \
--accept_eula y \
--apikey ProjectKeyFromIxiaProjectPortal \
```

> **Warning:** If you cut and paste commands from a PDF, first paste them into a test editor such as Notepad to confirm the syntax before pasting into the OS CLI. Direct cut and paste between PDF and CLI can contain dashes or other special characters that should not be part of the commands.

## Task 5. Map the Network Decoder to Ixia Clients

Complete the following steps to map the Network Decoder to the client machines to route the traffic to the Network Decoder.

1. Go to the Cloudlens public site (https://www.ixia.cloud).

2. Double-click on your project to open it.

3. Click the **Define Group** button or the Instances count.
   You should see two instances listed, one for your decoder and the other for the client machines.

4. Filter for the decoder instance and click **Save Search**.

5. Choose **Save as a tool**.

6. Specify a name for the tool, and the **Aggregation Interface**.
   Use a meaningful name for the Aggregation Interface (for example **cloudlens0**. This is a virtual interface that appears in the OS where your Tool is installed. You need to instruct your tool to

'listen' to that interface in a subsequent step.
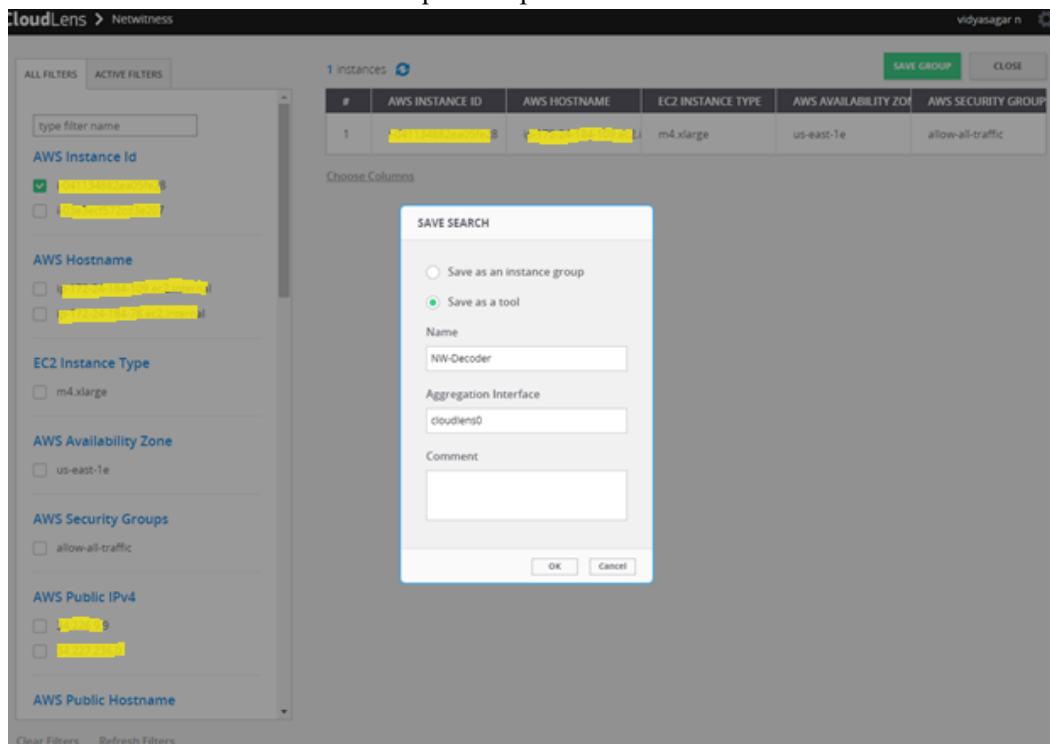


7. Filter the client host instance from the list, and click **Save Search**.
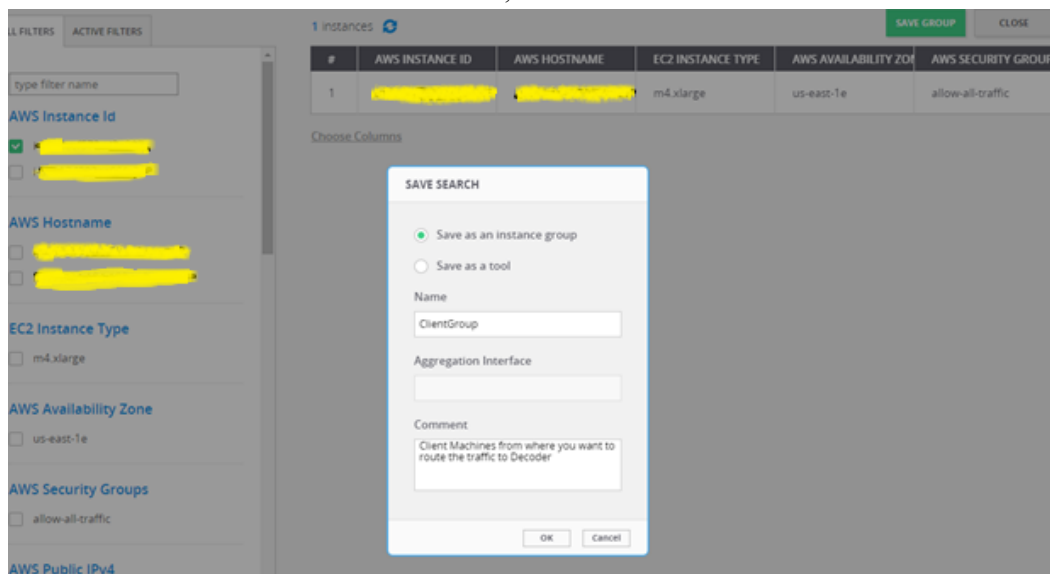


8. Navigate back to the top-level view of the project.
   Your client machine instance and Decoder instance are now displayed.

9. Drag a connection between the your client machine instance and Decoder instance to allow the flow of packets.
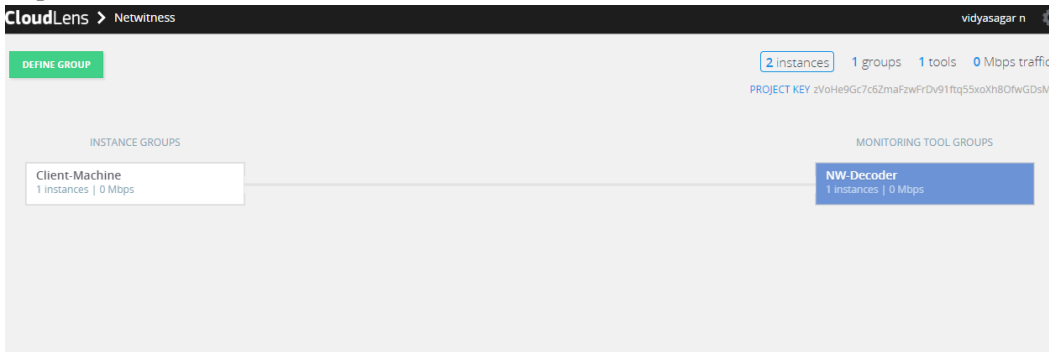


## Task 6. Validate CloudLens Packets Arriving at Decoder

Complete the following steps to validate that packets are actually arriving at the Network Decoder.

1. SSH to the Network Decoder.

2. Enter the following command.

   ```
   ifconfig
   ```

   The new aggregation interface you created is displayed.



3. Generate traffic from the client OS instance CLI (for example, `wget http://www.google.com/`).



4. SSH to Network Decoder to go to your Network Decoder instance CLI.

5. Enter the following commands to look for suitable results in the tcpdump.
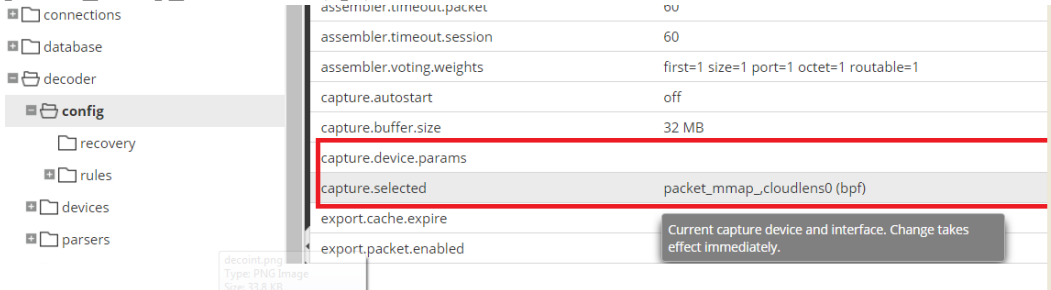
```
tcpdump -I Cloudlens0
```

```
74 packets dropped by kernel
root@ip-172-24-... ~]# tcpdump -i cloudlens0
tcpdump: WARNING: cloudlens0: no IPv4 address assigned
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on cloudlens0, link-type EN10MB (Ethernet), capture size 65535 bytes



4:37:11.408308 IP 175.2.141.156 > ip-172-24-18-... ec2.internal: ICMP echo request, id 132, seq 32849, length 8
4:37:11.408318 IP ip-172-24-18... ec2.internal > 175.2.141.156: ICMP echo reply, id 132, seq 32849, length 8
4:37:11.781923 IP 175.2.141.156 > ip-172-24-18-...ec2.internal: ICMP 175.2.141.156 protocol 1 unreachable, length 36
```

## Task 7. Set the Interface in the Network Decoder

Complete the following steps in the Network Decoder to set the interface to use for the Ixia integration.

1. SSH to the Network Decoder.

2. Enter the following commands to restart decoder service.
   ```
   $ sudo restart nwdecoder
   ```
   The Network Decoder is now set to capture network traffic.

3. Log in to NetWitness Platform and click ✂ **(Admin) > Services**.

4. Select a Decoder service and click ⚙ ⌄ > **View > Explore**.

5. Expand the **decoder** node and click **config** to view the configuration settings.

6. Set the **capture.selected** parameter to the following value.

   packet_mmap_,cloudlens0(bpf)

| | |
|---|---|
| ▣☐ connections | assembler.timeout.packet | 60 |
| ▣☐ database | assembler.timeout.session | 60 |
| ▣🗀 decoder | assembler.voting.weights | first=1 size=1 port=1 octet=1 routable=1 |
| ▣🗀 config | capture.autostart | off |
| 🗀 recovery | capture.buffer.size | 32 MB |
| ▣🗀 rules | capture.device.params | |
| ▣☐ devices | capture.selected | packet_mmap_,cloudlens0 (bpf) |
| ▣☐ parsers | export.cache.expire | |
| | export.packet.enabled | |

Current capture device and interface. Change takes effect immediately.

7. (Conditional) - If you have multiple capture interfaces on the Network Decoder, set the parameters with the following values.
   ```
   capture.device.params --> interfaces=cloudlens0,cloudlens1
   ```

```
capture.selected --> packet_mmap_,All
```
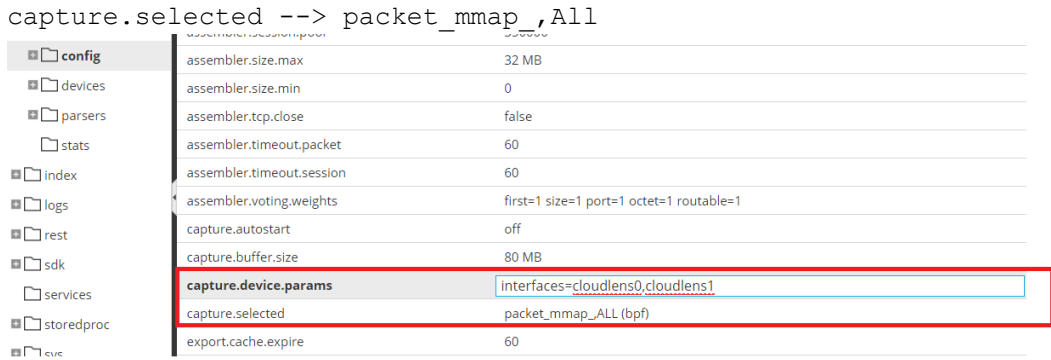


8. Restart the Decoder service after you set the **capture.selected** parameter.

# Integrate f5® BIG-IP with the Network Decoder

IG-IP Virtual Edition (VE) is an inline virtual server and load balancer. A common use case would be for the f5® box to be a virtual web server that presents a single IP address / host name that manages requests to a pool of web servers in the cloud.

All traffic to RSA NetWitness® Platform flows through the f5® BIG-IP VE virtual server.

The virtual server functions of the BIG-IP clone all traffic to a designated computer by re-writing mac addresses and loading them into a subnet shared with the destination sniffer. This guide describes how to set up the Decoder as the sniffer.

## f5® BIG-IP VE Deployment Information

f5® BIG-IP VE on AWS will be available through the AWS Marketplace and activated by a BYOL license. A thirty-day free trial is also available.

For more information on this solution refer to the f5® BIG-IP DNS Data Sheet (https://www.f5.com/pdf/products/big-ip-dns-datasheet.pdf).

## Task 1: Set Up a BIG-IP VE Virtual Server Instance

Set up a BIG-IP VE Virtual Server Instance according to the instructions in the "BIG-IP Virtual Edition 12.1.0 and Amazon Web Services: Multi-NIC Manual" ( https://support.f5.com/kb/en-us/products/big-ip_ltm/manuals/product/bigip-ve-multi-nic-setup-amazon-ec2-12-1-0.html). Complete all the steps through the last steps, "Creating a virtual server."

This virtual server performs packet capture. You may need to create multiple virtual servers to depending on your volume.

As part of creating the virtual server, you must have at least one server in your NetWitness Platform domain to handle the traffic routed by the virtual server (for example, you can create another instance in AWS to host the internal server).

## Task 2: Create a Clone Pool

1. Make sure that your Decoder has a network interface on the same subnet as one of the network interfaces on the BIG-IP VE instance.
   The clone pool sends packets to the Decoder by rewriting MAC addresses and sending them out a

---

network interface. MAC address rewriting can be used to route packets to another subnet.

2. Set up the clone pool within the BIG-IP VE virtual server according to the instructions in "K13392: Configuring the BIG-IP system to send traffic to an intrusion detection system (11.x - 13.x)" article (https://support.f5.com/kb/en-us/solutions/public/13000/300/sol13392.html).
This document explains how to create the clone pool, and how to make an existing virtual server copy traffic to the clone pool. In this case, we will place the Decoder instance in the clone pool.

## Guidelines

The following guidelines will help you to configure packet capture correctly using BIG-IP VE.

- The Decoder instance must have its own IP address on one of the same subnets as BIG-IP VE. BIG-IP uses that IP address to identify the Decoder as being part of the clone pool.

- When adding the Decoder instance to the clone pool, BIG-IP asks for a port number in addition to the IP address. This port number does not matter for the cloned traffic. The Decoder will receive all the cloned traffic, regardless of what port number was used here.

- By default, the AWS subnet shared by the Decoder and BIG-IP VE will not allow the cloned traffic to travel from the BIG-IP VE interface to the Decoder interface. You must disable the `source/dest. check` on both the Decoder and BIG-IP VE network interfaces in AWS.

- The Decoder instance must have a single network interface, eth0, by default. The Decoder captures traffic on this interface, but it may also receive administrative traffic on this interface. RSA recommends using network rules to filter out ssh and nwdecoder traffic from the capture stream. These are ports 22 (ssh) and 50004/56004 (nwdecoder).

## Troubleshooting Tips

There are areas to troubleshoot if packets are not being accepted by the Decoder.

- Make sure that the BIG-IP VE is sending the packets out of the correct interface.
The BIG-IP VE instance contains tcpdump. Use it to verify the cloned packets are being sent out the expected interface. If they are not, there is a problem in the setup of the clone pool or the virtual server.

- Make sure that the Decoder is receiving packets.
The Decoder has `tcpdump` installed on it. Use it to verify that the Decoder is receiving packets. If the Decoder is not capturing packets, make sure that

  - The AWS `source/dest. check` is turned off.

  - The Decoder is on the same subnet as the interface the BIG-IP VE is using to clone packets.

## Integrate VPC Traffic Mirroring with the Network Decoder

VPC Traffic Mirroring allows users to capture and inspect network traffic to analyze packets without using any third-party packet forwarding agents. The solution provides insight and access to network traffic across VPC infrastructure. Users can copy network traffic at any ENI (Elastic Network Interfaces) in VPC, and send it to NetWitness Platform to analyze, monitor, and troubleshoot performance issues.

You must complete the following tasks to integrate the Network Decoder with VPC Traffic Mirroring:

Task 1. Configure the Network Decoder as a VPC Traffic Mirroring Destination

Task 2. Configure a VPC Traffic Mirroring Filter

Task 3. Configure a VPC Traffic Mirroring Session

Task 4. Setup a new VXLAN interface on the Network Decoder

Task 5. Validate VPC Traffic Mirroring Packets Arriving at Network Decoder

## Task 1. Configure the Network Decoder as a VPC Traffic Mirroring Destination.

1. Open the VPC service console view at https://console.aws.amazon.com/vpc/home.

2. In the navigation panel, select **Traffic Mirroring**.

3. Select **Mirror Targets**.

## Task 2. Configure a VPC Traffic Mirroring Filter

You must configure a VPC Traffic Mirroring Filter to send only the required packets to the Network Decoder. You can determine if the inbound or outbound traffic needs to be captured or not.

> **Note:** Make sure the UDP port 4789 is open on the AWS instance of Network Decoder.



## Task 3. Configure a VPC Traffic Mirroring Session

You must configure a VPC Traffic Mirroring Session to mirror the traffic by a communication channel between source ENI and destination ENI.

## Task 4. Set Up a new VXLAN Interface on the Network Decoder

To capture the UDP enabled traffic you must create an interface and tunnel it to Network Decoder by performing the following steps.

1.  SSH to the Decoder.

2.  Enter the following commands.

    ```
    sudo ip link add tun0 type vxlan id <VXNLAN ID> local any dev <primary
    interface ex: eth0> dstport 4789

    sudo ip link set tun0 up

    ifconfig
    ```

```
tun0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 8951
        inet6 fe80::bc58:afff:fe06:ec29  prefixlen 64  scopeid 0x20<link>
        ether be:58:af:06:ec:29  txqueuelen 1000  (Ethernet)
        RX packets 989  bytes 74140 (72.4 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 8 overruns 0  carrier 0  collisions 0
```

3. To create a firewall rule in the Network Decoder to allow traffic through the tunnel.

   a. Open the IP tables file using the command `vi /etc/sysconfig/iptables`.

   b. Append the line `-I INPUT -p udp -m udp --dport 4789 -j ACCEPT`.

   c. Restart IP tables by using the following commands.
   ```
   service iptables restart
   service iptables status
   ```

4. To set the interface in the Network Decoder.

   a. Log in to NetWitness Platform, select the `decoder/config` node in Explorer view of the Network Decoder service.

   b. Set the `capture.selected = packet_mmap_,tun0` parameter.
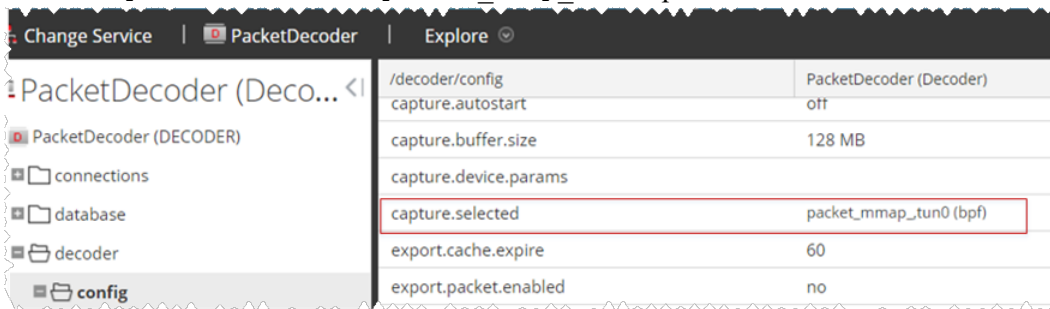


5. (Conditional) If you have multiple tunnels on the Network Decoder.

   a. Restart the Decoder service after you create the tunnel in Network Decoder.

   b. Log in to NetWitness Platform, select the `decoder/config` node in Explorer view of the Network Decoder service, and set the following parameters.

   `capture.device.params = interfaces=tun0,tun1,tun2`

   `capture.selected = packet_mmap_,All`
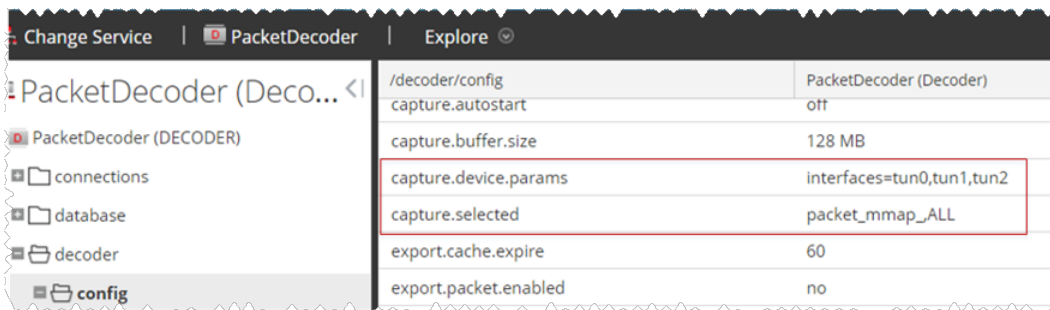
6. Restart the Decoder service.

```
$ sudo restart nwdecoder
```

The user should be all set to capture the network traffic in the Network Decoder.

## Task 5. Validate VPC Traffic Mirroring Packets Arriving at the Network Decoder

Perform the following steps to validate if the Network Decoder is receiving the network data (packets) successfully.

1. Generate traffic from the client OS instance CLI (for example, `wget` `http://www.google.com/`).



2. Enter the `tcpdump -i tun0` command to look for suitable results in the tcpdump.



3. The NetWitness Platform reflects meta values as shown below.

**Note:** You can mirror traffic from an EC2 instance that is supported by the AWS Nitro system (A1, C5, C5d, C5n, I3en, M5, M5a, M5ad, M5d, p3dn.24xlarge, R5, R5a, R5ad, R5d, T3, T3a, and z1d).

**Note:** For more information, see "New – VPC Traffic Mirroring" documentation at https://aws.amazon.com/blogs/aws/new-vpc-traffic-mirroring/.

# AWS Instance Configuration Recommendations

> **Note:** These recommendations can be used as a baseline for 11.5.0.0 and adjusted as needed.

This topic contains the minimum AWS instance configuration settings recommended for the RSA NetWitness® Platform virtual stack components.

- EC2 Instance:

  - Instance type adjustments - you must adjust instance types according to your ingestion rate, content and parsers, dashboard reports, scheduled reports, investigations, and active users.

  - Recommended settings - the recommended settings in the NW component instance tables below were calculated under the following conditions.

    - Ingestion rates of 15,000 EPS and 1.5 Gbps were used.

    - All the components were integrated.

    - The Log stream includes a Log Decoder, Concentrator, and Archiver.

    - The Packet stream includes a Network Decoder and Concentrator.

    - The Endpoint Hybrid stream includes a Endpoint Server, Concentrator and Log Decoder.

    - Respond is receiving alerts from the Reporting Engine and Event Stream Analysis.

    - The background load includes reports, charts, alerts, investigation, and respond.

- Block Storage

  For more information on the required volumes and the storage allocations, see the Storage Guide for RSA NetWitness® Platform 11.x.

## Archiver

| EC2 Instance | | | |
|---|---|---|---|
| **EPS** | **Instance Type** | **Enhanced Networking Enabled** | **Tenancy Type - Dedicated - Run a Dedicated Instance** |
| 5,000 | m4.xlarge<br>No of CPU: 4<br>Memory: 16 GB | No | Yes |
| 10,000 | m4.2xlarge<br>No of CPU: 8<br>Memory: 32 GB | No | Yes |
| 15,000 | m4.4xlarge<br>No of CPU: 16<br>Memory: 64 GB | No | Yes |

| Cloud Provider Block Storage | | | |
|---|---|---|---|
| **Volumes** | **Device** | **Volume Type** | **IOPS/Baseline Throughput** |
| / (root) | /dev/sda1 | General Purpose SSD | N/A |
| usr,var,opt,home,tmp | /dev/sdf | General Purpose SSD | N/A |
| archiver | /dev/sdg | Throughput Optimized HDD | 240 MB/s |
| workbench | /dev/sdh | Throughput Optimized HDD | N/A |

# Broker

| EC2 Instance | | |
|---|---|---|
| **Instance Type** | **Enhanced Networking Enabled** | **Tenancy Type - Dedicated - Run a Dedicated Instance** |
| m4.xlarge No of CPU: 4 Memory: 16 GB | No | Yes |

| Cloud Provider Block Storage | | | |
|---|---|---|---|
| **Volumes** | **Device** | **Volume Type** | **IOPS/Baseline Throughput** |
| / (root) | /dev/sda1 | General Purpose SSD | N/A |
| usr,var,opt,home,tmp | /dev/sdf | General Purpose SSD | N/A |
| broker | /dev/sdg | General Purpose SSD | N/A |

# Concentrator - Log Stream

| EC2 Instance | | | |
|---|---|---|---|
| **EPS** | **Instance Type** | **Enhanced Networking Enabled** | **Tenancy Type - Dedicated - Run a Dedicated Instance** |
| 5,000 | m4.xlarge<br>No of CPU: 4<br>Memory: 16 GB | No | Yes |
| 10,000 | m4.2xlarge<br>No of CPU: 8<br>Memory: 32 GB | No | Yes |
| 15,000 | m4.4xlarge<br>No of CPU: 16<br>Memory: 64 GB | No | Yes |

| Cloud Provider Block Storage | | | |
|---|---|---|---|
| **Volumes** | **Device** | **Volume Type** | **IOPS/Baseline Throughput** |
| / (root) | /dev/sda1 | General Purpose SSD | N/A |
| usr,var,opt,home,tmp | /dev/sdf | General Purpose SSD | N/A |
| index | /dev/sdg | Provisioned IOPS | 10,000 |
| session, metadb | /dev/sdh | Throughput Optimized HDD | 240 MB/s |

# Packet Stream Solutions

## Concentrator - Gigamon Solution

| EC2 Instance | | | |
| --- | --- | --- | --- |
| Mbps/Gbps | Instance Type | Enhanced Networking Enabled | Tenancy Type - Dedicated - Run a Dedicated Instance |
| 500 Mbps | c4.4xlarge<br>No of CPU: 16<br>Memory: 30 GB | No | Yes |
| 1,000 Mbps | c4.8xlarge<br>No of CPU: 36<br>Memory: 60 GB | No | Yes |
| 1.5 Gbps | m4.10xlarge<br>No of CPU: 40<br>Memory: 160 GB | No | Yes |

## Concentrator - f5 BIG-IP Solution

To be updated when f5 BIG-IP performance testing is complete.

| EC2 Instance | | | |
| --- | --- | --- | --- |
| Mbps/Gbps | Instance Type | Enhanced Networking Enabled | Tenancy Type - Dedicated - Run a Dedicated Instance |
| 230 Mbps | m4.4xlarge<br>No. of CPU: 16<br>Memory: 64 GB | No | No |

| Cloud Provider Block Storage | | | |
|---|---|---|---|
| **Volumes** | **Device** | **Volume Type** | **IOPS/Baseline Throughput** |
| / (root) | /dev/sda1 | General Purpose SSD | N/A |
| usr,var,opt,home,tmp | /dev/sdf | General Purpose SSD | N/A |
| index | /dev/sdg | Provisioned IOPS | 15,000 |
| session, metadb | /dev/sdh | Throughput Optimized HDD | 240 MB/s |

## Decoder - Gigamon Solution

| EC2 Instance | | | |
|---|---|---|---|
| **Mbps/Gbps** | **Instance Type** | **Enhanced Networking Enabled** | **Tenancy Type - Dedicated - Run a Dedicated Instance** |
| 500 Mbps | c4.2xlarge<br>No of CPU: 8<br>Memory: 15 GB | Yes | Yes |
| 1000 Mbps | c4.4xlarge<br>No of CPU: 16<br>Memory: 30 GB | Yes | Yes |
| 1.5 Gbps | c4.8xlarge<br>No of CPU: 36<br>Memory: 60 GB | Yes | Yes |

## Decoder - f5 BIG-IP Solution

To be updated when f5 BIG-IP performance testing is complete.

| EC2 Instance | | | |
|---|---|---|---|
| **Mbps/Gbps** | **Instance Type** | **Enhanced Networking Enabled** | **Tenancy Type - Dedicated - Run a Dedicated Instance** |
| 230 Mbps | m4.4xlarge No. of CPU: 16 Memory: 64 GB | No | No |

| Cloud Provider Block Storage | | | |
|---|---|---|---|
| **Volumes** | **Device** | **Volume Type** | **IOPS/Baseline Throughput** |
| / (root) | /dev/sda1 | General Purpose SSD | N/A |
| usr,var,opt,home,tmp | /dev/sdf | General Purpose SSD | N/A |
| index,session,meta | /dev/sdg | Throughput Optimized HDD | 240 MB/s |
| packet | /dev/sdh | Throughput Optimized HDD | 240 MB/s |

# ESA and Context Hub on Mongo Database

| EPS | EC2 Instance | | |
| --- | --- | --- | --- |
| | Instance Type | Enhanced Networking Enabled | Tenancy Type - Dedicated - Run a Dedicated Instance |
| 9,000 | m4.2xlarge No of CPU: 8 Memory: 32 GB | No | Yes |
| 18,000 | r4.2xlarge No of CPU: 8 Memory: 61 GB | No | Yes |
| 30,000 Aggregation Rate | r4.4xlarge No of CPU: 16 Memory: 122 GB | No | Yes |

| Cloud Provider Block Storage | | | |
| --- | --- | --- | --- |
| Volumes | Device | Volume Type | IOPS/Baseline Throughput |
| / (root) | /dev/sda1 | General Purpose SSD | N/A |
| usr,var,opt,home,tmp | /dev/sdf | General Purpose SSD | N/A |
| apps (/opt/rsa) | /dev/sdg | General Purpose SSD | N/A |

# Log Collector (Syslog, Netflow, and File Collection Protocols)

| EC2 Instance | | | |
|---|---|---|---|
| EPS | Instance Type | Enhanced Networking Enabled | Tenancy Type - Dedicated - Run a Dedicated Instance |
| 30,000 NON SSL | c4.2xlarge<br>No of CPU: 8<br>Memory: 15 GB | No | Yes |

| Cloud Provider Block Storage | | | |
|---|---|---|---|
| Volumes | Device | Volume Type | IOPS/Baseline Throughput |
| / (root) | /dev/sda1 | General Purpose SSD | N/A |
| usr,var,opt,home,tmp | /dev/sdf | General Purpose SSD | N/A |
| logcollector | /dev/sdg | General Purpose SSD | N/A |

# Log Decoder

| EC2 Instance | | | |
|---|---|---|---|
| EPS | Instance Type | Enhanced Networking Enabled | Tenancy Type - Dedicated - Run a Dedicated Instance |
| 5,000 | c4.2xlarge<br>No of CPU: 8<br>Memory: 15 GB | Yes | Yes |
| 10,000 | c4.4xlarge<br>No of CPU: 16<br>Memory :30 GB | Yes | Yes |
| 15,000 | c4.8xlarge<br>No of CPU: 36<br>Memory: 60GB | Yes | Yes |

| Cloud Provider Block Storage | | | |
|---|---|---|---|
| Volumes | Device | Volume Type | IOPS/Baseline Throughput |
| / (root) | /dev/sda1 | General Purpose SSD | N/A |
| usr,var,opt,home,tmp | /dev/sdf | General Purpose SSD | N/A |
| index,session,meta | /dev/sdg | Throughput Optimized HDD | 240 MB/s |
| packet | /dev/sdh | Throughput Optimized HDD | 240 MB/s |

# NW Server, Reporting Engine, Respond and Health & Wellness

| EC2 Instance | | |
|---|---|---|
| **Instance Type** | **Enhanced Networking Enabled** | **Tenancy Type - Dedicated - Run a Dedicated Instance** |
| m4.2xlarge<br>No of CPU: 8<br>Memory: 32 GB | No | Yes |
| m4.4xlarge<br>No of CPU: 16<br>Memory: 64 GB | No | Yes |

| Cloud Provider Block Storage | | | |
|---|---|---|---|
| **Volumes** | **Device** | **Volume Type** | **IOPS/Baseline Throughput** |
| / (root) | /dev/sda1 | General Purpose SSD | N/A |
| usr,var,opt,home,tmp | /dev/sdf | General Purpose SSD | N/A |
| uax,ipdb | /dev/sdg | General Purpose SSD | N/A |
| redb,rehome | /dev/sdh | General Purpose SSD | N/A |

# NetWitness Endpoint Hybrid

| EC2 Instance | | | |
|---|---|---|---|
| **Agents** | **Instance Type** | **Enhanced Networking Enabled** | **Tenancy Type - Dedicated - Run a Dedicated Instance** |
| 15,000 agents | m4.10xlarge No of CPU: 40 Memory: 160 GB RAM | Yes | Yes |

| Cloud Provider Block Storage | | | |
|---|---|---|---|
| **Volumes** | **Device** | **Volume Type** | **IOPS/Baseline Throughput** |
| / (root) | /dev/sda1 | General Purpose SSD | N/A |
| usr,var,opt,home,tmp | /dev/sdf | General Purpose SSD | N/A |
| index,session,meta (Log Decoder) | /dev/sdg | Throughput Optimized HDD | 240 MB/s |
| packet (Log Decoder) | /dev/sdh | Throughput Optimized HDD | 240 MB/s |
| index (Concentrator) | /dev/sdi | Provisioned IOPS | 10,000 |
| session,meta (Concentrator) | /dev/sdj | Throughput Optimized HDD | 240 MB/s |
| mongoDB | /dev/sdl | Throughput Optimized HDD | 240 MB/s |

# Appendix A. Silent Installation Using CLI

You can use the following Command Line Interface commands to run the installation script (`nwsetup-tui`) without getting prompted for inputs. This enables you to automate the installation of a host by supplying response to the scripts prompts through the command line.

1. After you have created a base image on the host, log in to the host with the `root` credentials.

2. Submit the `nwsetup-tui` script with the `--silent` command and the arguments that you want to apply.

   The following command string is an example of how you would install a basic NW Server host.

   ```
   nwsetup-tui --silent --is-head=true --host-name=new-host --master-
   pass=netwitness --deploy-pass=netwitness --repo-type=1 --custom-
   firewall=false --ip-override=false --eula=true
   ```

3. (Conditional - For Component Hosts Only) Install the appropriate service **Category** on the newly provisioned host in the NetWitness Platform Hosts view.

   a. Log into NetWitness Platform and go to  (**Admin**) > **Hosts**.

      The **New Hosts** dialog is displayed with the **Hosts** view grayed out in the background

      > **Note:** If the **New Hosts** dialog is not displayed, click **Discover** in the Hosts view toolbar.

   b. Select the host in the **New Hosts** dialog and click **Enable**.

      The **New Hosts** dialog closes and the host is displayed in the **Hosts** view.

   c. Select that host in the **Hosts** view (for example, **Event Stream Analysis**) and click .

      The **Install Services** dialog is displayed.

   d. Select the appropriate host type in **Category** and click **Install**.

## Arguments

| Argument | Description |
|---|---|
| `--help-install-opts` | Display all the arguments in this table. |
| `--eula` | Accept or decline the End User License Agreement (EULA). Specify: <br><br> • `true` (default) to accept the agreement <br><br> • `false` to decline it and cancel the installation. <br><br> For example: `--eula=true` |

| Argument | Description |
|---|---|
| --is-head | Designate the host as the NW Server host or a component host. Specify:<br><br>• `true` for NW Server host.<br><br>• `false` for Component host.<br><br>For example: `--is-head=true` |
| --host-name | Specify new hostname. If you do not specify this argument, NetWitness Platform retains the existing hostname.<br><br>For example: `--host-name=<hostname>` |
| --master-pass | Enter master password. For example: `--master-pass=<password>` |
| --deploy-pass | Enter deployment password. For example: `--deploy-pass=<password>` |
| --iface-name | Specify network interface.<br><br>For example: `--iface-name=eth0` |
| --ip-override | Accept or override IP address found for this host or change the IP configuration found on the host. Specify:<br><br>• `true` provide IP address.<br><br>• `false` use IP address found on the host.<br><br>For example: `--ip-override=false` |
| --ip-type | Select ip address configuration type. Specify:<br><br>• `1` Static IP Configuration)<br><br>• `2` DCHP<br><br>For example: `--ip-type=1` |
| --ip-addr | For Static IP configuration, enter IP Address for static address.<br><br>For example: `--ip-addr=<ip-address>` |
| --ip-netmask | For Static IP configuration, enter Subnet Mask for static address. For example: `--ip-gateway=<subnet-mask>` |
| --ip-gateway | For Static IP configuration, enter default gateway for static address. For example: `--ip-gateway=<default-gateway>` |
| --ip-nameserver | IP address assigned to DNS server. `--ip-nameserver=<ip-address>` |

| Argument | Description |
|---|---|
| `--ip-nameserver-secondary` | Optional - IP address assigned to a secondary DNS server. For example: `--ip-nameserver-secondary=<ip-address>` |
| `--ip-domain` | For Static IP configuration, enter Local Domain Name for static address. For example: `--ip-domain=<default-gateway>` |
| `--repo-type` | Select type of update repository. Specify:<br><br>• `1` Local repository<br><br>• `2` External repository<br><br>For example: `--repo-type=1` |
| `--repo-url` | For an external update repository, specify the url of the repository. For example: `--repo-url=<url>` |
| `--head-ip` | For a component host, specify IP Address of the NW Server.<br><br>For example: `--head-ip=<ip-address>` |
| `--custom-firewall` | Disable default firewall configuration and use your custom configuration. Specify:<br><br>• `true` use custom firewall configuration.<br><br>• `false` use default firewall configuration.<br><br>For example: `--custom-firewall=true` |
| `--use-nat` | Configure the host to use Network Address Translation (NAT) based IP addresses:<br><br>• `true` use NAT IPs to connect to other hosts<br><br>• `false` do not use NAT IPs to connect to other hosts (default)<br><br>For example: `--use-nat=false` |