DANOS-Vyatta edition

**Disaggregated Network
Operating System
Version 2009a**

VMware Installation Guide
October 2020

Contents

Chapter 1. Copyright Statement

© 2020 IP Infusion Inc. All Rights Reserved.

This documentation is subject to change without notice. The software described in this document and this documentation are furnished under a license agreement or nondisclosure agreement. The software and documentation may be used or copied only in accordance with the terms of the applicable agreement. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or any means electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's internal use without the written permission of IP Infusion Inc.

IP Infusion Inc.
3965 Freedom Circle, Suite 200
Santa Clara, CA 95054
+1 408-400-1900

http://www.ipinfusion.com/.

For support, questions, or comments via E-mail, contact:

support@ipinfusion.com.

Trademarks:

IP Infusion is a trademark of IP Infusion. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners.

Use of certain software included in this equipment is subject to the IP Infusion, Inc. End User License Agreement at http://www.ipinfusion.com/license. By using the equipment, you accept the terms of the End User License Agreement.

Chapter 2. Preface

Document conventions

The document conventions describe text formatting conventions, command syntax conventions, and important notice formats used in this document.

Notes, cautions, and warnings

Notes, cautions, and warning statements may be used in this document. They are listed in the order of increasing severity of potential

hazards.

Note: A Note provides a tip, guidance, or advice, emphasizes important information, or provides a reference to related information.

Attention: An Attention statement indicates a stronger note, for example, to alert you when traffic might be interrupted or the device might reboot.

CAUTION: A Caution statement alerts you to situations that can be potentially hazardous to you or cause damage to hardware, firmware, software, or data.

DANGER: A Danger statement indicates conditions or situations that can be potentially lethal or extremely hazardous to you. Safety labels are also attached directly to products to warn of these conditions or situations.

Text formatting conventions

Text formatting conventions such as boldface, italic, or Courier font are used to highlight specific words or phrases.

Format	Description
bold text	Identifies command names. Identifies keywords and operands.
italic text	Identifies emphasis. Identifies variables. Identifies document titles.
`Courier font`	Identifies CLI output. Identifies command syntax examples.

Command syntax conventions

Bold and italic text identify command syntax components. Delimiters and operators define groupings of parameters and their logical relationships.

Convention	Description
bold text	Identifies command names, keywords, and command options.
italic text	Identifies a variable.
[ ]	Syntax components displayed within square brackets are optional. Default responses to system prompts are enclosed in square brackets.
{ **x** | **y** | **z** }	A choice of required parameters is enclosed in curly brackets separated by vertical bars. You must select one of the options.
x | **y**	A vertical bar separates mutually exclusive elements.
< >	Nonprinting characters, for example, passwords, are enclosed in angle brackets.
...	Repeat the previous element, for example, *member*[*member*...].
\	Indicates a "soft" line break in command examples. If a backslash separates two lines of a command input, enter the entire command at the prompt without the backslash.

Chapter 3. About This Guide

This guide describes how to install and upgrade the DANOS-Vyatta edition on VMware.

Chapter 4. Installing the System

Overview of installing on VMware ESX and ESXi

Like other virtualization platforms, VMware products provide the ability to run multiple virtual systems on a single hardware platform. The router can be run on VMware ESX and ESXi. VMware ESX and ESXi are virtualization platforms that run directly on system hardware in a 64-bit environment.

IPI provides a prebuilt VMware template that can be used to create router virtual machines on VMware ESX or ESXi. IPI also provides an ISO that can be installed on VMware ESX or ESXi.

The router can also run in an OpenStack cloud with VMware vSphere as the hypervisor.

VMware feature restrictions

This section describes a restricted vRouter feature, VRRP in RFC-Compliant mode, that does not function in the VMware environment.

> 📝 **Note:**

VRouter features that do not work in the VMware environment include the behavior of RFC-compliant MAC addresses for VRRP interfaces.

To prevent RFC compatibility issues when running a vRouter on a VMware platform, by default, RFC compliant MAC address behavior is not enabled on the vRouter. To enable MAC address RFC-compliance, you must run the `set interfaces <interface> vrrp vrrp-group <group-id> rfc-compatibility` command.

The default behavior of non-RFC-compliant MAC addresses is required for any deployment in which VMware provides Layer 2 services, because vSwitch does not support true MAC learning and blocks traffic when MAC addresses move between the ports of an L2 switch. In addition, a key component of RFC-compliant VRRP is that MAC addresses must be able to freely move between routers participating in a group. For more information about MAC support on VMware, refer to VMware documentation.

Preparing for installation

The VM install image is the simplest way to get a router up and running quickly. If the parameters of the VM install image do not meet your requirements or additional settings are required, then the router can be installed from the ISO.

Before installing, prepare for the installation.

- Download the router VMware files from the site provided to you.

- Install VMware on a server. The installation procedures assume that the VMware system is operational. VMware ESX or ESXi and accompanying documentation can be obtained from VMware at http://www.vmware.com.
- Ensure that you have a minimum of 10 GB of free space on the system for a root partition. A minimum of 4 GB of free space is recommended for a production installation.
- Ensure that you have a minimum of 4 GB of RAM on the system. A minimum of 4 GB is recommended.
- The data plane supports from 2 to 128 CPUs. At least 4 CPUs are recommended for optimum performance.

Installing the router ISO for VMware

Follow the procedure in this section to install the router ISO on VMware ESX or ESXi.

Before installing the router ISO on VMware ESX or ESXi, download the router ISO file. Ensure the file is saved to a location that is accessible from the virtual machine you create in the following procedure. Refer to Downloading the router files for VMware for more information.

To install the ISO on VMware ESX or ESXi:

1. Create a Debian GNU/Linux 6 (64-bit) virtual machine.
2. Point the DVD drive on the virtual machine to the downloaded ISO.
3. Set up the virtual machine so that it boots from the DVD. Ensure the virtual DVD drive is set to connect at bootup.
4. Reboot the virtual machine. The virtual machine boots the ISO.
5. Log on to the virtual machine as the `tmpuser` user and enter the default password of `tmppwd`.
6. At the Vyatta command line, type the `install image` command and follow the prompts to install the system.
7. When the installation process is complete, power down the system by using the `poweroff` command.
8. Reset the virtual DVD so that it no longer points to the ISO.
9. Start the virtual machine.
10. Test the installation.

Testing your installation

After the system has successfully booted, you see the `node login:` prompt. This prompt indicates that the system is operational.

Perform the following procedures:

- Verify the release and system type

• Verify connectivity

Verifying the release and system type

The next step is to confirm that the correct release is running and it is running on the device that you expect.

To verify the release and system type:

1. Log in as the `tmpuser` user. Use the default password of `tmppwd` unless you have changed it.
2. Enter the `show version` command.
   • The `Version:` line shows the version number of the system that is running. Make sure that the displayed result is the correct version.
   • The `System type:` line shows the type of hardware on which the system is running and whether it is in a virtual environment. Make sure that the displayed result is the correct version.
   • The `Boot via:` line shows the type of system that is running.

Verifying connectivity

After you confirm that the correct version is running, you must confirm that the system can be accessed on the local network. Configure an Ethernet interface on the system and ping the interface from another host on the network.

📝 **Note:** In the system, a data plane interface is an abstraction that represents the underlying physical or virtual Ethernet interface of the system. The terms Ethernet interface and data plane interface are synonymous in this guide.

To test the system connectivity, perform the following steps:

1. At the command prompt, enter the commands that are shown in the example, substituting an IP address from your existing subnet. The example uses the following network and IP address.
   • The network is 192.168.1.0/24.
   • The IP address of the interface is 192.168.1.81.

   Make the appropriate substitutions for your network, as shown in the following example.

   ```
   tmpuser@node:~$ configure
   tmpuser@node# set system login user vyatta level superuser
   tmpuser@node# set system login user vyatta authentication plaintext
    vyatta
   tmpuser@node# set interfaces dataplane dp0sN address 192.168.1.81/24
   tmpuser@node# set service ssh
   tmpuser@node# set service telnet
   ```

```
tmpuser@node# commit
```

2. To ping the device from an outside network, add the management "routing-instance" (VRF) configuration below.

```
tmpuser@node# set routing routing-instance management description
 management
tmpuser@node# set routing routing-instance management interface dp0sN
tmpuser@node# set routing routing-instance management protocols static
 route 0.0.0.0/0 next-hop 192.168.1.1
tmpuser@node# set routing routing-instance management service ssh
tmpuser@node# commit
tmpuser@node# exit
tmpuser@node:~$
```

3. From another host on the same subnet, ping the interface to ensure that it is up. From a Linux or Windows command prompt, enter the following command, substituting the IP address you assigned to the interface.

```
ping 192.168.1.81
```

If the system can be reached, you see replies from it in response to the pings. If so, your system is installed and can be accessed on your network.

Chapter 5. Upgrading the System

Release-specific upgrade information

Your system may have special upgrade considerations, depending on the release.

For release-specific upgrade information, and to ensure that configuration information is correctly preserved across upgrades, consult the release notes for your release.

Upgrade overview

The method you use to upgrade your system depends on the way it was installed last time.

- If you installed the system by using the router VMware template, upgrade it by using the instructions given in Upgrading a template-installed system.
- If you installed the system by using the router ISO, upgrade it by using the instructions given in Upgrading an ISO-installed system.

Before upgrading

Before upgrading your system, perform the following tasks.

- Save your existing configuration file for reference. Your configuration file is named `config.boot` and is located in the directory `/config`.
- Make sure you have enough space on your root partition to load the image. You can determine the amount of space available by using the `show system storage` command.

Upgrading the router ISO using "add system image"

When the ISO of the router VMware template changes, perform an upgrade to the ISO by following the procedure in this section.

The `add system image` command uses a router ISO file as the image source. It installs the image with existing images and sets the new image as the default boot image. The new image is run the next time the system is rebooted.

To prepare for the upgrade:

Determine the location of the router ISO file and record the name of the file.

To upgrade the ISO:

1. Enter the `add system image` command. Use the location and name of the router ISO file as arguments in the command, as shown in [Sample session: "add system image"](#).

2. Before you reboot, confirm that the new image is loaded and ready to run the next time the system is rebooted. (Enter the `show system image` command; see [Sample session: "add system image"](#).)

3. Reboot the system by entering the `reboot` command. The system restarts with the new system image.

Sample session: "add system image"

The following example shows a session in which the `add system image` command is used to upgrade to the *xxx*.`iso` system image, where *xxx* is the file name of the ISO image you have downloaded. The following example uses the 3.2R1 image.

Adding a system image

```
tmpuser@vyatta:~$ add system image /home/vyatta/xxx.iso
Checking MD5 checksums of files on the ISO image...OK.
Done!
What would you like to name this image? [3.2R1]:
OK.  This image will be named: 3.2R1
Installing "3.2R1" image.
Copying new release files...
Would you like to save the current configuration
directory and config file? (Yes/No) [Yes]:
Copying current configuration...
Would you like to save the SSH host keys from your
current configuration? (Yes/No) [Yes]:
Copying SSH keys...
Setting up grub configuration...
Done.
tmpuser@vyatta:~$
```

The following example shows how to display installed images.

Displaying system images

```
tmpuser@vyatta:~$ show system image
The system currently has the following image(s) installed:
   1: xxx (default boot)
   2: yyy (running version)
```

Chapter 6. Installation and Upgrade Commands

add system image

Adds a binary system image to the currently running system.

```
add system image  { iso-filename | iso-URL [ username username password
password ] }
```

iso-filename

    Name of the image file to be added.

iso-URL

    URL location of the image file to be added.

username *username*

    Specifies the username that is required to log in to the remote system at the indicated URL location.

password *password*

    Specifies the password that is required to log in to the remote system at the indicated URL location. If the username is specified, then a password must also be specified.

Operational mode

Use this command to add a binary image to the currently running system. A system image can be added to a system that was installed by using a disk-based installation or an image-based installation. After an image is added, it is set as the new default boot image and is run the next time the system is booted.

The command validates the MD5 checksums of the files that are contained in the ISO image to ensure that the image has not been corrupted. In addition, the command does not allow more than one copy of an image to exist on the same system.

The *iso-filename* and *iso-URL* arguments provide the source for the ISO image file.

The following table shows how to specify the file syntax for different file locations.

Loca-tion	Specification
An ab-solute path	For *iso-filename*, use standard UNIX file specification.
A rel-ative path	For *iso-filename*, you can also specify the path name relative to the current directory.
FTP serv-er	Use the following syntax for the *iso-URL* argument: `ftp://user:passwd@host/image-file` where *user* is the username on the host, *passwd* is the password that is associated with the username, *host* is the host name or IP address of the FTP server, and *image-file* is the ISO image file, including the path. Alternatively, the username and password can be specified as **username** and **password** arguments of the `add system image` command.

Loca-tion	Specification
	If you do not specify *user* and *passwd*, you are prompted for them.
SCP server	Use the following syntax for the *iso-URL* argument: `scp://user:passwd@host/image-file` where *user* is the username on the host, *passwd* is the password that is associated with the username, *host* is the host name or IP address of the SCP server, and *image-file* is the ISO image file, including the path. Alternatively, the username and password can be specified as **username** and **password** arguments to the `add system image` command. If you do not specify *user* and *passwd*, you are prompted for them.
HTTP server	Use the following syntax for the *iso-URL* argument: `http://host/image-file` where *host* is the host name or IP address of the HTTP server and *image-file* is the ISO image file, including the path relative to the HTTP root directory.
TFTP server	Use the following syntax for the *iso-URL* argument: `tftp://host/image-file` where *host* is the host name or IP address of the TFTP server, and *image-file* is the ISO image file, including the path relative to the TFTP root directory.

clone system image

Creates a copy of a system image that is installed on the local system or a remote system.

```
clone system image  [ user@host: ] source-image-name new-image-name [ clean ]
```

user

Username on a remote host. A username is required for remote host access through SCP and is not required for cloning a local system image.

host

Host name or IP address of a remote host. The host name or IP address is required for remote access through SCP and is not required for cloning a local system image.

source-image-name

Name of a system image to be copied. The source image can exist on the local system or a remote system.

new-image-name

Name of the new (copied) system image. An image with this name must not exist on the system.

clean

Creates an empty read/write directory tree for the new image, which is a new image that is functionally equivalent to the source image as it existed when it was originally installed.

Operational mode

Use this command to create a copy of a system image that is installed on the local system or a remote system.

If *user@host* is specified, the image is fetched from the named host by using the Secure Copy Protocol (SCP). If *user@host* is omitted, *source-image-name* is the name of an image

that exists on the system, and *new-image-name* is the image name that the system uses for the clone. No image that is named *source-image-name* can exist on the system.

Command completion is performed for local image names if *user@host* is not specified. No command completion is performed on remote image names if *user@host* is specified.

If the **clean** argument is omitted, the command copies the `squashfs` file that is being used by the image named *source-image-name* and the read/write directory tree of *source-image-name*. If the **clean** argument is given, the read/write directory tree of *source-image-name* is not copied. Instead, an empty read/write directory tree is created for the new image, which creates a new image that is functionally equivalent to the source image as it existed when it was initially installed.

Images created by this command behave the same as images that are installed by install image or add system image.

Both the HTTPS and SSH services must be enabled on the remote system for clone system image to work properly. The HTTPS service is enabled by using the `set service https` command in configuration mode. The SSH service is enabled by using the `set service ssh` command in configuration mode.

delete system image

Deletes an image from the local disk drive.

```
delete system image  [ image-name ]
```

When the command is entered without an image name, the system prompts for the image to delete.

> **image-name**
>
>> Name of an image to be deleted.

Operational mode

Use this command to delete an image from the local disk drive.

The image and all its local files, including its configuration file, are destroyed. Because this command is destructive, the system prompts for confirmation.

Command completion displays all valid completions for the *image-name* argument. If the *image-name* argument is omitted, the system displays a list of available images and prompts you to select one.

If the system was originally installed in disk-based mode, an **image-name** option is available that you can use to direct that the disk-based installation must be deleted.

The system does not allow you to delete the currently running system image. However, the system does allow you to delete the image currently selected to be run at the next reboot. If you delete that image, the system uses the currently running image when the system is next rebooted.

install image

Installs a binary image of the system.

```
install image
```

Operational mode

Use this command to install a binary image of the system.

After the installation is completed, you can add multiple image versions to the same partition by using the `add system image` command, and you can then choose which version to boot by using the `set system image default-boot` command. This functionality allows you to move easily between different versions of the system.

If you have a new system and want to install from scratch, you can boot from the *.iso and run the `install image` command to install the image to the disk. The `install image` command operates similarly to the `install system` command—it creates and formats a new disk partition and then installs the image to the partition while preserving the system configuration.

rename system image

Renames an image.

```
rename system image  old-image-name new-image-name
```

old-image-name

Name of an existing image to be renamed.

new-image-name

New name for the image.

Operational mode.

Use this command to rename an image.

The old name must match the name of an image on the system. The system does not allow you to rename the currently running system image. The new system image name cannot be in use by another image.

set system image default-boot

Selects an image to be run when the system is next rebooted.

```
set system image default-boot  [ image-name ]
```

If the command is used without specifying an image name, the system displays a list of available images and prompts you to select one.

image-name

> Name of an image to be run when the system is rebooted.

Operational mode

Use this command to select an image to run when the system is next rebooted.

When multiple system images have been installed by using the `add system image` command, you can use this command to direct the system to boot from a specific system image the next time the system is restarted.

Command completion displays all valid completions for the *image-name* argument. If the *image-name* argument is omitted, the system displays a list that shows all images that are installed on the system and prompts you to select one. If the system was originally installed in disk-based mode, a special **image-name** option is available so that you can select the disk-based system as the default system from which to boot.

show system image

Displays a list of all images that are installed on the system.

```
show system image [ storage | version ]
```

storage

> Displays the amount of disk space that is used by each image.

version

> Includes the image version number in the display of system images.

Operational mode

Use this command to display a list of all images that are installed on the system.

The command output identifies the image that is currently running and the image that has been selected to run when the system is next rebooted. If the system was originally installed in disk-based mode, one of the image names identifies that installation.

spawn

Allows you to run any native Linux command through the operational mode infrastructure.

```
spawn  command-name [ text ]
```

command-name

> Command to run

text

> An argument to a command

Operational mode

Use this command to run any native Linux command through the operational mode infrastructure.

Commands that are spawned are run by the local shell, and with the current user's permissions. The spawned command provides a documented and supported way of running any Linux command.

The spawned command prevents any changes to the set of modeled commands from effecting the user's ability to run native commands consistently, if they are using this supported method.

Chapter 7. Licensing

About Licenses

DANOS-Vyatta edition supports the following node-locked license types, where the generated license activates just one device:

- Perpetual license - Permanent license.
- Subscription license - Limited duration license for evaluation

Enforcement Action

When a device license is expired or the device does not have the license, CPU cores will be limited to 1 core.

📝 **Note:** When a license is active, CPU cores are capped as per license file.

Delayed Enforcement

When a device license is released, to avoid the service interruption license enforcement is scheduled to happen 20 minutes after the license release time. This duration allows the user to install the license.

📝 **Note:** During this interim period any license installation failure will not lead to enforcement.

Host Identifier

A host is identified by the licensing module with the UUID prefixed with "IPI-". All the characters must be in uppercase.

The UUID is one time configurable for the VMs. Use the `show license` command after starting the machine.

Generate the License File

The *DANOS-Vyatta edition* device license file needs to be generated from the license portal provided to you.

The default name of a generated license file is `host-id.bin`. For example:

```
IPI-CC37ABBE0340.bin
```

Copy the License File to the Device

Use the `licensing get` command or `license-get` NetConf RPC to copy the license file from a remote file server to the device. Or use Linux utilities such as `curl` or `wget`.

licensing get command

```
licensing get URL dest-filepath FILEPATH routing-instance VRF
```

URL

> License file URL

FILEPATH

> Local filesystem's absolute path or path with file name to store the downloaded file

VRF

> Routing instance name

Note: Use the routing-instance name when the interface to be used in the download is part of a non-default routing-instance.

Download success:

```
tmpuser@node:~$ ls
tmpuser@node:~$ licensing get
 ftp://10.12.65.89:2121/IPI-A2E11F714C0B47E8B5237F34C0225387.bin
 dest-filepath /home/tmpuser/
########################################################################
 100.0%
tmpuser@node:~$
tmpuser@node:~$ ls
IPI-A2E11F714C0B47E8B5237F34C0225387.bin
tmpuser@node:~$
```

Download failure:

```
tmpuser@node:~$  licensing get
ftp://10.12.65.89:2121/IPI-A2E11F714C0B47E8B5237F34C0225387.bin
dest-filepath /home/tmpuser/
curl: (7) Couldn't connect to server
tmpuser@node:~$
```

NetConf license-get RPC

```
rpc license-get {
    configd:call-rpc "vyatta-license-rpc --license-get";
    description "Install license on device";
    input {
        leaf url  {
            type string;
            mandatory true;
            description "License file URL";
        }
```

```
        leaf dest-filepath  {
            type string;
            mandatory true;
            description "Local filesystem's absolute path or
            path with file name to store the downloaded file";
        }

        leaf routing-instance  {
            type string;
            description "Download license through this routing-instance";
        }
    }

    output {
        leaf license-response {
            description "License file download status";
            type string;
        }
    }
}
```

Download success:

```
license_get.xml:
<license-get xmlns="urn:vyatta.com:mgmt:vyatta-op-licensing-v1:1">
<url>ftp://10.12.65.89:2121/IPI-A2E11F714C0B47E8B5237F34C0225387.bin</url>
<dest-filepath>/home/tmpuser/</dest-filepath>
</license-get>

netconf> rpc license_get.xml
<?xml version='1.0' encoding='UTF-8'?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
 xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
 message-id="urn:uuid:5e3a0a66-f666-43c1-875b-af7eddf7c4d9">
<ok/>
</rpc-reply>
netconf>
```

Download failure:

```
license_get.xml:
<license-get xmlns="urn:vyatta.com:mgmt:vyatta-op-licensing-v1:1">
<url>ftp://10.12.65.89:2121/IPI-A2E11F714C0B47E8B5237F34C0225387.bin</url>
<dest-filepath>/home/tmpuser/</dest-filepath>
<routing-instance>wrong-routing-instance</routing-instance>
</license-get>

netconf> rpc license_get.xml
<?xml version='1.0' encoding='UTF-8'?>
```

```
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
 xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
 message-id="urn:uuid:2c959683-cf00-4e3f-b756-d605ec6201ed">
<license-response
 xmlns="urn:vyatta.com:mgmt:vyatta-op-licensing-v1:1">curl: (7) Couldn't
 connect to server
</license-response>
</rpc-reply>
netconf>
```

Install the License

Use the `licensing install` command or `license-install` NetConf RPC to install the license file present on the device local file system. The device clock needs to be up to date, otherwise an incorrect clock on the device may lead to license installation failure. See License Troubleshooting to resolve installation failures.

licensing install command

```
licensing install PATH
```

PATH

License file absolute path

NetConf license-install RPC

```
rpc license-install {
        configd:call-rpc "vyatta-license-rpc --license-install";
        description "Install license on device";
        input {
            leaf file-path  {
                type string;
                mandatory true;
                description "Local filesystem path of the
                    license file to be installed";
            }
        }
        output {
            leaf license-response {
                description "License installation status";
                type string;
            }
        }
}
```

Installing on a Device with an Inactive License

Command:

```
tmpuser@node:~$
tmpuser@node:~$ licensing
 install /home/tmpuser/IPI-A2E11F714C0B47E8B5237F34C0225387.bin
tmpuser@node:~$
tmpuser@node:~$ show licensing
status      : active
feature     : DVE-vSDE-BASE
environment : Virtual
host-id     : IPI-A2E11F714C0B47E8B5237F34C0225387
type        : subscription
enforcement : CPU_COUNT=2
validity-in-days : 357
tmpuser@node:~$
```

NetConf RPC:

```
netconf> rpc license_install.xml
<?xml version='1.0' encoding='UTF-8'?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
 xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
 message-id="urn:uuid:0da2341c-33e8-4059-b561-60633fc1abd0">
<ok/>
</rpc-reply>
netconf>

netconf> rpc license_show.xml
<?xml version='1.0' encoding='UTF-8'?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
 xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
 message-id="urn:uuid:34621de9-5320-4da4-91c2-5bb2a2f6ff34">
<license-status xmlns="urn:vyatta.com:mgmt:vyatta-op-licensing-v1:1">
<enforcement>CPU_COUNT=2</enforcement>
<validity-in-days>357</validity-in-days>
<status>active</status>
<feature>DVE-vSDE-BASE</feature>
<host-id>IPI-A2E11F714C0B47E8B5237F34C0225387</host-id>
<model>node-locked</model>
<type>subscription</type>
</license-status>
</rpc-reply>
netconf>
```

Installing on a Device with an Active License

Any active license on the device must be released before installing a new license.

Command:

```
tmpuser@node:~$ show licensing
status      : active
feature     : DVE-vSDE-BASE
environment : Virtual
host-id     : IPI-A2E11F714C0B47E8B5237F34C0225387
type        : subscription
enforcement : CPU_COUNT=2
validity-in-days : 357
tmpuser@node:~$
tmpuser@node:~$ licensing
 install /home/tmpuser/IPI-A2E11F714C0B47E8B5237F34C0225387.bin
Device already have a valid license
tmpuser@node:~$
```

NetConf RPC:

```
license_show.xml:
<show-license-status xmlns="urn:vyatta.com:mgmt:vyatta-op-licensing-v1:1">
</show-license-status>

license_install.xml:
<license-install xmlns="urn:vyatta.com:mgmt:vyatta-op-licensing-v1:1">
<file-path>/home/tmpuser/IPI-A2E11F714C0B47E8B5237F34C0225387.bin</file-pat
h>
</license-install>


netconf> rpc license_show.xml
<?xml version='1.0' encoding='UTF-8'?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
 xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
 message-id="urn:uuid:f338e420-7131-4fa8-993a-171bec666be8">
<license-status xmlns="urn:vyatta.com:mgmt:vyatta-op-licensing-v1:1">
<model>node-locked</model>
<type>subscription</type>
<enforcement>CPU_COUNT=2</enforcement>
<validity-in-days>357</validity-in-days>
<status>active</status>
<feature>DVE-vSDE-BASE</feature>
<host-id>IPI-A2E11F714C0B47E8B5237F34C0225387</host-id>
</license-status>
</rpc-reply>
netconf>
netconf> rpc license_install.xml
<?xml version='1.0' encoding='UTF-8'?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
 xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
 message-id="urn:uuid:feb28a3e-2f57-475e-bc21-4ecfe28273b1">
<license-response
 xmlns="urn:vyatta.com:mgmt:vyatta-op-licensing-v1:1">Device already have a
 valid license</license-response>
```

```
</rpc-reply>
netconf>
```

View License Status

Command:

```
show licensing
```

NetConf RPC:

```
rpc show-license-status {
 configd:call-rpc "vyatta-license-rpc --show-license";
 description "This RPC will display the license status informations";
 output {
  leaf license-response {
   description "show-license-status rpc failure response";
   type string;
  }
  container license-status {
   leaf status {
    description "License installation status";
    type enumeration {
     enum "inactive";
     enum "active";
    }
   }

   leaf feature {
 description "This attribute refers to the device feature/SKU name, which
 is used in license validation. The feature/SKU name present in the license
 file must match with device feature/SKU name";
    type string;
   }

   leaf type {
    description "Indicates the installed license type";
    type enumeration {
     enum "subscription";
     enum "perpetual";
    }
   }

   leaf enforcement {
    description "This attribute refers to the enforcement attribute value";
    type string;
   }

   leaf validity-in-days {
```

```
      description "License validity in days, and this field exisits when the
 license type is trial";
      type uint64;
    }

    leaf model {
description "License model, indicates the method used by the device to
 acquire the license";
     type enumeration {
      enum "node-locked";
      enum "license-server";
     }
     }

    leaf host-id {
description "License identifier, this is the device base MAC or UUID for VM
 prefixed with IPI- , and this attribute is used in license validation";
      type string;
    }

    leaf error {
     description "Error observed while installing the license";
     type uint64;
    }

    leaf error-string {
description "Error messages in user readable text form corresponding to the
 error observed during license installation";
      type string;
    }
   }
  }
}
```

Installation Success

Command:

```
tmpuser@node:~$ show licensing
status      : active
feature     : DVE-vSDE-BASE
environment : Virtual
host-id     : IPI-E8C57A186F1B
type        : subscription
validity-in-days    : 59
```

NetConf RPC:

```
license_show.xml:
<show-license-status xmlns="urn:vyatta.com:mgmt:vyatta-op-licensing-v1:1">
</show-license-status>

netconf> rpc license_show.xml
<?xml version='1.0' encoding='UTF-8'?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
 xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
 message-id="urn:uuid:f338e420-7131-4fa8-993a-171bec666be8">
<license-status xmlns="urn:vyatta.com:mgmt:vyatta-op-licensing-v1:1">
<model>node-locked</model>
<type>subscription</type>
<enforcement>CPU_COUNT=2</enforcement>
<validity-in-days>357</validity-in-days>
<status>active</status>
<feature>DVE-vSDE-BASE</feature>
<host-id>IPI-A2E11F714C0B47E8B5237F34C0225387</host-id>
</license-status>
</rpc-reply>
```

Installation Failure

Command:

```
tmpuser@node:~$ show licensing
status      : inactive
feature     : DVE-vSDE-BASE
environment : Virtual
host-id     : IPI-E8C57A2E4F1B
error       : 14
error-string: License not available
```

NetConf RPC:

```
license_show.xml:
<show-license-status xmlns="urn:vyatta.com:mgmt:vyatta-op-licensing-v1:1">
</show-license-status>

netconf> rpc license_show.xml
<?xml version='1.0' encoding='UTF-8'?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
 xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
 message-id="urn:uuid:471f9d0e-c1e9-4a98-9042-d35c73864b8e">
  <license-status xmlns="urn:vyatta.com:mgmt:vyatta-op-licensing-v1:1">
    <status>inactive</status>
    <feature>DVE-vSDE-BASE</feature>
    <host-id>IPI-A2E11F714C0B47E8B5237F34C0225387</host-id>
```

```
    <error>14</error>
    <error-string>License not available</error-string>
  </license-status>
</rpc-reply>
netconf>
```

License Troubleshooting

Table 1. License Troubleshooting

Error code	Description	Resolution
1	Failed to create license error object	Report the error to support team .
2	Failed to create license environment	Report the error to support team .
3	Failed to add trusted storage as license source	Trusted storage path does not exist or does not have required permission. Report the error to the support team.
4	Failed to create trusted storage folder	Creation of trusted storage inside /config/license-manager parent directory is not successful. This should not fail, but if you see this error report to the support team.
5	Failed to create license object	Report the error to support team.
6	Failed to process license	Downloaded license files could have been corrupted, download the license file from the portal again and try installing. Also compare the checksum of the license file between the file present in the device and the one downloaded from the portal.
7	Failed to get mapped SKU in license	The generated license could have been mapped to a different software feature than what is required for the device. Use the `show licensing` command to understand the SKU to be mapped while generating the license. This happens mostly when customers have purchased multiple software features / SKUs.
8	Device SKU is not found in license	Generate the license file by mapping the correct device feature in the portal.
9	Failed to acquire license	Report the error to support team.
10	Failed to return acquired license	Report the error to support team.
11	License file is not the latest one, please install the recently downloaded license or download again from the portal	When license is generated successively say first and second, installation of second on top of the first is accepted, but not vice versa.
12	Failed to update private storage	Report the error to support team.
13	Failed to fetch data from private storage	Report the error to support team.
14	License not available	Device does not have any license activated.
15	Subscription license is expired	Validity of the device license has ended, download new license from portal and install if you have purchased.
16	Device already have a valid license	To install new license on the already license activated device. 1. Use `licensing release` command or `license-release` RPC to remove current license

Table 1. License Troubleshooting (continued)

Error code	Description	Resolution
		2. Use `licensing install` command or `license-install` RPC to install new license
17	Device does not have a valid license	Device does have any license activated.
18	File not found	Check whether the specified absolute path, and the file exists on the device.
19	Empty file	Could be a copy or the download issue, please download the license into the device using the alternate method or compare the checksum of the license file between the file present in the device and the one downloaded from the portal.
20	License start date is in future	Device clock must be up to date to install the license successfully.
21	Invalid host-id	The license is generated with a different HOST ID, please cross check and download the license with the correct HOST-ID.
22	License activation time (lifetime) elapsed, please download the license again from the portal	License file downloaded from the portal should be installed within 42 days of generation, otherwise you need to download the license file from portal again