CHARON-SSP for Linux - User's Guide

Contents

About This Guide

This preface describes the conventions and organisation of this user's guide. It also describes its intended audience and how to obtain further copies of this guide, related documentation and further support.

Intended Audience

This user guide is targeted at anyone attempting to install, configure or manage the CHARON-SSP line of SPARC system virtual machines. While the content of this manual is targeted at general users (not just system managers and administrators) a general, working knowledge of the host platform and its conventions is expected.

This user guide covers the following Stromasys products:

- CHARON-SSP/4M for Linux,
- CHARON-SSP/4U for Linux,
- CHARON-SSP Virtual Environment,
- CHARON-SSP Manager for Linux,
- CHARON-SSP Manager for Windows, and
- CHARON-SSP Agent for Linux.

Document Structure

This user guide is organised into the following chapters:

- Introduction – Provides an overview of the virtual machines and the devices and guest systems they support.
- Installation – Details the installation and removal of the CHARON-SSP virtual machine software and associated utilities.
- Using the CHARON-SSP Software – Describes how to use the CHARON-SSP software.
- License Management – Describes how to operate the Sentinel HASP license utilities.
- Configuration Reference – Provides a complete reference for the CHARON-SSP configuration file format.
- Console Reference – Provides a complete reference for the CHARON-SSP OpenBoot console environment.
- Command Line Utilities Reference – Provides a comprehensive reference of the supporting command line utilities.

Obtaining Documentation

The latest version of this manual and other, related manuals and documentation can be found at the Stromasys Support website, here:

http://www.stromasys.com/services/documentation/

Complete Software and Product Descriptions (SPD) and a copy of the End User License Agreement (EULA) can also be obtained here:

http://www.stromasys.com/products/

Obtaining Technical Assistance

The CHARON-SSP family of SPARC virtual machines can be covered by a number of different support options. The support options themselves are listed in the section Emulator Licensing However, for further, more complete information about which option is best for you and purchasing, contact your regional sales team using one of the methods below:

Region	Phone	Address	Local Open Hours
Australasia-Pacific apac.sales@stromasys.com	+852 2853 1600	28/F Room D, Tower B, Billion Centre 1 Wang Kwong Road, Kowloon Bay Hong Kong, S.A.R. China	8 a.m. - 5 p.m.
Americas ams.sales@stromasys.com	+1 919 239 8450	2840 Plaza Place, Ste 450 Raleigh, NC 27612 U.S.A.	8 a.m. - 5 p.m.

Europe, Middle-East and Africa emea.sales@stromasys.com	+41 22 794 1070	Avenue Louis-Casai 84 5th Floor 1216 Cointrin Switzerland	8 a.m. - 5 p.m.

Alternatively, the support centre can be contacted by email at support@stromasys.com.

If you have purchased CHARON-SSP via a Value Added Reseller (VAR) then please contact them directly.

Conventions

Throughout the document(s) these conventions are followed:

Notation	Description
`$`	The dollar sign in interactive examples indicates an operating system prompt for VMS. The dollar sign can also indicate non superuser prompt for UNIX / Linux.
`#`	The number sign represents the superuser prompt for UNIX / Linux.
`>`	The right angle bracket in interactive examples indicates an operating system prompt for Windows command (cmd.exe).
`User input`	Bold monospace type in interactive examples indicates typed user input.
<**`path`**>	Bold monospace type enclosed by angle brackets indicates command parameters and parameter values.
`Output`	Monospace type in interactive examples, indicates command response output.
`[]`	In syntax definitions, brackets indicate items that are optional.
...	In syntax definitions, a horizontal ellipsis indicates that the preceding item can be repeated one or more times.
`dsk0`	Italic monospace type, in interactive examples, indicates typed context dependent user input.
	This symbol represents the Enter key without typed user input. Used, for example, to tell the user to select the default value by pressing enter.

The following definitions apply:

Term	Description
Host	The system on which the emulator runs, also called the charon server
Guest	The emulated system, in which the Tru64 or VMS system runs

The following product naming conventions will be followed:

The core SPARC virtual machines are available in the following formats:

- CHARON-SSP/4M – 32-bit SPARC V8, sun4m architecture
- CHARON-SSP/4U – 64-bit SPARC V9 sun4u architecture

However, for the most part these products both support the same configuration mechanisms, systems console and interfaces. For simplicity, when describeing options, interfaces, etc. that apply to both products, they will be collectively referred to as CHARON-SSP. Only when describing platforms-specific features will the full product name be used.

Introduction

In 1987 Sun Microsystems released the SPARC V7 processor. A 32-bit RISC processor featuring the unique Berkeley RISC designed register-window. This was followed soon after in 1990 with the SPARC V8. This was a revision of the original SPARC V7, with the most notable inclusion of hardware divide and multiply instructions. The SPARC V8 processors were the basis for a number of Sun Microsystems servers and workstations such as the SAPRCstation 5, 10 and 20.

In 1993 the SPARV V8 was followed by the 64-bit SPARC V9 processor. This too became the basis for a number of Sun Microsystem servers and workstations, such as the Enterprise 250 and 450.

Through the development of newer products and the sale of Sun Microsystems to Oracle, software and systems developed for these older model SPARC-based workstations and servers has become harder to maintain. To fill the continued need for certain, end-of-life SPARC-based systems Stromasys S.A. has developed the CHARON-SSP line of virtual machine products. These products are a software-based, virtual machine replacement for the following native-hardware SPARC systems:

CHARON-SSP/4U supports the following virtual systems:

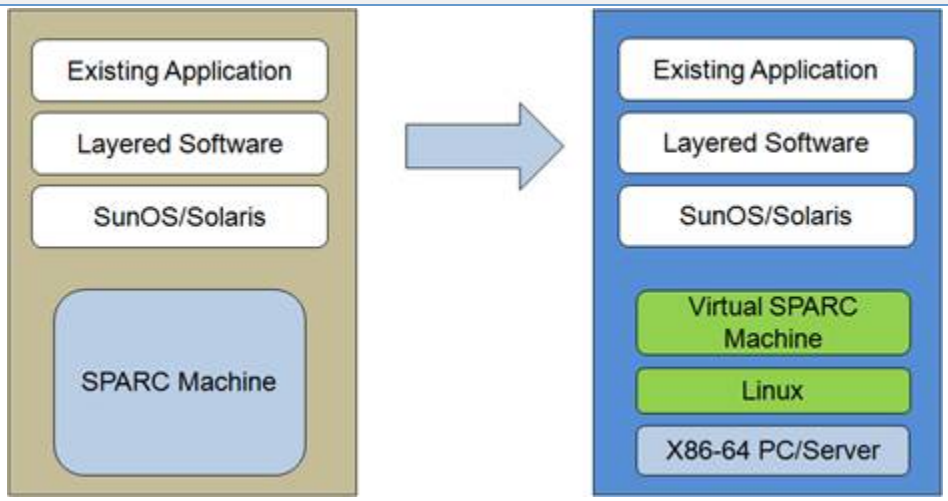
- Sun SPARCstation 20

CHARON-SSP/4M supports the following virtual systems:

- Sun Enterprise 450

The CHARON-SSP virtual machines allow users of Sun and Oracle SPARC-based computers to replace their native hardware counterparts, in such a way that there is little or no change required to the original system configuration. This means you can continue to run your applications and data without the need to switch or port to another platform. The CHARON-SSP virtual machines run on commodity, Intel 64 and AMD64 based systems ensuring your continued investment is protected.

Seamless Migration from Native to Virtual SPARC



The CHARON-SSP/4U virtual machines support the following guest operating system releases:

- SunOS 4.1.3 - 4.1.4
- Solaris 2.5 - 8

The CHARON-SSP/4M virtual machines support the following guest operating system releases:

- Solaris 2.5.1 - 10

> The virtual system must still be supported by the operating system release, i.e. it is not possible to install SunOS 4.1.3 on a virtual Enterprise 450 as hardware support for the Enterprise 450 was not included until Solaris 2.5.1.

Host System Requirements

To ensure maximum performance when emulating an SPARC-based system it is important to follow some guidelines regarding the set up of the host system. These are described in the following sections.

Hardware Requirements

To run the CHARON-SSP virtual machine products the host system must have at least two CPUs (or cores) and must be either an Intel 64 or AMD64 based processor.

When determining the best hardware configuration it is generally best to follow these three basic rules:

1. Use a processor with a speed of at least 2.8GHz.
2. Allow for 1 CPU (or core) plus the number of emulated CPUs.
3. Allow for a minimum of 2GB of RAM plus the total RAM of the emulated system.

There are other options available to achieve higher performance, including high-speed I/O controllers, but these basic rules are an excellent starting point to get a system up and running.

> Stromasys recommends running the CHARON-SSP virtual machine software on Oracle x86 Servers running Oracle Linux.

Operating System

The CHARON-SSP virtual machine products run on Linux-based systems. The following Linux distributions and releases are supported by Stromasys as host environments:

- Oracle Linux 6 - 7
- Fedora 17 - 20
- Red Hat Enterprise Linux 6 - 7
- CentOS 6 - 7

Supported Virtual Hardware

The different classes of CHARON-SSP virtual machine support a number of different hardware devices. The table below describes the device features, and where relevant, the quantity supported by different CHARON-SSP virtual machines.

CHARON-SSP Supported Virtual Hardware

	CHARON-SSP/4M	CHARON-SSP/4U
SPARC V8 (32-bit)	Y	
SPARC V9 (64-bit)		Y
No. of CPUs	4	24
Max. RAM	512MB	32GB
Ethernet Controllers	4	4
SCSI Controllers	1	2
SCSI Devices	7	30
Serial Ports	2	2

Installation

The CHARON-SSP for Linux suite of products provides much more than just a SPARC virtual machine. The software is available in four parts:

- CHARON-SSP/4M – 32-bit SPARC V8, sun4m architecture
- CHARON-SSP/4U – 64-bit SPARC V9 sun4u architecture
- CHARON-SSP Manager – A remote GUI virtual machine manager, and
- CHARON-SSP Agent – Bridge for communication between CHARON-SSP virtual machine and CHARON-SSP Manager.

All of these individual packages are available as a combined, virtual environment appliance:

- CHARON-SSP Virtual Environment – Combined SPARC Virtual Machine Appliance

The following sections describe the installation of the Sentinel HASP Licensing software, required by CHARON-SSP for Linux:

- Installing the Sentinel HASP Software

The following sections describe the installation, upgrading and, with exception of CHARON-SSP Virtual Environment, the removal of these software packages.

- Installing the CHARON-SSP Software – Installing, removing and upgrading CHARON-SSP/4M and CHARON-SSP
- Installing the CHARON-SSP Manager
- Installing the CHARON-SSP Agent
- Installing the CHARON-SSP Virtual Environment

Installing the Sentinel HASP Software

The Sentinel HASP runtime is a necessary software component used to license the CHARON-SSP for Linux virtual machine software. Without this package it is not possible to run the CHARON-SSP virtual machine software. The RPM package is included as part of the tar, gzip CHARON-SSP distribution.

The CHARON-SSP for Linux software comes in two separate install packages. The table below describes these:

CHARON-SSP for GNU/Linux Installation Packages

Product Name	Description	Distribution Package Name
CHARON-SSP/4M	32-bit SPARC V8 virtual machine based on the sun4m architecture.	`charon-ssp-4m-1.0.21-x86_64_rpm.tar.gz`
CHARON-SSP/4U	64-bit SPARC V9 virtual machine based on the sun4u architecture.	`charon-ssp-4u-1.0.21-x86_64_rpm.tar.gz`

To begin, unpack the distribution archive, like so (the example below shows CHARON-SSP/4M for Linux):

Unpack CHARON-SSP/4M for GNU/Linux Distribution

```
# cd /tmp
# tar xzvf /path/to/charon-ssp-4m-1.0.21-x86_64_rpm.tar.gz
```

All installation steps are required to be performed from a privileged account as denoted by the `'#'` prompt.

Prerequisites

Before installing the Sentinel HASP Runtime software, there are some prerequisite software packages to install. The following sections describe these packages

GNU C Library (i686) Installation

Before installing the Sentinel HASP runtime it is necessary to install the 32-bit compatibility version of the GNU C Library (glibc). To install this software on Red Hat Enterprise Linux (RHEL) based systems use the following commands:

Command to Install glibc.i686

```
# yum install glibc.i686
```

The command above should generate output very similar to the following:

i686 glibc Installation Output

```
Loaded plugins: langpacks
Resolving Dependencies
--> Running transaction check
---> Package glibc.i686 0:2.17-78.0.1.el7 will be installed
--> Processing Dependency: libfreebl3.so for package: glibc-2.17-78.0.1.el7.i686
--> Processing Dependency: libfreebl3.so(NSSRAWHASH_3.12.3) for package: glibc-2.17-78.0.1.el7.i68
--> Running transaction check
---> Package nss-softokn-freebl.i686 0:3.16.2.3-9.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package                 Arch      Version                 Repository       Size
================================================================================
Installing:
 glibc                   i686      2.17-78.0.1.el7         ol7_latest      4.2 M
Installing for dependencies:
 nss-softokn-freebl      i686      3.16.2.3-9.el7          ol7_latest      186 k

Transaction Summary
================================================================================
Install  1 Package (+1 Dependent package)

Total download size: 4.3 M
Installed size: 15 M
Is this ok [y/d/N]: y
Downloading packages:
(1/2): nss-softokn-freebl-3.16.2.3-9.el7.i686.rpm          | 186 kB   00:04
(2/2): glibc-2.17-78.0.1.el7.i686.rpm                      | 4.2 MB   00:14
--------------------------------------------------------------------------------
Total                                           299 kB/s | 4.3 MB  00:14
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : nss-softokn-freebl-3.16.2.3-9.el7.i686                       1/2
  Installing : glibc-2.17-78.0.1.el7.i686                                   2/2
  Verifying  : glibc-2.17-78.0.1.el7.i686                                   1/2
  Verifying  : nss-softokn-freebl-3.16.2.3-9.el7.i686                       2/2

Installed:
  glibc.i686 0:2.17-78.0.1.el7

Dependency Installed:
  nss-softokn-freebl.i686 0:3.16.2.3-9.el7

Complete!
```

Installing the Sentinel HASP Runtime

To install the Sentinal HASP runtime, use the following commands:

Sentinel HASP RPM Install Command

```
# yum localinstall aksusbd-2.2.1-1.x86_64.rpm
```

The installation process should generate output similar to the following:

Sentinel HASP Installation Output

```
Loaded plugins: langpacks
Examining aksusbd-2.2.1-1.x86_64.rpm: aksusbd-2.2.1-1.x86_64
Marking aksusbd-2.2.1-1.x86_64.rpm to be installed
Resolving Dependencies
--> Running transaction check
---> Package aksusbd.x86_64 0:2.2.1-1 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
Package       Arch         Version         Repository                     Size
================================================================================
Installing:
 aksusbd       x86_64       2.2.1-1         /aksusbd-2.2.1-1.x86_64       4.5 M

Transaction Summary
================================================================================
Install  1 Package

Total size: 4.5 M
Installed size: 4.5 M
Is this ok [y/d/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
 Installing : aksusbd-2.2.1-1.x86_64                   1/1
Starting aksusbd (via systemctl):  [  OK  ]
 Verifying  : aksusbd-2.2.1-1.x86_64                   1/1

Installed:
  aksusbd.x86_64 0:2.2.1-1

Complete!
```

Now that the Sentinel HASP software has been installed and it running correctly, it is possible to continue on to Installing the CHARON-SSP Software.

Post-Installation Tasks

The following post installations task are not required. However, for improved security they are highly recommended.

Securing the Sentinel HASP Configuration Files

The configuration file used by the Sentinel HASP software is insecure and open to malicious, unprivileged access and modification unless the following commands are executed:

Commands to Secure Sentinel HASP Configuration File

```
# chmod 0700 /etc/hasplm
# chmod 0600 /etc/hasplm/*
```

Securing the Sentinel HASP GUI

The web-based GUI interface is also open to malicious access unless secured appropriately. Use the following steps to secure the software:

Instructions for Securing Sentinel HASP GUI

Step	Description
1.	Open a web browser and navigate the **http://localhost:1947.** For CHARON-SSP Virtual Environment users, **double-click** the **Web Browser** icon under the **System** sections of the **System Utilities** window.
2.	**Click** on the left-hand menu item labelled **Configuration**.
3.	**Click** the **Basic Settings** tab.
4.	Under the entry labelled **Password Protection, click** the button labelled **Change Password**.
5.	At the **Change Password** window: • Leave the field labelled **Current Admin Password** blank (there is no password set by default). • Enter the intended password into the field labelled **New Admin Password**. • Repeat the intended password in the field labelled **Re-enter new Admin Password.** • **Click** the button labelled **Submit.**
5.	Back at the **Basic Settings** tab: • Under the section labelled **Password Protection,** select the radio button labelled **All ACC Pages.** • **Click** the button labelled **Submit** to save this change.
6.	To allow remote access to the Sentinel HASP GUI: • **Click** the **Basic Settings** tab. • **Select** the **Allow Remote Access to ACC** check box. • **Click** the button labelled **Submit.**

Removing the Sentinel HASP Software

To remove the Sentinel HASP software, execute the following command:

Sentinel HASP Uninstall Command

```
# yum remove aksusbd
```

Executing the command above should generate output similar to the following. When asked "Is this ok?" Check the list of software to ensure it only includes the **aksusbd** package and answer **y(**es) to proceed.

Sentinel HASP Uninstall Output

```
Loaded plugins: langpacks
Resolving Dependencies
--> Running transaction check
---> Package aksusbd.x86_64 0:2.2.1-1 will be erased
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package       Arch         Version        Repository                      Size
================================================================================
Removing:
 aksusbd       x86_64       2.2.1-1        @/aksusbd-2.2.1-1.x86_64       4.5 M

Transaction Summary
================================================================================
Remove  1 Package

Installed size: 4.5 M
Is this ok [y/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Stopping aksusbd (via systemctl):  [  OK  ]
  Erasing    : aksusbd-2.2.1-1.x86_64                                     1/1
  Verifying  : aksusbd-2.2.1-1.x86_64                                     1/1

Removed:
  aksusbd.x86_64 0:2.2.1-1

Complete!
```

Removing the Sentinel HASP license software does not remove any site local configuration files located in `/etc/hasplm`.

Installing the CHARON-SSP Software

The CHARON-SSP for Linux virtual machine software is shipped as a tar, gzip distribution package. To begin the installation you must first obtain the package archive. This can be downloaded from Stromasys or distributed by read-only media, such as a CD-ROM. If you do not have the software package, then please contact either Stromasys or your Value Added Reseller for further help.

The CHARON-SSP for Linux software comes in two separate install packages. The table below describes these:

CHARON-SSP for GNU/Linux Installation Packages

Product Name	Description	Distribution Package Name
CHARON-SSP/4M	32-bit SPARC V8 virtual machine based on the sun4m architecture.	`charon-ssp-4m-1.0.22-x86_64_rpm.tar.gz`
CHARON-SSP/4U	64-bit SPARC V9 virtual machine based on the sun4u architecture.	`charon-ssp-4u-1.0.22-x86_64_rpm.tar.gz`

To begin, unpack the distribution archive, like so (the example below shows CHARON-SSP/4M for Linux):

Unpack CHARON-SSP/4M for GNU/Linux Distribution

```
# cd /tmp
# tar xzvf /path/to/charon-ssp-4m-1.0.22-x86_64_rpm.tar.gz
```

All installation steps are required to be performed from a privileged account as denoted by the `'#'` prompt.

Prerequisites

Before installing the CHARON-SSP for Linux virtual machine software, it is first necessary to install the Sentinel HASP software. Without this software it will not be possible for CHARON-SSP to locate the Sentinel HASP and license the environment appropriately.

For instructions on installing the Sentinel HASP software, please see the section Installing the Sentinel HASP Software.

Installing the CHARON-SSP Software

To install the CHARON-SSP for Linux virtual machine software, execute the relevant `yum` command for the specific package, shown in the table below.

CHARON-SSP for GNU/Linux Installation Commands

Product Name	Installation Command
CHARON-SSP/4M	`yum localinstall charon-ssp-4m-1.0.22-x86_64.rpm`
CHARON-SSP/4U	`yum localinstall charon-ssp-4u-1.0.22-x86_64.rpm`

The following terminal output shows the installation of CHARON-SSP/4M:

CHARON-SSP/4M for GNU/Linux Install Output

```
# yum localinstall charon-ssp-4m-1.0.22-x86_64.rpm
Examining charon-ssp-4m-1.0.22-x86_64.rpm: charon-ssp-4m-1.0.22-1.x86_64
Marking charon-ssp-4m-1.0.22-x86_64.rpm to be installed
Resolving Dependencies
--> Running transaction check
---> Package charon-ssp-4m.x86_64 0:1.0.22-1 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package          Arch      Version       Repository                       Size
================================================================================
Installing:
 charon-ssp-4m    x86_64    1.0.22-1      /charon-ssp-4m-1.0.22-x86_64    1.9 M

Transaction Summary
================================================================================
Install  1 Package

Total size: 1.9 M
Installed size: 1.9 M
Is this ok [y/d/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : charon-ssp-4m-1.0.22-1.x86_64                                1/1
  Verifying  : charon-ssp-4m-1.0.22-1.x86_64                                1/1

Installed:
  charon-ssp-4m.x86_64 0:1.0.22-1

Complete!
```

Post-Installation Tasks

The following post installations task are not required. However, for improved usability they are recommended.

Add the CHARON-SSP for Linux Software to the Shell PATH

To add the CHARON-SSP for Linux software to a C Shell environment, add the following to the end of `.login`:

Add CHARON-SSP for GNU/Linux Software to C Shell `.login`

```
setenv PATH $PATH:/opt/charon-ssp/ssp-4m:/opt/charon-ssp/ssp-4u
```

To add the CHARON-SSP for Linux software to a Bourne Shell environment (e.g. bash or sh), add the following to the end of `.profile`, `.bash_profile` or `.bashrc`:

Add CHARON-SSP for GNU/Linux Software to Bourne Shell Login Profile

```
PATH=$PATH:/opt/charon-ssp/ssp-4m:/opt/charon-ssp/ssp-4u
export PATH
```

Removing the CHARON-SSP Software

To install the CHARON-SSP for Linux virtual machine software, execute the relevant `yum` command for the specific package, shown in the table below.

CHARON-SSP for GNU/Linux Installation Commands

Product Name	Installation Command
CHARON-SSP/4M	`yum remove charon-ssp-4m`
CHARON-SSP/4U	`yum remove charon-ssp-4u`

The following shows the removal of the CHARON-SSP/4M for Linux software. When asked "Is this ok?" Check the list of software to ensure it only includes the CHARON-SSP for Linux package begin removed and answer **y(**es) to proceed.

CHARON-SSP/4M for GNU/Linux Uninstall Output

```
# yum remove charon-ssp-4m
Resolving Dependencies
--> Running transaction check
---> Package charon-ssp-4m.x86_64 0:1.0.22-1 will be erased
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package          Arch      Version      Repository                        Size
================================================================================
Removing:
 charon-ssp-4m    x86_64    1.0.22-1     @/charon-ssp-4m-1.0.22-x86_64    1.9 M

Transaction Summary
================================================================================
Remove  1 Package

Installed size: 1.9 M
Is this ok [y/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction

  Erasing    : charon-ssp-4m-1.0.22-1.x86_64                                1/1
  Verifying  : charon-ssp-4m-1.0.22-1.x86_64                                1/1

Removed:
  charon-ssp-4m.x86_64 0:1.0.22-1

Complete!
```

During the uninstall process, only the CHARON-SSP for Linux software is removed. All user data, including virtual disks, configuration files and virtual tapes are left untouched. It is left to the user to archive and/or delete this files.

Upgrading CHARON-SSP for Linux

To upgrade the CHARON-SSP for Linux software, simply follow the instructions described in the section Installing the CHARON-SSP Software. The terminal output below shows an upgrade from CHARON-SSP/4M for Linux v1.0.18 to v1.0.22.

Upgrading CHARON-SSP/4M for GNU/Linux

```
# yum localinstall charon-ssp-4m-1.0.22-x86_64.rpm
Examining charon-ssp-4m-1.0.22-x86_64.rpm: charon-ssp-4m-1.0.2-1.x86_64
Marking charon-ssp-4m-1.0.22-x86_64.rpm as an update to charon-ssp-4m-1.0.18-1.x86_64
Resolving Dependencies
--> Running transaction check
---> Package charon-ssp-4m.x86_64 0:1.0.18-1 will be updated
---> Package charon-ssp-4m.x86_64 0:1.0.22-1 will be an update
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package          Arch      Version       Repository                       Size
================================================================================
Updating:
 charon-ssp-4m    x86_64    1.0.22-1      /charon-ssp-4m-1.0.22-x86_64    1.9 M

Transaction Summary
================================================================================
Upgrade  1 Package

Total size: 1.9 M
Is this ok [y/d/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Updating   : charon-ssp-4m-1.0.22-1.x86_64                                1/2

  Cleanup    : charon-ssp-4m-1.0.18-1.x86_64                                2/2
  Verifying  : charon-ssp-4m-1.0.22-1.x86_64                                1/2
  Verifying  : charon-ssp-4m-1.0.18-1.x86_64                                2/2

Updated:
  charon-ssp-4m.x86_64 0:1.0.22-1

Complete!
```

Installing the CHARON-SSP Manager

The CHARON-SSP Manager client software is available for both Windows and Linux. Both packages can be downloaded from Stromasys or distributed on read-only media, such as CD-ROM. If you do not have the software package, then please contact either Stromasys or your Value Added Reseller (VAR) for further help.

The following sections describe the procedures for installing, upgrading and removing CHARON-SSP Manager for Windows and Linux.

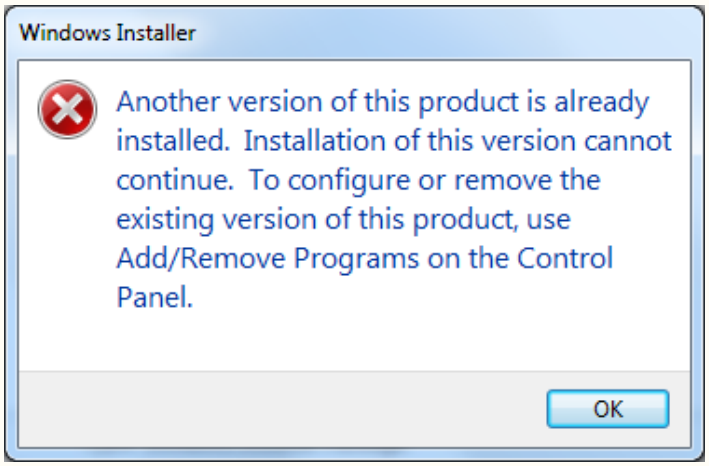
Installing CHARON-SSP Manager for Windows

The following sections cover the procedures for installing, removing and upgrading the CHARON-SSP Manager for Windows software.

Installing the CHARON-SSP Manager Software for Windows

The CHARON-SSP Manager for Windows software is shipped as a zipped archive package. To complete the installation, use the following instructions.

Installing CHARON-SSP Manager for Windows

Step	Description
1.	**Right-click** the zip archive, `SSP-Manager for Windows.zip` and select **Extract All...**
2.	A window titled **Extract Compressed (Zipped) Folders** will display. On this window: • **Click** the checkbox labelled **Show extracted files when complete.** • **Click** the button labelled **Extract.**
3.	A new **Windows Explorer** window will open showing the extracted packages.
4.	**Double-click** the `setup.exe` executable to begin the installation.
5.	If you are presented with an **Open File - Security Warning** window, **click** the button labelled **Run.**
6.	You should now be presented with the **SSP-Manager Setup Wizard.** To proceed with the installation **click** the button labelled **Next.** If you are presented with the following **Windows Installer** error, CHARON-SSP Manager for Windows is already installed and you need to follow the procedure documented in Upgrading the CHARON-SSP Manager for Windows. 
7.	To accept the default installation options, simply **click** the button labelled **Next** without modifying any options. Alternatively the following installation options can be adjusted: • **Click** the button labelled **Browse** to select an alternative installation target. • **Click** the appropriate radio button, **Everyone** or **Just for Me** to specify system-wide or private installation, respectively. • To determine the approximate disk usage, post install, **click** the button labelled **Disk Cost.** Once all options have been set, **click** the button labelled **Next.**
8.	Proceed with the installation by **clicking** the button labelled **Next.**
9.	Once the installation has completed, **click** the button labelled **Close** to exit the SSP-Manager Setup Wizard.
10.	The installation process will have created: • An **SSP-Manager** icon on the desktop, and • A **Stromasys/CHARON-Manager/SSP-Manager** folder under the **Start** menu.

Post Installation Tasks

The following post-installation tasks are not required however, they are highly recommended.

It is also recommended by Stromasys that **Windows Update** is checked for any critical Microsoft .NET Framework updates and that if any are available, they are installed.

Running CHARON-SSP Manager as an Administrator

To run the CHARON-SSP Manager for Windows correctly, it is necessary to run as the Administrator user. To configure this to be on by default, follow the steps described below.

Instructions for Running CHARON-SSP Manager as an Administrator

Step	Description
1.	Locate the CHARON-SSP Manager for Windows executable. The default installation will place it in the path, `C:\Program Files (x86)\Stromasys\CHARON-Manager\SSP-Manager\bin\ssp-manager.exe`.
2.	**Right-click** on the `ssp-manager.exe` executable. • **Click** the **Properties** item from the menu.
3.	In the window labelled **`ssp-manager.exe Properties`, click** the tab labelled **Compatibility**.
4.	In the **Privilege Level** panel, tick the checkbox labelled **Run this program as an administrator**.
5.	**Click** the button labelled **OK** to save the changes.

Removing the CHARON-SSP Manager for Windows

To remove the CHARON-SSP Manager software, follow the steps listed below for the specific host operating systems.

Microsoft Windows 8 and Windows Server 2012

Removing CHARON-SSP Manager from Microsoft Windows 8 and Windows Server 2012

Step	Task
1	• Press **WinKey+X** to access the **Power User Menu**. • Click **Control Panel**.
2	• Switch **View by** to **Small Icons** or **Large Icons**. • Click **Programs and Features**.
3	• Select **SSP-Manager** from the list of installed software. • Click **Uninstall**.

Microsoft Windows 7, Vista, Server 2008 and Server 2008 R2

Removing CHARON-SSP Manager from Microsoft Windows 7, Vista, Server 2008 and Server 2008 R2

Step	Task
1	Click **Start**, then click **Control Panel**.
2	• Switch **View by** to **Small Icons** or **Large Icons**. • Click **Programs and Features**.
3	• Select **SSP-Manager** from the list of installed software • Click **Uninstall**.

Microsoft Windows XP and Server 2003

Removing CHARON-SSP Manager from Microsoft Windows XP and Server 2003

Step	Task
1	Click **Start**, then click **Control Panel**.
2	Double-click **Add or Remove Programs**.
3	• Select **SSP-Manager** from the installed software list. • Click **Remove**.

Upgrading the CHARON-SSP Manager for Windows

To upgrade the CHARON-SSP Manager for Windows software, it is necessary to first remove the existing package, then install the new version.

The process for removing CHARON-SSP Manager for Windows is described in the section, Removing the CHARON-SSP Manager for Windows. Once removed, use the steps described in the section, Installing the CHARON-SSP Manager Software for Windows to complete the upgrade of CHARON-SSP Manager for Windows.

Installing the CHARON-SSP Manager for Linux

The following sections cover the procedures for installing, removing and upgrading the CHARON-SSP for Linux software.

Installing CHARON-SSP Manager for Linux

The CHARON-SSP Manager for Linux software is shipped as a tar, gzip distribution package. To begin the installation, unpack the distribution archive, like so:

Unpack CHARON-SSP Distribution

```
# cd /tmp
# tar xzvf /path/to/charon-manager-ssp-1.0.22-rpm.tar.gz
```

Once the archive is unpacked, remain in the directory **`/tmp`**. This is where the remaining installation steps will take place.

> All installation steps are required to be performed from a privileged account as denoted by the **`'#'`** prompt.

To install the CHARON-SSP Manager for Linux software, execute the following command:

CHARON-SSP Manager for Linux Install Command

```
# yum localinstall charon-manager-ssp-1.0.22-rpm
```

Executing this command will produce output similar to the following:

CHARON-SSP Manager for Linux Install Output

```
Loaded plugins: langpacks
Examining charon-manager-ssp-1.0.22.rpm: charon-manager-ssp-1.0.22-1.x86_64
Marking charon-manager-ssp-1.0.22.rpm to be installed
Resolving Dependencies
--> Running transaction check
---> Package charon-manager-ssp.x86_64 0:1.0.22-1 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package              Arch     Version       Repository                    Size
================================================================================
Installing:
 charon-manager-ssp   x86_64   1.0.22-1      /charon-manager-ssp-1.0.22   2.5 M

Transaction Summary
================================================================================
Install  1 Package

Total size: 2.5 M
Installed size: 2.5 M
Is this ok [y/d/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : charon-manager-ssp-1.0.22-1.x86_64                          1/1
  Verifying  : charon-manager-ssp-1.0.22-1.x86_64                          1/1

Installed:
  charon-manager-ssp.x86_64 0:1.0.22-1

Complete!
```

Post Installation Tasks

The following post installation tasks are not required. However, for greater usability they are highly recommended.

Installing the Xephyr X11 Server

To use the X11 Server from CHARON-SSP Manager (described in Accessing the Console) it is necessary to install the Xephyr X11 Server. Use the following commands to install the software:

Installing Xephyr X11 Server on RedHat Linux

```
# yum install xorg-x11-server-Xephyr
```

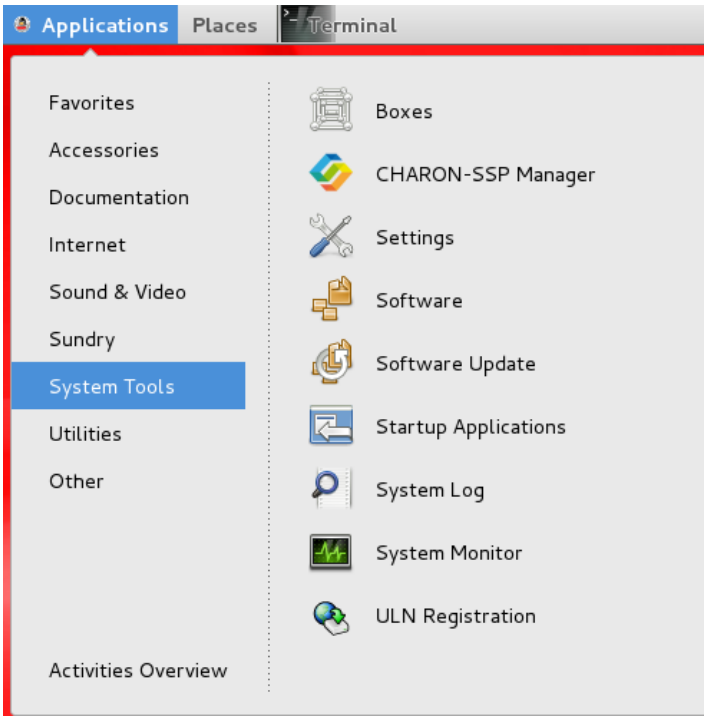
Create a CHARON-SSP Manager Menu Item

To create an entry in the Applications menu for the CHARON-SSP Manager follow the steps below.

Step	Description
1.	Using `sudo` or a privileged account, create the file `/usr/local/share/applications/charon-ssp-manager.desktop`.

2.	Add the following text to the file created in step 1 and save it.
	`[Desktop Entry]`
	`Version=1.0.22`
	`Name=CHARON-SSP Manager`
	`Exec=/opt/charon-manager/ssp-manager/ssp-manager`
	`Icon=/opt/charon-manager/ssp-manager/resource/charon.png`
	`Terminal=false`
	`Type=Application`
	`StartupNotify=true`
	`Categories=System;`

3.	Set the file protections and ownership appropriately:
	`# chmod 0644 /usr/local/share/applications/charon-ssp-manager.desktop`
	`# chown root:root /usr/local/share/applications/charon-ssp-manager.desktop`

4.	Navigate to **Applications/System Tools.** There should be an entry, much like the following: 

Removing the CHARON-SSP Manager Software

To remove the CHARON-SSP Manager for Linux software, execute the following command:

CHARON-SSP Manager for Linux Uninstall Command

```
# yum remove charon-manager-ssp
```

Executing the command above should generate output similar to the following. When asked "Is this ok?" Check the list of software to ensure it only includes the **`charon-manager-ssp`** package and answer **`y(`**es) to proceed.

CHARON-SSP Manager for Linux Uninstall Output

```
Loaded plugins: langpacks
Resolving Dependencies
--> Running transaction check
---> Package charon-manager-ssp.x86_64 0:1.0.22-1 will be erased
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package              Arch     Version      Repository                     Size
================================================================================
Removing:
 charon-manager-ssp   x86_64   1.0.22-1     @/charon-manager-ssp-1.0.22   2.5 M

Transaction Summary
================================================================================
Remove  1 Package

Installed size: 2.5 M
Is this ok [y/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Erasing    : charon-manager-ssp-1.0.22-1.x86_64                          1/1
  Verifying  : charon-manager-ssp-1.0.22-1.x86_64                          1/1

Removed:
  charon-manager-ssp.x86_64 0:1.0.22-1

Complete!
```

During the uninstall process, only the CHARON-SSP Manager for Linux software is removed. All user data, including configuration files are left untouched. It is left to the user to archive and/or delete this files.

Post Removal Tasks

The following post removal tasks are not required. However, they are highly recommended.

Remove the CHARON-SSP Manager Menu Item

To test, and remove if present, the CHARON-SSP Manager menu item, execute the following shell script.

Removing the CHARON-SSP Manager Menu Item

```
# MENUITEM=/usr/local/share/applications/charon-ssp-manager.desktop
# if [ -e "$MENUITEM" ]; then rm $MENUITEM; fi
```

The above commands should be executed either via `sudo` or from a privileged account.

Upgrading CHARON-SSP Manager for Linux

To upgrade the CHARON-SSP Manager for Linux software, simply follow the instructions described in the section Installing the CHARON-SSP Manager for Linux. The terminal output below shows an upgrade from CHARON-SSP Manager for Linux v1.0.18 to v1.0.22.

Upgrading CHARON-SSP/4M for Linux

```
# yum localinstall charon-manager-ssp-1.0.22.rpm
Loaded plugins: langpacks
Examining charon-manager-ssp-1.0.22.rpm: charon-manager-ssp-1.0.22-1.x86_64
Marking charon-manager-ssp-1.0.22.rpm as an update to charon-manager-ssp-1.0.18-1.x86_64
Resolving Dependencies
--> Running transaction check
---> Package charon-manager-ssp.x86_64 0:1.0.18-1 will be updated
---> Package charon-manager-ssp.x86_64 0:1.0.22-1 will be an update
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package              Arch     Version       Repository                    Size
================================================================================
Updating:
 charon-manager-ssp   x86_64   1.0.22-1      /charon-manager-ssp-1.0.22   2.5 M

Transaction Summary
================================================================================
Upgrade  1 Package

Total size: 2.5 M
Is this ok [y/d/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Updating   : charon-manager-ssp-1.0.22-1.x86_64                          1/2
  Cleanup    : charon-manager-ssp-1.0.18-1.x86_64                          2/2
  Verifying  : charon-manager-ssp-1.0.22-1.x86_64                          1/2
  Verifying  : charon-manager-ssp-1.0.18-1.x86_64                          2/2

Updated:
  charon-manager-ssp.x86_64 0:1.0.22-1

Complete!
```

Installing the CHARON-SSP Agent

The CHARON-SSP Agent for Linux software is shipped as a tar, gzip distribution package. To begin the installation you must first obtain the package archive. This can be downloaded from Stromasys or distributed by read-only media, such as a CD-ROM. If you do not have the software package, then please contact either Stromasys or your Value Added Reseller for further help.

To begin, unpack the distribution archive, like so:

Unpack CHARON-SSP Agent Distribution

```
# cd /tmp
# tar xzvf /path/to/charon-agent-ssp-1.0.22-x86_64_rpm.tar.gz
```

Once the archive is unpacked, remain in the directory **`/tmp`.** This is where the remaining installation steps will take place.

> All installation steps a required to be performed from a privileged account as denoted by the **`'#'`** prompt.

Installing the CHARON-SSP Agent Software

To install the CHARON-SSP Agent for Linux software, execute the following command:

CHARON-SSP Agent for Linux Install Command

```
# yum localinstall charon-agent-ssp-1.0.22-x86_64.rpm
```

Executing this command will produce output similar to the following:

CHARON-SSP Agent for Linux Install Output

```
Loaded plugins: langpacks
Examining charon-agent-ssp-1.0.22-x86_64.rpm: charon-agent-ssp-1.0.22-1.x86_64
Marking charon-agent-ssp-1.0.22-x86_64.rpm to be installed
Resolving Dependencies
--> Running transaction check
---> Package charon-agent-ssp.x86_64 0:1.0.22-1 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package            Arch     Version    Repository                          Size
================================================================================
Installing:
 charon-agent-ssp   x86_64   1.0.22-1   /charon-agent-ssp-1.0.22-x86_64   2.4 M

Transaction Summary
================================================================================
Install  1 Package

Total size: 2.4 M
Installed size: 2.4 M
Is this ok [y/d/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : charon-agent-ssp-1.0.22-1.x86_64                            1/1

  Verifying  : charon-agent-ssp-1.0.22-1.x86_64                            1/1

Installed:
  charon-agent-ssp.x86_64 0:1.0.22-1

Complete!
```

Post-Installation Tasks

The following post installations task are not required. However, for improved security they are highly recommended.

Securing the Configuration

By default the software installation directory (which also contains the configuration files) for the CHARON-SSP Agent for Linux is unsecure. To ensure that the software and, especially the configuration, is secure against accidental or intentional access or corruption, it is necessary to execute the following commands:

Commands to Secure CHARON-SSP Agent Install Tree

```
# chmod -R go-rwx /opt/charon-agent
# chmod -R a-x /opt/charon-agent/ssp-agent/etc
```

The above commands should be executed by `sudo` or from a privileged account.

Removing the CHARON-SSP Agent Software

To remove the CHARON-SSP Agent for Linux software, execute the following command:

CHARON-SSP Agent for Linux Uninstall Command

```
# yum remove charon-agent-ssp
```

Executing the command above should generate output similar to the following. When asked "Is this ok?" Check the list of software to ensure it only includes the **`charon-agent-ssp`** package and answer **`y`**(es) to proceed.

CHARON-SSP Agent for Linux Uninstall Output

```
Loaded plugins: langpacks
Resolving Dependencies
--> Running transaction check
---> Package charon-agent-ssp.x86_64 0:1.0.22-1 will be erased
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package           Arch    Version      Repository                          Size
================================================================================
Removing:
 charon-agent-ssp  x86_64  1.0.22-1     @/charon-agent-ssp-1.0.22-x86_64  2.4 M

Transaction Summary
================================================================================
Remove  1 Package

Installed size: 2.4 M
Is this ok [y/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction

  Erasing    : charon-agent-ssp-1.0.22-1.x86_64                              1/1
  Verifying  : charon-agent-ssp-1.0.22-1.x86_64                              1/1

Removed:
  charon-agent-ssp.x86_64 0:1.0.22-1

Complete!
```

During the uninstall process, only the CHARON-SSP Agent for Linux software is removed. All user data, including configuration files are left untouched. It is left to the user to archive and/or delete this files.

Upgrading CHARON-SSP Agent for Linux

To upgrade the CHARON-SSP Agent for Linux software, simply follow the instructions described in the section Installing the CHARON-SSP Agent Software. The terminal output below shows an upgrade from CHARON-SSP Agent for Linux v1.0.18 to v1.0.22.

Upgrading CHARON-SSP/4M for Linux

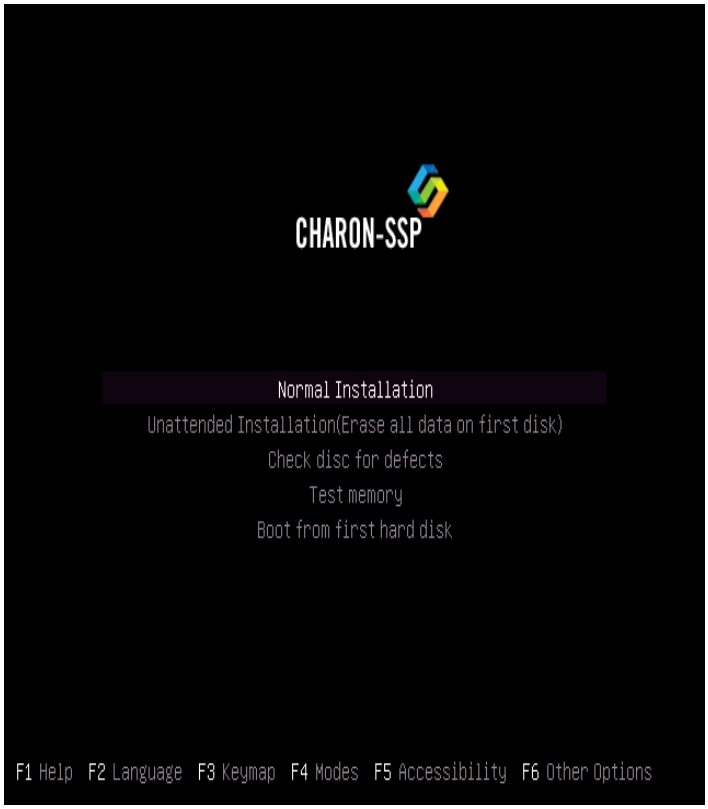
```
# yum localinstall charon-agent-ssp-1.0.22-x86_64.rpm
Loaded plugins: langpacks
Examining charon-agent-ssp-1.0.22-x86_64.rpm: charon-agent-ssp-1.0.22-1.x86_64
Marking charon-agent-ssp-1.0.22-x86_64.rpm as an update to charon-agent-ssp-1.0.18-1.x86_64
Resolving Dependencies
--> Running transaction check
---> Package charon-agent-ssp.x86_64 0:1.0.18-1 will be updated
---> Package charon-agent-ssp.x86_64 0:1.0.22-1 will be an update
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package            Arch     Version    Repository                          Size
================================================================================
Updating:
 charon-agent-ssp   x86_64   1.0.22-1   /charon-agent-ssp-1.0.22-x86_64   2.4 M

Transaction Summary
================================================================================
Upgrade  1 Package

Total size: 2.4 M
Is this ok [y/d/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Updating   : charon-agent-ssp-1.0.22-1.x86_64                            1/2


  Cleanup    : charon-agent-ssp-1.0.18-1.x86_64                            2/2
  Verifying  : charon-agent-ssp-1.0.22-1.x86_64                            1/2
  Verifying  : charon-agent-ssp-1.0.18-1.x86_64                            2/2

Updated:
  charon-agent-ssp.x86_64 0:1.0.22-1

Complete!
```

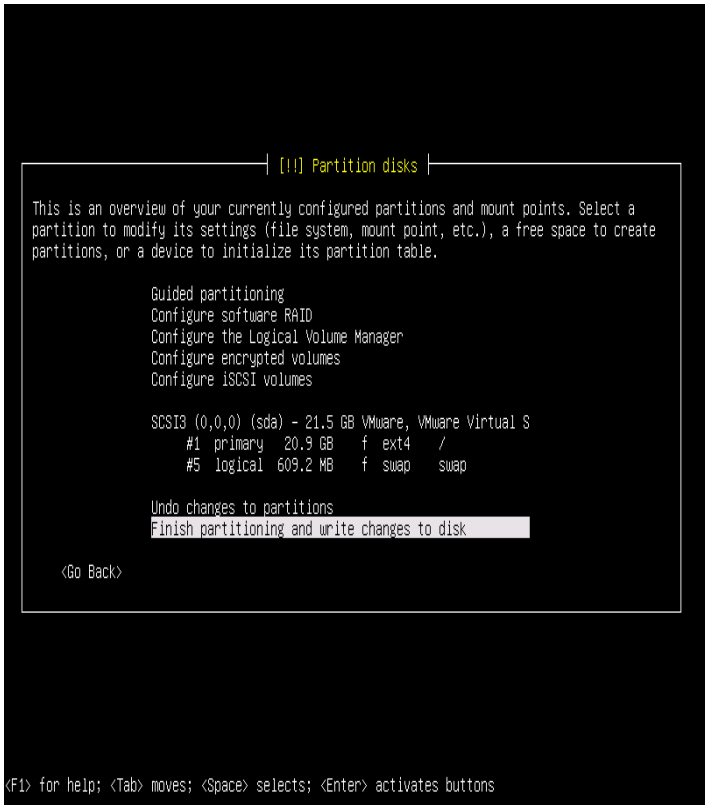
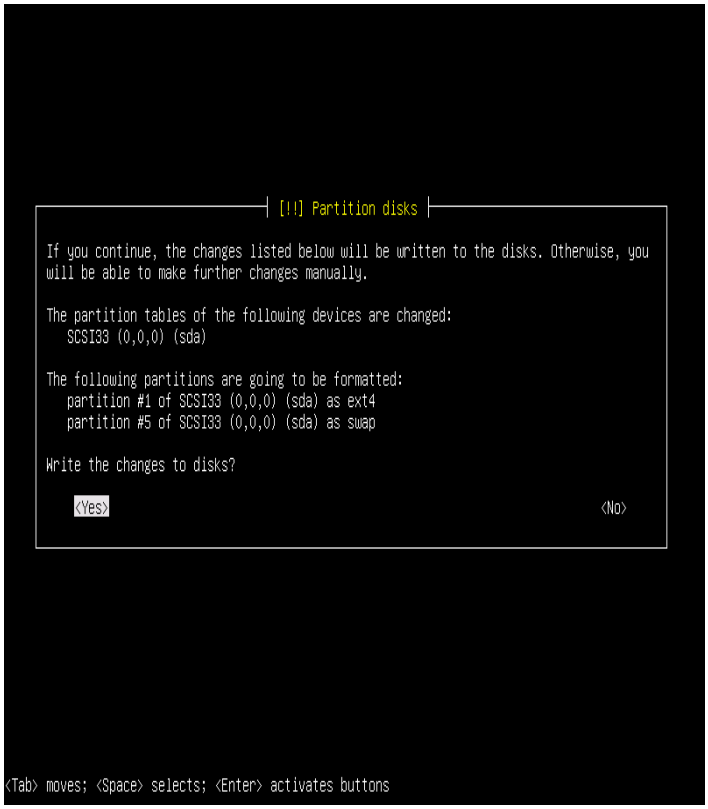
Installing the CHARON-SSP Virtual Environment

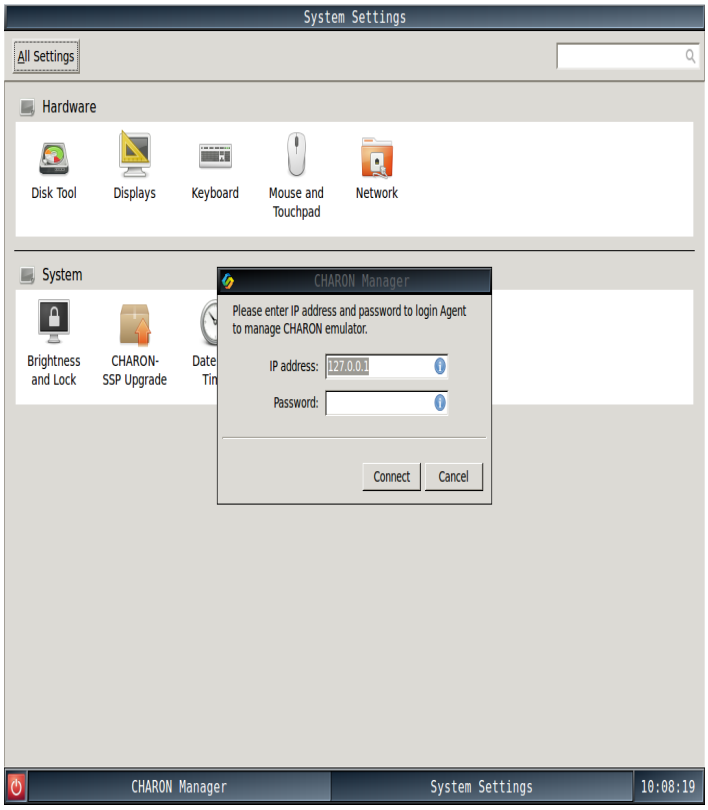
The CHARON-SSP Virtual Environment is an appliance version of the CHARON-SSP for Linux virtual machine and associated utilities. Distribution of this software is either via an ISO image or physical read-only media, such as a CD-ROM or DVD. If you do not have a copy of the CHARON-SSP Virtual Environment, then please contact either Stromasys or you Value Added Reseller for further help.

Installation Tasks

CHARON-SSP Installation Tasks

Step	Description
1.	If the host system is a virtual machine: • Configure for a Linux kernel 2.4.x x86-64 environment. • Use the requirements laid out in Host System Requirements to configure the virtual hardware. • Attach the CHARON-SSP Virtual Environment ISO or physical CD to the VM. • Power up the VM and boot from the CD. If the host system is a physical machine: • Load the CHARON-SSP CD. • Boot the system from the CD.
2.	Upon successful boot, you should see the following screen:  Select **Normal Installation** and press Enter.

3.	When presented with the following screen:  Select **Finish partitioning and write changes to disk** and press Return.
4.	When asked to confirm the changes, select **Yes** and press Return. 

4.	• The base installation will now continue. At completion the system will reboot. • Ensure that the CD media is ejected to prevent booting the CHARON-SSP Virtual Environment installer again.
5.	Following successful installation, you should be presented with the following screen: 

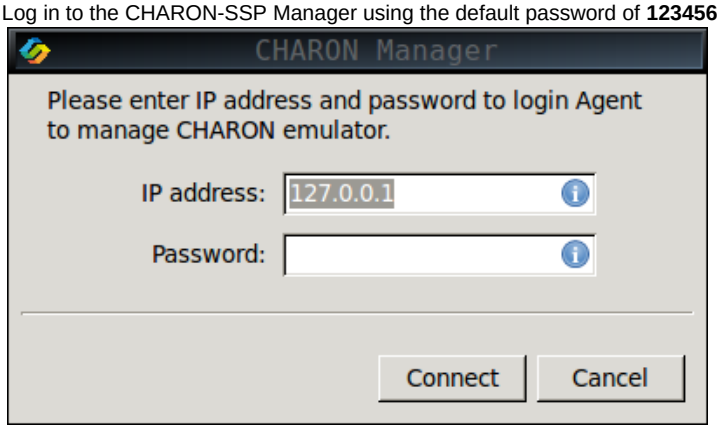
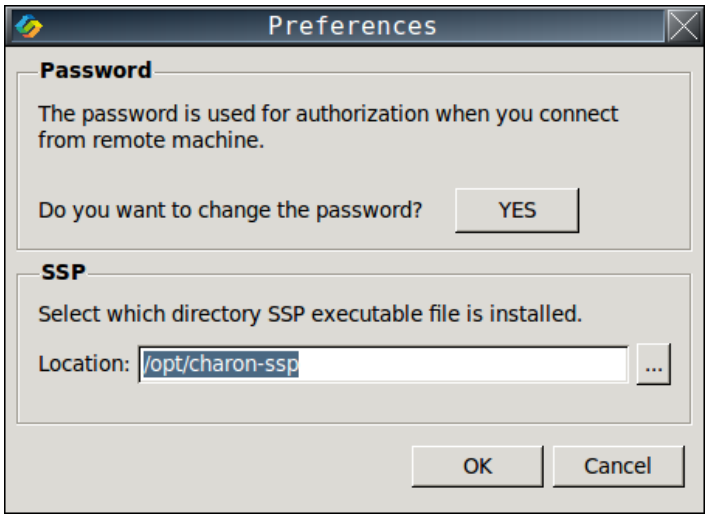
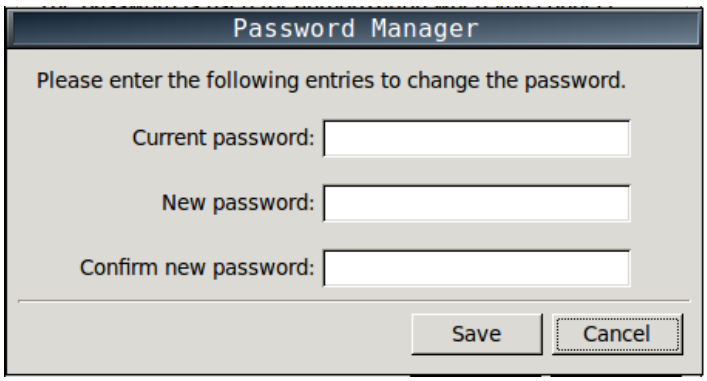
Post Installation Tasks

The following post installations task are not required. However, for improved security they are highly recommended.

Change CHARON-SSP Manager Password

The primary interface to the CHARON-SSP Virtual Environment is the CHARON-SSP Manager. For improved security it is highly recommended that the password be changed from the default of **123456**. The following instructions describe the necessary steps to change this password.

CHARON-SSP Manager Password Change Instructions

Step	Description
1.	Log in to the CHARON-SSP Manager using the default password of **123456** and click **Connect**. 
2.	Click the **Virtual Machine** menu.
3.	From the **Virtual Machine** drop-down menu, click **Preferences...**
4.	At the label **Do you want to change the password?**, click **YES**. 
5.	• Enter the existing password in the field, **Current password.** • Enter the new password in the fields, **New password** and **Confirm new password**. 
6.	Click **Save**.

Change CHARON-SSP Virtual Environment Password

The CHARON-SSP Virtual Environment runs under the **charon-ssp** user account. For improved security it is highly recommended that the default password of **foxandtree** is changed. The following instructions describe the necessary steps to change this password.

CHARON-SSP Virtual Environment Password Change Instructions

Step	Description
1.	Press **Ctrl+Winkey+F12** to start a shell.
2.	Execute the command **`passwd`**.
3.	The **`passwd`** program will provide the following prompts. Enter the appropriate information at each one: • `Current password:` • `New password:` • `Confirm new password:`
4.	Type **`exit`** to leave the shell and return to the CHARON-SSP Virtual Environment.

Set the Default Webpage to License Utility

To configure the web browser to automatically load the Sentinel HASP License utility on start up, follow the instructions below:

Instruction to Configure License Utility as Default Webpage

Step	Description
1.	Under the **System Settings** window, **click** the **Web Browser** icon.
2.	Once the Web Browser has loaded, **click Edit** > **Preferences.**
3.	Under the **General** tab in the **Preferences** window, enter the following into the field labelled **Home Page**: `http://127.0.0.1:1947`
4.	**Click** the button labelled **Close.**
5.	Exit the Web Browser and re-enter to see the License Utility load when the Web Browser starts.

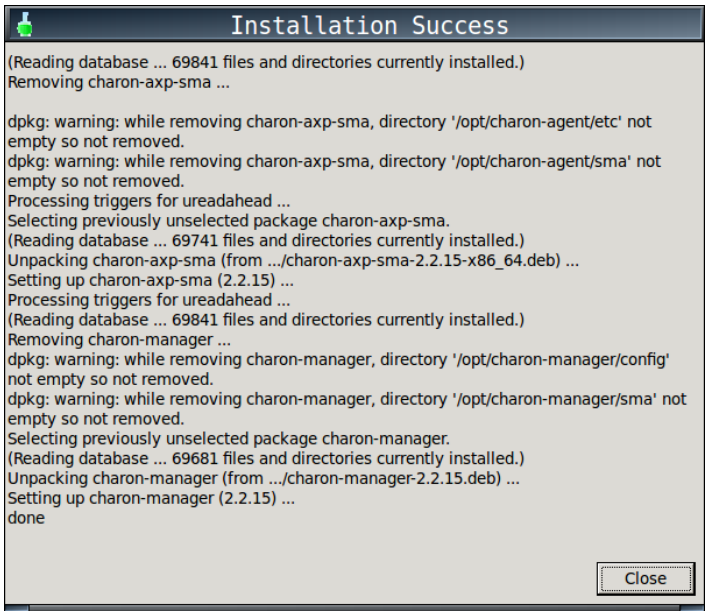
License Security

To ensure proper security for the CHARON-SSP Virtual Environment appliance, please follow the post-installation task describes in the section Installing the Sentinel HASP Software.

Upgrading CHARON-SSP Virtual Environment

The CHARON-SSP Virtual Environment is upgraded through the use of a USB memory stick. The software update kit can be obtained from Stromasys as a download, which can then be transferred to a USB memory stick. Once this has been done, follow the instructions below to complete the upgrade of the CHARON-SSP Virtual Environment software.

CHARON-SSP Virtual Evnrionment Upgrade Instructions

Step	Description
1.	**Double-click** the icon labelled **CHARON-SSP Upgrade** on the **System Settings** window.
2.	• A window titled **Preparation** will pop-up prompting for the insertion of the USB memory stick. • Insert the USB stick into a free port on the system and **click** the button labelled **Next**. If the CHARON-SSP Virtual Environment is installed on a virtual machine it may be necessary to attach the host connected USB memory stick to the virtual machine before beginning this step.
3.	The CHARON-SSP Virtual Environment upgrade utility will search the USB memory stick for the software kit. • Once found a window titled **Find charon-barebone-ssp-*x.x.x*.deb** (where ***x.x.x*** is replaced with the version number) will pop up. • **Click** the button labelled **UPGRADE** to proceed with the upgrade.
4.	Once the upgrade has completed, a window similar to the following will be displayed detailing the upgrade process. 
5.	To complete the installation it is recommended to reboot the CHARON-SSP Virtual Environment.

Using the CHARON-SSP Software

The following sections describe in detail each of the individual parts of CHARON-SSP for Linux and how to use them:

For the CHARON-SSP Virtual Environment appliance the most relevant documentation is Using the CHARON-SSP Manager. This describes the primary interface used to manage the hosted virtual machines. However, for the rare instances where it may be necessary to move to the command line, the sections Using CHARON-SSP from the Command Line and Using the CHARON-SSP Agent are equally useful.

> To access the command line under CHARON-SSP Virtual Environment press **Ctrl+WinKey+F12**.

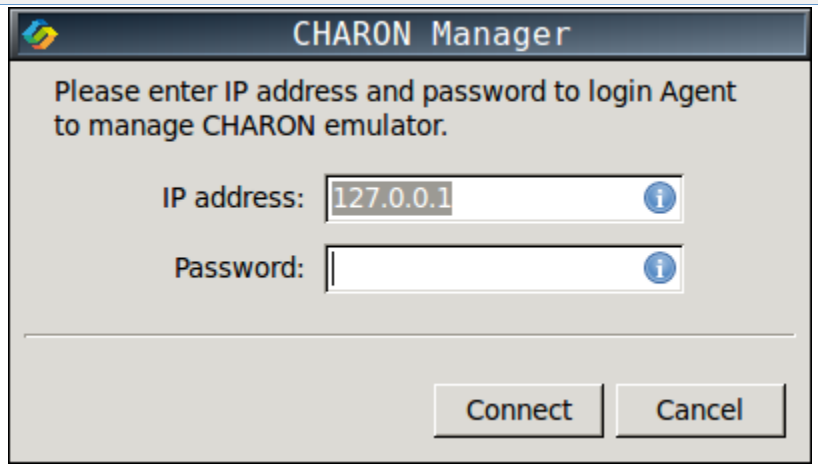
Using the CHARON-SSP Manager

The CHARON-SSP Manager is a GUI management interface for the CHARON-SSP for Linux virtual machine. Using this management utility it is possible to manage multiple virtual machines, licenses and virtual networks across a number of remote hosts. It is also the interface presented on the console of a system running the CHARON-SSP Virtual Environment appliance.

Getting Connected

Upon starting the CHARON-SSP Manager you will immediately be presented with a login window similar to the following.

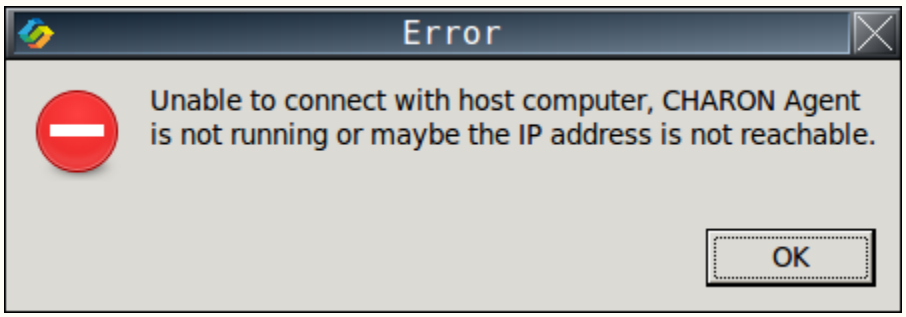
CHARON-SSP Manager Login Window



> For CHARON-SSP Virtual Environment users, the CHARON-SSP Manager window starts automatically on the console.

To proceed, enter the address of the system running the CHARON-SSP Agent into the field labelled **IP address** and the password for that system into the field labelled **Password.** The default password is **123456**. If the password was not changed by following the post-installation tasks, it is highly recommended that this is corrected immediately. Leaving the management console password unchanged poses a significant security risk.

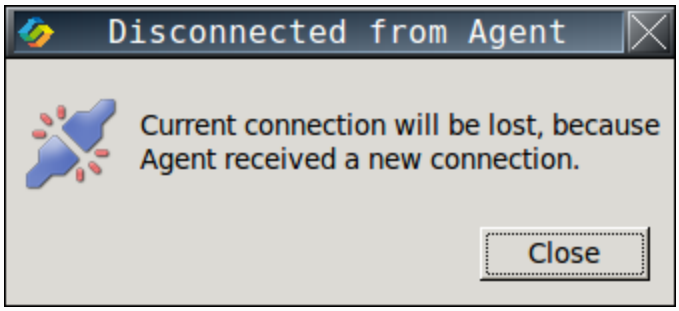
To connect to the CHARON-SSP Agent system click the button labelled **Connect**.

> If you receive an error similar to the following, check to ensure that the host specified in the **IP address** field is correct and the CHARON-SSP Agent is installed and running on the host.
>
> 
>

Managing the Virtual Environment

The following sections describe the different aspects of managing the virtual environment using the CHARON-SSP Manager.

> It is only possible to have one connection to the CHARON-SSP Agent at any one time. A second connection from an alternate CHARON-SSP Manager client will result in immediate disconnection and display of the following message.


Disconnected from Agent
Current connection will be lost, because
Agent received a new connection.
Close

Managing Virtual Machines

The following sections describe the different aspects of managing a CHARON-SSP virtual machine.

Creating a Virtual Machine

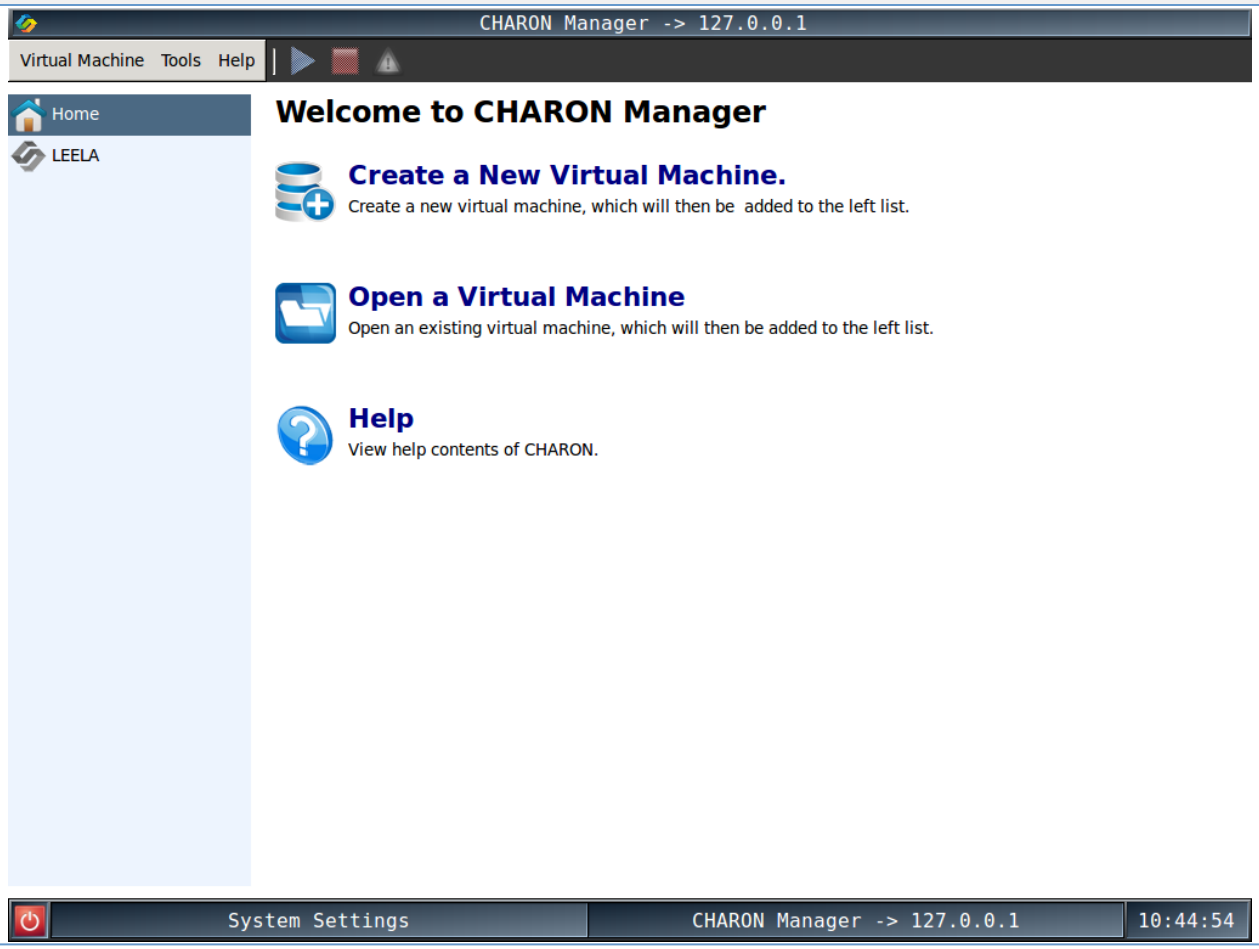
The first step to running a SPARC-based virtual machine is creating the initial configuration. To do this, follow the steps below.

Steps to Create a Virtual Machine

Step	Descriptions
1.	From the opening splash screen entitled **Welcome to CHARON Manager, click** the icon labelled **Create a New Virtual Machine**.
2.	Select the appropriate **Hardware Model** by clicking the radio button labelled with the SPARC model that most closely matches the system to you wish to run.
3.	Enter a name for the virtual machine into the field labelled, **Virtual machine name**.
4.	**Click** the button labelled **OK**.

The steps above will create a new virtual machine configuration. It will appear in the left hand pane of the management interface labelled with the **Virtual machine name** you specified. The screenshot below shows the management interface splash screen after the virtual machine LEELA was created.

CHARON-SSP Manager Opening Screen



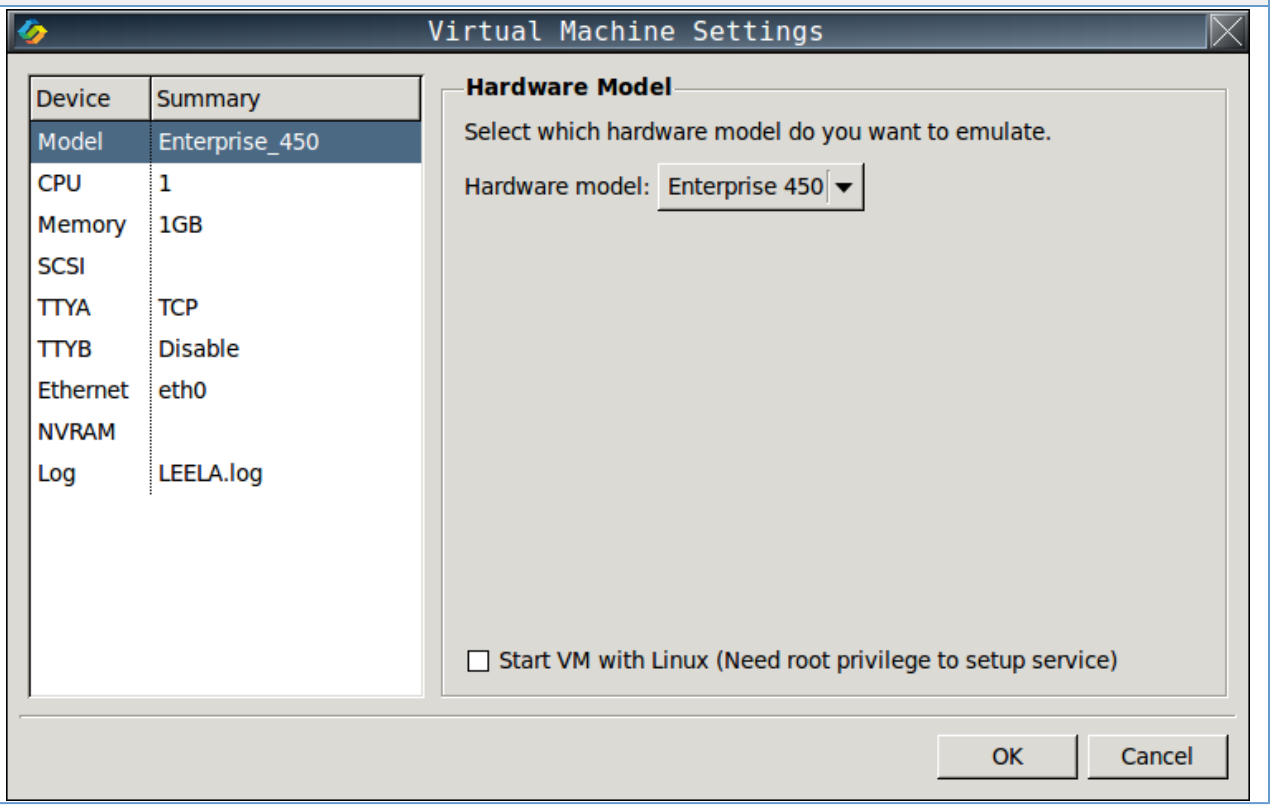
The initial creation of the virtual machine is very sparse. To complete the configuration it is important to continue on to Modifying the Virtual Machine.

Modifying a Virtual Machine

Whether completing configuration of a newly created virtual machine or adjusting the set up of an existing configuration the **Virtual Machine Settings** window can be opened by first **clicking** the name of the virtual machine in the left-hand pane of the management console. This should be followed by **clicking** the button labelled **Edit Virtual Machine.**

The example below shows the **Virtual Machine Settings** window for a virtual Enterprise 450 system.

Edit Virtual Machine Settings Window



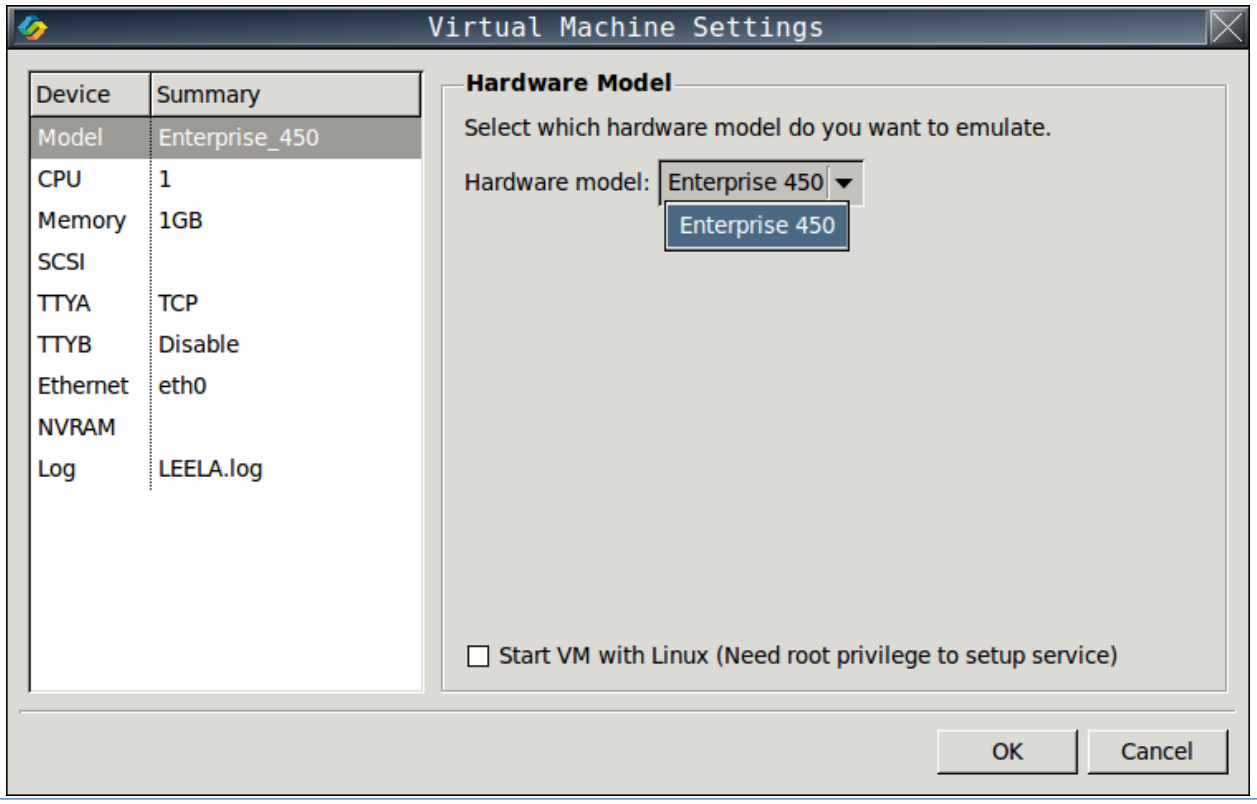
The following section describes each part of the **Virtual Machine Settings** window

> For any changes to take effect, the virtual machine must be restarted. However, it is also recommended that before making any configuration changes the virtual machine is shutdown correctly.

Model Configuration

To view or change the virtual machine model, select **Model** in the **Device** column of the left hand pane. The current setting will be displayed in the field labelled **Hardware Model**. To change the model, **click** the **Hardware Model** drop-down box and select the appropriate model (see the example below).

Virtual Machine Model Configuration Window



The models currently supported by CHARON-SSP/4M for Linux are:

- Sun SPARCstation 20

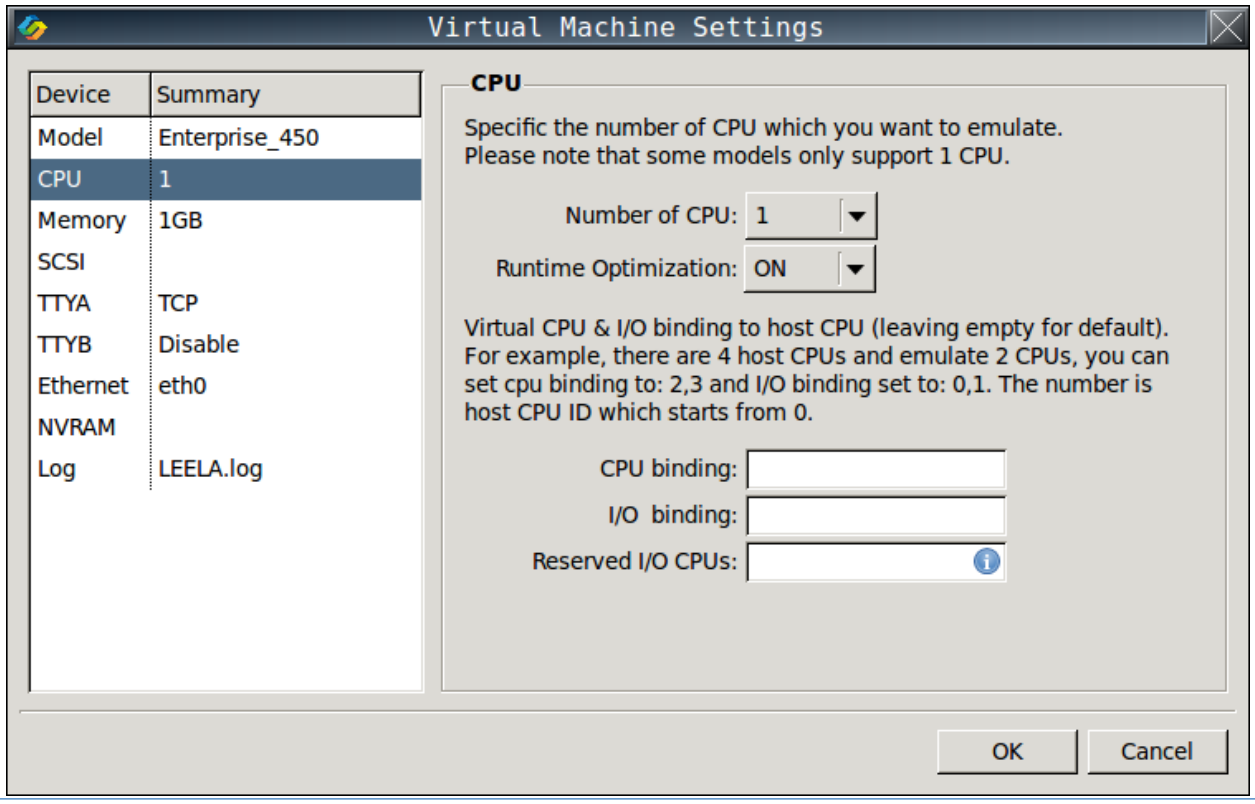
The models currently supported by CHARON-SSP/4U for Linux are:

- Sun Enterprise 450

CPU Configuration

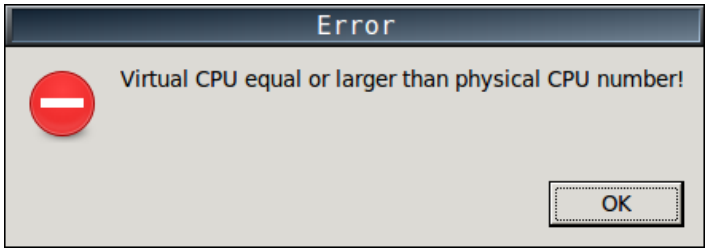
To view or change the current virtual machine CPU configuration, select **CPU** in the **Device** column of the left-hand pane.

Virtual Machine CPU Configuration Window



The following table lists each of the fields in the CPU configuration window and describes their operation.

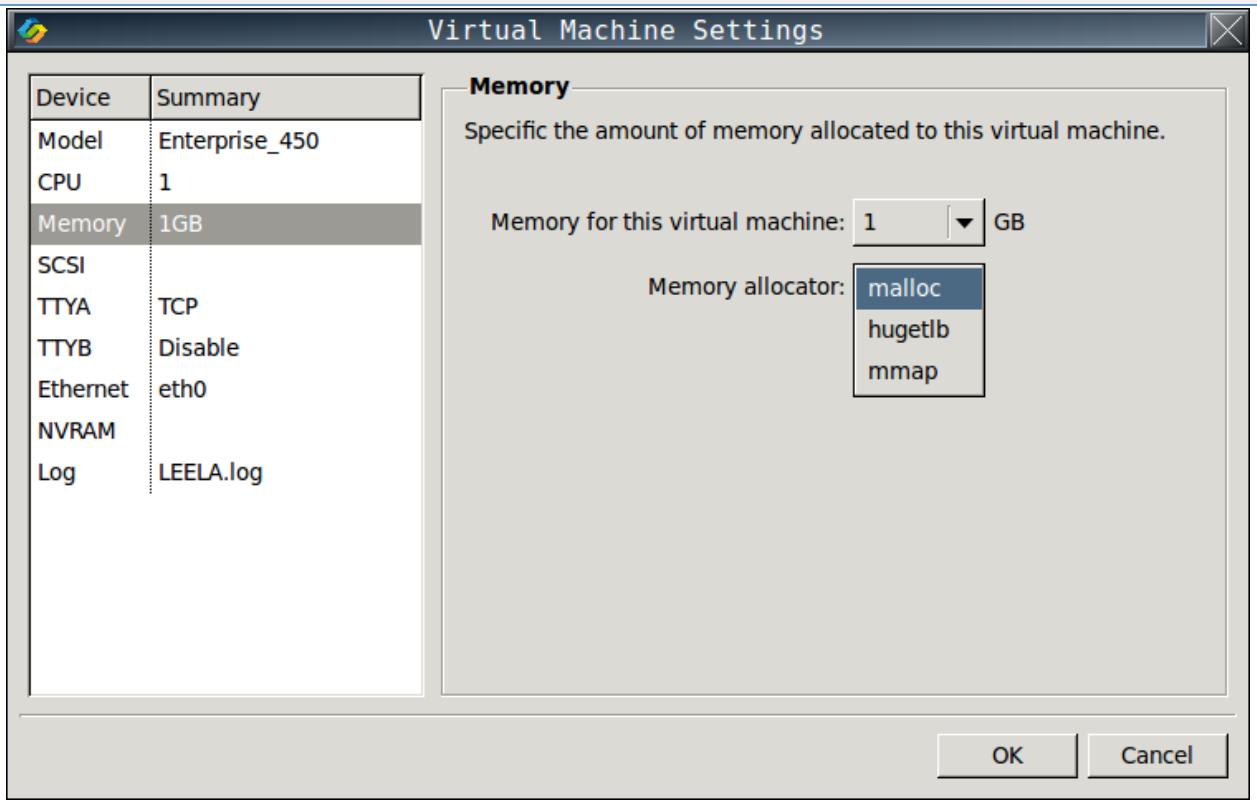
Virtual Machine CPU Configuration Fields

Field	Description
Number of CPU	Configure the number of CPUs attached to the virtual machine. Not all virtual SPARC systems support the configuration of multiple CPUs. For those virtual systems that do support multiple CPUs, they can still be restricted by the formula: `Physical CPUs = Virtual CPUs + 1` An attempt to configure more virtual CPUs than allowed will result in the following error message: 
Runtime Optimization	This option controls the Dynamic Instruction Translation (DIT). This option is on by default.
CPU binding	Assign specific host CPUs to the processing of SPARC instructions. This field is a comma-delimited list of CPU IDs. Leaving this blank will cause the virtual machine software to assign affinity itself.
I/O binding	Assign specific host CPUs to the processing of virtual machine I/O requests. This field is a comma-delimited list of CPU IDs. Leaving this blank will cause the virtual machine to assign I/O processing affinity itself.
Reserved I/O CPUs	Reserve a number of CPUs on the host system for the purpose of processing virtual machine I/O requests. If neither **I/O binding** and **Reserved I/O CPUs** are not set, then the virtual machine software will assign 1/3 (rounded down) of the host CPUs to I/O processing.

Memory Configuration

To view or change the current virtual machine memory configuration, select **Memory** in the **Device** column of the left-hand pane.

Virtual Machine Memory Configuration Widows



The following table lists each of the fields in the memory configuration window and describes their operation.

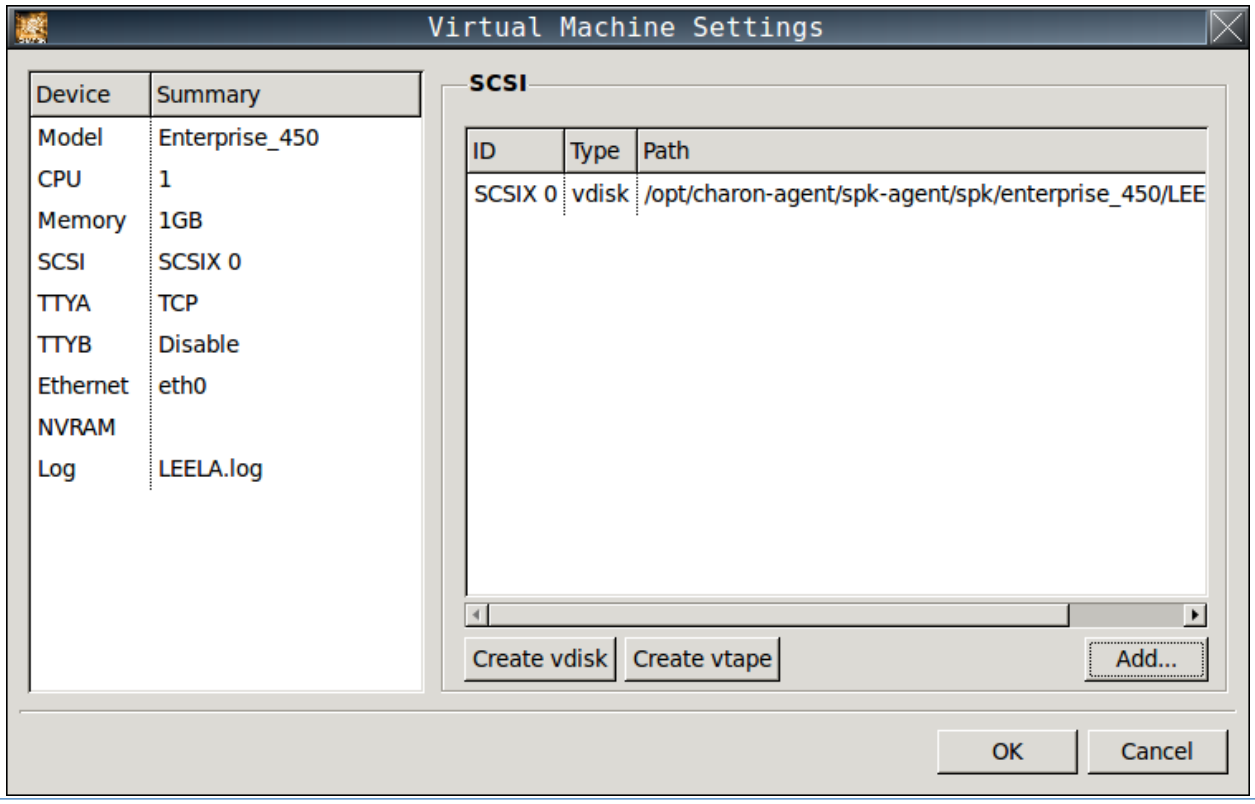
Virtual Machine Memory Configuration Fields

Field	Description	
Memory for this virtual machine	Set the amount of RAM allocated to the virtual machine. Memory can only be allocated in certain increments. The table below describes the allocation rules for each virtual machine model.	
	Virtual Machine Model	**Memory Allocation Rules**
	Sun SPARCstion 20	64MB, 128MB, 256MB and 512MB.
	Sun Enterprise 450	1 - 32 GB in 1 GB increments.
Memory allocator	This option specifies the memory allocation method used for the virtual machine. The default is **malloc**.	
	Allocator	**Description**
	malloc	All virtual machine RAM is allocated from system heap.
	hugetlb	All virtual machine RAM is allocate from hugetlbfs.
	mmap	All virtual machine RAM is allocated from file-backed virtual memory via mmap.

SCSI Configuration

To view or change the current virtual machine SCSI configuration, select **SCSI** in the **Device** column of the left-hand pane. This will open the SCSI configuration window, shown below.

Virtual Machine SCSI Configuration Window



From this window it is possible to create virtual disk and tape container files as well as attach virtual storage devices (both physical and container files) to the virtual machine.

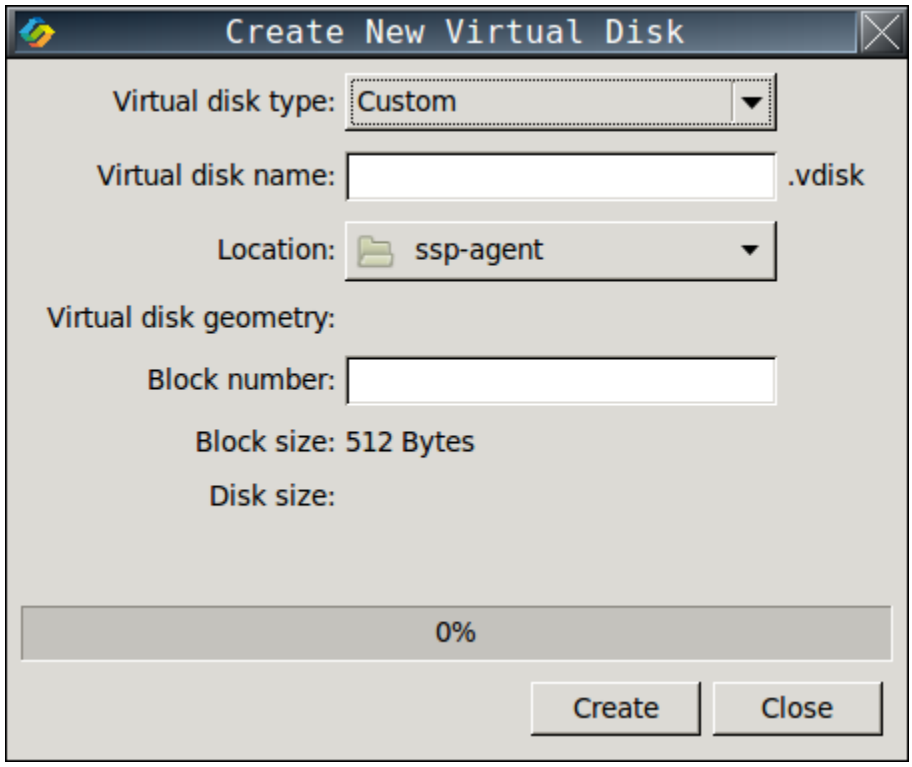
Creating Container Files

Often it is convenient to use container files for virtual disk and tape devices. The following sections describes creating both disk and tape container files.

Creating a New Virtual Disk Container File

To begin creating a virtual disk container file, **click** the button labelled **Create vdisk** in the SCSI device **Virtual Machine Settings** window. This will display the **Create New Virtual Disk** window shown below.

Create New Virtual Disk Window



To create a virtual disk container file, follow the instructions listed below.

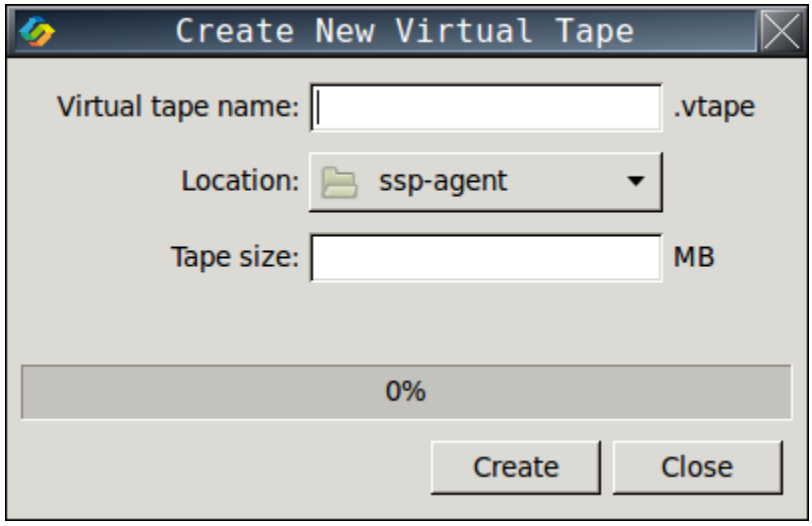
Steps for Creating a New Virtual Disk Container File

Step	Description
1.	Specify a name for the virtual disk container file in the field **Virtual disk name.**
2.	Select the location on the host file system for the container file by **clicking** the button **Location** and navigating to the correct path.
3.	Select the virtual disk type from the drop-down list **Virtual disk type.** • If specifying the type of **Custom**, enter the container file size as a number of 512 byte blocks at the field **Block number.** • If selecting an existing **Virtual disk type** the **Block number** field will be updated to match that model.
4.	**Click** the button labelled **Create** to generate the virtual disk container file. Depending on the size of the container file, this may take some time.

Creating a New Virtual Tape Container File

To begin creating a virtual tape container file, **click** the button labelled **Create vtape** in the SCSI device **Virtual Machine Settings** window. This will display the **Create New Virtual Tape** window shown below.

Create New Virtual Tape Windows



To create a virtual tape container file, follow the instructions listed below.

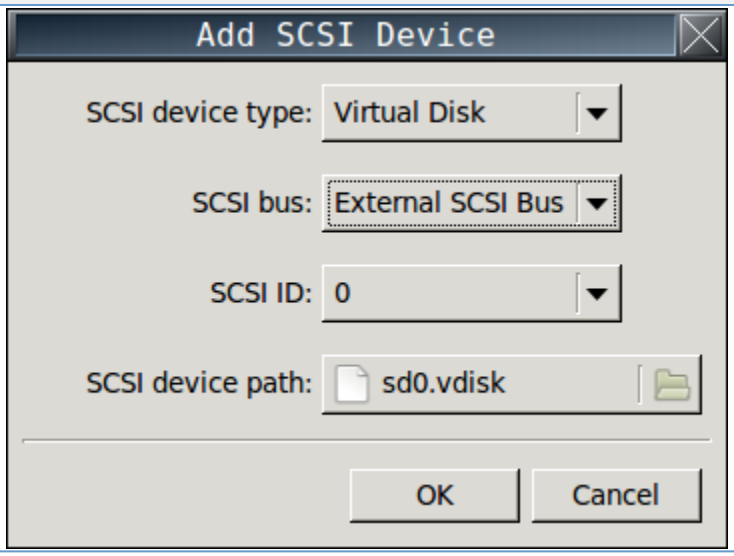
Steps for Creating a New Virtual Tape Container File

Step	Description
1.	Specify a name for the virtual tape container file in the field **Virtual tape name**
2.	Select the location on the host file system for the container file by **clicking** the button **Location** and navigating to the correct path.
3.	Specify the size for the virtual tape file in megabytes (MB) in the field **Tape size**.
4.	**Click** the button labelled **Create** to generate the virtual tape container file. Depending on the size of the container file, this may take some time.

Add/Edit Virtual SCSI Device

To add a new virtual disk device **click** the button labelled **Add...** To adjust an existing virtual disk device, select it from the list of configured devices and **click** the button labelled **Edit....** In both cases a widow similar to the one below will pop up to further configure the virtual disk device.

Add/Edit Virtual Disk Device Window (CHARON-SSP/4M shown)



The following table lists each of the fields in the **Add/Edit SCSI Device** configuration window and describes their operation.

Add/Edit Virtual Disk Device Configuration Fields

Field	Description	
SCSI device type	Drop-down list of configurable device types. The table below describes the available device types:	
	SCSI device type	**Description**
	Virtual Disk	Virtual disk device backed by a container file.
	Virtual CDROM	Virtual CDROM device, backed by a container file.
	Virtual Tape	Virtual tape device backed by a container file.
	Physical Disk	Virtual disk device connected to a host-attached physical disk device.
	Physical CDROM	Virtual CDROM device connected to a host-attached physical optical drive.
	Physical Tape	Virtual tape device connected to a host-attached physical tape drive.
SCSI bus	Specify either the **Primary SCSI Bus** and the **External SCSI Bus** on CHARON-SSP/4M virtual machines only.	
SCSI ID	SCSI device identification number. • For CHARON-SSP/4U-based virtual machines, acceptable values are a 3-bit narrow SCSI device ID between 0 and 7. • For CHARON-SSP/4M-based virtual machines, acceptable values are a 4-bit wide SCSI device ID between 0 and 15. In all cases the SCSI device ID 7 is reserved for the SCSI bus controller and cannot be used for a user configurable SCSI device.	
SCSI device path	**Click** the path to specify the location of the virtual SCSI device. The table below lists example device paths depending on the setting of the **SCSI device type** option.	
	SCSI Device Type	**Example SCSI Device Path**
	Virtual Disk	`/usr/local/vm/lela/scsi0.vdisk`
	Virtual CDROM	`/usr/local/share/iso/sunos_4.1.4.iso`
	Virtual Tape	`/usr/local/vm/lela/scsi1.vtape`
	Physical Disk	`/dev/sda`
	Physical CDROM	`/dev/sg1`
	Physical Tape	`/dev/st0`

Removing a Virtual Storage Device

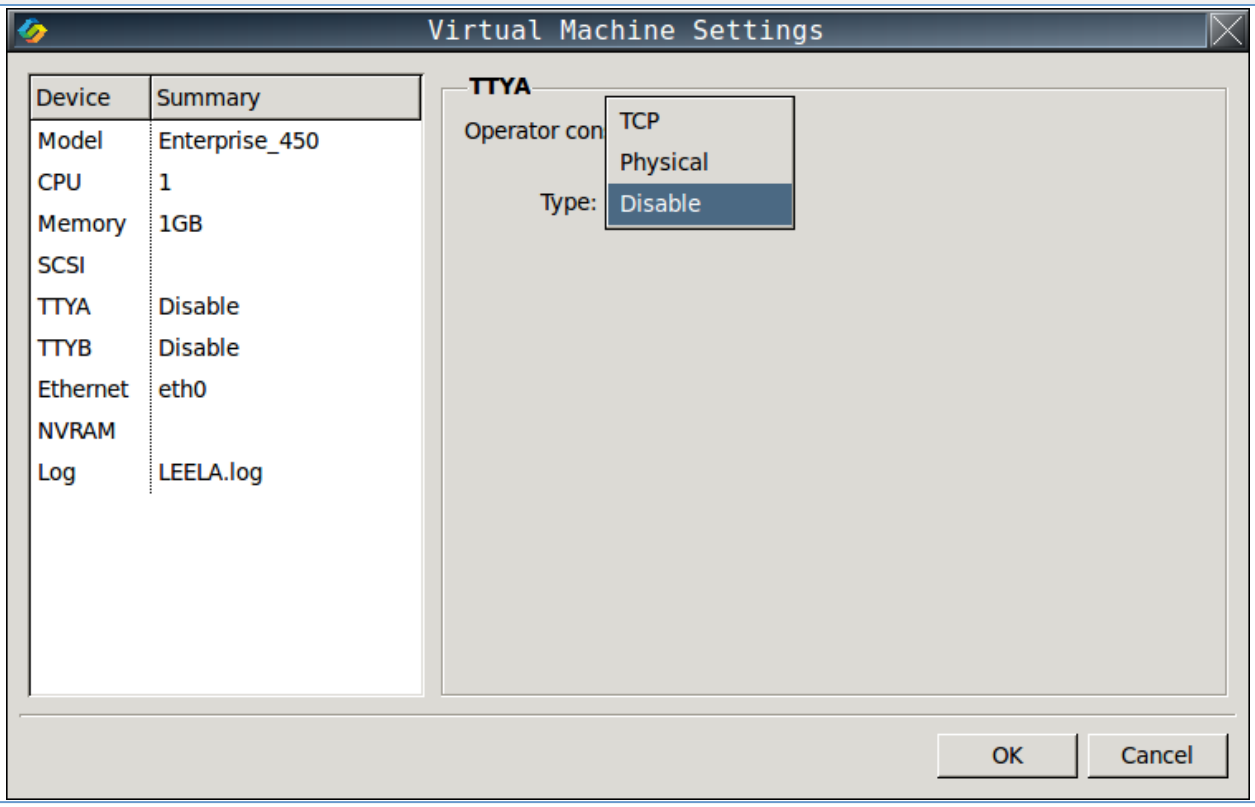
To remove a virtual storage device, select the device in the **Virtual Machine Settings** SCSI configuration window, then **click** the button labelled **Remove.** The device will be removed immediately and the management console does not ask for confirmation.

If the virtual SCSI storage device is attached to a container file the file itself is not removed when the configuration is.

TTYA Configuration

To view or change the current virtual machine console configuration, select **TTYA** in the **Device** column of the left-hand pane. This will open the **TTYA** configuration window, shown below. In this example the console is disabled.

Virtual Machine TTYA Configuration Window



It is possible to configure the virtual console device in one of three ways by selecting one of three options described below from the drop-down list labelled **Type**.

Virtual Machine Console Types

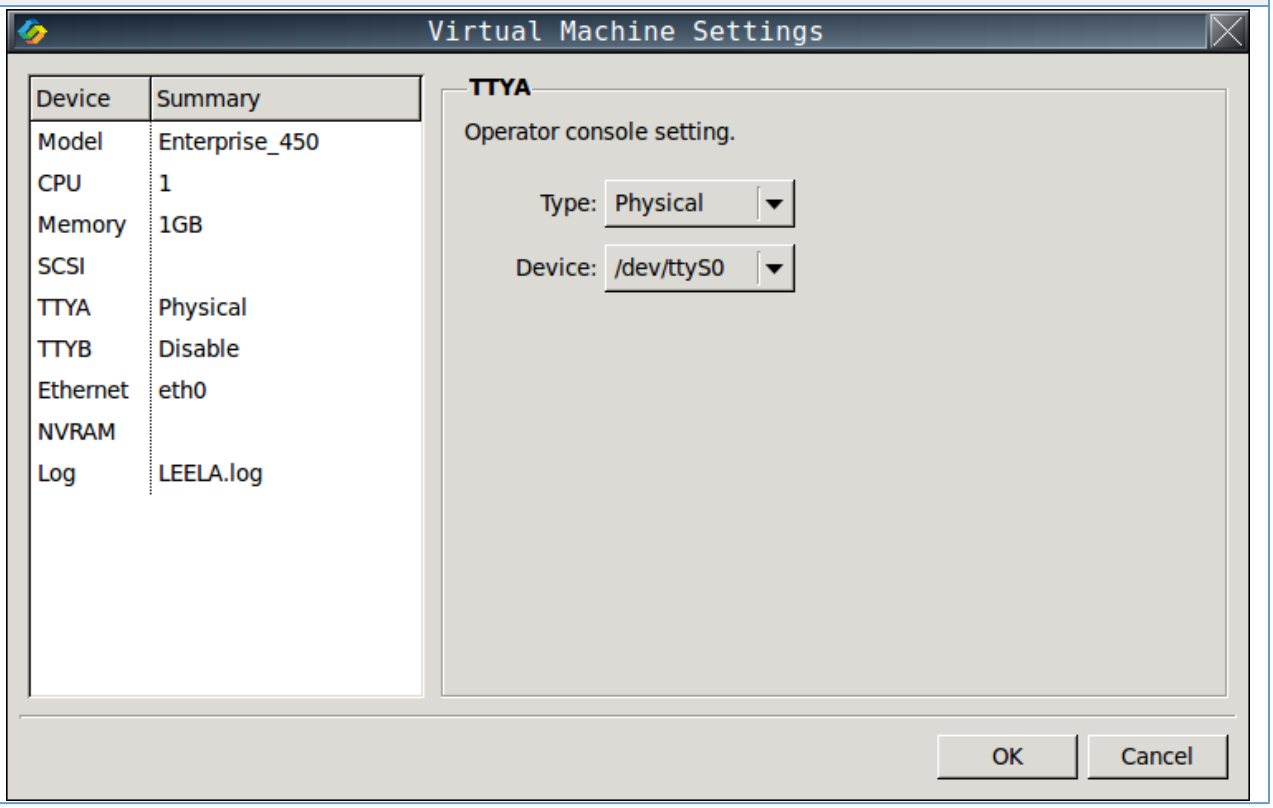
Type	Description
TCP	Configure the console device as a network device.
Physical	Configure the console device as physical terminal directly attached to the host system.
Disable	Disable the virtual console device entirely.

The following sections describe the specific configuration details of the physical and network consoles.

Physical Console Device

The image below shows the physical console device configuration window.

Virtual Machine TTYA Physical Device Configuration Window

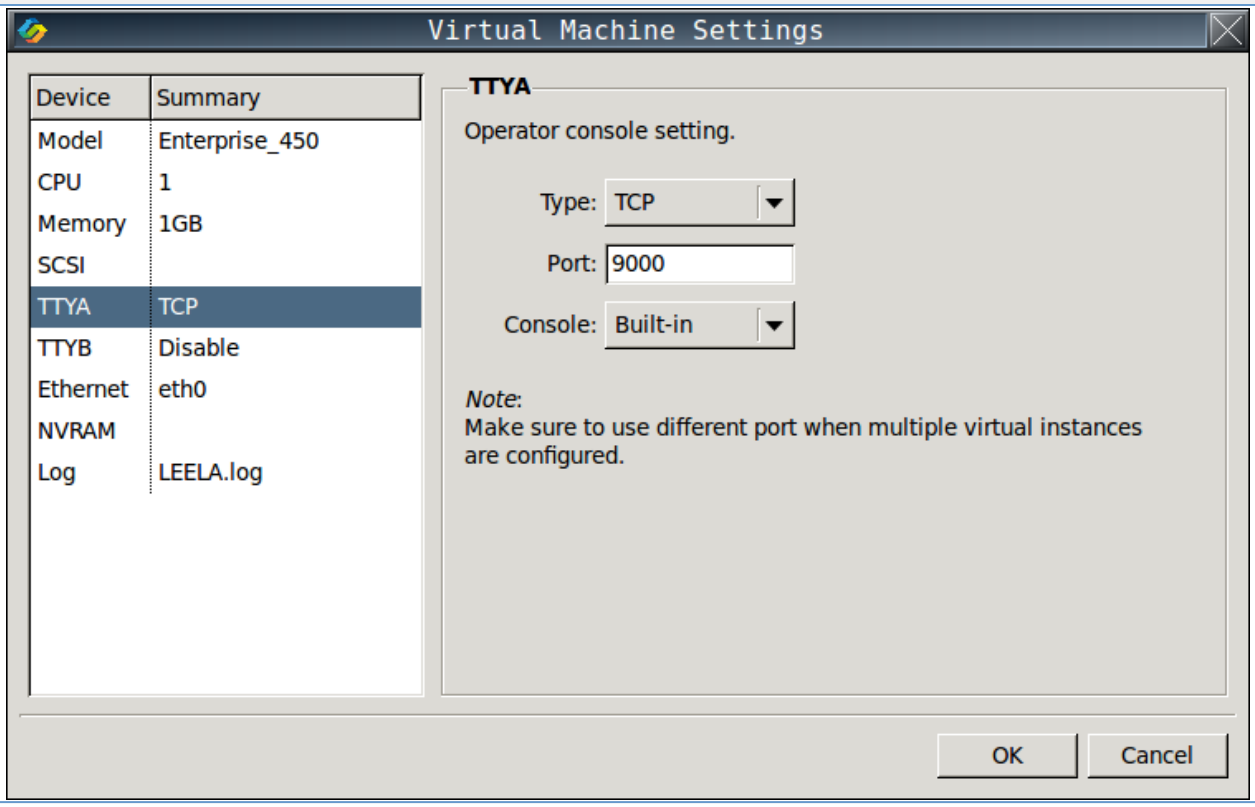


The only option for the physical terminal is **Device**. This is a drop-down list of all directly attached serial ports available on the host system.

Network Console Device

The image below shows the network console device configuration window.

Virtual Machine TTYA Network Device Configuration Window



The following table lists each of the fields in the network console device configuration window and describes their operation.

Virtual Machine TTYA Network Device Configuration Fields

Field	Description

Port	This option specifies the TCP/IP port to user when listening for incoming console client connections.
	A different port must be specified for each network console or serial port. Using a port that is already in use will result in the virtual machine log file.
	`2015-03-23 11:45:50 ERROR SocketIO Failed to open s`
	`2015-03-23 11:45:50 ERROR serial fail to init ser`
	`2015-03-23 11:45:50 ERROR vm Failed to initia`

Console	Specify in which way the network console will be viewed.

Console	Description
Built-in	The built-in console is displayed and accessible from the **Console** tab in the CHARON-AXP/SMA Manager.
External	An external network console device allows an external client (e.g. a telnet client) to attach to the port and inte

TTYB Configuration

To view or change the virtual machine TTYB configuration, select **TTYB** in the **Device** column of the left-hand pane.

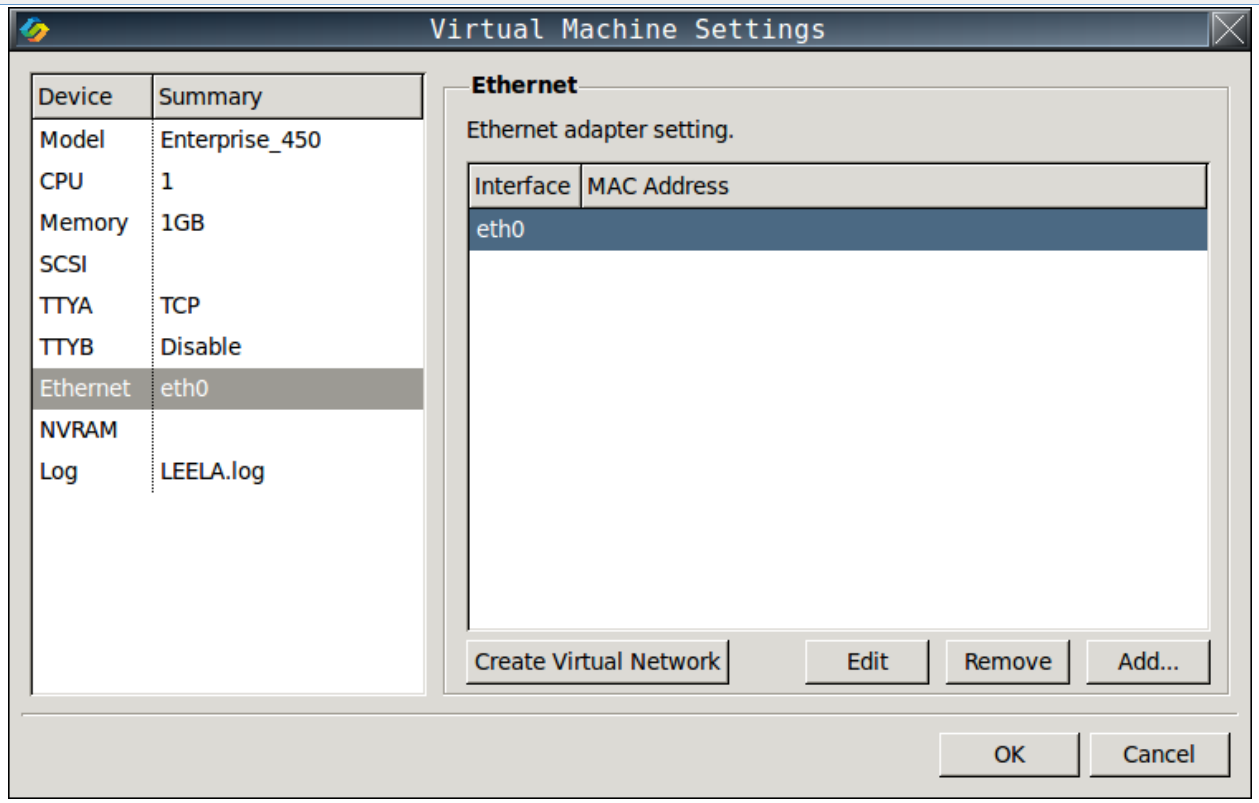
The virtual TTYB serial device can be configured as both a physical or network connected device. The configuration of this device is very similar

to TTYA. For further details related to configuring this device, consult the section TTYA Configuration.

Ethernet Configuration

To view or change the virtual machine Ethernet configuration, select **Ethernet** in the **Device** column of the left-hand pane.

Virtual Machine Ethernet Configuration Window

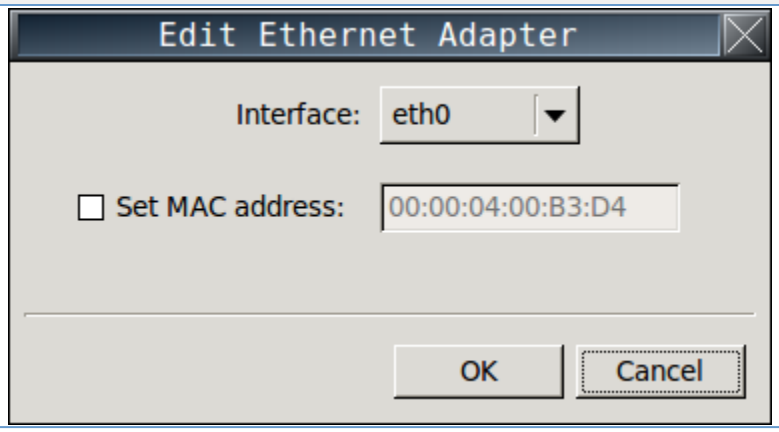


To remove an existing virtual Ethernet adapter, select the adapter from the list of configured devices and **click** the button labelled **Remove...**

To create a virtual network, **click** the button labelled **Create Virtual Network**. For further details on creating, changing and removing a virtual network ,see the section Managing Virtual Networks.

To add a new virtual Ethernet adapter **click** the button labelled **Add...** To adjust an existing virtual Ethernet adapter, select it from the list of configured devices and **click** the button labelled **Edit....** In both cases a widow similar to the one below will pop up to further configure the virtual Ethernet device.

Add/Edit Ethernet Adapter Window



The following table lists each of the fields in the **Add/Edit Ethernet Adapter** configuration window and describes their operation.

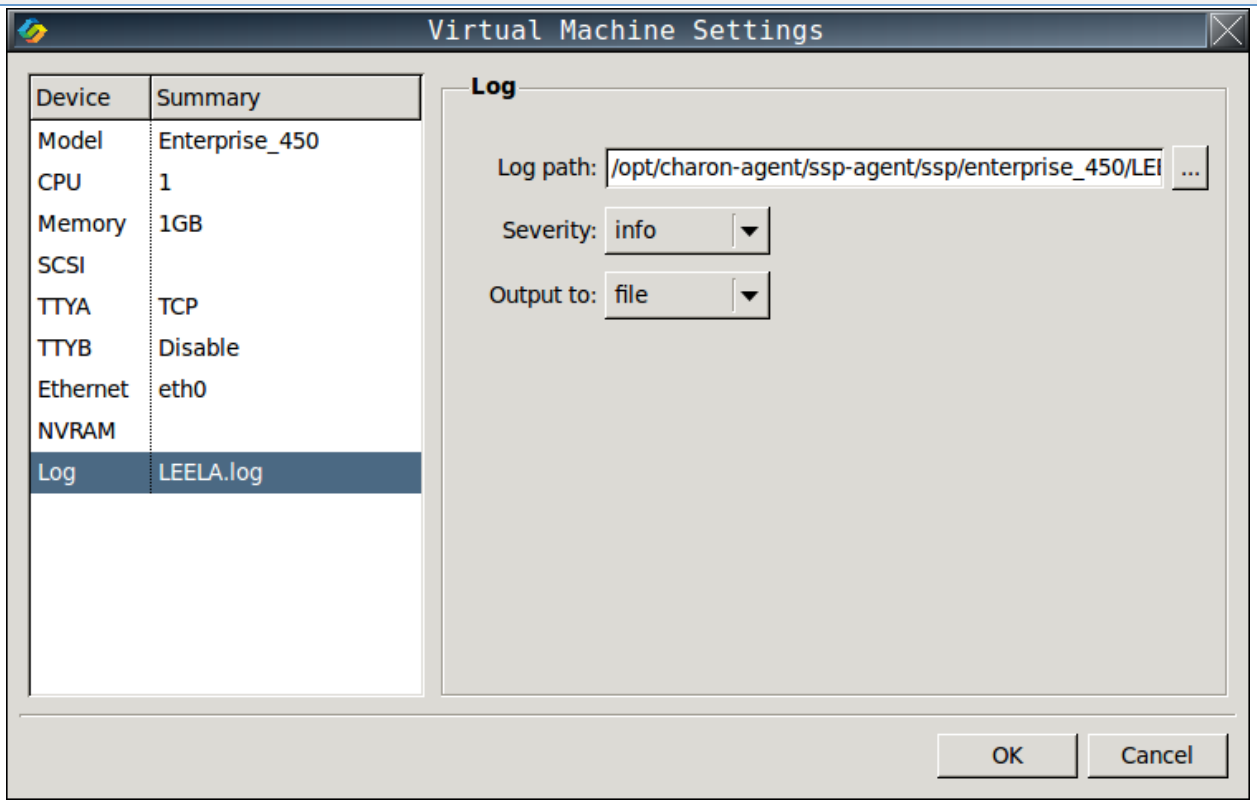
Virtual Ethernet Configuration Fields

Field	Description
Interface	Select the host attached Ethernet device to be connected to the virtual device. This field is a drop-down list of all the network adapters available on the host system.
Set MAC Address	To force the MAC address of the virtual Ethernet device to a specific value, select the checkbox and enter the address as size groups of two character hexadecimal digits, separated by a colon, e.g. 08:00:2b:aa:bb:cc. This option can be useful in the case where licensing is tied to a network adapter MAC address.

Log Configuration

To view or change the virtual machine logging configuration, select **Log** in the **Device** column of the left-hand pane.

Virtual Machine Log Configuration Windows



The following table lists each of the fields in the log configuration window and describes their operation.

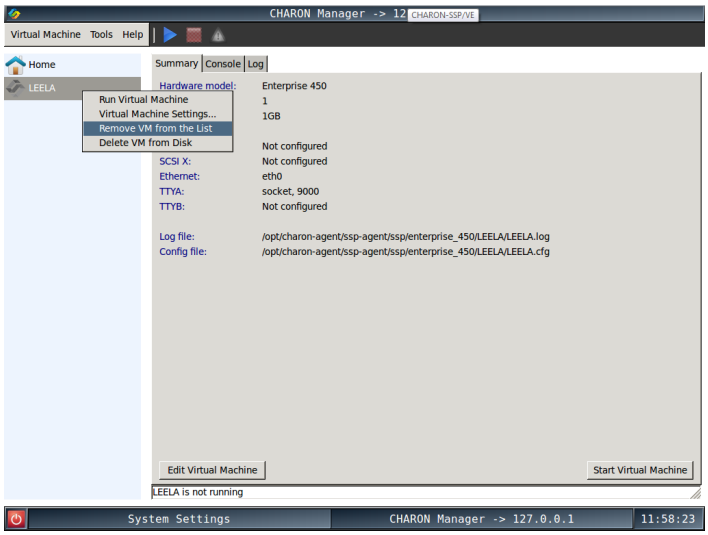
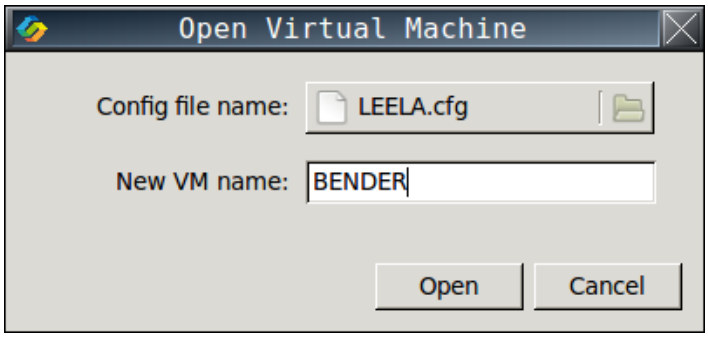
Virtual Machine Logging Configuration Fields

Field	Description	
Log path	Specify the path name for the log file.	
Severity	Set the minimum level of messages that should be reported. Legal values are debug, info, warning, error and fatal. The default is info.	
Output to	Indicate where virtual machine logging information should be written to. The default is file.	
	Option	**Description**
	file	Output virtual machine logging information only to the file configured in **Log path**.
	console	Output virtual machine logging information only to the virtual machine console.
	all	Output virtual machine logging information to both the file configured in **Log path** and the virtual machine console.

Renaming a Virtual Machine

To rename a virtual machine, follow the steps shown below.

Renaming a Virtual Machine

Step	Description
1.	**Click** the virtual machine to rename in the left hand pane.
2.	Take note of the name of the configuration file in the field **Config file**. This will be needed later in step 5.
3.	**Right-click** the virtual machine name and select **Remove VM from the List**.  This action will not ask for any confirmation and the virtual machine will be immediately removed from the list.
4.	**Click** the icon labelled **Open a Virtual Machine** on the main management console splash screen.
5.	A new window labelled **Open Virtual Machine** will appear. To continue: • **Click** the icon labelled **Config file name** • Navigate to configuration file saved from step 2 and **click** the button labelled **Open.**
6.	Enter the new name of the virtual machine in the field labelled **New VM name**. 
7.	**Click** the button labelled **Open**.

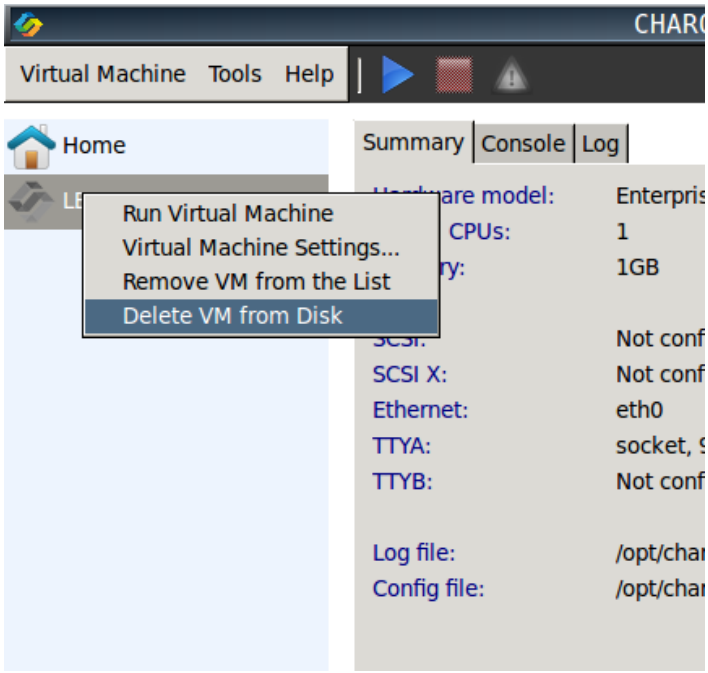
There is no confirmation and the newly renamed virtual machine will appear in the left-hand pane.

This process does not rename the configuration file, container files, log files or any other associated files. It simply changes the name of the virtual machine as it appears in the list.

Removing a Virtual Machine

The complete removal of a virtual machine must be performed in a number of steps. Each of these steps is detailed below.

Steps to Remove a Virtual Machine

Step	Description
1.	**Click** the name of the virtual machine that is to be removed in the left hand pane of the management console.
2.	• **Right-click** the name of the virtual machine in the left hand pane of the management console. • Select **Delete VM from Disk.**  The management console does not confirm this action and the configuration, log files and all associated container files are immediately deleted.
3.	Any container files (virtual disk or tape) that were part of the configuration are no longer on the system.

...

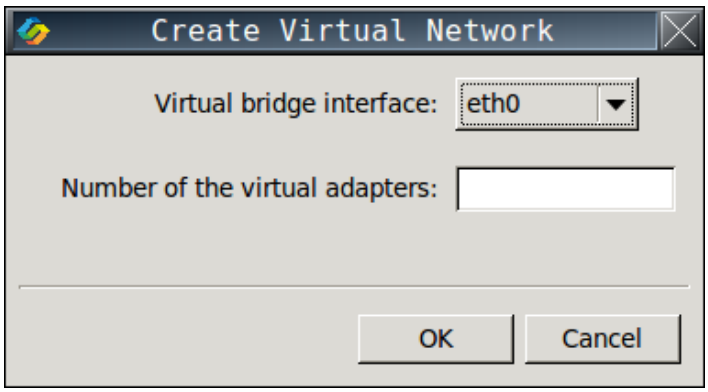
Managing Virtual Networks

Using the CHARON-SSP Agent in conjunction with the CHARON-SSP Manager it is possible to configure a collection of virtual network tap (TAP) devices that constitute a host-attached virtual LAN. These interfaces are bridged with a physical network device on the host system that allows the virtual LAN to connect to the physical LAN.

Creating a Virtual Network

To create a new virtual network, follow the instructions listed below.

Instructions for Creating a Virtual Network

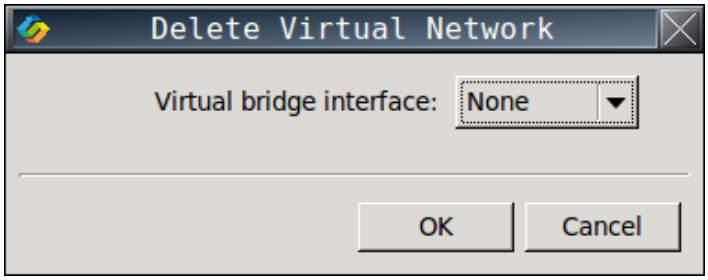
Step	Description
1.	Follow the menu path **Tools > Virtual Network > Create Virtual Network** to open the **Create Virtual Network** window, shown below. 
2.	Select the host adapter to bridge the virtual network devices with from the **Virtual bridge interface** drop-down list.
3.	In the field labelled **Number of the virtual adapters** enter the number of virtual network adapters to create.
4.	**Click** the button labelled **OK** to create the specified number of virtual network adapters.

The instructions detailed above will create a series of TAP devices, named `tap0` to `tap`*`N-1`*, where *N* is the value specified in the field **Number of the virtual adapters.** Theses devices can then be configured for use as virtual Ethernet controllers.

Deleting a Virtual Network

To delete a virtual network, follow the instructions listed below.

Instructions for Deleting a Virtual Network

Step	Description
1.	Follow the menu path **Tools > Virtual Network > Delete Virtual Network** to open the **Delete Virtual Network** window, shown below. 
2.	Select the physical interface serving the bridge from the **Virtual bridge interface** drop-down box.
3.	To delete **all** virtual network interfaces associated with this bridge, **click** the button labelled **OK.**

The instructions detailed above will immediately delete all TAP devices associated with physical interface selected in **Virtual bridge interface**.

Resizing a Virtual Network

To resize a virtual network, follow the instructions listed below.

Instructions for Resizing a Virtual Network

Step	Description
1.	Stop all virtual machine connected to the virtual network TAP devices.
2.	Delete the current virtual network, using the instructions detailed in Deleting a Virtual Network.
3.	Re-create the virtual network using the instructions detailed in Creating a Virtual Network. Ensure to specify the the new virtual network size in the field labelled **Number of the virtual adapters.**
4.	Reconfigure the Ethernet configuration of the virtual machines. This step is only necessary when shrinking the virtual network and only if they are configured for TAP devices that no longer exist.
5.	Start the attached virtual machines.

It is not recommended to shrink a virtual network as this can mean potentially having to adjust a number of virtual machine configurations.

Managing Licenses

The process of managing CHARON-SSP for Linux is described in full detail in the section Managing Licenses with CHARON-SSP Manager. For further complete information on additional command-line utilities and the SafeNet Sentinel Admin Control Center web-interface, see the section License Management.

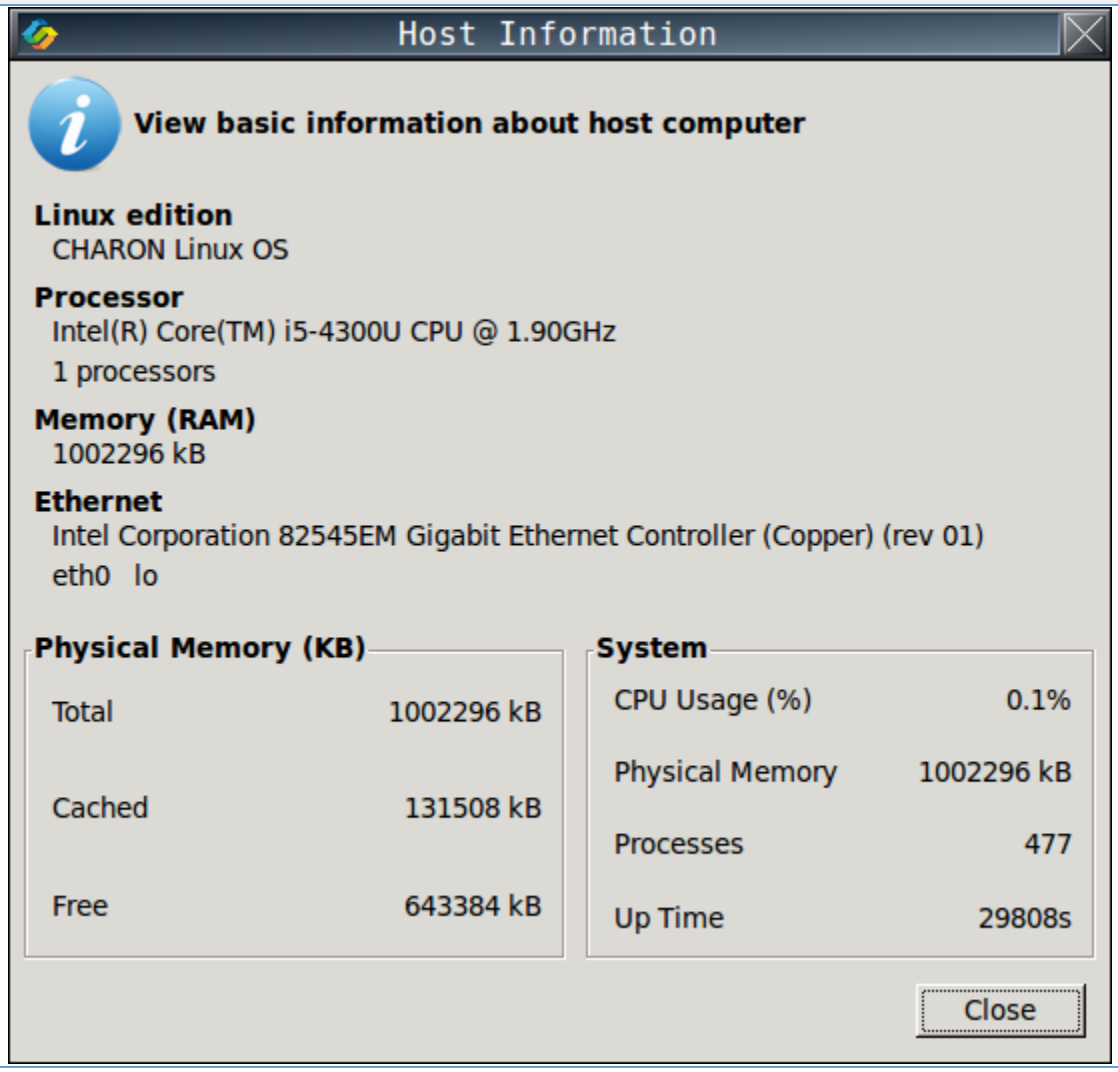
Miscellaneous Management Tasks

The following sections describe some other miscellaneous functionality provided by the CHARON-SSP Manager interface that may be useful in certain instances.

Gathering Host Information

To view the details of the system hosting the CHARON-SSP instance, follow the menu path Tools > **Host Information...** to open a window similar to the one below.

Host Information Window



This window provides details of the host system's hardware configuration and the operating system it is running on.

Determining the CHARON-SSP Manager Version

To determine the version of CHARON-SSP Manager currently running, follow the menu path **Help** > **About...** to open a window similar to the one shown below.

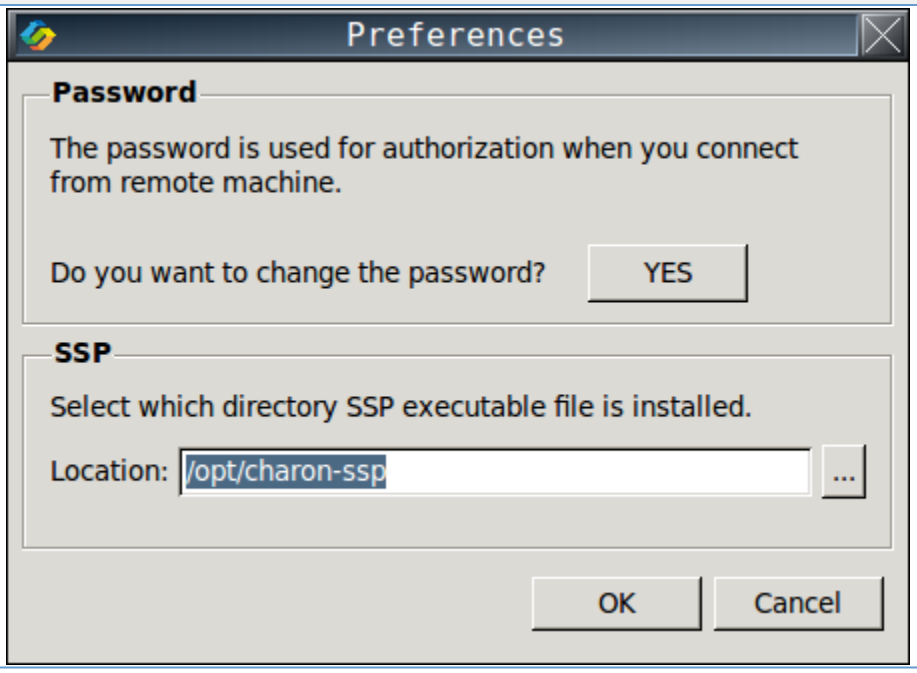
About CHARON Manager Windows

The example above shows the **About CHARON Manager** window for CHARON-SSP Manager version V1.0.13.

Modifying the CHARON-SSP Agent Preferences

To modify the preferences maintained by the CHARON-SSP Agent software, follow the menu path **Virtual Machine** > **Preferences...** to open a window similar to the one shown below.

Preferences Window



The CHARON-SSP Manager password can be modified by **clicking** the button labelled **YES**, adjacent to the label **Do you want to change the password?**

It is possible to alter the root location of the CHARON-SSP executables. This might be useful for example, if multiple versions of CHARON-SSP have been installed, this can be used to switch between them. To change the root location, alter the pathname in the field **Location**.

Using CHARON-SSP from the Command Line

Name

ss20 – CHARON-SSP/4M 32-bit SPARC V8, sun4m Architecture Virtual Machine

e450 – CHARON-SSP/4U 64-bit SPARC V9, sun4u Architecture Virtual Machine

Synopsis

```
ss20 | e450 [ [ -a alias name ] [ -d ] -f config file [ -s ] ] [ -h ] [ -k pid ] [ -l ]
```

Description

The sma utility is the lowest level access to the CHARON-SSP virtual machine software. It can be started in four different modes:

- Utility,
- Foreground,
- Background, and
- Interactive.

In utility mode it is possible to specify the **-l** and **-k** options to list and terminate running instances. In both foreground and background mode, the **sma** software can be used to start a running SPARC virtual machine. The difference between the two being that background mode will run the virtual machine as a daemon, releasing the controlling terminal. Lastly, interactive mode will run a virtual machine as well as start an interactive command line described in the section Interactive Mode

The following table describes all the options that can be passed to the **sma** command line utility.

sma Command Line Utility Options

Option	Description
-a alias name	Assign alias name to the new virtual machine instances. This option can be very useful when attempting to locate a specific instance in the list reported by the **-l** option. If this option is not specified, a name of the form `instance-%02d` will be assigned by the `sma` utility.
-d	Run the virtual machine as a daemon. This option cannot be specified with **-s**.
-f config file	When starting a new instance, use this option to specify the location of the virtual machine configuration file. For further details of the format of this file see the Configuration Reference section. This is not an optional argument.
-s	Start the `sma` virtual machine in interactive mode. See the section Interactive Mode below. This option cannot be specified with **-d**.
-h	Display a brief usage message.
-k pid	Stop the virtual machine instance specified by pid. Use the **-l** to determine the process id of the relevant instance. This option cannot be specified with an others.
-l	This option lists the currently running CHARON-SSP instances. The list is constructed of the following columns: • pid – process id of the virtual machine, • alias – instance alias specified by the `-a` option at startup, • start time – timestamp indicating when the virtual machine instance was started, • log time – timestamp indicating last event, and • log code – descriptive code indicating the type of event. This option cannot be specified with any others.

Interactive Mode

Running the **sma** virtual machine with the **-s** option will, as well as starting a virtual machine, start an interactive command line on the controlling terminal. This command line is indicated by the **CHARON/SS20>** or **CHARON/E450>** prompt and allows the user to query and manipulate the operation of the running virtual machine. The terminal output shown below shows the tail end of the virtual machine start up messages before displaying the prompt.

Interactive Mode Session

```
2015-02-25 15:59:31 INFO    vm       DIT is ON
2015-02-25 15:59:31 INFO    Thread   System has 2 processor(s).
2015-02-25 15:59:31 INFO    NVRAM    Initialize NVRAM with ./vm.nvram......
2015-02-25 15:59:31 INFO    Memory   Allocating 67108864 bytes memory from system...
2015-02-25 15:59:32 INFO    NET      Find 2 interfaces.
2015-02-25 15:59:32 INFO    NET      Connected to eth0.
2015-02-25 15:59:32 INFO    NET      Set 'eth0' to promiscuous mode successfully
2015-02-25 15:59:35 INFO    License  Virtual hardware model SPARCstation_20 is licensed
2015-02-25 15:59:35 INFO    License  Current UTC time:    2015-02-25 07:59:03
2015-02-25 15:59:35 INFO    License  Expiration UTC time: 2015-08-26 23:55:00
2015-02-25 15:59:35 INFO    Thread   Set CPU thread to No.1 processor
2015-02-25 15:59:35 INFO    NET      Network RX started on eth0
CHARON/SS20>
```

Interactive Mode does not refer to running the SPARC Open Boot console in the current tty.

The following command reference describes the syntax and operation of the commands supported in this version of CHARON-SSP for Linux.

Interactive Mode Command Reference

Command	Description	
help [*command*]	Display command line help.	
	Arguments	
	command	Specify a command name to return more specific help. This argument is optional.
d [*address size*]	Display a region of memory within the virtual machine. If no parameters are specified the command will use the previous values for *address* and *size*.	
	Arguments	
	address	Hexadecimal address of starting memory location to display.
	size	Hexadecimal length of the memory region. The default is `0x100`.
m *address value*	Set the 8-bit value at *address* to *value*.	
	Arguments	
	address	Address of memory location to update. This address is specified in hexadecimal notation.
	value	8 bit value to be stored at `address`. This value is specified in hexadecimal notation.
quit	Exit the virtual machine. This is equivalent of executing **power off** from the Alpha SRM console.	

Exit Status

The **ss20** and **e450** virtual machine exits **0** on success and **255** if an error occurs.

Examples

The following is a very basic example configuration file.

Example SPARCstation 20 Configuration

```
[system]
model = "SPARCstation_20"

[ram]
size = 64

[nvram]
path = ./vm.nvram

[ethernet]
interface = eth0

[ttya]
type = socket
port = 9000

[log]
severity = info
destination = console
path = vm.log
```

Assuming the configuration file above is stored in vm.config the following commands can be used to start the virtual machine and connect to the console.

```
# /opt/charon-spk-ss20/ss20 -f vm.config
```

This will generate output similar to the following on the current terminal:

CHARON-SSP Virtual Machine Output

```
**********************************************************

   CHARON-SPK/SS20 V1.0.13
   Copyright (C) 2015 Stromasys S.A. All Rights Reserved.

**********************************************************

2015-02-25 14:50:50 INFO    vm       Build time: Feb 11 2015 11:49:38
2015-02-25 14:50:50 INFO    vm       Version: CHARON-SPK/SS20 V1.0.13
2015-02-25 14:50:50 INFO    vm       Copyright (C) 2015 Stromasys S.A. All Rights Reserved.
2015-02-25 14:50:50 INFO    System   OS  : Linux localhost.localdomain 3.8.13-35.3.1.el7uek.x86_64
2015-02-25 14:50:50 INFO    System   CPU : Intel(R) Core(TM) i5-4300U CPU @ 1.90GHz (running under
2015-02-25 14:50:50 INFO    System   Total memory: 3683MB
2015-02-25 14:50:50 INFO    System   Free memory : 227MB
2015-02-25 14:50:50 INFO    vm       DIT is ON
2015-02-25 14:50:50 INFO    Thread   System has 2 processor(s).
2015-02-25 14:50:50 INFO    NVRAM    Initialize NVRAM with ./vm.nvram......
2015-02-25 14:50:50 INFO    Memory   Allocating 67108864 bytes memory from system...
2015-02-25 14:50:50 INFO    NET      Find 2 interfaces.
2015-02-25 14:50:50 INFO    NET      Connected to eth0.
2015-02-25 14:50:50 INFO    NET      Set 'eth0' to promiscuous mode successfully
```

The following shows the SPARC Open Boot console after the virtual machine has started:

Open Boot Console Ouptut

```
           SMCC SPARCstation 20 Emulator by Stromasys


CPU_#0       TI, TMS390Z50(3.x)       0Mb External cache

CPU_#1       ****** NOT installed ******
CPU_#1       ****** NOT installed ******
CPU_#1       ****** NOT installed ******


           >>>>> Power On Self Test (POST) is running .... <<<<<

SPARCstation 20 (1 X 390Z50), No Keyboard
Emulate OBP Rev. 2.25, 64 MB memory installed, Serial #12648430.
Ethernet address 2:c:29:4a:d3:29, Host ID: 72c0ffee.


Type  help  for more

ok
```

Using the CHARON-SSP Agent

The CHARON-SSP Agent is a Linux service that runs on a system that will be managed by the CHARON-SSP Manager. This service provides the interface between the CHARON-SSP Management GUI and the virtual machine software. A prerequisite of installing the CHARON-SSP Agent is that the CHARON-SSP for Linux virtual machine software must already be installed.

The following sections describe how to manipulate the CHARON-SSP Agent software.

> The default installation of CHARON-SSP Agent can be insecure. To ensure the most secure environment for the CHARON-SSP Agent and the virtual machines it interfaces to, please complete the post-installation tasks documented in the section Installing the CHARON-SSP Agent.

Starting the Agent Service

The CHARON-SSP Agent service can be started by executing the following command from a privileged account:

```
# service charon-agentd-ssp start
```

Stopping the Agent Service

The CHARON-SSP Agent can be stopped by executing the following command from a privileged account:

```
# service charon-agentd-ssp stop
```

TCP/IP Ports Used by CHARON-SSP Agent

Communication between the CHARON-SSP Agent and Manager is achieved over the TCP/IP port 9091. To ensure correct operation, the CHARON-SSP Agent must be able to bind to port **9091/tcp** and the CHARON-SSP Manager must be able to connect to the same port.

Accessing the Console

Depending on how the virtual machine console device has been configured, it is possible to access the console in a number of different ways. The sections listed below describe example configurations and other useful notes detailing how set up and use the different console access methods.

For complete reference on all configuration options related to accessing the virtual serial console device, see the section [ttya] Section of the Configuration Reference and the section Managing Virtual Machines.

Physical Console Access

For physical console access, the virtual machine must be configured to attach the virtual serial port to a physical serial port on the host system. The following configuration extract demonstrates how to attach the virtual serial port to the host device, `/dev/ttyS0`.

Configuratin ttya for Physical Console Access

```
# Virtual serial console attached to host device /dev/ttyS0.
[ttya]
type = physical
port = /dev/ttyS0
```

The remaining serial port configuration options, such as speed, parity and stop-bits must be configured using the `ttya-mode` variable in the Open Boot guest environment. The following example shows the default configuration values for `ttya-mode`.

Printing Configuration of `ttya-mode` Console Variable

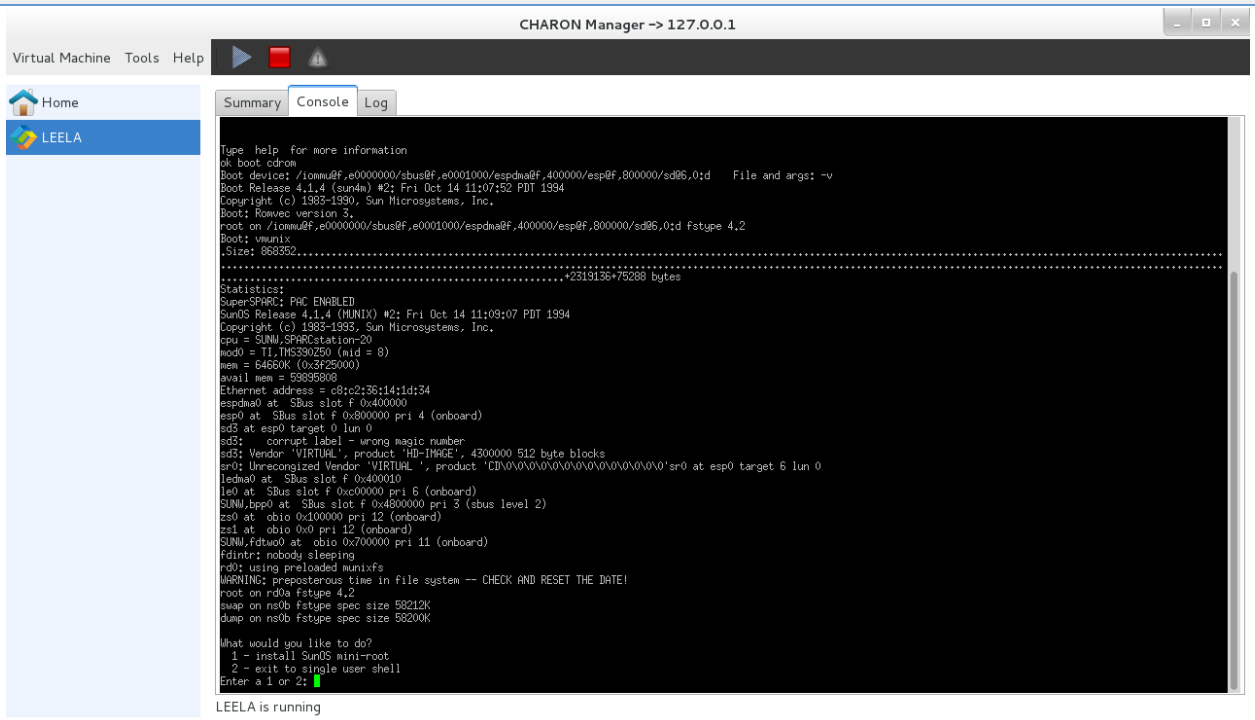
```
ok printenv
printenv
Parameter Name        Value                          Default Value
auto-boot?            false                          true
boot-file             -v
boot-device           disk:a disk1                   disk net
ttya-mode             9600,8,n,1,-                   9600,8,n,1,-
ttyb-mode             9600,8,n,1,-                   9600,8,n,1,-
```

For a complete reference on how to configure the serial options in Open Boot, see the [ttya] Section of the Console Reference chapter.

Console Access from CHARON-SSP Manager

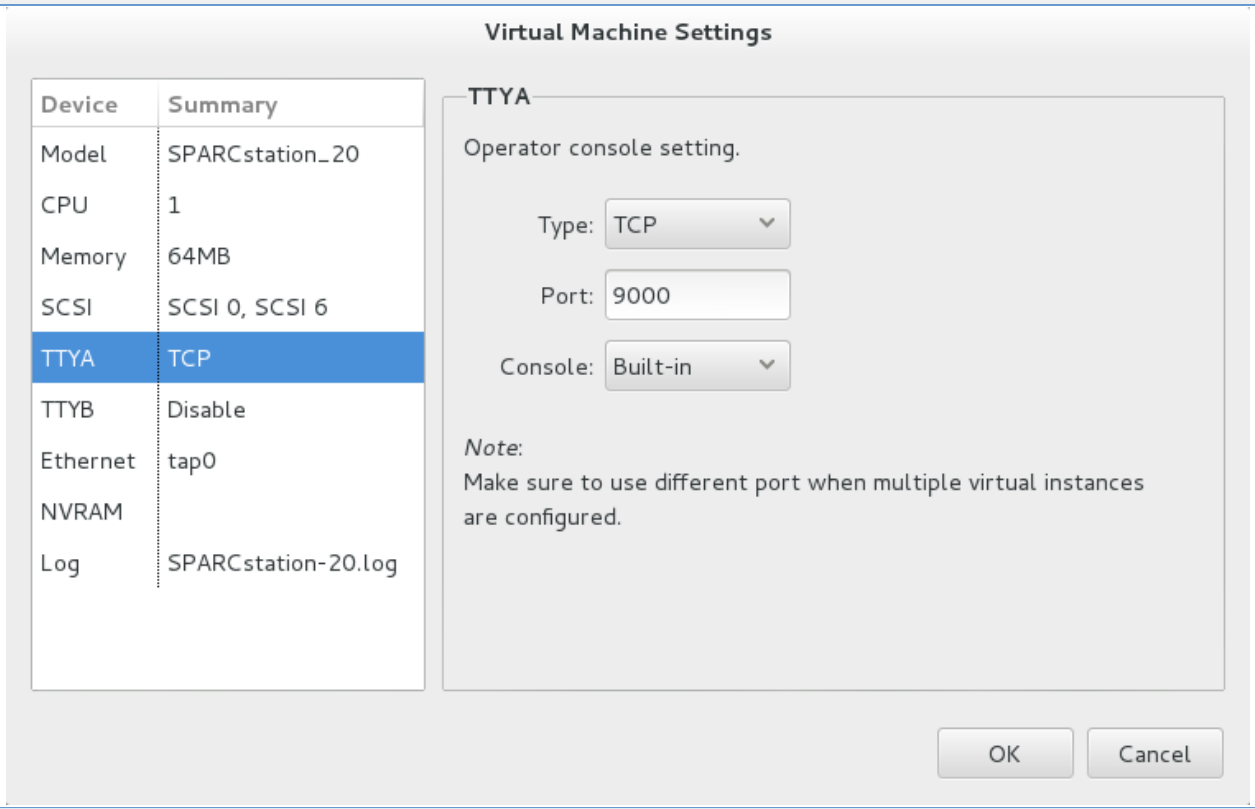
From the CHARON-SSP Manager it is possible to access the serial console, via the **Console** tab. The example below shows the console of a SPARCstation 20 that has booted the SunOS 4.1.4 installation CDROM.

CHARON-SSP Manager Console Tab



To configure the serial console to be accessed from the CHARON-SSP Manager, use a configuration for the TTYA section similar to the one below.

Configure TTYA for CHARON-SSP Manager Console Access



Console Access from the Network

The serial console can also be configured for remote access across the network. The configuration extra below demonstrates configuring the guest console, `ttya` to accept incoming connections from port `9000/tcp` on the host.

Configuring ttya for Network Console Access

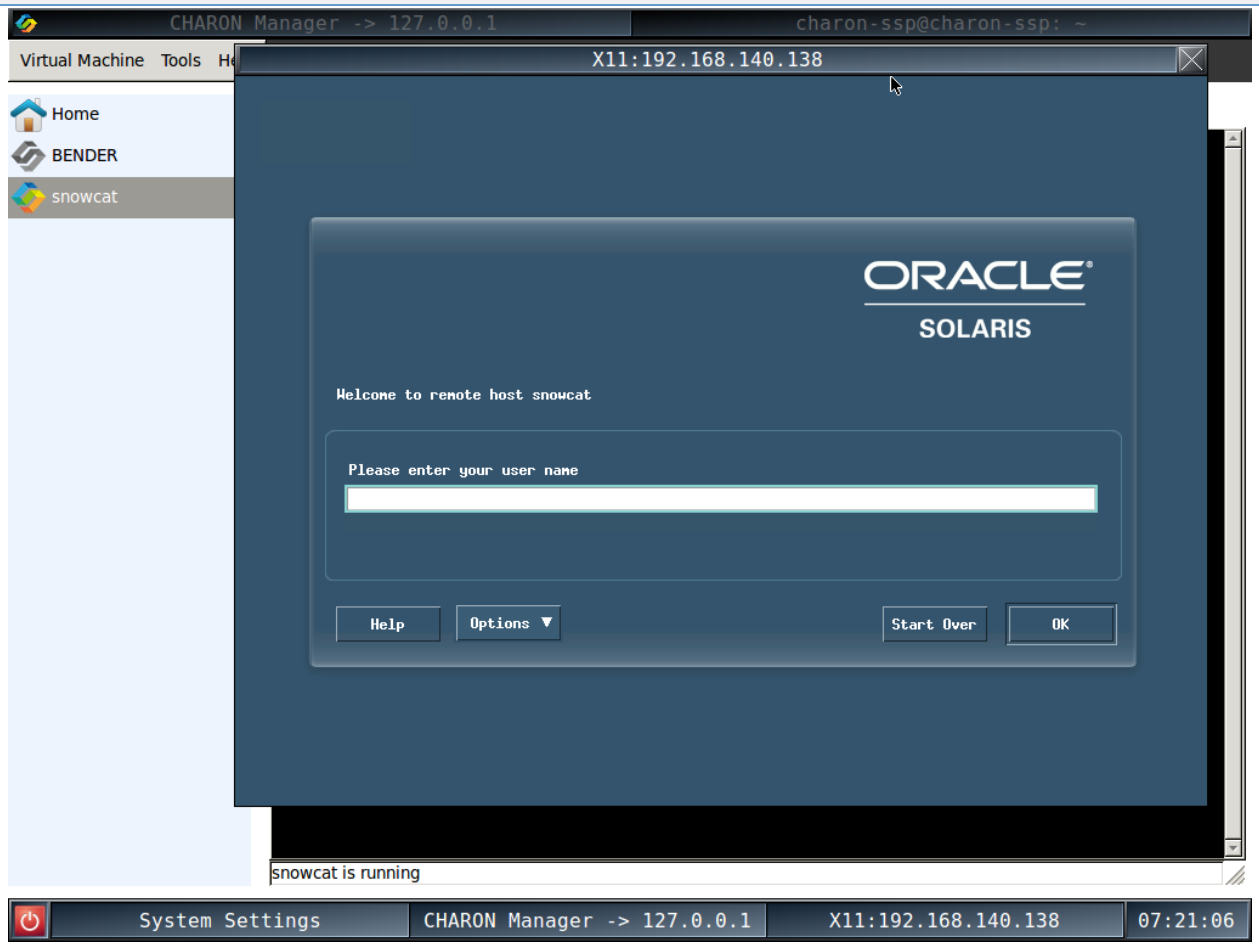
```
# SPARCstation 20 serial console redirected to network
# port 9000/tcp.
[ttya]
type = socket
port = 9000
```

At this point, any terminal emulator with telnet capability can be used to connect to the port and access the guest system's serial console.

Graphical Console Access

Although not a graphical console, per-se, the CHARON-SSP Manager has the ability to connect to an X11 login session using the XDMCP protocol. The screenshot below shows an X11 session from CHARON-SSP Manager to a guest running Oracle Solaris 10.

Example X11 Session to a Guest Runing Solaris 10



Enabling XDMCP

Before switching to the Xserver, it is first necessary to ensure that XDMCP is enabled on the guest system. The actions for enabling XDMCP are different, depending on the version of Solaris or SunOS installed on the guest. Follow the relevant sub-section below to configure XDMCP on your guest.

Solaris 2.5 and higher

Use the following instructions to enable remote login over XDMCP on Solaris 9 and lower.

Enable Remote XDMCP Login on Solaris 9 and Lower

Step	Description
1.	Edit the file `/usr/dt/config/Xconfig` `# vi /usr/dt/config/Xconfig`
2.	Locate the following line and insert a comment character, '#', at the beginning of the line. `Dtlogin.requestPort: 0`
3.	Save the configuration file and restart the Xserver: `# /etc/init.d/dtlogin restart`

Solaris 10 and HIgher

Use the following commands to enable remote login over XDMCP on Solaris 10 and higher.

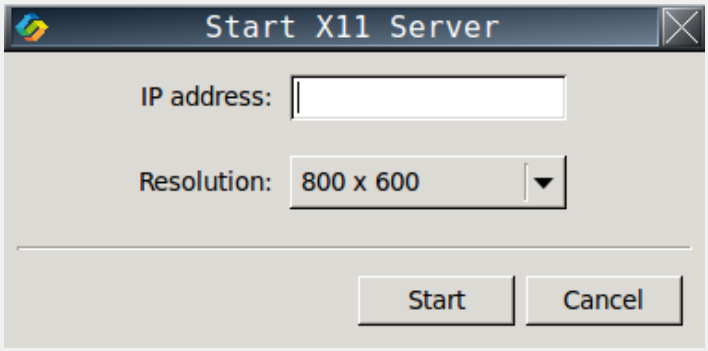
Enable Remote XDMCP Login on Solaris 10 and Higher

```
# svccfg -s cde-login setprop 'dtlogin/args=""'
# svcadm restart cde-login
```

Starting the Xserver

Once XDMCP has been enabled on the guest, use the following instructions to start the Xserver display.

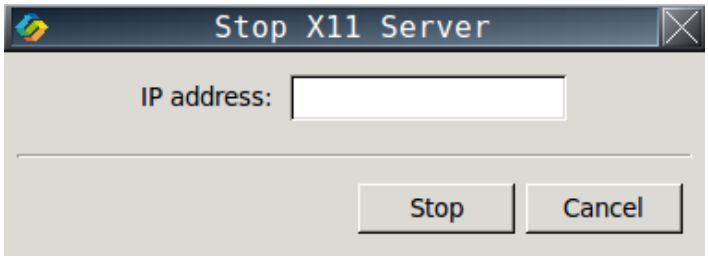
Instructions for Starting the X11 Server

Step	Description
1.	Open the **Start X11 Xserver** window (shown below) from CHARON-SSP Manager by following the menu path, **Tools > X11 Server > Start X11 Server**. 
2.	Configure the X11 Server by completing the fields: • Enter the address or name of the guest in the field **IP address.** • Select the X11 session resolutions from the **Resolution** drop down box.
3.	**Click** the button labelled **Start** to start the X11 server.

Stopping the Xserver

To stop the X11 Server, follow the instructions below.

Instructions for Stopping the X11 Server

Step	Description
1.	Open the **Stop X11 Xserver** window (shown below) from CHARON-SSP Manager by following the menu path, **Tools > X11 Server > Stop X11 Server**. 
2.	Enter the address or name of the guest into the field **IP address**.
3.	**Click** the button labelled **Stop** to termination the X11 session. If multiple sessions to the same host are open, it will be necessary to repeat these steps for each session.

License Management

All Stromasys CHARON virtual machine software products are licensed using a Sentinel HASP (Hardware Against Software Piracy) key. To use these products you must have both the physical key and a valid license file loaded on to it. The following CHARON-SSP products require this license key to operate:

- CHARON-SSP/4M for Linux
- CHARON-SSP/4U for Linux
- CHARON-SSP Virtual Environment

The following products work with the licensed products, but do not require the HASP a license themselves:

- CHARON-SSP Manager for Linux
- CHARON-SSP Manager for Windows
- CHARON-SSP Agent for Linux

Licensing CHARON-SSP for Linux

It is possible to gather license information and apply license keys using a collection of tools. These different tools provide both command line and GUI interfaces to managing the licenses for CHARON-SSP products. They also allow for licenses to be managed either locally on the host system or remotely from a client (running either GNU/Linux or Microsoft Windows).

All licensing techniques follow the same basic four steps:

1. Generate a C2V (customer to vendor) key information file;
2. Submit the C2V information to Stromasys Orders Administration;
3. Receive one or more V2C (vendor to customer) license key files; and
4. Apply the V2C files to the local system.

The following sections describe how to apply each of these steps using the different tools available.

> For customers licensed with USB HASP keys; please note these keys contain a built-in battery, which must not be completely discharged. It is recommended that unused keys are connected to spare USB ports from time to time for charging. In the event that a key fails, do not discard the key. Please contact Stromasys Customer Support immediately.

Managing Licenses with CHARON-SSP Manager

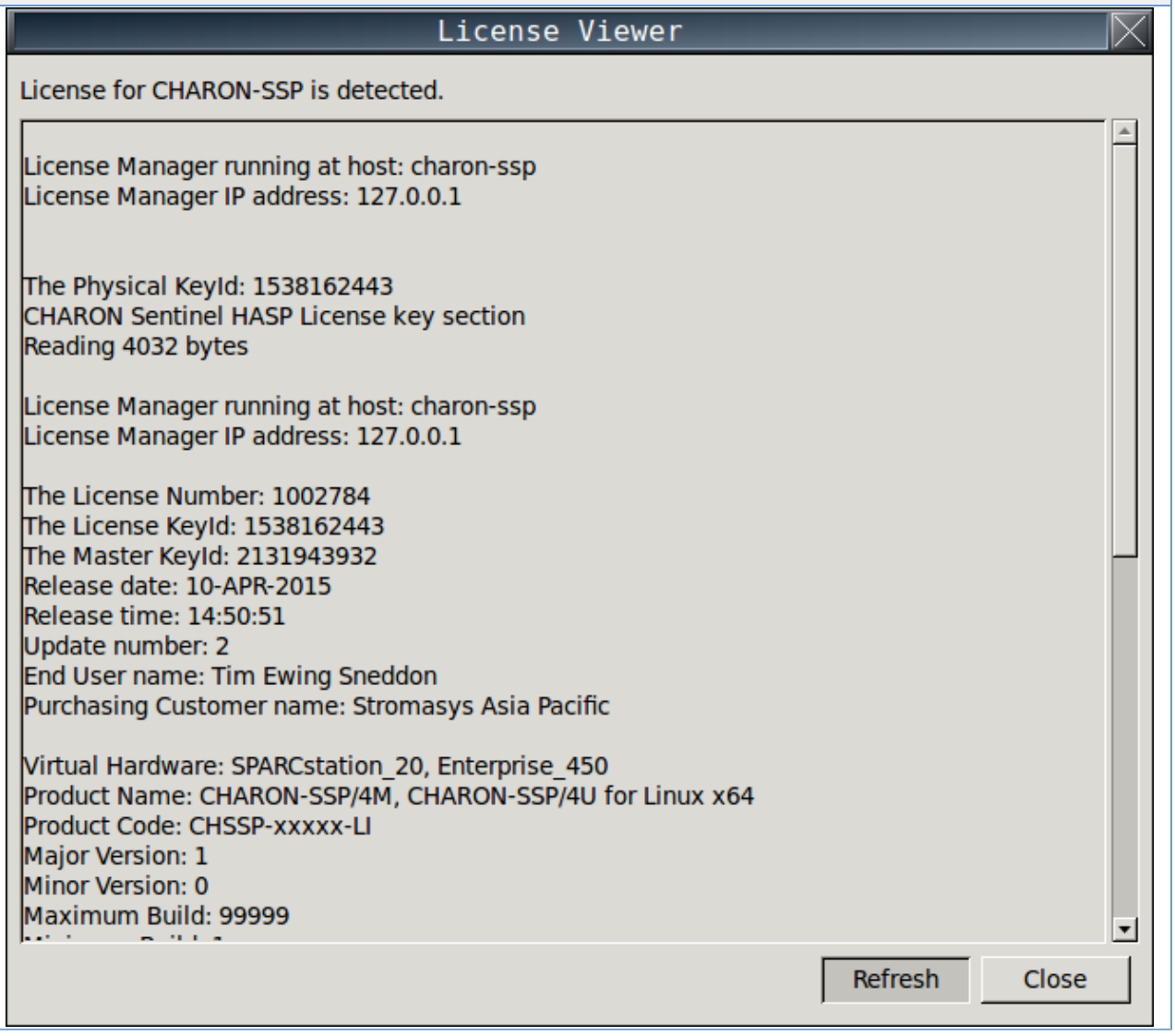
The CHARON-SSP Manager provides a friendly graphical interface for the management of licenses. The following sections describe how to use these tools:

- Viewing the License Details
- Gathering Customer to Vendor (C2V) Detail
- Applying Vendor to Customer (V2C) License Update

Viewing the License Details

To view the license details of the attached USB HASP **click** the menu path **Tools > License Viewer...** This will load a **License Viewer** window, similar to the following.

CHARON-SSP Manager License Viewer



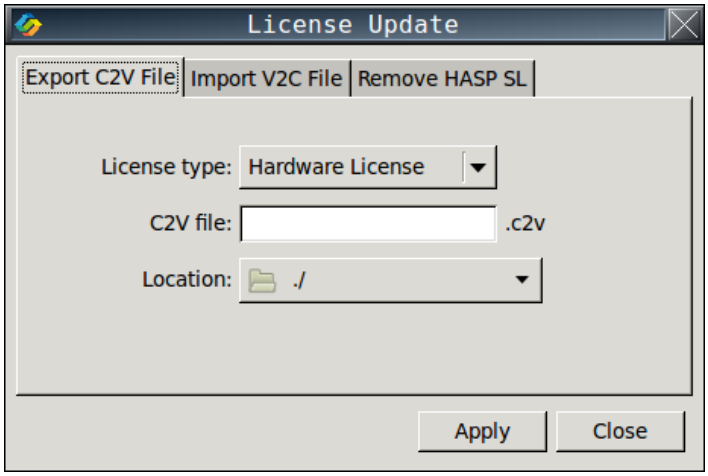
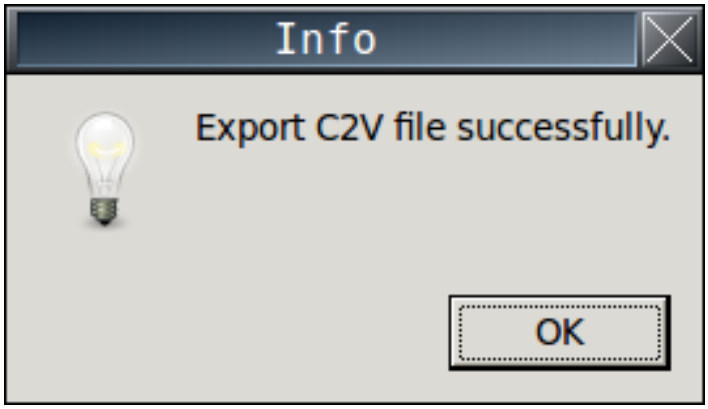
If the license detail does not display and the USB HASP has recently been disconnected or exchanged, it may be necessary to **click** the button labelled **Refresh**.

To exit the window, **click** the button labelled **Close**.

Gathering Customer to Vendor (C2V) Detail

The following steps describe the process of gathering the customer to vendor (C2V) data file. This information is used by Stromasys to generate a license data file

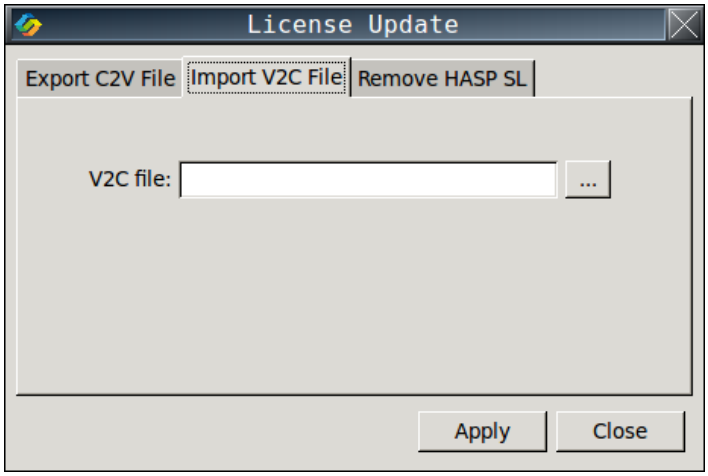
Instructions for Gathering Customer to Vendor (C2V) Data

Step	Description
1.	Open the **License Update** window, click the menu path **Tools > License Update.** 
2. .	**Click** the tab labelled **Export C2V File.** Select the type of license in **License type** drop-down box. There are two options to choose from: • **Hardware License** – ensure that the USB HASP key has been connected to the host system. • **Software License** – ensure that no USB HASP key has been attached.
3.	• Provide a file name in the **C2V file** field. • Specify where the C2V file should be saved on the local system (where the manager is running) by **clicking** the path adjacent to the **Location** label. • **Click** the button labelled **Apply**.
4.	After a few moments the utility will response with the following dialogue box:  **Click** the button labelled **OK** to proceed.
4.	Locate the `*.c2v` file saved above and send it via email to Stromasys Orders Administration at orders@stromasys.com.

Applying Vendor to Customer (V2C) License Update

After sending the C2V (customer to vendor) file to Stromasys Orders Administration (see Gathering Customer to Vendor (C2V) Detail) a response will be generated and returned. Depending on certain factors there may be one or two files attached to the response. Once the response from Stromasys Orders Administration has been received, follow the instructions below to apply the V2C (vendor to customer) files and license the software.

Instructions for Applying Vendor to Customer (V2C) Data

Step	Description
1.	Save the V2C (vendor to customer) files received from Stromasys Orders Administration. Depending on the license type, this may be one or two files: 1. A license file to format the key (optional, for hardware licenses only). The file name is of the format **`*_fmt.v2c`**. 2. The license key file. The file name is of the format **`*.v2c`**. Steps **2.** and **3.** must be performed for each of the license keys files, starting with the format key file.
2.	Open the **License Update** window, click the menu path **Tools > License Update.** 
3.	Locate the saved V2C (vendor to customer) files by **clicking** the button labelled **...**
4.	Apply the license file by clickingthe button labelled **Apply**.
5.	If a second license key file was supplied, repeat from step **2**. using the second file.

All V2C (vendor to customer) files are sequentially numbered and can only be applied once and in order. After the V2C license data files has been applied it is not possible to reuse them.

...

Managing Licensing from the Command Line

The CHARON-SSP Agent provides two command line utilities for the management of licenses. The following sections describe how to use these tools:

- Viewing the License Details
- Gathering Customer to Vendor (C2V) Detail
- Applying Vendor to Customer (V2C) License Updates

Complete documentation for the two utilities used in this section can be found in the section Command Line Utilities Reference.

- hasp_srm_view
- hasp_update

Viewing the License Details

To view the current details of the license key, use the `hasp_srm_view` utility. The following shows and example of the output generated by this utility.

Example of License Data Output from `hasp_srm_view`

```
License Manager running at host: localhost.localdomain
License Manager IP address: 127.0.0.1


The Physical KeyId: 1538162443
CHARON Sentinel HASP License key section
Reading 4032 bytes

License Manager running at host: localhost.localdomain
License Manager IP address: 127.0.0.1

The License Number: 1002784
The License KeyId: 1538162443
The Master KeyId: 2131943932
Release date: 10-APR-2015
Release time: 14:50:51
Update number: 2
End User name: Tim Ewing Sneddon
Purchasing Customer name: Stromasys Asia Pacific

Virtual Hardware: SPARCstation_20, Enterprise_450
Product Name: CHARON-SSP/4M, CHARON-SSP/4U for Linux x64
Product Code: CHSSP-xxxxx-LI
Major Version: 1
Minor Version: 0
Maximum Build: 99999
Minimum Build: 1
Host CPU supported: X64
Host Operating System required: LINUX
CPU's allowed: 24
Maximum virtual memory: 32768MB
Instances allowed: 3
Released product expiration date: 28-May-2015
Field Test product expiration date: 28-May-2015
```

Gathering Customer to Vendor (C2V) Detail

To gather the customer to vendor (C2V) data file, use the following command:

Command to Gather Customer to Vendor Data File

```
$ hasp_srm_view -c2v /path/to/keydata.c2v
```

The resulting file should then be sent to Stromasys Orders Administration via email, orders@stromasys.com.

Applying Vendor to Customer (V2C) License Updates

License keys received from Stromasys Orders Administration can be applied using the `hasp_update` command line utility. In some cases, depending on the license type there maybe one or two files to apply.

1. A license file to format the key (optional, for hardware licenses only). The file name is of the format **`*_fmt.v2c`**.
2. The license key file. The filename is of the format **`*.v2c`**.

If there are multiple keys, it is important that they are applied in the correct order. To apply a V2C file, use the following command:

Command to Apply Vendor to Customer (V2C) License File

```
$ hasp_update u /path/to/key.v2c
```

For multiple key files, this command should be used for both, consecutively.

> All V2C (vendor to customer) files are sequentially numbered and can only be applied once and in order. After the V2C license data files has been applied it is not possible to reuse them.

Using the SafeNet Sentinel Admin Control Center

The SafeNet Sentinel Admin Control Center (ACC) provides a simple web interface for the query and management of both hardware and software Sentinel license keys. It is installed and enabled as part of the Sentinel HASP Software isntallation, described in the section Installing the Sentinel HASP Software. The following sections describe some of the common tasks associated with the SafeNet Sentinel ACC.

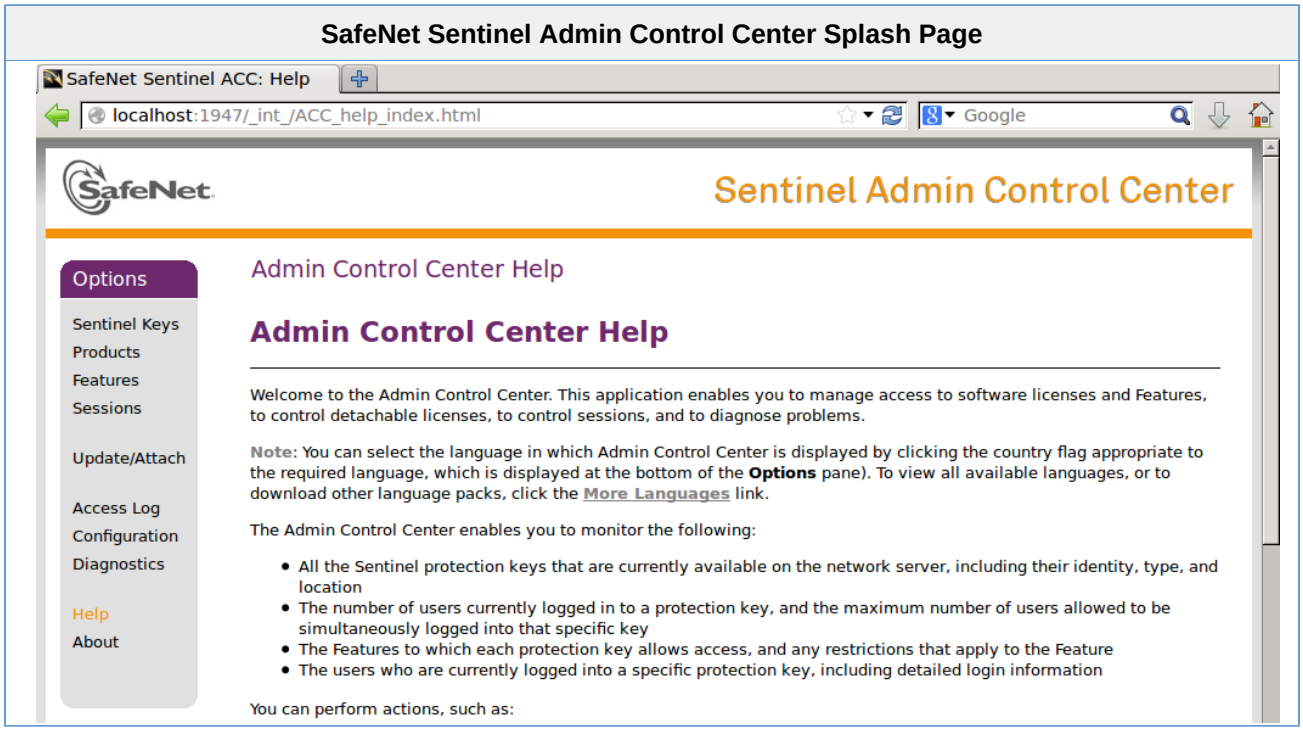
- Connecting to the Admin Control Center
- Displaying Attached Licenses
- Applying V2C (Vendor to Customer) License Files
  - Troubleshooting License Key Application

> The SafeNet Sentinel ACC is not secure at installation. To ensure a secure installation, please follow the Post-Installation Tasks described in Installing the Sentinel HASP Software.

Connecting to the Admin Control Center

To connect to the SafeNet Sentinel ACC, use a web browser to connect to port **1947/tcp** on the CHARON-SSP for Linux host system. If running a web browser on the CHARON-SSP for Linux host, use the URL `http://localhost:1947` to connect to the Admin Control Center. The screenshot below shows the default splash page, shown immediately after connecting to the Admin Control Center.

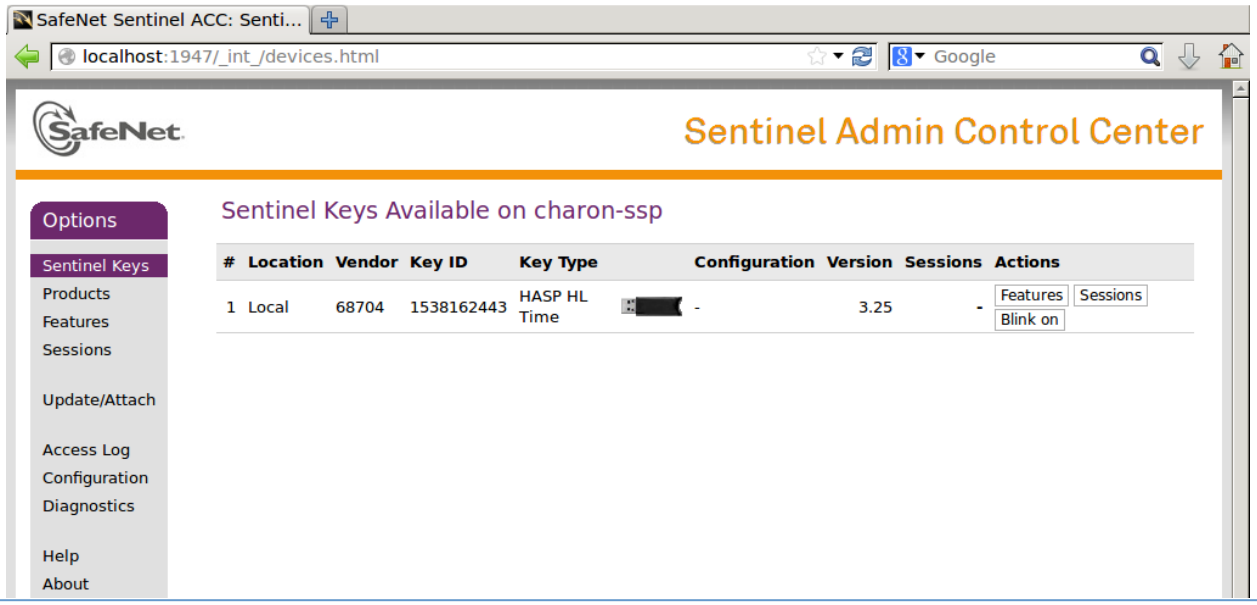
SafeNet Sentinel Admin Control Center Splash Page



Displaying Attached Licenses

From the **Options** menu on the left hand-side of the SafeNet Admin Control Center, **click** the **Sentinel Keys** item to display all attached keys and licenses. The screen shot below shows an attached Sentinel USH HASP hardware license key.

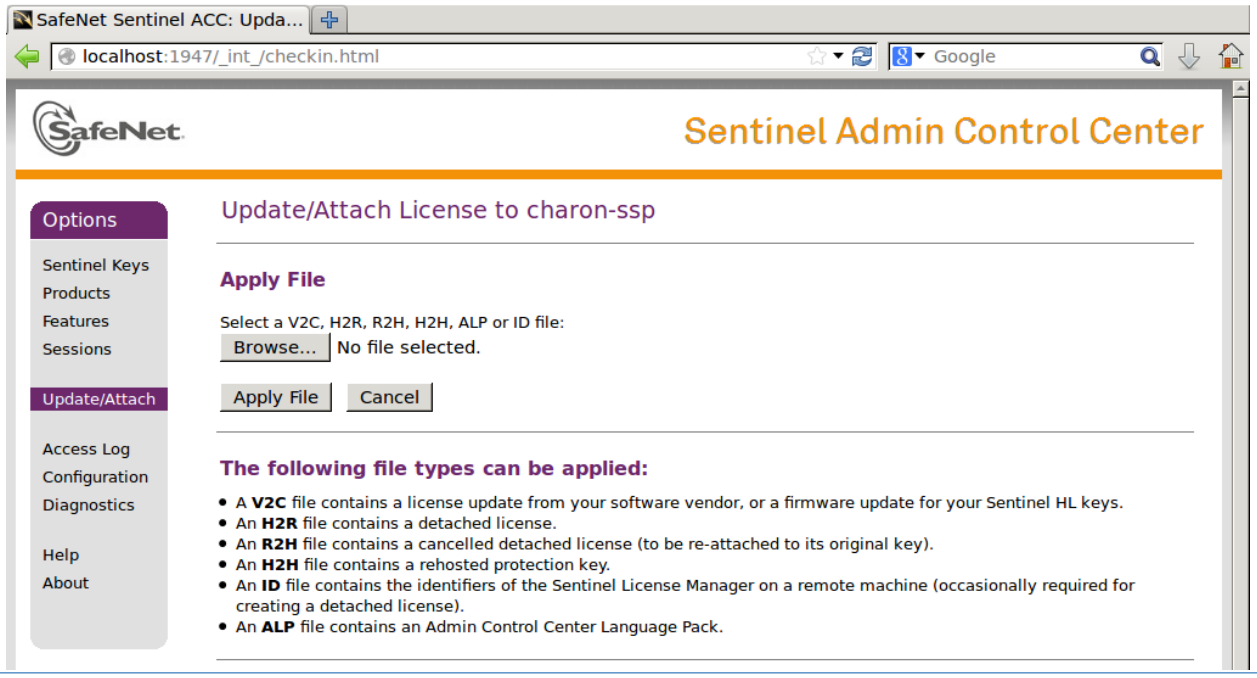
SafeNet Sentinel Admin Control Center - Sentinel Keys



Applying V2C (Vendor to Customer) License Files

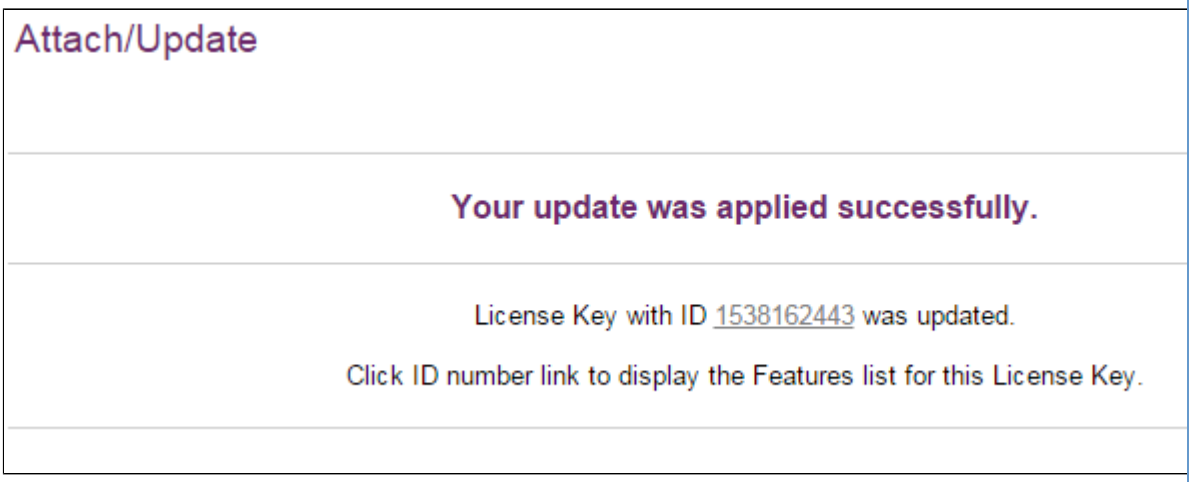
License keys received from Stromasys Orders Administration can be applied through the Safenet Sentinel ACC Update/Attach form. The screenshot below shows this form.

SafeNet Sentinel Admin Control Center - Update/Attach



To load a license key using the SafeNet Sentinel ACC Update/Attach form, follow the instructions below.

Applying a License Key File with SafeNet Sentinel ACC

Step	Description
1.	Save the V2C (vendor to customer) files received from Stromasys Orders Administration. Depending on the license type, this r 1. A license file to format the key (optional, for hardware licenses only). The file name is of the format **`*_fmt.v2c`**. 2. The license key file. The file name is of the format **`*.v2c`**. Steps **2.** and **3.** must be performed for each of the license keys files, starting with the format key file.
2.	**Click** the button labelled **Browse..** or **Choose File** and select the license key file.
3.	**Click** the button labelled **Apply File**.
4.	After the license key has been successfully applied the SafeNet Sentinel ACC will respond with a message similar to the follow  If the license fails to apply, consult the section Troubleshooting License Key Application.
5.	If a second license key file was supplied, **click** the **Update/Attach** item from the left-hand **Options** menu and repeat from step

It is not possible to generate a C2V (customer to vendor) file from the SafeNet Sentinel ACC. These must be generated with either the command line tools (see section Managing Licensing from the Command Line) or the CHARON-SSP Manager GUI (see section Managing Licenses with CHARON-SSP Manager).

Troubleshooting License Key Application

In the event there is a failure in loading the license key, use the following table as a guide to . If these solutions do not resolve the problem, please contact the Stromasys Customer Support Centre, using the details in the Obtaining Technical Assistance section of About This Guide.

SafeNet Sentinel ACC License Application Troubleshooting Steps

Error No.	Error ID	Problem	Resolution
54	HASP_UPDATE_TOO_OLD	Trying to install a V2C (vendor to customer) file with an update counter that is out of sequence with the update counter on the Sentinel protection key. The update counter value in the V2C file is lower than the value in Sentinel protection key.	Apply all other license key updates, even if they have expired and attempt to reapply the failed update. If this still does not resolve the problem, contact Stromasys Orders Administration with a new C2V (customer to vendor) file.

Configuration Reference

The CHARON-SSP virtual machines can be configured using a text-based configuration file or the CHARON-SSP Manager GUI.

This section describes the format and content of the text-based configuration file. The configuration file is made up of a number of sections that describe various aspects of the virtual machine environment. The remaining sections describes the different sections and their syntax.

Syntax

The configuration file format for CHARON-SSP follows the “INI file” format originally pioneered by MS-DOS and 16-bit Microsoft Windows. The syntax is described in EBNF (Extended Bakus Naur Format) below.

Configuration file syntax in EBNF

```
file     := (section)+
section  := ‘[‘ NAME ‘]’ ‘\n’
            (key)+
key      := NAME ‘=’ VALUE ‘\n’
```

The following briefly describes each configuration file syntax element.

Section

The configuration file is divided into sections. These sections are denoted by names enclosed in ‘[‘ and ‘]’. Figure … shows an example header.

```
[system]
```

There is no “end of section” delimiter. A section ends when either the end of file is reached or a new section is encountered.

Properties

Configuration options (properties) are specified as key/value pairs. Key names are separated from values by the ‘=’ character. All characters following the ‘=’ character to the end of the line are considered part of the value. Figure n shows an example key/value pair.

```
port = /dev/ttyS0
```

> Configuration property names are case-sensitive. This means that “port” and “Port” are not equivalent.

Comments

Comments are ignored by the emulator. They are provided for human readers and writers to leave informative notes. A comment starts with the '#' character and continues to the end of the line. Figure n shows some example comments.

```
# CHARON-SSP configuration file comment
```

Blank Lines

Blank lines are ignored by the emulator. They should be used to break up the configuration file, making it easier for a human reader to analyze.

Reference

The following sections describe how to manipulate the configuration objects and their members to set up a virtual SPARC machine:

[cpu] Section

This section controls options related to the operation of the virtual machines CPUs. The following list each of the available options:

- dit – Enable/disable Dynamic Instruction Translation.
- number – Specify number of virtual CPUs.

dit

Enable/disable Dynamic Instruction Translation.

Syntax

```
dit = on | off
```

Description

This option enables of disables the Dynamic Instruction Translation (DIT) mode. This can provide significant performance increase if enabled.

> Enabling this option will consume approximately 600MB of additional host memory.

number

Specify number of virtual CPUs.

Syntax

```
number = cpu-count
```

Description

This option specifies the total number of CPUs the virtual machine is to provide. The table below lists each supported virtual machine and the maximum number of CPUs.

Maximum Number of CPUs per Virtual Machine Model

Machine	Model Name	Max. CPUs
SPARCstation_20	Sun SPARCstation 20	1
Enterprise_450	Sun Enterprise 450	24

[ethernet] Section

This section describes the virtual Ethernet adapters attached to the virtual machine. The following list describes the available options:

- interface – Attach virtual Ethernet interface to host attached adapter.
- mac – Specify virtual Ethernet adapter MAC address.

For virtual models that support the feature, it is possible to configure multiple Ethernet controllers. For an example configuration and details on section naming, see [ethernet_n] Section.

interface

Attach virtual Ethernet interface to host attached adapter.

Syntax

```
interface = host-device
```

Description

The **`interface`** option is used to attach the virtual Ethernet adapter to a physical host adapter.

mac

Specify virtual Ethernet adapter MAC address.

Syntax

```
mac = aa-bb-cc-dd-ee-ff
```

Description

This optional parameter can be used to force the physical address of the network adapter to a certain address. This can be particularly relevant in the case where a software product may be licensed against the network adapter of a native SPARC system.

[ethernet_n] Section

This section is used for virtual machine configurations where there are multiple Ethernet controllers. The properties are the same as those described in the [ethernet] Section. However, the section naming is slightly different in that the names are suffixed with an underscore and the controller number.

The configuration example below demonstrates the configuration of two virtual Ethernet adapters.

Example Ethernet Adapter Configuration

```
[ethernet_1]
interface = intbr0

[ethernet_2]
interface = intbr1
mac = 08-00-2b-aa-bb-cc
```

[log] Section

This section describes the configuration of the CHARON-SSP for Linux virtual machine logging facility. The following lists the properties supported by this section:

- destination – Logging facility output destination.
- path – Logging output path.
- severity – Logging severity level.

> It is important to note that all properties in this section must be configured correctly to ensure that the virtual machine will start. If the following message is received, something in the logging section is not configured correctly.
>
> ```
> Failed to setup log!
> ```
>
> Faults in the logging facility configuration could be:
>
> - Incorrect path or insufficient privilege to create the file,
> - Unconfigured property or properties, or
> - Misconfigured property value.

destination

Logging facility output destination.

Syntax

```
destination = log-destination
```

Description

This property controls the destination of the logging facility output. The table below lists all possible values for *`log-destination`*.

Logging Destination Keywords

Destination	Description
`all`	Write output to all possible destinations.
`console`	Write output to the `/dev/console` device only.
`file`	Write output to the file specified by **`path`** only.

path

Logging output path.

Syntax

```
path = log-path
```

Description.

Specify a path in *`log-path`* to write virtual machine logging messages to.

> This configuration property must be present, even if **`destination`** is set to a value other than `all` or `file`.

severity

Logging severity level.

Syntax

```
severity = level
```

Description

Virtual machine logging messages are arranged into levels to make messages more relevant and reduce log file size. The `severity` property controls which level messages are included in the logging output. The list below lists these levels from most to least verbose. Setting logging to a specific level also includes all levels below it. For example, setting `severity` to `warning` ensures that `error` and `fatal` are also included in the output and other levels are not.

Logging Severity Levels

Level	Description
`debug`	Debug and all lower level messages are logged.
`info`	Informational and all lower level messages are logged.
`warning`	Warning and all lower level messages are logged.
`error`	Error and all lower level messages are logged.
`fatal`	Only fatal error messages are logged.

[nvram] Section

This section is used to configure the location of the NVRAM backing file store as well as some other NVRAM options. The list below describes the options that can be set:

- hostid – Set Sun Host ID.
- path – Specify location of console NVRAM file.

hostid

Set Sun Host ID.

Syntax

```
hostid = hex-hostid
```

Description

This optional property can be used to configure the Sun Host ID of the virtual machine. The value of *hex-path* is of the format *0xnnnnnnnn*.

path

Specify location of console NVRAM file.

Syntax

```
path = nvram-path
```

Description

This property specifies the location of the console NVRAM image. It contains information such as environment variable settings, boot flags, etc. Set *nvram-path* to a location on the host system where this information can be stored.

[ram] Section

This section controls the virtual machine memory environment. The following list describes these options:

- allocator – Virtual machine memory allocator.
- size-- Virtual machine memory size.

allocator

Virtual machine memory allocator.

Syntax

```
allocator = memory-allocator
```

Description

The **`allocator`** option is used to indicate to the virtual machine which memory allocation method the host system will use to allocate the virtual memory environment. The *memory-allocator* value can be set according to the able below:

Virtual Machine Memory Allocators

Allocator	Description
hugetbl	All virtual machine RAM is allocate from hugetlbfs.
malloc	All virtual machine RAM is allocated from system heap.
mmap	All virtual machine RAM is allocated from file-backed virtual memory via mmap.

size

Virtual machine memory size.

Syntax

```
size = memory-size
```

Description

This option specifies the amount of host memory that the virtual machine should make available to the guest. It is specified in megabytes (MB). T he table below describes the allocation rules for each virtual machine model.

Virtual Machine Allocation Increments and Maximum Sizes

Virtual Machine Model	Memory Allocation Rules
Sun SPARCstion 20	64MB, 128MB, 256MB and 512MB.
Sun Enterprise 450	1 - 32 GB in 1 GB increments.

[scsi_n] Section

This section describes the virtual SCSI storage device configuration. Each device requires a separate section heading, where *n* is replaced with a SCSI ID number between 0 and 15. The example below shows the configuration entries for a physical tape device attached at SCSI ID 5. All devices are attached to the internal bus of the SPARC virtual machine.

Physical tape device attached at SCSI ID 5

```
[scsi_5]
path = /dev/tape
type = tape
```

The following list describes the available options for `[scsi_n]` sections:

- path – Virtual SCSI storage device path.
- type – Specify type of virtual SCSI storage device.

For virtual models that support the feature, it is possible to configure additional virtual SCSI devices attached to a external SCSI controller. For an example configuration and details on section naming, see [scsix_n] Section.

> The SCSI ID 7 is reserved for the virtual SCSI adapter, consequently it is not possible to configure a section titled `[scsi_7].`

path

Virtual SCSI storage device path.

Syntax

```
interface = host-device
```

Description

The **`interface`** option is used to attach the virtual Ethernet adapter to a physical host adapter.

...reference below...

type

Specify type of virtual SCSI storage device.

Syntax

```
type = device-type
```

Description

This property describes the type of the virtual SCSI storage device. It is possible to attach a range of devices in various formats, including container files and physical devices. The table below describes each possible value for *`device-type`*.

Virtual Storage Device Types

Device Type	Description	Example Path
`cdrom`	Physical optical device.	`/dev/cdrom`
`disk`	Physical disk device.	`/dev/sda`
`iso`	Virtual CD-ROM ISO container file.	`/usr/local/share/iso/suns-4.1.4.iso`
`tape`	Physical tape device.	`/dev/tape`
`vdisk`	Virtual disk container file.	`/usr/local/vm/bender/disk0.vdisk`
`vtape`	Virtual tape container file.	`/usr/local/vm/leela/tape0.vtape`

[scsix_n] Section

These sections are used for virtual machine configurations that support the external (although not necessarily physically external) SCSI controller, such as the Sun Enterprise 450. Each device requires a separate section heading, where *n* is replaced with a SCSI ID number between 0 and 15. The example below demonstrates a configuration of three devices (two CD-ROMs backed by ISO container files and one physical disk) attached to the external SCSI bus.

Example External SCSI Device Configuration

```
[scsix_0]
type = disk
path = /dev/sr0

[scsix_5]
type = iso
path = /usr/local/share/iso/sunos4.1.4.iso

[scsix_6]
type = iso
path = /usr/local/share/iso/solaris.1.1.2.iso
```

The section properties describes in the [scsi_n] Section are also applicable here.

> The SCSI ID 7 is reserved for the virtual SCSI adapter, consequently it is not possible to configure a section titled `[scsix_7]`.

[system] Section

The system section is used to configure "system-wide" properties of the virtual machine. The list below describes the options that can be set:

- cpu_affinity – Assign virtual CPU processing to a specific host CPU.
- io_affinity – Assign virtual machine I/O processing to a specific host CPU.
- io_cpus – Number of host CPUs reserved for virtual machine I/O processing.
- machine – Specify the Alpha system model of the virtual machine.

cpu_affinity

Assign virtual CPU processing to a specific host CPU.

Syntax

```
cpu_affinity = cpu-affinity [, cpu-affinity [, ... ] ]
```

Description

This option is a comma delimited list of host CPUs (or cores) that the virtual machine is to assign to virtual CPU threads.

The virtual machine will assign affinity automatically if this option is not set.

io_affinity

Assign virtual machine I/O processing to a specific host CPU.

Syntax

```
io_affinity = io-affinity [, io-affinity [, ... ] ]
```

Description

This option accepts a comma delimited list of specific host CPUs (or cores) the virtual machine will assign to I/O processing.

If this directive is unset the virtual machine will assign affinity automatically.

io_cpus

Number of host CPUs reserved for virtual machine I/O processing.

Syntax

```
io_cpus = cpu-count
```

Description

Use this option to reserve a specific number of host CPUs (or cores) for virtual machine I/O processing.

If both io_affinity and io_cpus are not set, the virtual machine will automatically reserve one third of the host CPUs (or cores) for I/O processing.

machine

Specify the SPARC system model of the virtual machine.

Syntax

```
machine = "model-name"
```

Description

The model keyword is used to indicate the specific model of the SPARC-based system to emulate. The following table lists the possible values of *model-name* and the systems they represent.

Virtual Machine Model Names

Model name	System Name
`Enterprise_450`	Sun Enterprise 450
`SPARCstation_20`	Sun SPARCstation 20

> The *`model-name`* specified here must correspond to what is configured on the HASP license dongle.

[ttya] Section

The ttya section is used to configure the first serial port (sometimes used as the console) on the SPARC virtual machine. Using these options it is possible to attach the virtual serial port to a network socket or a physical serial port attached to the host. The list below describes the options that can be configured:

- port – Virtual serial port specification.
- start_console – Start PuTTY at virtual machine boot.
- type – Virtual serial port type.

port

Virtual serial port specification.

Syntax

```
port = port-spec
```

Description

This option is dependent on the setting of the **type** option. The table below describes the valid values for *port-spec*.

Virtual Serial Port Types

Port Type	Port Specification	Description
physical	Path to physical device, e.g. /dev/ttyS0	The virtual serial port will be attached to a physical serial port attached to the host.
socket	TCP/IP socket number, e.g. 9000	The virtual serial port will be attached to a network port that can be connected to
terminal		

start_console

Start PuTTY at virtual machine boot.

Syntax

```
start_console = on | off
```

Description

This option enables (or disables) the automatic starting of PuTTY when the virtual machine starts.

This option is enabled by default.

type

Virtual serial port type.

Syntax

```
type = port-type
```

Description

This option configures how the serial console port will be connected. The table below lists the possible values for *`port-type`* and their purpose.

Virtual Serial Port Types

Port Type	Description
`physical`	The virtual serial port will be connected to a physical, host-attached serial port.
`socket`	The virtual serial port is to be connected to a network socket.
`terminal`	

> For details of the port specification, see the `port` section.

[ttyb] Section

The ttya section is used to configure the first serial port (sometimes used as the console) on the SPARC virtual machine. Using these options it is possible to attach the virtual serial port to a network socket or a physical serial port attached to the host. The list below describes the options that can be configured:

- port – Virtual serial port specification.
- type – Virtual serial port type.

port

Virtual serial port specification.

Syntax

```
port = port-spec
```

Description

This option is dependent on the setting of the **type** option. The table below describes the valid values for *port-spec*.

Virtual Serial Port Types

Port Type	Port Specification	Description
physical	Path to physical device, e.g. /dev/ttyS1	The virtual serial port will be attached to a physical serial port attached to the host.
socket	TCP/IP socket number, e.g. 9000	The virtual serial port will be attached to a network port that can be connected to
terminal		

type

Virtual serial port type.

Syntax

```
type = port-type
```

Description

This option configures how the alternate serial port will be connected. The table below lists the possible values for *port-type* and their purpose.

Virtual Serial Port Types

Port Type	Description
physical	The virtual serial port will be connected to a physical, host-attached serial port.
socket	The virtual serial port is to be connected to a network socket.
terminal	

> For details of the port specification, see the port section.

Console Reference

The CHARON-SSP SPARC virtual machines use a subset of the Sun OpenBoot console found on native Sun workstations and servers. The figure below shows the initial console screen at boot on a virtual SPARCstation 20.

SPARCstation 20 OpenBoot Console

```
           SMCC SPARCstation 20 Emulator by Stromasys


CPU_#0       TI, TMS390Z50(3.x)       0Mb External cache

CPU_#1       ****** NOT installed ******
CPU_#1       ****** NOT installed ******
CPU_#1       ****** NOT installed ******


           >>>>> Power On Self Test (POST) is running .... <<<<<

SPARCstation 20 (1 X 390Z50), No Keyboard
Emulate OBP Rev. 2.25, 64 MB memory installed, Serial #12648430.
Ethernet address 2:c:29:4a:d3:29, Host ID: 72c0ffee.


Type  help  for more information

Can not load boot block!
ok
```

Reference

The following is a list of the currently supported console commands:

banner

Display power-on banner.

Syntax

```
banner
```

Description

Use this command to display the power-on banner.

Example

The following example demonstrates the output of the `banner` command on CHARON-SSP/L10 configured as a SPARCstation 20.

Example `banner` Command Output

```
ok banner
banner
SPARCstation 20 (1 X 390Z50), No Keyboard
Emulate OBP Rev. 2.25, 64 MB memory installed, Serial #12648430.
Ethernet address 2:c:29:4a:d3:29, Host ID: 72c0ffee.
```

boot

Load operating system.

Syntax

```
boot [ device-alias ] [ boot-args ]
```

Description

This commands boots the specified *device-alias* passing any optional *boot-args* to the kernel.

For a complete list of device aliases, see the `devalias` command.

Example

The following example demonstrates the output of the `boot` command on CHARON-SSP/L10 configured as a SPARCstation 20 and booting SunOS 4.1.4 from CD-ROM.

Example `boot` Command Output

```
ok boot cdrom
boot cdrom
Boot device: /iommu@f,e0000000/sbus@f,e0001000/espdma@f,400000/esp@f,800000/sd@6,0:d    File and a
Boot Release 4.1.4 (sun4m) #2: Fri Oct 14 11:07:52 PDT 1994
Copyright (c) 1983-1990, Sun Microsystems, Inc.
Boot: Romvec version 3.
root on /iommu@f,e0000000/sbus@f,e0001000/espdma@f,400000/esp@f,800000/sd@6,0:d fstype 4.2
Boot: vmunix
.Size: 868352.....................................................................................
.....................................................................................
.....................................................................................
.....................................................................+2319136+751
Statistics:
SuperSPARC: PAC ENABLED
SunOS Release 4.1.4 (MUNIX) #2: Fri Oct 14 11:09:07 PDT 1994
Copyright (c) 1983-1993, Sun Microsystems, Inc.
```

devalias

Display device aliases.

Syntax

```
devalias
```

Description

This commands display the current device aliases. This shows the link between the aliases, such as `cdrom` and the devices shown in the device tree, listed by `show-devs` .

Example

The following example demonstrates the output of the `devalias` command.

Example `devalias` Command Output

```
ok devalias
devalias
ttyb            /obio/zs@0,100000:b
ttya            /obio/zs@0,100000:a
keyboard!       /obio/zs@0,0:forcemode
keyboard        /obio/zs@0,0
floppy          /obio/SUNW,fdtwo
scsi            /iommu/sbus/espdma@f,400000/esp@f,800000
net-aui         /iommu/sbus/ledma@f,400010:aui/le@f,c00000
net-tpe         /iommu/sbus/ledma@f,400010:tpe/le@f,c00000
net             /iommu/sbus/ledma@f,400010/le@f,c00000
disk            /iommu/sbus/espdma@f,400000/esp@f,800000/sd@3,0
cdrom           /iommu/sbus/espdma@f,400000/esp@f,800000/sd@6,0:d
tape            /iommu/sbus/espdma@f,400000/esp@f,800000/st@4,0
tape1           /iommu/sbus/espdma@f,400000/esp@f,800000/st@5,0
tape0           /iommu/sbus/espdma@f,400000/esp@f,800000/st@4,0
disk3           /iommu/sbus/espdma@f,400000/esp@f,800000/sd@3,0
disk2           /iommu/sbus/espdma@f,400000/esp@f,800000/sd@2,0
disk1           /iommu/sbus/espdma@f,400000/esp@f,800000/sd@1,0
disk0           /iommu/sbus/espdma@f,400000/esp@f,800000/sd@0,0
```

help

Display OpenBoot console help.

Syntax

```
help [ command ]
```

Description

Use this command to display the list of commands supported by the OpenBoot console. For brief help on individual commands specify the ***command*** parameter.

Example

The following example demonstrates the output of the `help` command.

Example `help` Command Output

```
ok help
help
We only support following commands:
boot      devalias    printenv    setenv   probe-scsi
reset     show-devs   banner      history  help

Enter 'help command-name' for more help
Examples:  help setenv
```

history

Display console command history.

Syntax

```
history
```

Description

This command displays a list of all commands previously entered at the OpenBoot Console.

Example

The following example demonstrates the output of the `history` command.

Example `history` Command Output

```
ok history
history
   1   printenv
   2   help
   3   help devalias
   4   help history
   5   help probe-scsi
   6   probe-scsi
   7   show-devs
   8   banner
```

printenv

Display environment variables.

Syntax

```
printenv
```

Description

Use this command to print the current and default values of OpenBoot console variables.

Example

The following example demonstrates the output of the `printenv` command.

Example `printenv` Command Output

```
ok printenv
printenv
Parameter Name        Value                          Default Value
auto-boot?            false                          true
boot-file             -v
boot-device           disk:a disk1                   disk net
ttya-mode             9600,8,n,1,-                   9600,8,n,1,-
ttyb-mode             9600,8,n,1,-                   9600,8,n,1,-
```

probe-scsi

Scan SCSI bus for attached devices.

Syntax

```
probe-scsi
```

Description

This command scan the SCSI bus to locate attached devices.

Example

The following example demonstrates the output of the `probe-scsi` command on system with a single virtual CD-ROM..

Example `probe-scsi` Command Output

```
ok probe-scsi
probe-scsi
Target 0
  Unit 0   Disk     virtual Scsicdrom (c)SRI0200
```

quit

Shutdown virtual machine.

Syntax

```
quit
```

Description

Use this command to shutdown the virtual machine.

Example

The following example demonstrates the output of the `quit` command on CHARON-SSP configured as a SPARCstation 20.

Example `banner` Command Output

```
ok quit
quit
The system will be shutdown soon...
```

reset

Restart the system.

Syntax

```
reset
```

Description

This command restarts the SPARC virtual machine.

Example

The following example demonstrates the output of the `reset` command on CHARON-SSP/L10 configured as a SPARCstation 20.

Example `reset` Command Output

```
ok reset
reset

           SMCC SPARCstation 20 Emulator by Stromasys


CPU_#0       TI, TMS390Z50(3.x)       0Mb External cache

CPU_#1       ****** NOT installed ******
CPU_#1       ****** NOT installed ******
CPU_#1       ****** NOT installed ******


           >>>>> Power On Self Test (POST) is running .... <<<<<

SPARCstation 20 (1 X 390Z50), No Keyboard
Emulate OBP Rev. 2.25, 64 MB memory installed, Serial #12648430.
Ethernet address 2:c:29:4a:d3:29, Host ID: 72c0ffee.


Type  help  for more information
ok
```

setenv

Set console environment variables.

Syntax

```
setenv variable value
setenv variable --
```

Description

This command sets the console configuration `variable` to `value`. For a complete list of possible `variable` names and details, see the table below.

To restore `variable` to its default value, specify `--` in place of `value`.

`setenv` Variables

Variable	Default Value	Description
`auto-boot?`	`false`	If `true`, boots automatically after power on or reset.
`boot-file`	empty string	A string of arguments to be passed to the boot loader (e.g. -a or -v).
`boot-device`	`disk net`	Space delimited list of devices to attempt booting from.
`ttya-mode`	`9600,8,n,1,-`	Serial line configuration for ttya
`ttyb-mode`	`9600,8,n,1,-`	Serial line configuration for ttyb

> Changes to environment variables are stored in NVRAM and permanent. However, they only take effect after executing the `reset` command.

Example

The following example demonstrates the output of the `setenv` command.

Example {{setenv} Command Output

```
ok setenv auto-boot? true
setenv auto-boot? true
auto-boot? =          true
```

show-devs

Display device tree.

Syntax

```
show-devs
```

Description

This command displays the tree of devices visible from the console.

Example

The following example demonstrates the output of the `show-devs` command.

Example show-devs Command Output

```
ok show-devs
show-devs
/TI,TMS390Z50@f,f8ffffc
/SUNW,sx@f,80000000
/eccmemctl@f,0
/virtual-memory@0,0
/memory@0,0
/obio
/iommu@f,e0000000
/openprom
/aliases
/options
/packages
/obio/power@0,a01000
/obio/auxio@0,800000
/obio/SUNW,fdtwo@0,700000
/obio/interrupt@0,400000
/obio/counter@0,300000
/obio/eeprom@0,200000
/obio/zs@0,0
/obio/zs@0,100000
/iommu@f,e0000000/sbus@f,e0001000
/iommu@f,e0000000/sbus@f,e0001000/SUNW,bpp@f,4800000
/iommu@f,e0000000/sbus@f,e0001000/ledma@f,400010
/iommu@f,e0000000/sbus@f,e0001000/espdma@f,400000
/iommu@f,e0000000/sbus@f,e0001000/ledma@f,400010/le@f,c00000
/iommu@f,e0000000/sbus@f,e0001000/espdma@f,400000/esp@f,800000
/iommu@f,e0000000/sbus@f,e0001000/espdma@f,400000/esp@f,800000/st
/iommu@f,e0000000/sbus@f,e0001000/espdma@f,400000/esp@f,800000/sd
/packages/obp-tftp
/packages/deblocker
/packages/disk-label
```

Command Line Utilities Reference

In many cases it may be preferrable to be able to perform maintenance and management tasks for the command line of the Linux host system. This sections describes how to set up the PATH environment variable to use these utilities as well as a comprehensive reference.

Prerequisites

The utilities described in this reference section are installed as a part of the CHARON-SSP Agent for Linux software. Depending on your environment you many not need or want support for the GUI environment. If this is the case it is recommended that following installation of the CHARON-SSP Agent (described in the section, Installing the CHARON-SSP Agent) the Agent service is immediately disabled.

Disabling the CHARON-SSP Agent Service

Use the following commands to disable the CHARON-SSP Agent service and avoid automatic start up at system boot.

Disable the CHARON-SSP Agent Service

```
# chkconfig charon-agentd-ssp off
```

Configure the Shell Path

To add the command line utilities to a C Shell environment, add the following to the end of `.login`:

Add Command Line Utilities to C Shell `.login`

```
setenv PATH $PATH:/opt/charon-agent/ssp-agent/utils/license
setenv PATH $PATH:/opt/charon-agent/ssp-agent/utils/mkdisk
setenv PATH $PATH:/opt/charon-agent/ssp-agent/utils/mktape
setenv PATH $PATH:/opt/charon-agent/ssp-agent/utils/ncu
```

To add the command line utilities to a Bourne Shell environment (e.g. bash or sh), add the following to the end of `.profile`, `.bash_profile` or `.bashrc`:

Add Command Line Utilities to Bourne Shell Login Profile

```
PATH=$PATH:/opt/charon-agent/ssp-agent/utils/license
PATH=$PATH:/opt/charon-agent/ssp-agent/utils/mkdisk
PATH=$PATH:/opt/charon-agent/ssp-agent/utils/mktape
PATH=$PATH:/opt/charon-agent/ssp-agent/utils/ncu
export PATH
```

Reference

The following utilities can be used from the command line to support and manipulate the CHARON-SSP host environment.

hasp_srm_view

Name

hasp_srm_view –CHARON Sentinel HASP Utility

Synopsis

```
hasp_srm_view [OPTION]
```

Description

The **hasp_srm_view** utility provides a simple command line utility for gathering Sentinel license information. If no options are specified, **-l**, is specified by default.

-?, -h, -help

Display the utility usage message.

-c2v FILENAME

Collect the Sentinel HASP key status information and write it to **FILENAME**.

-fgp FILENAME

Collect the host fingerprint information for generating a Sentinel software license to **FILENAME**.

-l

Show the product license details currently attached to the host.

-tfr LICENSEID [FILENAME]

-tfr <LicenseID> <recipient file> - to transfer HASP SL license (V2C file)
-tfr <LicenseID> - to remove HASP SL license (V2C file) from the local host

-idf

to get transfer recipient (ID) file "recipient.id"

Exit Status

The **hasp_srm_view** utility exits **0** on success and **255** if an error occurs.

Examples

The following example shows the output of the `-l` qualifier for an attached Sentinel USB HASP key.

```
License Manager running at host: localhost.localdomain
License Manager IP address: 127.0.0.1


The Physical KeyId: 663427931
CHARON Sentinel HASP License key section
Reading 4032 bytes

License Manager running at host: localhost.localdomain
License Manager IP address: 127.0.0.1

The License Number: 1002783
The License KeyId: 663427931
The Master KeyId: 2131943298
Release date: 24-MAR-2015
Release time: 11:47:56
Update number: 3
End User name: Stromasys Asia Pacific
Purchasing Customer name: Stromasys Asia Pacific

Virtual Hardware: SPARCstation_20
Product Name: CHARON-SSP/4M for Linux x64
Product Code: CHSSP-xxxxx-LI
Major Version: 1
Minor Version: 0
Maximum Build: 99999
Minimum Build: 1
Host CPU supported: X64
Host Operating System required: LINUX
CPU's allowed: 1
Maximum virtual memory: 512MB
Instances allowed: 4
Released product expiration date: 01-Oct-2015
Field Test product expiration date: 01-Oct-2015
```

The example below demonstrates creating a C2V (customer to vendor) file for the purpose of requesting a license from Stromasys.

```
hasp_srm_view -c2v /tmp/hasp.c2v
```

hasp_update

Name

hasp_update –Sentinel HASP Update and Transfer Utility

Synopsis

```
hasp_update u filename
```

Description

The **hasp_update** utility provides a simple command-line interface for manipulating the HASP License Key.

u

Apply the HASP key update found in **filename**.

Exit Status

The **hasp_update** virtual machine exits **0** on success and **255** if an error occurs.

Examples

The following example demonstrates the application of a V2C (vendor-to-customer) license key file.

Applying a Vendor to Customer (V2C) License Key Update

```
# hasp_update u /tmp/0002_1002784_27-May-2015.v2c

press ENTER
```

mkdskcmd

Name

mkdskcmd –CHARON virtual disk container creation utility.

Synopsis

```
mkdskcmd [OPTION] ...
```

Description

Create virtual disk container files for use the the CHARON family of virtual machines. By default, this utility will display a usage message.

Mandatory arguments to long options are mandatory for short options too.

`-a, --avtable PATHNAME`

Use this option to specify an alternate location, **`PATHNAME`**, of the known disk device table.

`-c, --blcount BLOCKCOUNT`

Specify the number of blocks, **`BLOCKCOUNT`**, in the virtual disk container file. Use this option with **`-z, --blsize`** to set the block size.

`-d, --disk NAME`

Specify the **`NAME`** of a known disk type. Use **`-l, --list`** to see a list of disk types supported by the utility.

`-h, --help`

Display the utility usage message.

`-l, --list`

Display a list of the known disk types.

`-o, --output FILENAME`

Specify the pathname of the virtual disk container file.

`-s, --silent`

Do not write any output to the terminal.

`-z, --blsize BLOCKSIZE`

Specify the **`BLOCKSIZE`** in bytes when creating a custom virtual disk container file. This option must be used with **`-c, --blcount`**.

Exit Status

The `mkdskcmd` utility exits **`0`** on success and **`non-zero`** if an error occurs.

Examples

The following example creates the virtual disk container file `/usr/local/vm/leela/disk0.vdisk` using the geometry of a Seagate ST446452W 46GB disk drive.

```
mkdskcmd -o /usr/local/vm/leela/disk0.vdisk -d ST446452W
```

This example creates a virtual disk container file, `/usr/local/vm/bender/disk0.vdisk`, using a block size of 4,096 bytes and a total of 16,384 blocks.

```
mkdskcmd -o /usr/local/vm/bender/disk0.vdisk -z 4096 -c 16384
```