



# Cisco Modeling Labs 1.6 Release Notes

**First Published:** June 12, 2020

**Last Updated:** June 12, 2020

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies are considered un-Controlled copies and the original on-line version should be referred to for latest version.

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco website at [www.cisco.com/go/offices](http://www.cisco.com/go/offices).

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: [www.cisco.com/go/trademarks](http://www.cisco.com/go/trademarks). Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

© 2020 Cisco Systems, Inc. All rights reserved.

## Contents

<b><i>Preface</i></b>	<b>4</b>
<b><i>Start Here</i></b>	<b>6</b>
<b><i>Availability</i></b>	<b>6</b>
<b><i>Software Version History</i></b>	<b>6</b>
<b><i>Enhancements and New Features</i></b>	<b>6</b>
<b><i>Migrating to CML 1.6</i></b>	<b>11</b>
<b><i>CML Server Component Versions</i></b>	<b>12</b>
<b><i>Important Notes</i></b>	<b>12</b>
<b><i>Known Issues and Caveats</i></b>	<b>13</b>

## Preface

### Documentation Conventions

This document uses the following conventions.

Convention	Indication
<b>bold font</b>	Commands and keywords and user-entered text appear in <b>bold font</b> .
<i>italic font</i>	Document titles, new or emphasized terms, and arguments for which you supply values are in <i>italic font</i> .
[ ]	Elements in square brackets are optional.
{ x   y   z }	Required alternative keywords are grouped in braces and separated by vertical bars.
[ x   y   z ]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
courier font	Terminal sessions and information the system displays appear in <i>courier font</i> .
< >	Nonprinting characters such as passwords are in angle brackets.
[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

**Note:** Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the manual.

**Caution:** Means *reader be careful*. In this situation, you might perform an action that could result in equipment damage or loss of data.

**Warning:**

---

#### IMPORTANT SAFETY INSTRUCTIONS

**Means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Use the statement number provided at the end of each warning to locate its translation in the translated safety warnings that accompanied this device.**

#### SAVE THESE INSTRUCTIONS

Regulatory: Provided for additional information and to comply with regulatory and customer requirements.

---

## Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, using the Cisco Bug Search Tool (BST), submitting a service request, and gathering additional information, see [What's New in Cisco Product Documentation](#).

To receive new and revised Cisco technical content directly to your desktop, you can subscribe to the [What's New in Cisco Product Documentation RSS feed](#). The RSS feeds are a free service.

## Start Here

The CML 1.6 release is an incremental update to CML 1.5, including bug fixes, updated reference platforms, and some enhancements since the CML 1.5.145 release. The major additions to this release are support for IPv6 on the primary and shared networks, and the availability of a set of additional shared networks. Unlike in the 1.5 release, the changes introduced in CML 1.6 have not affected the configuration and installation process in significant ways.

### NOTE

- If deploying on a Cisco UCS C220M4 with Cisco 12G SAS Modular RAID Controller you must enable RAMdisk using the UWM System Configuration pages in order to support IOS XRv images.

## Availability

Installation images for VMware Workstation, Workstation Player, Fusion Pro, ESXi and bare metal systems (ISO) are available now.

## Software Version History

Version	Release Date	Initial Release
CML1.6.65	July-8-2019	Initial release
CML1.6.67	June-12-2020	Bug fixes for SaltStack security vulnerabilities.

## Enhancements and New Features

### IPv6 access on the primary network

With the 1.6 release, both automatic and static optional configuration of an IPv6 address is available on the primary (management) network used to access the CML deployment. Either at install time, in the startup configuration, or post-deployment, using the `vir1_setup` tool, the primary network's addressing can be adjusted. All required CML services are available via both IPv4 and IPv6. Static IP configuration of cluster compute nodes is available in the UWM System Configuration, or in the `vir1_setup` tool. Automatic configuration on the primary network can be switched from SLAAC to DHCPv6, if required in a particular deployment environment.

### Additional shared networks

Up to six shared networks are made available in the UWM System configuration, in addition to the original pair of shared networks, named 'flat' and 'flat1'. These new networks are named composed of a fixed prefix 'shared-', and a configurable suffix; the expectation is to use this suffix to distinguish these new networks by their purpose, or equipment connected through them.

Like with the original two shared networks, a physical interface of the main CML (controller) host may be selected to provide a connection of this network with the outside lab environment. This is optional, hence no additional interfaces are required on the host in order to install and use CML 1.6. Without an external interface, the shared networks are available to bridge together nodes from different simulations running in the same deployment.

Both an IPv4 and an IPv6 subnet may be configured on any of the eight shared networks. The original two shared networks must have an IPv4 subnet configured for compatibility reasons. The new shared networks require at least one subnet to be configured.

The IPv6 subnet for a shared network is fixed with a /64 prefix. The simulation nodes shall receive their IPv6 addresses via SLAAC, provided by the built-in facility. Thus, the IPv6 address of each host is computed from their interface MAC address, which can be controlled using the 'static\_mac' extension. This extension can be set of a node to control the management interface addressing, or on the L2 FLAT element connected to a data interface used for this shared network.

For Cisco router nodes that don't support SLAAC, notably IOS XRv, the initial configuration may contain a '(no)ipv6 address' line in the initial configuration. The correct IPv6 address will be injected into that line when the node is starting, using the same mechanism employed for IPv4 addresses on nodes that don't support DHCP.

In case the nodes in the shared network are supposed to communicate IPv6 traffic through to other networks in the outside lab environment, a suitable router is expected to be configured in the outside lab. That router should be advertising the same IPv6 prefix to avoid misconfiguration.

## Controlling reported host URLs and IP addresses (in a cluster)

The location of the CML host used by individual users may vary in particular scenarios, such as if a NAT is employed between the users and the deployed lab, or if the CML services are being accessed over a shared network and an OpenVPN connection. The CML services do not make assumptions over the presence and validity of a DNS system in the lab where it is deployed either - no reverse lookups are used to determine a public hostname for any CML host, be it the controller or compute hosts in a cluster.

A configuration option named 'virl\_local\_ip' was used in these situations to inform the CML services what hostname or IP to use to refer to itself. No equivalent option existed for compute hosts in a cluster; this resulted in the reported links for serial consoles of nodes deployed to computes using the primary interface IP address of that compute host, which is usually incorrect in such scenarios.

The generated links in both UWM and the desktop UI now consistently reuse the same host that the client is using, be it a hostname or IP address. Thus, for single-host deployments, and the controller node in a cluster, there is no need to configure virl\_local\_ip or anything else in this respect, anymore.

In a cluster deployment, in cases where the primary network address of the compute is not to be used, a different configuration can be applied. Log into the CML controller node, and issue a command such as `sudo crudini --set /etc/virl/virl-core.ini cluster resolve_compute1 "public-hostname-or-ip"`. For controller that has the set hostname 'X', use the 'resolve\_X' to force the value regardless of what the user configured; this is not required, and may be counterproductive though. After the configurations are present for all hosts, use `sudo systemctl restart virl-std virl-uwm` to apply the changes.

## Improved control over generated MAC addresses

The MAC addresses of simulation nodes' interfaces are assigned by one of three mechanisms: 1. static\_mac - for management interfaces, and interfaces connected to L2 FLAT or L3 SNAT simulation elements; 2. management addresses are generated using a pattern consisting of a constant prefix, and a unique simulation and node number; and 3; all other interfaces use a random MAC address with a constant prefix like 'fa:16:3e'.

In cases where shared networks are inter-connected between multiple CML or OpenStack deployments, the same address might be assigned to two nodes. The UWM system configuration can now be used to control and avoid clashes by picking a different prefix for addresses generated by the second and third mechanism.

## Baremetal installations use hardware enablement kernels

The ISO installer contains a newer Linux kernel release, which is run by default on all new deployments. This kernel should allow for certain newer hardware components to be recognized properly when installed on a physical computer. The older kernel can still be selected when booting the server from the bootloader menus.

Due to performance issues with some newer kernels available by default to the Ubuntu 16.04 (Xenial) system, the new kernel is installed from a Ubuntu 18.04 (Bionic) repository. Only the kernel packages shall be installed from this repository, all other packages are disabled from being considered for installation.

The ISO installer still does not support installations in UEFI mode; servers should be configured to Legacy (BIOS) mode before installation, as described in installation instructions.

## Bug fixes and minor enhancements in 1.6.65

- Autonetkit/Live visualization
  - VIRLDEV-4878 - IOSvL2 configuration for the cisco/cisco account consistent with IOSv
  - VIRLDEV-4738 - Custom configuration for router loopback 0 interface missing from generated configuration
  - VIRLDEV-6561 - Live vis services failed to communicate after port changes
  - VIRLDEV-6165 - Live vis path traces from LXC nodes were failing due to incorrect MTU settings in the LXC
  - VIRLDEV-4922 - Collect log action fails for NX-OSv
  - VIRLDEV-6357 - Connecting to management IP of server and LXC nodes failed
- CML Client
  - VIRLDEV-6375 - Upgrade the bundled java versions
  - VIRLDEV-6242 - Offer current shared networks for L2 FLAT host network setting
  - VIRLDEV-6198 - Allow setting backend service URL using an IPv6 address
  - VIRLDEV-6022 - Remember size of the extension edit window
  - VIRLDEV-6599 - Remove dependency on potentially vulnerable package
  - VIRLDEV-6707 - Sign the Mac OS X GUI application with an official Cisco key
- UWM
  - VIRLDEV-5570 - Remember previous values for some system configuration values for proper cleanup
  - VIRLDEV-5571 - Fix order of system configuration actions when multiple system configuration values are changed
  - VIRLDEV-6361 - Background tasks may leak memory and stop updating external state
  - VIRLDEV-5851 - Cluster health was green when active computes were missing
  - VIRLDEV-6180 - Do not report license failure before activation
  - VIRLDEV-6240 - Ensure each compute host is used in cluster system operation check



Enhancements and New Features

- VIRTDEV-6246 - Add support for Git operations via REST API
- VIRTDEV-6133 - Multiple user creation does not work as expected
- VIRTDEV-6307 - Support git source for browser-based editing of a .virl file
- VIRTDEV-6348 - Fix preview of .virl file synced from the browser-based editor
- VIRTDEV-6451 - Fix API wrongly returning 401 unauthorized instead of 403 forbidden if resource operation is prohibited by System configuration controls
- VIRTDEV-5488 - SSH console can be higher than browser window, failing to display most recent lines
- VIRTDEV-6523 - Volumes cannot be created
- VIRTDEV-6241 - System operation check fails in maintenance mode
- VIRTDEV-6164 - OpenVPN configuration file is being cached by browsers unchanged even when modified during system configuration
- VIRTDEV-6008 - Restrict access to VM Control pages to uwmadmin based on system configuration.

■ CML Core

- VIRTDEV-6079 - allow up to 128 data interfaces in a node subtype. Note that large numbers of interfaces on a node may slow down its deployment and teardown, in addition to potential increased resource requirements by the node itself. All data interfaces from first up to the highest interface present on a node in the topology file must be simulated.
- VIRTDEV-6072 - Interface count for IOS XRv 9000 was lowered to 8 data interfaces. This matches official documentation of the current IOS XRv 9000 images. Future releases of IOS XRv 9000 may increase the number of available interfaces
- VIRTDEV-6073 - Interface count for NX-OSv 9000 was increased to 64 data interfaces, and its hw\_bios switched to OVMF.fd.
- Note that operating a NX-OSv 9000 node with more interfaces places increased demands on memory for the node, meaning that a different flavor may be required for such nodes. See also known issue VIRTDEV-6416.
- VIRTDEV-4947 - Allow node to set its serial and machine UUID. Due to infrastructure constraints, only one node with a given UUID may be placed in the CML system at a time.
- VIRTDEV-6555 - Change method of connecting static serial ports with node serial consoles directly
- VIRTDEV-6528 - Allow nodes to specify common names for dummy interface networks. A node extension allows connecting special interfaces between management and data interfaces
- VIRTDEV-6258 - Initial configuration management/shared/snatch IP address injection shall not rewrite 'ip address dhcp'
- VIRTDEV-6061 - Traffic capture does not start for node in a site element
- VIRTDEV-6095 - Docker node in a site, or with non-standard name does not start
- VIRTDEV-6169 - Fix apparent overlap in TCP port ranges
- VIRTDEV-6105 - Live capture does not report port clashes

- VIRTDEV-6817 - Increased memory limit in lxc-sshd (and derived subtypes including management LXC) to 32 MB to avoid shutoffs due to limit being reached
- VIRTDEV-6888 - HWE kernel dropped overlayfs alias for overlay filesystem
- Infrastructure
  - VIRTDEV-5457 - Disabled screensaver on baremetal console
  - VIRTDEV-6079 - Support PCI multifunction for interfaces. An image property can control how interface hardware is simulated; may be useful if more than 28 interfaces are simulated on a node
  - VIRTDEV-6098 - Nodes are not proportionately distributed across cluster. The main weight is now placed on the number of unoccupied CPUs on a host
  - VIRTDEV-6102 - OpenStack and other infrastructure services only listen on internal interfaces
  - VIRTDEV-6116 - qemu was generating large amounts of messages in individual VM logs (IOSv nodes) wrt register access
  - VIRTDEV-6100 - add generated password protection to redis service
  - VIRTDEV-6254 - randomize keystone service token
  - VIRTDEV-6159 - generate new OpenVPN keys whenever OpenVPN is re-enabled
  - VIRTDEV-6267 - propagate built bridge module to computes
  - VIRTDEV-6201 - Switch to docker community edition; update coreos image
  - VIRTDEV-6224 - Compute minion validates MTU on cluster network on startup
  - VIRTDEV-6343 - Do not run Apache on compute hosts
  - VIRTDEV-6338 - Increase timeout of libvirtd waiting on kvm to avoid nodes going into error state on smaller hosts
  - VIRTDEV-6312 - Upgrade from obsolete and vulnerable pycrypto to pycryptodome
  - VIRTDEV-6313 - Upgrade from theoretically vulnerable paramiko versions
  - VIRTDEV-6231 - Change login message of disabled computes
  - VIRTDEV-6187 - Allow changing cluster subnet on non-cluster deployment
  - VIRTDEV-6543 - Disabling cluster in viri\_setup does not work as expected
  - VIRTDEV-6317 - Kernel bridge module must be compiled with 'retpoline'-supporting gcc on latest kernels
  - VIRTDEV-6342 - Make system services wait for interfaces they depend on
  - VIRTDEV-6745 - Allow VM metadata to be downloaded using neutron metadata service

## Bug fixes and minor enhancements in 1.6.67

- Infrastructure
  - VIRTDEV-7395 Fix ARP neighbor table overflow

## Migrating to CML 1.6

- VIRTDEV-7409 Set default value for 'licence\_renew\_interval'
- VIRTDEV-7411 Update OpenStack patches to match the latest OpenStack modules
- VIRTDEV-7421 Upgrade salt version to 3000.x
- Increase limit for virtlogd
- Flush keystone token table periodically
- Pin Werkzeug to version below 1.0.0

## Security vulnerabilities in Salt

All CML releases older than 1.6.67 are using a version of salt, 2017.7.2, that has known security vulnerabilities, CVE-2020-11651 and CVE-2020-11652. See the [Cisco Security Advisory cisco-sa-salt-2vx545AG](#) for more information about the vulnerabilities and whether your CML instance may have been running a vulnerable service.

CML 1.6.67 fixes the problem by using a newer version of salt, 3000.3, which contains fixes for CVE-2020-11651 and CVE-2020-11652.

## Migrating to CML 1.6

### Performing a New Installation

You must perform a full reinstallation to migrate to CML 1.6. Please use the updated CML 1.6 installation guides posted at <https://www.cisco.com/c/en/us/support/cloud-systems-management/modeling-labs/products-installation-guides-list.html>. Select the instructions on that site appropriate for your selected installation option.

### Upgrade CML Client

You should update CML Client to the most recent 1.6.0 build. Older releases should still work since there were no changes in the file format or APIs from CML 1.2 or 1.3. However, running the latest version is generally recommended.

To download the new CML Client:

1. Open a web browser and navigate to the CML host or virtual machine's IP address.
2. Login to the User Workspace Management (UWM).
3. Select **CML Server** from the menu that appears on the left.
4. Select the **Download** sub-menu.
5. Select **CML Client** from the list of options.
6. From the list of files presented, download the CML Client appropriate to your local platform (setup EXE for Windows, DMG for OS X, or zip file for Linux).
7. Install CML Client.

Once you have installed CML Client, you may want to update the node types shown in the Palette to match any changes on the CML server:

1. Launch CML Client

## CML Server Component Versions

2. Select **File > Preferences > Node Subtypes**.
3. Click the **Fetch From Server** button.
4. Click OK.

## CML Server Component Versions

This release contains the following component versions:

- OpenStack Mitaka
- CML\_CORE 0.10.37.35
- AutoNetkit 0.24.1/0.23.12
- Topology Visualization Engine 0.17.28
- Live Network Collection Engine 0.12.11
- CML Client 1.6.0-534

## Cisco Platform VMs

Baremetal ISO installers bundle fewer images, which may be added later through UWM.

- IOSv - 15.7(3)M (New)
- IOSv L2 - 15.2.1 (06.2018) (New)
- IOS XRv - 6.1.3 image
- ASAv 9.9.2 (New)
- CoreOS 1632.2.1 (New)
- Ubuntu 16.04.3 Cloud-init image

## Linux Container Images

- Ubuntu 16.04 LXC
- iPerf 2.0.2 LXC
- Ostinato-drone 0.8 LXC

## Important Notes

### CML System Scaling

- Do NOT oversubscribe hardware resources at multiple levels.

Known Issues and Caveats

- It is possible to oversubscribe CPU and memory resources at both the CML System Configuration level and at the VMware ESXi level.
- By default, CML applies an oversubscription factor of 2.0 for memory resources and 3.0 for CPU resources.
- The recommended configuration is to use dedicated resources for the CML VM at the ESXi layer and control the hardware oversubscription via the **UWM > CML Server > System Configuration** pages.
- System performance should be closely monitored and the following caveats should be taken into account when running large topologies to this scale.
  - The ability to run larger simulations, approaching the node limit or the total CPU and memory capacity of the system, is truly a factor of available resources (memory, CPU, i/o speed, networking configuration, etc.). In particular, node types that are *heavier* than IOSv might or might not work depending on available memory and CPU resources.
  - Additional features (routing protocols, MPLS, ...) might impact the ability to reach the node limit by using more shared resources of the simulation environment.
  - At this time, when launching large simulations, approaching the node limit or the system memory and CPU capacity, users must stagger the launch manually (see below for instructions on performing a staggered launch). Most of the Cisco node types place a higher load on the CPU just as the node boots up and loads its configuration. Cisco node types do not always react well to CPU starvation. The system generally functions properly with modest CPU oversubscription, but running simulations close to the total hardware capacity, especially with CPU oversubscription, and starting all of the nodes at once can lead to CPU starvation. A staggered launch will help to avoid this problem.

### Staggered Launch of a Topology Simulation

When launching a large topology simulation (i.e., a topology that approaches the limits of the CML installation's hardware), it is recommended to avoid booting up every node at once when the simulation first starts. Instead, stagger the launch so that only a subset of the nodes is booting up at once. In the current release, topologies are *not* automatically staggered during launch. To perform a staggered launch of a topology simulation, the topology must indicate which nodes to start when the simulation is first launched. The back end will start the simulation, but it will only boot those nodes. The remaining nodes will remain in off / ABSENT state until they are manually started.

In CML Client, set the *Exclude from Launch* setting on nodes that you do *not* want to boot when the simulation first starts. As a starting point, pick a number of nodes equal to the number of physical cores, N, on your system. Try setting the "exclude from launch" setting on all nodes except for that "initial set" of nodes. Note that CML Client supports bulk editing: select multiple nodes at once in the topology editor, click the Properties view, and then check the edit a value to apply or remove the setting to all selected nodes. Once the *Exclude from Launch* setting has been applied to all but N nodes of the topology, the topology is ready for a staggered launch.

Start the simulation. Wait for just the initial set of N nodes to boot up and settle down. The nodes should at least go to ACTIVE - REACHABLE state, and it's probably best to leave them for a few minutes even after that to make sure that the configuration is loaded and the initial protocol processing is complete. In the running simulation view, select another batch of N nodes, right-click and select *Start Node*. Wait until that batch finishes booting up. Then start another batch of N nodes. Repeat until all of the nodes are booted up, running and ACTIVE - REACHABLE. Note that a REACHABLE state is only achieved if the initial or automatic configuration of a starting node achieves that the management IP address is present on the first interface of the node; initial configuration as generated by AutoNetkit is designed to accomplish this.

### Known Issues and Caveats

Component	JIRA	Note

<b>BBE</b>	VIRLDEV-5159	<p>The browser based editor might occasionally generate invalid CML files. This has been observed with large topologies (hundreds of nodes). In such a case, the CML server refuses to start the topology.</p> <p><b>Workaround:</b> Double check the generated links or use CML Client.</p>
	VIRLDEV-6522	<p>Topology is not loading in browser based editor when remote (HTTP) .virl file option is selected. This may be caused by the remote webserver's default configurations, which do not allow loading of content from other domains (here, the CML server's domain).</p> <p><b>Workaround:</b> Save the file locally first.</p> <p><b>Workaround:</b> Configure the web server originating the downloaded .virl file to include a header <code>Access-Control-Allow-Origin</code> with a value <code>"*"</code>.</p>
	VIRLDEV-6066	<p>The Browser-Based Editor (BBE) does not support IOS XRv 9000 nodes, nor custom node subtypes. If a topology uses an IOS XRv 9000 node, attempting to open the topology in the BBE results in a blank page. No topology is shown on the canvas.</p> <p><b>Workaround:</b> Use CML Client if you need to use IOS XRv 9000 or custom nodes.</p>
<b>Core</b>	VIRLDEV-6951	<p>In a simulation with 4 and more docker nodes, the coreos node fails to start. An error is shown in the messages console stating:</p> <p>(ERROR) [date time] Failed to start simulation " name" : Failed to create config file for node "~coreos" in simulation " name" : 'NoneType' object is unsliceable</p> <p><b>Workaround:</b> All docker nodes must start together with the first (management LXC) and second (coreos) node, in the first batch of nodes to start. This can be achieved by setting the backend's batch size to 2 + number_of_docker_nodes, e.g. 7 when at most five docker nodes need to be started. This number should not be made arbitrarily large, especially if larger VM nodes, such as IOS XRv 9000, are to be started at the same time, even in different simulations. To change the value, use the following commands:</p> <pre>sudo crudini --set /etc/virl/virl-core.ini orchestration node_op_batch_size 7</pre> <pre>sudo service virl-std restart</pre>
	VIRLDEV-5506	<p>Simulation element naming restrictions are inconsistent. Using names for nodes, networks, custom subtypes and their interfaces, images, users and projects with e.g. non-alphanumeric or accented characters, or with some symbols may lead to failures in various stages of a simulation's run.</p> <p>Neither the simulation engine nor UI prevents a launch of such topologies outright. A future version of CML may place new restrictions on the names of these elements.</p> <p>Using non-accented English letters, numbers, simple dash, dot and underscore is generally safe. Symbols like colon, percent-sign, slash and quotes should be avoided. Some platforms may place additional restrictions on file names when downloaded from UWM, such as traffic capture files.</p>

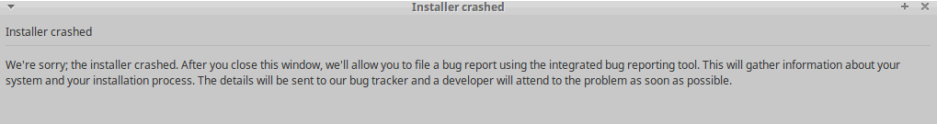
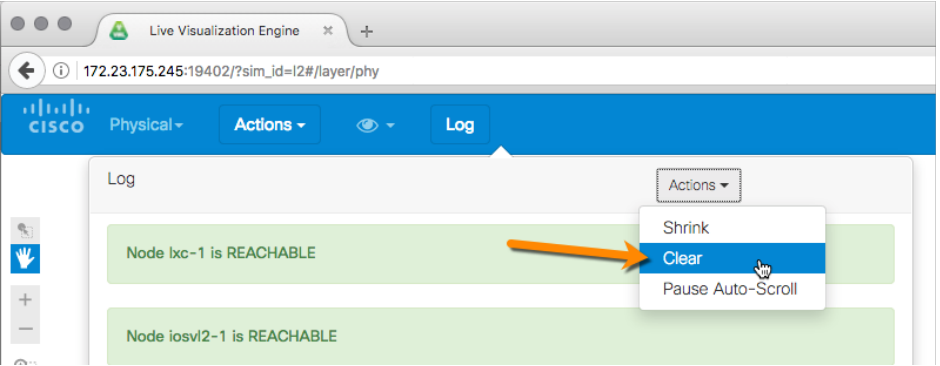
	VIRLDEV-6314	<p>Simulation nodes cannot be restarted from SHUTOFF state. A node may drop into this state when it stops executing – either by being shut down from within, by crashing its operating system or the virtualization layer, or if the host operating system reboots for any reason.</p> <p>There is no command in UWM nor CML Client to revive such a node. Individual nodes may or may not recover from abrupt stops, but the revival can at least be attempted. Only information committed to the virtual disk drive of the node can be recovered. In case the system is configured to store node virtual drives in memory, the data will not be available when the host shuts down or reboots.</p> <p><b>Workaround:</b> use the following commands on the CML server to revive all shutoff LXC nodes and OpenStack VM nodes, respectively:</p> <pre>sudo lxc-ls --stopped   xargs -rn1 sudo lxc-start -n openstack server list --status shutoff --all-projects -f value -c ID   xargs -r openstack server start</pre> <p>Individual nodes whose ID has been identified from the lists can be started by 'sudo lxc-start -n &lt;id&gt;' and 'openstack server start &lt;id&gt;', respectively. In case there are numerous nodes to revive, it is suggested to revive them in small batches.</p>
	VIRLDEV-5042	<p>Very large topologies between 100 and 300 nodes might misbehave when leaving ANK configuration generation parameters at default which produces huge topology files due to generation of a full iBGP mesh. This can manifest in:</p> <ul style="list-style-type: none"> <li>timeouts while waiting for ANK to generate the topology</li> <li>errors when displaying configuration differences in CML Client</li> <li>errors when downloading the resulting topology file in CML Client or in UWM</li> <li>runtime errors where nodes might not be coming up or put too much strain on system resources due to unrealistic configurations.</li> </ul> <p><b>Example:</b> A 300 node topology with default ANK settings will produce 300x300 = 90,000 iBGP configurations for the topology which will result in a &gt;&gt;10MB topology file.</p> <p><b>Workaround:</b> If ANK configuration generation is required, it is suggested to use valid constraints like multiple ASs, route reflectors and other means to split the simulation domain into more manageable chunks.</p> <p>Note that even if you are not using ANK to generate full configurations, it is still possible to generate IP addresses and default accounts but not full configurations by running ANK in "infrastructure only" mode. In the UI, click on the topology background, and set the "Infrastructure Only" property to true in the topology's AutoNetkit page in the Properties view before invoking ANK to Build Initial Configurations.</p>

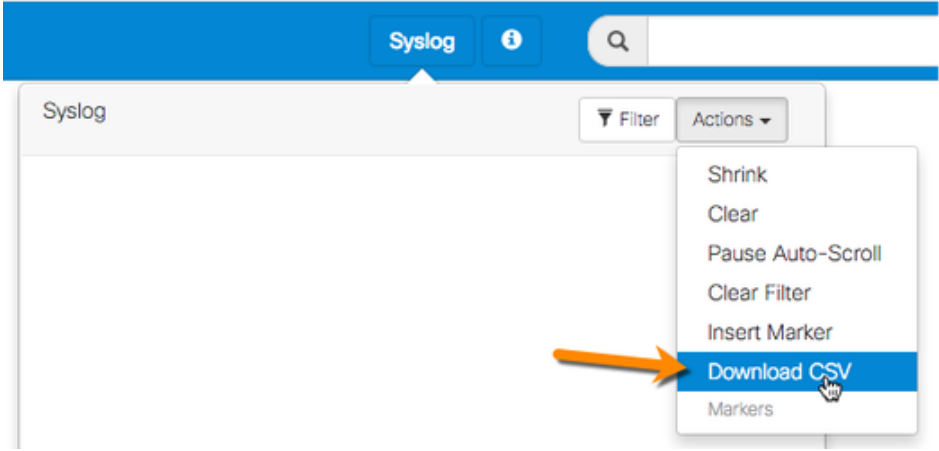
	VIRLDEV-5360	<p>When configuring 'logging console' in cases where neither the Jumphost nor the LXC Management node are available (e.g. off), configuration extraction may fail due to unexpected output as the extraction mechanism is falling back to use the console, and relying on clean and uninterrupted outputs from the nodes.</p> <p>Also, configuration extraction might fail when consoles are opened via UWM.</p> <p><b>Workaround:</b> Don't configure any logging on the console and / or don't turn off the management LXC / Jumphost. Don't have consoles open via UWM when extracting configurations.</p>
	VIRLDEV-4588	<p>Docker image names can not contain upper case letters.</p> <p><b>Workaround:</b> Use lower case image names</p>
	VIRLDEV-4710	<p>The rules governing the effective name of an LXC image are not consistent between creation, modification, and use in the simulation. The name is produced as a combination of the owning project, subtype name, and version suffix set by the user when the image is created. The use of subtype name can be overridden by the subtype definition's <code>baseline_image</code> attribute, usually to make use of a different subtype's installed images by that subtype.</p> <p><b>Workaround:</b> Do not set this property for custom LXC subtypes. It is also recommended that an LXC image, when being added, is not marked for use by a specific project, and that the <code>Modify container</code> function is not used to alter the suffix of the name.</p>
	VIRLDEV-4819	<p>Machines with high CPU count might require a different amount of database and front-end processes to be able to deal with requests.</p> <p>Symptoms of the issue can be:</p> <p>failed sim starts</p> <p>failed UWM dialogs dealing with node resources or simulations</p> <p>By default, the values are set for small machines which should be fine for CML edition. CML has the ability to adjust these settings based on built-in empirical values which are applied whenever a rehost is run.</p> <p>Alternatively, run <code>'salt-call -linfo state.sls openstack.worker_pool'</code> or use the <code>vir_l_setup</code> script, menu item 'Maintenance → 3 Reset OpenStack worker pools'.</p>
Infrastructure	VIRLDEV-6628	<p>The <code>vir_l_setup</code> tool lets an administrator perform system-wide modifications even if simulations are running. It is highly recommended to stop all simulations before running the <code>vir_l_setup</code> tool to make modifications to the system. The configuration steps may fail in unexpected ways if simulations have deployed artifacts in the system.</p>

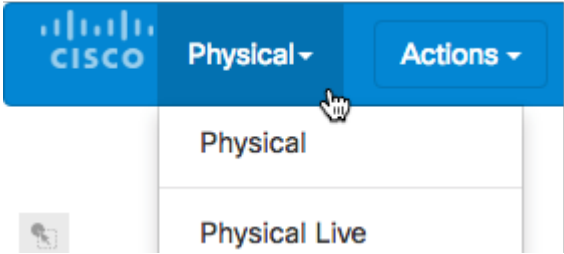


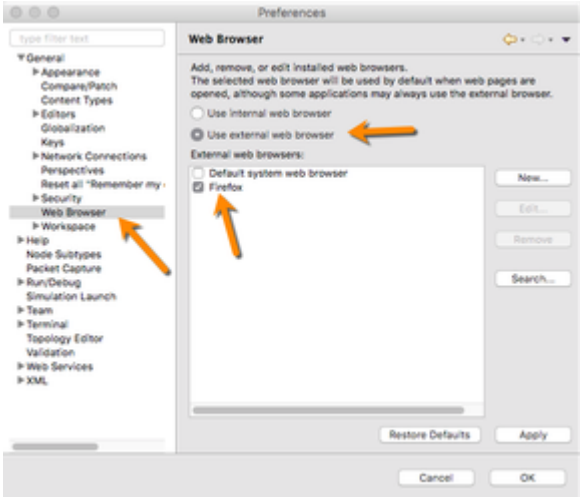
<p>VIRLDEV-5326</p>	<p>Docker by default creates SNAT / MASQUERADING iptables entries for the default docker0 bridge. This can interfere with simulation network traffic when used IPv4 networks are overlapping.</p> <p>The default 172.17.0.0/16 masquerading entry has been removed. However, there is at least one entry left for the Docker registry with a /32 address which can not be used in any simulation.</p> <p><b>Workaround:</b> Don't use the 172.17.0.0/16 network whenever possible. Check the masqueraded IP addresses currently in use by the local registry by typing <code>sudo iptables -L -v -t nat</code></p> <p><b>Example (excerpt):</b></p> <pre> 0      0 MASQUERADE  tcp  -- any    any      172.17.0.2          172.17.0.2          tcp dpt:5000 </pre>
<p>VIRLDEV-6243</p>	<p>Live snapshot does not create modified image. A running simulation node's current contents of its hard drive can be captured to form a new image. In case the snapshot is taken on a running node, the modifications to the original image as the node launched with need not be captured correctly.</p> <p>Note that this functionality is mainly geared towards custom server images, less so for capturing router VM state.</p> <p><b>Workaround:</b> Create the snapshot from a shutdown (not stopped) node. The shutdown should occur from within the node, i.e. via logging into its console, and running 'sudo shutdown -h now' command.</p> <p>See notes on VIRLDEV-6314 on how a shutoff node can be restarted.</p> <p><b>Workaround:</b> Enable and install the qemu-guest-agent package prior to snapshotting.</p> <p>Before the node that shall be built is started in a simulation, edit the base image, to add 'hw_qemu_guest_agent=yes' in its property list. During image preparation, install qemu-guest-agent into the VM node.</p>
<p>VIRLDEV-5365</p>	<p>When the local time of the host computer that runs the VM is in a timezone &gt;0 (e.g. east of Greenwich), NTP might step the clock back at system start which might confuse STD and therefore results in a licensing issue due to invalid time.</p> <p><b>Workaround:</b> Restart STD/UWM using <code>sudo salt-call -l info state.sls vir1.restart</code></p>

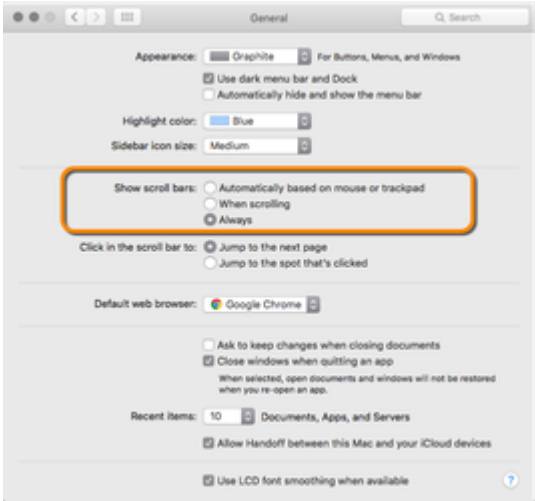
	VIRLDEV-5387	<p>NTP does not sync under certain circumstances.</p> <p><b>Workaround:</b> Restart NTPd using <code>sudo systemctl restart ntp</code>, check with <code>ntpq -p</code>.</p> <p>Sometimes, it was sometimes necessary to kill the ntpd process manually:</p> <pre>sudo systemctl stop ntp sudo killall ntpd sudo systemctl start ntp</pre>
	VIRLDEV-6432	<p>Launching an OpenVPN client in Linux for IPv6-only shared network does not bring up tunnel interface.</p> <p>This does not affect Windows nor Mac OS X clients.</p> <p><b>Workaround:</b> once the tunnel connection is established, run <code>sudo ip link set tap0 up</code>.</p>
	VIRLDEV-6078	<p>The APT cache is not ready on fresh install. The CML build process clears the database used by Ubuntu to recognize installable packages. An attempt to install a new package will result in an error stating the package is not available. Resolve by running '<code>sudo apt-get update</code>' prior to package installation.</p>
	VIRLDEV-6524 VIRLDEV-6542	<p>IPv6 literals are unsupported as sources for image downloads. The underlying tools used for the actual downloads of VM/LXC packages (curl) and docker packages (docker) cannot use an IPv6 address as the hostname part in URLs.</p> <p><b>Workaround:</b> Use a valid DNS name for the image source host. As a last resort, edit the CML server's /etc/hosts file to add a custom hostname association for the IP address.</p>
	VIRLDEV-6821	<p>Baremetal installs create a swap partition of same size as available memory.</p> <p>In cases where the physical disk is comparatively small to the amount of RAM on the server, the main partition may become too small to operate larger simulations.</p> <p><b>Workaround:</b> In the installer, select manual partitioning instead of automatic; contact support if assistance is required.</p>

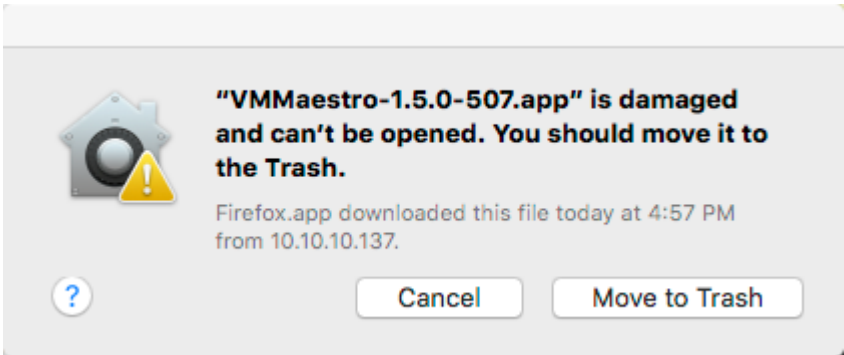
	<p>VIRLDEV-6732</p>	<p>Failed baremetal installation may offer to send an automatic bug report.</p> <p>In some cases, if the baremetal installer fails, it may offer to collect troubleshooting information, and upload it to a bug tracker for developer review.</p>  <p>This is not an offer made by Cisco, the tool would not upload the information to any Cisco infrastructure, and it is very unlikely to result in actual support being provided.</p> <p>Use the official and appropriate support channels for this product instead.</p>
	<p>VIRLDEV-5085</p>	<p>In LiveVis, when selecting the 'Clear Log' option a pop-up message might include a "Prevent this page from opening additional dialogs". If the user selects that option may prevent the system from collecting further log messages. This is a known issue and it is also browser dependent.</p>  <p><b>Workaround:</b> Do not select this option when offered by the browser.</p>

<p><b>Live Vis</b></p>	<p>VIRLDEV-4899</p>	<p>Downloading the Syslog as CSV from within Live Vis is not working on Safari and results in a page error. This is a known limitation with the Safari browser.</p>  <p><b>Workaround:</b> Use a different browser like Firefox or Chrome.</p>
	<p>VIRLDEV-4898</p>	<p>Extracting configurations from a running topology within Live Visualization is not working as expected when using Safari. The document returned is shown as XML text, rendered in the browser and is not offered to 'save' into the Downloads folder. This is a known limitation.</p> <p><b>Workaround:</b> Use a different browser like Firefox or Chrome or save the resulting XML text manually into a .virl file.</p>
	<p>VIRLDEV-4698</p>	<p>In Live Visualization, NX-OSv interfaces are listed twice in the interface table, one time showing IP, other showing None. When retrieving the interface table from an NX-OSv device, some entries show twice in the output. This is only cosmetic and might cause a spurious extra line being drawn.</p>
	<p>VIRLDEV-4223</p>	<p>Live Visualization might not work properly when changing the UWM port. UWM System configuration offers the ability to change it's listening port from 19400 to another port. When doing so, this might have side effects in Live Visualization, specifically with creating packet captures, and opening serial console connections, which are performed using UWM.</p> <p><b>Workaround:</b> Do not change the UWM port.</p>

	<p>VIRLDEV-4202</p>	<p>The overlay menu button to switch among different overlays might get blocked after updating Live Vis Web server port. In UWM, it is possible to change the Live Vis Web server port from 19402 to a different port. This might result in a blocked overlay menu button (not showing any entries). The image below illustrates the correct menu behavior whereas when broken, the entire drop down part of the menu is missing.</p>  <p><b>Workaround:</b> None</p>
	<p>VIRLDEV-4695</p>	<p>In Live Visualization, the 'Physical Live' Overlay does not correctly show physical links for XRv nodes. This is a known defect.</p> <p><b>Workaround:</b> None</p>
	<p>VIRLDEV-4949</p>	<p>Sometimes, BGP VPNv4 links are not being drawn in Live Visualization.</p> <p><b>Workaround:</b> None</p>
	<p>VIRLDEV-4342</p>	<p>Large (80-300+) topologies might not render properly and perform slower than expected in Live Vis. <b>Workaround:</b> None</p>
	<p>VIRLDEV-5061</p>	<p>Preview of a topology before launching the simulation does not render properly with certain browsers. This happens with Windows and Internet Explorer. IE is not a supported browser.</p> <p><b>Workaround:</b> Switch to Chrome or Firefox to prevent the issue from showing.</p>
<p><b>UWM</b></p>	<p>VIRLDEV-2167</p>	<p>When importing projects into the system, the uwmadmin password in vir.ini was out of sync with the database. To avoid this issue, the uwmadmin project is skipped when importing projects into the system.</p> <p><b>Workaround:</b> Manually change the password after importing to get the correct password into the system.</p>
	<p>VIRLDEV-6613</p>	<p>UWM does not let one specify a launched management network as a shared network other than 'flat'.</p> <p><b>Workaround:</b> see workaround for VIRLDEV-6614 below</p>
	<p>VIRLDEV-6592</p>	<p>The UWM admin-mode button to 'Stop all simulations' does not ask for confirmation. Do not press this button accidentally.</p>

	<p>VIRLDEV-6838</p>	<p>In a cluster, System configuration changes to the DNS server settings are not propagated to compute hosts.</p> <p><b>Workaround:</b> After completing the system configuration change (which applies it to the controller), log into the controller and run the CLI command:</p> <pre>sudo salt "compute*" state.sls common.cluster.compute.config.primary</pre>
	<p>VIRLDEV-6614</p>	<p>CML Client does not permit setting a different shared management network. Only the original 'flat' network may be selected, even if 'flat1' or other shared networks are available.</p> <p><b>Workaround:</b> Preferably use the 'flat' shared management network. To switch to another shared management network, go to the 'Extensions' tab of the topology properties (shown when nothing is selected in the topology). Edit the 'management_network' extension to use the value 'flat1' or 'shared-X' where X is the name given to the shared network in UWM System Configuration.</p>
<p>CML Client</p>	<p>VIRLDEV-4434</p>	<p>In CML Client, use external browser for both ANK vis and Live Vis. We've had reports about using the internal browser for ANK and Live Visualization from within CML Client.</p>  <p><b>Workaround:</b> Use an external browser, configure as shown above.</p>
	<p>./.</p>	<p>Terminal preference for detached internal terminals - this function has been deprecated in CML Client 1.2.4 onwards.</p> <p><b>Workaround:</b> You can manual 'tear' the terminal pane from the main CML Client window. Use this in conjunction with the CML Client preference (Cisco terminal) - " multiple tabs for one simulation" .</p>

<p>VIRLDEV-3525</p>	<p>In CML Client (1.2.8 build 474), the scroll bar in the 'Preferences &gt; Node Subtypes' dialog doesn't work properly on OS X 10.11 and newer.</p>  <p><b>Workaround:</b> Configure scroll bars to 'always show' in the General section of the System preferences as shown above.</p>
<p>VIRLDEV-6359</p>	<p>Telnet/ssh to management does not work in cluster if jumphost node is running on a compute host. If a simulation does not enable the management LXC, an alternate 'jumphost' node is used to proxy traffic into the simulation's management network. In a cluster environment, the jumphost may be launched on a compute node. CML Client falsely assumes the jumphost is running on the controller host, and thus connections to it fail.</p> <p><b>Workaround:</b> Do not disable the management LXC for simulations; use of such nodes is strongly preferred to the jumphost.</p>
<p>VIRLDEV-4820</p>	<p>Occasionally, the message 'Unexpected end of input within/between OBJECT entries' can be observed in CML Client's <i>Simulations</i> panel. This is mostly cosmetic as it recovers automatically.</p> <p><b>Workaround:</b> None</p>
<p>VIRLDEV-5042</p>	<p>While running ANK, when clicking 'yes' on "View configuration changes dialog" the dialog disappears resulting in 'Unable to connect to Vis Server' error in console. This happens for large topologies.</p> <p><b>Workaround:</b> None</p>

	VIRLDEV-6288	<p>Sometimes, installing a new version of CML Client on an OS X system that already has CML Client installed leads OS X to report that the application is damaged.</p>  <p><b>Workaround:</b></p> <p>1) The most likely cause of that error on OS X is a combination of OS X's default security settings, lack of signing of the OS X version, and some terrible error reporting from OS X. Try this:</p> <p>Open the OS X System Preferences</p> <p>Click Security &amp; Privacy</p> <p>Click General.</p> <p>If necessary, click the little lock on the bottom to unlock the page.</p> <p>Switch the option to say "Allow apps downloaded from" &gt; Anywhere.</p> <p>Close the Security &amp; Privacy dialog</p> <p>Now try to launch CML Client.</p> <p>If that works, you can now go back and change your Security &amp; Privacy setting back to their defaults.</p> <p>Unfortunately, since Sierra, that Security &amp; Privacy dialog no longer shows in the security preferences. In that case, your options are:</p> <p>Remove the "quarantine" attribute that may be causing the symptoms you're seeing:</p> <p>Open the Terminal.app.</p> <p>Remove the quarantine attribute:</p> <pre>sudo xattr -d -r com.apple.quarantine /Applications/CML-1.5.0-510/</pre> <p>Re-enable the install from "Anywhere" option for Sierra.</p> <p>See <a href="http://osxdaily.com/2016/09/27/allow-apps-from-anywhere-macos-gatekeeper/">http://osxdaily.com/2016/09/27/allow-apps-from-anywhere-macos-gatekeeper/</a></p> <p>Then follow the instructions above to run CML Client.</p>
--	--------------	--



	VIRLDEV-6755	<p>On OS X, webservices cannot be reached over IPv6 in many configurations.</p> <p>A likely issue with Java IPv6 support causes the UI application to fail contacting the webservices over IPv6, with a "No route to host" error being reported. This happens even if other applications running on the same device are able to properly route to the backend server. This issue may be affected by the presence and configuration of wired/wireless interfaces on the device.</p> <p><b>Workaround:</b> Use IPv4 when communicating with the webservices via the UI application.</p>
	VIRLDEV-3616	<p>When running ASAv nodes using the bundled ASAv image, the console of the ASAv is bound to the serial port. By default, 'vanilla' ASAv images downloaded from CCO are having their console bound to the 'VGA' screen which is accessible in CML using the VNC option. However, access to the console is <b>EITHER</b> on the serial port <b>OR</b> on the VGA screen, never both. Since it is not known where the console is bound to (serial or VGA), both options are offered to the user but only one option will succeed.</p>
VMs	VIRLDEV-5092	<p>NX-OSv / NX-OSv 9000 nodes remain UNREACHABLE and connect to monitor port doesn't connect.</p> <p>NX-OSv (Titanium) nodes might not be available on the management interface. This is a reference platform issue where sometimes the Mgmt0 interface is stuck in 'down/up' state. E.g. the interface is 'admin up' but the link is indicated as 'down'.</p> <p><b>Workaround:</b> Manually issue a 'shut / no shut' sequence on the management interface of the affected node</p> <p>NX-OSv 9000 nodes create unnecessary broadcast traffic on the management interface. This is a reference platform issue where NX-OSv 9000 nodes respond to frames not owned by the node which might result in a broadcast storm of IP redirect packets on the management network.</p> <p><b>Workaround:</b> Configure 'no ip directed-broadcast' on the management interface of NX-OSv 9000 nodes.</p> <p>NX-OSv 9000 nodes occasionally (less than 5 in 100 launches) do not become reachable even with the aforementioned workaround. A restart of the affected node usually resolves this.</p>
	VIRLDEV-4682	<p>The subtype for Ostinato has changed between the 1.2 and 1.3 release. When migrating to CML 1.3 from 1.2 and thus staying on the OpenStack Kilo release, the Ostinato 0.8.1 image will not be used due to the mismatch in the subtype (lxc-ostinato vs. lxc-ostinato-drone).</p> <p><b>Workaround:</b> Change subtype for affected nodes to 'lxc-ostinato-drone'</p>

	VIRLDEV-4955	<p>IOS XRv 9000 duplicate management IP configuration. IOS XRv 9000 nodes acquire the same IP for both the XR operating system layer and its associated management interface as well as for an underlying Linux layer which uses the same 'physical' interface.</p> <p><b>Workaround:</b> None</p>
	./.	<p>IOSv 15.6(2)T - On boot-up the following (similar) message may be observed:%SYS-3-CPUHOG: Task is running for (1997)msecs, more than (2000)msecs (0/0),process = TTY Background.-Traceback= 114ECF8z 130425z 15E20Ez 15DF30z 15DD3Dz 157D75z 158A2Bz 1589BFz 159B67z 153672z 3C9740Az 3C868CEz 3C89BEFz 5125F91z 491D86Cz 492E540z - Process " Crypto CA" , CPU hog, PC 0x00157D2C</p> <p><b>Workaround:</b> This is cosmetic and can be ignored.</p>
	VIRLDEV-6297	<p>IOS XRv cannot use more than 16 interfaces. As a regression from CML 1.2, IOS XRv fails to pass any traffic shortly after booting whenever more than 16 data interfaces are present on the node. This is currently attributed to non-specific changes in the qemu/kvm software.</p> <p>The subtype definition in CML 1.6 was adjusted to only allow up to 16 data interfaces.</p> <p><b>Workaround:</b> None. Setting up an alternate qemu 2.0 installation alongside the default 2.5 may allow the nodes to run as before.</p>
	./.	<p>IOS XRv 9000 release 6.5.1 has a recommended memory requirement of 12 GB; the minimum observed memory required to boot and run the system is 8 to 10GB. The VM will actually consume this amount of memory. This means that memory reservation, which employs simple memory overcommit calculation, is at increased risk of allowing memory exhaustion to occur on the simulation hosts.</p> <p>When IOS XRv 9000 is starting, it writes a large amount of data (at least 3.5GB, up to 7GB with release 6.5.1) into its virtual hard drive. This amount of data is stored separately for each instance of an IOS XRv 9000 on a given host, unlike the base image data which is shared. The largest among the other node subtypes exhibiting similar behavior is NX-OSv 9000, writing about 250 MB on startup.</p> <p>In case several such nodes are started on the same host, the amount of data can potentially add up. Make sure the available space on the hosts' hard drive is plentiful. If the system has ramdisk-based storage of nodes' virtual disks enabled, then this space allocation counts towards memory usage, but is not accounted for in memory reservation calculations, potentially also leading to memory exhaustion.</p>

Known Issues and Caveats

	./.	<p>When IOS XRv 9000 is booting, if you enter text at the console, the boot will drop to a CLlsetup setup prompt for the admin user account. This prompt prevents the pre- blocks CVAC, causing initial configuration application to be abandoned.</p> <p><b>Workaround:</b> Do not open a console until the node becomes reachable, or don't enter any text into the console (not even enter to check if any output will be generated), or for releases of IOS XRv 9000 up to 6.4.1, enter a valid pair of credentials (which are named neither cisco nor admin) as soon as the prompt appears. The latter workaround may be ineffective in newer IOS XRv 9000 releases.</p> <p>In case the initial configuration is not loaded after all, it may still be loaded manually. After setting up the admin account, log in using this new account. Then, enter command: run cp /etc/sysconfig/iosxr_config.txt cvac/ If that command succeeds, go into config mode, and enter: load cvac/iosxr_config.txt If the configuration load succeeds, enter commit and then end to exit config more.</p>
	VIRLDEV-6074	<p>CML has provided support for a reference platform image based on the Nexus 9k code since December 2016. This subtype for this image is NX-OSv 9000. Cisco has changed the name of this image to Nexus 9000v. CML has no Nexus 9000v subtype.</p> <p><b>Workaround:</b> None. Just use the NX-OSv 9000 for Nexus 9000v images.</p>

	VIRLDEV-6416	<p>Default NX-OSv 9000 memory allocation may not sufficient. When launching a topology simulation that uses multiple NX-OSv 9000 nodes, some of them may fail to boot.</p> <p>The "flavor" used by default for NX-OSv 9000 was changed to use 6 GB of RAM with the release of NX-OSv 9000 version 7.0.3.I6.3, because the previously lowered memory allocation of 4GB has proved to be too small in practice. We continue to work with the Nexus team to determine optimal default resource settings for NX-OSv 9000 and the other router VM images.</p> <p>Increasing the number of allocated interfaces on the NX-OSv 9000 nodes (e.g. above 10 interfaces) will further increase the memory requirements in an unspecified manner. A satisfactory allocation may also depend on the task load on the node, and thus require some level of experimentation for an individual topology in larger configurations.</p> <p><b>Workaround:</b> If your NX-OSv 9000 nodes fail to start properly, or log errors indicating memory issues, we recommend replacing the NX-OSv 9000 flavor with a new flavor that has a larger memory allocation.</p> <p>Login to the UWM as the uwadmin, and navigate to <b>Node resources &gt; Flavors</b>.</p> <p>Click the <b>Delete</b> icon in the <b>Options</b> column for the NX-OSv 9000 flavor to delete the default flavor.</p> <p>Click the <b>Add</b> button to add a new flavor.</p> <p>On the Create Flavor form, enter the values</p> <p>Name: NX-OSv 9000</p> <p>RAM (MB): 6144 (and larger)</p> <p>Virtual CPUs: 2</p> <p>Disk (GB): 0</p> <p>Click the <b>Create</b> button.</p> <p>Simulations that you launch after making this change will use the new flavor and the (minimum) 6 GB memory allocation for all NX-OSv 9000 nodes.</p>
--	--------------	---