Clavister Virtual Series
Getting Started Guide for KVM

cOS Core version: 12.00.17

Clavister AB
Sjögatan 6J
SE-89160 Örnsköldsvik
SWEDEN

Head office/Sales: +46-(0)660-299200
Customer support: +46-(0)660-297755
www.clavister.com

Published 2019-04-03

Clavister Virtual Series
Getting Started Guide for KVM
cOS Core version: 12.00.17

Published 2019-04-03

Copyright Notice

Disclaimer

The information in this document is subject to change without notice. Clavister makes no representations or warranties with respect to the contents hereof and specifically disclaims any implied warranties of merchantability or fitness for a particular purpose. Clavister reserves the right to revise this publication and to make changes from time to time in the content hereof without any obligation to notify any person or parties of such revision or changes.

Limitations of Liability

UNDER NO CIRCUMSTANCES SHALL CLAVISTER OR ITS SUPPLIERS BE LIABLE FOR DAMAGES OF ANY CHARACTER (E.G. DAMAGES FOR LOSS OF PROFIT, SOFTWARE RESTORATION, WORK STOPPAGE, LOSS OF SAVED DATA OR ANY OTHER COMMERCIAL DAMAGES OR LOSSES) RESULTING FROM THE APPLICATION OR IMPROPER USE OF THE CLAVISTER PRODUCT OR FAILURE OF THE PRODUCT, EVEN IF CLAVISTER IS INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. FURTHERMORE, CLAVISTER WILL NOT BE LIABLE FOR THIRD-PARTY CLAIMS AGAINST CUSTOMER FOR LOSSES OR DAMAGES. CLAVISTER WILL IN NO EVENT BE LIABLE FOR ANY DAMAGES IN EXCESS OF THE AMOUNT CLAVISTER RECEIVED FROM THE END-USER FOR THE PRODUCT.

Table of Contents

Preface

Target Audience

The target audience for this guide is the administrator who wants to run the cOS Core network operating system under a KVM virtual environment with QEMU as the hypervisor on a Linux platform. The guide takes the user from the installation of cOS Core through to startup of the software, including network connections and initial cOS Core configuration.

Text Structure

The text is divided into chapters and subsections. Numbered subsections are shown in the table of contents at the beginning of the document.

Text links

Where a "See section" link is provided in the main text, this can be clicked on to take the reader directly to that reference. For example, see *Section 6.6, “Setup Troubleshooting ”*.

Web links

Web links included in the document are clickable. For example, *http://www.clavister.com*.

Notes to the main text

Special sections of text which the reader should pay special attention to are indicated by icons on the left hand side of the page followed by a short paragraph in italicized text. There are the following types of such sections:

Note

This indicates some piece of information that is an addition to the preceding text. It may concern something that is being emphasized or something that is not obvious or explicitly stated in the preceding text.

Tip

This indicates a piece of non-critical information that is useful to know in certain situations but is not essential reading.

Caution

This indicates where the reader should be careful with their actions as an undesirable situation may result if care is not exercised.

Important

This is an essential point that the reader should read and understand.

Warning
This is essential reading for the user as they should be aware that a serious situation may result if certain actions are taken or not taken.

Trademarks

Certain names in this publication are the trademarks of their respective owners.

cOS Core is the trademark of Clavister AB.

Windows, *Windows XP*, *Windows Vista* and *Windows 7* are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Apple, Mac, Mac OS, and Macintosh are trademarks of Apple Inc.

Chapter 1: Overview & Requirements

cOS Core with KVM

By using the open source *Kernel-based Virtual Machine* (KVM) software, it is possible to have a single computer running multiple, virtual Clavister Next Generation Firewalls with each virtual firewall running a separate copy of cOS Core. This technique is referred to as *virtualization* and each virtual Clavister Next Generation Firewall can be said to be running in its own *virtual machine*. This is the basis for the Clavister *Virtual Series* of products which also includes cOS Core running in the VMware and Hyper-V virtual environments.

cOS Core is an Operating System

cOS Core is a network security operating system that is not built on a pre-existing operating system like Linux. Instead, it is itself both the operating system and firewall. This means cOS Core has modest resource requirements under KVM, such as only needing a very small disk space footprint of under 64 megabytes. A detailed list of resource requirements can be found later in this chapter.

Important: A virtual host should run only cOS Core as a guest

To provide maximum security, the virtual host should be running cOS Core as the only guest. This defends against security attacks against vulnerable hardware, where local data in a processor might be read by other software sharing the same processor. The attacks known as "Spectre" and "Meltdown" are examples of this.

cOS Core Management

Not only can cOS Core run in its own virtual machine under KVM, the management workstation that is used to administer cOS Core can also run under the same KVM installation or it can be on a separate, external computer. To perform management tasks, the management workstation may run InControl, the Web Interface or a CLI console through a secure shell client.

KVM Runs Under Linux With QEMU

KVM itself is not a hypervisor but provides an infrastructure for creating virtual machines. It is the *Quick EMUlator* (QEMU) that provides the hypervisor functions under the Linux operation system and this is also required when using KVM to create cOS Core virtual machines. The combination is known as *QEMU-KVM* and is distributed as a single package so that the two can be installed together.

Referencing KVM Documentation

This guide describes the steps involved when installing cOS Core with KVM on x86 based hardware as well as covering many of the issues that may be encountered with cOS Core running in a KVM virtual environment.

The guide tries to deal specifically with the subject of cOS Core running under KVM and, unless relevant, does not detail the installation of KVM itself or issues which are related only to KVM. Pure KVM subjects are best explained by other, KVM specific, documentation.

Server Hardware Requirements

The server running KVM must satisfy the following criteria:

- The hardware architecture must be 64-bit x86.
- It must support virtualization using Intel VT-x or AMD-V architectures.
- Intel processors must be from either the Intel *Core* (or later) workstation or Intel server family of products.
- It must support a ratio of 1-to-1 for threads to defined virtual CPU.
- It must support at least one core per virtual CPU.

Intel Hardware Drivers

The following additional hardware driver requirements should be noted:

- If used, the IXGBE (Intel 10 Gigabit network) driver should be version 3.21.2 or later.
- Also note that if used, the IXGBE driver should be recompiled with the LRO (Large Packet Receive) feature **disabled**. Information about how to do this can be found in the README file included with the driver distribution.
- If used, the IGB (Intel Network) driver should be version 5.2.5 or later.

cOS Core Resource Requirements

A single installation of cOS Core as a virtual firewall will require the following resource requirements:

- Minimum of 1 x virtual CPU.
- Minimum 128 Mbytes RAM (512 recommended, depending on features used).
- Maximum 4096 Mbytes RAM. Requirements differ according to cOS Core license.
- 64 Mbytes of free disk space for cOS Core installation.

Supported Linux Distributions

KVM with QEMU will run under the Linux operating system and will require one of the following Linux distributions:

- Ubuntu version 13.04 or later.
- openSUSE version 12.2 or later.
- Centos version 6.4 or later.
- Red Hat Enterprise Linux 6.4 or later.

Other Linux distributions might be used successfully but have not been tested by Clavister with cOS Core. The installation of Linux will not be discussed further in this guide. It is assumed the administrator is familiar with basic Linux networking.

Supported KVM Distributions

cOS Core can run under the latest distribution of KVM. These distributions also include QEMU. The QEMU release (or later) that must be used for cOS Core to function properly:

- QEMU emulator version 2.0.0 (Debian 2.0.0+dfsg-2ubuntu1.17) © 2003-2008 Fabrice Bellard.

Other distributions might be used successfully but have not been tested by Clavister. The installation of QEMU with KVM will not be discussed further in this guide and the administrator should refer to the software's own documentation. The QEMU/KVM binaries for a particular Linux distribution can normally can normally be installed from the repositories of the distribution.

Note that the SeaBIOS version used with KVM for guest x86 operating systems should be version 1.7.4 or later.

Additional Linux Software

The following should also be installed on the base Linux system:

- The *libvirt* virtualization API must be installed under Linux. This is needed for the libvirt daemon (*libvirtd*) and the *virsh* command.
- The *vhost-net* enhancement for networking is required. This moves packets between cOS Core and the host system using the Linux kernel instead of QEMU and provides a significant performance boost to throughput. The Clavister setup script described later will terminate with an error message if this is not installed.
- Either *bridge-utils* or *Open vSwitch* must be installed to provide networking functions. It is not possible to install both. If the virtual machine will be part of a cOS Core HA cluster, then *Open vSwitch* must be installed.

  If a high availability (HA) cluster is to be set up, *Open vSwitch* must be installed. HA will not function with *bridge-utils* and this must be removed. HA setup is discussed further in *Chapter 7, High Availability Setup*.

Software Tools for Management

The following are the software requirements for management:

- KVM virtual machine management software must be installed under Linux. This guide assumes that the *Virtual Machine Manager* (virt-manager) software is installed.

- Optionally use a VNC client on a separate computer workstation to access the Linux environment. This guide assumes *UltraVNC Viewer* is used but other clients may be suitable.

The installation of these software tools will not be discussed further in this guide. The administrator should refer to the tool's own documentation for guidance.

The cOS Core installation files for these servers can be downloaded from the Clavister website *https://www.clavister.com*. KVM files and further information can be found by going to *http://www.linux-kvm.org*.

Chapter 2: Installation

This section describes the overall installation steps of cOS Core in a virtual environment. It includes details of customer registration and license installation. The steps are organized into the following stages:

A. Register as a User and Download cOS Core

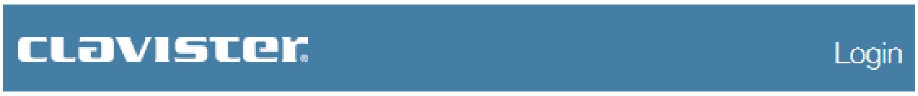
B. Create a cOS Core Virtual Machine

C. Configure cOS Core for Management Access

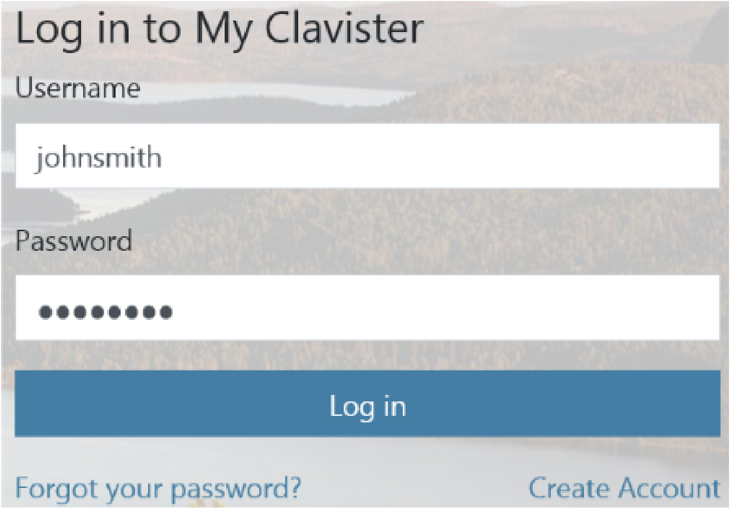
D. Register a License and Bind it to cOS Core

A. Register as a User and Download cOS Core

1. Open a web browser, go to ***https://www.clavister.com*** and select the **Login** link at the top of the page.



2. The *MyClavister* login page is presented. If you are already registered, log in and skip to step **8**. If you are a new customer accessing *MyClavister* for the first time, click the **Create Account** link.



Log in to My Clavister

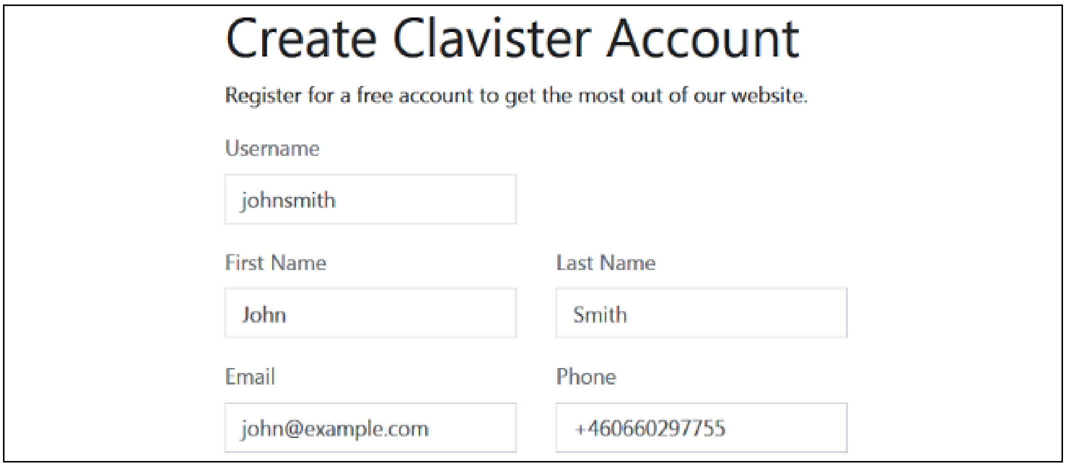
Username

johnsmith

Password

●●●●●●●●

3. The registration page is now presented. The required information should be filled in. In the example below, a user called *John Smith* is registering.



Create Clavister Account

Register for a free account to get the most out of our website.

Username

johnsmith

First Name

John

Last Name

Smith

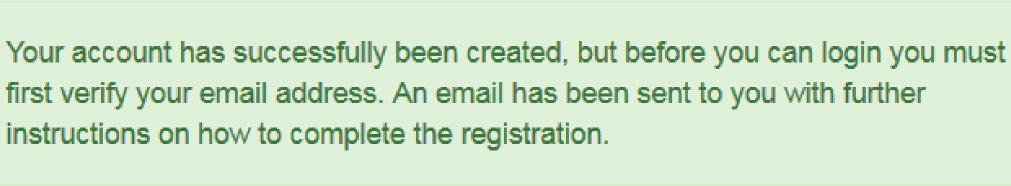
Email

john@example.com

Phone

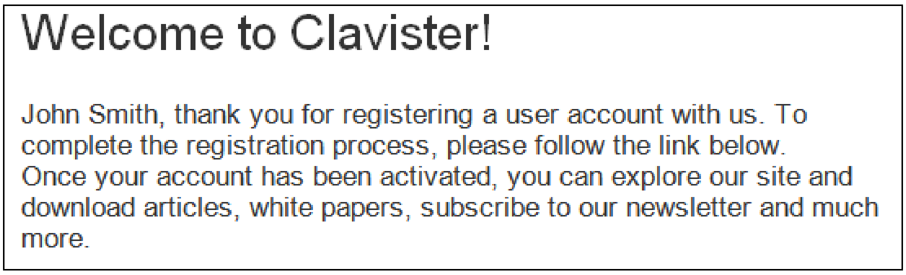
+460660297755

4. When the registration is accepted, an email is sent to the email address given so that the registration can be confirmed.



Your account has successfully been created, but before you can login you must first verify your email address. An email has been sent to you with further instructions on how to complete the registration.

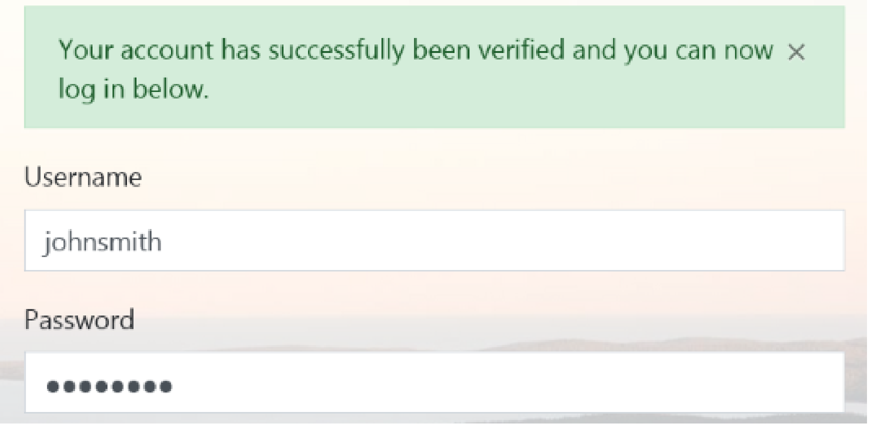
5. Below is an example of the heading in the email that would be received.



Welcome to Clavister!

John Smith, thank you for registering a user account with us. To complete the registration process, please follow the link below.
Once your account has been activated, you can explore our site and download articles, white papers, subscribe to our newsletter and much more.

6. The confirmation link in the email leads back to the Clavister website to show that confirmation has been successful and logging in is now possible.

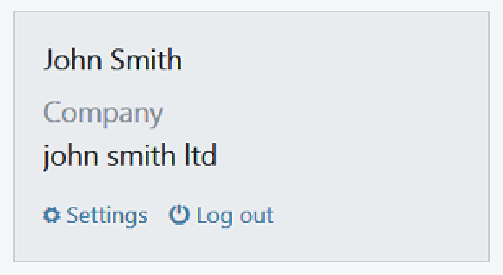


Your account has successfully been verified and you can now log in below. ×

Username: johnsmith

Password: ●●●●●●●●

7. After logging in, the customer name is displayed with links for changing settings and logging out.

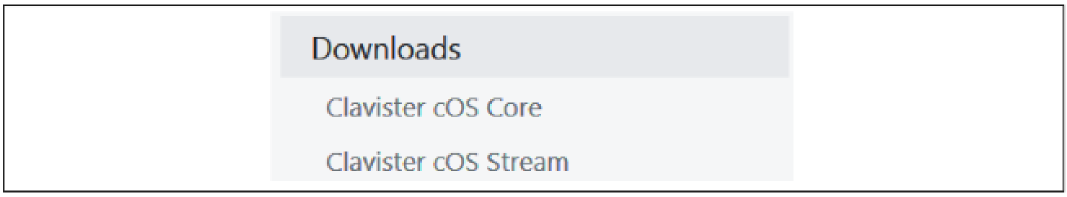


John Smith

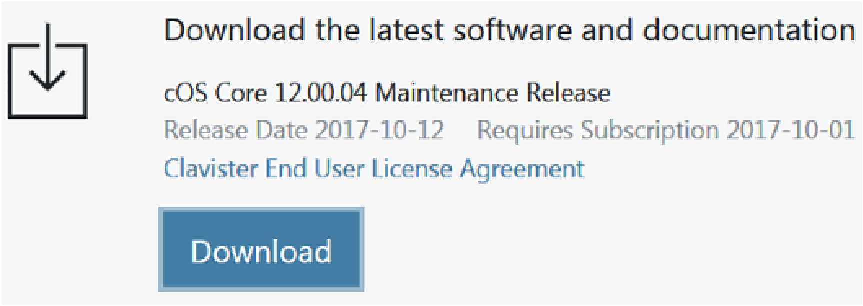
Company

john smith ltd

8. To download cOS Core for VMware, select *Downloads* and then *cOS Core*.



9. Press the *Download* button next to the desired product and version number to get a list in a popup window of all the different distributions available for that version. The button for the latest version is always at the top.



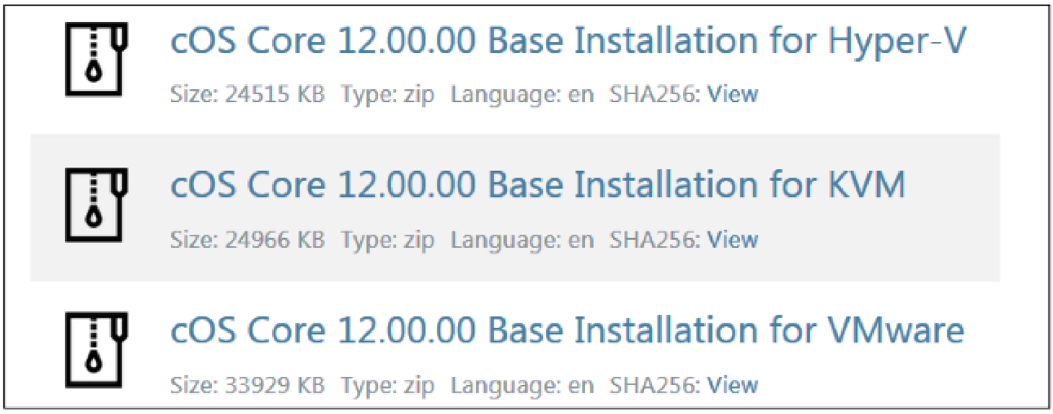
Download the latest software and documentation

cOS Core 12.00.04 Maintenance Release

Release Date 2017-10-12 Requires Subscription 2017-10-01

Clavister End User License Agreement

10. Select and download the ZIP file containing the cOS Core distribution to the local disk. Note that sometimes the *Base* distribution may not be available for a minor bug-fix revision (for example, version 12.00.05) so the latest *Base* distribution (for example, version 12.00.00) must be installed first and then the single upgrade distribution applied to get to the desired version.



cOS Core 12.00.00 Base Installation for Hyper-V

Size: 24515 KB Type: zip Language: en SHA256: View

cOS Core 12.00.00 Base Installation for KVM

Size: 24966 KB Type: zip Language: en SHA256: View

cOS Core 12.00.00 Base Installation for VMware

Size: 33929 KB Type: zip Language: en SHA256: View

B. Create a cOS Core Virtual Machine

Unzip the downloaded file and follow the instructions for creating the relevant cOS Core virtual machine using the instructions in ***Chapter 3, Importing Virtual Machines***.

C. Configure cOS Core for Management Access

1. cOS Core can now be configured using the CLI for public Internet access and to allow management access via the **If1** interface. The CLI steps are as follows:

   i. The cOS Core *address book* is automatically filled with address objects for the IPv4 address and network of all the Ethernet interfaces. Assign the IPv4 address of the **If1** interface. This will be used for remote management:

```
Device:/> set Address IP4Address
                        InterfaceAddresses/If1_ip
                        Address=203.0.113.10
```

Next, assign the IPv4 network for the **If1** interface:

```
Device:/> set Address IP4Address
                        InterfaceAddresses/If1_net
                        Address=203.0.113.0/24
```

ii. An *all-nets* default route needs to be added to the *main* routing table which includes the gateway address of a router for public Internet access. Unless there is a narrower route that matches for traffic, this route will be used. To add the route, the CLI context needs to be changed to be the *main* routing table:

```
Device:/> cc RoutingTable main
```

The command prompt will change to show that the current context is the *main* routing table:

```
Device:/main>
```

Now, routes can be added to the *main* table. Assuming that the *If1* interface is connected to a router with the IPv4 address *203.0.113.1* then a default route is added with the following CLI:

```
Device:/main> add Route Interface=If1 Network=all-nets
                        Gateway=203.0.113.1
```

iii. Next, restore the CLI context to the default:

```
Device:/> cc
```

iv. For management access, the *RemoteManagement* object needs to be changed so it allows the source IP to connect. This could be a specific IPv4 address or network but here it is set to *all-nets* so any source IP will be acceptable:

```
Device:/> set RemoteManagement HTTP_If1 Network=all-nets
```

Normally, an *IP Rule* configuration object should be created for any data traffic to be allowed to flow to or from cOS Core but management access does not require a separate rule.

v. The cOS Core configuration changes now needs to be activated:

```
Device:/> activate
```

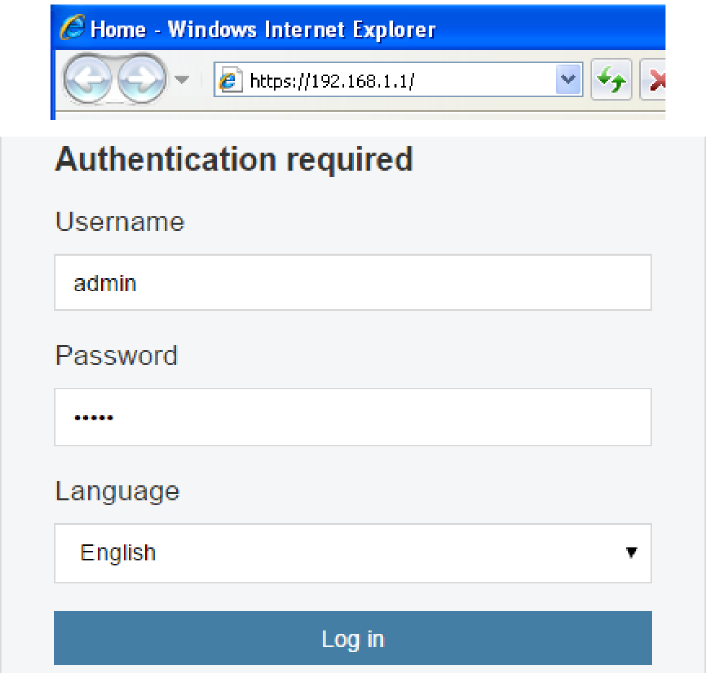
Following activation, the changes must be committed permanently within 30 seconds otherwise the configuration will revert back to the original configuration and the changes will be lost. This is a check by cOS Core that the administrator has not been locked out by the changes:

```
Device:/> commit
```

6. Finally, open a web browser and surf to the IP address of the **If1** interface. The cOS Core login dialog should appear and the default administrator credentials of username *admin* with password *admin* can be used to log in. By default, only the *HTTPS* protocol can be used so the connection will be encrypted. With *HTTPS*, cOS Core will send a self-signed certificate and the browser will prompt for that certificate to be accepted.

   Ít is possible to enable unencrypted HTTP for the management connection but this is not recommended.

   When connecting through the Web Interface for the first time, the *cOS Core Setup Wizard* will automatically try to start as a browser popup window. Browser popups may be disallowed so the browser will ask if the wizard popup should be allowed. For this section, the wizard is not used and its popup window can be dismissed. Using the wizard is described in *Section 6.2, "Web Interface and Wizard Setup"*.



Home - Windows Internet Explorer

https://192.168.1.1/

Authentication required

Username

admin

Password

•••••

Language

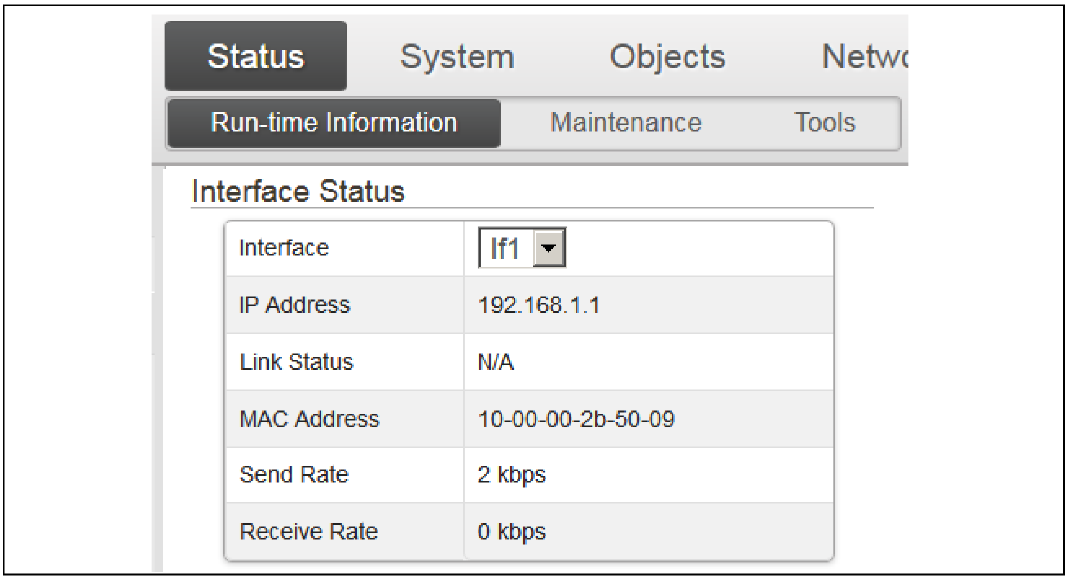
English

D. Register a License and Bind it to cOS Core

1. A cOS Core license for VMware must be associated with a MAC address on the virtual machine. To get a MAC address, open the cOS Core Web Interface and go to **Status > Run-time Information > Interfaces** and make a note of the MAC address for the **If1** interface.

   Alternatively, the following CLI command can be used to obtain the MAC address:

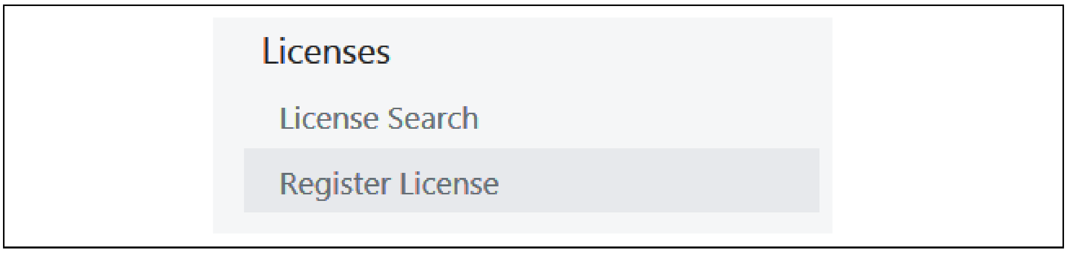
```
Device:/> ifstat If1
```



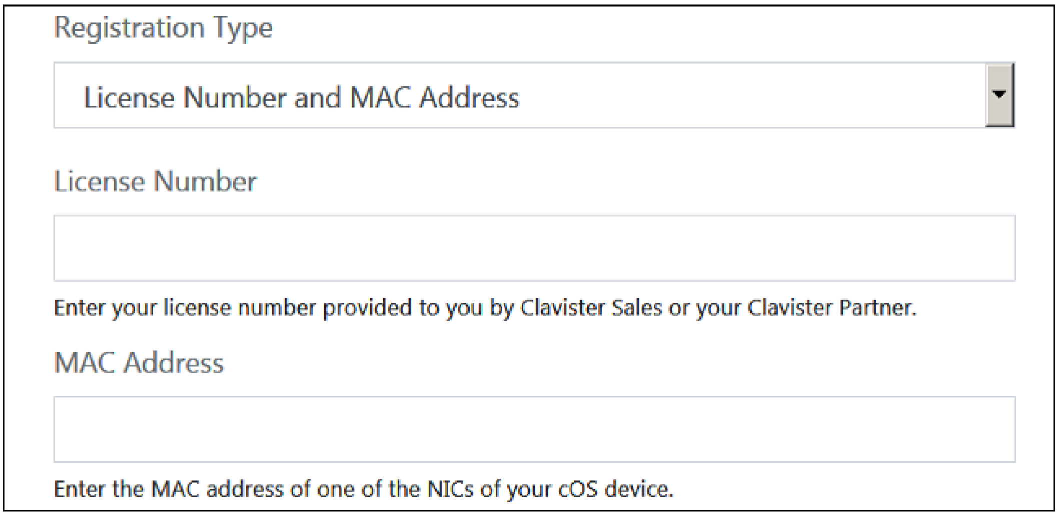
Interface Status

Interface	If1
IP Address	192.168.1.1
Link Status	N/A
MAC Address	10-00-00-2b-50-09
Send Rate	2 kbps
Receive Rate	0 kbps

2. Now, log in to the Clavister website and select **Register License**.



3. The registration page is displayed. Select the option **License Number and MAC address** then enter the MAC address noted earlier with the license number and click **Register License**. The license number will be given to you by your Clavister reseller.



Registration Type

License Number and MAC Address

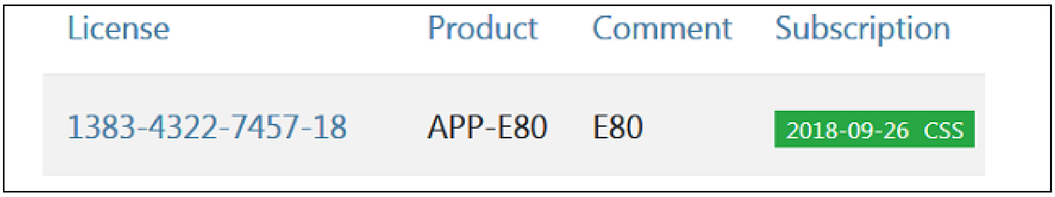
License Number

Enter your license number provided to you by Clavister Sales or your Clavister Partner.

MAC Address

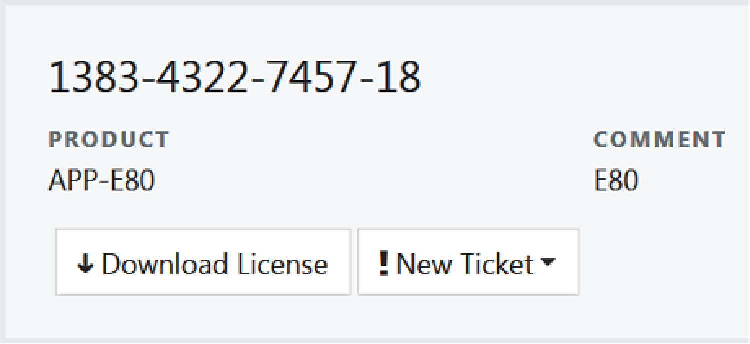
Enter the MAC address of one of the NICs of your cOS device.

4. After the license is registered and associated with the MAC address, the license file can be downloaded from the license list. And example list entry in the list is shown below. The entry also shows the current support agreement expiry date.



License	Product	Comment	Subscription
1383-4322-7457-18	APP-E80	E80	2018-09-26 CSS

5. Clicking on an entry in the list will open a display of the license details with a **Download License** button displayed at the top. An example button is shown below.

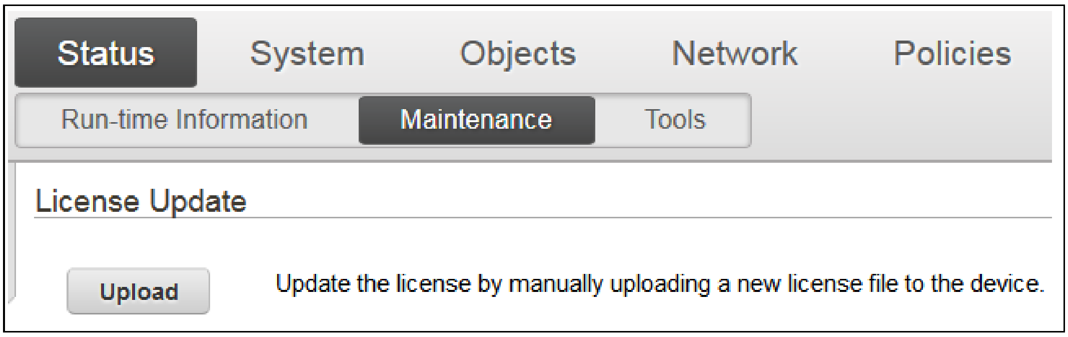


1383-4322-7457-18

PRODUCT: APP-E80

COMMENT: E80

6. After clicking the button and downloading the license, go back to the cOS Core Web Interface and go to **Status** > **Maintenance** > **License**. Select *Upload* to upload the license file from the management computer to cOS Core.



License Update

Update the license by manually uploading a new license file to the device.

The 2 hour evaluation time limit will now be removed and cOS Core will only be restricted by the capabilities defined by the license.

Chapter 3: Importing Virtual Machines

This chapter describes importing a disk image distribution of a virtual machine running cOS Core. It is assumed that KVM software with QEMU has been installed and is running in the appropriate environment and the relevant software tools have also been installed. This initial setup has been described previously in *Chapter 1, Overview & Requirements*.

Virtual machine creation can be done in one of the following ways:

- **Manually** using virtual machine manager. This is described next in *Section 3.1, "Manual Setup"*.

- **Automatically** using a script. An example script is supplied by Clavister and this is described next in *Section 3.2, "Script Setup"*.

cOS Core Memory Requirements

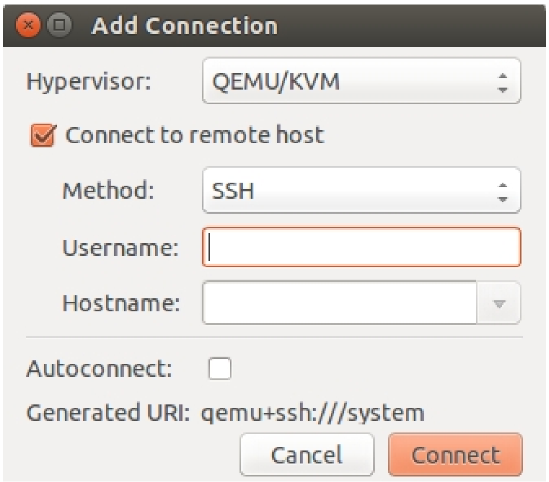
The minimum amount of memory required for cOS Core to run is 128 Mbytes and memory should not be reduced below this level. However, certain memory demanding features cannot run in this ammount of memory so a minimum of 512 Mbytes is recommended. This allocation may need to be increased depending on the cOS Core license available and the number of connections/tunnels that will be open simultaneously. The maximum memory allocation possible for cOS Core is 4096 Mbytes. Anything available above this will not be used.

If the allocated memory is insufficient during operation, cOS Core will output console messages indicating this while trying to reduce the number of open connections/tunnels. Eventually, cOS Core will enter *safe mode* where only management access is possible.

3.1. Manual Setup

This section describes using *Virtual Machine Manager* to manually create a cOS Core virtual machine.

1. Start *Virtual Machine Manager* and select the menu options **File > Add Connection**.



Add Connection

Hypervisor: QEMU/KVM

☑ Connect to remote host

Method: SSH

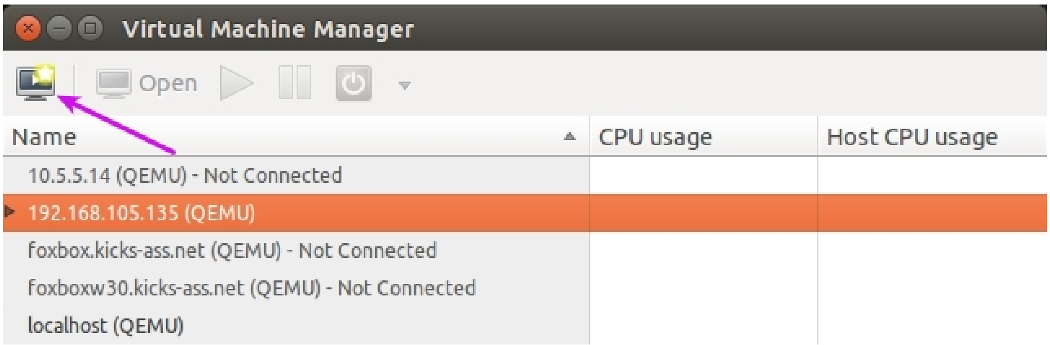
Username:

Hostname:

Autoconnect: □

Generated URI: qemu+ssh:///system

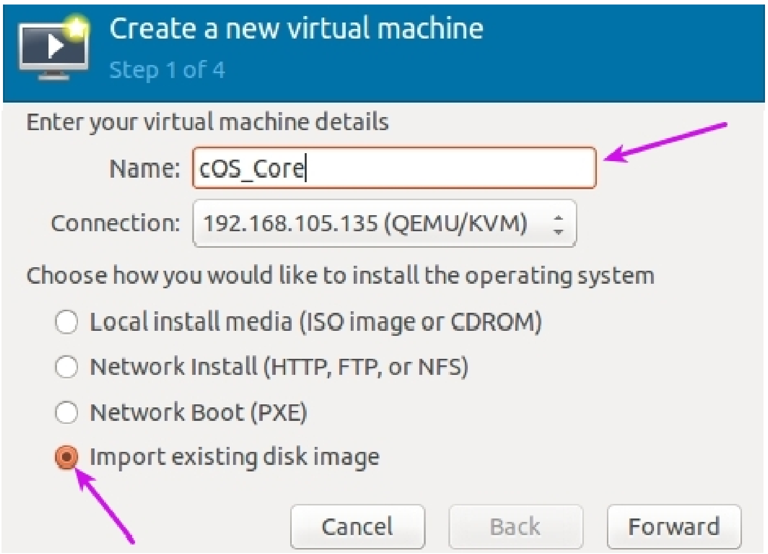
2. Double click on the new host to open it. You will be prompted for the password.



Virtual Machine Manager

Name	CPU usage	Host CPU usage
10.5.5.14 (QEMU) - Not Connected		
192.168.105.135 (QEMU)		
foxbox.kicks-ass.net (QEMU) - Not Connected		
foxboxw30.kicks-ass.net (QEMU) - Not Connected		
localhost (QEMU)		

3. Select **Create new virtual machine** to start the new VM wizard. Give the new virtual machine a name and select *Import existing disk image* then **Forward**.



Create a new virtual machine

Step 1 of 4

Enter your virtual machine details

Name: cOS_Core

Connection: 192.168.105.135 (QEMU/KVM)

Choose how you would like to install the operating system

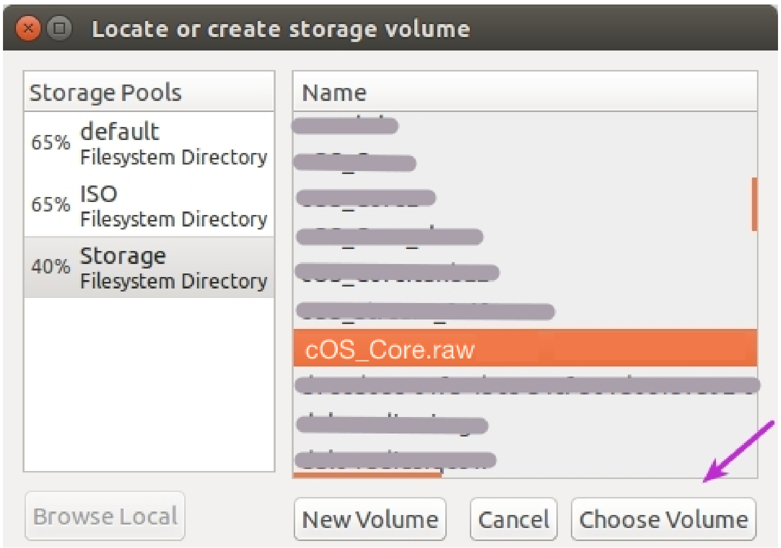
○ Local install media (ISO image or CDROM)

○ Network Install (HTTP, FTP, or NFS)

○ Network Boot (PXE)

● Import existing disk image

4. Select the cOS Core image file and select **Choose Volume**. For this description, it is assumed that the image file has the name *cOS_Core.qcow2* but the actual filename will include a version number.



Locate or create storage volume

Storage Pools	
65%	default Filesystem Directory
65%	ISO Filesystem Directory
40%	Storage Filesystem Directory

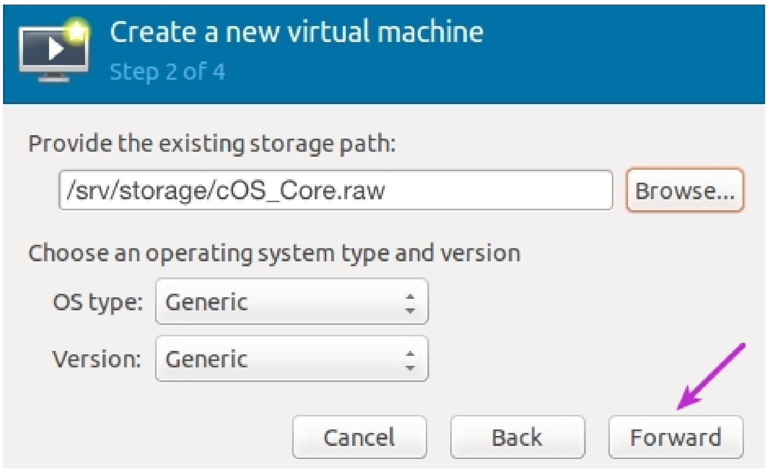
Name

[redacted]

cOS_Core.raw

[redacted]

5. In the next step, leave the **OS type** and **Version** as *Generic* and select **Forward**.



Create a new virtual machine

Step 2 of 4

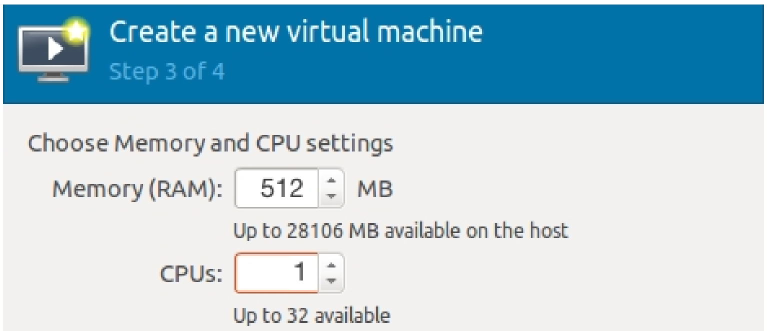
Provide the existing storage path: /srv/storage/cOS_Core.raw

Choose an operating system type and version

OS type: Generic

Version: Generic

6. Set the amount of RAM and the number of CPUs that are required. Only one CPU is required. cOS Core requires 128 Mbytes of memory as an absolute minimum. However, some memory demanding cOS Core features such as Anti-Virus scanning, IDP and Application Control will not be able to function in 128 Mbytes. For this reason, 512 Mbytes is recommended. The maximum possible is 4096 Mbytes. such as



Choose Memory and CPU settings

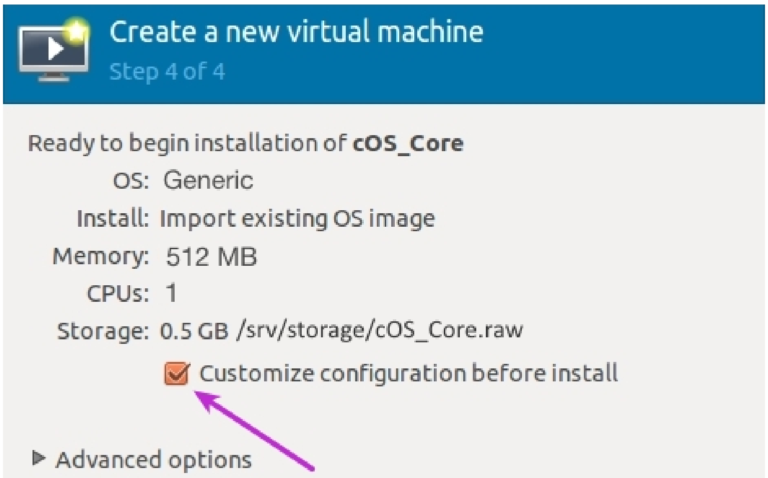
Memory (RAM): 512 MB

Up to 28106 MB available on the host

CPUs: 1

Up to 32 available

7. Check the option **Customize configuration before install** and then select **Finish** to exit the wizard.



Ready to begin installation of **cOS_Core**

OS: Generic

Install: Import existing OS image

Memory: 512 MB

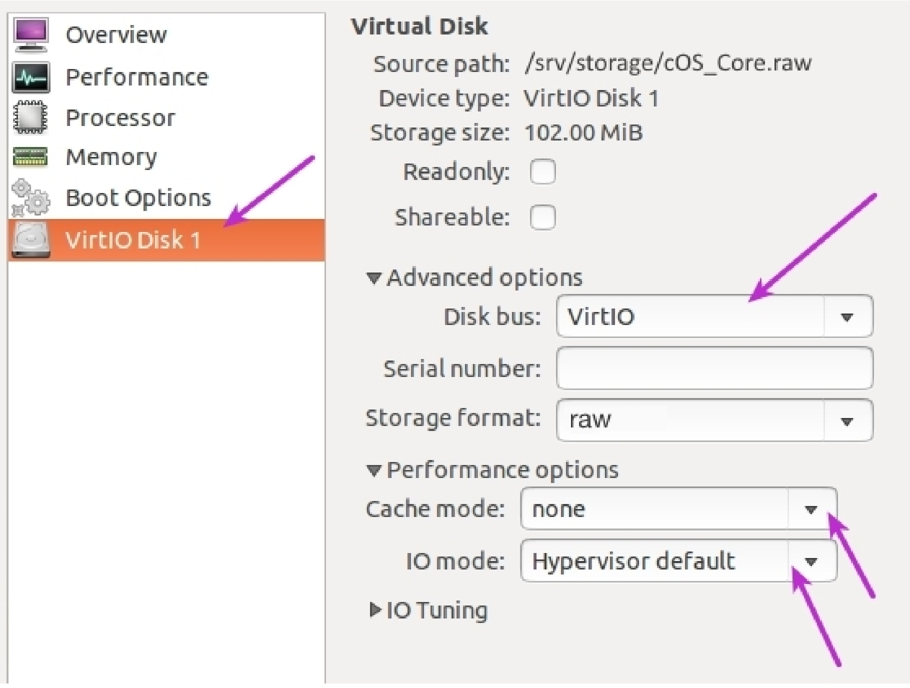
CPUs: 1

Storage: 0.5 GB /srv/storage/cOS_Core.raw

☑ Customize configuration before install

8. Select the disk from the navigation menu and make sure the following values are entered for the disk driver:

   - **Disk bus:** VirtIO
   - **Cach mode:** none
   - **IO mode:** Hypervisor default



Virtual Disk

Source path: /srv/storage/cOS_Core.raw

Device type: VirtIO Disk 1

Storage size: 102.00 MiB

Readonly: □

Shareable: □

▼ Advanced options

Disk bus: VirtIO

Serial number:

Storage format: raw

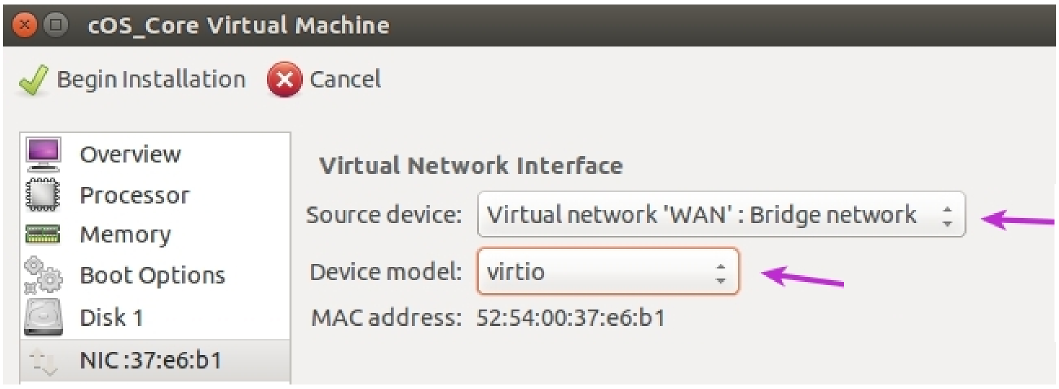
▼ Performance options

Cache mode: none

IO mode: Hypervisor default

▶ IO Tuning

9. Set the **Source device** to the defined WAN bridge and set the **Device model** to *virtio*. Select **Apply**, then select **Add Hardware**.



Virtual Network Interface

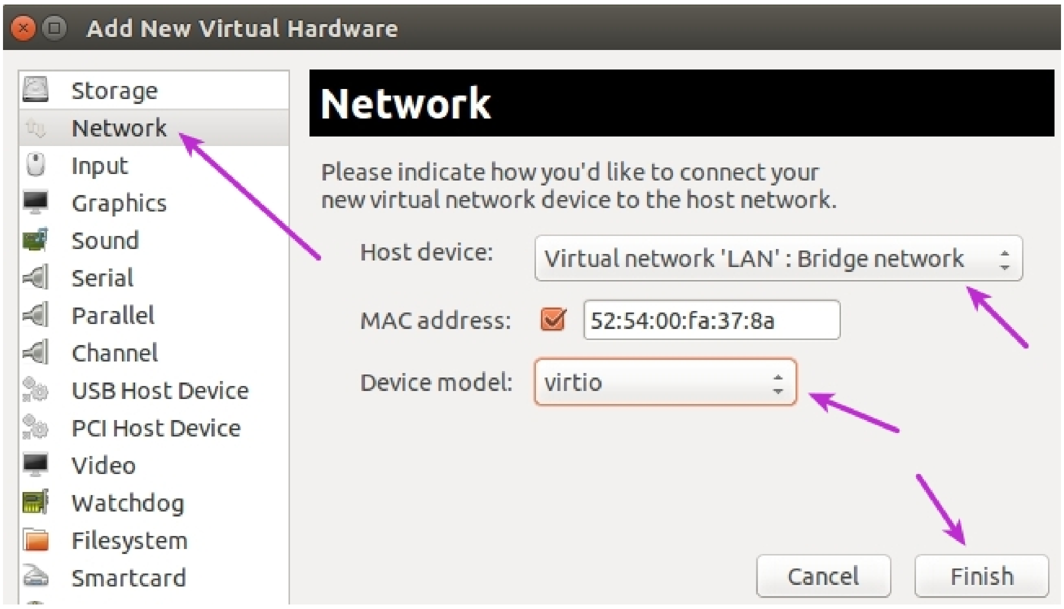
Source device: Virtual network 'WAN' : Bridge network

Device model: virtio

MAC address: 52:54:00:37:e6:b1

10. Select **Network** in the navigation menu and then set **Host device** to the defined LAN bridge and **Device model** to *virtio*.

    Repeat this process for the third interface.



Network

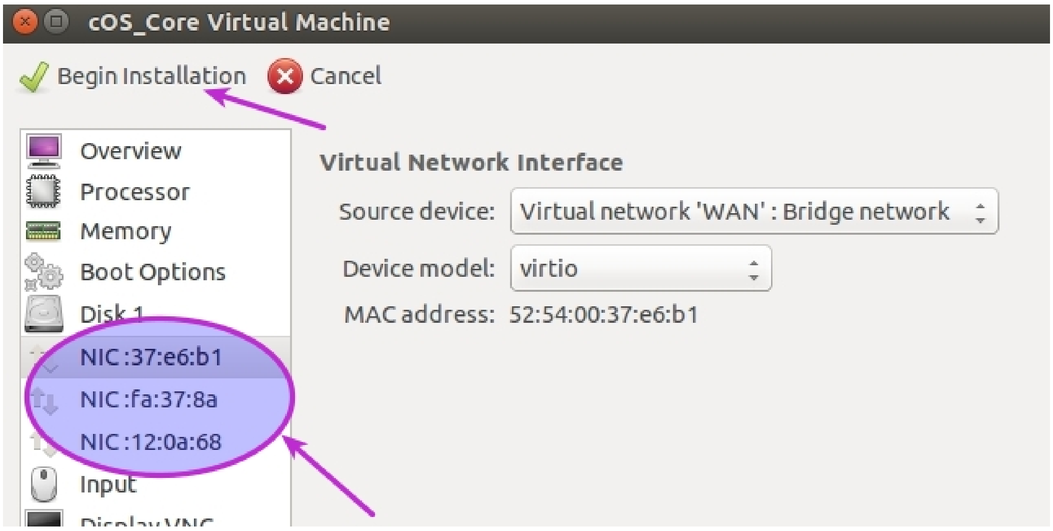
Please indicate how you'd like to connect your new virtual network device to the host network.

Host device: Virtual network 'LAN' : Bridge network

MAC address: ☑ 52:54:00:fa:37:8a

Device model: virtio

11. There are now three network interfaces attached to the virtual machine so **Begin installation** can now be selected. This will also boot up the virtual machine.



Virtual Network Interface

Source device: Virtual network 'WAN' : Bridge network

Device model: virtio

MAC address: 52:54:00:37:e6:b1

3.2. Script Setup

An example script for this setup called **prepare.sh** can be downloaded from:

http://github.com/Clavister

This script is written in *bash* and is not supported by Clavister. It is provided only as a reference script for cOS Core setup under KVM and it can be freely used, modified or redistributed under the GPL open source license. As far as Clavister is aware, the script is suitable for KVM running under most Linux distributions.

The process for creating the virtual machine with the script can be summarized as follows:

- Run the script *prepare.sh*. This go through a series of questions to create an XML file for initial configuration of the virtual machine. The script will optionally create the virtual machine using this file. One of the key tasks performed by the script is to map cOS Core's virtual Ethernet interfaces to networking bridges.

- Use the *virsh define* command to create the virtual machine using the created XML file as input if the administrator chose not to do it with the script.

Install *bridge-utils* or *Open vSwitch*

Either *bridge-utils* or *Open vSwitch* must be installed for networking functions. Both cannot be installed at the same time. If the virtual firewall is going to be part of a cOS Core HA cluster then *Open vSwitch* must be installed. However, Open vSwitch can also be used for standalone virtual firewalls.

The *prepare.sh* script will ask which of the two is installed and configure the networking accordingly.

Detailed Steps for Virtual Machine Definition

Once the Linux system has been set up with the required software installed, the series of steps for creating a cOS Core virtual machine are as follows:

1. Download the cOS Core distribution package file from the Clavister *Customer Web* to a local workstation computer. The Clavister website will require customer login credentials.

2. Upload the following files to the Linux computer's disk using the *Secure Copy* (SCP) protocol and make a note of their location.

   i. The cOS Core image file for KVM which can be downloaded by logging in to the Clavister website.

   ii. The script *prepare.sh* or a modified version of it.

   Many SCP clients can be used for this. For example, open source *puTTY*.

3. Open a console to access Linux. Note that the script **must** be run as root and the script will check that this is the case.

4. Change the working directory to be the location of the uploaded files then run the script *prepare.sh* using the command:

```
[root@linux]# ./prepare.sh
```

Optionally, the filename of the cOS Core virtual machine image can also be specified in the command line:

```
[root@linux]# ./prepare.sh <vm_image_filename>
```

When it runs, the script will prompt for the following:

i. **The Clavister product:** The script can be used with all Clavister's security products. Select *cOS Core* for this question.

ii. **The firewall name:** The name of the virtual machine and also the name of the XML generated by the script. This is the name that will be displayed when using Virtual Machine Manager.

iii. **Networking**: The administrator must tell the script if *bridge-utils* or *Open vSwitch* is being used for networking. If the selected networking package is not detected, the script will terminate.

iv. **The interface mapping**: A default mapping of cOS Core virtual Ethernet interfaces to networking bridges will be performed by the script and displayed. The script will ask if this mapping should be changed, allowing the administrator to select an alternative mapping.

v. **Creating the virtual firewall**: A virtual machine running cOS Core can be created by the script. If the administrator chooses not to do this, it must be done manually using the *virsh* utility as described later. A reason not to let the script create the virtual machine is if the XML configuration file is to be checked and possibly altered manually.

6. After the script completes and if the administrator chose not to create the virtual machine, an XML file will have been created which is then used to create it manually. Assume that the name chosen for the firewall is *my_firewall*. The XML configuration file created by the script will be *my_firewall.xml*. The following Linux command will create the virtual machine:

```
[root@linux]# virsh define my_firewall.xml
```

The XML file can be examined and edited manually before this step but it is recommended to make changes later.

Changing the Virtual Machine Configuration

The initial configuration parameters of the virtual machine created will be those specified in the configuration XML file created by the script but these can be changed later as required. For example, the amount of RAM memory allocated may need to be increased. Making these changes on an existing virtual machine is described in *Chapter 4, Configuring Virtual Machines*.

3.3. The Management Interface

The KVM Console

When cOS Core starts, KVM will display a console which represents the console that is normally directly connected to the local console port of a physical Clavister Next Generation Firewall. This console is accessed by using VNC to connect to the IP address and port previously specified when running the script *prepare.sh*.

This console displays output from cOS Core exactly as it would be displayed with a non-virtual Clavister Next Generation Firewall. It will show the initial startup sequence output and this can be interrupted, if required, by key presses in order to enter the boot menu. After startup, the KVM console can be used to issue CLI commands to configure cOS Core further and this is described in *Section 6.4, "Manual CLI Setup"*.

Tip: Changing focus back from the KVM console

KVM will keep focus in the console window after clicking it. Use the key combination ***Ctrl-Alt*** *to release this focus.*

The Default Virtual Ethernet Interfaces

By default, the standard cOS Core installation provides three virtual Ethernet interfaces. To function, these virtual NICs **must be mapped** to the correct bridge or physical Ethernet interface by changing the *Source device* property for the interface using Virtual Machine Manager (*virt-manager*). Doing this is described in *Chapter 4, Configuring Virtual Machines*. The *Device model* property will remain as the default value of *virtio*

cOS Core assigns the following default logical names to the virtual interfaces:

- Interface names: *Ifn*. For example, the first interface is *If1*.
- IP address objects: *Ifn_ip*. For example, the first address object is *If1_ip*.
- Netmask IP objects: *Ifn_net*. For example, the first netmask is *If1_net*.

Connecting to the Virtual Clavister Next Generation Firewall

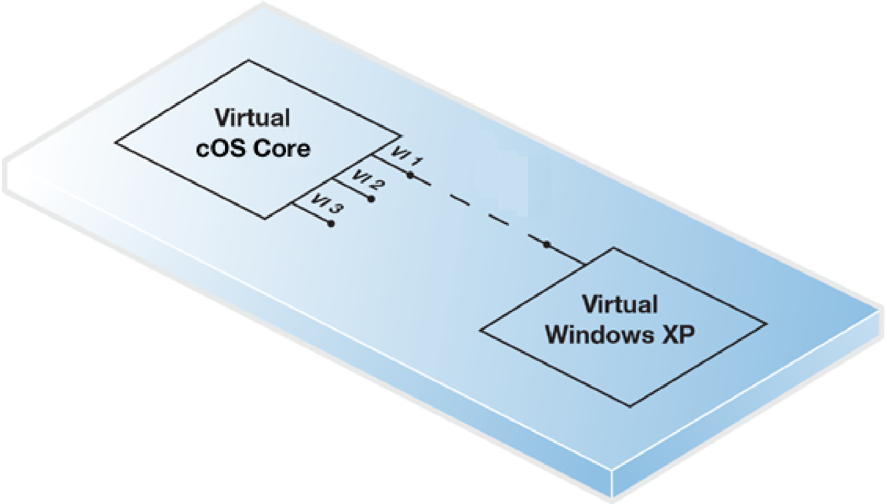
The first virtual Ethernet interface, *If1*, will be assigned the IP address **192.168.1.1** by cOS Core. This is the default cOS Core management interface and connection to it can be done from a web browser (using the cOS Core Web Interface) or SSH client (using the cOS Core CLI) just as it is done with a non-KVM installation.

The workstation running the web browser or SSH client can be one of the following:

- **A virtual workstation running under the same KVM host.**

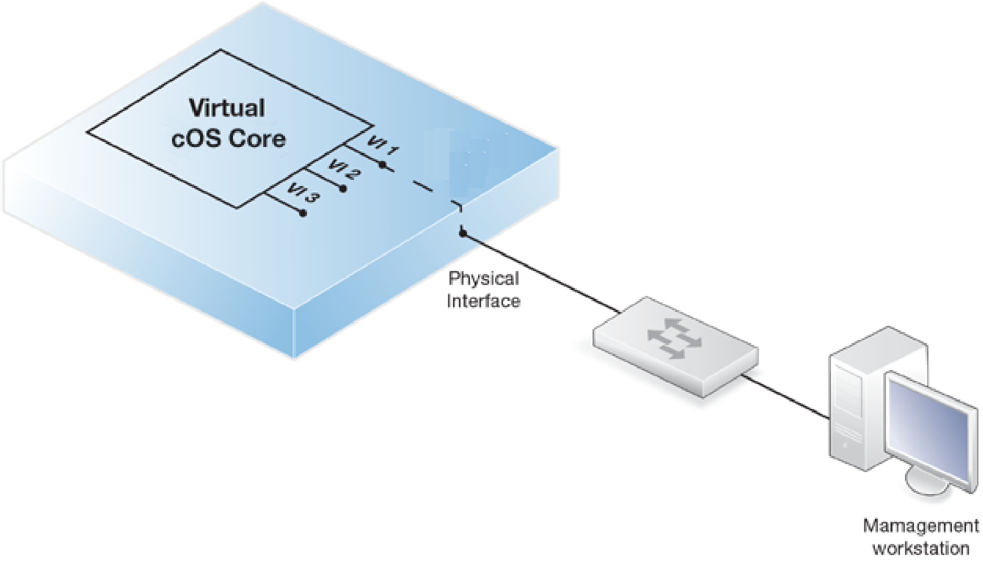
  In this case, a Linux or Open vSwitch bridge can be used to connect the virtual Ethernet interface with a virtual Ethernet interface on the virtual workstation. The virtual workstation might be, for example, a Windows XP installation as shown below.

  For this option to function, KVM must be configured so that the virtual Ethernet interface on both cOS Core and the workstation are on the same bridge.



- **A physically separate workstation computer.**

  In this case, a *macvtap* adapter can be configured to connect the virtual Ethernet interface to a physical interface. Physical connection is then made between that physical interface and an interface on a physically separate workstation computer.



In both the above cases, the real or virtual workstation PC needs its connecting Ethernet interface configured with an IP address on the same network as the cOS Core interface. Once this is done, the management workstation and the Clavister Next Generation Firewall can communicate and initial cOS Core setup can then be performed in exactly the same way as a non-virtual firewall. This is described next in *Chapter 6, Configuring cOS Core*.

Setup with Multiple Virtual Clavister Next Generation Firewalls

When there are multiple virtual machines running cOS Core under one KVM host, the IP address of the management virtual Ethernet interface must be different for the different virtual machines if administration is to be done through the Web Interface or SSL client.

The recommended way to change the management interface IP address is to enter CLI commands into the cOS Core console which is displayed by KVM after cOS Core starts. The commands to do this are as follows:

1. Set the IP address of the default management interface *If1_ip*. In this example, it will be set to *10.0.0.1*:

   ```
   Device:/> set Address IP4Address
                           InterfaceAddresses/If1_ip
                           Address=10.0.0.1
   ```

2. Now set the network of the interface. This object has the name *If1_net*.

   ```
   Device:/> set Address IP4Address
                           InterfaceAddresses/If1_net
                           Address=10.0.0.0/24
   ```

3. As a check, the current management rule for HTTP access can be displayed:

   ```
   Device:/> show RemoteManagement RemoteMgmtHTTP
   ```

These steps should then be followed by an *activate* and then a *commit* command to deploy the changes.

These same steps could be performed through the Web Interface but as soon as the changes are committed, the administrator has 30 seconds to log back in to cOS Core before the changes are undone and cOS Core reverts to the previous configuration.

Chapter 4: Configuring Virtual Machines

Once the cOS Core virtual machine disk image is imported (this is described in the previous chapter), the KVM virtual machine environment will have a set of default parameters. For example, the virtual interfaces available to cOS Core. This section describes how these parameters should be configured.

Important: Read the previous chapter first

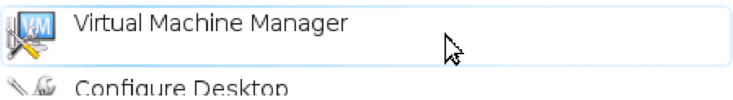
*This chapter deals with changing the default virtual machine configuration **after** it has been imported. Importing is described in **Chapter 3, Importing Virtual Machines** and that should be read first.*

Displaying the Current Configuration

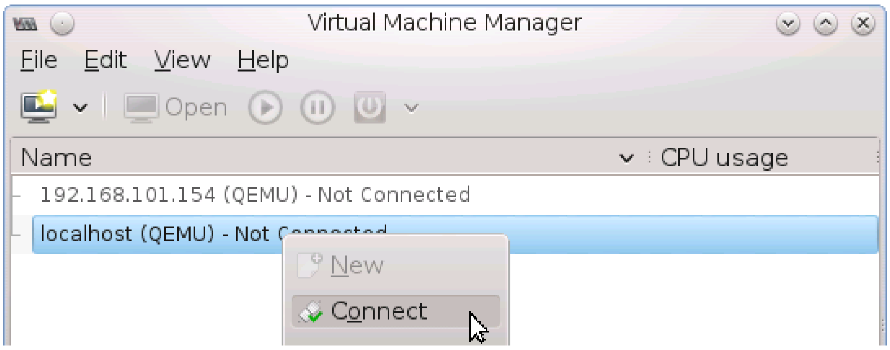
The current KVM configuration of a virtual machine can be displayed with the following steps:

1. Use a client console to access Linux and start the *virt-manager* software.

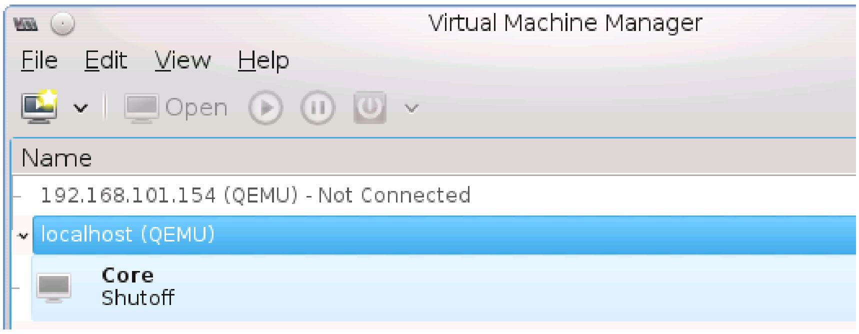
2. Start the *virt-manager* software. The *virt-manager* virtual machine manager software will be used throughout this guide but alternative virtual management software may be used.



3. Connect to KVM by right-clicking on *localhost (QEMU)* and selecting the *Connect* menu option (or double-click on the *localhost* line).

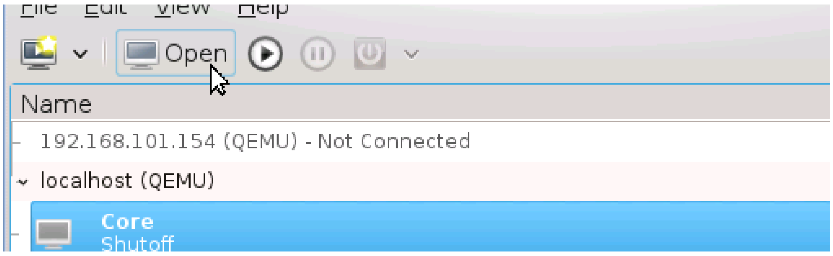

Virtual Machine Manager
File Edit View Help
Open
Name
CPU usage
192.168.101.154 (QEMU) - Not Connected
localhost (QEMU) - Not Connected
New
Connect

4. A list of currently defined virtual machines will be shown.

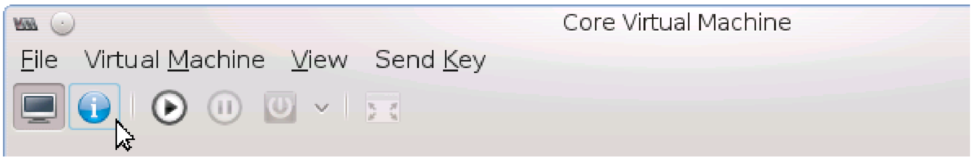


Changing the Configuration

1. To display and edit the currently selected virtual machine's configuration, first press the *Open* button in the toolbar.

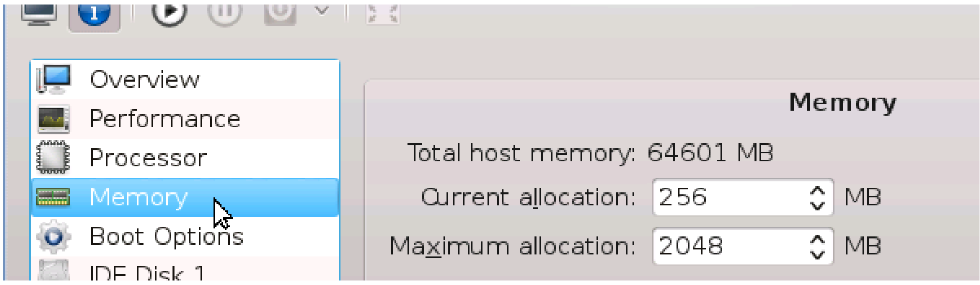


2. The status dialog for this virtual machine will display. Press the information button in the toolbar.



3. The configuration of the virtual machine will now be displayed.

4. To change a configuration parameter, select it in the left hand navigation list and then alter the displayed values. For example, changing the default RAM memory allocation is shown below.



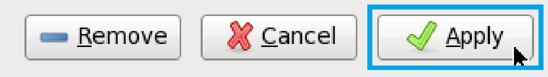
Memory

Total host memory: 64601 MB

Current allocation: 256 MB

Maximum allocation: 2048 MB

5. Save any configuration changes by pressing the *Apply* button.



Chapter 5: Adding Extra Interfaces

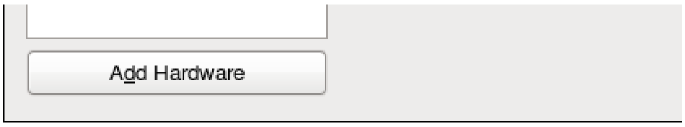
The default virtual machine created by the script *prepare.sh* has three virtual interfaces configured for cOS Core. These have logical cOS Core names **If1**, **If2** and **If3**.

If more virtual interfaces are required, these can be added later but must be manually configured. This chapter explains how extra interfaces can be added so they are correctly configured.

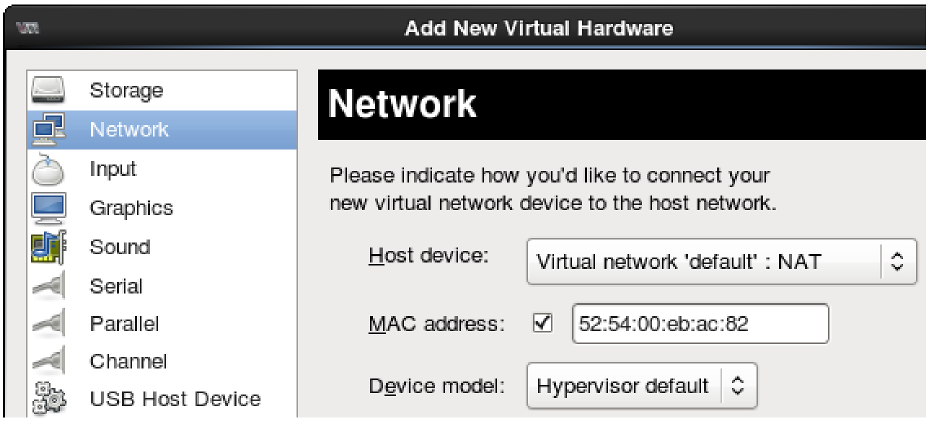
The steps are as follows:

1. Open *virt-manager* and open the configuration dialog for the virtual machine. Doing this is described previously in *Chapter 4, Configuring Virtual Machines*.

2. Press the *Add Hardware* button at the bottom of the configuration dialog.



3. The *Add New Virtual Hardware* dialog will be displayed. Select *Network* from the options on the left.



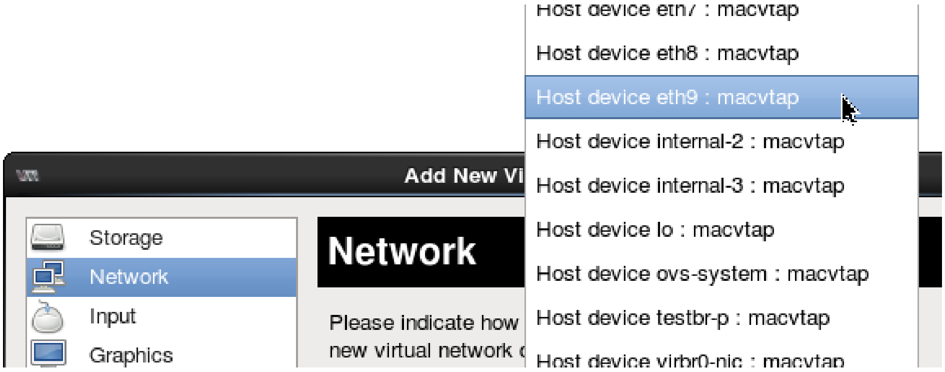
Please indicate how you'd like to connect your new virtual network device to the host network.

Host device: Virtual network 'default' : NAT

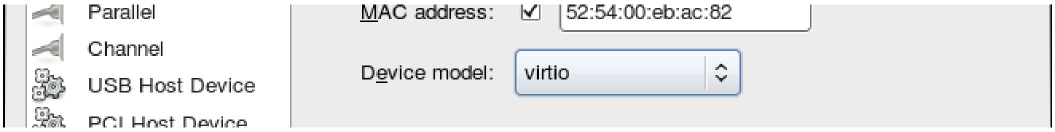
MAC address: ☑ 52:54:00:eb:ac:82

Device model: Hypervisor default

4. Now select the *Host Device* which will be the physical interface or bridge to be mapped to this virtual interface. In the screenshot below, a physical Ethernet interface called **eth9** will be selected for the mapping.


Host device eth7 : macvtap
Host device eth8 : macvtap
Host device eth9 : macvtap
Host device internal-2 : macvtap
Host device internal-3 : macvtap
Host device lo : macvtap
Host device ovs-system : macvtap
Host device testbr-p : macvtap
Host device virbr0-nic : macvtap
Add New Vi
Storage
Network
Input
Graphics
Network
Please indicate how
new virtual network c

5. Next, select the *Device Model* to have the value *Virtio*.



MAC address: ☑ 52:54:00:eb:ac:82

Device model: virtio

6. Close the new hardware dialog by pressing the *Finish* button.

7. The new interface will now appear in the list at the left of the configuration dialog. Press the *Apply* button to save the changed configuration.

8. Although the virtual interface has now been added to the virtual machine, cOS Core will not automatically add it to the current configuration. To add the interface, run the following cOS Core CLI command:

```
Device:/> pciscan -cfgupdate
```

The output from this command will confirm that a new interface has been added. If it is the first added and no previous ones have been deleted it should have the logical name **If4**.

Follow the *pciscan* command with the *activate* and *commit* CLI commands to save the configuration changes.

Caution: Adding and deleting cOS Core interfaces

cOS Core allows logical interfaces to be deleted. If this is done the ordering of subsequently added logical interfaces can become unpredictable and may not necessarily have the first logical name that is available. For example, if cOS Core interface ***If2*** *is deleted from the configuration, the next interface added using* ***pciscan*** *may not become* ***If2****.*

Chapter 6: Configuring cOS Core

6.1. Management Workstation Connection

The Default Management Interface

After first time startup, cOS Core scans the available Ethernet interfaces and makes management access available on the first interface found and assigns the IPv4 address **192.168.1.1** to it.

With installation under KVM, the default management interface is the cOS Core **If1** interface.

Alternative cOS Core Setup Methods

Initial cOS Core software configuration can be done in one of the following ways:

- **Through a web browser.**

  A standard web browser running on a standalone computer (also referred to as the *management workstation*) can be used to access the cOS Core *Web Interface*. This provides an intuitive graphical interface for cOS Core management. When this interface is accessed for the first time, a *setup wizard* runs automatically to guide a new user through key setup steps. The wizard can be closed if the administrator wishes to go directly to the Web Interface to perform setup manually.

  The wizard is recommended for its simplification of initial setup and is described in detail in *Section 6.2, "Web Interface and Wizard Setup"*.

- **Through a terminal console using CLI commands.**

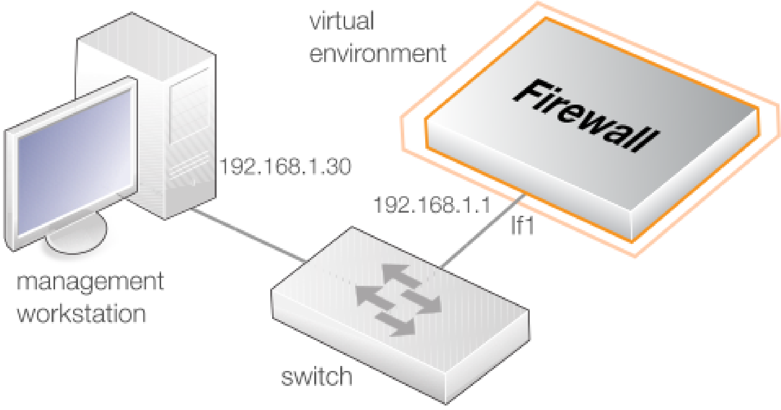
  The setup process can alternatively be performed using console CLI commands and this is described in *Section 6.4, "Manual CLI Setup"*. The CLI allows step by step control of setup and should be used by administrators who fully understand both the CLI and setup process.

  CLI access can be remote, across a network to a cOS Core interface using a similar connection to that used with the Web Interface. Alternatively, CLI access can be direct, through the KVM console window.

Network Connection Setup

For setup using the Web Interface or using remote CLI, a management workstation computer must be first physically connected to cOS Core across a network. This connection is described previously in *Chapter 3, Importing Virtual Machines*.

The logical cOS Core management interface with KVM is **If1** and the corresponding physical Ethernet port associated with this should be connected to the same network as the management workstation (or a network accessible from the workstation via one or more routers). Typically the connection is made via a switch or hub in the network using a regular straight-through Ethernet cable as illustrated below.



For connection to the public Internet, one of the other interfaces should be connected to an ISP and this interface is sometimes referred to below and in the setup wizard as the *WAN* interface.

Workstation Ethernet Interface Setup

Traffic is able to flow between the designated workstation interface and the Clavister Next Generation Firewall interface because they are on the same IP network. This means the workstation interface must be first assigned the following static IPv4 addresses:

- **IP address:** *192.168.1.30*
- **Subnet mask:** *255.255.255.0*
- **Default gateway:** *192.168.1.1*

Tip: Using another workstation interface IP address

The IPv4 address assigned to the management workstation's Ethernet interface, could be any address from the ***192.168.1.0/24*** *network. However, the IP chosen must be different from* ***192.168.1.1*** *which is used by cOS Core's default management interface.*

The following appendices at the end of this guide describe how to set up the management workstation IP with different platforms:

- ***Appendix A, Windows 7 IP Setup***.
- ***Appendix B, Windows 8/8.1/10 IP Setup***.
- ***Appendix C, Apple Mac IP Setup***.

6.2. Web Interface and Wizard Setup

This section describes the setup when accessing cOS Core for the first time through a web browser. The cOS Core user interface accessed in this way is called the *Web Interface*. It assumes that a physical network connection has been set up from a management computer to the default management Ethernet interface, as described in *Section 6.1, "Management Workstation Connection"*.

Note: Some browser screenshot images have been modified

Some of the screenshot images in this section have been modified from original screenshots to suit this document's page format. However, all relevant details in the images have been preserved.

Connect to cOS Core By Browsing to *https://192.168.1.1*

Using a standard web browser, enter the address *https://192.168.1.1* into the navigation window, as shown below.

https://192.168.1.1

Note: HTTP access is disabled for cOS Core 11.01 and later

For cOS Core version 11.01 and later, HTTP management access is disabled in the default configuration and HTTPS must be used. Unencrypted access with HTTP can be enabled by the administrator but this is not recommended.

Troubleshooting

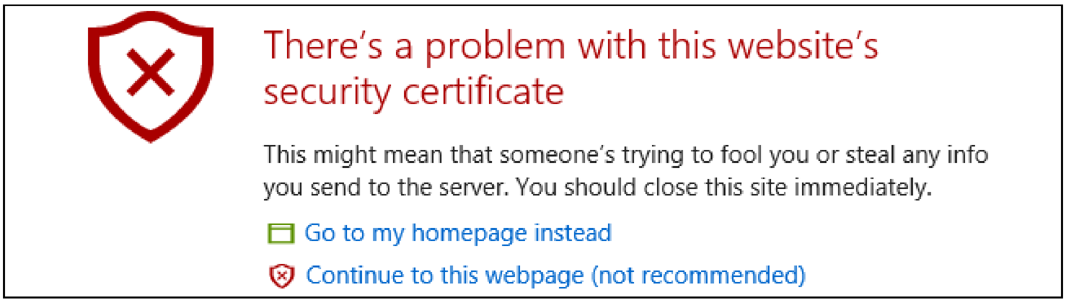
If there is no response from cOS Core and the reason is not clear, refer to the checklist in *Section 6.6, "Setup Troubleshooting "*.

Important: Do not access cOS Core via a proxy server

Make sure the web browser doesn't have a proxy server configured for the cOS Core management IP address.

The cOS Core Self-signed Certificate

When responding to the first *https://* request in a browser session, cOS Core will send a self-signed certificate to the browser. All browsers will automatically flag this self-signed certificate as posing a potential security risk. In the latest Microsoft browser, the following error message will be displayed in the browser window.



There's a problem with this website's security certificate

This might mean that someone's trying to fool you or steal any info you send to the server. You should close this site immediately.

Go to my homepage instead
Continue to this webpage (not recommended)

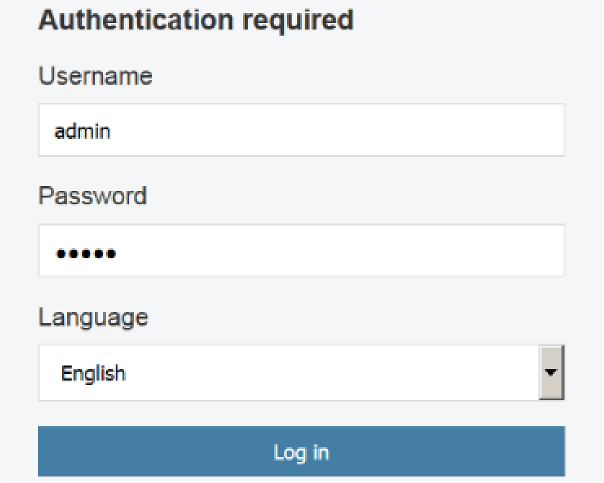
The browser should now be told to accept the Clavister certificate by choosing the option to continue.

Note: Sending a CA signed certificate can be configured

It is possible to configure cOS Core to use a CA signed certificate instead of its default self-signed certificate for the management login. Doing this is described in the cOS Core Administration Guide.

The Login Dialog

cOS Core will next respond like a web server with the initial login dialog page, as shown below.



Authentication required

Username: admin

Password: •••••

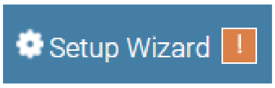
Language: English

The available Web Interface language options are selectable at the bottom of this dialog. This defaults to the language set for the browser if cOS Core supports that language.

Enter the administrator username as **admin** and use the default password **admin**.

Starting the Setup Wizard

After logging in for the first time, the Web Interface will appear and the cOS Core setup wizard should begin automatically as a popup window. If the wizard is blocked by the browser, it can be started manually by pressing the *Setup Wizard* button in the Web Interface toolbar (shown below).



Once the wizard is started, the first dialog displayed is the wizard welcome screen which is shown below.

Clavister Setup Wizard

WELCOME

Welcome to the Clavister Setup Wizard.

Canceling the Wizard

The setup wizard can be canceled at any point before the final *Activate* screen and run again by choosing the *Setup Wizard* option from the Web Interface toolbar. Once any configuration changes have been made and activated, either through the wizard, Web Interface or CLI, then the wizard cannot be run since the wizard requires that cOS Core has the factory defaults.

The Wizard Assumes Internet Access will be Configured

The wizard assumes that Internet access will be configured. If this is not the case, for example if the Clavister Next Generation Firewall is being used in *Transparent Mode* between two internal networks, then the configuration setup is best done with individual Web Interface steps or through the CLI instead of through the wizard.

Advantages of the Wizard

The wizard makes setup easier because it automates what would otherwise be a more complex set of individual setup steps. It also reminds you to perform important tasks such as setting the date and time and configuring a log server.

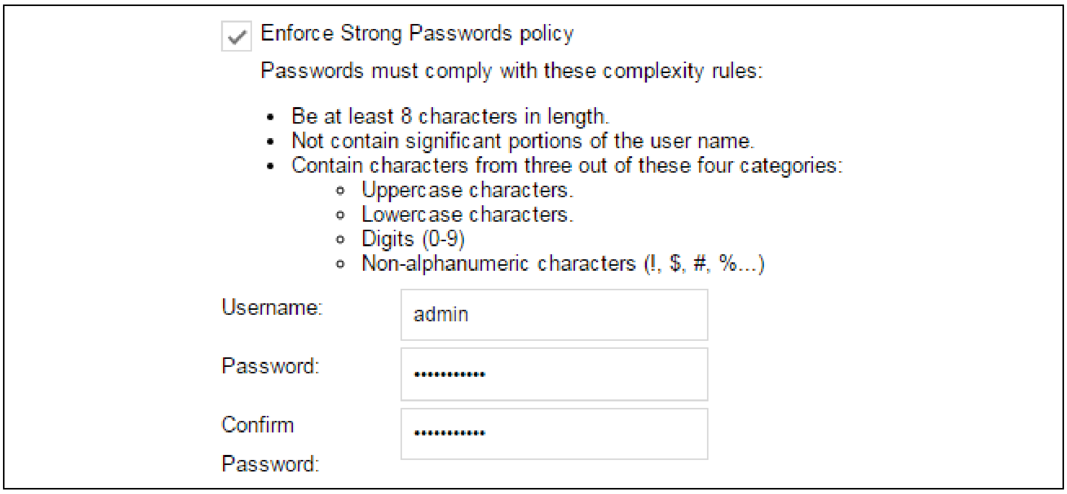
The steps that the wizard goes through after the welcome screen are listed next.

Wizard step 1: Enter a new admin password and optionally change the username

The first step in setup with the wizard is to enter a new password for the *admin* user. Always doing this is recommended. The *admin* username can also be changed if this is required. The next screenshot shows this step.

The *Enforce Strong Passwords* option is only present in cOS Core versions 11.05 and later. This is a global setting that will enforce the listed strong passwords rules for **all** users in any local user database in the configuration. If required, this option can be disabled later. It is recommended to leave this option enabled, which means that the default password of *admin* must be changed to a conforming strong password before the wizard can move on to the next step.

Note that restoring cOS Core to factory defaults will restore the original *admin/admin* credential combination for management access.



☑ Enforce Strong Passwords policy

Passwords must comply with these complexity rules:

- Be at least 8 characters in length.
- Not contain significant portions of the user name.
- Contain characters from three out of these four categories:
  - Uppercase characters.
  - Lowercase characters.
  - Digits (0-9)
  - Non-alphanumeric characters (!, $, #, %...)

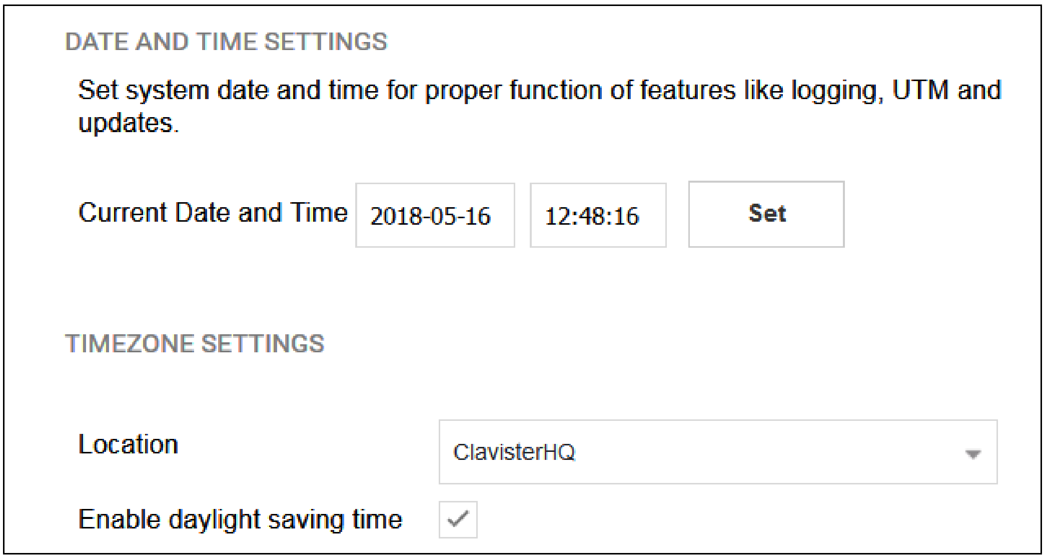
Username: admin

Password: •••••••••••

Confirm Password: •••••••••••

Wizard step 2: Set the date and time

Many cOS Core functions rely on an accurate date and time, so it is important that this is set correctly in the fields shown below. The default time zone location is *ClavisterHQ* which means the default location and time zone will be Stockholm. If this is not correct it should be changed to another location and timezone using the drop-down list.



DATE AND TIME SETTINGS

Set system date and time for proper function of features like logging, UTM and updates.

Current Date and Time 2018-05-16 12:48:16

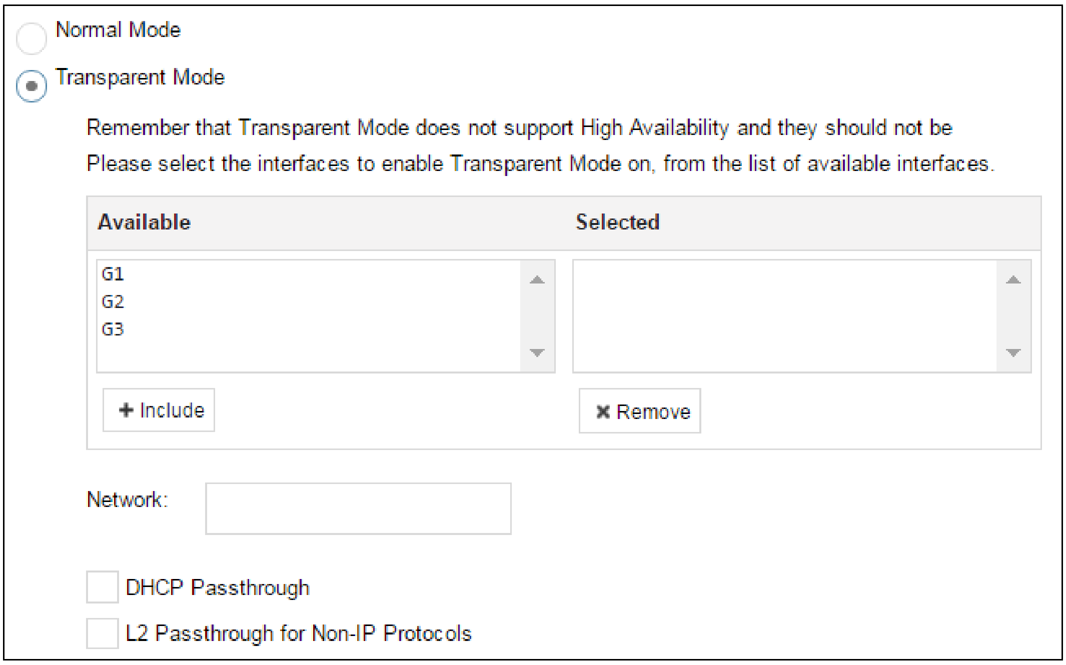
TIMEZONE SETTINGS

Location ClavisterHQ

Enable daylight saving time ☑

Wizard step 3: Select transparent mode interfaces

This step allows any transparent mode interfaces to be set up. If no transparent mode interfaces are required, leave this dialog in the default **Normal Mode** and go to the next step. Transparent mode interfaces can be configured at any time later, outside of the wizard.



○ Normal Mode

● Transparent Mode

Remember that Transparent Mode does not support High Availability and they should not be

Please select the interfaces to enable Transparent Mode on, from the list of available interfaces.

Available	Selected
G1 G2 G3	

Network:

□ DHCP Passthrough

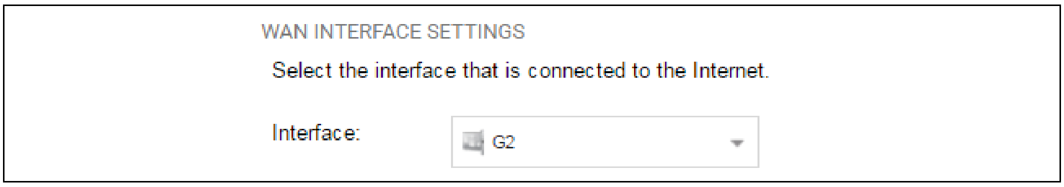
□ L2 Passthrough for Non-IP Protocols

Note: This step is only available with version 11.04 or later

The step to optionally set up transparent mode interfaces in the startup wizard is only available with cOS Core version 11.04 or later. Also, the available interface list shown above will vary according to the platform on which cOS Core is running.

Wizard step 4: Select the *WAN* interface

Next, you will be asked for the *WAN* interface that will be used to connect to your ISP for Internet access.



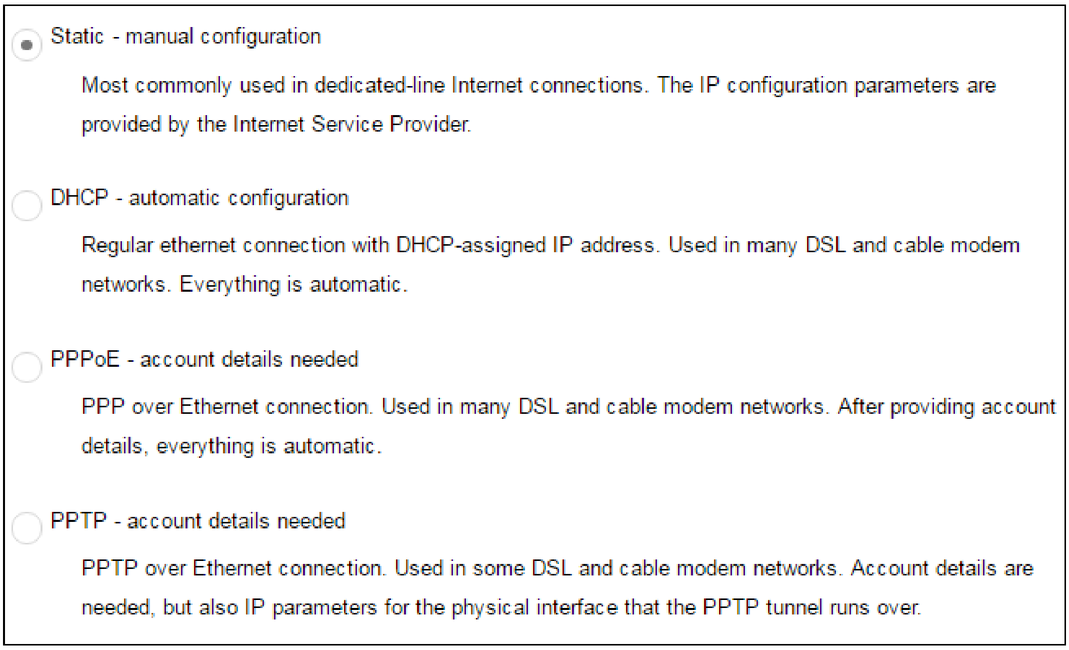
WAN INTERFACE SETTINGS

Select the interface that is connected to the Internet.

Interface: G2

Wizard step 5: Select the *WAN* interface settings

This step selects how the WAN connection to the Internet will function. It can be one of *Manual configuration*, *DHCP*, *PPPoE* or *PPTP* as shown below.



● Static - manual configuration

Most commonly used in dedicated-line Internet connections. The IP configuration parameters are provided by the Internet Service Provider.

○ DHCP - automatic configuration

Regular ethernet connection with DHCP-assigned IP address. Used in many DSL and cable modem networks. Everything is automatic.

○ PPPoE - account details needed

PPP over Ethernet connection. Used in many DSL and cable modem networks. After providing account details, everything is automatic.

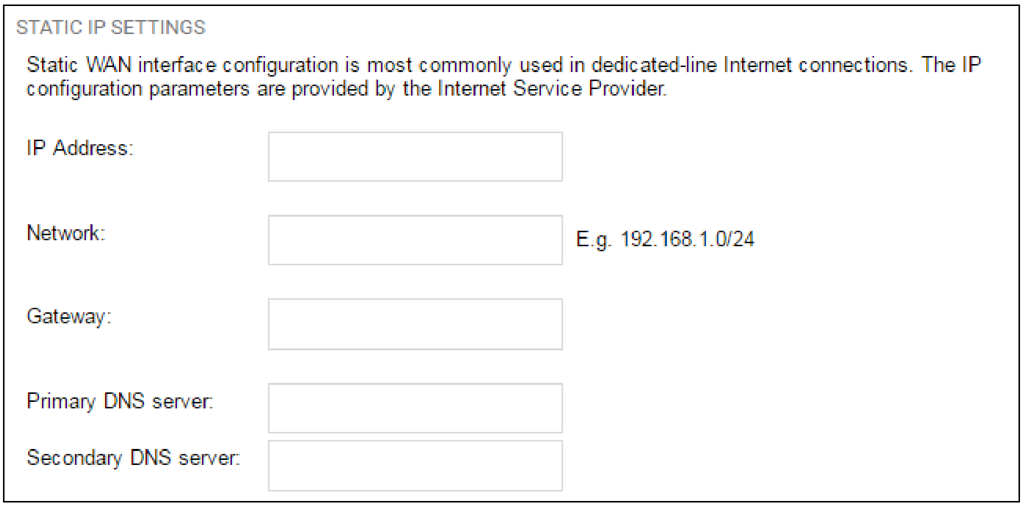
○ PPTP - account details needed

PPTP over Ethernet connection. Used in some DSL and cable modem networks. Account details are needed, but also IP parameters for the physical interface that the PPTP tunnel runs over.

These four different connection options are discussed next in the following subsections **5A** to **5D**.

- **5A. Static - manual configuration**

  Information supplied by the ISP should be entered in the next wizard screen. All fields need to be entered except for the *Secondary DNS server* field.



STATIC IP SETTINGS

Static WAN interface configuration is most commonly used in dedicated-line Internet connections. The IP configuration parameters are provided by the Internet Service Provider.

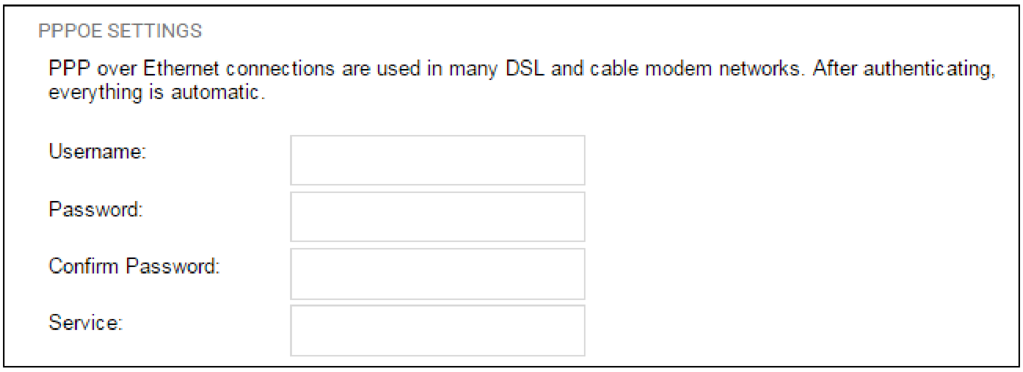
IP Address:		
Network:		E.g. 192.168.1.0/24
Gateway:		
Primary DNS server:		
Secondary DNS server:		

- **5B. DHCP - automatic configuration**

  All required IP addresses will automatically be retrieved from the ISP's DHCP server with this option. No further configuration is required for this so it does not have its own wizard screen.

- **5C. PPPoE settings**

  The username and password supplied by your ISP for PPPoE connection should be entered. The *Service* field should be left blank unless the ISP supplies a value for it.



PPPOE SETTINGS

PPP over Ethernet connections are used in many DSL and cable modem networks. After authenticating, everything is automatic.

Username:

Password:

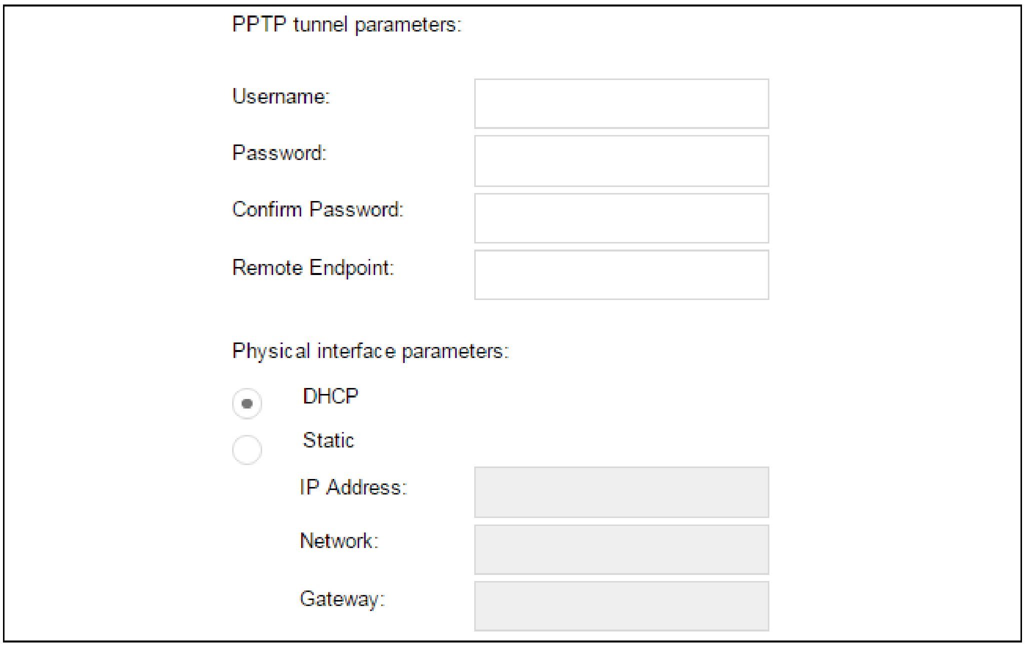
Confirm Password:

Service:

DNS servers are set automatically after connection with PPPoE.

- **5D. PPTP settings**

  The username and password supplied by your ISP for PPTP connection should be entered. If DHCP is to be used with the ISP then this should be selected, otherwise *Static* should be selected followed by entering the static IP address supplied by the ISP.



PPTP tunnel parameters:

Username:

Password:

Confirm Password:

Remote Endpoint:

Physical interface parameters:

● DHCP

○ Static

IP Address:

Network:

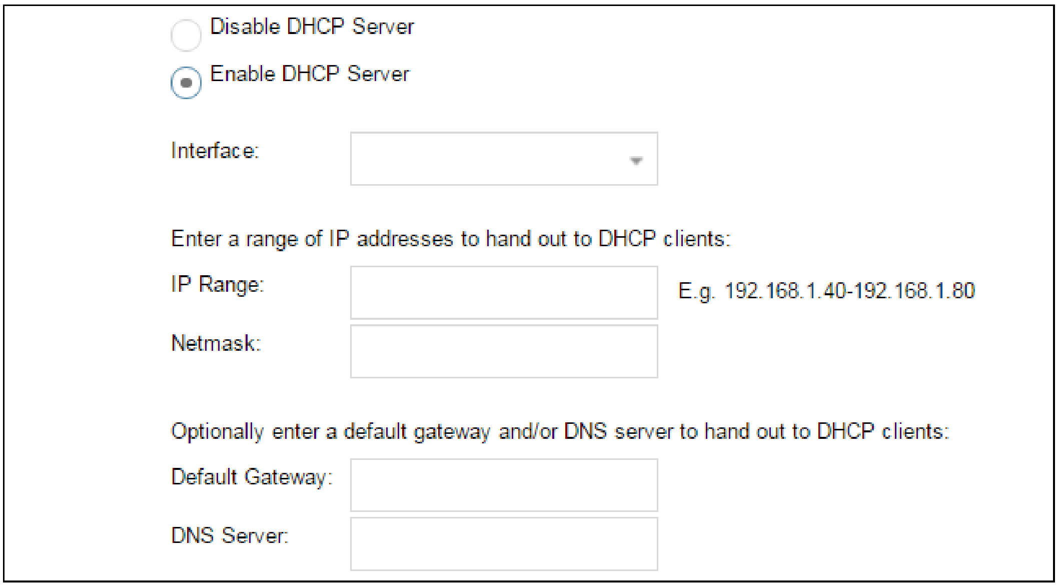
Gateway:

DNS servers are set automatically after connection with PPTP.

Wizard step 6: DHCP server settings

If the Clavister Next Generation Firewall is to function as a DHCP server, it can be enabled here in the wizard on a particular interface or configured later.

For example, the private IPv4 address range might be specified as *192.168.1.50 - 192.168.1.150* with a netmask of *255.255.255.0*.



○ Disable DHCP Server

● Enable DHCP Server

Interface:

Enter a range of IP addresses to hand out to DHCP clients:

IP Range: E.g. 192.168.1.40-192.168.1.80

Netmask:

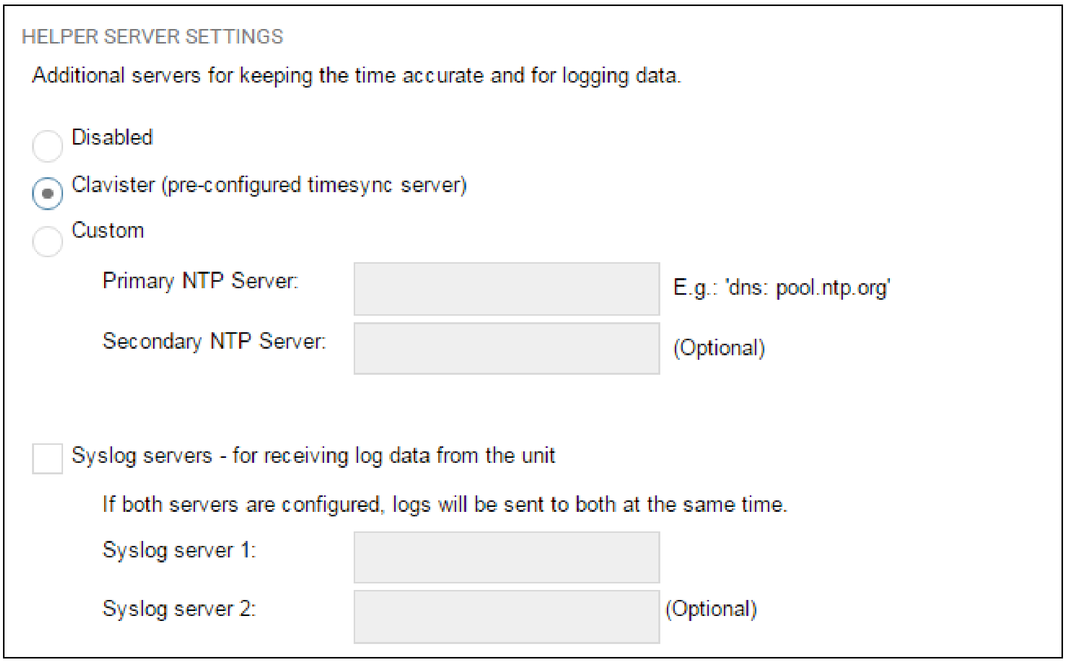
Optionally enter a default gateway and/or DNS server to hand out to DHCP clients:

Default Gateway:

DNS Server:

Wizard step 7: Helper server settings

Optional NTP and Syslog servers can be enabled here in the wizard or configured later. *Network Time Protocol* servers keep the system date and time accurate. Syslog servers can be used to receive and store log messages sent by cOS Core.



HELPER SERVER SETTINGS

Additional servers for keeping the time accurate and for logging data.

○ Disabled

● Clavister (pre-configured timesync server)

○ Custom

Primary NTP Server: E.g.: 'dns: pool.ntp.org'

Secondary NTP Server: (Optional)

□ Syslog servers - for receiving log data from the unit

If both servers are configured, logs will be sent to both at the same time.

Syslog server 1:

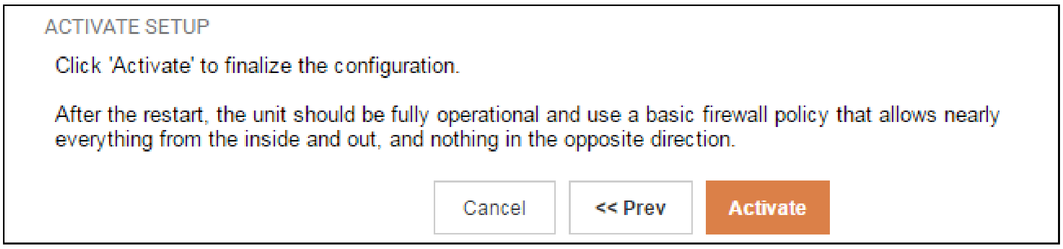
Syslog server 2: (Optional)

For the default gateway, it is recommended to specify the IP address *192.168.1.1* and the DNS server specified should be the DNS supplied by your ISP.

When specifying a hostname as a server instead of an IP address, the hostname should be prefixed with the string *dns:*. For example, the hostname *host1.company.com* should be entered as *dns:host1.company.com*.

Wizard step 8: Activate setup

The final step is to activate the setup by pressing the *Activate* button. After this step the Web Interface returns to its normal appearance and the administrator can continue to configure the system.



ACTIVATE SETUP

Click 'Activate' to finalize the configuration.

After the restart, the unit should be fully operational and use a basic firewall policy that allows nearly everything from the inside and out, and nothing in the opposite direction.

Running the Wizard Again

Once the wizard has been successfully finished and activated, it cannot be run again. The exception to this is if the Clavister Next Generation Firewall has its factory defaults restored in which case the unit will behave as though it were being started for the first time.

Uploading a License

Without a valid license installed, cOS Core operates in *demo mode* (demonstration mode) and will cease operations 2 hours after startup. To remove this restriction, a valid license must be uploaded to cOS Core. Doing this is described in *Section 6.5, "Installing a License"*

6.3. Manual Web Interface Setup

This section describes initial cOS Core configuration performed directly through the Web Interface, without using the setup wizard. Configuration is done as a series of individual steps, giving the administrator more direct control over the process. Even if the wizard is used, this section can also be read as a good introduction to using the Web Interface for configuring key aspects of cOS Core.

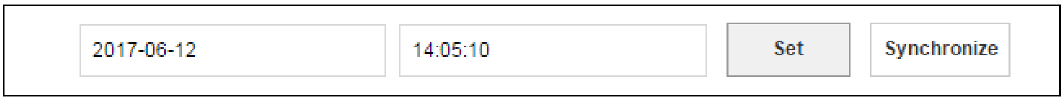
Ethernet Interfaces

The physical connection of external networks to the Clavister Next Generation Firewall is through the various *Ethernet interfaces* which are provided by the hardware platform. In a virtual environment, these are the *virtual interfaces* provided by the hypervisor. On first-time startup, cOS Core scans for these interfaces and determines which are available and allocates their names. The first interface detected in the scan always becomes the initial default management interface and this cannot be changed beforehand.

All cOS Core interfaces are logically equal for cOS Core and although their physical capabilities may be different, any interface can perform any logical function. With cOS Core under KVM, the virtual *If1* interface is always the management interface. Assuming the normal KVM total of 3 virtual interfaces, the other two virtual interfaces will automatically be given the names *If2* and *If3* by cOS Core. For this section, we will assume that the *If2* interface will be used for connection to the public Internet and the *If1* interface will also be used for connection to a protected, local network.

Setting the Date and Time

Many cOS Core functions rely on an accurate date and time, so it is important that this is set correctly. To do this, select **System > Device > Date and Time**. The current system time is displayed and this can be changed by selecting the date and time fields then manually entering the desired figures. Pressing the **Set** button will then set the time to the entered values.



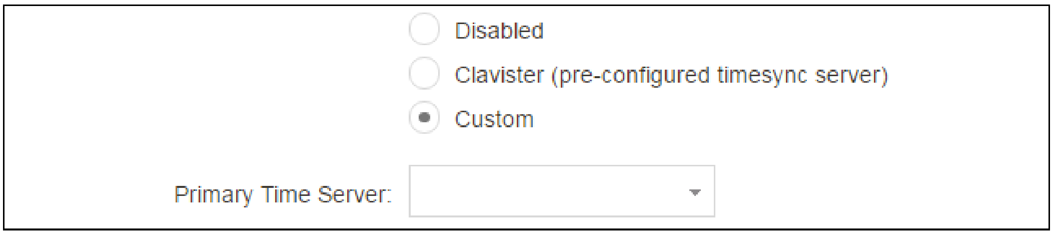
2017-06-12 14:05:10

Also choose the correct time zone from the **Location** drop-down list. The default location is *ClavisterHQ* which is Stockholm time.

Location: ClavisterHQ

Alternatively, the **Synchronize** button can be pressed to get the current date and time from the configured **Network Time Protocol** (NTP) server. In the default configuration, Clavister's own NTP server is automatically configured. However, accessing this server requires Internet access.

Configuring a custom NTP server configuration is shown below.



○ Disabled
○ Clavister (pre-configured timesync server)
● Custom

Primary Time Server:

Note: Specifying a URL for the time server

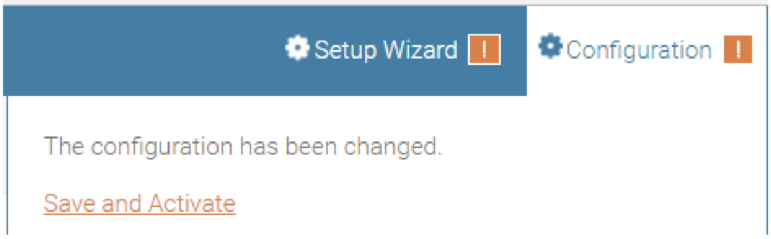
*For cOS Core versions prior to 12.00.09 a time server URL must have the prefix "**dns:**".*

*For version 12.00.09 and later, an **FQDN Address** object must be used instead of a direct URL reference. See the relevant cOS Core Administration Guide for more explanation.*

Once the values are set correctly, we can press the **OK** button to save the values while we move on to more steps in cOS Core configuration. Although changed values like this are saved by cOS Core, they do not become active until the entire saved configuration becomes the current and active configuration. We will look at how to do this next.

Activating Configuration Changes

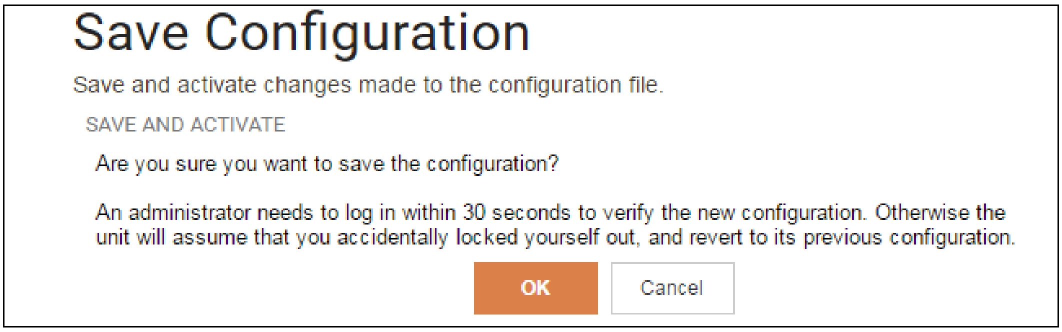
To activate any cOS Core configuration changes made so far, select the **Save and Activate** option from the **Configuration** menu (this procedure is also referred to as *deploying* a configuration).



The configuration has been changed.

Save and Activate

A dialog is then presented to confirm that the new configuration is to become the running configuration.



Save Configuration

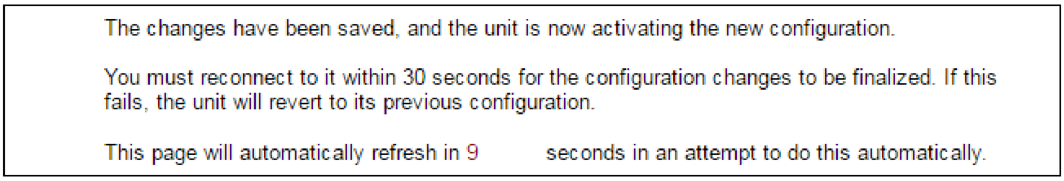
Save and activate changes made to the configuration file.

SAVE AND ACTIVATE

Are you sure you want to save the configuration?

An administrator needs to log in within 30 seconds to verify the new configuration. Otherwise the unit will assume that you accidentally locked yourself out, and revert to its previous configuration.

After clicking **OK**, cOS Core *reconfiguration* will take place and, after a short delay, the Web Interface will try and connect again to the firewall.

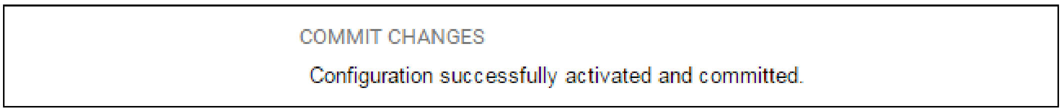


The changes have been saved, and the unit is now activating the new configuration.

You must reconnect to it within 30 seconds for the configuration changes to be finalized. If this fails, the unit will revert to its previous configuration.

This page will automatically refresh in 9 seconds in an attempt to do this automatically.

If no reconnection is detected by cOS Core within 30 seconds (this length of time is a setting that can be changed) then cOS Core will revert back to the original configuration. This is to ensure that the new configuration does not accidentally lock out the administrator. After reconfiguration and successful reconnection, a success message is displayed indicating successful reconfiguration.



COMMIT CHANGES

Configuration successfully activated and committed.

Reconfiguration is a process that the cOS Core administrator may initiate often. Normally, reconfiguration takes a brief amount of time and causes only a slight delay in traffic throughput. Active user connections through the Clavister Next Generation Firewall should rarely be lost.

Tip: How frequently to commit configuration changes

It is up to the administrator to decide how many changes to make before activating a new configuration. Sometimes, activating configuration changes in small batches can be appropriate in order to check that a small set of changes work as planned.

However, it is not advisable to leave changes uncommitted for long periods of time, such as overnight, since any system outage will result in these edits being lost.

Automatic Logout

If there is no activity through the Web Interface for a period of time (the default is 15 minutes), cOS Core will automatically log the user out. If they log back in through the same web browser session then they will return to the point they were at before the logout occurred and no saved (but not yet activated) changes are lost.

Setting Up Internet Access

Next, we shall look at how to set up public Internet access with the CLI. There are four options for setting up access which are listed below and then described in detail.

A. Static - manual configuration.

B. DHCP - automatic configuration.

C. PPPoE setup

D. PPTP setup

The individual manual steps to configure these connection alternatives with the Web Interface are discussed next.

A. Static - manual configuration

Manual configuration means that there will be a direct connection to the ISP and all the relevant IP addresses for the connecting interface are fixed values provided by the ISP which are entered into cOS Core manually.

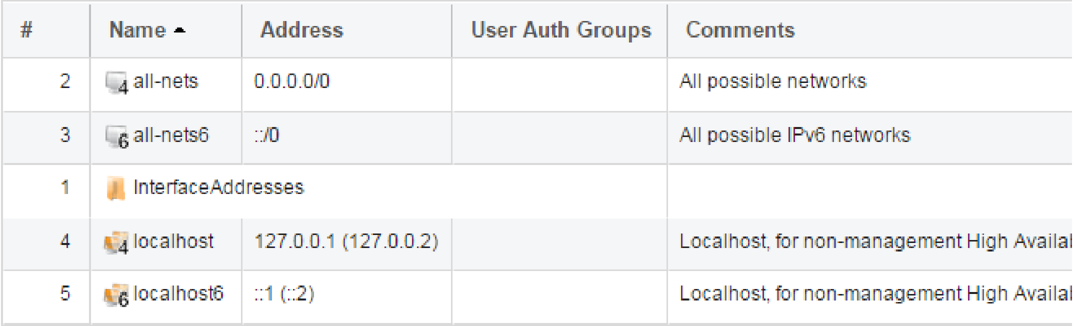
Note: The interface DHCP option should be disabled

For static configuration of the Internet connection, the DHCP option must be disabled in the properties of the interface that will connect to the ISP.

The initial step is to set up a number of IPv4 address objects in the cOS Core *Address Book*. Let us assume that the interface used for Internet connection is to be *If2* and that the static IPv4 address for this interface is to be *203.0.113.35*, the ISP's gateway IPv4 address is *203.0.113.1*, and the network to which they both belong is *203.0.113.0/24*.

Now, add the gateway *IP4 Address* object using the address book name *wan_gw* and assign it the IPv4 address *203.0.113.1*. The ISP's gateway is the first router hop towards the public Internet from the Clavister Next Generation Firewall. Go to **Objects > Address Book** in the Web Interface.

The current contents of the address book will be listed and will contain a number of predefined objects automatically created by cOS Core after it scans the interfaces for the first time. The screenshot below shows the initial address book for the KVM.

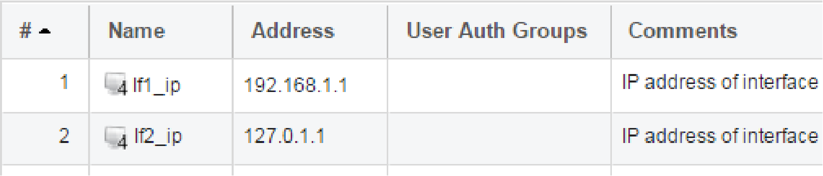


#	Name ▲	Address	User Auth Groups	Comments
2	all-nets	0.0.0.0/0		All possible networks
3	all-nets6	::/0		All possible IPv6 networks
1	InterfaceAddresses			
4	localhost	127.0.0.1 (127.0.0.2)		Localhost, for non-management High Availal
5	localhost6	::1 (::2)		Localhost, for non-management High Availal

Note: The all-nets address

The IPv4 address object ***all-nets*** *is a wildcard address that should never be changed and can be used in many types of cOS Core rules to refer to any IPv4 address or network range.*

For the KVM, all the Ethernet interface related address objects are gathered together in an *address book folder* called *InterfaceAddresses*. By clicking on this folder, it will be opened and the individual address objects it contains can be viewed. Predefined addresses in the folder are shown below.



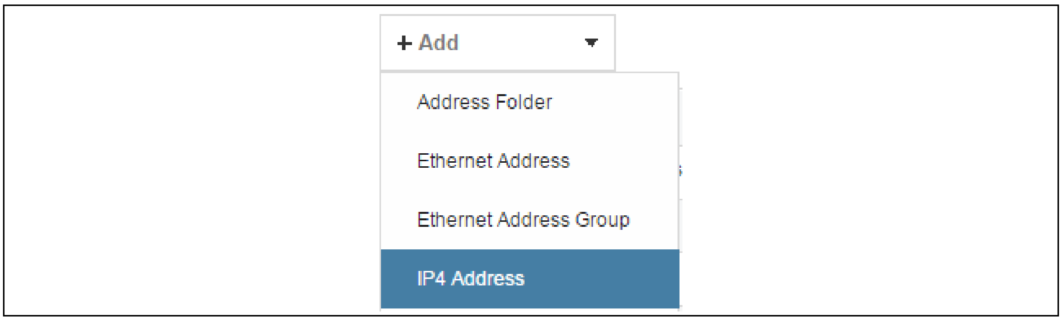
# ▲	Name	Address	User Auth Groups	Comments
1	If1_ip	192.168.1.1		IP address of interface
2	If2_ip	127.0.1.1		IP address of interface

On initial startup, two IPv4 address objects are created automatically for each interface detected by cOS Core. One IPv4 address object is named by combining the physical interface name with the suffix "*_ip*" and this is used for the IPv4 address assigned to that interface. The other address object is named by combining the interface name with the suffix "*_net*" and this is the network to which the interface belongs.

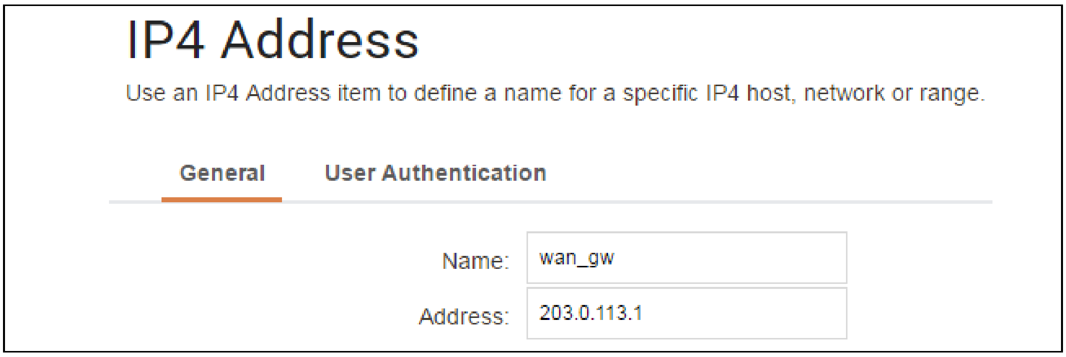
Tip: Creating address book folders

New folders can be created when needed and provide a convenient way to group together related IP address objects. The folder name can be chosen to indicate the folder's contents.

Now click the **Add** button at the top left of the list and choose the *IP4 Address* option to add a new address to the folder.



Enter the details of the object into the properties fields for the *IP4 Address* object. Below, the IPv4 address *203.0.113.1* has been entered for the address object called *wan_gw*. This is the IP of the ISP's router which acts as the gateway to the public Internet.



IP4 Address

Use an IP4 Address item to define a name for a specific IP4 host, network or range.

Name: wan_gw

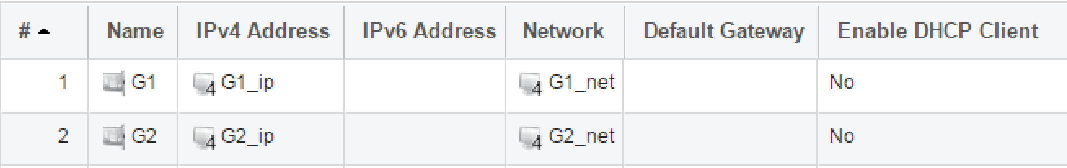
Address: 203.0.113.1

Click the **OK** button to save the values entered.

Then set up *If2_ip* to be *203.0.113.35*. This is the IPv4 address of the *If2* interface which will connect to the ISP's gateway.

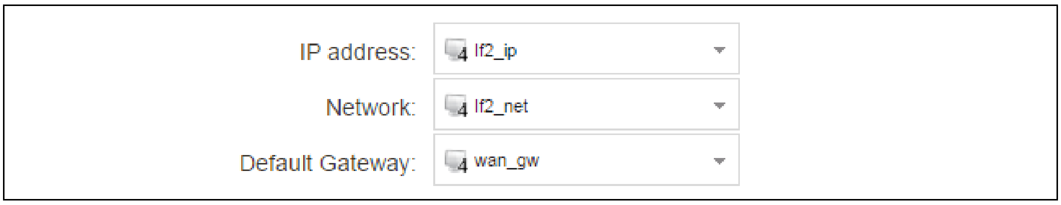
Lastly, set the *IP4 Address* object *If2_net* to be *203.0.113.0/24*. Both the address objects *If2_ip* and *wan_gw* must belong to the same network in order for the interface to communicate with the ISP.

Together, these three IPv4 address objects will be used to configure the interface connected to the Internet which, in this example, is *If2*. Select **Network > Interfaces and VPN > Ethernet** to display a list of the physical interfaces and address book objects assigned to them. The first lines of the default interface list for the KVM are shown below.



# ▲	Name	IPv4 Address	IPv6 Address	Network	Default Gateway	Enable DHCP Client
1	G1	G1_ip		G1_net		No
2	G2	G2_ip		G2_net		No

Click on the interface in the list which is to be connected to the Internet. The properties for this interface will now appear and the settings can be changed including the default gateway.



IP address: If2_ip

Network: If2_net

Default Gateway: wan_gw

Press **OK** to save the changes. Although changes are remembered by cOS Core, the changed configuration is not yet activated and won't be activated until cOS Core is told explicitly to use the changed configuration.

Remember that DHCP should **not** be enabled when using static IP addresses and also that the IP address of the *Default Gateway* (which is the ISP's router) **must** be specified. As explained in more detail later, specifying the *Default Gateway* also has the additional effect of automatically adding a route for the gateway in the cOS Core routing table.

At this point, the connection to the Internet is configured but no traffic can flow to or from the Internet since all traffic needs a minimum of the following two cOS Core configuration objects to exist before it can flow through the Clavister Next Generation Firewall:

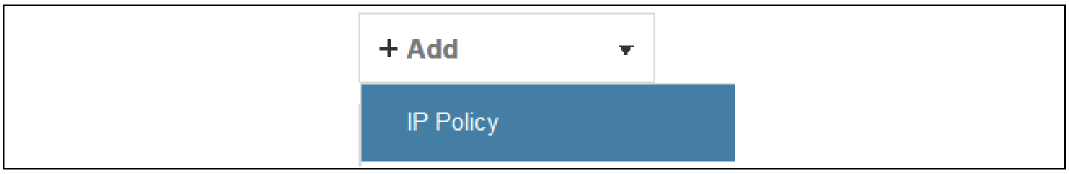
- An *IP Policy* object in the IP rule set that explicitly allows traffic to flow from a given source network and source interface to a given destination network and destination interface.
- A *route* defined in a cOS Core routing table which specifies on which interface cOS Core can find the traffic's destination IP address.

  If multiple matching routes are found, cOS Core uses the route that has the smallest (in other words, the narrowest) IP range.

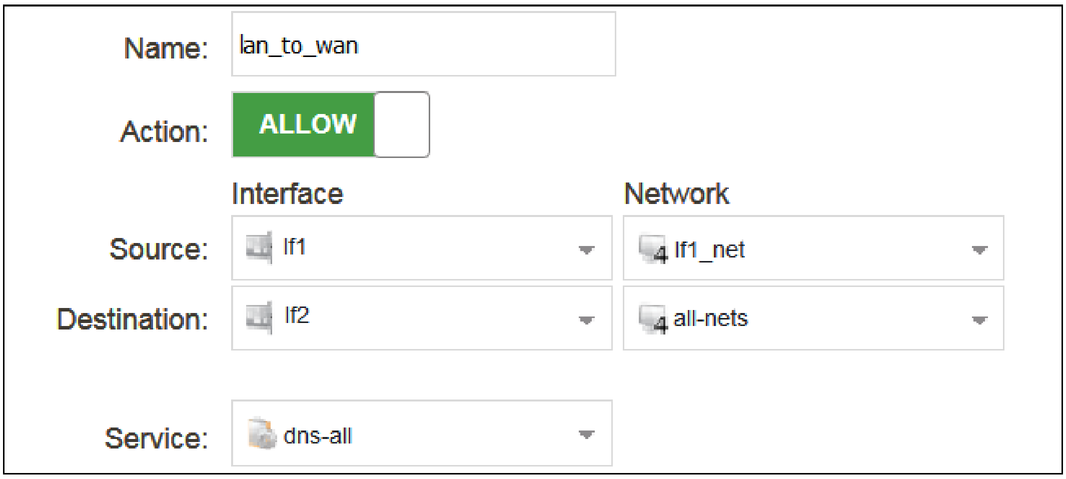
An IP policy therefore needs to be defined that will allow traffic from clients to the Internet. In this case, that web browsing is to be allowed from the protected private network *If1_net* connected to the interface *If1*.

To do this, first go to **Policies > Firewalling > Main IP Rules**. The *main* IP rule set will now be displayed.

To add a new IP policy, press the **Add** button and select **IP Policy** from the menu.



The properties for the new object will appear. In this example, the policy will be called *lan_to_wan*. The *Service* is set to *http-all* which is suitable for most web browsing (it allows both HTTP and HTTPS connections).



Name: lan_to_wan

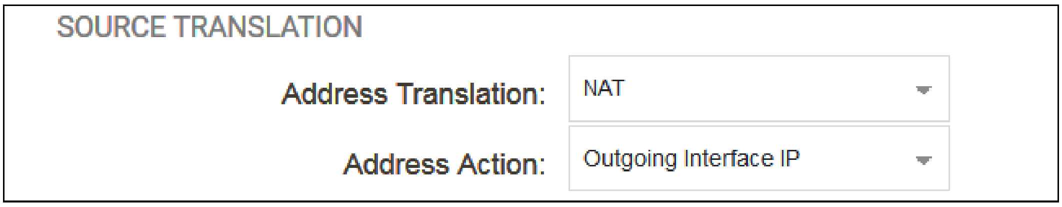
Action: ALLOW

	Interface	Network
Source:	If1	If1_net
Destination:	If2	all-nets

Service: dns-all

The destination network is specified as the predefined *IP4 Address* object *all-nets*. This is used since it cannot be known in advance to which IP address web browsing will be directed and *all-nets* allows browsing to any IP address. IP rule sets are processed in a top down fashion, with the search ending at first matching entry. An *all-nets* entry like this should be placed towards the end of the rule set since other rules with narrower destination addresses should trigger first.

In addition to entering the above for the policy, the *Source Translation* should be set to NAT and the *Address Action* left as *Outgoing Interface IP*. Note that the default source translation value for an IP policy is *Auto* and this would also provide NAT translation between a private and public IP address but NAT is specified explicitly in this section for clarity.



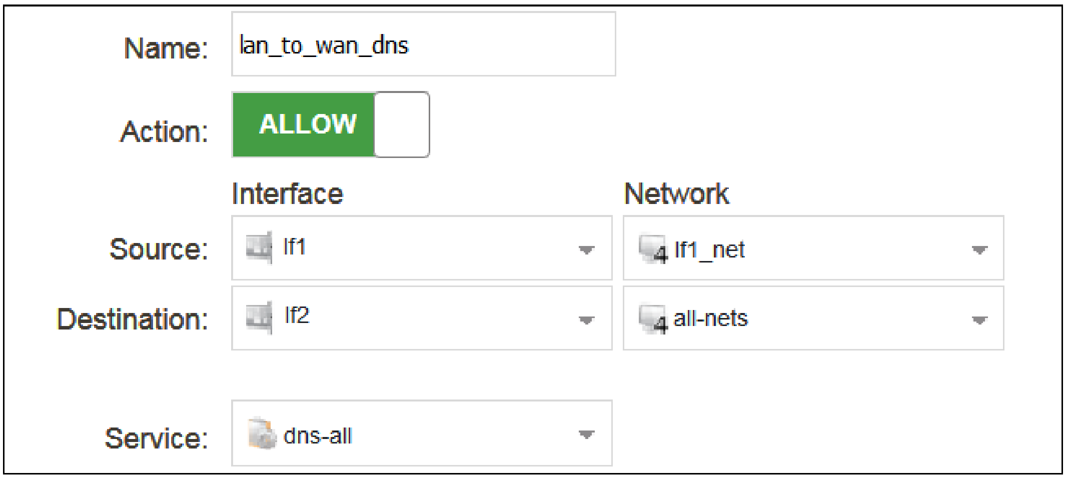
SOURCE TRANSLATION

Address Translation: NAT

Address Action: Outgoing Interface IP

By using *NAT*, cOS Core will use the destination interface's IP address as the source IP. This means that external hosts will send their responses back to the interface IP and cOS Core will automatically forward the traffic back to the originating local host. Only the outgoing interface therefore needs to have a public IPv4 address and the internal network topology is hidden.

For web browsing, public DNS lookup also needs to be allowed in order to resolve URLs into IP addresses. The service *http-all* does not include the *DNS* protocol so a similar IP rule set entry that allows this is needed. This could be done with a single IP policy that uses a custom service which combines the *HTTP* and *DNS* protocols but the recommended method is to create an entirely new IP set entry that specifies the service as *dns-all*. This method provides the most clarity when the configuration is examined for any problems. The screenshot below shows a new IP policy called *lan_to_wan_dns* being created to allow DNS.



Name: lan_to_wan_dns

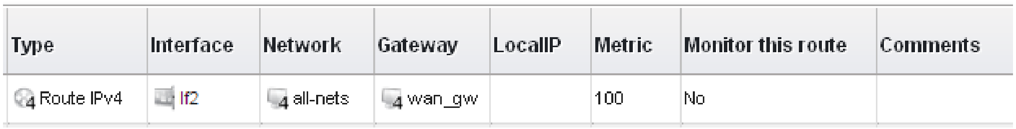
Action: ALLOW

	Interface	Network
Source:	If1	If1_net
Destination:	If2	all-nets

Service: dns-all

As was done for HTTP, NAT should also be enabled with this IP policy so all DNS queries are sent out by cOS Core with the outgoing interface's IP address as the source IP.

For the Internet connection to work, a *route* also needs to be defined so that cOS Core knows on which interface the web browsing traffic should leave the Clavister Next Generation Firewall. This route will define the interface where the network *all-nets* (in other words, any network) will be found. If the default *main* routing table is opened by going to **Network > Routing > Routing Tables > main**, the route needed should appear as shown below.



Type	Interface	Network	Gateway	LocalIP	Metric	Monitor this route	Comments
Route IPv4	If2	all-nets	wan_gw		100	No	

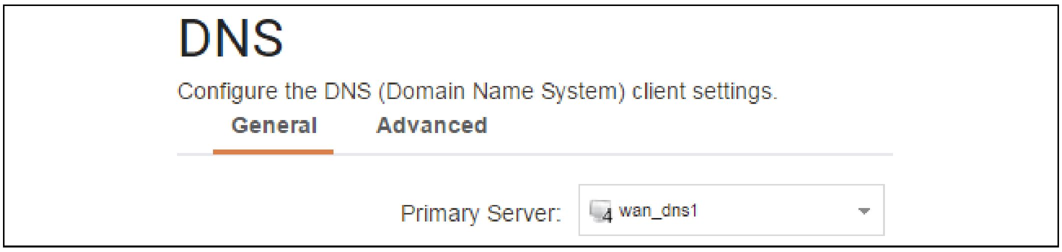
This required *all-nets* route is, in fact, added automatically after specifying the *Default Gateway* for a particular Ethernet interface and this was done earlier when setting up the required *IP4 Address* objects.

Note: Disabling automatic route generation

*Automatic route generation is enabled and disabled with the setting "**Automatically add a default route for this interface using the given default gateway**" which can be found in the properties of the interface.*

As part of the setup, it is also recommended that at least one DNS server is also defined in cOS Core. This DNS server or servers (a maximum of three can be configured) will be used when cOS Core itself needs to resolve URLs which is the case when a URL is specified in a configuration object instead of an IP address. It is also important for certificate handling

Assume an IPv4 address object called *wan_dns1* has already been defined in the address book and this is the address for the first DNS server. By choosing **System > Device > DNS**, the DNS server dialog will open and this object from the address book can be assigned as the first server.



DNS

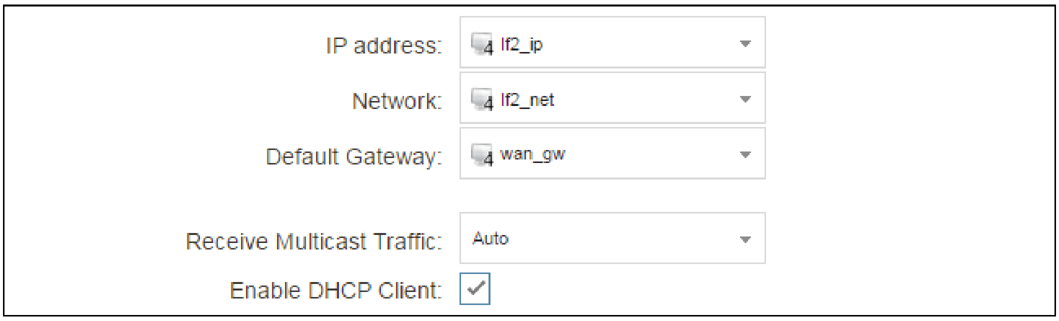
Configure the DNS (Domain Name System) client settings.

Primary Server: wan_dns1

B. DHCP - automatic configuration

All the required IP addresses for Internet connection can, alternatively, be automatically retrieved from an ISP's DHCP server by enabling the **DHCP Client** option for the interface connected to the ISP. This option is enabled by first selecting **Network > Interfaces and VPN > Ethernet** to display a list of all the interfaces.

Click the *If2* interface in the list to display its properties and select the option to enable the interface as a DHCP client.



IP address: If2_ip

Network: If2_net

Default Gateway: wan_gw

Receive Multicast Traffic: Auto

Enable DHCP Client: ☑

Usually, a DHCP *Host Name* does not need to be specified but can sometimes be used by an ISP to uniquely identify this Clavister Next Generation Firewall as a particular DHCP client to the ISP's DHCP server.

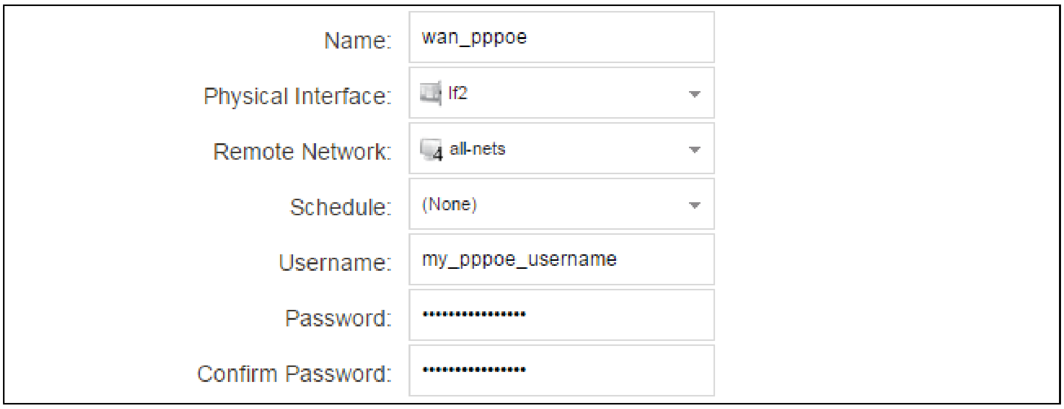
On connection to the ISP, all required IP addresses are retrieved automatically from the ISP via DHCP and cOS Core automatically sets the relevant address objects in the address book with this information.

For cOS Core to know on which interface to find the public Internet, a *route* has to be added to the *main* cOS Core routing table which specifies that the network *all-nets* can be found on the interface connected to the ISP and this route must also have the correct *Default Gateway* IP address specified. This *all-nets* route is added automatically by cOS Core during the DHCP address retrieval process.

After all IP addresses are set via DHCP and an *all-nets* route is added, the connection to the Internet is configured but no traffic can flow to or from the Internet since there is no IP rule set entry defined that allows it. As was done in the previous option (**A**) above, we must therefore define an IP policy that will allow traffic from the source network *If1_net* and source interface *If1* to flow to the destination network *all-nets* and the destination interface *If2*.

C. PPPoE setup

For PPPoE connection, we must create a PPPoE tunnel interface associated with the physical Ethernet interface. Assume that the physical interface is *If2* and the PPPoE tunnel object created is called *wan_pppoe*. Go to **Network > Interfaces and VPN > PPPoE** and select **Add > PPPoE Tunnel**. These values can now be entered into the PPPoE Tunnel properties dialog.

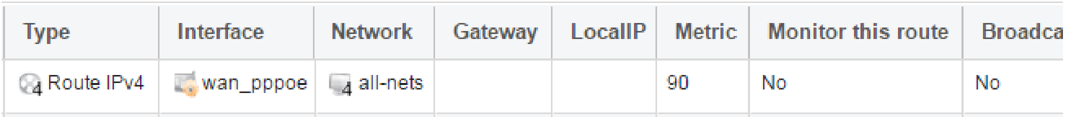


Name:	wan_pppoe
Physical Interface:	If2
Remote Network:	all-nets
Schedule:	(None)
Username:	my_pppoe_username
Password:	•••••••••••••••••
Confirm Password:	•••••••••••••••••

An ISP will supply the correct values for *pppoe_username* and *pppoe_password* in the dialog above.

The PPPoE tunnel interface can now be treated exactly like a physical interface by the policies defined in cOS Core rule sets.

There also has to be a route associated with the PPPoE tunnel to allow traffic to flow through it, and this is automatically created in the *main* routing table when the tunnel is defined. If we go to **Network > Routing > Routing Tables > main** we can see this route.



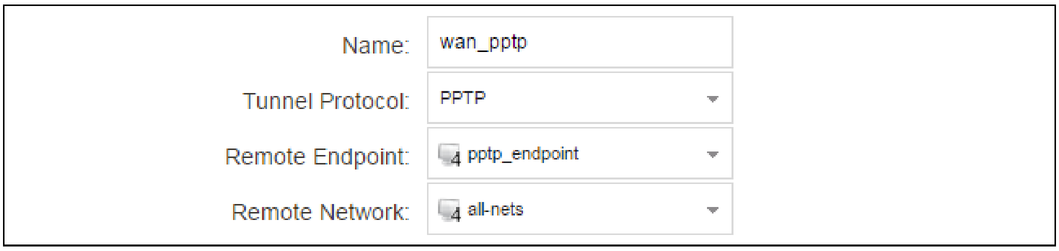
Type	Interface	Network	Gateway	LocalIP	Metric	Monitor this route	Broadca
Route IPv4	wan_pppoe	all-nets			90	No	No

If the PPPoE tunnel object is deleted, this route is also automatically deleted.

At this point, no traffic can flow through the tunnel since there is no IP rule set entry defined that allows it. As was done in option **A** above, we must define an IP policy that will allow traffic from the source network *If1_net* and source interface *If1* to flow to the destination network *all-nets* and the destination interface. Here, the destination interface is the PPPoE tunnel that has been defined.

D. PPTP setup

For PPTP connections, a PPTP client tunnel interface object needs to be created. Let us assume that the PPTP tunnel will be called *wan_pptp* with a remote endpoint *203.0.113.1* which has been defined as the *IP4 Address* object *pptp_endpoint*. Go to **Network** > **Interfaces and VPN** > **PPTP/L2TP Clients** and select **Add** > **PPTP/L2TP Client**. The values can now be entered into the properties dialog and the *PPTP* option should be selected.



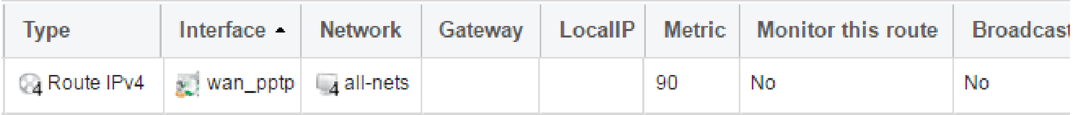
Name:	wan_pptp
Tunnel Protocol:	PPTP
Remote Endpoint:	4 pptp_endpoint
Remote Network:	4 all-nets

An ISP will supply the correct values for *pptp_username*, *pptp_password* and the remote endpoint. An interface is not specified when defining the tunnel because this is determined by cOS Core looking up the *Remote Endpoint* IP address in its routing tables.

The PPTP client tunnel interface can now be treated exactly like a physical interface by the policies defined in cOS Core rule sets.

There also has to be an associated route with the PPTP tunnel to allow traffic to flow through it, and this is automatically created in the *main* routing table when the tunnel is defined. The destination network for this route is the *Remote Network* specified for the tunnel and for the public Internet this should be *all-nets*.

If we go to **Network** > **Routing** > **Routing Tables** > **main** we can see this route.



Type	Interface ▲	Network	Gateway	LocalIP	Metric	Monitor this route	Broadcast
4 Route IPv4	wan_pptp	4 all-nets			90	No	No

If the PPTP tunnel object is deleted, this route is also automatically deleted.

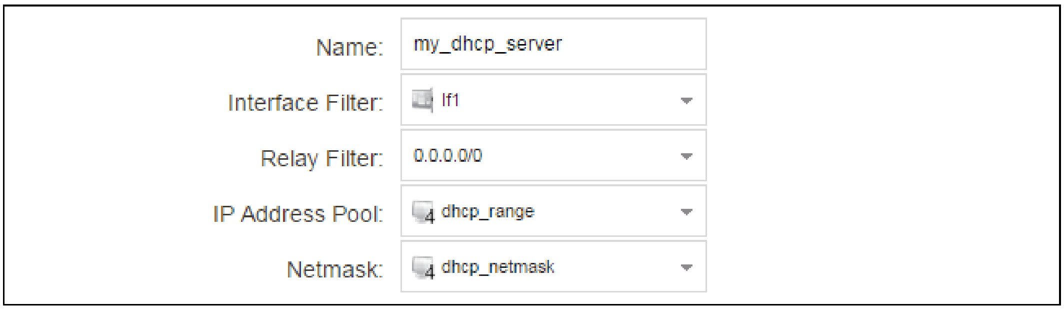
At this point, no traffic can flow through the tunnel since there is no IP rule entry defined that allows it. As was done in option **A** above, we must define an IP policy that will allow traffic from a designated source network and source interface (in this example, the network *If1_net* and interface *If1*) to flow to the destination network *all-nets* and the destination interface which is the PPTP tunnel that has been defined.

DHCP Server Setup

If the Clavister Next Generation Firewall is to act as a DHCP server then this can be set up in the following way:

First, create an *IP4 Address* object which defines the address range to be handed out. Here, it is assumed that this has the name *dhcp_range*. It is also assumed that another *IP4 Address* object *dhcp_netmask* has been created which specifies the netmask.

We now create a DHCP server object called *my_dhcp_server* which will only be available on the *If1* interface. To do this, go to **Network** > **Network Services** > **DHCP Servers** and select **Add** > **DHCP Server**. The server properties can now be specified.



Name: my_dhcp_server

Interface Filter: If1

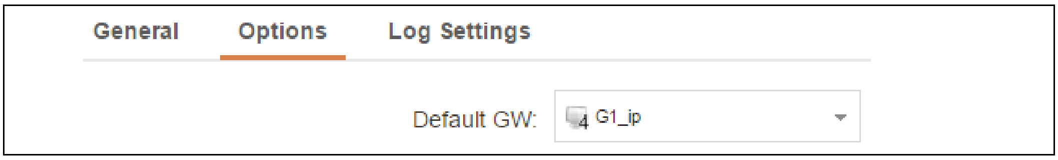
Relay Filter: 0.0.0.0/0

IP Address Pool: dhcp_range

Netmask: dhcp_netmask

An example IP pool range might be *196.168.1.10 - 192.168.1.20* with a netmask of *255.255.0.0*.

In addition, it is important to specify the *Default gateway* for the server. This will be handed out to DHCP clients on the internal networks so that they know where to find the public Internet. The default gateway is always the IPv4 address of the interface on which the DHCP server is configured, in this case, *If1_ip*. To set the default gateway, select the **Options** tab.



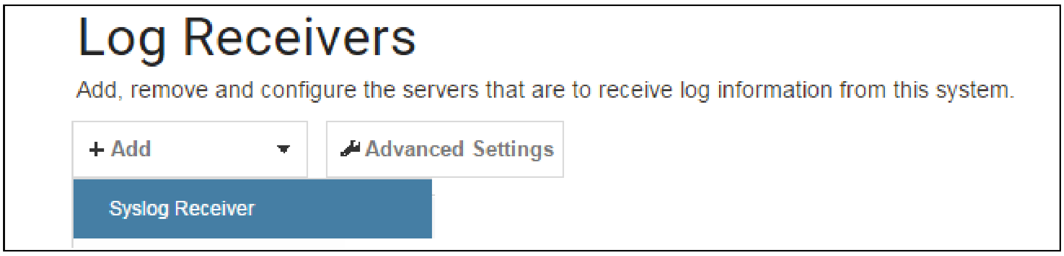
Default GW: G1_ip

Also in the **Options** tab, we should specify the DNS address which is handed out with DHCP leases. This could be set, for example, to be the IPv4 address object *dns1_address*.

Syslog Server Setup

Although logging may be enabled, no log messages are captured unless at least one log server is set up to receive them and this is configured in cOS Core. *Syslog* is one of the most common server types.

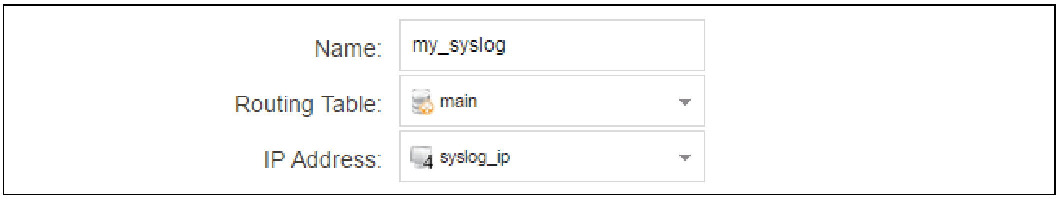
First we create an *IP4 Address* object called, for example, *syslog_ip* which is set to the IPv4 address of the server. We then configure the sending of log messages to a Syslog server from cOS Core by selecting **System > Device > Log and Event Receivers** and then choosing **Add > Syslog Receiver**.



Log Receivers

Add, remove and configure the servers that are to receive log information from this system.

The Syslog server properties dialog will now appear. We give the server a name, for example *my_syslog*, and specify its IPv4 address as the *syslog_ip* object.



Name: my_syslog

Routing Table: main

IP Address: syslog_ip

Tip: Address book object naming

*The cOS Core address book is organized alphabetically so when choosing names for IP address objects it is best to have the descriptive part of the name first. In this case, use **syslog_ip** as the name and not **ip_syslog**.*

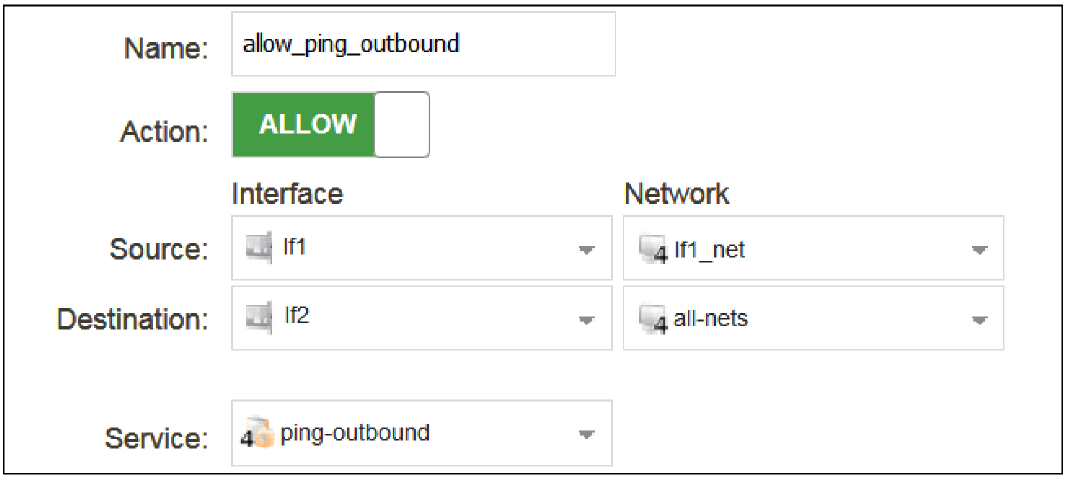
Allowing ICMP *Ping* Requests

As a further example of setting up IP rule set entries, it can be very useful to allow ICMP *Ping* requests to flow through the Clavister Next Generation Firewall. As discussed earlier, the cOS Core will drop any traffic unless a rule set entry explicitly allows it. Let us suppose that we wish to allow the pinging of external hosts with the ICMP protocol by computers on the internal network *G1_net*.

There can be several IP rule sets defined in cOS Core but there is only one rule set defined by default and this is called *main*. To add an entry to it, first select **Policies > Firewalling > Main IP Rules**.

The *main* rule set list contents are now displayed. Press the **Add** button and select **IP Policy**.

The properties for a new IP policy will appear and we can add the entry, in this case called *allow_ping_outbound*.



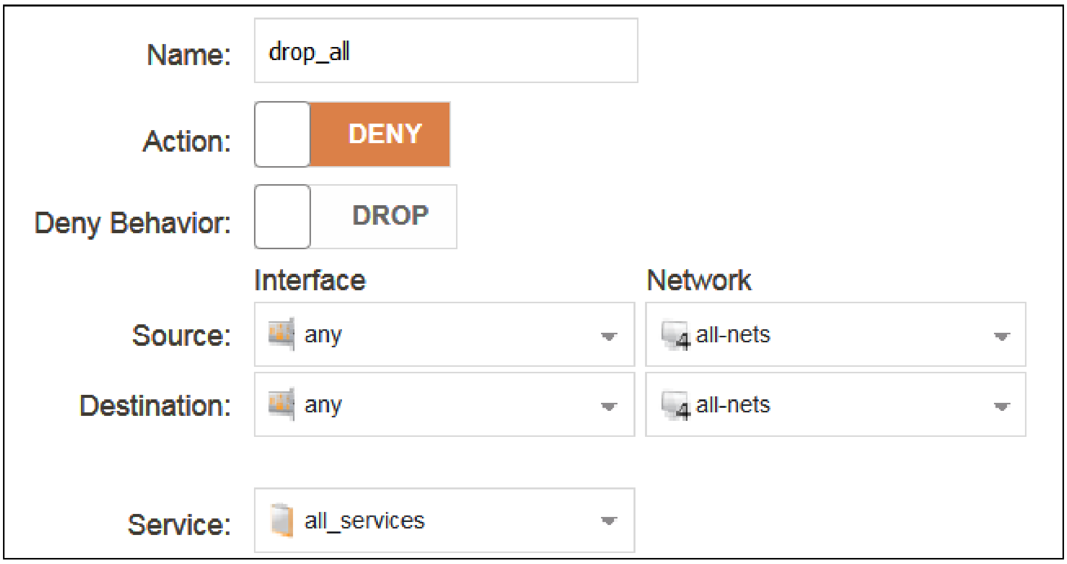
Name:	allow_ping_outbound	
Action:	ALLOW	
	Interface	Network
Source:	If1	If1_net
Destination:	If2	all-nets
Service:	ping-outbound	

As with previous policy definitions, NAT should also be enabled if the protected local hosts have private IPv4 addresses. The ICMP requests will then be sent out from the Clavister Next Generation Firewall with the IP address of the interface connected to the ISP as the source interface. Responding hosts will send back ICMP responses to this single IP and cOS Core will then forward the response to the correct private IPv4 address.

Adding a Drop All Policy

The top-down nature of the IP rule set scanning has already been discussed earlier. If **no** matching entry is found for a new connection then the *default rule* is triggered. This entry is hidden and cannot be changed. Its action is to drop all such traffic as well as generate a log message for the drop.

In order to gain control over the logging of dropped traffic, it is recommended to create a drop all policy as the last entry in the *main* IP rule set. This policy will have the source and destination network set to *all-nets* and the source and destination interface set to *any*. The service should be set to *all_services* in order to capture all types of traffic.



Name: drop_all

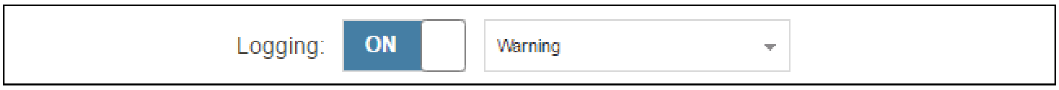
Action: DENY

Deny Behavior: DROP

	Interface	Network
Source:	any	all-nets
Destination:	any	all-nets

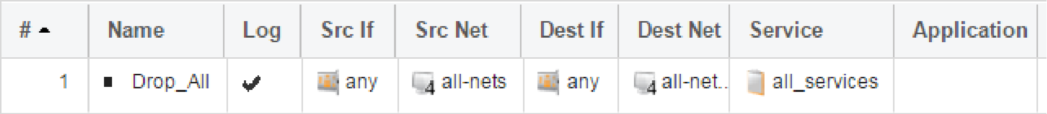
Service: all_services

Logging is enabled by default for an IP rule set entry which means that a log message will be sent to all configured log servers whenever the entry triggers. Only log events that have a specified severity or above will be sent. The administrator can choose the minimum severity for log messages in each IP rule set entry, as shown below.



Logging: ON Warning

If this IP policy were the only one defined, the *main* IP rule set listing would be as shown below.



# ▲	Name	Log	Src If	Src Net	Dest If	Dest Net	Service	Application
1	■ Drop_All	✔	any	all-nets	any	all-net..	all_services	

A Valid License Must Be Installed

Lastly, a valid license should be installed to remove the cOS Core 2 hour demo mode limitation. Without a license installed, cOS Core will have full functionality during the 2 hour period following startup, but after that, only management access will be possible. Installing a license is described in *Section 6.5, "Installing a License"*.

6.4. Manual CLI Setup

This chapter describes the setup steps using CLI commands instead of the setup wizard.

The CLI is accessible using either one of two methods:

- Using an SSH (Secure Shell) client, across a network connection to the IPv4 address *192.168.1.1* on the default management Ethernet interface. The physical network connection setup to the computer running the client is described in *Section 6.1, "Management Workstation Connection"* and is the same as that used in *Section 6.2, "Web Interface and Wizard Setup"*.

  If there is a problem with the workstation connection, a help checklist can be found in *Section 6.6, "Setup Troubleshooting "*.

- Using a terminal or computer running a console emulator connected directly to the local console port on the KVM.

The CLI commands listed below are grouped so that they mirror the options available in the setup wizard.

Confirming the Connection

Once connection is made to the CLI, pressing the **Enter** key will cause cOS Core to respond. The response will be a normal CLI prompt if connecting directly through the local console port and a username/password combination will not be required (a password for this console can be set later).

```
Device:/>
```

If connecting remotely through an SSH (Secure Shell) client, an administration username/password must first be entered and the initial default values for these are username *admin* and password *admin*. When these are accepted by cOS Core, a normal CLI prompt will appear and CLI commands can be entered.

Changing the Password

To change the administration username or password, use the *set* command to change the current CLI object category (also referred to as the *context*) to be the *LocalUserDatabase* called *AdminUsers*.

```
Device:/> cc LocalUserDatabase AdminUsers
Device:/AdminUsers>
```

Tip: Using tab completion with the CLI

The tab key can be pressed at any time so that cOS Core gives a list of possible options in a command.

Now set the username and password for the administrator. Both are case sensitive. In the example below, the username is set to the value *new_name* and the password is set to the value *new_pass*.

```
Device:/AdminUsers> set User Admin Name=new_name Password=new_pass
```

The new username/password combination should be remembered and the password should be composed in a way which makes it difficult to guess. The next step is to return the CLI to the default context which is the top level of object categories.

```
Device:/AdminUsers> cc
Device:/>
```

Setting the Date and Time

Many cOS Core functions, such as event logging and certificate handling, rely on an accurate date and time. It is therefore important that this is set correctly using the *time* command. A typical usage of this command might be:

```
Device:/> time -set 2017-06-24 14:43:00
```

Notice that the date is entered in *yyyy-mm-dd* format and the time is stated in 24 hour *hh:mm:ss* format.

Ethernet Interfaces

The connection of external networks to the Clavister Next Generation Firewall is via the various *Ethernet interfaces* which are provided by the hardware platform. On first-time startup, cOS Core determines which interfaces are available and allocates their names. One interface is chosen as the initial default management interface and this can only be changed after initial startup.

All cOS Core interfaces are logically equal for cOS Core and although their physical capabilities may be different, any interface can perform any logical function. With the KVM, the **If1** interface is the default management interface. The other interfaces can be used as required. For this section, it is assumed that the **If2** interface will be used for connection to the public Internet and the **If1** interface will also be used for connection to a protected, local client network.

Setting Up Internet Access

Next, we shall look at how to set up public Internet access with the CLI. There are four options for setting up access which are listed below and then described in detail.

A. Static - manual configuration.

B. DHCP - automatic configuration.

C. PPPoE setup.

D. PPTP setup.

The individual manual steps to configure these connection alternatives with the CLI are discussed next.

A. Static - manual configuration

We first must set or create a number of IPv4 address objects. It is assumed here that the interface used for Internet connection is *If2*, the ISP gateway IPv4 address is *203.0.113.1*, the IPv4 address for the connecting interface will be *203.0.113.35* and the network to which they both belong is *203.0.113.0/24*.

First, add the gateway IPv4 address object if it does not already exist:

```
Device:/> add Address IP4Address wan_gw Address=203.0.113.1
```

This is the address of the ISP's gateway which is the first router hop towards the public Internet. If this IP object already exists, it can be given the IP address with the command:

```
Device:/> set Address IP4Address wan_gw Address=203.0.113.1
```

Now, set the gateway on the *If2*. interface which is connected to the ISP:

```
Device:/> set Interface Ethernet If2 DefaultGateway=wan_gw
```

Next, set the IP address of the *If2_ip* address object which is the IP assigned to the interface:

```
Device:/> set IP4Address InterfaceAddresses/If2_ip Address=203.0.113.35
```

Note: Qualifying the names of IP objects in folders

*On initial startup of the KVM, cOS Core automatically creates and fills the **InterfaceAddresses** folder in the cOS Core address book with Ethernet interface related IPv4 address objects.*

*Note that when an IP address object which is located in a folder is specified in the CLI, the object name must be qualified with the name of its parent folder. For example, to reference the address **If2_ip**, it must be qualified with the folder name **InterfaceAddresses** so it becomes **InterfaceAddresses/If2_ip**.*

If an object is not contained in a folder and is at the top level of the address book then no qualifying parent folder name is needed.

Now, set the IP object *If2_net*. which will be the IP network of the connecting interface:

```
Device:/> set IP4Address InterfaceAddresses/If2_net Address=203.0.113.0/24
```

It is recommended to verify the properties of the *If2*. interface using the following command:

```
Device:/> show Interface Ethernet If2
```

The typical output from this will be similar to the following:

```
                  Property  Value
--------------------------  --------------------------
                     Name:  If2
                       IP:  InterfaceAddresses/If2_ip
                  Network:  InterfaceAddresses/If2_net
           DefaultGateway:  wan_gw
                Broadcast:  203.0.113.255
                PrivateIP:  <empty>
                    NOCHB:  <empty>
                      MTU:  1500
                   Metric:  100
              DHCPEnabled:  No
           EthernetDevice:  0:If2  1:<empty>
          AutoSwitchRoute:  No
AutoInterfaceNetworkRoute:  Yes
  AutoDefaultGatewayRoute:  Yes
  ReceiveMulticastTraffic:  Auto
     MemberOfRoutingTable:  All
                 Comments:  <empty>
```

Setting the default gateway on the interface has the additional effect that cOS Core automatically creates a route in the default *main* routing table that has the network *all-nets* routed on the interface. This means that we do not need to explicitly create this route.

Even though an *all-nets* route is automatically added, no traffic can flow without the addition of an IP rule set entry which explicitly allows traffic to flow. Let us assume we want to allow web browsing from the protected network *If1_net* which is connected to the interface *If1*.

The following command will add an IP policy called *lan_to_wan* to allow traffic from *If1_net* through to the public Internet:

```
Device:/> add IPPolicy Name=lan_to_wan
                        SourceInterface=If1
                        SourceNetwork=InterfaceAddresses/If1_net
                        DestinationInterface=If2
                        DestinationNetwork=all-nets
                        Service=http-all
                        Action=Allow
```

IP policies have a default value of *Auto* for the type of source translation. This means that if the source is a private IPv4 address and the destination is a public address, NAT will be performed automatically using the IP address of the outgoing interface as the new source address. Therefore the above IP policy will work both for connection to another private IP address or to public addresses on the Internet.

Instead of relying on the *Auto* option, this section will specify NAT translation explicitly for clarity. The above IP policy with explicit NAT translation becomes the following:

```
Device:/main> add IPPolicy Name=lan_to_wan
                        SourceInterface=If1
                        SourceNetwork=InterfaceAddresses/If1_net
                        DestinationInterface=If2
                        DestinationNetwork=all-nets
                        Service=http-all
                        Action=Allow
                        SourceAddressTranslation=NAT
                        NATSourceAddressAction=OutgoingInterfaceIP
```

Specifying *NATSourceAddressAction=OutgoingInterfaceIP* is not necessary as this is the default value but it is included here for clarity.

The service used is *http-all* which will allow web browsing but does not include the DNS protocol to resolve URLs into IP addresses. To solve this problem, a custom service could be used in the above which combines *http-all* with the *dns-all* service. However, the recommended method, which provides the most clarity to a configuration, is to create a separate IP policy for DNS:

```
Device:/main> add IPPolicy Name=lan_to_wan_dns
                        SourceInterface=If1
                        SourceNetwork=InterfaceAddresses/If1_net
                        DestinationInterface=If2
                        DestinationNetwork=all-nets
                        Service=dns-all
                        Action=Allow
                        SourceAddressTranslation=NAT
                        NATSourceAddressAction=OutgoingInterfaceIP
```

It is recommended that at least one DNS server is also defined in cOS Core. This DNS server or servers (a maximum of three can be configured) will be used when cOS Core itself needs to resolve URLs which will be the case when a URL is specified in a configuration instead of an IP address. If we assume an IP address object called *dns1_address* has already been defined for the first DNS server, the command to specify the first DNS server is:

```
Device:/> set DNS DNSServer1=dns1_address
```

Assuming a second IP object called *dns2_address* has been defined, the second DNS server is specified with:

```
Device:/> set DNS DNSServer2=dns2_address
```

B. DHCP - automatic configuration

Alternatively, all required IP addresses can be automatically retrieved from the ISP's DHCP server by enabling DHCP on the interface connected to the ISP.

If the interface on which DHCP is to be enabled is *If2*, then the command is:

```
Device:/> set Interface Ethernet If2 DHCPEnabled=Yes
```

Once the required IP addresses are retrieved with DHCP, cOS Core automatically sets the relevant address objects in the address book with this information.

For cOS Core to know on which interface to find the public Internet, a *route* has to be added to the *main* cOS Core routing table which specifies that the network *all-nets* can be found on the interface connected to the ISP and this route must also have the correct *Default Gateway* IP address specified. This *all-nets* route is added automatically by cOS Core during the DHCP address retrieval process. Automatic route generation is a setting for each interface that can be manually enabled and disabled.

After all IP addresses are set via DHCP and an *all-nets* route is added, the connection to the Internet is configured but no traffic can flow to or from the Internet since there is no IP rule set entry defined that allows it. As was done in the previous option (**A**) above, we must therefore manually define an IP policy that will allow traffic from a designated source network and source interface (in this example, the network *If1_net* and interface *If1*) to flow to the destination network *all-nets* and the destination interface *If2*.

C. PPPoE setup

For PPPoE connection, create the PPPoE tunnel interface on the interface connected to the ISP. The interface *If2*. is assumed to be connected to the ISP in the command shown below which creates a PPPoE tunnel object called *wan_ppoe*:

```
Device:/> add Interface PPPoETunnel wan_ppoe
                        EthernetInterface=If2
                        username=pppoe_username
                        Password=pppoe_password
                        Network=all-nets
```

The ISP will supply the correct values for *pppoe_username* and *pppoe_password* in the dialog above.

The PPPoE tunnel interface can now be treated exactly like a physical interface by the policies defined in cOS Core rule sets.

There also has to be a route associated with the PPPoE tunnel to allow traffic to flow through it and this is automatically created in the *main* routing table when the tunnel is defined. If the PPPoE tunnel object is deleted, this route is also automatically deleted.

At this point, no traffic can flow through the tunnel since there is no IP rule set entry defined that allows it. As was done in option **A** above, we must define an IP policy that will allow traffic from a designated source network and source interface (in this example, the network *If1_net* and interface *If1*) to flow to the destination network *all-nets* and the destination interface which is the PPPoE tunnel that has been defined.

D. PPTP setup

For PPTP connection, first create the PPTP tunnel interface. It is assumed below that we will create a PPTP tunnel object called *wan_pptp* with the remote endpoint *203.0.113.1*:

```
Device:/> add Interface L2TPClient wan_pptp
                        Network=all-nets
                        Username=pptp_username
                        Password=pptp_password
                        RemoteEndpoint=203.0.113.1
                        TunnelProtocol=PPTP
```

Your ISP will supply the correct values for *pptp_username*, *pptp_password* and the remote endpoint.

Your ISP will supply the correct values for *pptp_username*, *pptp_password* and the remote endpoint. An interface is not specified when defining the tunnel because this is determined by cOS Core looking up the *Remote Endpoint* IP address in its routing tables.

The PPTP client tunnel interface can now be treated exactly like a physical interface by the policies defined in cOS Core rule sets.

There also has to be an associated route with the PPTP tunnel to allow traffic to flow through it, and this is automatically created in the *main* routing table when the tunnel is defined. The destination network for this route is the *Remote Network* specified for the tunnel and for the public Internet this should be *all-nets*.

As with all automatically added routes, if the PPTP tunnel object is deleted then this route is also automatically deleted.

At this point, no traffic can flow through the tunnel since there is no IP rule set entry defined that allows it. As was done in option **A** above, we must define an IP policy that will allow traffic from a designated source network and source interface (in this example, the network *If1_net* and interface *If1*) to flow to the destination network *all-nets* and the destination interface which is the PPTP tunnel that has been defined.

Activating and Committing Changes

After any changes are made to a cOS Core configuration, they will be saved as a new configuration but will not yet be activated. To activate all the configuration changes made since the last activation of a new configuration, the following command must be issued:

```
Device:/> activate
```

Although the new configuration is now activated, it does not become permanently activated until the following command is issued within 30 seconds following the *activate*:

```
Device:/> commit
```

The reason for two commands is to prevent a configuration accidentally locking out the administrator. If a lock-out occurs then the second command will not be received and cOS Core will revert back to the original configuration after the 30 second time period (this time period is a setting that can be changed).

DHCP Server Setup

If the Clavister Next Generation Firewall is to act as a DHCP server then this can be set up in the following way:

First define an IPv4 address object which has the address range that can be handed out. Here, we will use the IPv4 range *192.168.1.10 - 192.168.1.20* as an example and this will be made available on the *If1* interface which is connected to the protected internal network *If1_net*.

```
Device:/> add Address IP4Address dhcp_range
                        Address=192.168.1.10-192.168.1.20
```

The DHCP server is then configured with this IP address object on the appropriate interface. In this case we will call the created DHCP server object *my_dhcp_server*.

```
Device:/> add DHCPServer my_dhcp_server
                        IPAddressPool=dhcp_range
                        Interface=If1
                        Netmask=255.255.255.0
                        DefaultGateway=InterfaceAddresses/If1_ip
                        DNS1=dns1_address
```

It is important to specify the default gateway for the DHCP server since this will be handed out to DHCP clients on the internal network so that they know where to find the public Internet. The default gateway is always the IP address of the interface on which the DHCP server is configured. In this case, *If1_ip*.

NTP Server Setup

Network Time Protocol (NTP) servers can optionally be configured to maintain the accuracy of the system date and time. Suppose that synchronization is to be setup with the two NTP servers at hostname *pool.ntp.org* and IPv4 address *203.0.113.5*.

First, an *FQDNAddress* object needs to set up for the hostname:

```
Device:/> add Address FQDNAddress ts1_fqdn Address=pool.ntp.org
```

Next, set the servers to use for date and time synchronization:

```
Device:/> set DateTime TimeSyncEnable=Yes
                TimeSyncServer1=ts1_fqdn
                TimeSyncServer2=203.0.113.5
```

Syslog Server Setup

Although logging may be enabled, no log messages are captured unless a server is set up to receive them and *Syslog* is the most common server type. If the Syslog server's address is *192.0.2.10* then the command to create a log receiver object called *my_syslog* which enables logging is:

```
Device:/> add LogReceiverSyslog my_syslog IPAddress=192.0.2.10
```

Allowing ICMP *Ping* Requests

As a further example of setting up IP rule set entries, it can be useful to allow ICMP *Ping* requests to flow through the Clavister Next Generation Firewall. As discussed earlier, cOS Core will drop any traffic unless an rule set entry explicitly allows it. Let us suppose that we wish to allow the pinging of external hosts with the ICMP protocol by computers on the internal *If1_net* network. The commands to allow this are as follows.

Add an IP policy called *allow_ping_outbound* to allow ICMP pings to pass:

```
Device:/> add IPPolicy Name=allow_ping_outbound
                        SourceInterface=If1
                        SourceNetwork=InterfaceAddresses/If1_net
                        DestinationInterface=If2
                        DestinationNetwork=all-nets
                        Service=ping-outbound
                        Action=Allow
                        SourceAddressTranslation=NAT
                        NATSourceAddressAction=OutgoingInterfaceIP
```

This IP policy uses NAT and this is necessary if the protected local hosts have private IPv4 addresses. The ICMP requests will be sent out from the firewall with the IP address of the interface connected to the ISP as the source interface. Responding hosts will send back ICMP responses to this single IP and cOS Core will then forward the response to the correct private IP address.

Adding a Drop All Policy

Scanning of IP rule sets is done in a top-down fashion. If **no** matching rule set entry is found for a new connection then the *default rule* is triggered. This rule is hidden and cannot be changed and its action is to drop all such traffic as well as generate a log message for the drop.

In order to gain control over the logging of dropped traffic, it is recommended to create a drop all policy as the last entry in the *main* IP rule set. This policy will have the source and destination network set to *all-nets* and the source and destination interface set to *any*. The service should be set to *all_services* in order to capture all types of traffic.

The following IP policy will drop all remaining traffic as well as turning off logging for that traffic:

```
Device:/main> add IPPolicy Name=drop_all
                        SourceInterface=any
                        SourceNetwork=any
                        DestinationInterface=any
                        DestinationNetwork=all-nets
                        Service=all_services
                        Action=Deny
                        LogEnabled=No
```

A Valid License Must Be Installed

Lastly, a valid license should be installed to remove the cOS Core 2 hour demo mode limitation. Without a license installed, cOS Core will have full functionality during the 2 hour period following startup, but after that, only management access will be possible. Installing a license is described in *Section 6.5, "Installing a License"*.

6.5. Installing a License

Each virtual copy of cOS Core running under KVM requires a unique license file to be installed. Without a license, cOS Core will function for only 2 hours from startup in *demo mode* (demonstration mode). To end demo mode, a valid license file must be installed.

The following should be noted for license installation with cOS Core running in a virtual environment:

- The first time a license is installed, it must be done manually. This means that the administrator must log in to the Clavister website, download a license file to the management computer and then upload it to the firewall using either the Web Interface or SCP. This section describes the steps for manual installation in detail.
- For cOS Core version 12.00.09 and later, any subsequent updates to the original license can either by installed manually or the *automatic update* feature can be enabled which means that cOS Core detects when a license update is available and gives the administrator the option for cOS Core to automatically download and install the new license. Internet access is required for this feature.

  The automatic update feature is enabled in the Web Interface by going to **Status > Maintenance > My Clavister** and entering the relevant login credentials for the Clavister website. This creates a link between cOS Core and the Clavister license server. This can also be done in the CLI with the following command:

  ```
  Device:/> license -myclavister -username=myuser -password=mypass
  ```

  The automatic update feature is discussed further in the separate *cOS Core Administration Guide*.

Installing a cOS Core License Manually

To install a license manually, perform the following steps:

1. Obtain a *license code* from a cOS Core reseller for a virtual environment license.
2. Create a cOS Core virtual machine in the virtual environment.
3. Make a note of the MAC address of one of the cOS Core virtual Ethernet interfaces. A MAC address can be found through the Web Interface or with the following CLI command:

   ```
   Device:/> ifstat If1
   ```

   Note that this MAC address will be associated with the license and cannot be changed later.
4. If not already done, register as a user and enter your organization details, by choosing the *Login* link at *https://www.clavister.com*.
5. Log in to the Clavister website and go to **Licenses > Register License** then select the registration option **License Number and MAC Address**.
6. Enter the license code and MAC address. This will cause a new license to be generated and stored on the website. This license will appear in the user's license list on the site.
7. Download the created license from the website to the local disk of the management computer by clicking on the entry in the license list.

8. Upload the license from the management computer to cOS Core using the Web Interface or SCP. In virtual environments, cOS Core cannot automatically fetch the license. In the Web Interface, go to **Status > Maintenance > License** and press the **Upload** button to select the license file on disk.

9. After upload, perform a restart or reconfigure on cOS Core to complete installation of the license. A restart is recommended in case memory requirements have changed.

10. Following a restart, the 2 hour demo mode will end and the cOS Core capabilities will only be restricted by the installed license.

Examining the License Contents

The contents of a Clavister license (*.lic*) file can be examined by opening it in a standard text editor. cOS Core licenses for virtual environments contain the line:

```
Virtual Hardware: Yes
```

The license contents also specifies how many virtual interfaces are available on one virtual machine. The default value can be upgraded by purchasing the appropriate license.

KVM and cOS Core Lockdown Mode

When cOS Core is run in demo mode (that is to say, without a valid license), it will operate for two hours before it enters *lockdown mode*.

When cOS Core enters *lockdown mode* under KVM, it will consume all KVM resources. When this happens it is necessary to shut down the cOS Core virtual machine instance since nothing further can be done with cOS Core itself until it is restarted. In other words, restarting cOS Core should **only** be done via the KVM management interface once *lockdown mode* is entered.

General information about cOS Core licensing can be found in the *cOS Core Administration Guide*.

6.6. Setup Troubleshooting

This appendix deals with connection problems that might occur when connecting a management workstation to a Clavister Next Generation Firewall.

If the management interface does not respond after the Clavister Next Generation Firewall has powered up and cOS Core has started, there are a number of simple steps to troubleshoot basic connection problems:

1. Check that the correct interface is being used.

The most obvious problem is that the wrong interface has been used for the initial connection to the management workstation. Only the first interface found by cOS Core is activated for the initial connection from a browser after cOS Core starts for the first time.

2. Check that the workstation IP is configured correctly.

The second most obvious problem is if the IP address of the management workstation running the web browser is not configured correctly.

3. Using the *ifstat* CLI command.

To investigate a connection problem further, use the KVM console after cOS Core starts. When you press the enter key with the console, cOS Core should respond with the a standard CLI prompt. Now enter the following command once for each interface:

```
Device:/> ifstat <if-name>
```

Where *<if-name>* is the name of the cOS Core management interface. By default this is the KVM *If1* interface. This command will display a number of counters for that interface. The *ifstat* command on its own can list the names of all the KVM interfaces.

If the *Input* counters in the hardware section of the output are not increasing then the error is likely to be in the cabling. However, it may simply be that the packets are not getting to the Clavister Next Generation Firewall in the first place. This can be confirmed with a packet sniffer if it is available.

If the *Input* counters are increasing, the management interface may not be attached to the correct physical network. There may also be a problem with the routing information in any connected hosts or routers.

4. Using the *arpsnoop* CLI command.

A final diagnostic test is to try using the console command:

```
Device:/> arpsnoop -all
```

This will show the *ARP* packets being received on the different interfaces and confirm that the correct connections have been made to the correct interfaces.

6.7. System Management

Upgrades Under KVM

When running under KVM, upgrades of cOS Core are done just as they are on a non-virtualized cOS Core installation, by installing upgrade packages through the normal cOS Core user interfaces. It is not necessary to create a new virtual machine for a new version.

Increasing IPsec Performance with AES-NI

If the underlying hardware platform supports *AES-NI* acceleration, this can be made use of by cOS Core to significantly accelerate IPsec throughput when AES encryption is used. This acceleration is enabled by default.

If disabled, this feature can be enabled in the Web Interface by going to **Network > Interfaces and VPN > Advanced Settings** and clicking the checkbox **Enable AES-NI acceleration**. In the CLI, use the command:

```
Device:/> set Settings IPsecTunnelSettings AESNIEnable=Yes
```

After enabling, cOS Core must be rebooted for this option to take effect.

To check if the underlying platform supports AES-NI, use the CLI command:

```
Device:/> cpuid
```

If AES-NI is supported, *aes* will appear in the *Feature flags* list in the output from the command.

Chapter 7: High Availability Setup

This section provides the extra information needed to correctly set up a cOS Core high availability (HA) cluster under KVM. A cOS Core cluster consists of two Clavister Next Generation Firewalls, one is the *master*, the other the *slave*. Each of these firewalls will run its own separate virtual machine. The interfaces of the two gateways in a cluster need to be connected together in matching pairs through switches. It is the creation and connection of virtual switches for the cluster that is described in this section.

Important: Interface pairs should have matching bus, slot, port

In an HA cluster made up of two virtual Clavister Next Generation Firewalls, the bus, slot and port numbers of the two virtual interfaces in each HA interface pairing should be the same. If they are not, unexpected behavior could occur.

Open vSwitch Installation

HA setup with KVM requires that *Open vSwitch* is installed on the Linux system. Open vSwitch will be used to provide virtual switches so that matching interfaces of the master and slave in the cluster can be connected together. The installation of Open vSwitch itself will not be discussed further here. Refer to the software's own documentation for help with installation.

Open vSwitch is open source software that can be used in situations other than high availability to implement various networking solutions with KVM.

Note: The bridge-utils package must be removed

*Before installing Open vSwitch, the package **bridge-utils** must be removed from the Linux system.*

A Single Physical Server is Assumed

This section assumes that both the virtual firewalls in the HA cluster are installed on the same hardware server. In practice, two servers will probably be used for hardware redundancy and both will have KVM and Open vSwitch installed on them.

The configuration of the connections between two separate servers will not be discussed in this section and it is up to the administrator to choose the most appropriate way of doing this. One approach is to use *VLAN tagging* with Open vSwitch so internal bridge traffic can pass between the physical servers that make up the HA cluster.

Setup of cOS Core

The initial setup of the two separate virtual firewalls is done as normal so they are initially working as separate gateways. Before running the *HA Setup Wizard* on each unit to create the HA cluster, it is necessary to first correctly configure the virtual networking to emulate the hardware connections that would normally be present between the master and slave units.

Configuring Open vSwitch for HA

Assuming Open vSwitch has been installed, it is necessary to create KVM separate *virtual switches* so that the pairs of matching interfaces from the firewalls in the cluster are connected together on each switch.

This is done with the following steps:

> ***A.*** *Define an Open vSwitch* bridge *for each interface pair.*
>
> ***B.*** *Configure the HA master and slave virtual machines to connect the cOS Core interfaces to the relevant bridge.*

These two steps are described next.

A. Define an Open vSwitch bridge for each interface pair.

Assuming that all of the three default virtual interfaces (*If1*, *If2* and *If3*) on each firewall are to be connected together, three Open vSwitch *bridges* must be created:

- **br1-internal** - Connecting together the *If1* interfaces. This is created using the Linux command:

```
[root@linux]# ovs-vsctl add-br br1-internal
```

- **br2-external** - Connecting together the *If2* interfaces **and** also connected to the public Internet via a physical interface called *eth0* (this name will vary between configurations). This is created using the Linux commands:

```
[root@linux]# ovs-vsctl add-br br2-external
[root@linux]# ovs-vsctl add-port br2-external eth0
```

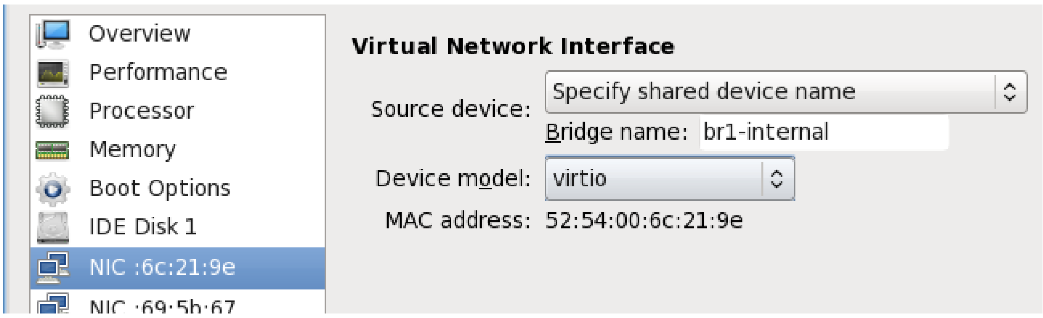
- **br3-internal** - Connecting together the *If3* interfaces. This is created using the Linux command:

```
[root@linux]# ovs-vsctl add-br br3-internal
```

B. Connect interface pairs to the relevant bridge.

It is assumed that *virt-manager* will be used to configure each of the two virtual machines in the HA cluster.

Assume that the interface *If1* is to be associated with Open vSwitch bridge *br1-internal* on both master and slave gateways. The intuitive approach is to select the NIC entry in the navigation menu that corresponds to the *If1* interface and enter the *Bridge name*:



Virtual Network Interface

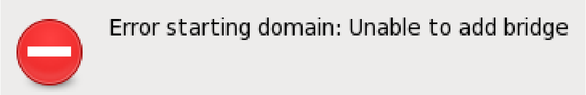
Source device: Specify shared device name

Bridge name: br1-internal

Device model: virtio

MAC address: 52:54:00:6c:21:9e

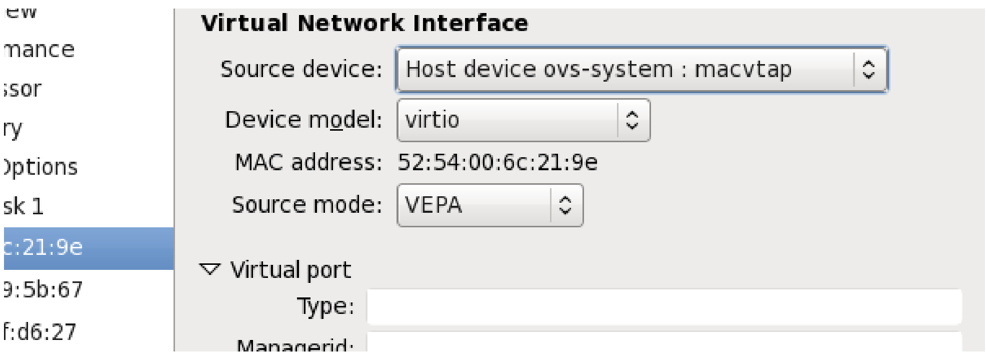
However, if this is now applied and the virtual machine started, it will give an error message:



Error starting domain: Unable to add bridge

To get around this issue, allocate the Open vSwitch bridge using the following steps:

1. Open the properties of the HA cluster's master firewall in *virt-manager*. Change the *Source device* to be something using *macvtap* so that the *Type* of the *Virtual port* can be set:



Virtual Network Interface

Source device: Host device ovs-system : macvtap

Device model: virtio

MAC address: 52:54:00:6c:21:9e

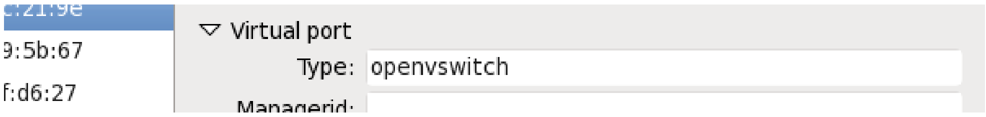
Source mode: VEPA

Virtual port

Type:

Managerid:

2. Set the *Type* to be *openvswitch*.

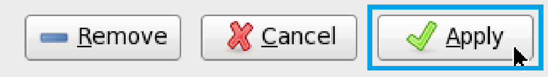


Virtual port

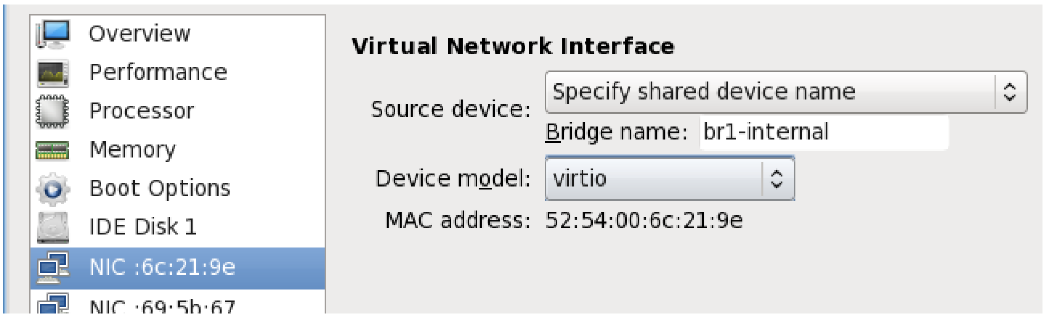
Type: openvswitch

Managerid:

3. Save this setting by selecting the **Apply** button.



4. Now, change the *Source device* setting back to *Specify shared device name* and set it to be the Open vSwitch bridge connected to the interface. In this case, *br1-internal*.



Virtual Network Interface

Source device: Specify shared device name

Bridge name: br1-internal

Device model: virtio

MAC address: 52:54:00:6c:21:9e

5. Now select the **Apply** button and repeat the process with the remaining interfaces *If2* and *If3*, connecting them to the bridges *br2-external* and *br3-internal*.

6. Repeat the process for the slave firewall.

The networking for an HA cluster on a single hardware server is now complete. When the firewalls are on different servers, the procedure is similar. However, the administrator should then decide how they want to connect the Open vSwitch bridges on each server together. *VLAN tagging* can be used to separate the internal bridges on each server. Each pair of cluster interfaces uses a different VLAN ID to separate its traffic from the other pairs of interfaces.

Chapter 8: SR-IOV Setup

Overview

Single Root I/O Virtualization (SR-IOV) is a specification that can allow direct access to an external PCI Ethernet interface by cOS Core running under KVM. It is only available on Intel based hardware.

The direct access provided by SR-IOV can give dramatically higher traffic throughput capability for a virtual Clavister Next Generation Firewall since it circumvents the overhead involved with normal virtual interfaces. A disadvantage of using SR-IOV is the static nature of configurations that use it.

Important: SR-IOV consumes an entire core

*When SR-IOV is enabled, cOS Core will consume **virtually all** the resources of the processor core on which it runs. This is true even if cOS Core has no traffic load. The reason for this is that SR-IOV uses continuous interface polling to check for new traffic.*

SR-IOV Interfaces for cOS Core

By default, cOS Core provides three virtual Ethernet ports with the logical cOS Core names **I1**, **I2** and **I3**. The setup procedure described in this section adds hardware PCI Ethernet ports as additional interfaces with logical cOS Core names **I4**, **I5** and so on.

Once the setup is complete, only traffic routed through these additional ports will benefit from the throughput increases provided by SR-IOV.

Prerequisites for SR-IOV

In order to make use of SR-IOV with cOS Core under KVM, the following is required:

- Support for IOMMU with IOMMU enabled in the BIOS.
- Hardware support for SR-IOV with Intel™ VT-d or AMD-Vi.
- Support for SR-IOV enabled in the BIOS.
- Available slots supporting PCI Express v2.0 (5.0GT/s) x8 Lanes with ARI and ACS.
- An Intel Ethernet Converged Network Adapter x520/X540 with Intel 82599 chipsets (10 Gb) or

an Intel i350 adapter (1 Gb).

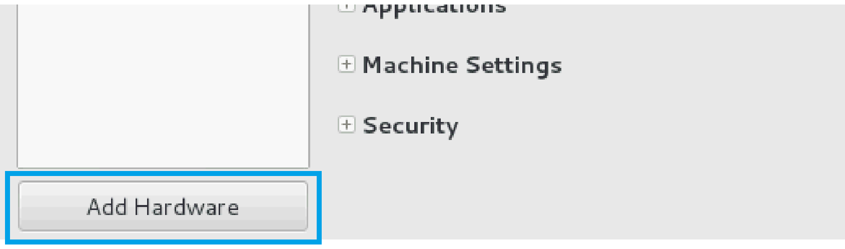
Set up of the hardware platform for virtualization is not discussed further here. For details on this subject refer to the Intel document entitled: ***Using Intel Ethernet and the PCI-SIG Single Root I/O Virtualization (SR-IOV) and Sharing Specification on Red Hat Enterprise Linux*** .

Adding SR-IOV Interfaces

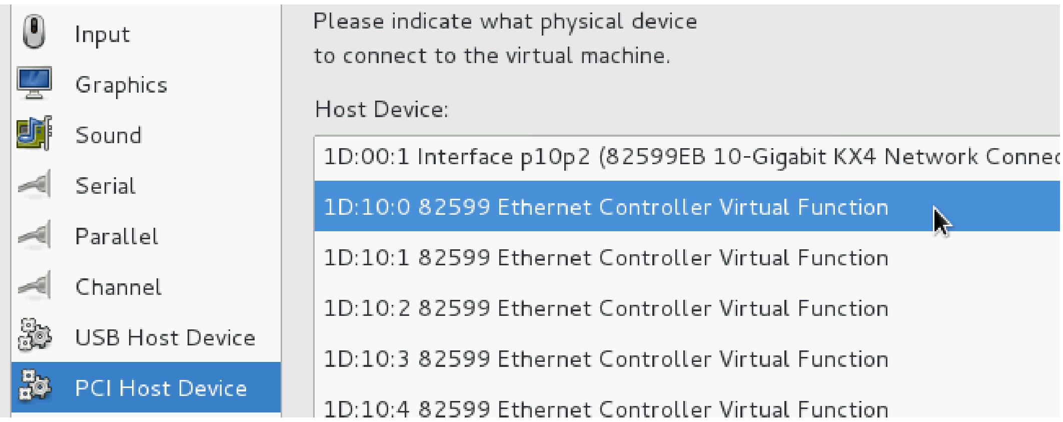
The following are the steps for SR-IOV interface setup with cOS Core:

1. If it is running, stop the cOS Core virtual machine.

2. In *virt-manager*, select **Add Hardware**.



3. Select **PCI Host Device** and the correct virtual function. For the first PCI device, the final digit in the numeric designation on the left should be even (below it is **1D:10:0**). Press **Finish** to add the device.

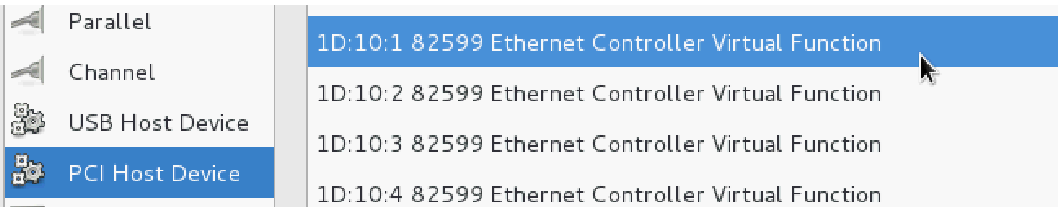


Please indicate what physical device to connect to the virtual machine.

Host Device:

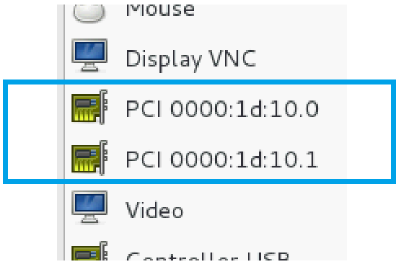
1D:00:1 Interface p10p2 (82599EB 10-Gigabit KX4 Network Connec
1D:10:0 82599 Ethernet Controller Virtual Function
1D:10:1 82599 Ethernet Controller Virtual Function
1D:10:2 82599 Ethernet Controller Virtual Function
1D:10:3 82599 Ethernet Controller Virtual Function
1D:10:4 82599 Ethernet Controller Virtual Function

4. The same is repeated if a second PCI device is added but the final digit of the selected virtual function should be odd.



1D:10:1 82599 Ethernet Controller Virtual Function
1D:10:2 82599 Ethernet Controller Virtual Function
1D:10:3 82599 Ethernet Controller Virtual Function
1D:10:4 82599 Ethernet Controller Virtual Function

5. The new adapters are now listed in *virt-manager*.



6. Start the cOS Core virtual machine.

7. Start cOS Core again and issue the following console CLI command:

```
Device:/> pciscan -cfgupdate
```

   cOS Core will scan the available interfaces and include the added PCI interfaces into the configuration. Some example output is shown below.

8. Finally, save the configuration changes using the following commands:

```
Device:/> activate

Device:/> commit
```

Note: Do not pre-assign the SR-IOV MAC address

For any usage of SR-IOV interfaces with cOS Core, the MAC address should not be preassigned by the hypervisor so that it is fixed. This will prevent cOS Core from controlling the MAC address which can be needed in certain circumstances.

Achieving Maximum Throughput

Once the SR-IOV interfaces exist as logical interfaces in cOS Core they can used for both receiving and sending in traffic as well as being part of rule sets and other cOS Core objects.

On order to reach much higher throughput speeds, traffic must both enter and leave the firewall via SR-IOV interfaces. Having the traffic enter or leave on a normal interface will create a bottleneck, reducing throughput back to non-SR-IOV speeds.

Features Not Supported by SR-IOV Interfaces

The following cOS Core features are not supported by SR-IOV interfaces:

- Proxy ARP/ND using XPUBLISH is not supported.

- Multicast is not supported.

Chapter 9: FAQ

This appendix collects together answers to a selection of *Frequently Asked Questions* that can be helpful in solving various issues with cOS Core running under KVM.

Question Summary

1. The 2 hour cOS Core demo mode time limit has expired. What do I do?
2. Are upgrades of cOS Core done differently under KVM?
3. How do I release the focus from the KVM console window?
4. Do all virtual interfaces have to be configured as virtio NICs?
5. How do I manage multiple virtual firewalls?
6. How much increase in throughput can SR-IOV provide?

Questions and Answers

1. The 2 hour cOS Core demo mode time limit has expired. What do I do?

cOS Core will not respond after it enters *lockdown mode* after 2 hours and will consume all the KVM resources. In this situation, the KVM virtual machine must be stopped and then restarted so that cOS Core restarts and enters a new 2 hour evaluation period.

2. Are upgrades of cOS Core done differently under KVM?

No. cOS Core upgrades are performed under KVM just as they would be in non-KVM environments.

3. How do I release the focus from the KVM console window?

KVM keeps focus in the console window. To click outside the console window, press the key combination **Ctrl-Alt** .

4. Do all virtual interfaces have to be configured as virtio NICs?

Yes. cOS Core will not work with virtual interfaces that are not configured to use the virtio driver. Any added interfaces must have the *Device model* property set to *virtio*.

5. How do I manage multiple virtual firewalls?

The IP address of the management virtual Ethernet interface for cOS Core must be different for the different virtual firewalls running under a single hypervisor.

6. How much increase in throughput can SR-IOV provide?

The performance increase provided by SR-IOV can be dramatic if traffic both enters and leaves via SR-IOV interfaces. A quadrupling of maximum throughput is possible.

Appendix A: Windows 7 IP Setup

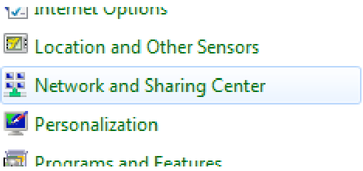
If a PC running Microsoft Windows 7 is being used as the cOS Core management workstation, the computer's Ethernet interface connected to the Clavister Next Generation Firewall must be configured with an IP address which belongs to the network *192.168.1.0/24* and is different from the firewall's address of *192.168.1.1*.

The IP address *192.168.1.30* will be used for this purpose and the steps to set this up with Windows 7 are as follows:

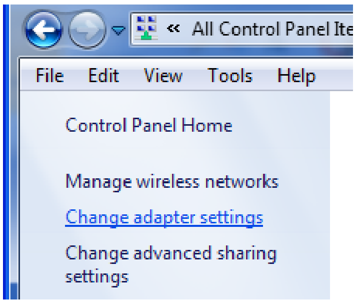
1. Press the Windows **Start** button.

2. Select the **Control Panel** from the start menu.

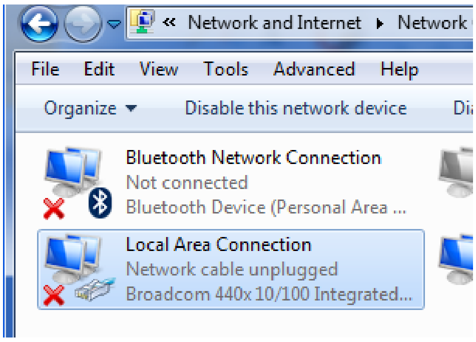
3. Select **Network & Sharing Center** from the control panel.



4. Select the **Change adapter settings** option.

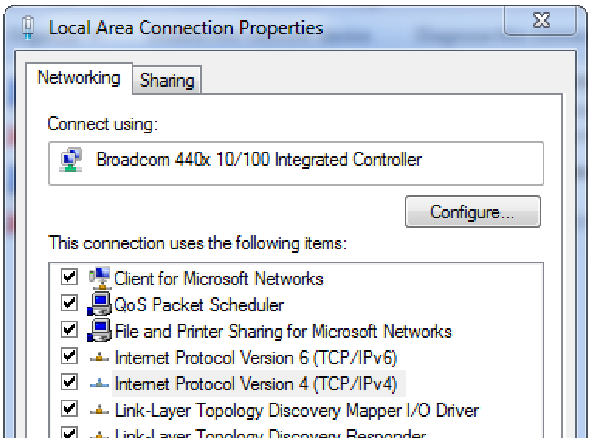


5. A list of adapters will appear and will include the Ethernet interfaces. Select the interface that will connect to the firewall.



« Network and Internet ▸ Network
File Edit View Tools Advanced Help
Organize ▾ Disable this network device Di
Bluetooth Network Connection
Not connected
Bluetooth Device (Personal Area ...
Local Area Connection
Network cable unplugged
Broadcom 440x 10/100 Integrated...

6. The properties for the selected interface will appear.



Local Area Connection Properties

Connect using:

Broadcom 440x 10/100 Integrated Controller

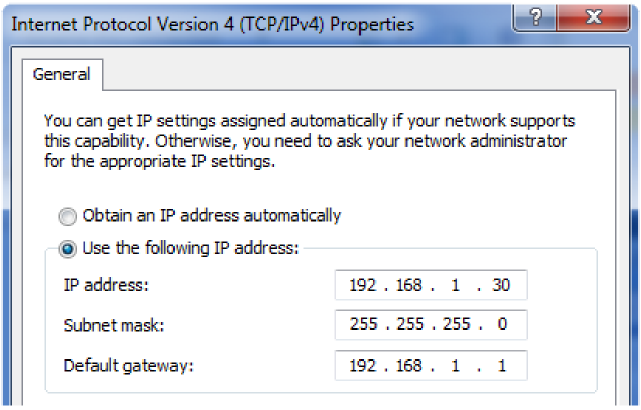
This connection uses the following items:

- ☑ Client for Microsoft Networks
- ☑ QoS Packet Scheduler
- ☑ File and Printer Sharing for Microsoft Networks
- ☑ Internet Protocol Version 6 (TCP/IPv6)
- ☑ Internet Protocol Version 4 (TCP/IPv4)
- ☑ Link-Layer Topology Discovery Mapper I/O Driver
- ☑ Link-Layer Topology Discovery Responder

Select and display the properties for *Internet Protocol Version 4 (TCP/IPv4)*.

7. In the properties dialog, select the option **Use the following IP address** and enter the following values:

   - **IP Address:** *192.168.1.30*
   - **Subnet mask:** *255.255.255.0*
   - **Default gateway:** *192.168.1.1*



Internet Protocol Version 4 (TCP/IPv4) Properties

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically

● Use the following IP address:

IP address: 192 . 168 . 1 . 30

Subnet mask: 255 . 255 . 255 . 0

Default gateway: 192 . 168 . 1 . 1

DNS addresses can be entered later once Internet access is established.

8. Click **OK** to close this dialog and close all the other dialogs opened since step **(1)**.

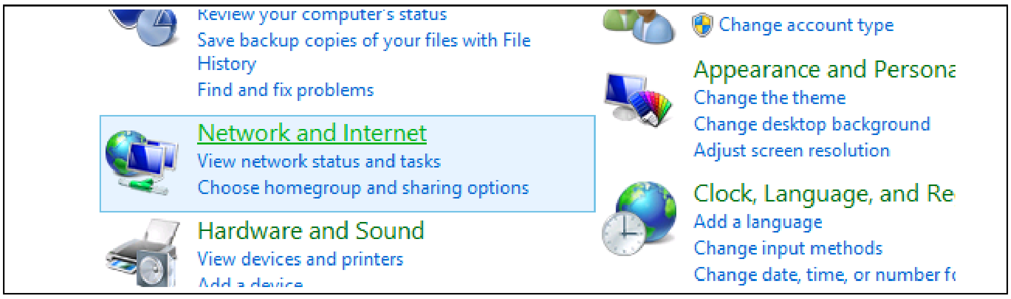
Appendix B: Windows 8/8.1/10 IP Setup

If a computer running Windows is being used as the cOS Core management workstation and a DHCP server is not enabled on the cOS Core management interface, the management computer's Ethernet interface connected to the Clavister Next Generation Firewall should be configured with an IPv4 address which belongs to the network *192.168.1.0/24*. That address must be different from the firewall's default management interface address of *192.168.1.1*.

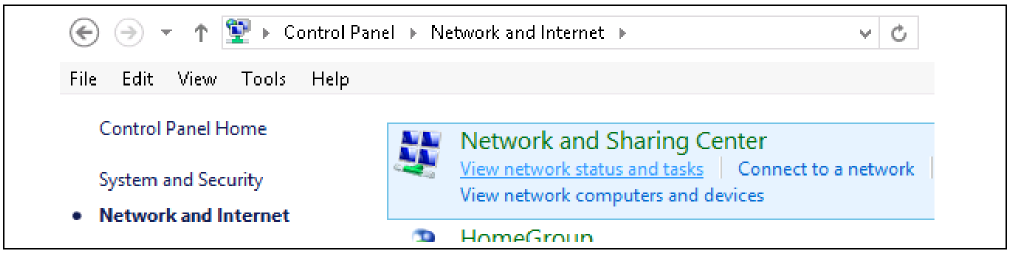
The IPv4 address *192.168.1.30* will be used for this purpose and the steps to set this up with Windows 8, 8.1 or 10 are as follows:

1. Open the Windows **Control Panel** (the *Category* view is assumed here).

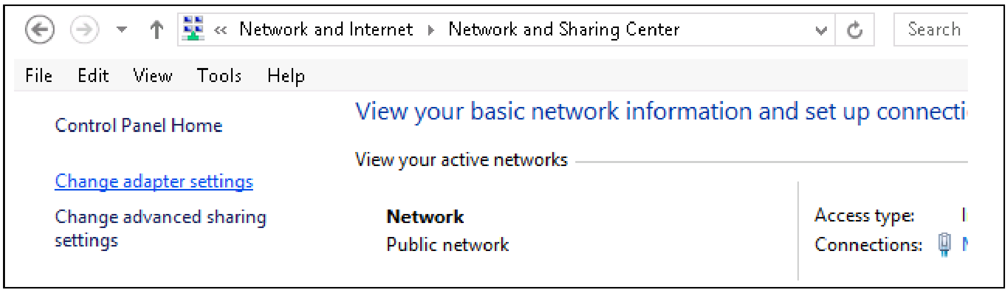
2. Select **Network & Internet** from the control panel.


Review your computer's status
Save backup copies of your files with File History
Find and fix problems
Network and Internet
View network status and tasks
Choose homegroup and sharing options
Hardware and Sound
View devices and printers
Add a device
Change account type
Appearance and Persona
Change the theme
Change desktop background
Adjust screen resolution
Clock, Language, and Re
Add a language
Change input methods
Change date, time, or number f

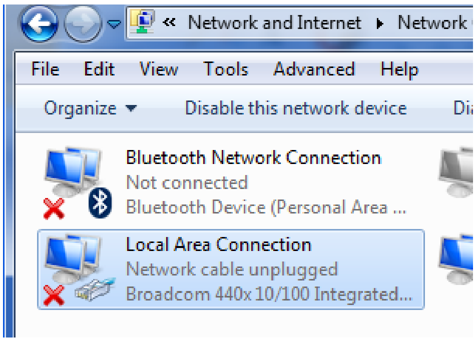
3. Then, select the **Network & Sharing Center** option.


Control Panel ▸ Network and Internet ▸
File Edit View Tools Help
Control Panel Home
System and Security
Network and Internet
Network and Sharing Center
View network status and tasks
Connect to a network
View network computers and devices
HomeGroup

4. Now, select the **Change adapter settings** option.


« Network and Internet ▸ Network and Sharing Center
Search
File Edit View Tools Help
Control Panel Home
Change adapter settings
Change advanced sharing settings
View your basic network information and set up connecti
View your active networks
Network
Public network
Access type:
Connections:

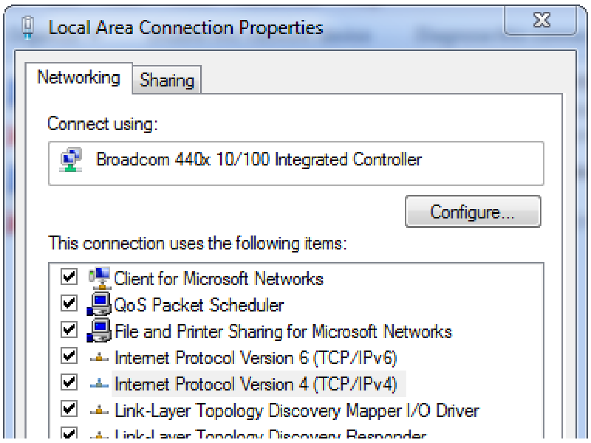
5. A list of adapters will appear and will include the Ethernet interfaces. Select the interface that will connect to the firewall.


« Network and Internet ▸ Network
File Edit View Tools Advanced Help
Organize ▾ Disable this network device Di

Bluetooth Network Connection
Not connected
Bluetooth Device (Personal Area ...

Local Area Connection
Network cable unplugged
Broadcom 440x 10/100 Integrated...

6. The properties for the selected interface will appear.



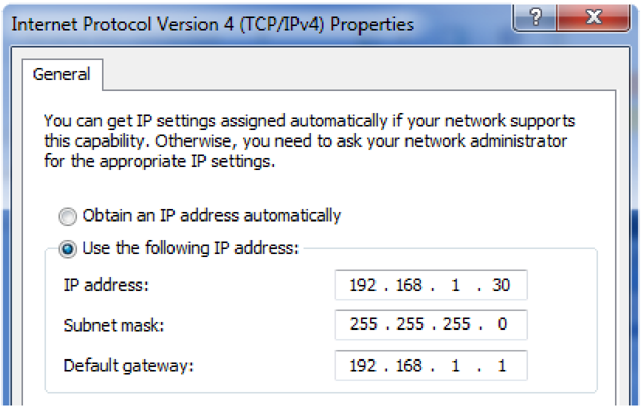
Connect using:
Broadcom 440x 10/100 Integrated Controller

This connection uses the following items:
☑ Client for Microsoft Networks
☑ QoS Packet Scheduler
☑ File and Printer Sharing for Microsoft Networks
☑ Internet Protocol Version 6 (TCP/IPv6)
☑ Internet Protocol Version 4 (TCP/IPv4)
☑ Link-Layer Topology Discovery Mapper I/O Driver
☑ Link-Layer Topology Discovery Responder

Select and display the properties for *Internet Protocol Version 4 (TCP/IPv4)*.

7. In the properties dialog, select the option **Use the following IP address** and enter the following values:

- **IP Address:** *192.168.1.30*
- **Subnet mask:** *255.255.255.0*
- **Default gateway:** *192.168.1.1*



You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically
● Use the following IP address:
IP address: 192 . 168 . 1 . 30
Subnet mask: 255 . 255 . 255 . 0
Default gateway: 192 . 168 . 1 . 1

DNS addresses can be entered later once Internet access is established.

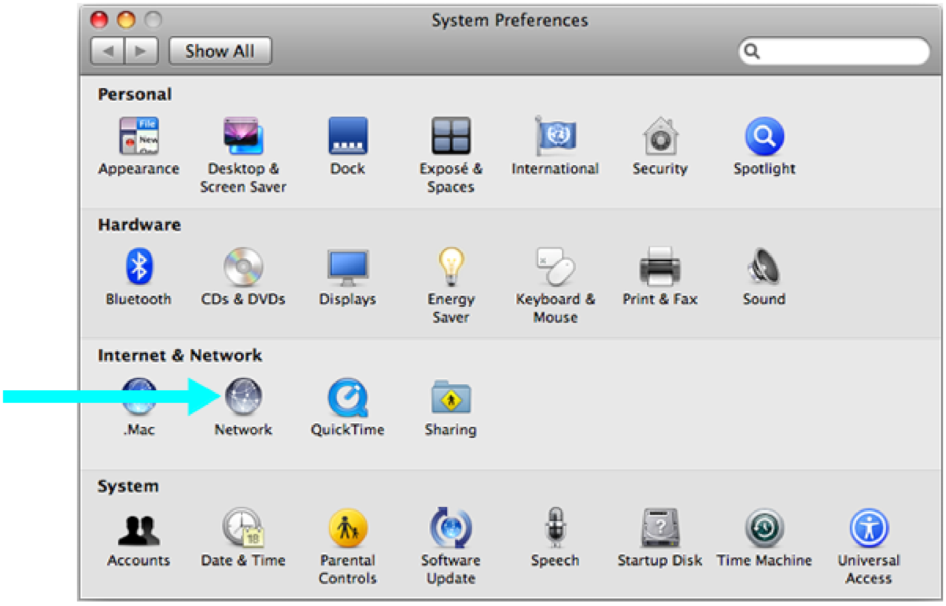
8. Click **OK** to close this dialog and close all the other dialogs opened since step **(1)**.

Appendix C: Apple Mac IP Setup

An Apple Mac can be used as the management workstation for initial setup of a Clavister Next Generation Firewall. To do this, a selected Ethernet interface on the Mac must be configured correctly with a static IP. The setup steps for this with Mac OS X are:

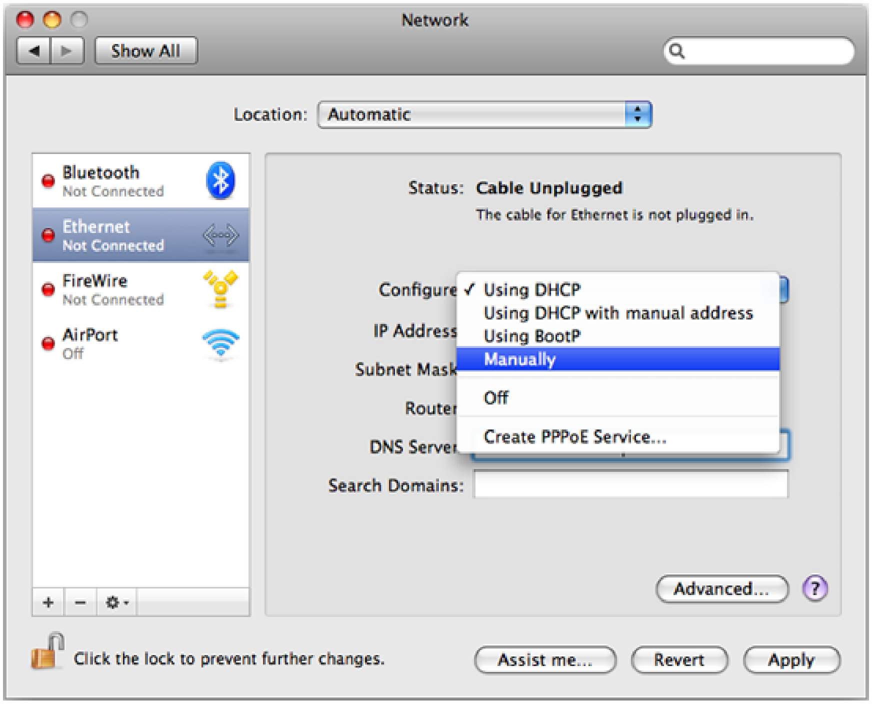
1. Go to the **Apple Menu** and select **System Preferences**.

2. Click on **Network**.



3. Select **Ethernet** from the left sidebar menu.

4. Select **Manually** in the **Configure** pull down menu.



Location: Automatic

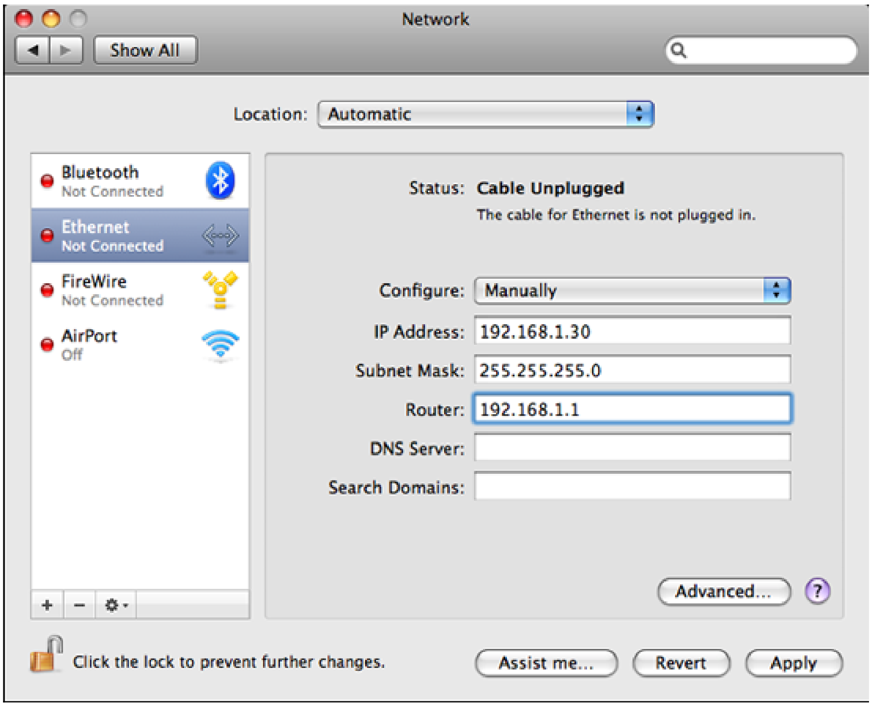
Status: **Cable Unplugged**

The cable for Ethernet is not plugged in.

Click the lock to prevent further changes.

5. Now set the following values:

   - **IP Address:** *192.168.1.30*
   - **Subnet Mask:** *255.255.255.0*
   - **Router:** *192.168.1.1*



Location: Automatic

Status: Cable Unplugged
The cable for Ethernet is not plugged in.

Configure: Manually
IP Address: 192.168.1.30
Subnet Mask: 255.255.255.0
Router: 192.168.1.1
DNS Server:
Search Domains:

Click the lock to prevent further changes.

6. Click **Apply** to complete the static IP setup.

CONNECT. PROTECT

Clavister AB
Sjögatan 6J
SE-89160 Örnsköldsvik
SWEDEN

Head office/Sales: +46-(0)660-299200
Customer support: +46-(0)660-297755

www.clavister.com