stromasys

Charon-AXP V4.12 for Windows User's Guide

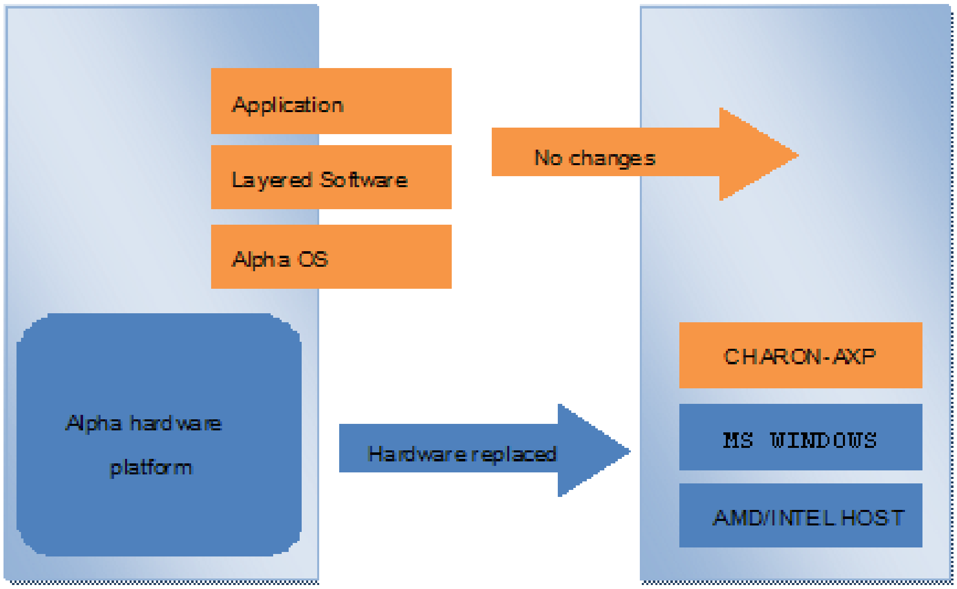
Contents

Introduction

Table of Contents

General Description

Alpha Hardware Virtualization allows users of Alpha (Previously known as DIGITAL, Compaq, or HP Alpha) computers to move application software and user data to a modern Intel or AMD based x64 compatible platform without having to make changes to software and data. Alpha Hardware Virtualization is a software solution that replaces Alpha hardware.



This approach is best understood when the Alpha Hardware Virtualization Software is viewed as a special interface between the old Alpha software and a new hardware platform. Basically, the CHARON software presents a Alpha hardware interface to the original Alpha software, so that the existing software cannot detect a difference. This means no changes have to be made to the existing software. User programs and data can be copied to a new modern industry standard server (64-bit Intel or AMD) and continue to run for many more years.

The Alpha virtualization software is designed to replace single and multi-CPU Alpha computer systems, including:

- AlphaServer 400
- AlphaServer 800
- AlphaServer 1000
- AlphaServer 1000A
- AlphaServer 1200
- AlphaServer 2000
- AlphaServer 2100
- AlphaServer 4000
- AlphaServer 4100
- AlphaServer DS10
- AlphaServer DS10L
- AlphaServer DS15
- AlphaServer DS20
- AlphaServer DS25
- AlphaServer ES40
- AlphaServer ES45
- AlphaServer GS80
- AlphaServer GS160
- AlphaServer GS320

The principles of Alpha Hardware Virtualization

Virtualized hardware

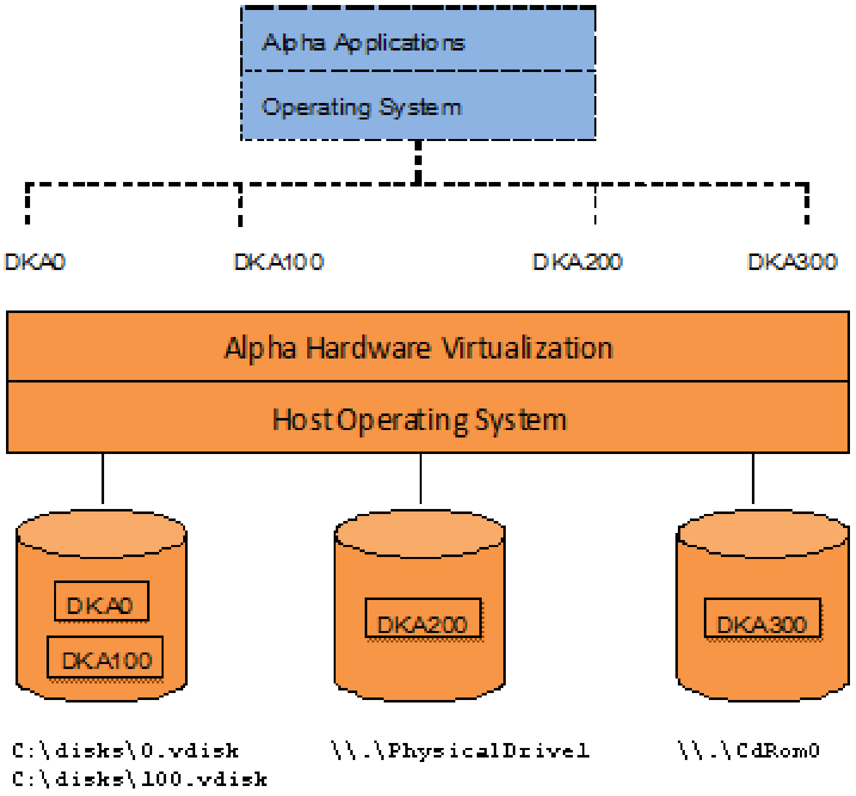
CHARON-AXP virtualizes various Alpha architectures and meets or exceeds the performance level of these Alpha systems when run on the recommended hardware platform.

The following table shows which hardware boards CHARON virtualizes:

Subsystem	Covered Alpha hardware
Serial Lines Controllers	On-board serial line port COM2, Family of PCI controllers: PBXDA-BA, PBXDA-BB, PBXDA-AC, PBXDA-AC
IDE/ATAPI CD-ROM Controller	Virtual Acer Labs 1543C
PCI Fibre Channel Controller	KGPSA-CA
PCI SCSI Controller	KZPBA
PCI FDDI Controller	DEFPA
PCI Network Controllers	DE435, DE450, DE500AA, DE500BA, DE602, DE602AA

Host platform

The Virtualization Software presents standard Alpha devices to the Alpha operating system, allowing the OS to function as though it were still running on a Alpha computer. For example, virtual disk container files in a directory or physical devices of the host Windows platform are presented by the Virtualization Software to the Alpha OS as emulated SCSI disks attached to a PCI SCSI adapter.



With the use of current storage technology, disks do not have to be physically attached to the Host platform, they can also reside on a SAN or iSCSI storage structure. A similar translation process is also valid for other emulated hardware devices.

About this guide

Table of contents

Obtaining Documentation

The latest released version of this manual and other related documentation are available on the Stromasys support website at Product Documentation and Knowledge Base.

Obtaining Technical Assistance or General Product Information

Obtaining Technical Assistance

Several support channels are available to cover the Charon virtualization products.

If you have a support contract with Stromasys, please visit http://www.stromasys.com/support/ for up-to-date support telephone numbers and business hours. Alternatively, the support center is available via email at support@stromasys.com.

If you purchased a Charon product through a Value-Added Reseller (VAR), please contact them directly.

Obtaining General Product Information

If you require information in addition to what is available on the Stromasys Product Documentation and Knowledge Base and on the Stromasys web site y ou can contact the Stromasys team using https://www.stromasys.com/contact/, or by sending an email to info@stromasys.com.

For further information on purchases and the product best suited to your requirements, you can also contact your regional sales team by phone:

Region	Phone	Address
Americas	+1 919 239 8450	Stromasys LLC 871 Marlborough Ave, suite 100, Riverside CA 92507 USA
Europe, Middle-East and Africa	+41 22 794 1070	Avenue Louis-Casai 84 1216 Cointrin Switzerland

Conventions

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.

Definitions

Term	Description
Host	The system on which the emulator runs, also called the Charon server
Guest	The operating system running on a Charon instance, for example, Tru64 UNIX, OpenVMS, Solaris, MPE or HP-UX

Related documents

- Charon-AXP V4.12 for Windows - User's Guide (still internal!)
- Charon-AXP V4.12 for Windows - Release Notes

CHARON-AXP for Windows installation

Table of contents

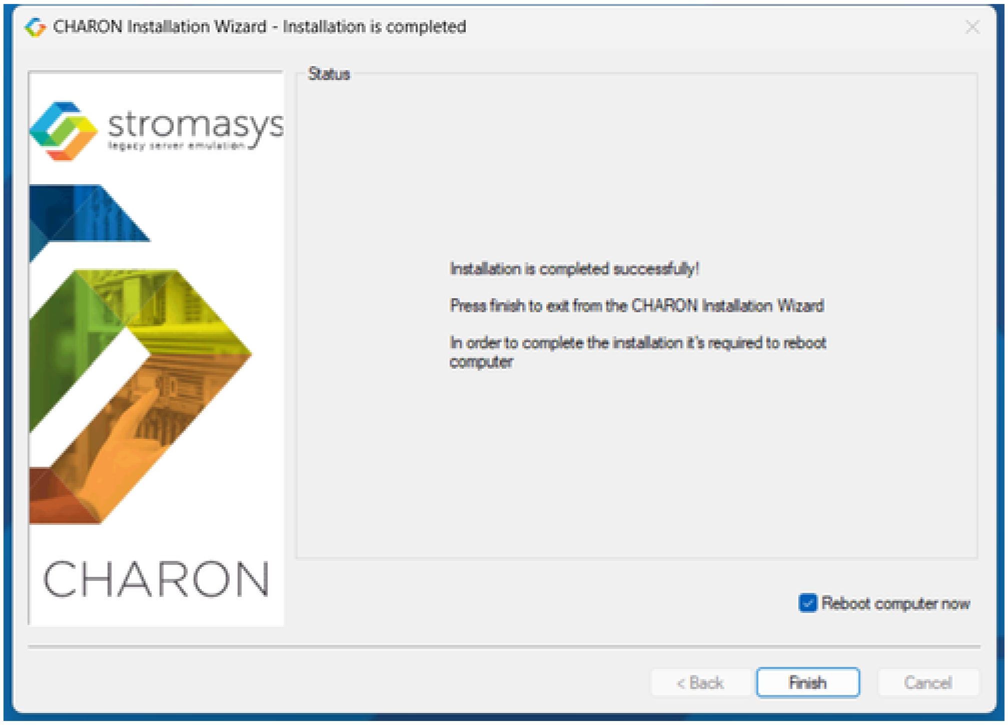
Introduction

The CHARON-AXP product is distributed in the form of a zip file that contains the CHARON-AXP installation executable ("InstallShell.exe"). The InstallShell procedure offers the option to select which modules to install (optionally). It is recommended to install all modules (the default) by not selecting "Display components" checkbox.

⚠ Before installing CHARON-AXP, please perform all hardware and software host system checks to ensure the host platform meets minimum CHARON-AXP installation requirements

CHARON-AXP installation steps:

- Optionally, create a specific CHARON username with administrative privileges.
- Extract the CHARON-AXP installation procedure files from its archive.
- Install CHARON-AXP by running the "InstallShell.exe" installer and follow the instructions.
- Install CHARON-AXP license (hardware dongle or software license).
- Configure CHARON-AXP host system, for example, network adapter, disable some Windows features, etc. (see section *Host system preparation*).



Installation is completed successfully!

Press finish to exit from the CHARON Installation Wizard

In order to complete the installation it's required to reboot computer

☑ Reboot computer now

Host Hardware Requirements

Number of CPU cores

Each CHARON-AXP emulated CPU requires a corresponding physical core. The total number of host CPUs must exceed the number of emulated CPUs since some of the host CPUs must be dedicated to serving CHARON I/O operations and fulfill host operating system needs. If several CHARON instances run in parallel, the required number of CPU cores is cumulative.

The following table shows the minimum and recommended number of CPUs required for each models for one instance (note that each model instance is able to run on 2 CPU cores hosts, but this configuration does not support emulation of all the virtual CPUs):

CHARON-AXP models	Minimum number of host CPU cores	Recommended number of host CPU cores
AlphaServer 400 - AlphaServer 4100	2	2
AlphaServer DS10/DS10L/DS15	2	2
AlphaServer DS20/DS25	4	4
AlphaServer ES40/ES45	6	8
AlphaServer GS80	10	16
AlphaServer GS160	18	32
AlphaServer GS320	34	48

When starting, the CHARON-AXP software checks the available number of host CPU cores. This check is based on the maximum number of AXP CPUs that can be emulated if this number is not restricted by the "n_of_cpus" parameter. If the available number of host CPU cores is below this number, CHARON-AXP will issue a warning message even if the requirements for the configured number of AXP CPUs are fulfilled. The CHARON-AXP software will work despite this warning if the requirements for the configured number of AXP CPUs are fulfilled.

Please note:

- **Hyperthreading must be switched off completely**. Disable hyperthreading in the BIOS settings of the physical host or, for a VMware virtual machine, edit the virtual machine properties, select the Resources tab then select Advanced CPU. Set the Hyperthreaded Core Sharing mode to *None*.

CPU type and speed

Platform emulation is a complex and CPU-intensive task. The performance of the individual cores of the host processors is the single most important factor which determines the emulated CPU performance. For the best performance, Stromasys recommends using the fastest available x86_64 processors of the latest generation. At the time of writing, the current generation of Intel server processors is "4th Gen Intel Xeon® Scalable Processors". The current generation of AMD server processors is "4th Generation AMD EPYC™ Processors".

There is a trade-off between the number of cores a processor provides versus the base (or "sustained all-core") frequency – the higher the number of cores, the lower the frequency. In order to choose the best processor, you must understand the workload you wish to emulate. If your workload can be easily distributed across a large number of CPUs ("multi-threaded"), a host processor with a higher number of cores running at a lower frequency might be appropriate. If, on the other hand, your workload conists of single-threaded tasks, probably a host processor with a low number of cores running at the highest possible frequency would be best.

Both Intel and AMD processors have the capability of increasing the frequency on one or very few cores when the other cores are idle. Intel calls this "Max Turbo Frequency" and AMD calls it "Max. Boost Clock". Because the emulator consists of many threads that all need their own host CPUs, this single/few CPU turbo/boost frequency is of no consequence. You should consider only the "Base Frequency/Clock", or, if a processor is capable, the "all-core turbo/boost speed", which is the frequency that all cores can reach when under load.

in general we recommend that the CPU frequency be 3 GHz or higher.

Memory

The minimum host memory size depends on the amount of Alpha memory to be emulated and the number of CHARON-AXP instances to be run on one host.
The minimum host memory is calculated according to the following formula:

> minimum host memory = (2Gb + the amount of Alpha memory emulated) per CHARON-AXP instance.

Disk storage

The total amount of disk space required for CHARON-AXP can be calculated as a sum of all the disk/tape image sizes plus 500 MB for the CHARON software plus space required for the normal host OS. Temporary disk storage is often needed when setting up a new server, for saveset storage, software installation kits, etc.

Ethernet adapters

CHARON-AXP networking requires dedicated host Ethernet adapters; their number must be equal to the emulated adapters to be configured in CHARON-AXP. One adapter (optionally) can be left to the host for TCP/IP networking etc.

Software Requirements

- Hypervisors
  - VMware ESXi 5.5 and 6.x up to 8.0 (requires a supported Windows operating system on top of an ESXi virtual machine
  - Microsoft Hyper-V (requires a supported Windows operating system on top of a Hyper-V virtual machine)
  - kvm
  - Nutanix AHV
- Microsoft Windows Server
  - Microsoft Windows Server 2022
  - Microsoft Windows Server 2019
  - Microsoft Windows Server 2016 Standard and Datacenter Editions, 64 bit version
- Deprecated Microsoft Windows Server (***no longer supported by Microsoft – Stromasys strongly discourages their use***)
  - Microsoft Windows Server 2012 R2 Standard and Datacenter Editions, 64 bit version (please notice that only R2 revision is supported)
  - Microsoft Windows Server 2008 R2 (SP1) Standard and Enterprise Editions, 64 bit version (please notice that only R2 revision is supported)
- Microsoft Windows Workstation
  - Windows 11 Professional and Enterprise Editions, 64 bit version
  - Windows 10 Professional and Enterprise Editions, 64 bit version
- Deprecated Microsoft Windows Workstation (***no longer supported by Microsoft – Stromasys strongly discourages their use***)
  - Windows 8.1 Professional Edition, 64 bit version
  - Microsoft Windows 7 Professional and Ultimate (SP1) Editions, 64 bit version

Host system preparation

Important points:

- All antivirus, screen saver, automatic (scheduled) backup and any other CPU consuming software as well as software that is able to temporarily lock CHARON files (such as automatic indexing) must be turned off, uninstalled and disabled. Alternatively it is also possible to exclude all CHARON executables and material folders from scanning for viruses, but please note that scanning slows down CHARON host anyway - it may be critical for some installations requiring maximum performance.
- The power scheme must be set to "High Performance" with all the "Sleep" and "Standby" modes turned off.
- The automatic installation of Microsoft updates must be disabled. MS updates to the CHARON host must be done only in specific service maintenance periods established by the system administrator. Before applying new updates one must shutdown the operating system running on CHARON and stop all the running CHARON instances and services.

If a network-wide license (red dongle or software license) is going to be used, do the following:

- *On host that will act as network licence server (where the network license will reside)*: open port 1947 for both TCP and UDP
- *On the client*, if broadcast search for remote licenses is to be used, UDP traffic from port 1947 of the license server to ports 30000-65535 of the client must be permitted.
- *On server and client*: set default gateway

Please consult the Windows online documentation for details.

If stricter firewall rules are required, it is possible to open the ports 30000-65535 and 1947 only for the "Sentinel HASP License Manager" (hasplms.exe) service (installed by CHARON-AXP).

If any magneto-optical (MO) drive installed on the host system is going to be used with CHARON (mapped as "\\.\PhysicalDrive<N>" to CHARON emulated disk controllers) the host "MediaChangeNotification" (MSN) service must be switched off manually for these drives according to the following procedure:

1. Type "regedit" in the search field under "Start" menu or press Windows+R and enter "regedit"; press Enter to run the program.
2. Find the "AlwaysDisableMCN" parameter in the following path:

```
HKEY_LOCAL_MACHINE
 + SYSTEM
    + CurrentControlSet
       + Enum
          + SCSI
             + Disk&Ven_DEC&Prod_RWZ53_____(C)DEC  <- This one depends on Vendor and Model of your MO drive
                + 5&fd233cf&0&000500 <- This one depends on hardware connection (SCSI ID of MO drive, location of SCSI HBA on
the host's mainboard, etc ...)
                   + Device Parameters
                      + MediaChangeNotification
                         + AlwaysDisableMCN = DWORD:00000000
```

3. Change the "AlwaysDisableMCN" parameter to "1", for example: "AlwaysDisableMCN = DWORD:00000001".
4. Reboot the host system.

Before installation

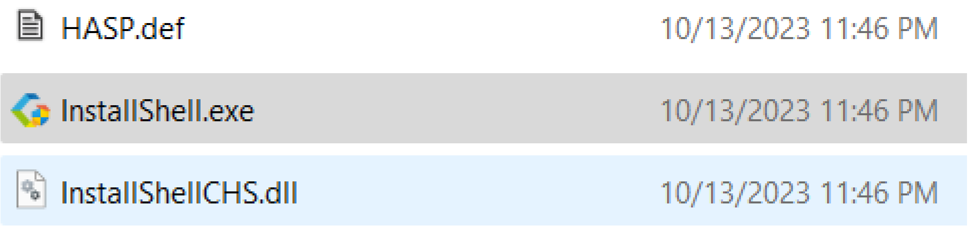
1. Login as the local system administrator ("Administrator") on the host system.
2. [optional] Create a special user for running CHARON-AXP. This user must have all the administrative privileges. Please consult with Windows online documentation for details.
   Please note:
   The CHARON installation procedure will create a special group called CHARON-GRP and the current user will be automatically included to this group. Do not remove this group and do not remove any CHARON user from it, otherwise a problem with virtual memory allocation may appear on CHARON startup.
   The created user may belong to some domain, but please note that in this case you have to add this user to the CHARON-GRP manually as described in this article, and then reboot the CHARON host.
3. Stay logged in as local system administrator ("Administrator") or log off and log in as the CHARON-AXP user having administrative privileges (if this option has been chosen).
4. Create a special directory for the CHARON-AXP distribution kit and copy the provided files there.

Upgrade cases

From version	Action
4.7 and below	First uninstall all CHARON products.
4.8 and above	The version 4.12 can be installed without uninstalling the previous version(s). **Before upgrading**, please run the Network Control Center and take note of all network interfaces dedicated to Charon. After the upgrade, they will have to be dedicated again to Charon as they will all appear released to the host. **Please note** it is mandatory to update all existing CHARON Virtual Machine to make them run the latest version using the Virtual Machine Manager, VM Configuration tab - otherwise they will remain at the previous installed version.

Distribution preparation

- Extract the content of the distribution archive to the current directory or copy the content of the provided distribution directory to the current directory.
  ⚠ The kit folder must not be the same as the target installation folder (Example: C:\Program Files\CHARON\Build_20410) otherwise the installation procedure may report "Charon is in use".
- Run the "InstallShell.exe" file:

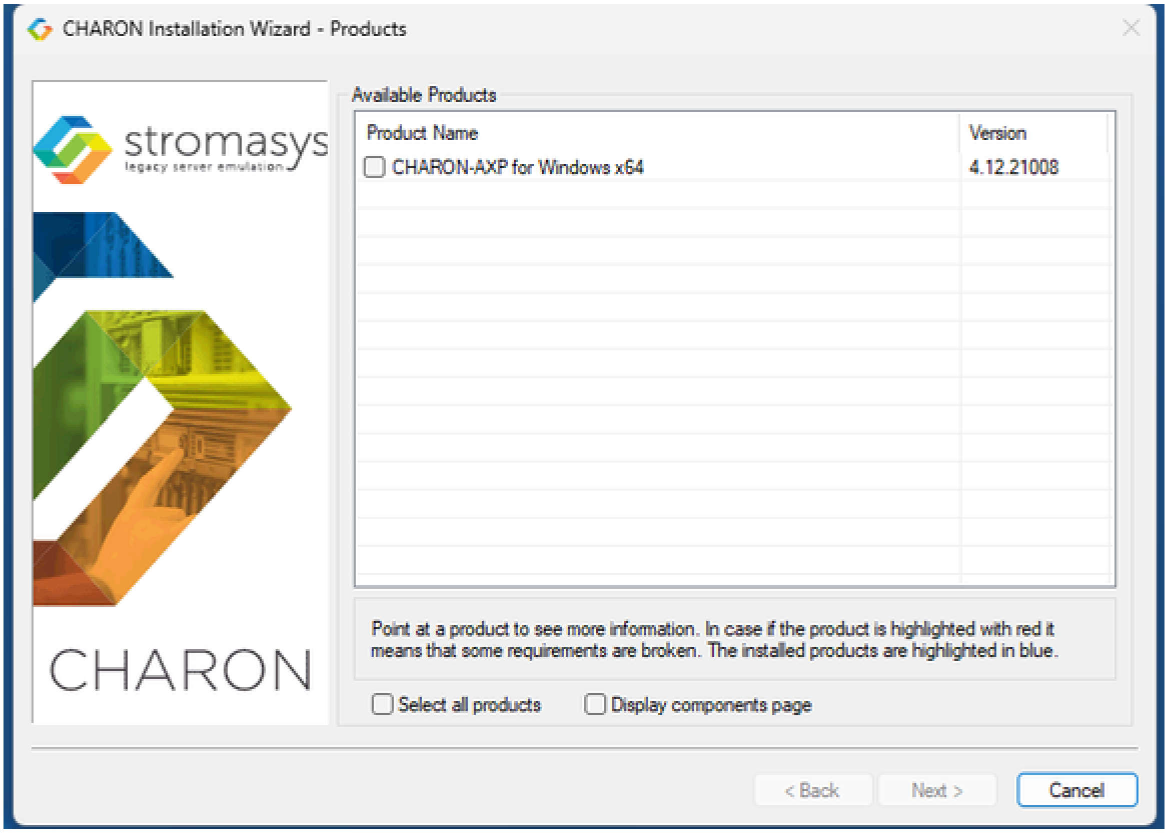


```
HASP.def              10/13/2023 11:46 PM
InstallShell.exe      10/13/2023 11:46 PM
InstallShellCHS.dll   10/13/2023 11:46 PM
```

Installation

The following description assumes that this is the first installation of CHARON-AXP on the target host. Installation of additional CHARON products follows the same procedure.

The first dialog lists the CHARON-AXP product:



CHARON Installation Wizard - Products

Available Products

Product Name	Version
□ CHARON-AXP for Windows x64	4.12.21008

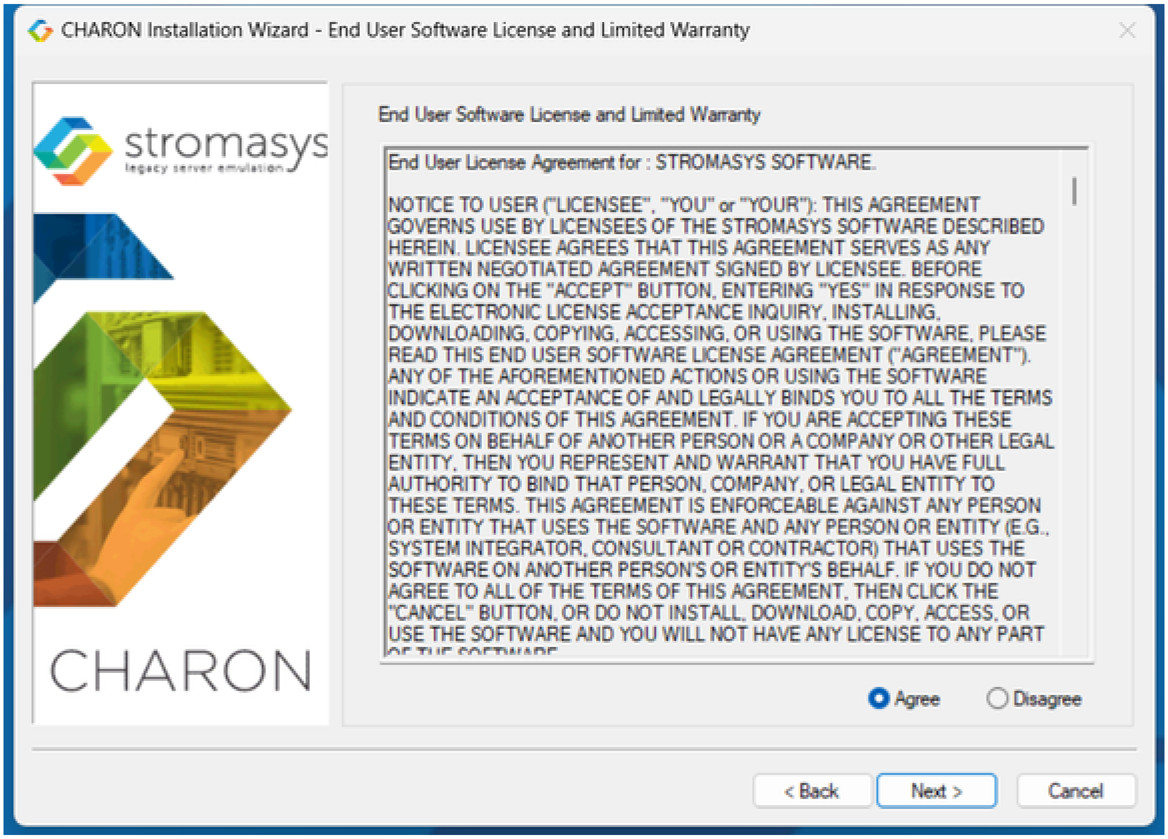
Point at a product to see more information. In case if the product is highlighted with red it means that some requirements are broken. The installed products are highlighted in blue.

□ Select all products □ Display components page

Select the CHARON-AXP product, or all products, by checking the corresponding box. Select the "Display components page" to include the products components page if you need to choose individual components. Stromasys recommends installing all components.

Please note: the CHARON products may be highlighted in red color. This indicates that the CHARON host configuration does not meet CHARON minimal requirements. To see what requirements are not met, point the mouse cursor on that product and a pop-up box with details will appear.

Press the "Next" button.



CHARON Installation Wizard - End User Software License and Limited Warranty
stromasys
legacy server emulation
CHARON
End User Software License and Limited Warranty

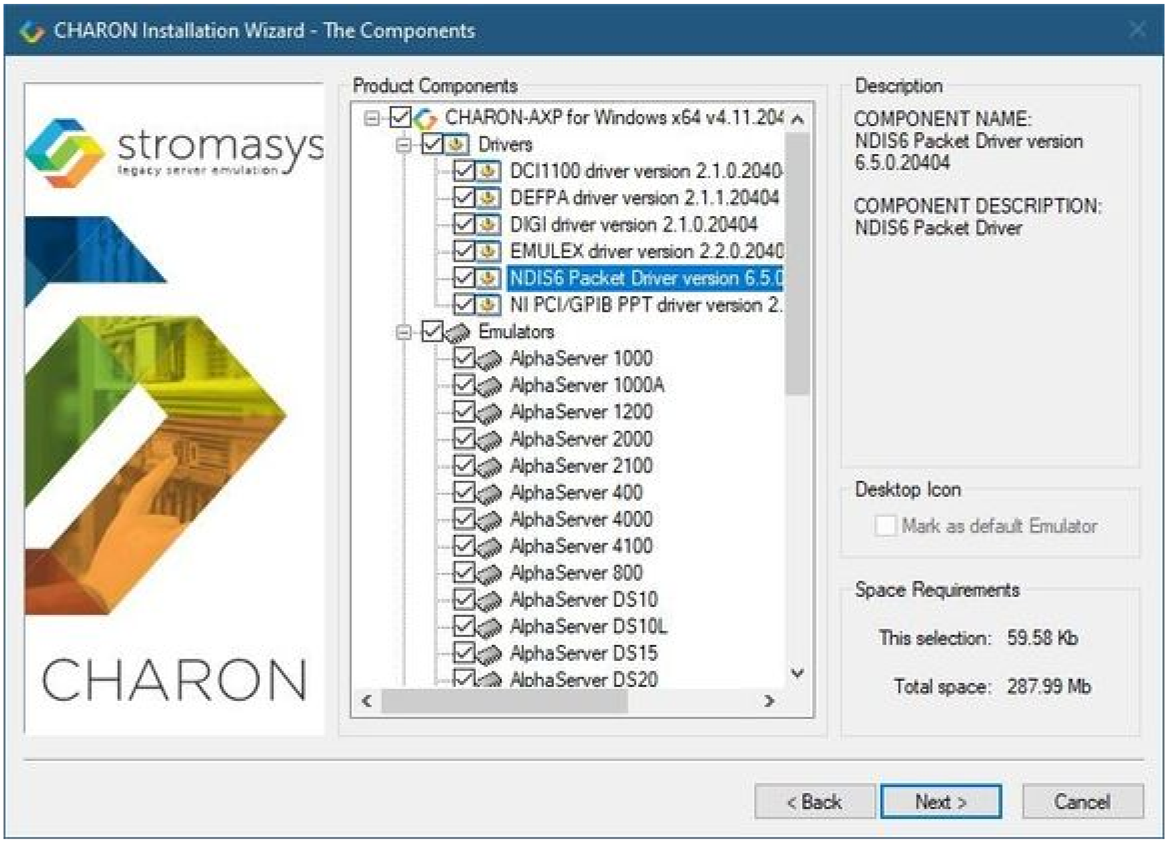
End User License Agreement for : STROMASYS SOFTWARE.

NOTICE TO USER ("LICENSEE", "YOU" or "YOUR"): THIS AGREEMENT GOVERNS USE BY LICENSEES OF THE STROMASYS SOFTWARE DESCRIBED HEREIN. LICENSEE AGREES THAT THIS AGREEMENT SERVES AS ANY WRITTEN NEGOTIATED AGREEMENT SIGNED BY LICENSEE. BEFORE CLICKING ON THE "ACCEPT" BUTTON, ENTERING "YES" IN RESPONSE TO THE ELECTRONIC LICENSE ACCEPTANCE INQUIRY, INSTALLING, DOWNLOADING, COPYING, ACCESSING, OR USING THE SOFTWARE, PLEASE READ THIS END USER SOFTWARE LICENSE AGREEMENT ("AGREEMENT"). ANY OF THE AFOREMENTIONED ACTIONS OR USING THE SOFTWARE INDICATE AN ACCEPTANCE OF AND LEGALLY BINDS YOU TO ALL THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF YOU ARE ACCEPTING THESE TERMS ON BEHALF OF ANOTHER PERSON OR A COMPANY OR OTHER LEGAL ENTITY, THEN YOU REPRESENT AND WARRANT THAT YOU HAVE FULL AUTHORITY TO BIND THAT PERSON, COMPANY, OR LEGAL ENTITY TO THESE TERMS. THIS AGREEMENT IS ENFORCEABLE AGAINST ANY PERSON OR ENTITY THAT USES THE SOFTWARE AND ANY PERSON OR ENTITY (E.G., SYSTEM INTEGRATOR, CONSULTANT OR CONTRACTOR) THAT USES THE SOFTWARE ON ANOTHER PERSON'S OR ENTITY'S BEHALF. IF YOU DO NOT AGREE TO ALL OF THE TERMS OF THIS AGREEMENT, THEN CLICK THE "CANCEL" BUTTON, OR DO NOT INSTALL, DOWNLOAD, COPY, ACCESS, OR USE THE SOFTWARE AND YOU WILL NOT HAVE ANY LICENSE TO ANY PART OF THE SOFTWARE

● Agree ○ Disagree

Read the license, check "Agree" and press the "Next" button.

If the "Display components page" option was selected, the setup procedure will display a dialog for the CHARON-AXP product components:



Product Components

- ☑ CHARON-AXP for Windows x64 v4.11.204
  - ☑ Drivers
    - ☑ DCI1100 driver version 2.1.0.2040
    - ☑ DEFPA driver version 2.1.1.20404
    - ☑ DIGI driver version 2.1.0.20404
    - ☑ EMULEX driver version 2.2.0.2040
    - ☑ NDIS6 Packet Driver version 6.5.0
    - ☑ NI PCI/GPIB PPT driver version 2.
  - ☑ Emulators
    - ☑ AlphaServer 1000
    - ☑ AlphaServer 1000A
    - ☑ AlphaServer 1200
    - ☑ AlphaServer 2000
    - ☑ AlphaServer 2100
    - ☑ AlphaServer 400
    - ☑ AlphaServer 4000
    - ☑ AlphaServer 4100
    - ☑ AlphaServer 800
    - ☑ AlphaServer DS10
    - ☑ AlphaServer DS10L
    - ☑ AlphaServer DS15
    - ☑ AlphaServer DS20

Description

COMPONENT NAME:
NDIS6 Packet Driver version
6.5.0.20404

COMPONENT DESCRIPTION:
NDIS6 Packet Driver

Desktop Icon

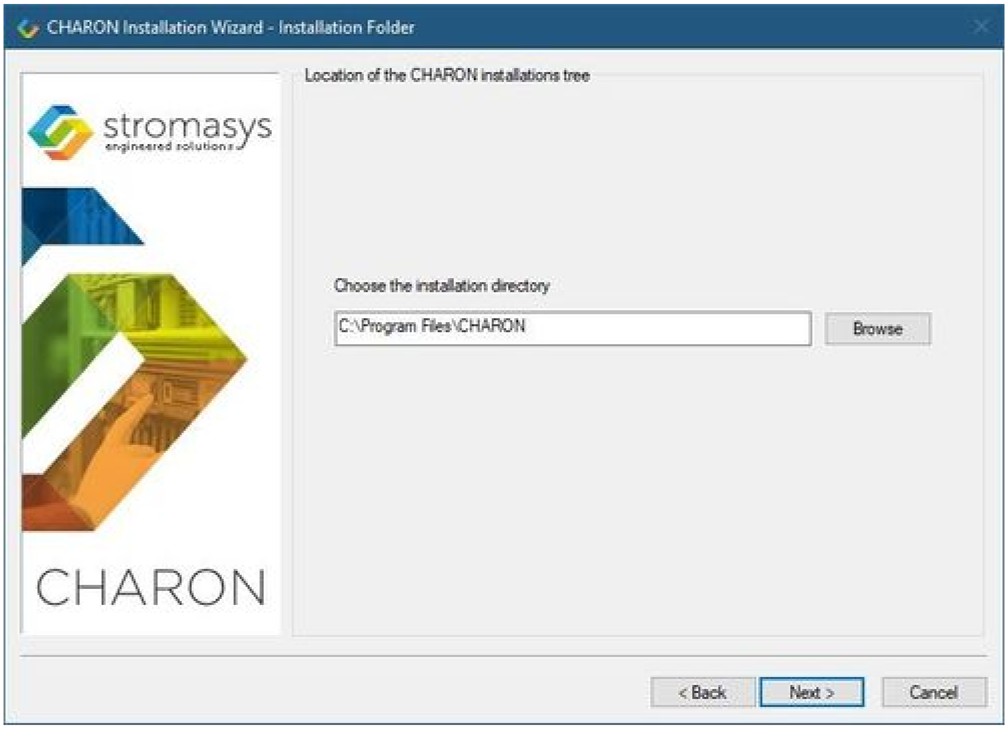
□ Mark as default Emulator

Space Requirements

This selection: 59.58 Kb

Total space: 287.99 Mb

⚠ It is strongly recommended to leave the selection as it is unless you clearly understand what the impact is if you uncheck some modules.



CHARON Installation Wizard - Installation Folder

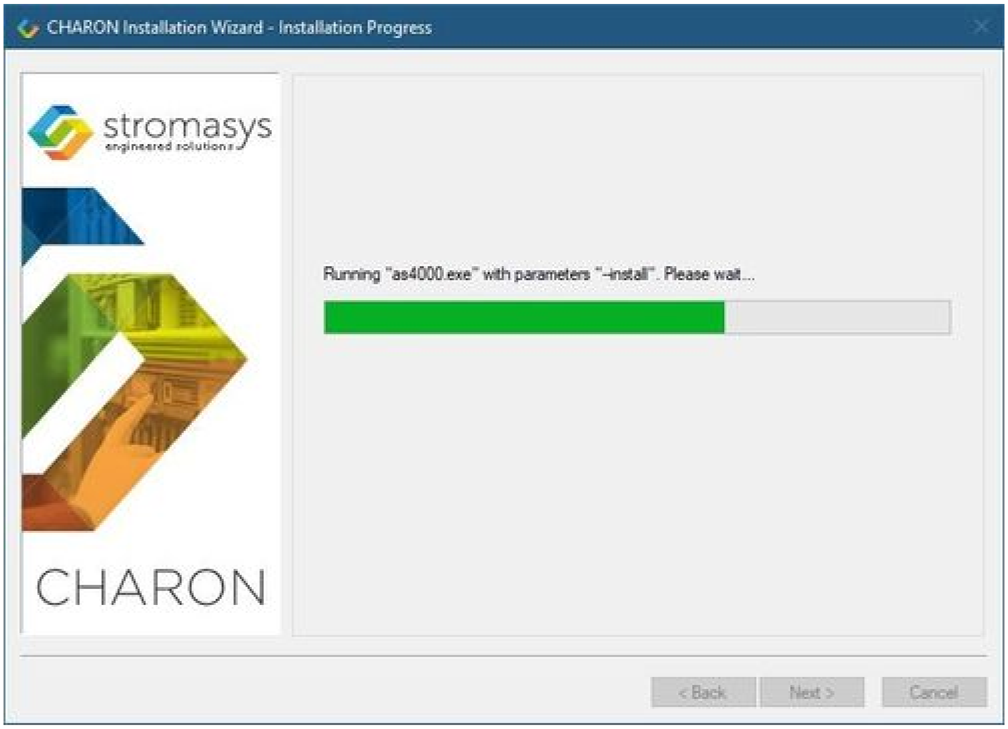
Location of the CHARON installations tree

Choose the installation directory

C:\Program Files\CHARON

Select a directory to be used for the CHARON installation or use the default. The path can either be entered manually or selected using the "Browse" button.

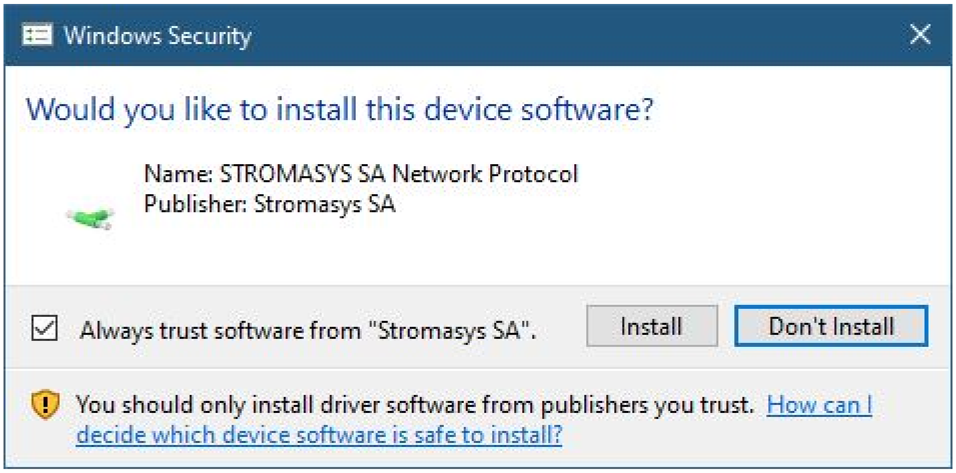
Once done, press the "Next" button to start the installation of the selected components.



CHARON Installation Wizard - Installation Progress

Running "as4000.exe" with parameters "--install". Please wait...

During the CHARON installation procedure, you will have to confirm the installation of the CHARON-specific network driver. Press the "Install" button:



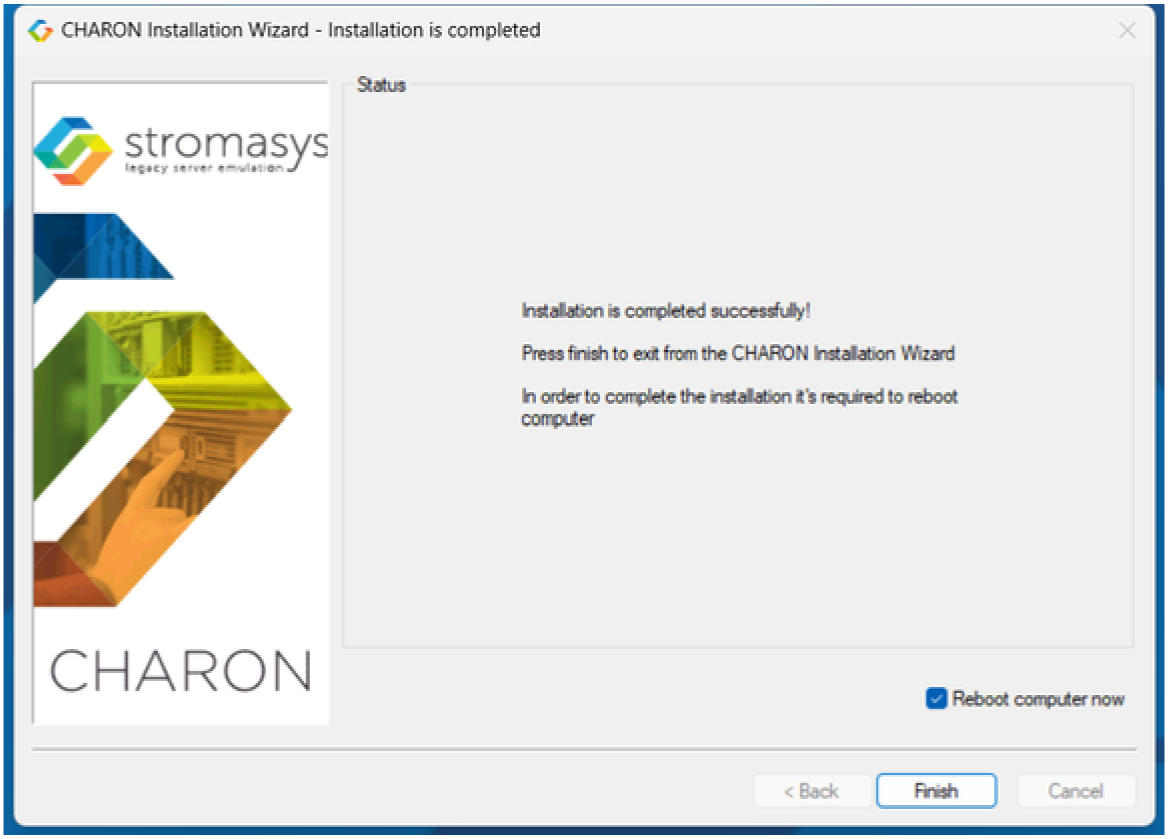
Would you like to install this device software?

Name: STROMASYS SA Network Protocol
Publisher: Stromasys SA

☑ Always trust software from "Stromasys SA".

You should only install driver software from publishers you trust. How can I decide which device software is safe to install?

Once the installation is done, the following dialog will be displayed:



Installation is completed successfully!

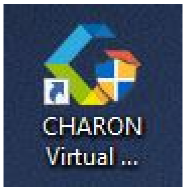
Press finish to exit from the CHARON Installation Wizard

In order to complete the installation it's required to reboot computer

☑ Reboot computer now

⚠ It is strongly recommended to reboot the CHARON host system immediately: leave the "Reboot computer now" box checked and press the "Finish" button.

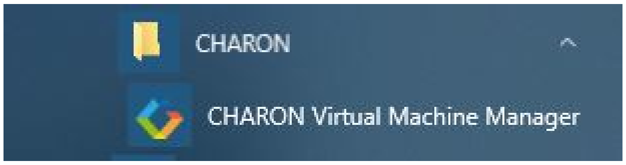
Once the host system is up again, you will notice a new shortcut, "CHARON Virtual Machine Manager", created on the desktop during the installation procedure:



The CHARON Virtual Machine Manager is essential for managing Charon emulator instances. The Charon installation always keeps most up-to-date version of Charon Virtual Machine Manager.

Please note: in case of an upgrade from V4.8, V4.9, V4.10 or V4.11, you must update all existing CHARON virtual machines to allow them to use the latest version - otherwise they will remain at the previous version. See the following chapters for details.

The CHARON installation procedure creates the following simple structure under the "Start" menu:



CHARON-AXP home directory

By default CHARON-AXP is installed in the "C:\Program Files\CHARON" directory. It has the following sub-directories:

Directory	Description
Build_XXX\x64	Contains product executables and libraries
Charon License Utility*	Contains License Management related utilities : to check expiration, to view content, to update license content
Drivers	Contains the CHARON drivers
InstallShell	Contains the CHARON installation procedure executables
Logs	Contains the CHARON installation log
NCC	Contains the Charon Network Control Center executable
Redistributables	Contains the redistributables needed for CHARON running (already installed, except the license alert script)
Reports *	Contains files related to build a report as an html file and collect this report with various useful files to analyze the installation : config and logs files
Virtual Disk Images	Contains CHARON virtual disk images. For example "charoncp013.vdisk" and some OpenVMS tools (ovms_tools.vdisk) used for debugging plus the PCI I/O Bypass controller
Virtual Machine Manager	Contains the most up-to-date version of CHARON Virtual Machine Manager
Virtual Machine Templates	Contains templates for creating CHARON virtual machines
Virtual Machines	Contains a default directory to store the CHARON virtual machines

Folders indicated with a * : will be created and populated if you have installed previously some additional tools known as Charon Toolkit Components.

The "Virtual Machine Templates" directory contains template configuration files to build CHARON virtual machines using the CHARON Virtual Machine Manager. We will focus our attention on this subject in the next chapter.

License installation

Regular HASP USB dongle

The license is provided on a dedicated USB device ("dongle"). Simply plug the dongle into one of the USB sockets on the emulator host.

Please note:

- If the CHARON host is accessed remotely, please note that the contents of a regular HASP license cannot be displayed using RDP. ILO or iDRAC, some other console-capable program must be used. Network HASP (red dongles) licenses have no restrictions with respect to remote access.
- Microsoft Hyper-V is not capable of passing USB devices through to virtual machines, which includes HASP license dongles.

Network HASP USB dongle

If the CHARON license is a network license (red USB dongle), it is possible either to connect it to the host USB port (to use it locally and provide licenses to other hosts on the local network at the same time) or to install it on a local network "license server" for remote access from this particular host.

If a remote license server is to be used:

- Copy the file "`hasp_install\haspdinst.zip`" from the CHARON distribution to any directory on the server, for example "`C:\Temp`".
- Extract the contents of this archive to the same directory.
- Login as "*Administrator*" on the server and open "`cmd.exe`" from the "Start" menu.
- Switch to that directory.
- Install the extracted file:
  Example:

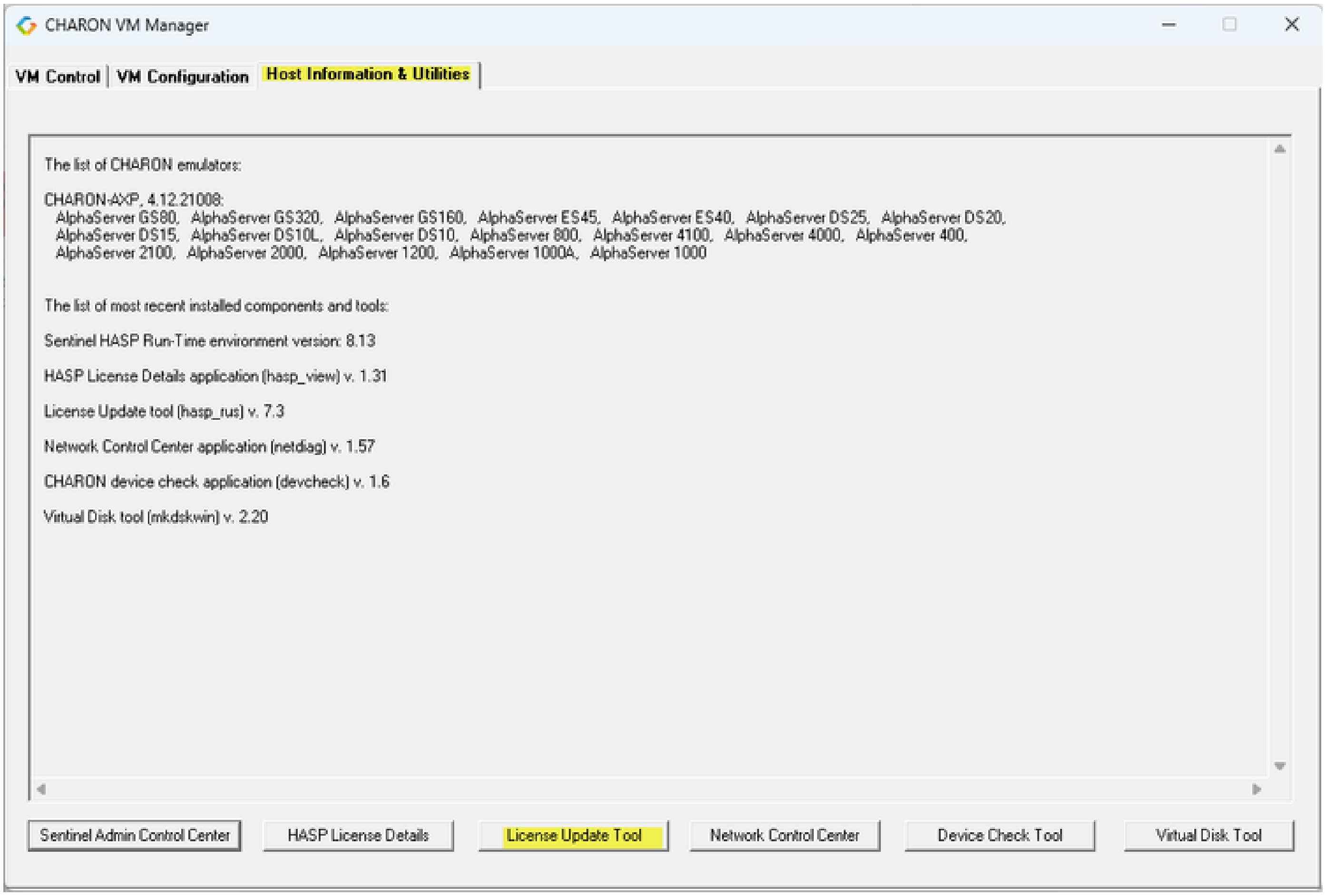
  ```
  ...> cd c:\temp
  c:\temp> haspdinst.exe -fr -kp -nomsg
  c:\temp> haspdinst.exe -install -cm
  ```

- Copy the file "`hasp_install\hasplib.zip`" from the CHARON distribution to any directory on the server, for example "`C:\Temp`".
- Extract the contents of this archive to the same directory.
- Copy the file "`haspvlib_68704.dll`" to "`C:\Program Files (x86)\Common Files\Aladdin Shared\HASP`" (in case of x64 host) or "`C:\Program Files\Common Files\Aladdin Shared\HASP`" (in case of x86 host)
- Connect the network HASP dongle to the server USB port.

Software license

If the CHARON license is a software license (SL), it is installed on the host using the following procedure:

1. Double click the "CHARON Virtual Machine Manager" icon on the desktop or select this utility in the tray menu; select the "Host Information and Utilities" section and press the "License Update Tool" button:



The list of CHARON emulators:

CHARON-AXP, 4.12.21008:
AlphaServer GS80, AlphaServer GS320, AlphaServer GS160, AlphaServer ES45, AlphaServer ES40, AlphaServer DS25, AlphaServer DS20, AlphaServer DS15, AlphaServer DS10L, AlphaServer DS10, AlphaServer 800, AlphaServer 4100, AlphaServer 4000, AlphaServer 400, AlphaServer 2100, AlphaServer 2000, AlphaServer 1200, AlphaServer 1000A, AlphaServer 1000

The list of most recent installed components and tools:

Sentinel HASP Run-Time environment version: 8.13

HASP License Details application (hasp_view) v. 1.31

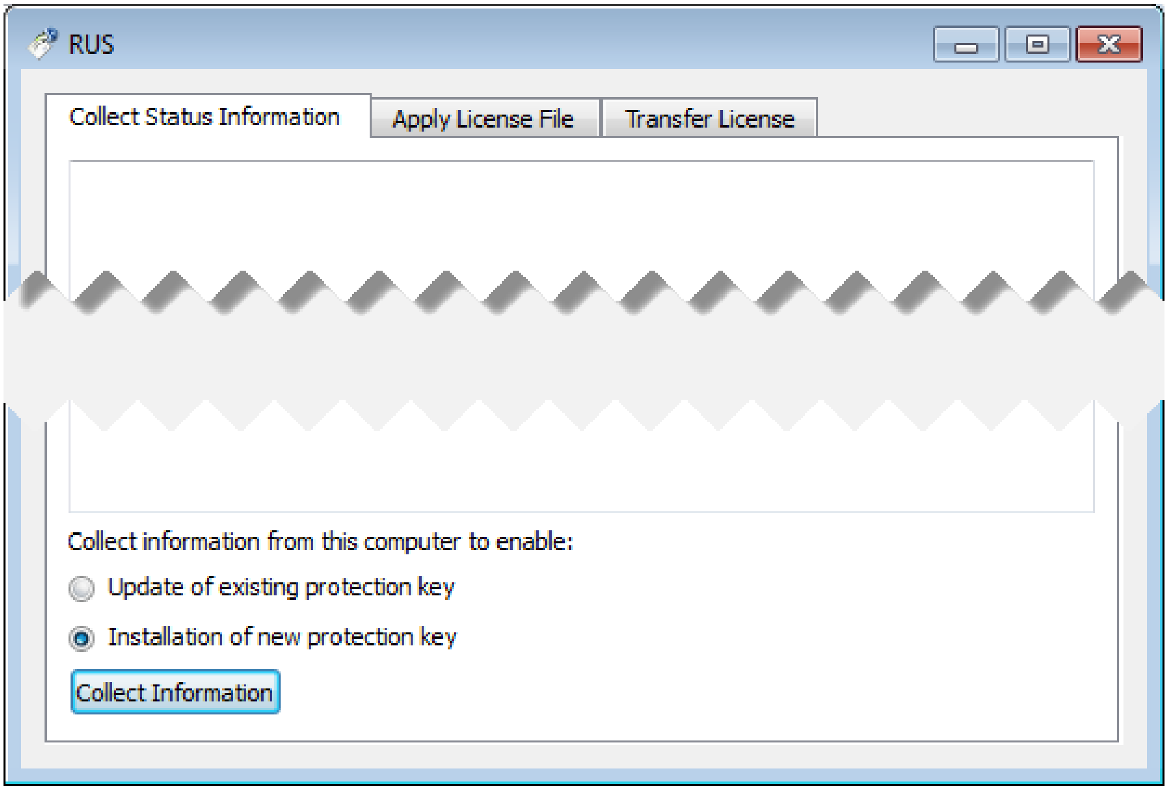
License Update tool (hasp_rus) v. 7.3

Network Control Center application (netdiag) v. 1.57

CHARON device check application (devcheck) v. 1.6

Virtual Disk tool (mkdskwin) v. 2.20

The following default dialog window will appear:



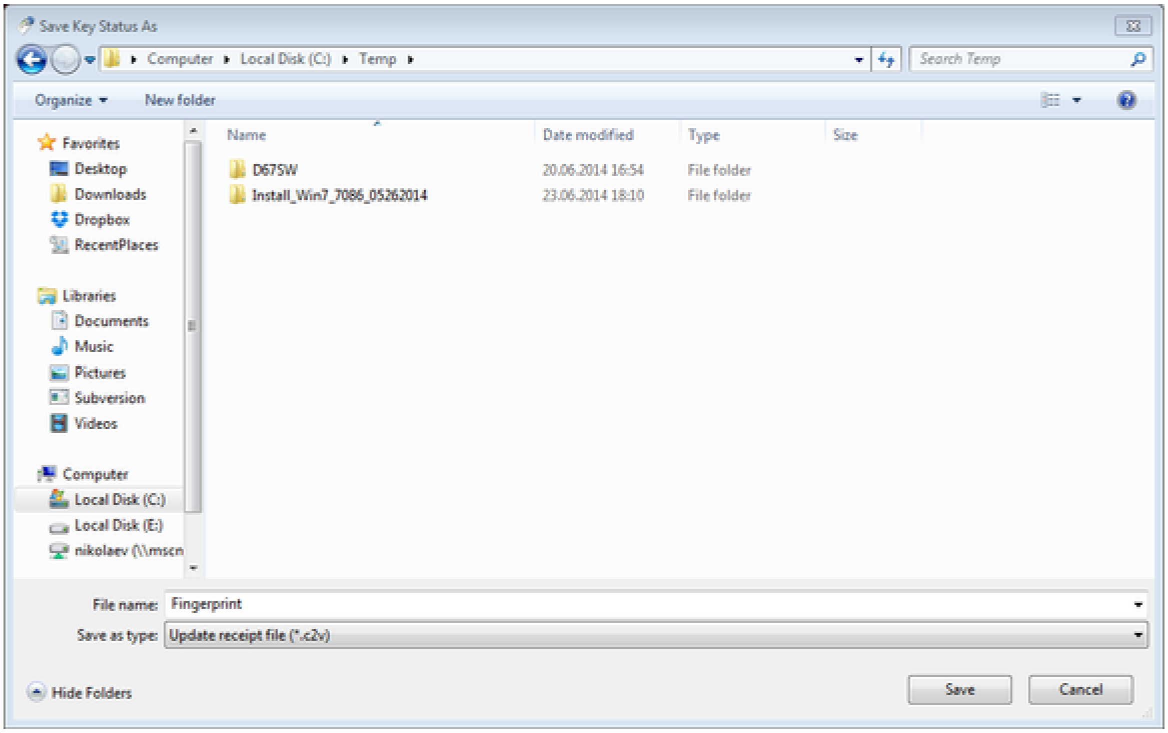
Collect information from this computer to enable:

○ Update of existing protection key

● Installation of new protection key

If you're installing a "Provisional" (demo) license, jump to step 5.

2. Ensure no license dongle is connected, select the "Installation of new protection key" radio-box and press the "Collect Information" button. The following window will appear:



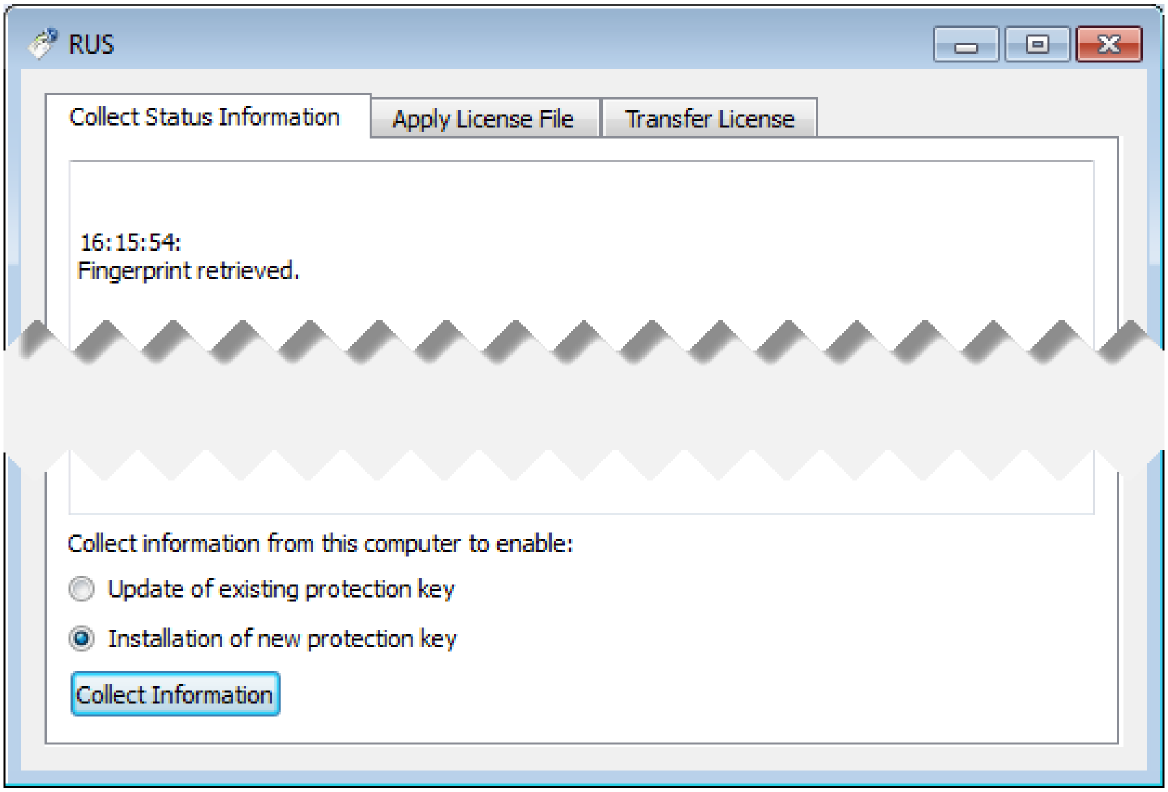
Name	Date modified	Type	Size
D675W	20.06.2014 16:54	File folder	
Install_Win7_7086_05262014	23.06.2014 18:10	File folder	

File name: Fingerprint

Save as type: Update receipt file (*.c2v)

Specify the directory and name of the system "fingerprint" file to be created the press the "Save" button.

3. The "License Update Service" will save the fingerprint file under the given name and will report the status:



16:15:54:
Fingerprint retrieved.

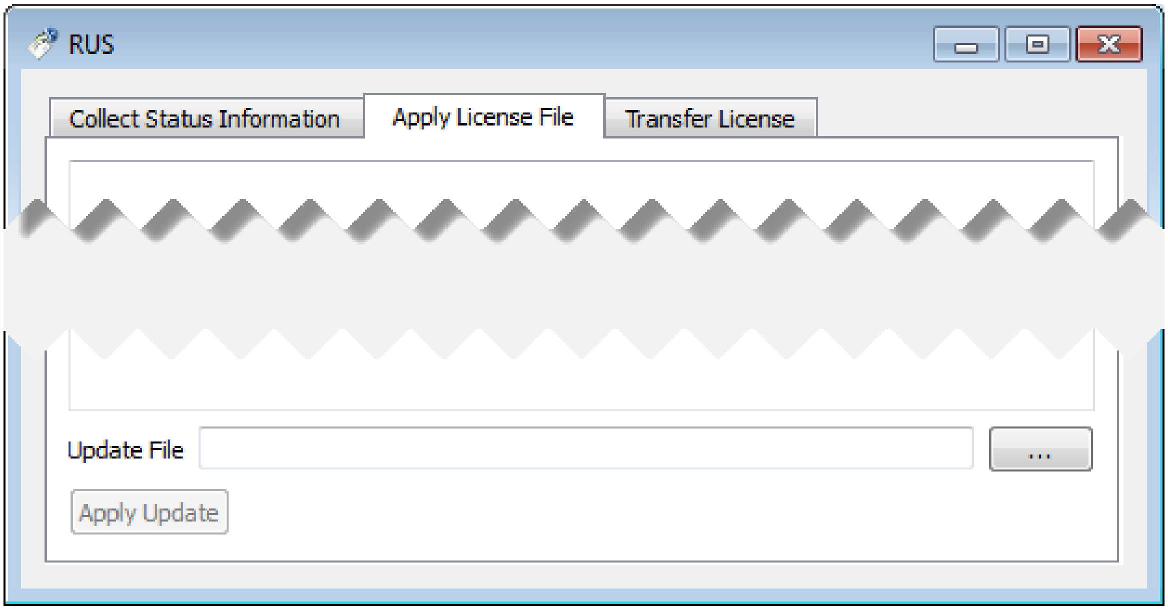
Collect information from this computer to enable:

○ Update of existing protection key

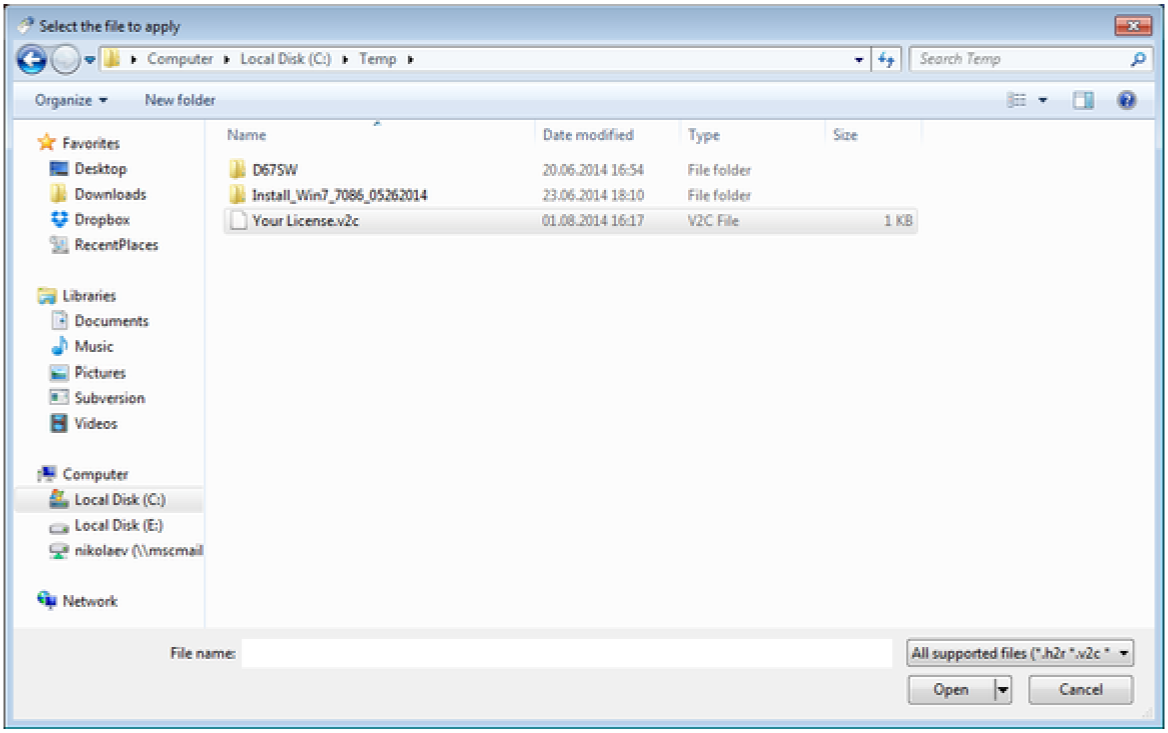
● Installation of new protection key

4. Send the resulting file to Stromasys. In return Stromasys will provide you with a ".v2c" file, for example "Your License.v2c".

5. Copy the received v2c file to the CHARON host in a folder of your choice and select the "Apply License File" tab:



6. Press the "..." button and browse for the received v2c file.
   Example:



Name	Date modified	Type	Size
D675W	20.06.2014 16:54	File folder	
Install_Win7_7086_05262014	23.06.2014 18:10	File folder	
Your License.v2c	01.08.2014 16:17	V2C File	1 KB

File name:

All supported files (*.h2r *.v2c *

   Press the "Open" button to apply the license.

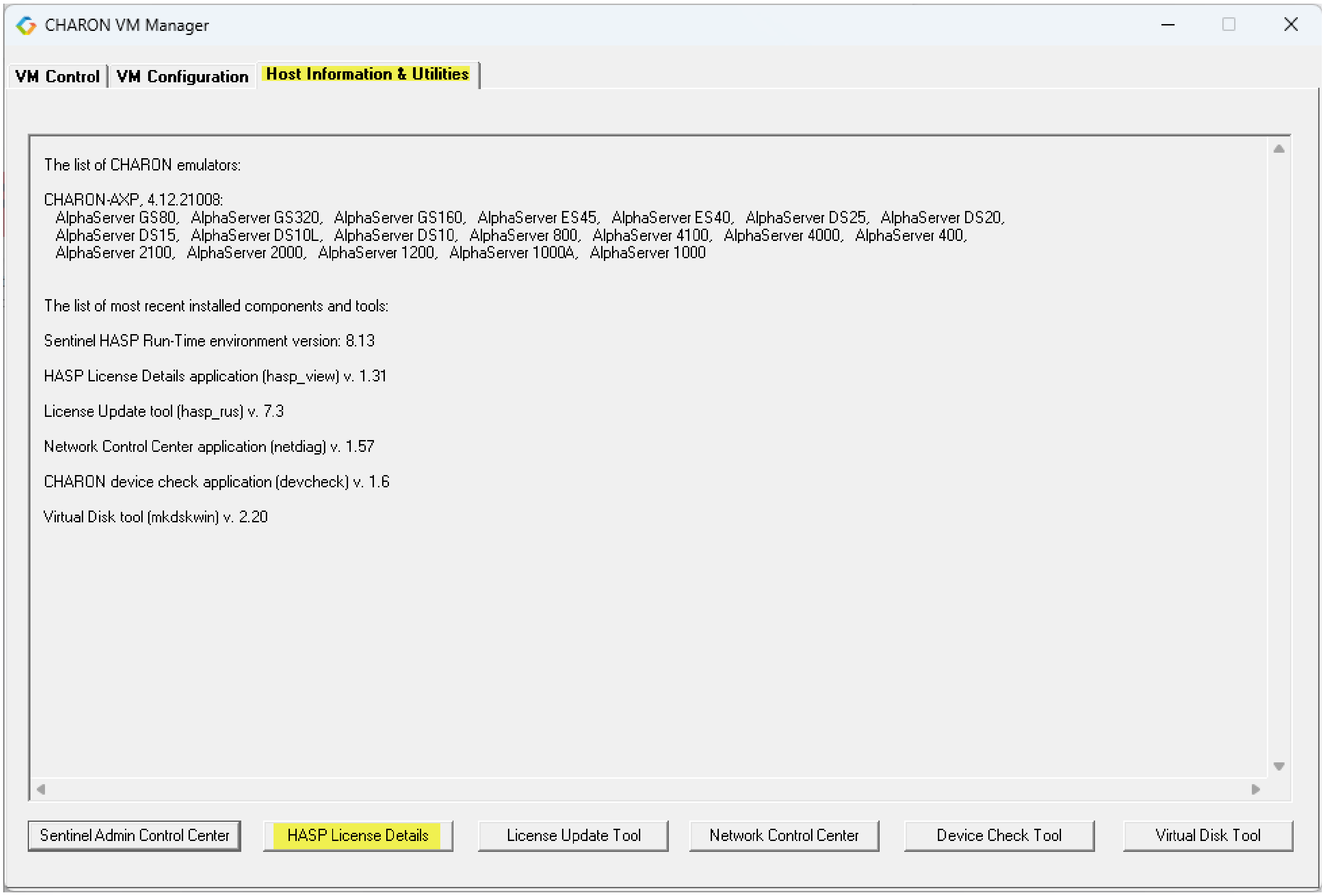
7. From the "CHARON Virtual Machine Manager" utility (see above) select the "Host Information and Utilities" section and press the "Sentinel Admin Control Center" button or (optionally) invoke the system default web browser and enter the URL http://localhost:1947 to display the "**Sentinel Admin Control Center**" (**ACC**) web interface. This interface allows you to view and manage the CHARON licenses.

Ensure that the software license is now visible in the "**Sentinel Keys**" section of the **ACC**.

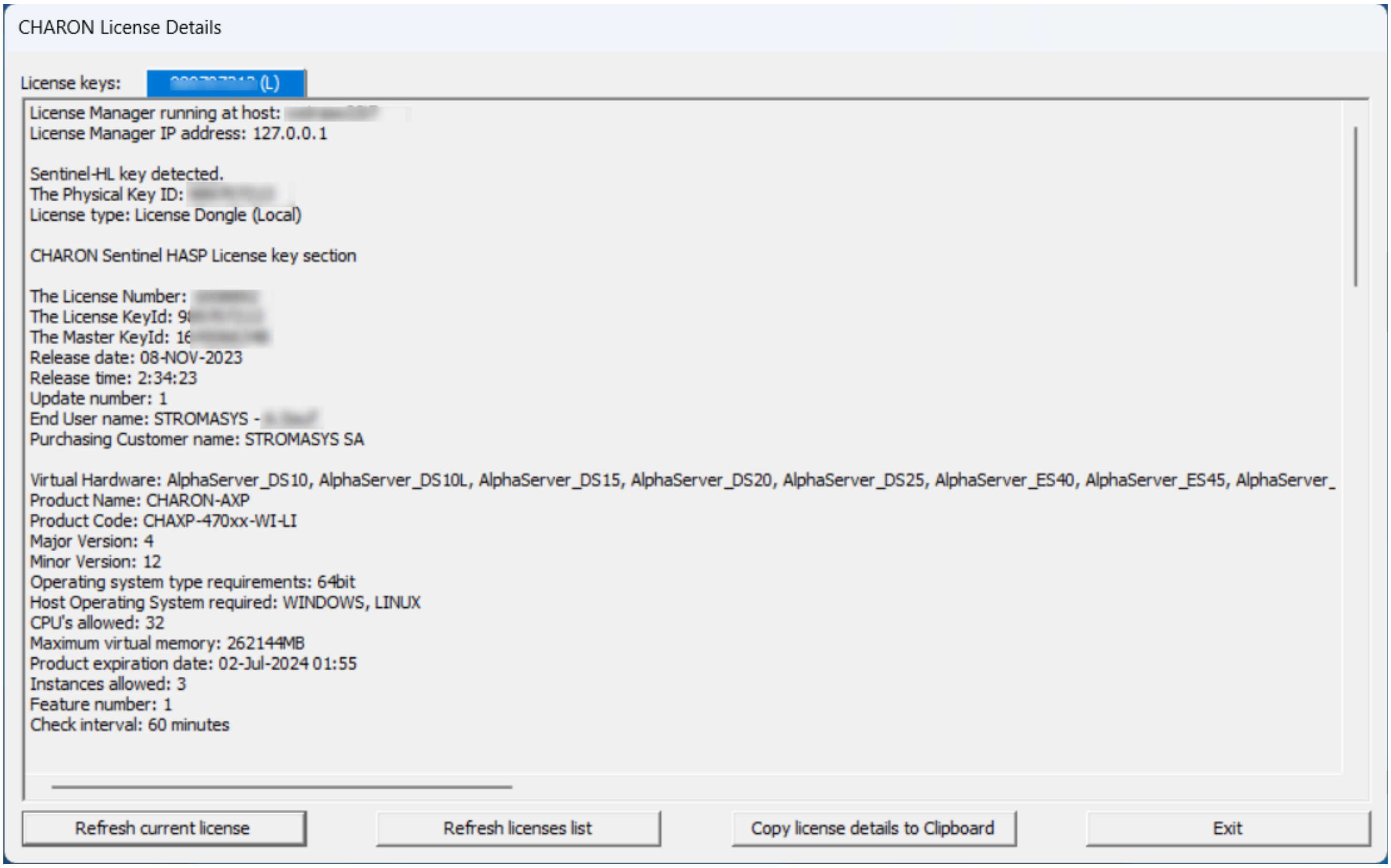
Please note: software licenses are always network licenses. Hence, they have no restrictions with respect to being displayed or accessed via a remote connection.

License validity verification

Check the CHARON license validity by invoking the "Charon License Details" utility to make sure that the CHARON license is visible and is correct. Doubl e click the "CHARON Virtual Machine Manager" icon on the desktop or select this utility in the tray menu; select the "Host Information and Utilities" section and press the "HASP License Details" button:



```
The list of CHARON emulators:

CHARON-AXP, 4.12.21008:
  AlphaServer GS80,  AlphaServer GS320,  AlphaServer GS160,  AlphaServer ES45,  AlphaServer ES40,  AlphaServer DS25,  AlphaServer DS20,
  AlphaServer DS15,  AlphaServer DS10L,  AlphaServer DS10,  AlphaServer 800,  AlphaServer 4100,  AlphaServer 4000,  AlphaServer 400,
  AlphaServer 2100,  AlphaServer 2000,  AlphaServer 1200,  AlphaServer 1000A,  AlphaServer 1000


The list of most recent installed components and tools:

Sentinel HASP Run-Time environment version: 8.13

HASP License Details application (hasp_view) v. 1.31

License Update tool (hasp_rus) v. 7.3

Network Control Center application (netdiag) v. 1.57

CHARON device check application (devcheck) v. 1.6

Virtual Disk tool (mkdskwin) v. 2.20
```



CHARON License Details

License keys: [redacted] (L)

```
License Manager running at host: [redacted]
License Manager IP address: 127.0.0.1

Sentinel-HL key detected.
The Physical Key ID: [redacted]
License type: License Dongle (Local)

CHARON Sentinel HASP License key section

The License Number: [redacted]
The License KeyId: [redacted]
The Master KeyId: [redacted]
Release date: 08-NOV-2023
Release time: 2:34:23
Update number: 1
End User name: STROMASYS - [redacted]
Purchasing Customer name: STROMASYS SA

Virtual Hardware: AlphaServer_DS10, AlphaServer_DS10L, AlphaServer_DS15, AlphaServer_DS20, AlphaServer_DS25, AlphaServer_ES40, AlphaServer_ES45, AlphaServer_
Product Name: CHARON-AXP
Product Code: CHAXP-470xx-WI-LI
Major Version: 4
Minor Version: 12
Operating system type requirements: 64bit
Host Operating System required: WINDOWS, LINUX
CPU's allowed: 32
Maximum virtual memory: 262144MB
Product expiration date: 02-Jul-2024 01:55
Instances allowed: 3
Feature number: 1
Check interval: 60 minutes
```

Verify the the license content is correct and no error message is shown

- Verify the content of the license looks correct. Check for example the license number, the major and minor versions, the minimum and maximum build numbers, the CHARON-AXP products and allowed hardware (CHARON-AXP models). More details on the license content can be found in the CHARON-AXP Licensing chapter of this Guide.

Troubleshooting

If the CHARON license content cannot be displayed by the "CHARON License Details" utility or is incorrect, check that the license is available and correctly used:

1. From the "CHARON Virtual Machine Manager" utility (see above) select the "Host Information and Utilities" section and press the "Sentinel Admin Control Center" button or (optionally) invoke the system default web browser and enter the URL http://localhost:1947 to display the "**Sentinel Admin Control Center**" (**ACC**) web interface.
2. Click on the "**Sentinel Keys**" link to open up the "**Sentinel Keys Section**" page
3. Make sure that one and only one CHARON HASP or SL license is present.

ⓘ To facilitate troubleshooting, Stromasys recommends to enable logging from the Sentinel Admin Control Center as described in this article: Enabling logging in Sentinel Admin Control Center.

Problem	Action
No license is displayed	Make sure that all the recommendations above about remote access to the host are fulfilled (if remote access takes place) and the HASP USB key is not broken and its LED indicator is lit (meaning that it is used by the host).
Only one License key / SL is seen and its content is incorrect	Contact STROMASYS to request a new license update.
Several License keys / SLs are displayed	Remove all of them except the one provided by STROMASYS for the installed version of CHARON.

Removing licenses can be done by physical disconnection of the corresponding USB HASP keys from the CHARON host and physical disconnection of the network HASP keys from all hosts on a local network (or by disabling remote access to network licenses from the CHARON host - see detailed explanation below).

Software licenses can also be uninstalled using the method, described in the "Removing CHARON-AXP Software Licenses" chapter of this Guide

For license servers accessible only via non-broadcast search it is also possible to disable access to network licenses if only a local license is to be used: Click on the "**Configuration**" link to open the "**Configuration for Sentinel Manager**" page.

Uncheck the "**Allow Access to Remote Licenses**" checkbox from the "**Access to Remote License Managers**" tab then press the "**Submit**" button to apply changes.

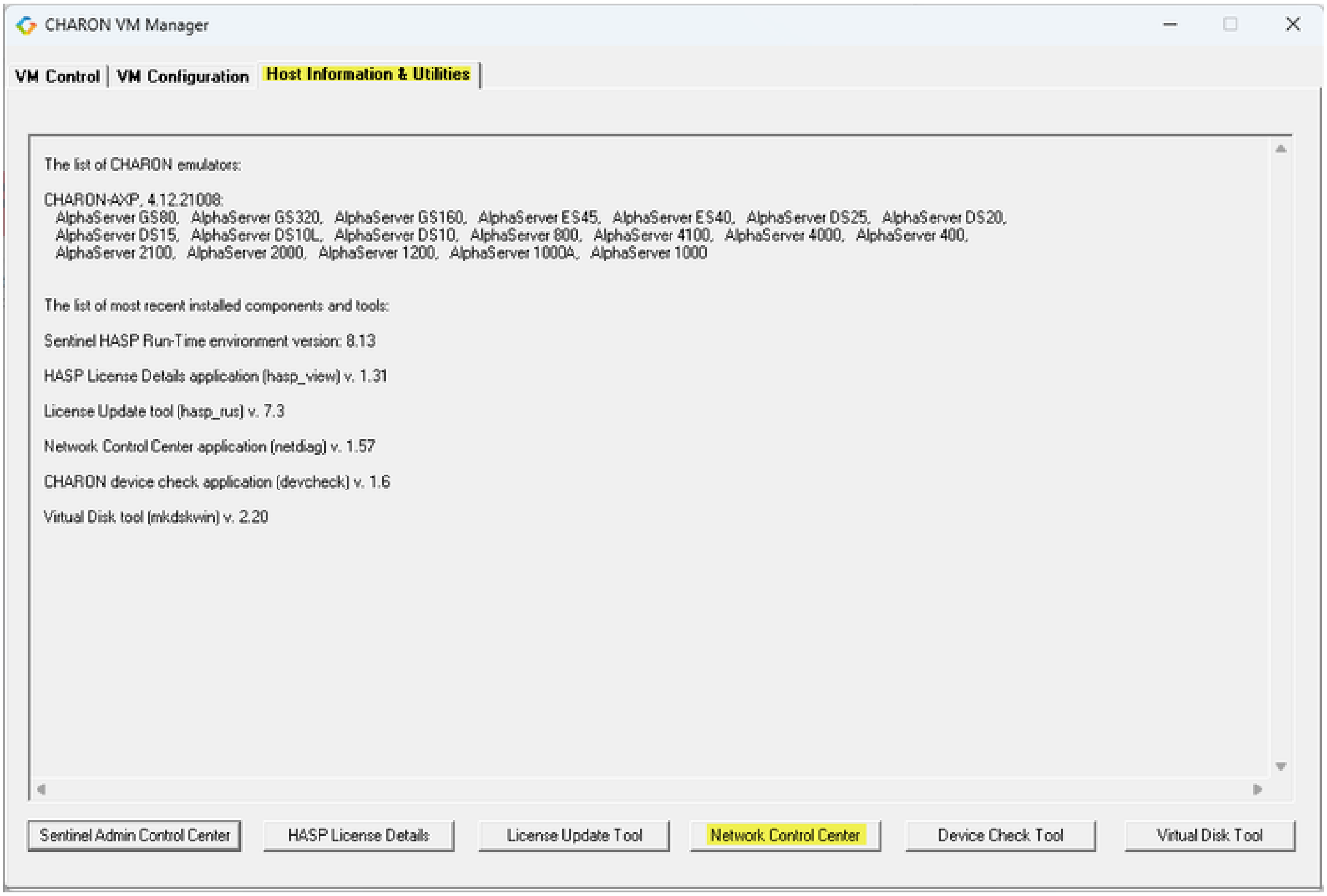
Please note:

- Starting with Charon-AXP/VAX 4.9 for Linux and Charon-AXP/VAX version 4.8 for Windows the Charon emulator products do not follow the settings in the Sentinel ACC with respect to querying remote license servers and network visibility. They perform a **broadcast search** for network licenses even if this has been disabled in the Sentinel ACC. If this behavior has to be prevented for specific reasons, the network access of the system has to be temporarily restricted or disabled, for example by blocking the relevant traffic in a firewall. Another possibility would be to block access to the network license at the license server side.
- Note that such methods can negatively impact other functions of the system or, in the case of blocking access to a network license on the server, even the functions on other license clients.
- It is possible to have several licenses available to CHARON-AXP at the same time. In this case you have to specify in the CHARON-AXP configuration file the license key ids to be used.
  Example:
  ```
  set session license_key_id="1877752571,354850588"
  ```

Network configuration

If the CHARON virtual machine needs access to the network, one or more dedicated network interfaces cleared from any other protocols including TCP/IP will have to be defined.

Configuring the network is done with the "Network Control Center" (NCC) utility. Double click the "CHARON Virtual Machine Manager" icon on the desktop or select this utility in the tray menu; select the "Host Information and Utilities" section and press the "Network Control Center" button:



The list of CHARON emulators:

CHARON-AXP, 4.12.21008:
AlphaServer GS80, AlphaServer GS320, AlphaServer GS160, AlphaServer ES45, AlphaServer ES40, AlphaServer DS25, AlphaServer DS20, AlphaServer DS15, AlphaServer DS10L, AlphaServer DS10, AlphaServer 800, AlphaServer 4100, AlphaServer 4000, AlphaServer 400, AlphaServer 2100, AlphaServer 2000, AlphaServer 1200, AlphaServer 1000A, AlphaServer 1000

The list of most recent installed components and tools:

Sentinel HASP Run-Time environment version: 8.13

HASP License Details application (hasp_view) v. 1.31

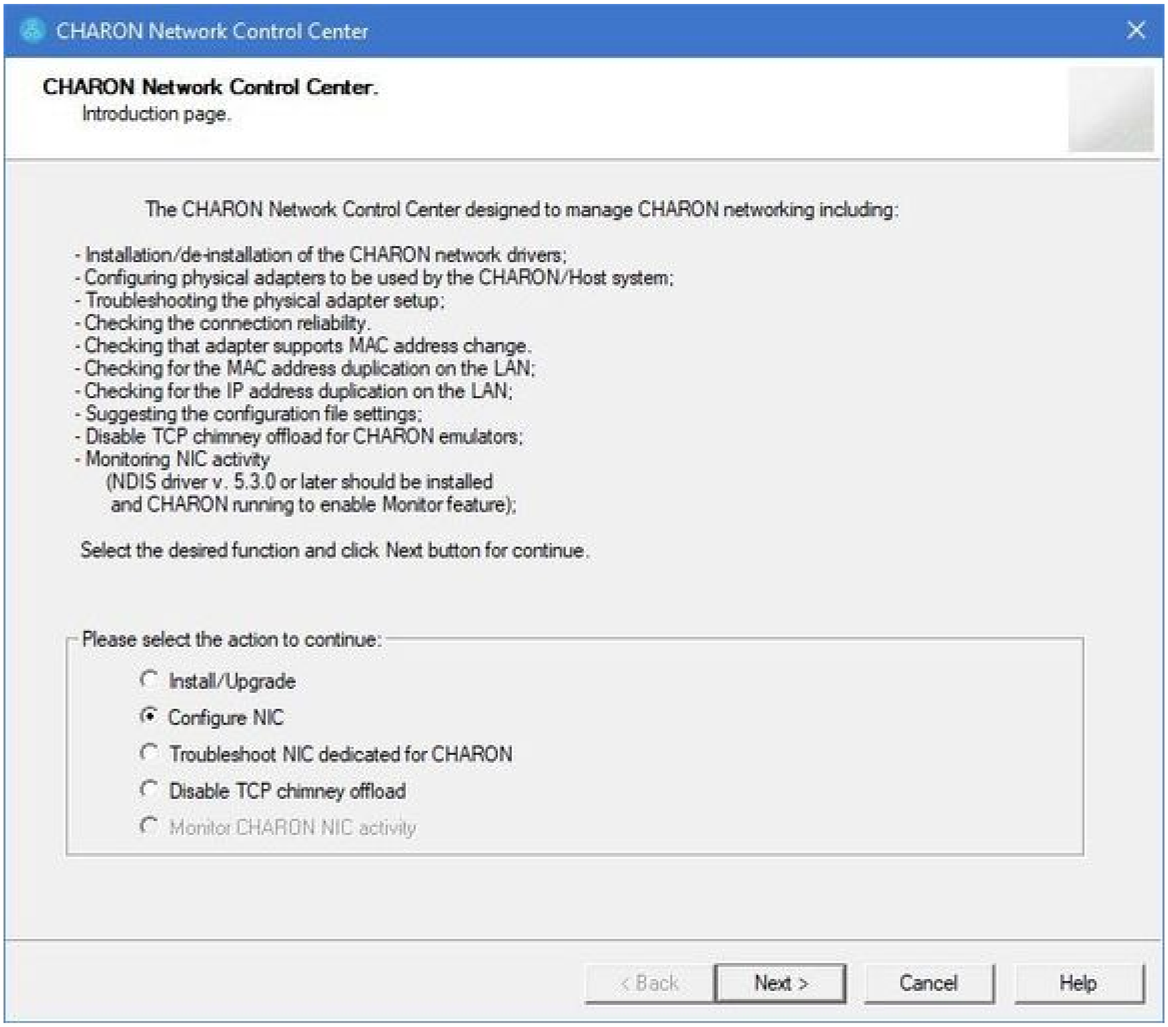
License Update tool (hasp_rus) v. 7.3

Network Control Center application (netdiag) v. 1.57

CHARON device check application (devcheck) v. 1.6

Virtual Disk tool (mkdskwin) v. 2.20

In the appearing dialog, select "Configure NIC":



CHARON Network Control Center.
Introduction page.

The CHARON Network Control Center designed to manage CHARON networking including:

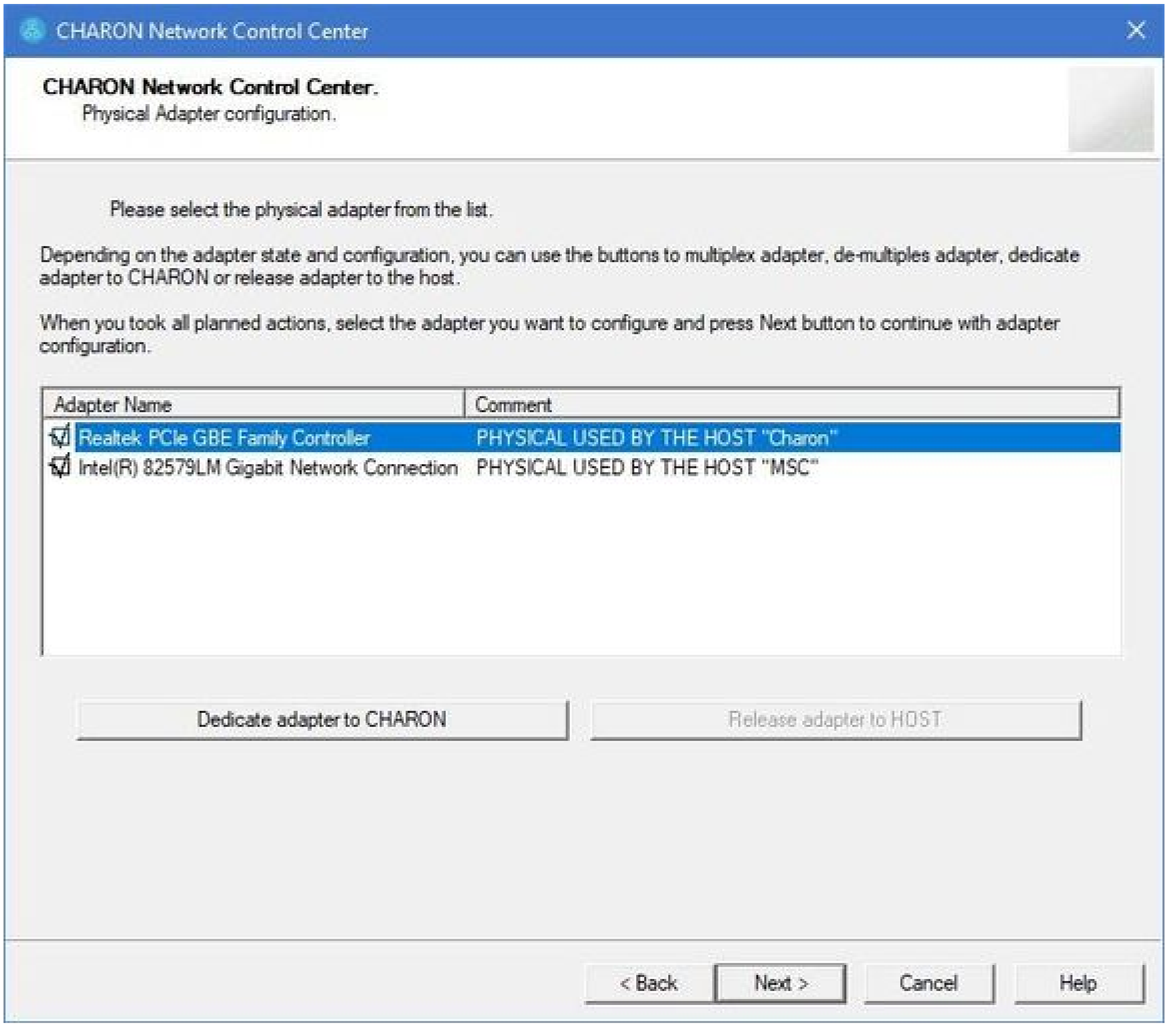
- Installation/de-installation of the CHARON network drivers;
- Configuring physical adapters to be used by the CHARON/Host system;
- Troubleshooting the physical adapter setup;
- Checking the connection reliability.
- Checking that adapter supports MAC address change.
- Checking for the MAC address duplication on the LAN;
- Checking for the IP address duplication on the LAN;
- Suggesting the configuration file settings;
- Disable TCP chimney offload for CHARON emulators;
- Monitoring NIC activity
  (NDIS driver v. 5.3.0 or later should be installed
  and CHARON running to enable Monitor feature);

Select the desired function and click Next button for continue.

Please select the action to continue:

○ Install/Upgrade
● Configure NIC
○ Troubleshoot NIC dedicated for CHARON
○ Disable TCP chimney offload
○ Monitor CHARON NIC activity

Press the "Next" button, the following dialog will appear:



CHARON Network Control Center.
Physical Adapter configuration.

Please select the physical adapter from the list.

Depending on the adapter state and configuration, you can use the buttons to multiplex adapter, de-multiples adapter, dedicate adapter to CHARON or release adapter to the host.

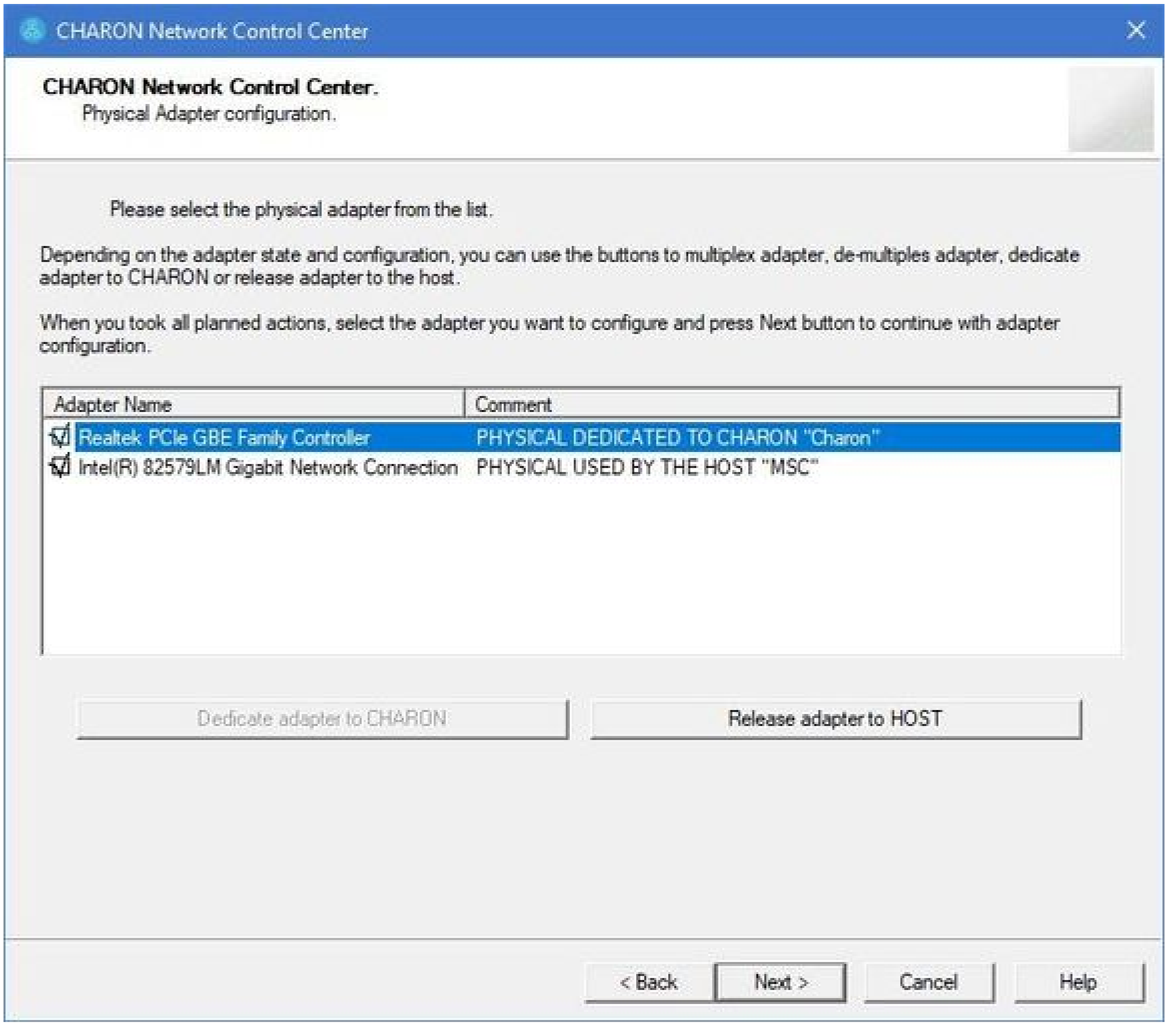
When you took all planned actions, select the adapter you want to configure and press Next button to continue with adapter configuration.

Adapter Name	Comment
Realtek PCIe GBE Family Controller	PHYSICAL USED BY THE HOST "Charon"
Intel(R) 82579LM Gigabit Network Connection	PHYSICAL USED BY THE HOST "MSC"

Select the interface to be dedicated to CHARON (in our example it is Realtek PCIe GBE Family Controller named "Charon") and press the "Dedicate adapter to CHARON" button.

CHARON supports VLAN adapters. If you are going to use them, proceed with their installation and configuration according to the network adapter vendor User's Guide and then select the VLAN interface in the dialog shown above, the same as you would for a regular network interface. There is no difference in configuring a regular network adapter and a VLAN one, so all the provided instructions are fully applicable for VLAN adapters as well.

After a few seconds the chosen interface will be assigned to CHARON:



CHARON Network Control Center.
Physical Adapter configuration.

Please select the physical adapter from the list.

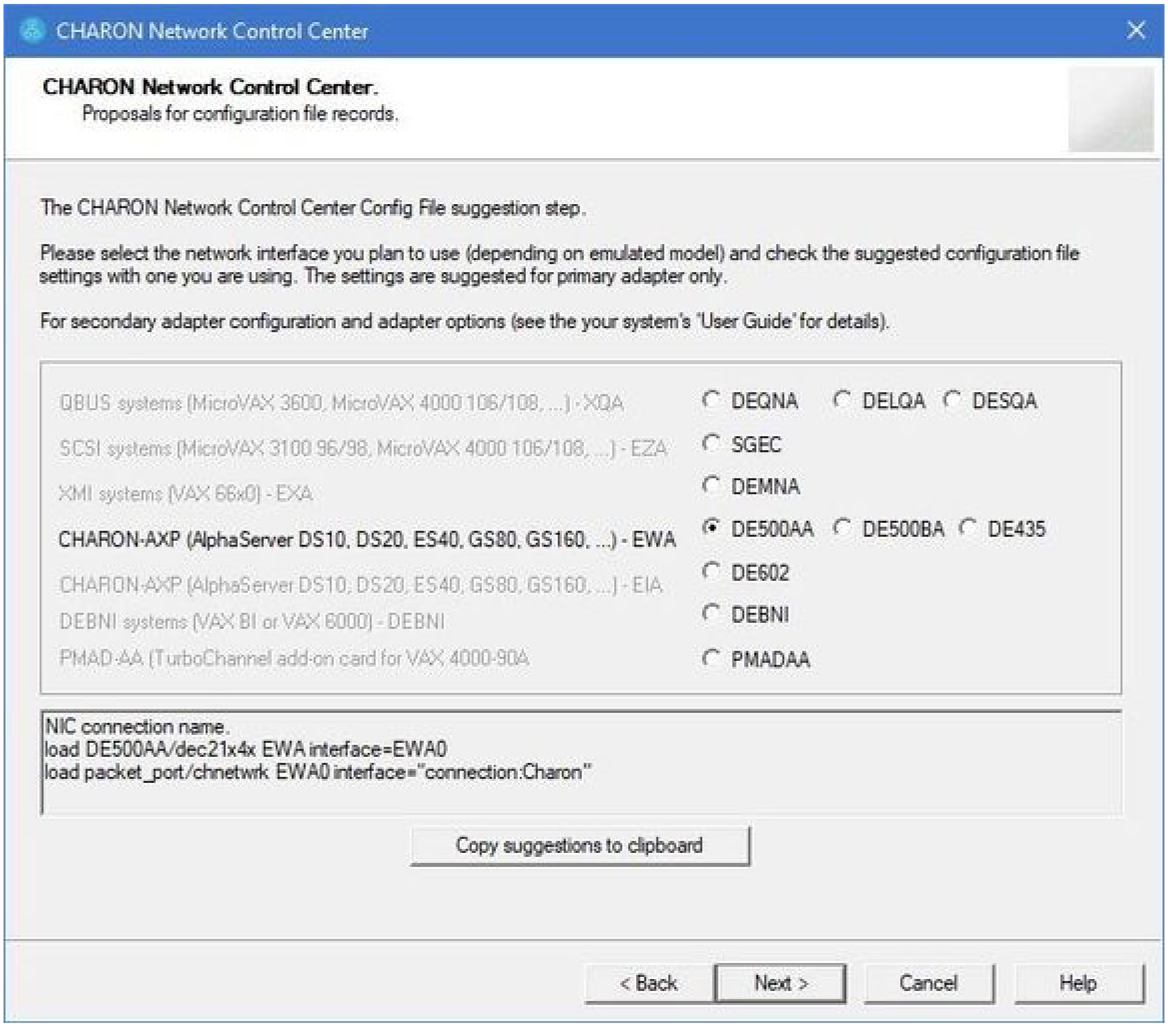
Depending on the adapter state and configuration, you can use the buttons to multiplex adapter, de-multiples adapter, dedicate adapter to CHARON or release adapter to the host.

When you took all planned actions, select the adapter you want to configure and press Next button to continue with adapter configuration.

Adapter Name	Comment
Realtek PCIe GBE Family Controller	PHYSICAL DEDICATED TO CHARON "Charon"
Intel(R) 82579LM Gigabit Network Connection	PHYSICAL USED BY THE HOST "MSC"

It is possible to release the interface back to the CHARON host by selecting the target interface and pressing the "Release adapter to HOST" button.

The next step displays the text to be included in the CHARON configuration file for the interface dedicated to CHARON. Select the target interface and press the "Next" button. The following dialog will appear:



CHARON Network Control Center

CHARON Network Control Center.
Proposals for configuration file records.

The CHARON Network Control Center Config File suggestion step.

Please select the network interface you plan to use (depending on emulated model) and check the suggested configuration file settings with one you are using. The settings are suggested for primary adapter only.

For secondary adapter configuration and adapter options (see the your system's 'User Guide' for details).

QBUS systems (MicroVAX 3600, MicroVAX 4000 106/108, ...) - XQA	○ DEQNA ○ DELQA ○ DESQA
SCSI systems (MicroVAX 3100 96/98, MicroVAX 4000 106/108, ...) - EZA	○ SGEC
XMI systems (VAX 66x0) - EXA	○ DEMNA
CHARON-AXP (AlphaServer DS10, DS20, ES40, GS80, GS160, ...) - EWA	● DE500AA ○ DE500BA ○ DE435
CHARON-AXP (AlphaServer DS10, DS20, ES40, GS80, GS160, ...) - EIA	○ DE602
DEBNI systems (VAX BI or VAX 6000) - DEBNI	○ DEBNI
PMAD-AA (TurboChannel add-on card for VAX 4000-90A	○ PMADAA

```
NIC connection name.
load DE500AA/dec21x4x EWA interface=EWA0
load packet_port/chnetwrk EWA0 interface="connection:Charon"
```

Selecting the target emulated network interface displays the configuration lines specific for the selected CHARON host network interface. Press the "Copy suggestions to clipboard" button to copy the suggested configuration lines. They can be pasted into the CHARON-AXP configuration file during the next editing session by pressing "Ctrl-V".

Press the "Cancel" button to exit from the "Network Control Center" utility.

Disabling Offload Parameters on Dedicated NIC

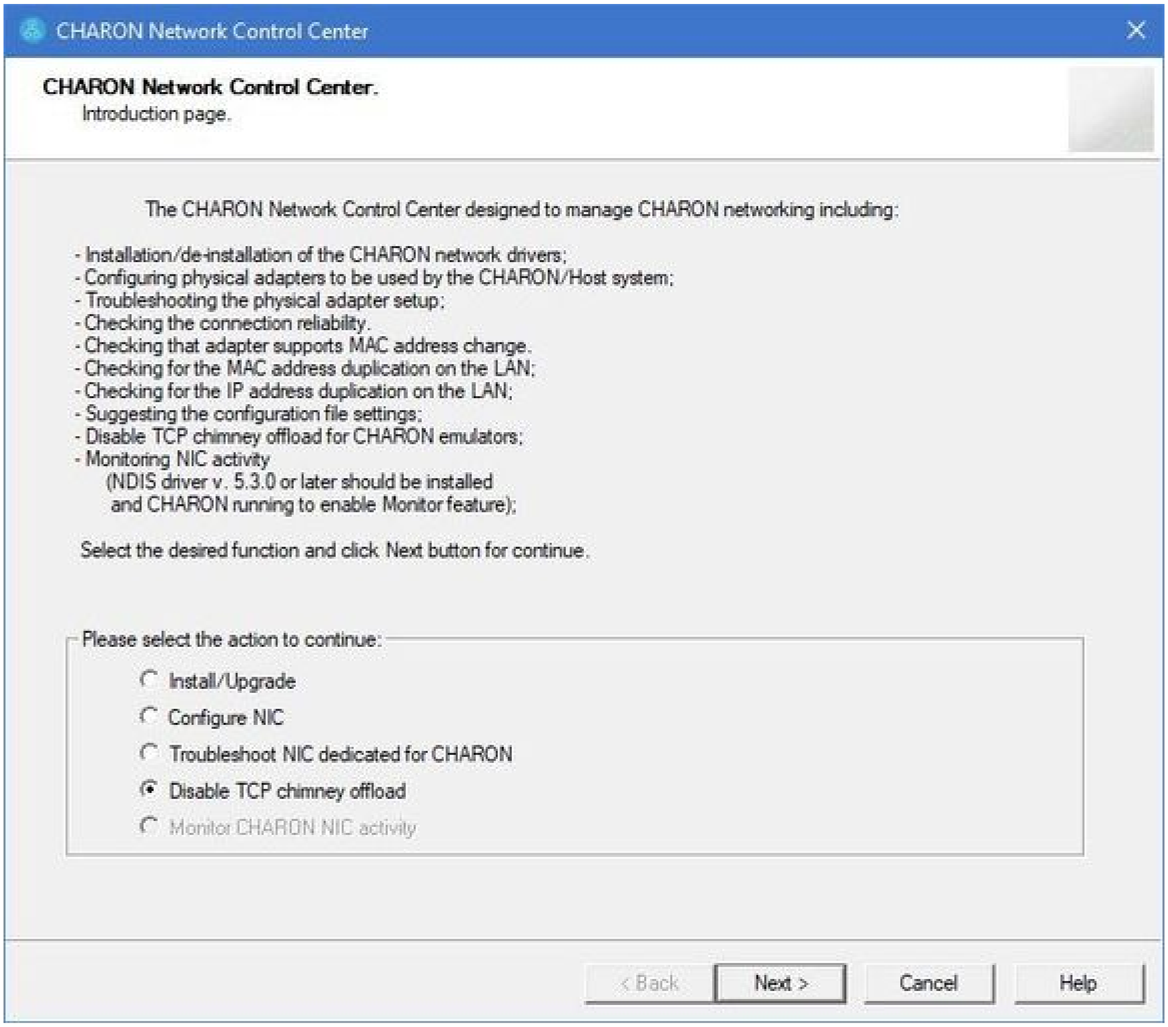
For every Network Interface (NIC) that you will dedicate for use with the Charon emulator, you should disable various offloading functions in the Windows driver.

Disabling TCP Chimney Offloading

Windows Version specific notes:

- On Windows Server 2019 and higher this feature is disabled by default (deprecated). So, unless the feature was manually enabled, no configuration is required.
- Windows Server 2008 R2, Windows Server 2012 R2, Windows Server 2016, Windows 7, Windows 8 & 8.1, and Windows 10: before using the Network Control Center Utility to disable the feature, ensure that the Windows Firewall service and BFE service are running.

Restart the "Network Control Center" utility again and select **Disable TCP chimney offload**:



CHARON Network Control Center

CHARON Network Control Center.
Introduction page.

The CHARON Network Control Center designed to manage CHARON networking including:

- Installation/de-installation of the CHARON network drivers;
- Configuring physical adapters to be used by the CHARON/Host system;
- Troubleshooting the physical adapter setup;
- Checking the connection reliability.
- Checking that adapter supports MAC address change.
- Checking for the MAC address duplication on the LAN;
- Checking for the IP address duplication on the LAN;
- Suggesting the configuration file settings;
- Disable TCP chimney offload for CHARON emulators;
- Monitoring NIC activity
(NDIS driver v. 5.3.0 or later should be installed
and CHARON running to enable Monitor feature);

Select the desired function and click Next button for continue.

Please select the action to continue:

○ Install/Upgrade
○ Configure NIC
○ Troubleshoot NIC dedicated for CHARON
● Disable TCP chimney offload
○ Monitor CHARON NIC activity

Press the **Next** button to apply the **Disable TCP chimney offload** parameters settings.

Press the **Cancel** button to exit from the "Network Control Center" utility.

Please note:

- When the NIC properties are configured in Windows, a reboot of the Charon host is required for the changes, specifically changes to the offload parameters, to take effect.
- Do not forget to cleanly shut down any running guests before rebooting the Windows host!

Manually Configuring/Checking the Connection Items

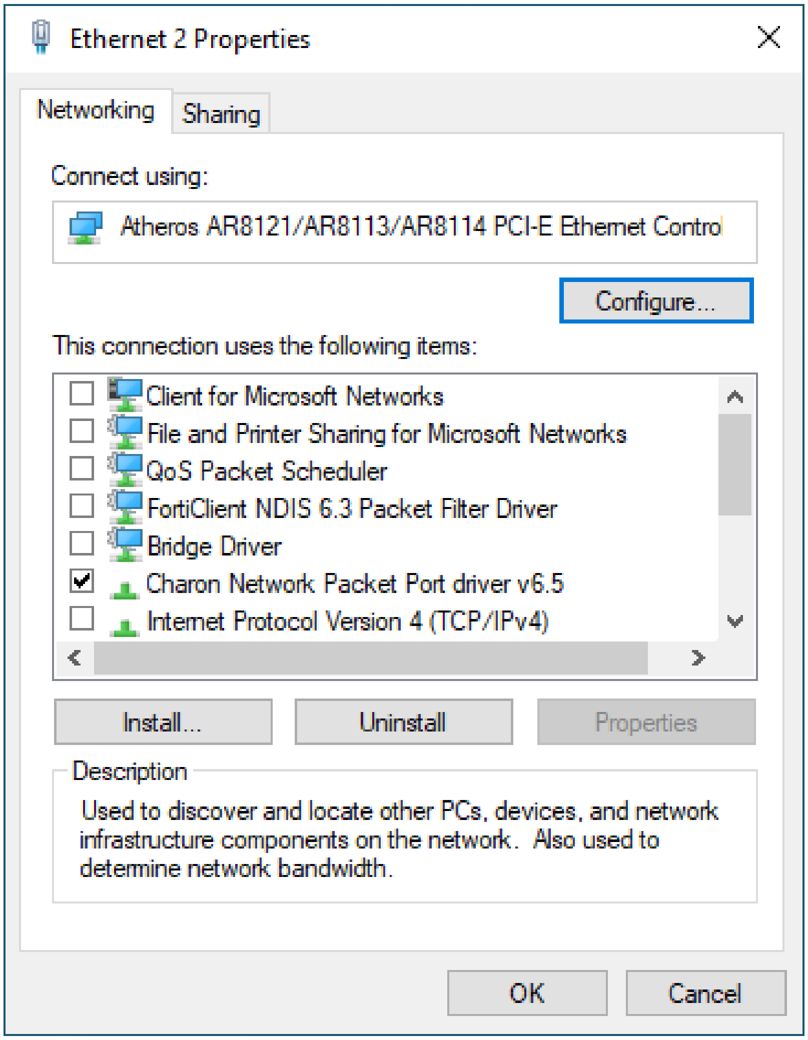
For a NIC that is dedicated to Charon, the following three items should be checked, and all others unchecked:

- Charon Network Packet Port driver
- Link-Layer Topology Discovery Responder
- Link-Layer Topology Discovery Mapper I/O Driver

Disable Network Interface Attributes Manually

For every Network Interface (NIC) that you will dedicate to the Charon emulator, you should manually disable various offloading functions in the Windows driver. The following description applies to Windows 10 Pro. It may differ somewhat for other Windows versions.

Open the "Settings" dialog, select "Network & Internet", then "Change Adapter Options". A new dialog window will appear showing all the NICs known to the system. For all NICs that will be used by Charon, right-click on the NIC and choose "Properties". The following dialog should appear:



Ethernet 2 Properties

Connect using:

Atheros AR8121/AR8113/AR8114 PCI-E Ethernet Contro

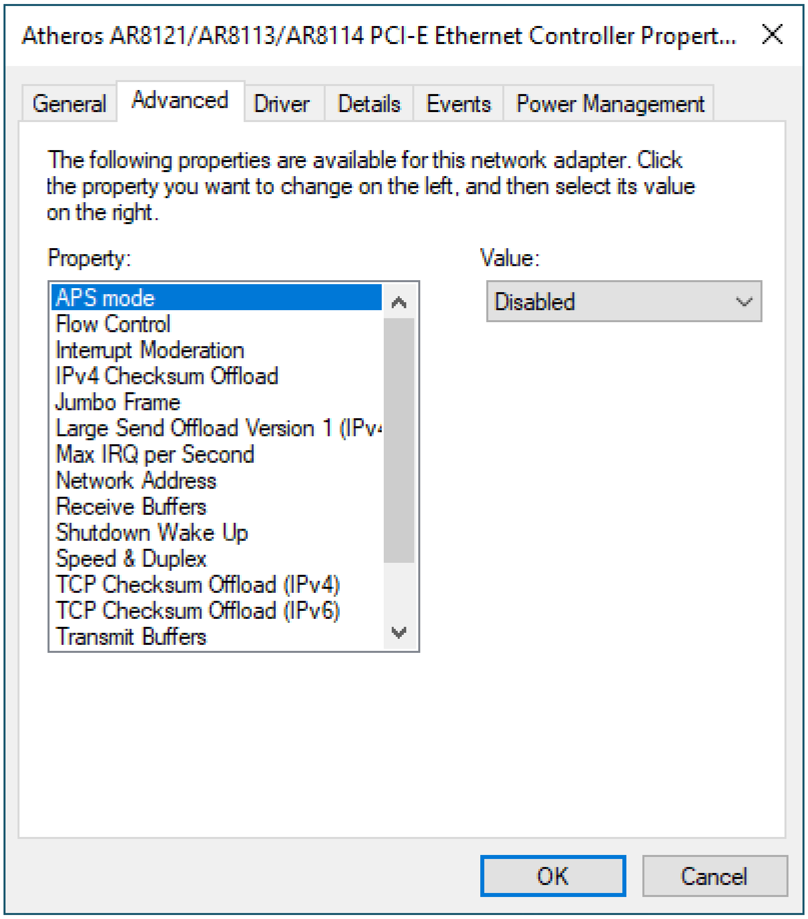
This connection uses the following items:

- ☐ Client for Microsoft Networks
- ☐ File and Printer Sharing for Microsoft Networks
- ☐ QoS Packet Scheduler
- ☐ FortiClient NDIS 6.3 Packet Filter Driver
- ☐ Bridge Driver
- ☑ Charon Network Packet Port driver v6.5
- ☐ Internet Protocol Version 4 (TCP/IPv4)

Description

Used to discover and locate other PCs, devices, and network infrastructure components on the network. Also used to determine network bandwidth.

Click the "Configure" button to open. The following dialog window should appear. Select the "Advanced" tab.



Atheros AR8121/AR8113/AR8114 PCI-E Ethernet Controller Propert...

The following properties are available for this network adapter. Click the property you want to change on the left, and then select its value on the right.

Property:

- APS mode
- Flow Control
- Interrupt Moderation
- IPv4 Checksum Offload
- Jumbo Frame
- Large Send Offload Version 1 (IPv
- Max IRQ per Second
- Network Address
- Receive Buffers
- Shutdown Wake Up
- Speed & Duplex
- TCP Checksum Offload (IPv4)
- TCP Checksum Offload (IPv6)
- Transmit Buffers

Value: Disabled

Go through every "Property" and look at the options available in the "Value" drop-down. The properties that you see here will differ depending on the networking hardware, the network driver, and the Windows version, so please keep in mind that the specific examples mentioned below are exactly that – examples. If the value "disable" is offered, you should select that unless it is on the list of things below of things you should not change. For properties with numeric values, you should not change the value.

The following are examples of properties that must be disabled:

- Adaptive Inter-Frame Spacing
- IPv4 Checksum Offload
- Jumbo Packet
- Large Send Offload V1 or V2 (IPv4 & IPv6)
- TCP Checksum Offload (IPv4 & IPv6)
- UDP Checksum Offload (IPv4 & IPv6)

Examples of properties that should not be changed:

- Packet Priority & VLAN
- Interrupt Moderation
- Receive Side Scaling
- Flow Control
- Locally Administered Address
- Maximum Number of RSS Queues
- Receive Buffers
- Transmit Buffers

After the installation

If you plan to use the local system administrator acount ("Administrator") or the CHARON-AXP user having administrative privileges, no other actions are required.

If the CHARON-AXP user belongs to an AD domain, you must add this user to the CHARON-GRP group as described in this article and then reboot the CHARON host.

Otherwise it is possible to use a standard account (both local and domain) for running CHARON-AXP:

1. Login as the local system administrator ("Administrator") on the host system.
2. Create a special user for running CHARON-AXP. This user must have administrative or standard privileges (in this case login as Administrator for managing NDIS drivers and running NCC utility). Please consult the Windows online documentation for details.
3. Add this user to the CHARON-GRP group as it is described in this article and then reboot CHARON host.
4. Log in as the new user.

Running and managing CHARON-AXP for Windows

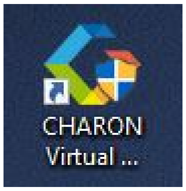
Table of Contents

Starting CHARON Virtual Machine Manager

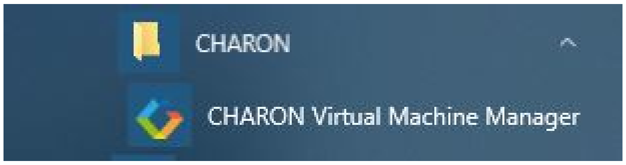
CHARON execution is controlled by the "CHARON Virtual Machine Manager" that manages each CHARON instance, called a "CHARON Virtual Machine" , running on the host.

The CHARON installation procedure creates a shortcut on the desktop and under the Windows Start menu for the CHARON Virtual Machine Manager.

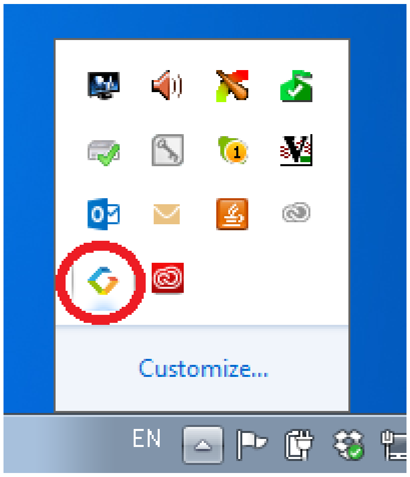
Desktop shortcut:



"Start" menu shortcut:



The CHARON Virtual Machine Manager runs automatically at Windows startup. If it is minimized to tray, it can be re-opened in the following way:



After clicking the icon, the CHARON Virtual Machine Manager window appears.

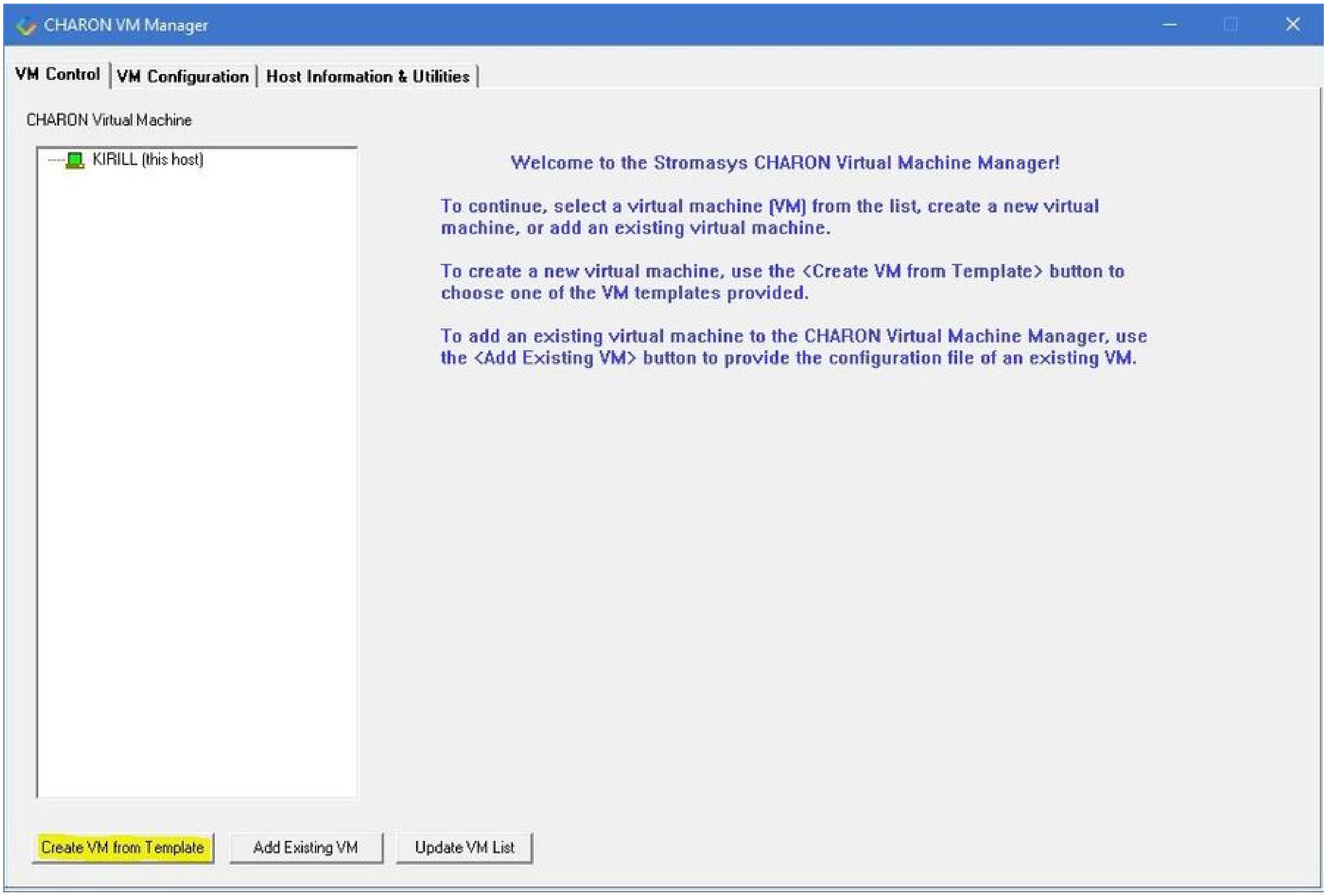
Creating a CHARON Virtual Machine

Creating a CHARON VM using provided templates

CHARON installation contains a set of template files for creating a base configuration for each AXP model. Once the CHARON Virtual Machine (VM) is created using these templates it can be configured further to meet your requirements and all the specifics of the emulated system.

⚠ By default the CHARON templates do not contain any specification for disks and network; you need to update the configuration file created from the template to specify your specific disk and network configuration.

Start the CHARON Virtual Machine Manager (see above) and press the "Create VM from Template" button:



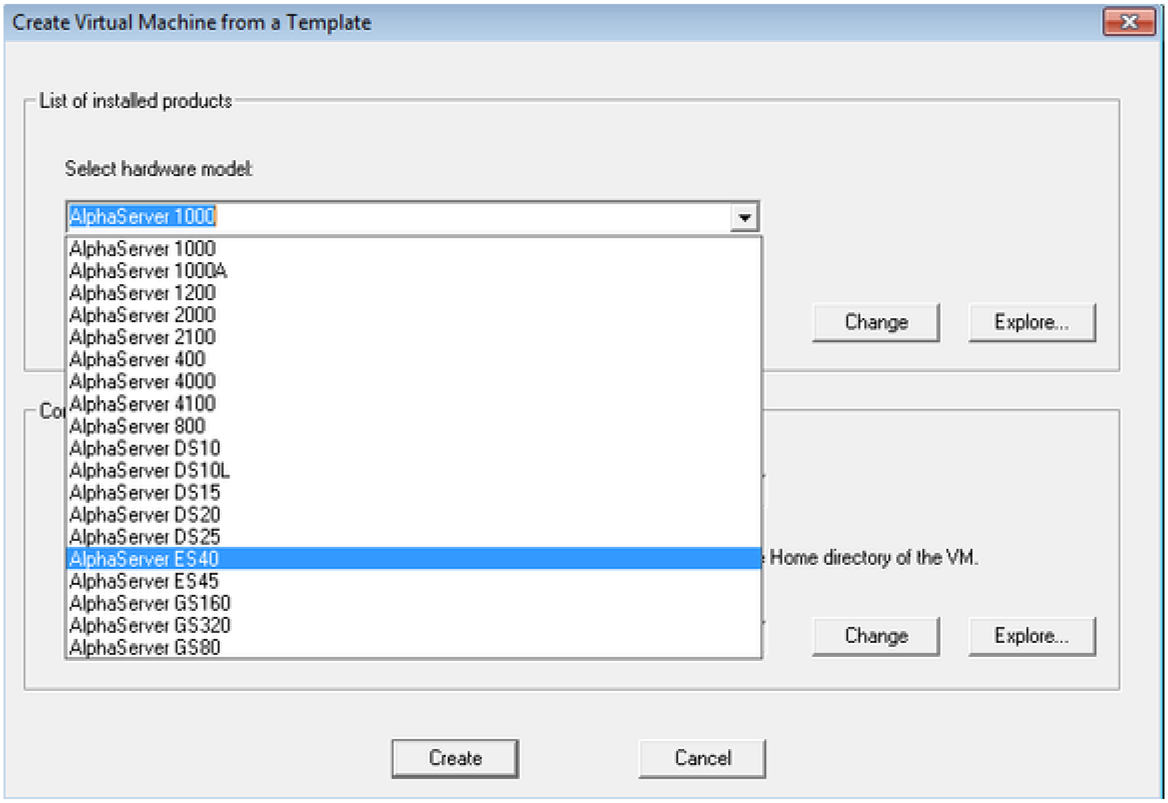
Welcome to the Stromasys CHARON Virtual Machine Manager!

To continue, select a virtual machine (VM) from the list, create a new virtual machine, or add an existing virtual machine.

To create a new virtual machine, use the <Create VM from Template> button to choose one of the VM templates provided.

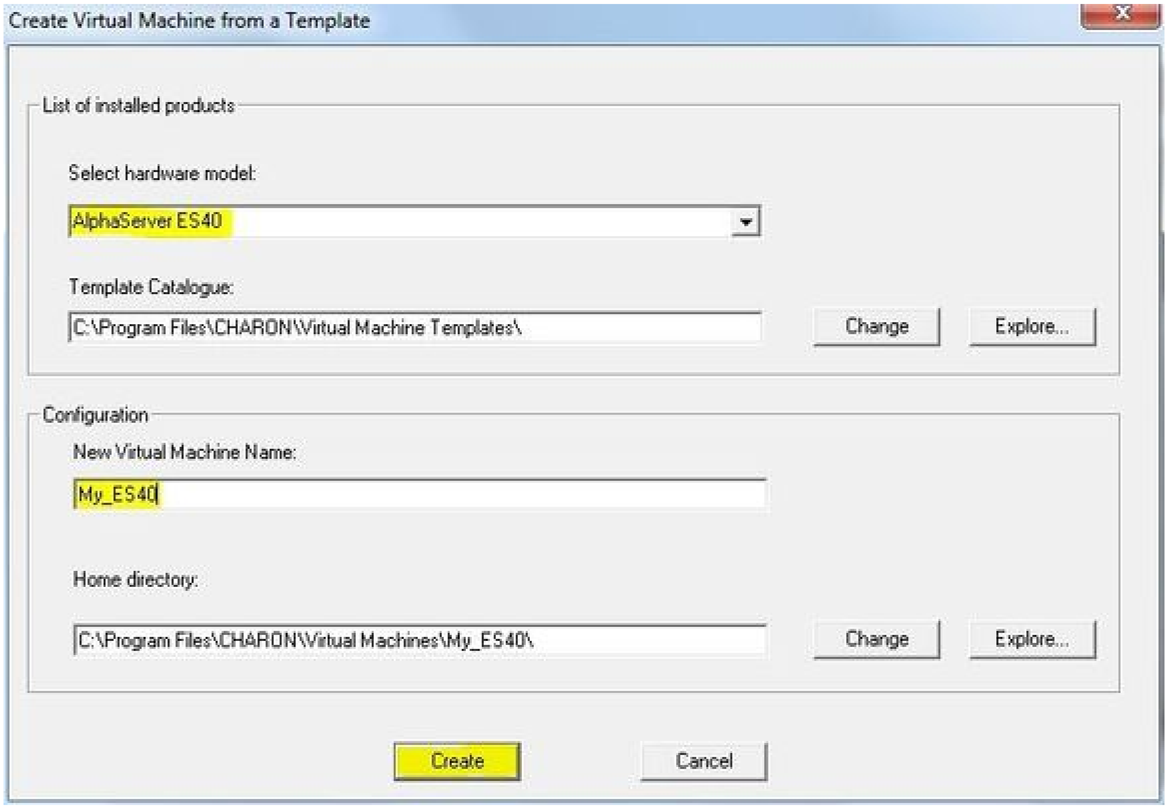
To add an existing virtual machine to the CHARON Virtual Machine Manager, use the <Add Existing VM> button to provide the configuration file of an existing VM.

In the dialog window that appears select the Alpha hardware model:



Note the Home Directory for the VM (see screenshot below), it will be used by default for storing logs, virtual disk images, toy and container files if no path is specified in the configuration file.

Specify a name for the VM and press the "Create" button:


Create Virtual Machine from a Template
List of installed products
Configuration
Change
Explore...
Change
Explore...
Create
Cancel

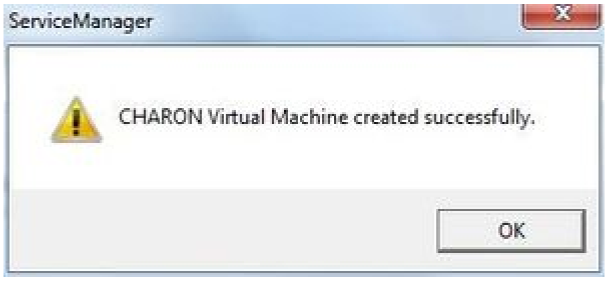
Select hardware model: AlphaServer ES40

Template Catalogue: C:\Program Files\CHARON\Virtual Machine Templates\

New Virtual Machine Name: My_ES40

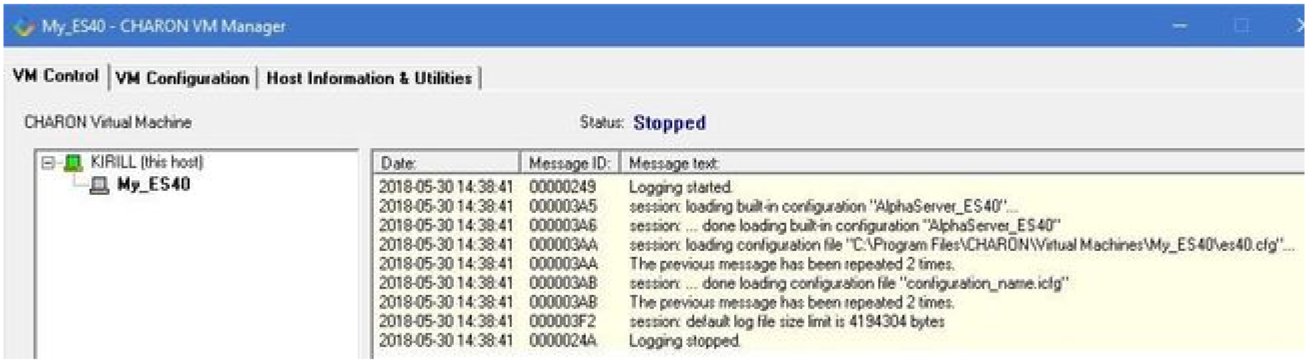
Home directory: C:\Program Files\CHARON\Virtual Machines\My_ES40\

The CHARON Virtual Machine Manager will report the VM has been created successfully:



CHARON Virtual Machine created successfully.

It will then appear in the left pane:



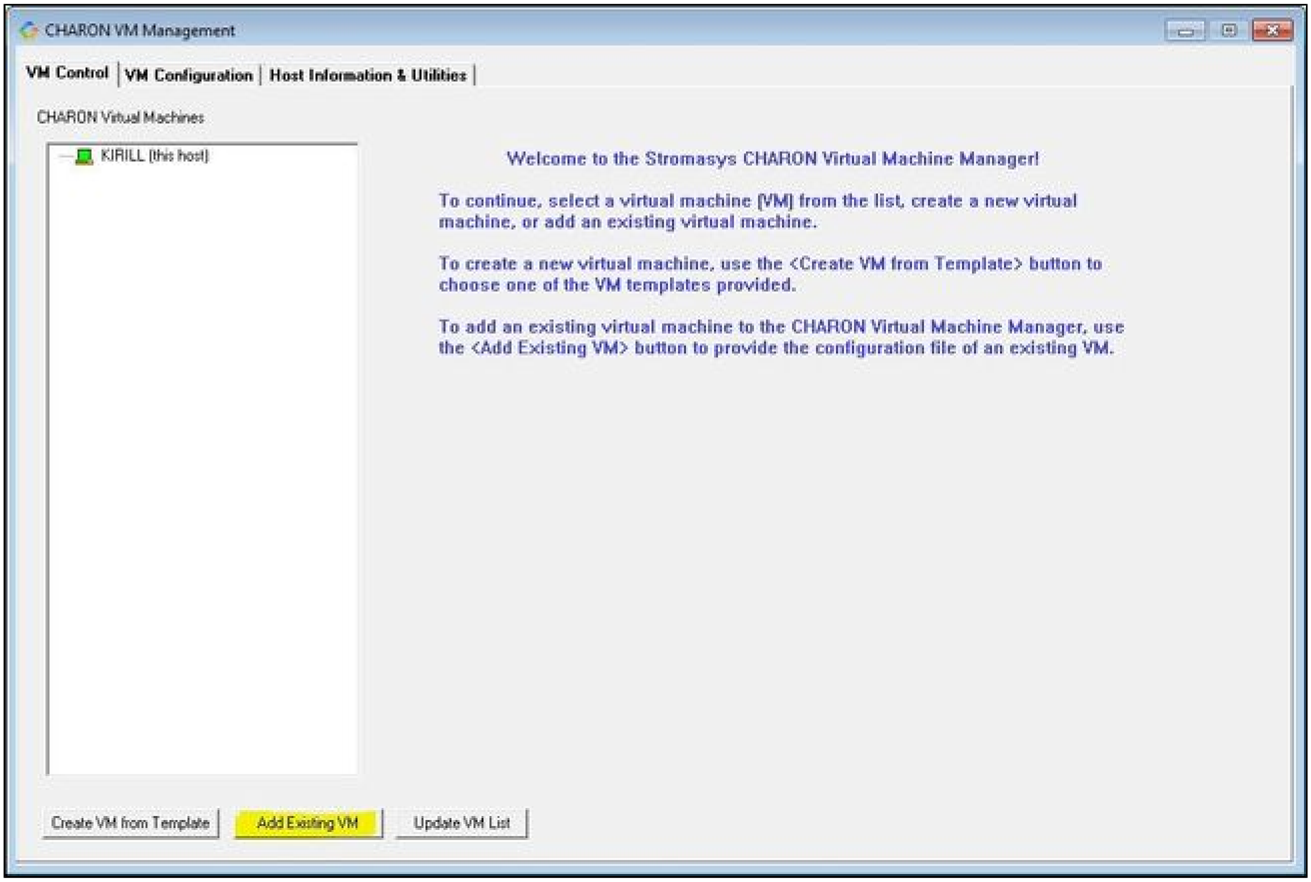
Status: Stopped

```
Date:                Message ID:  Message text:
2018-05-30 14:38:41  00000249     Logging started.
2018-05-30 14:38:41  000003A5     session: loading built-in configuration "AlphaServer_ES40"...
2018-05-30 14:38:41  000003A6     session: ... done loading built-in configuration "AlphaServer_ES40"
2018-05-30 14:38:41  000003AA     session: loading configuration file "C:\Program Files\CHARON\Virtual Machines\My_ES40\es40.cfg"...
2018-05-30 14:38:41  000003AA     The previous message has been repeated 2 times.
2018-05-30 14:38:41  000003AB     session: ... done loading configuration file "configuration_name.icfg"
2018-05-30 14:38:41  000003AB     The previous message has been repeated 2 times.
2018-05-30 14:38:41  000003F2     session: default log file size limit is 4194304 bytes
2018-05-30 14:38:41  0000024A     Logging stopped.
```

Creating a CHARON VM using an existing configuration file

It is also possible to use an existing configuration file for creating a CHARON Virtual Machine.

Open the CHARON Virtual Machine Manager (see above) and press the "Add Existing VM" button:



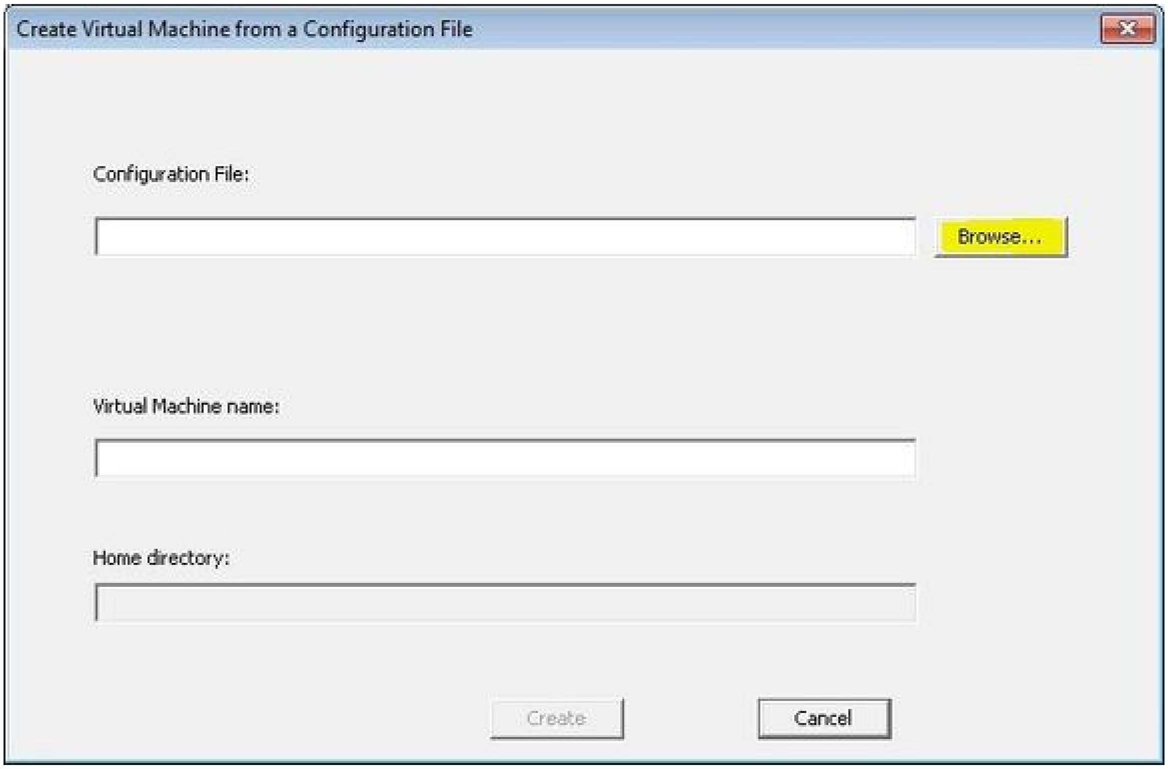
Welcome to the Stromasys CHARON Virtual Machine Manager!

To continue, select a virtual machine (VM) from the list, create a new virtual machine, or add an existing virtual machine.

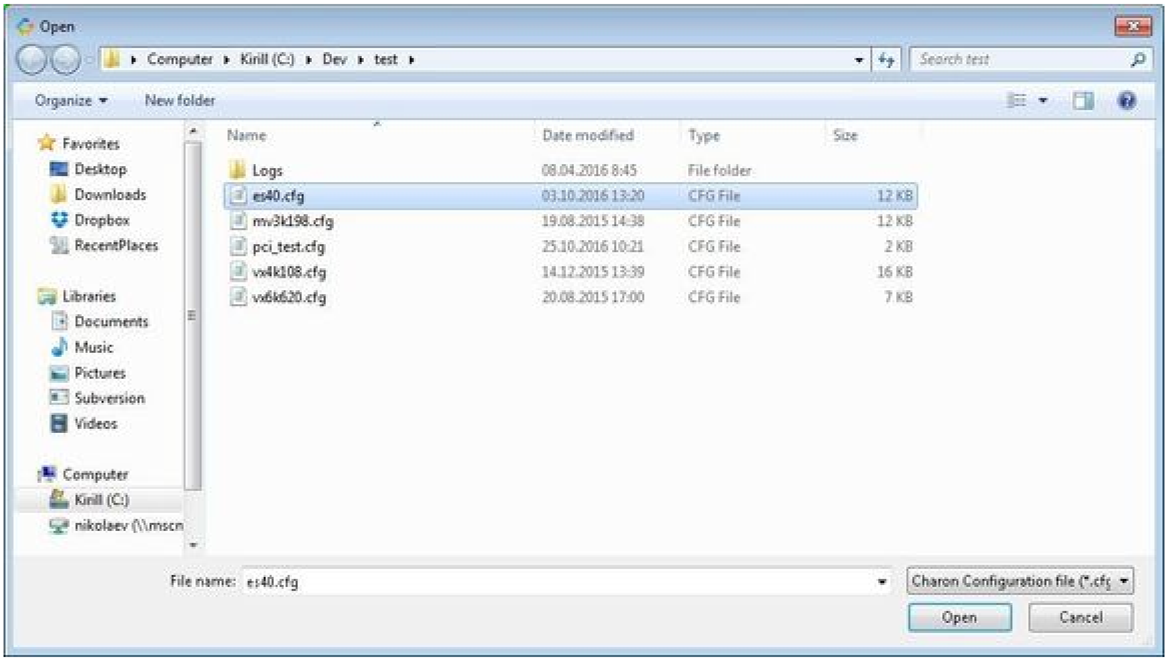
To create a new virtual machine, use the <Create VM from Template> button to choose one of the VM templates provided.

To add an existing virtual machine to the CHARON Virtual Machine Manager, use the <Add Existing VM> button to provide the configuration file of an existing VM.

In the dialog window that appears select the target configuration file by clicking the "Browse" button:



Select the configuration file and click the "Open" button:



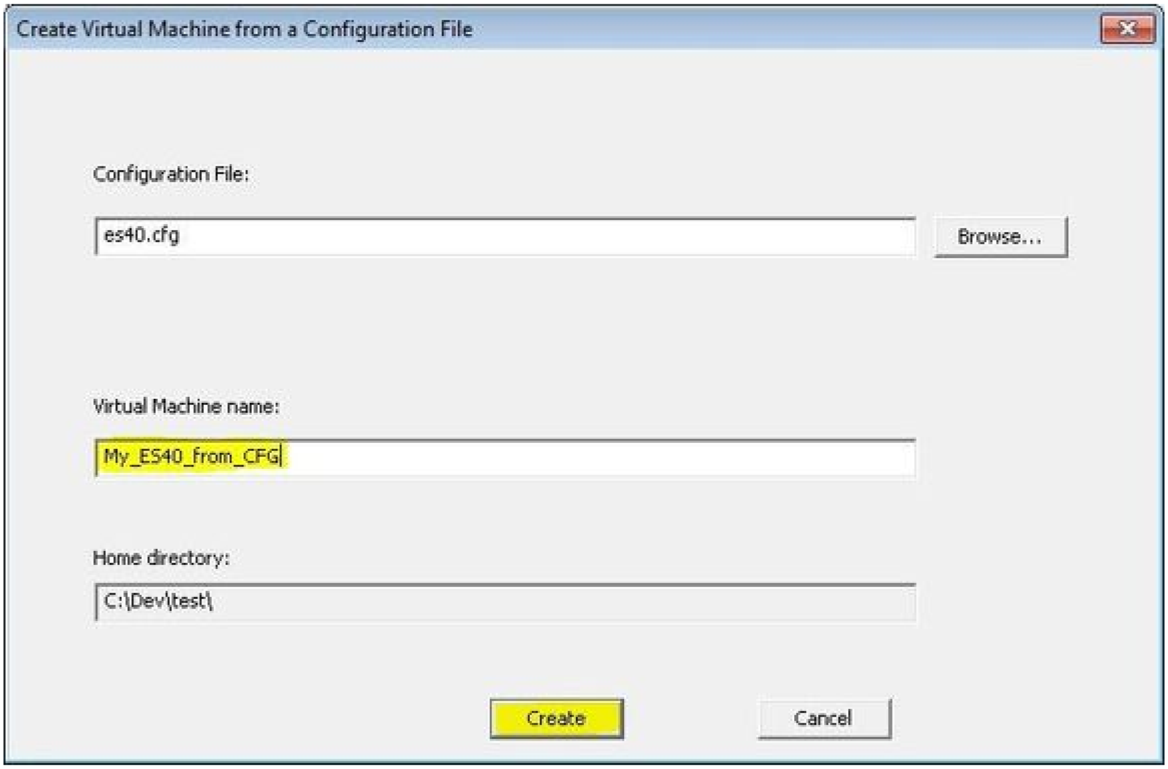
Name	Date modified	Type	Size
Logs	08.04.2016 8:45	File folder	
es40.cfg	03.10.2016 13:20	CFG File	12 KB
mv3k198.cfg	19.08.2015 14:38	CFG File	12 KB
pci_test.cfg	25.10.2016 10:21	CFG File	2 KB
vx4k108.cfg	14.12.2015 13:39	CFG File	16 KB
vx6k620.cfg	20.08.2015 17:00	CFG File	7 KB

File name: es40.cfg

⚠ Never select a configuration file template provided in the Charon installation folder as these files could be overwritten with patching.

Enter the CHARON Virtual Machine name and note the "Home directory" for this VM, it will be used by default for storing logs, virtual disk images, toy and container files if no path is specified in the configuration file.

Press the "Create" button:



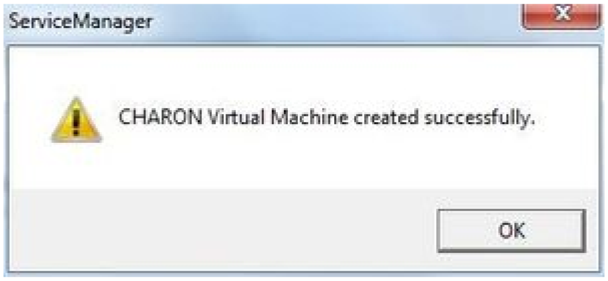
Create Virtual Machine from a Configuration File

Configuration File: es40.cfg

Virtual Machine name: My_ES40_from_CFG

Home directory: C:\Dev\test\

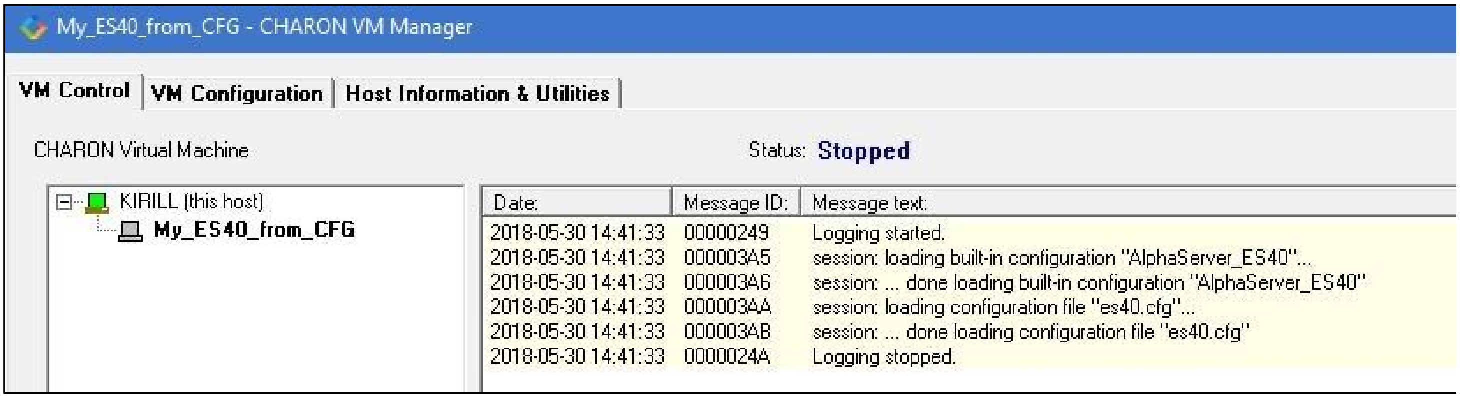
The CHARON Virtual Machine Manager will report the VM has been created successfully:



ServiceManager

CHARON Virtual Machine created successfully.

It will then appear in the left pane:



My_ES40_from_CFG - CHARON VM Manager

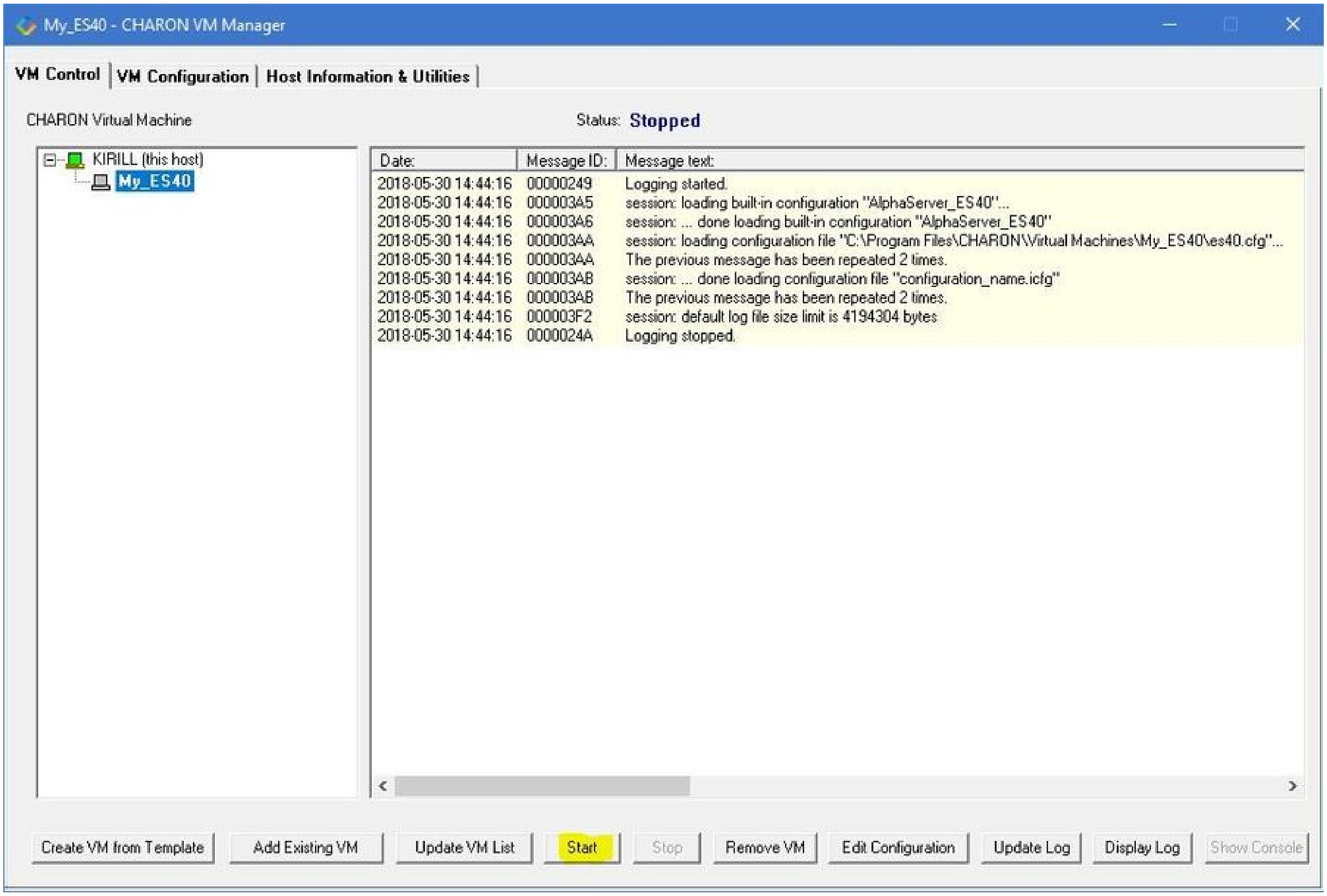
CHARON Virtual Machine

Status: Stopped

```
Date:                Message ID:  Message text:
2018-05-30 14:41:33  00000249     Logging started.
2018-05-30 14:41:33  000003A5     session: loading built-in configuration "AlphaServer_ES40"...
2018-05-30 14:41:33  000003A6     session: ... done loading built-in configuration "AlphaServer_ES40"
2018-05-30 14:41:33  000003AA     session: loading configuration file "es40.cfg"...
2018-05-30 14:41:33  000003AB     session: ... done loading configuration file "es40.cfg"
2018-05-30 14:41:33  0000024A     Logging stopped.
```

Running a CHARON Virtual Machine

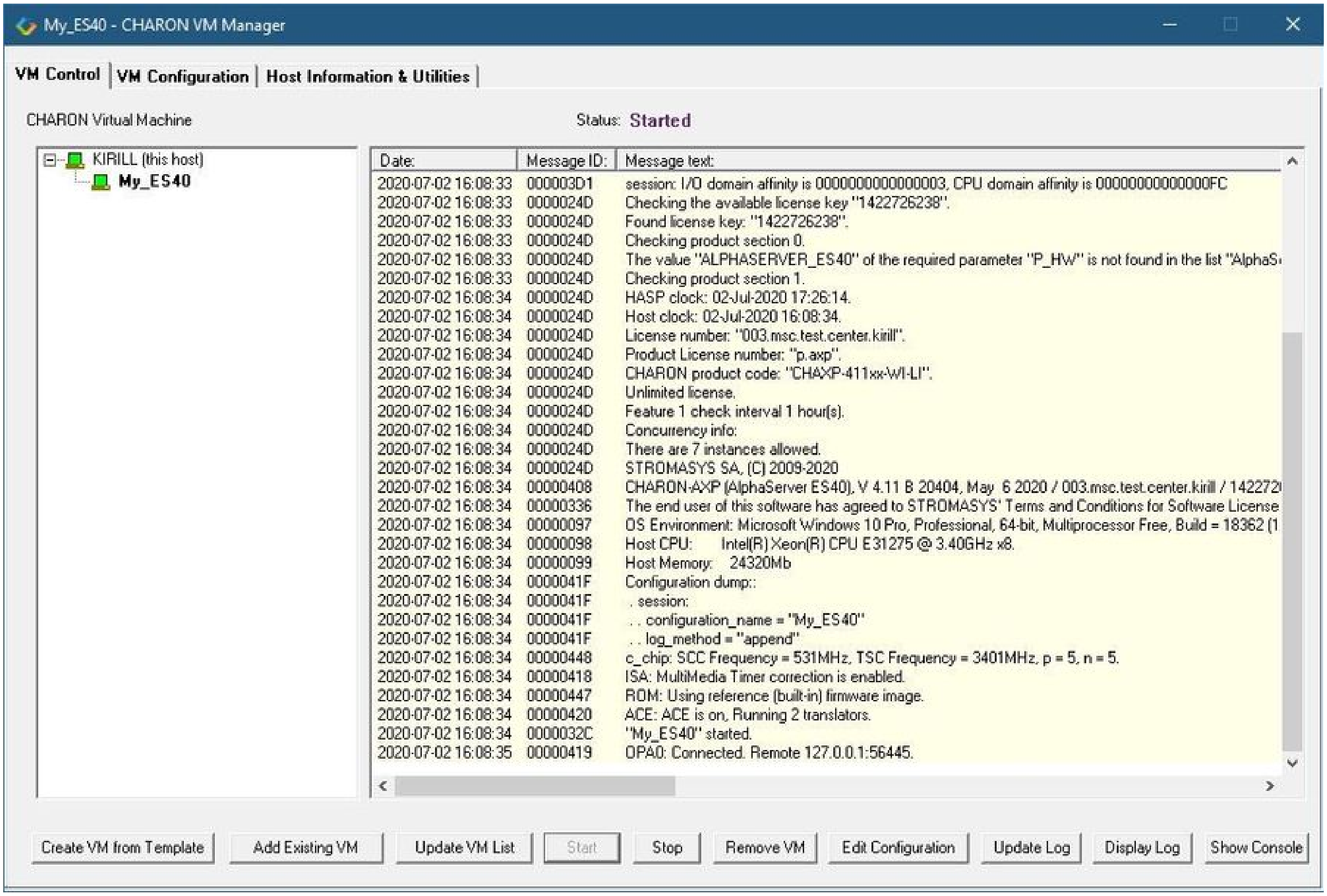
Select the target CHARON VM in the left pane and press the "Start" button:



Status: Stopped

```
Date:                Message ID:  Message text:
2018-05-30 14:44:16  00000249     Logging started.
2018-05-30 14:44:16  000003A5     session: loading built-in configuration "AlphaServer_ES40"...
2018-05-30 14:44:16  000003A6     session: ... done loading built-in configuration "AlphaServer_ES40"
2018-05-30 14:44:16  000003AA     session: loading configuration file "C:\Program Files\CHARON\Virtual Machines\My_ES40\es40.cfg"...
2018-05-30 14:44:16  000003AA     The previous message has been repeated 2 times.
2018-05-30 14:44:16  000003AB     session: ... done loading configuration file "configuration_name.icfg"
2018-05-30 14:44:16  000003AB     The previous message has been repeated 2 times.
2018-05-30 14:44:16  000003F2     session: default log file size limit is 4194304 bytes
2018-05-30 14:44:16  0000024A     Logging stopped.
```

The CHARON Virtual Machine Manager will display the execution log:



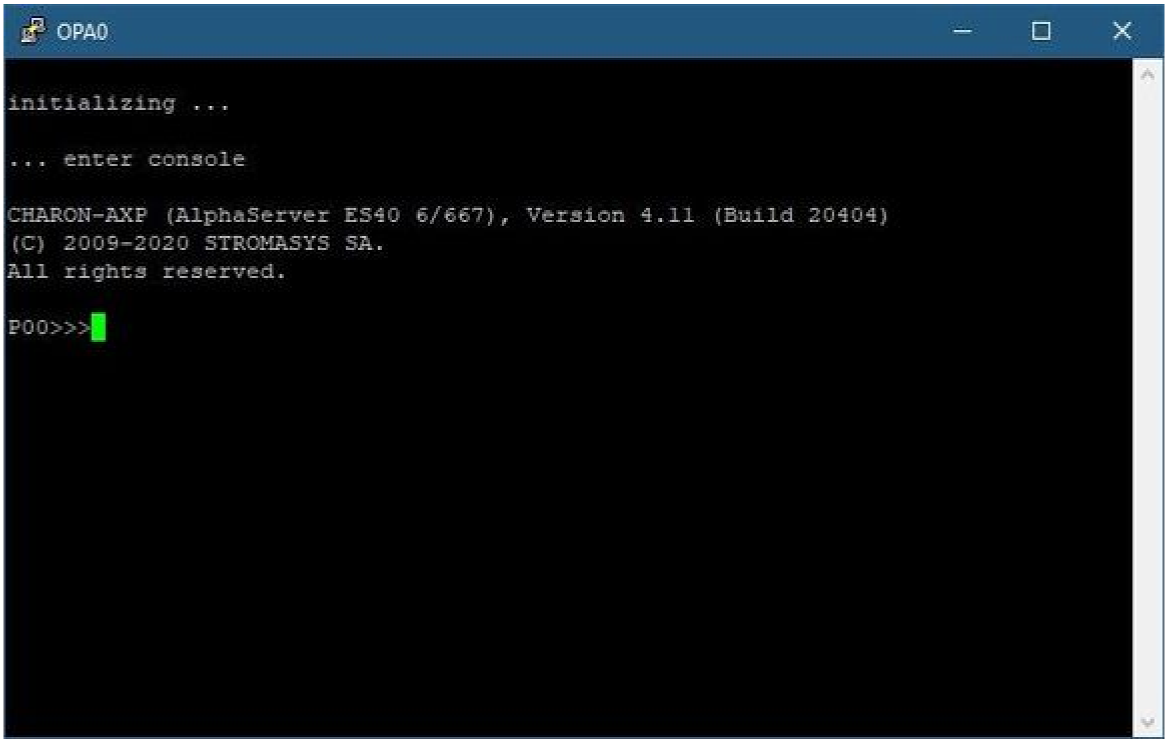
Status: Started

```
Date:                Message ID:  Message text:
2020-07-02 16:08:33  000003D1     session: I/O domain affinity is 0000000000000003, CPU domain affinity is 00000000000000FC
2020-07-02 16:08:33  0000024D     Checking the available license key "1422726238".
2020-07-02 16:08:33  0000024D     Found license key: "1422726238".
2020-07-02 16:08:33  0000024D     Checking product section 0.
2020-07-02 16:08:33  0000024D     The value "ALPHASERVER_ES40" of the required parameter "P_HW" is not found in the list "AlphaS
2020-07-02 16:08:33  0000024D     Checking product section 1.
2020-07-02 16:08:34  0000024D     HASP clock: 02-Jul-2020 17:26:14.
2020-07-02 16:08:34  0000024D     Host clock: 02-Jul-2020 16:08:34.
2020-07-02 16:08:34  0000024D     License number: "003.msc.test.center.kirill".
2020-07-02 16:08:34  0000024D     Product License number: "p.axp".
2020-07-02 16:08:34  0000024D     CHARON product code: "CHAXP-411xx-WI-LI".
2020-07-02 16:08:34  0000024D     Unlimited license.
2020-07-02 16:08:34  0000024D     Feature 1 check interval 1 hour(s).
2020-07-02 16:08:34  0000024D     Concurrency info:
2020-07-02 16:08:34  0000024D     There are 7 instances allowed.
2020-07-02 16:08:34  0000024D     STROMASYS SA, (C) 2009-2020
2020-07-02 16:08:34  00000408     CHARON-AXP (AlphaServer ES40), V 4.11 B 20404, May  6 2020 / 003.msc.test.center.kirill / 142272
2020-07-02 16:08:34  00000336     The end user of this software has agreed to STROMASYS' Terms and Conditions for Software License
2020-07-02 16:08:34  00000097     OS Environment: Microsoft Windows 10 Pro, Professional, 64-bit, Multiprocessor Free, Build = 18362 (1
2020-07-02 16:08:34  00000098     Host CPU:      Intel(R) Xeon(R) CPU E31275 @ 3.40GHz x8.
2020-07-02 16:08:34  00000099     Host Memory:    24320Mb
2020-07-02 16:08:34  0000041F     Configuration dump::
2020-07-02 16:08:34  0000041F      . session:
2020-07-02 16:08:34  0000041F      . . configuration_name = "My_ES40"
2020-07-02 16:08:34  0000041F      . . log_method = "append"
2020-07-02 16:08:34  00000448     c_chip: SCC Frequency = 531MHz, TSC Frequency = 3401MHz, p = 5, n = 5.
2020-07-02 16:08:34  00000418     ISA: MultiMedia Timer correction is enabled.
2020-07-02 16:08:34  00000447     ROM: Using reference (built-in) firmware image.
2020-07-02 16:08:34  00000420     ACE: ACE is on, Running 2 translators.
2020-07-02 16:08:34  0000032C     "My_ES40" started.
2020-07-02 16:08:35  00000419     OPA0: Connected. Remote 127.0.0.1:56445.
```

Note the color meaning:

Colors	Meaning	Example
Red	Error	This is an error message
Light blue	Warning	This is a warning message
Light yellow	Information	This is an informational message
Purple	Begin/End of the Windows system log	Begin / End

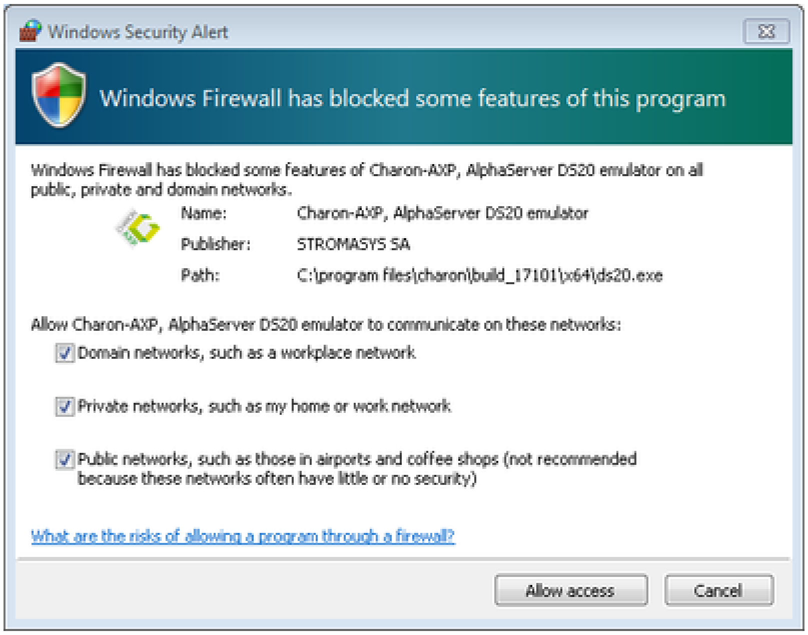
The console of this particular CHARON VM will then appear (if defined in the configuration file):



```
initializing ...

... enter console

CHARON-AXP (AlphaServer ES40 6/667), Version 4.11 (Build 20404)
(C) 2009-2020 STROMASYS SA.
All rights reserved.

P00>>>
```

The total number of devices (both controllers and units, including disks) displayed by CHARON SRM console is limited to 48. Note that it does not affect the actual number of configured devices provided to CHARON guest OS.

Depending on settings, Windows may display a Firewall warning dialog asking to confirm CHARON access to different networks.



Windows Firewall has blocked some features of this program

Windows Firewall has blocked some features of Charon-AXP, AlphaServer DS20 emulator on all public, private and domain networks.

Name: Charon-AXP, AlphaServer DS20 emulator

Publisher: STROMASYS SA

Path: C:\program files\charon\build_17101\x64\ds20.exe

Allow Charon-AXP, AlphaServer DS20 emulator to communicate on these networks:

☑ Domain networks, such as a workplace network

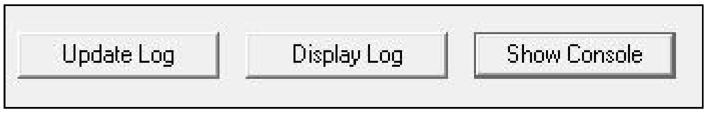
☑ Private networks, such as my home or work network

☑ Public networks, such as those in airports and coffee shops (not recommended because these networks often have little or no security)

What are the risks of allowing a program through a firewall?
Allow access
Cancel

Check the desired options and press the "Allow access" button.

If you closed the CHARON VM console, it is always possible to re-open it by pressing the "Show Console" button:

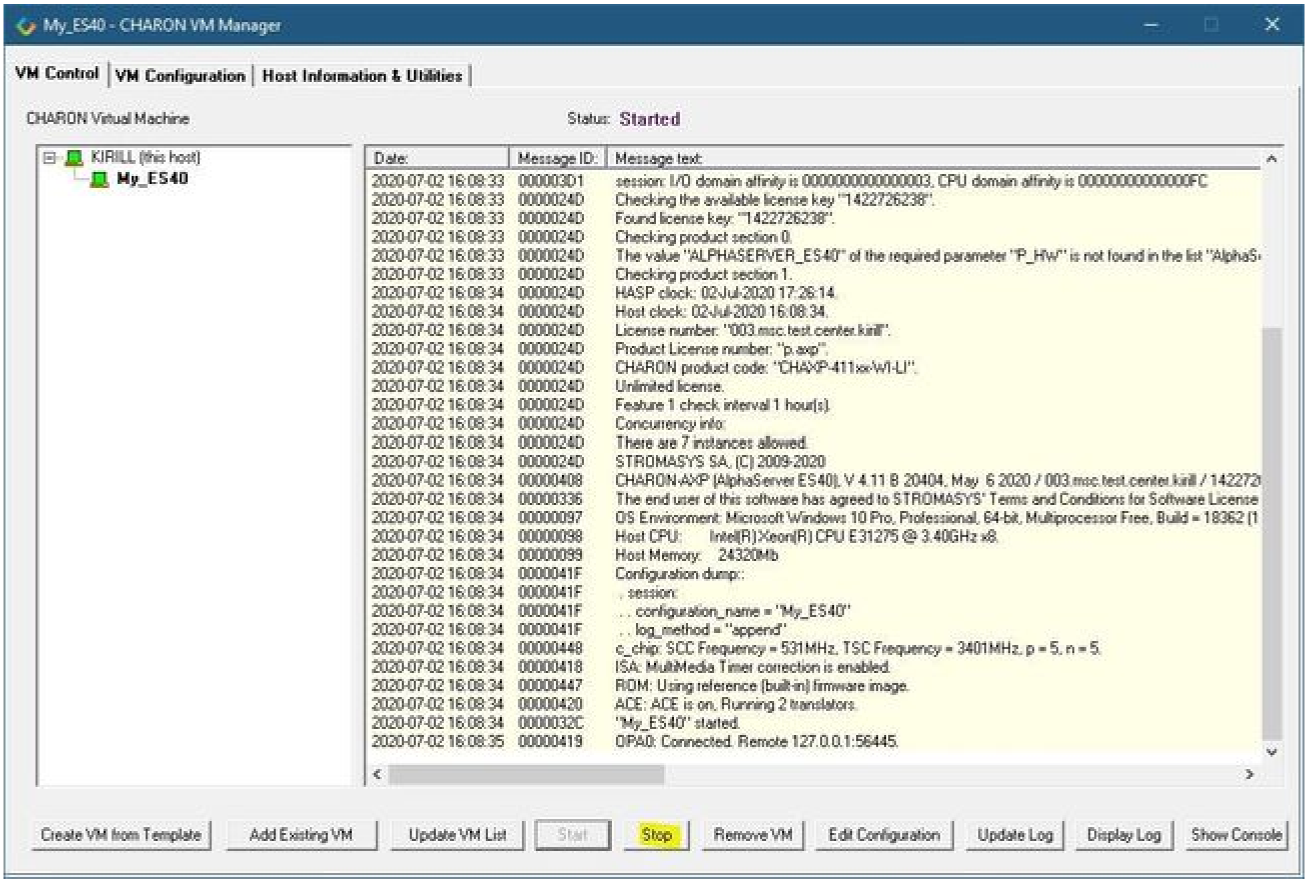


Closing the console will not stop the CHARON Virtual Machine

Stopping a CHARON Virtual Machine

⚠ Stopping a running emulator is the equivalent of using the power-switch to power off a running physical Alpha – if you do this while an operating system is running, you risk data corruption and loss. Thus you should always shut down the Tru64 UNIX/OpenVMS guest before powering off the emulator, if possible. Powering off the emulator while the guest is still running should be the last resort after all attempts to shut the guest down have failed.

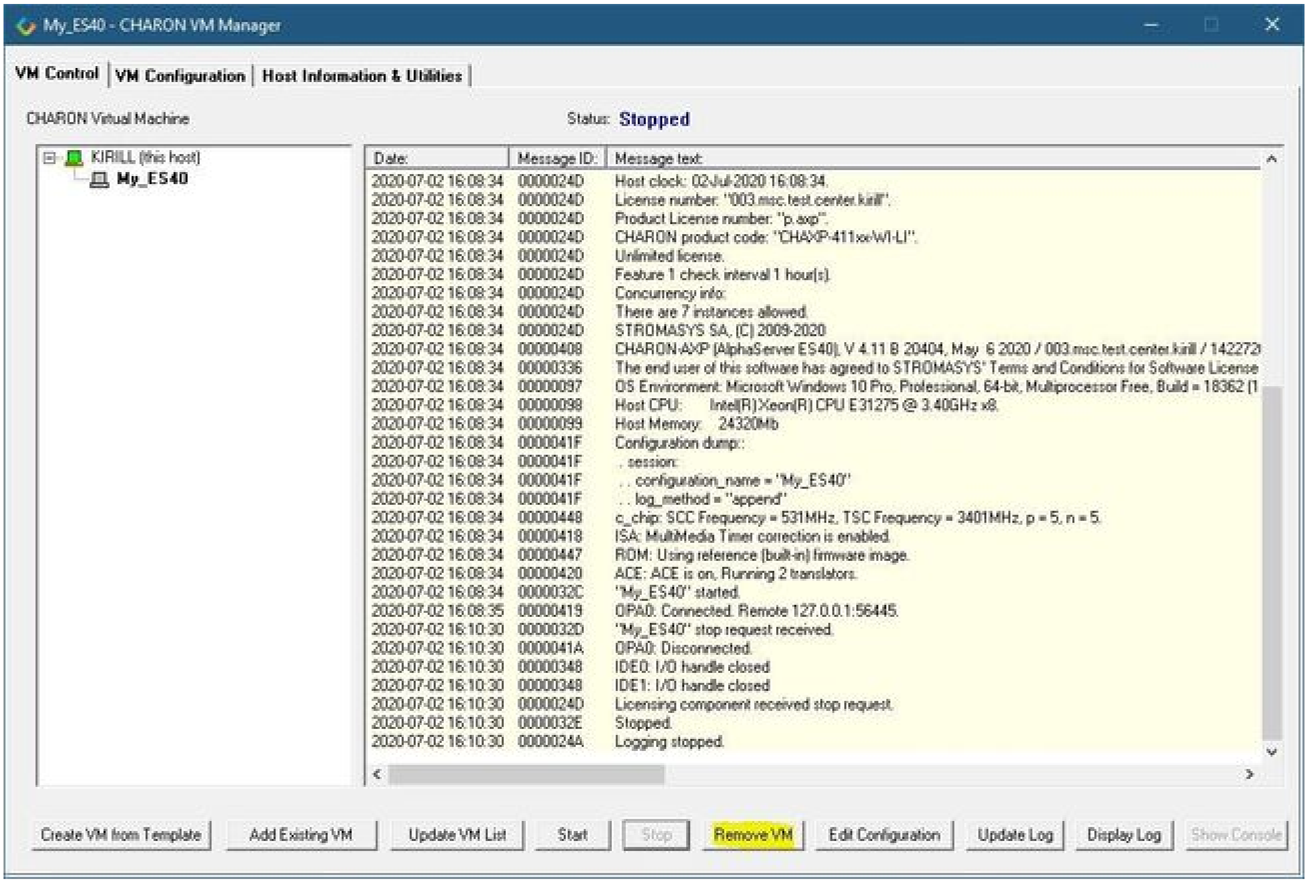
Select the target CHARON VM in the left pane and press the "Stop" button. A dialog will appear to confirm, select "Yes" to continue to stop the VM:



```
Date:               Message ID:  Message text:
2020-07-02 16:08:33 000003D1     session: I/O domain affinity is 0000000000000003, CPU domain affinity is 00000000000000FC
2020-07-02 16:08:33 0000024D     Checking the available license key "1422726238".
2020-07-02 16:08:33 0000024D     Found license key: "1422726238".
2020-07-02 16:08:33 0000024D     Checking product section 0.
2020-07-02 16:08:33 0000024D     The value "ALPHASERVER_ES40" of the required parameter "P_HW" is not found in the list "AlphaS
2020-07-02 16:08:33 0000024D     Checking product section 1.
2020-07-02 16:08:34 0000024D     HASP clock: 02-Jul-2020 17:26:14.
2020-07-02 16:08:34 0000024D     Host clock: 02-Jul-2020 16:08:34.
2020-07-02 16:08:34 0000024D     License number: "003.msc.test.center.kiril".
2020-07-02 16:08:34 0000024D     Product License number: "p.axp".
2020-07-02 16:08:34 0000024D     CHARON product code: "CHAXP-411xx-WI-LI".
2020-07-02 16:08:34 0000024D     Unlimited license.
2020-07-02 16:08:34 0000024D     Feature 1 check interval 1 hour(s).
2020-07-02 16:08:34 0000024D     Concurrency info:
2020-07-02 16:08:34 0000024D     There are 7 instances allowed.
2020-07-02 16:08:34 0000024D     STROMASYS SA, (C) 2009-2020
2020-07-02 16:08:34 00000408     CHARON-AXP (AlphaServer ES40), V 4.11 B 20404, May  6 2020 / 003.msc.test.center.kiril / 142272
2020-07-02 16:08:34 00000336     The end user of this software has agreed to STROMASYS' Terms and Conditions for Software License
2020-07-02 16:08:34 00000097     OS Environment: Microsoft Windows 10 Pro, Professional, 64-bit, Multiprocessor Free, Build = 18362 (1
2020-07-02 16:08:34 00000098     Host CPU:      Intel(R) Xeon(R) CPU E31275 @ 3.40GHz x8.
2020-07-02 16:08:34 00000099     Host Memory:    24320Mb
2020-07-02 16:08:34 0000041F     Configuration dump::
2020-07-02 16:08:34 0000041F      . session:
2020-07-02 16:08:34 0000041F      . . configuration_name = "My_ES40"
2020-07-02 16:08:34 0000041F      . . log_method = "append"
2020-07-02 16:08:34 00000448     e_chip: SCC Frequency = 531MHz, TSC Frequency = 3401MHz, p = 5, n = 5.
2020-07-02 16:08:34 00000418     ISA: MultiMedia Timer correction is enabled.
2020-07-02 16:08:34 00000447     ROM: Using reference (built-in) firmware image.
2020-07-02 16:08:34 00000420     ACE: ACE is on, Running 2 translators.
2020-07-02 16:08:34 0000032C     "My_ES40" started.
2020-07-02 16:08:35 00000419     OPA0: Connected. Remote 127.0.0.1:56445.
```

Removing a CHARON Virtual Machine

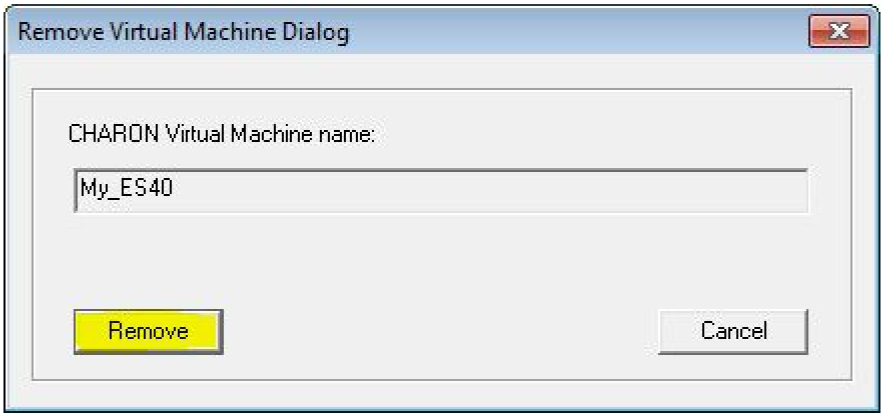
To remove a CHARON VM, select it and pressing the "Remove VM" button:



```
Date:               Message ID:  Message text:
2020-07-02 16:08:34 0000024D     Host clock: 02-Jul-2020 16:08:34.
2020-07-02 16:08:34 0000024D     License number: "003.msc.test.center.kirill".
2020-07-02 16:08:34 0000024D     Product License number: "p.axp".
2020-07-02 16:08:34 0000024D     CHARON product code: "CHAXP-411xx-WI-LI".
2020-07-02 16:08:34 0000024D     Unlimited license.
2020-07-02 16:08:34 0000024D     Feature 1 check interval 1 hour(s).
2020-07-02 16:08:34 0000024D     Concurrency info:
2020-07-02 16:08:34 0000024D     There are 7 instances allowed.
2020-07-02 16:08:34 0000024D     STROMASYS SA, (C) 2009-2020
2020-07-02 16:08:34 00000408     CHARON-AXP (AlphaServer ES40), V 4.11 B 20404, May  6 2020 / 003.msc.test.center.kirill / 142272
2020-07-02 16:08:34 00000336     The end user of this software has agreed to STROMASYS' Terms and Conditions for Software License
2020-07-02 16:08:34 00000097     OS Environment: Microsoft Windows 10 Pro, Professional, 64-bit, Multiprocessor Free, Build = 18362 (1
2020-07-02 16:08:34 00000098     Host CPU:      Intel(R) Xeon(R) CPU E31275 @ 3.40GHz x8.
2020-07-02 16:08:34 00000099     Host Memory:   24320Mb
2020-07-02 16:08:34 0000041F     Configuration dump::
2020-07-02 16:08:34 0000041F      . session:
2020-07-02 16:08:34 0000041F      . . configuration_name = "My_ES40"
2020-07-02 16:08:34 0000041F      . . log_method = "append"
2020-07-02 16:08:34 00000448     c_chip: SCC Frequency = 531MHz, TSC Frequency = 3401MHz, p = 5, n = 5.
2020-07-02 16:08:34 00000418     ISA: MultiMedia Timer correction is enabled.
2020-07-02 16:08:34 00000447     ROM: Using reference (built-in) firmware image.
2020-07-02 16:08:34 00000420     ACE: ACE is on, Running 2 translators.
2020-07-02 16:08:34 0000032C     "My_ES40" started.
2020-07-02 16:08:35 00000419     OPA0: Connected. Remote 127.0.0.1:56445.
2020-07-02 16:10:30 0000032D     "My_ES40" stop request received.
2020-07-02 16:10:30 0000041A     OPA0: Disconnected.
2020-07-02 16:10:30 00000348     IDE0: I/O handle closed
2020-07-02 16:10:30 00000348     IDE1: I/O handle closed
2020-07-02 16:10:30 0000024D     Licensing component received stop request.
2020-07-02 16:10:30 0000032E     Stopped.
2020-07-02 16:10:30 0000024A     Logging stopped.
```

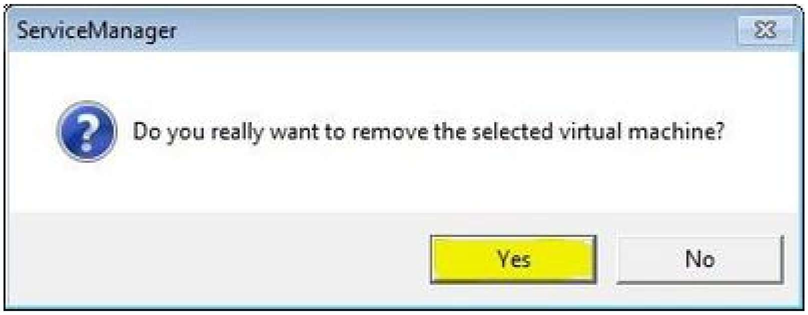
⚠ The Virtual machine must be stopped before removing it (see Running and managing CHARON-AXP for Windows#Stopping a CHARON Virtual Machine chapter above).

A confirmation dialog will be displayed:



CHARON Virtual Machine name: My_ES40

Press the "Remove" button and confirm the CHARON VM removal by pressing the "Yes" button:



Do you really want to remove the selected virtual machine?

Modifying a CHARON Virtual Machine

Select the target CHARON VM and switch to the "VM Configuration" tab to see its features:

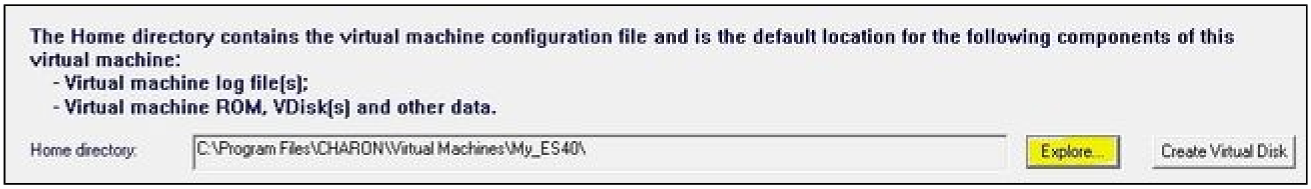
750

Using this dialog it is possible to:

- Explore the Home Directory of the CHARON VM.
- Create virtual disks by pressing the "Create Virtual Disks" button.
- Edit the CHARON VM configuration file.
- Change the startup mode.
- Select the CHARON version used for this Virtual Machine.

Exploring CHARON Virtual Machine Home Directory

Press the "Explore" button to open up a Windows Explorer window showing the content of the selected CHARON VM home folder:



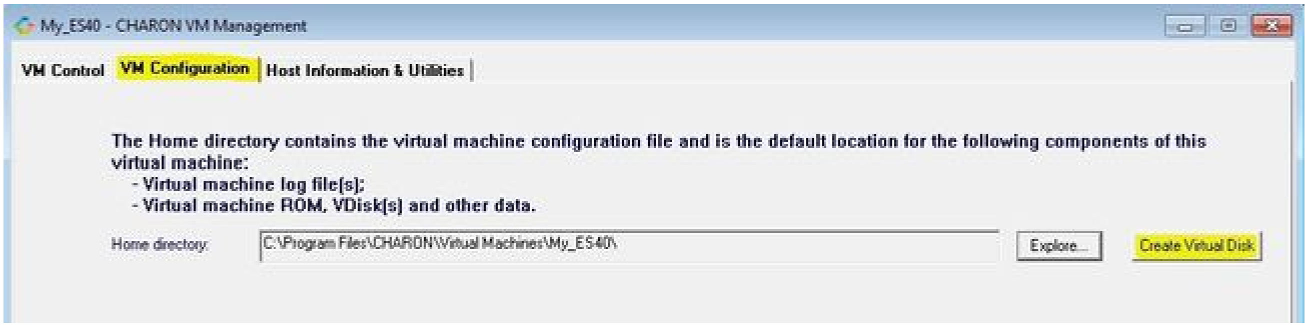
The Home directory contains the virtual machine configuration file and is the default location for the following components of this virtual machine:
- Virtual machine log file(s);
- Virtual machine ROM, VDisk(s) and other data.

Home directory: C:\Program Files\CHARON\Virtual Machines\My_ES40\

This content includes the CHARON VM configuration file and may also include virtual disks and tapes images, logs, ROM files, etc.

Making a new virtual disk

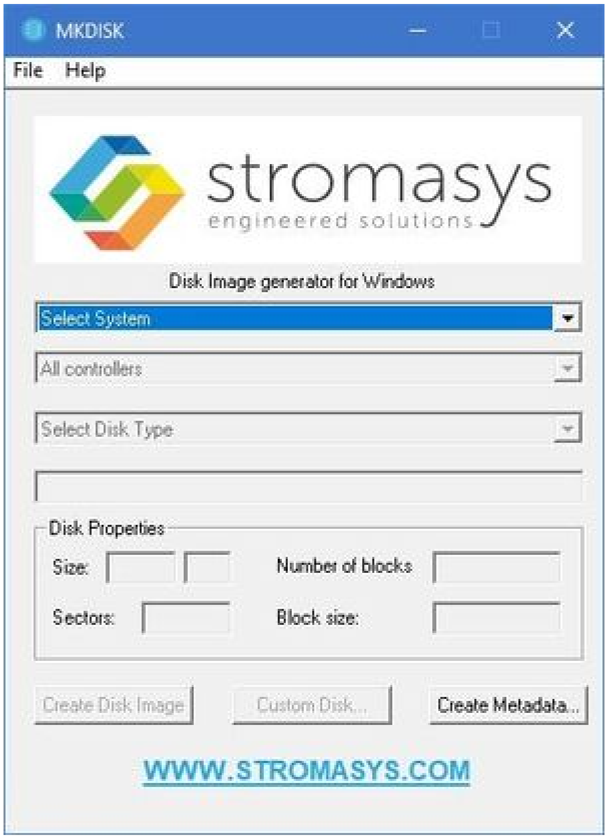
It is very convenient to use the "Create Virtual Disk" button to create empty disk images in the CHARON VM Home Directory:



The Home directory contains the virtual machine configuration file and is the default location for the following components of this virtual machine:
- Virtual machine log file(s);
- Virtual machine ROM, VDisk(s) and other data.

Home directory: C:\Program Files\CHARON\Virtual Machines\My_ES40\

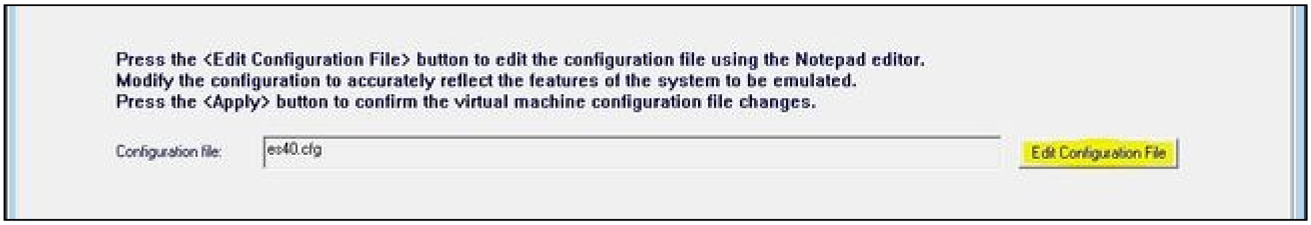
The MkDisk utility will be started for this operation:



Refer to the "Utilities" section of this User's Guide for more information about the MkDisk utility.

CHARON Virtual Machine configuration file

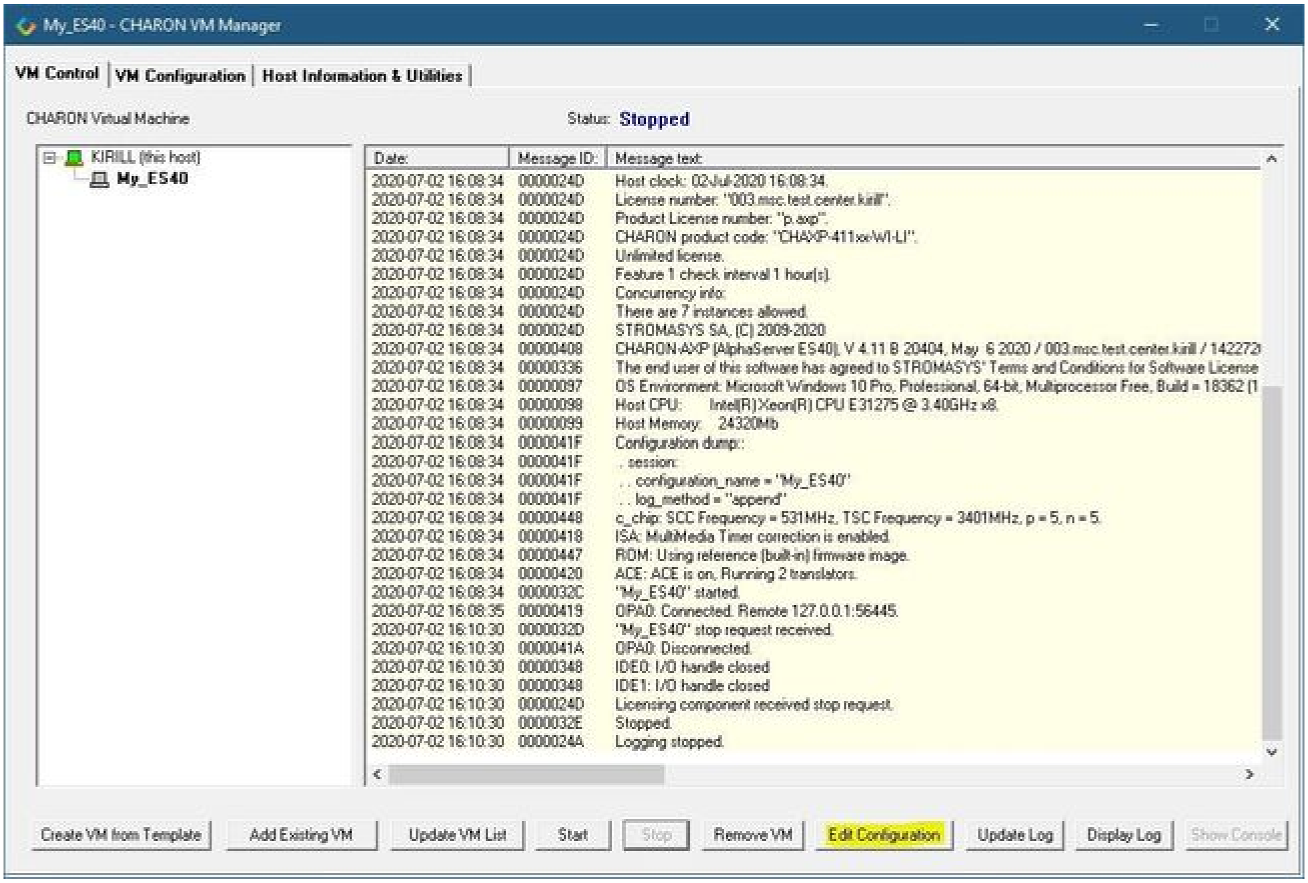
Press the "Edit Configuration file" button:



Press the <Edit Configuration File> button to edit the configuration file using the Notepad editor.
Modify the configuration to accurately reflect the features of the system to be emulated.
Press the <Apply> button to confirm the virtual machine configuration file changes.

Configuration file: es40.cfg

Alternatively it is possible to select the target CHARON VM and press the "Edit Configuration" button:



Status: Stopped

```
Date:                Message ID:  Message text:
2020-07-02 16:08:34  0000024D     Host clock: 02-Jul-2020 16:08:34.
2020-07-02 16:08:34  0000024D     License number: "003.msc.test.center.kirill".
2020-07-02 16:08:34  0000024D     Product License number: "p.axp".
2020-07-02 16:08:34  0000024D     CHARON product code: "CHAXP-411xx-WI-LI".
2020-07-02 16:08:34  0000024D     Unlimited license.
2020-07-02 16:08:34  0000024D     Feature 1 check interval 1 hour(s).
2020-07-02 16:08:34  0000024D     Concurrency info:
2020-07-02 16:08:34  0000024D     There are 7 instances allowed.
2020-07-02 16:08:34  0000024D     STROMASYS SA, (C) 2009-2020
2020-07-02 16:08:34  00000408     CHARON-AXP (AlphaServer ES40), V 4.11 B 20404, May  6 2020 / 003.msc.test.center.kirill / 142272
2020-07-02 16:08:34  00000336     The end user of this software has agreed to STROMASYS' Terms and Conditions for Software License
2020-07-02 16:08:34  00000097     OS Environment: Microsoft Windows 10 Pro, Professional, 64-bit, Multiprocessor Free, Build = 18362 (1
2020-07-02 16:08:34  00000098     Host CPU:      Intel(R) Xeon(R) CPU E31275 @ 3.40GHz x8.
2020-07-02 16:08:34  00000099     Host Memory:    24320Mb
2020-07-02 16:08:34  0000041F     Configuration dump::
2020-07-02 16:08:34  0000041F      . session:
2020-07-02 16:08:34  0000041F      . . configuration_name = "My_ES40"
2020-07-02 16:08:34  0000041F      . . log_method = "append"
2020-07-02 16:08:34  00000448     c_chip: SCC Frequency = 531MHz, TSC Frequency = 3401MHz, p = 5, n = 5.
2020-07-02 16:08:34  00000418     ISA: MultiMedia Timer correction is enabled.
2020-07-02 16:08:34  00000447     ROM: Using reference (built-in) firmware image.
2020-07-02 16:08:34  00000420     ACE: ACE is on, Running 2 translators.
2020-07-02 16:08:34  0000032C     "My_ES40" started.
2020-07-02 16:08:35  00000419     OPA0: Connected. Remote 127.0.0.1:56445.
2020-07-02 16:10:30  0000032D     "My_ES40" stop request received.
2020-07-02 16:10:30  0000041A     OPA0: Disconnected.
2020-07-02 16:10:30  00000348     IDE0: I/O handle closed
2020-07-02 16:10:30  00000348     IDE1: I/O handle closed
2020-07-02 16:10:30  0000024D     Licensing component received stop request.
2020-07-02 16:10:30  0000032E     Stopped.
2020-07-02 16:10:30  0000024A     Logging stopped.
```

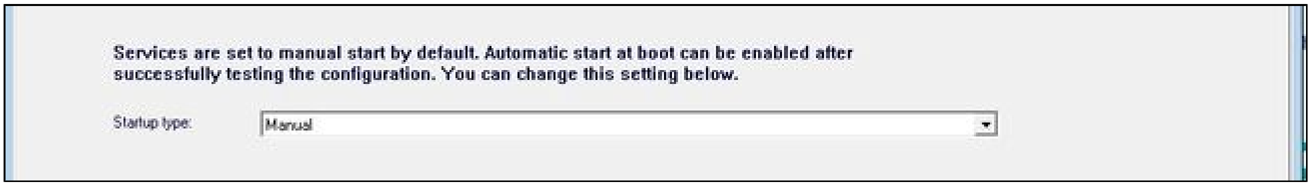
The configuration file specifies all the settings of the CHARON VM. This will be discussed in the next chapter.

CHARON Virtual Machine startup mode

By default, the CHARON VM startup mode is set to "Manual" meaning a user has to start the CHARON VM manually using the CHARON Virtual Machine Manager after a host reboot. If the CHARON VM has to start automatically, select the "Automatic" option.

⚠ Setting the mode to "Automatic" will only start the CHARON VM, it will not boot the operating system. This part will be discussed in the next chapter.

Select the CHARON VM startup behavior in the "Startup type" drop down list:



Services are set to manual start by default. Automatic start at boot can be enabled after successfully testing the configuration. You can change this setting below.

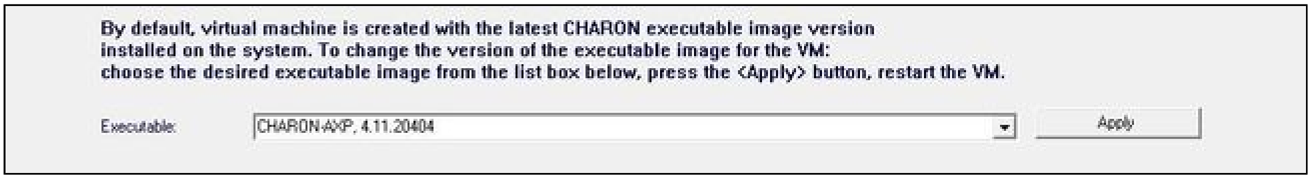
Startup type: Manual

CHARON version to be used for CHARON VM

The CHARON installation procedure allows several versions of CHARON to be installed on the host at the same time (V4.8 and later only). This option can be used for testing.

⚠ If a patch has been applied to a specific version, only the patched version will appear. If you need to roll back to the previous version, please uninstall the patch according to the documentation provided with the patch.

Select the target CHARON version from the "Executable" drop-down list and press the "Apply" button:

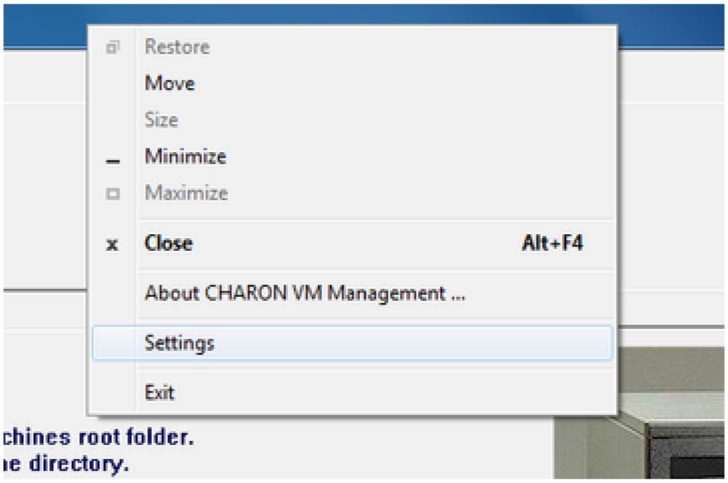
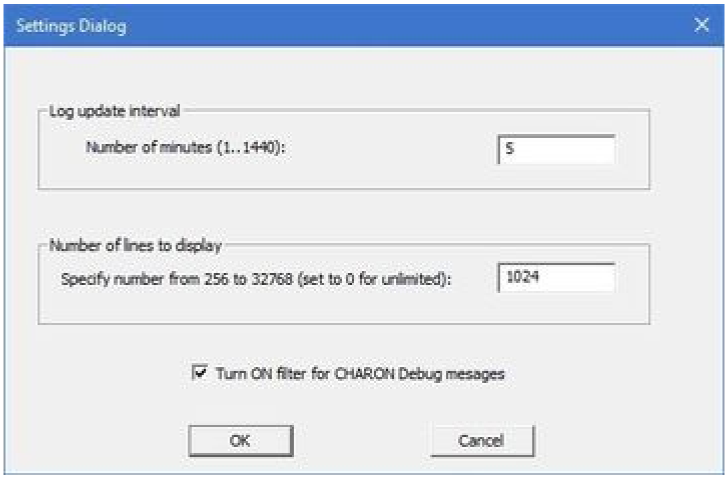


By default, virtual machine is created with the latest CHARON executable image version installed on the system. To change the version of the executable image for the VM: choose the desired executable image from the list box below, press the <Apply> button, restart the VM.

Executable: CHARON-AXP, 4.11.20404

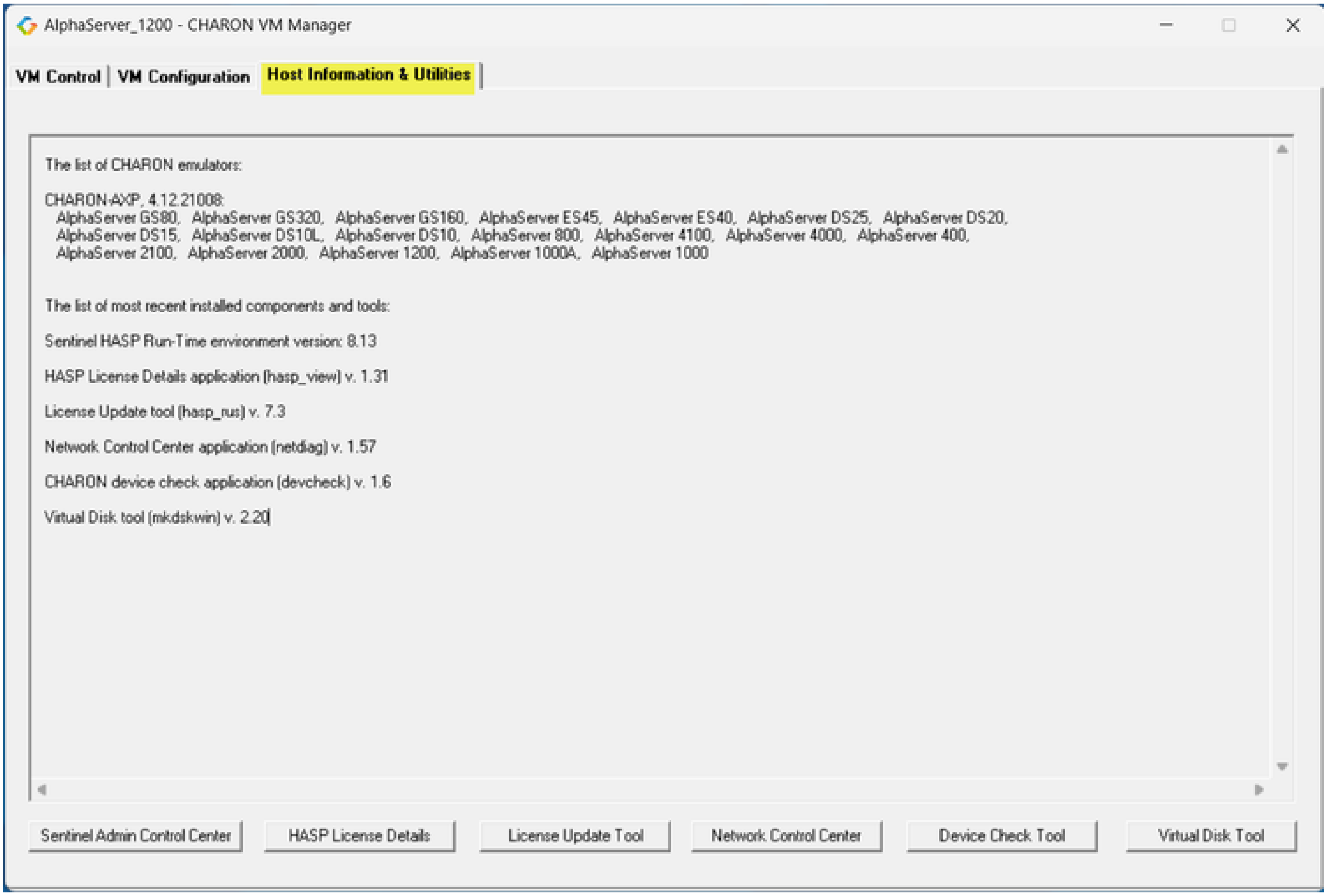
Service functions

Additional functionality is available and described below:

Function	Description
Update Log and Number of lines to display	The CHARON Virtual Machine Manager updates the displayed log every 5 minutes and displays up to 1024 lines by default. It can be changed in the "Settings" dialog of the program system menu. Right click on top of the CHARON Virtual Machine Manager window and select "Settings":  Restore Move Size Minimize Maximize Close Alt+F4 About CHARON VM Management ... Settings Exit chines root folder. ıe directory. In the appearing dialog you can change the interval: Settings Dialog Log update interval Number of minutes (1..1440): 5 Number of lines to display Specify number from 256 to 32768 (set to 0 for unlimited): 1024 ☑ Turn ON filter for CHARON Debug mesages To see most up-to-date version of the log, press the "Update Log" button.
Display Log	Available from the VM control tab, bottom right. Creates a separate window containing the selected CHARON VM log
Show Console	Available from the VM control tab, bottom right, if the VM is running. Opens the selected CHARON VM console if it has been closed

Running CHARON utilities

Select the "Host Information & Utilities" tab to get a quick access to the CHARON utilities:



```
The list of CHARON emulators:

CHARON-AXP, 4.12.21008:
  AlphaServer GS80,  AlphaServer GS320,  AlphaServer GS160,  AlphaServer ES45,  AlphaServer ES40,  AlphaServer DS25,  AlphaServer DS20,
  AlphaServer DS15,  AlphaServer DS10L,  AlphaServer DS10,  AlphaServer 800,  AlphaServer 4100,  AlphaServer 4000,  AlphaServer 400,
  AlphaServer 2100,  AlphaServer 2000,  AlphaServer 1200,  AlphaServer 1000A,  AlphaServer 1000

The list of most recent installed components and tools:

Sentinel HASP Run-Time environment version: 8.13

HASP License Details application (hasp_view) v. 1.31

License Update tool (hasp_rus) v. 7.3

Network Control Center application (netdiag) v. 1.57

CHARON device check application (devcheck) v. 1.6

Virtual Disk tool (mkdskwin) v. 2.20
```

These utilities are described below:

Button	Description
Sentinel Admin Control Center	Used to view CHARON licenses and manage them.
HASP License Details	Used to display the CHARON license content.
License Update Tool	Used to manage the CHARON licenses, collect the host system fingerprint.
Network Control Center	Used to configure the CHARON network.
Device Check tool	Used to review the system resources that can be mapped to CHARON.
Virtual Disk Tool	GUI-based utility used to create custom or standard CHARON virtual disk containers.

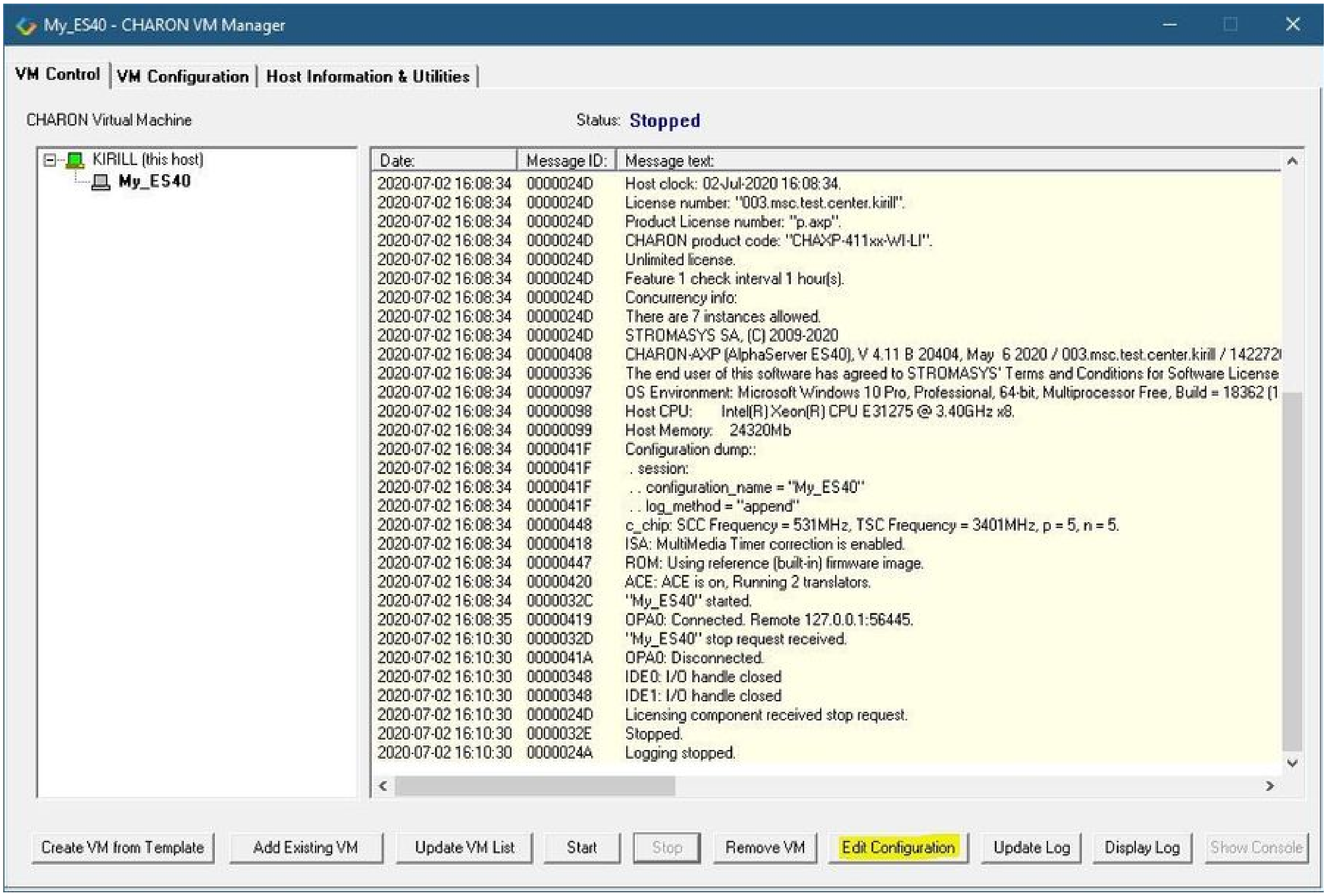
CHARON-AXP for Windows configuration

Table of Contents

Creation of CHARON VM configuration

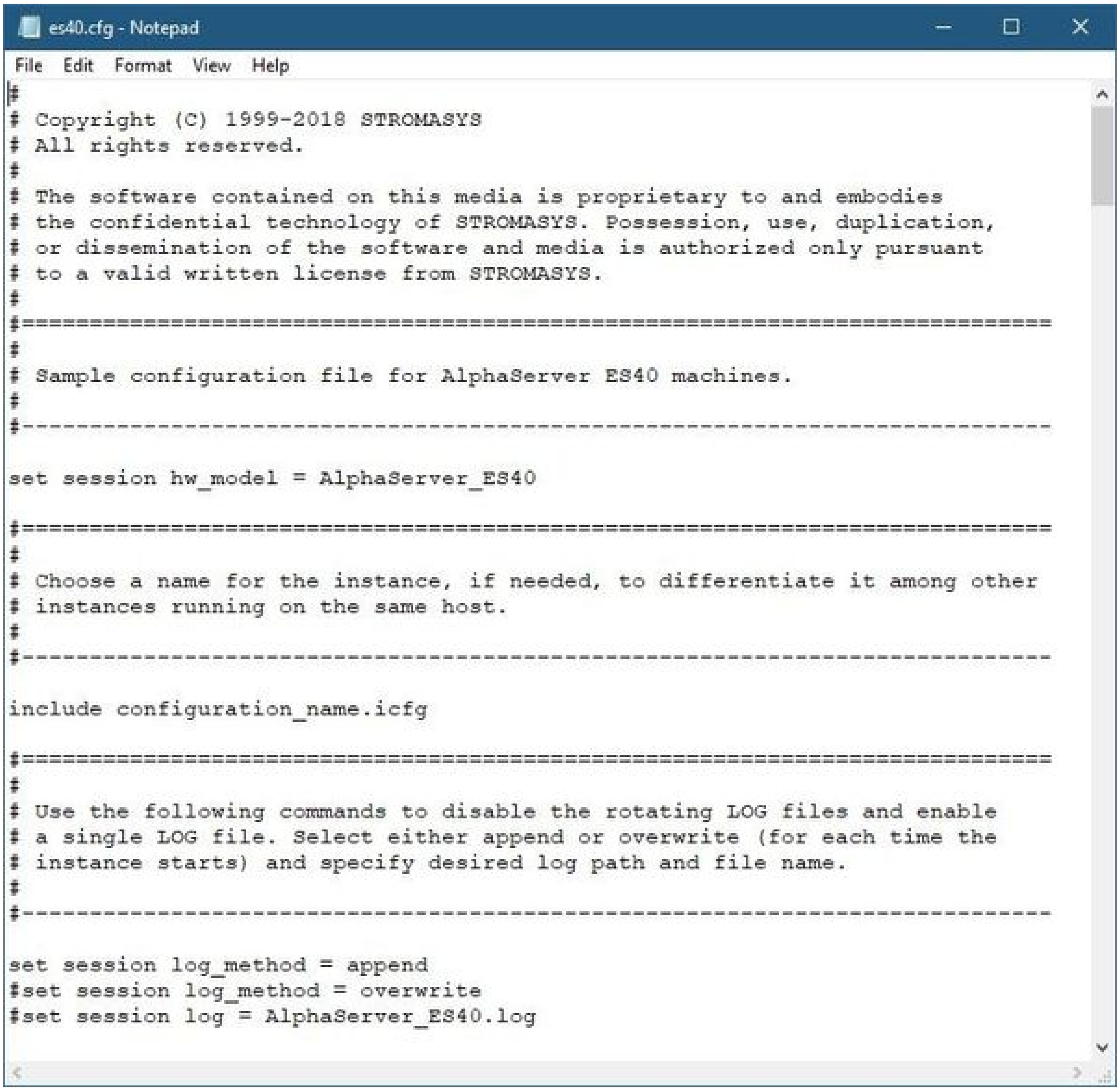
When a CHARON Virtual Machine (VM) is created using the CHARON Virtual Machine Manager, the text-based configuration file is copied from a template file. This configuration file must be customized for your specific configuration. The configuration file consists of directives written in the Charon-AXP "configuration syntax" which define all aspects of the emulated system, including amount of memory, number of CPUs, peripheral devices, the location of various associated files, to name some examples.

To edit the configuration file, open the CHARON Virtual Machine Manager from its shortcut on Desktop / Start menu or from the tray menu item, select the target CHARON VM and press the "Edit Configuration" button:



```
Date:                Message ID:   Message text:
2020-07-02 16:08:34  0000024D      Host clock: 02-Jul-2020 16:08:34.
2020-07-02 16:08:34  0000024D      License number: "003.msc.test.center.kirill".
2020-07-02 16:08:34  0000024D      Product License number: "p.axp".
2020-07-02 16:08:34  0000024D      CHARON product code: "CHAXP-411xx-WI-LI".
2020-07-02 16:08:34  0000024D      Unlimited license.
2020-07-02 16:08:34  0000024D      Feature 1 check interval 1 hour(s).
2020-07-02 16:08:34  0000024D      Concurrency info:
2020-07-02 16:08:34  0000024D      There are 7 instances allowed.
2020-07-02 16:08:34  0000024D      STROMASYS SA, (C) 2009-2020
2020-07-02 16:08:34  00000408      CHARON-AXP (AlphaServer ES40), V 4.11 B 20404, May  6 2020 / 003.msc.test.center.kirill / 142272
2020-07-02 16:08:34  00000336      The end user of this software has agreed to STROMASYS' Terms and Conditions for Software License
2020-07-02 16:08:34  00000097      OS Environment: Microsoft Windows 10 Pro, Professional, 64-bit, Multiprocessor Free, Build = 18362 (1
2020-07-02 16:08:34  00000098      Host CPU:      Intel(R) Xeon(R) CPU E31275 @ 3.40GHz x8.
2020-07-02 16:08:34  00000099      Host Memory:    24320Mb
2020-07-02 16:08:34  0000041F      Configuration dump::
2020-07-02 16:08:34  0000041F       . session:
2020-07-02 16:08:34  0000041F       . . configuration_name = "My_ES40"
2020-07-02 16:08:34  0000041F       . . log_method = "append"
2020-07-02 16:08:34  00000448      c_chip: SCC Frequency = 531MHz, TSC Frequency = 3401MHz, p = 5, n = 5.
2020-07-02 16:08:34  00000418      ISA: MultiMedia Timer correction is enabled.
2020-07-02 16:08:34  00000447      ROM: Using reference (built-in) firmware image.
2020-07-02 16:08:34  00000420      ACE: ACE is on, Running 2 translators.
2020-07-02 16:08:34  0000032C      "My_ES40" started.
2020-07-02 16:08:35  00000419      OPA0: Connected. Remote 127.0.0.1:56445.
2020-07-02 16:10:30  0000032D      "My_ES40" stop request received.
2020-07-02 16:10:30  0000041A      OPA0: Disconnected.
2020-07-02 16:10:30  00000348      IDE0: I/O handle closed
2020-07-02 16:10:30  00000348      IDE1: I/O handle closed
2020-07-02 16:10:30  0000024D      Licensing component received stop request.
2020-07-02 16:10:30  0000032E      Stopped.
2020-07-02 16:10:30  0000024A      Logging stopped.
```

Notepad will be used to edit the configuration file:



```
#
# Copyright (C) 1999-2018 STROMASYS
# All rights reserved.
#
# The software contained on this media is proprietary to and embodies
# the confidential technology of STROMASYS. Possession, use, duplication,
# or dissemination of the software and media is authorized only pursuant
# to a valid written license from STROMASYS.
#
#=============================================================================
#
# Sample configuration file for AlphaServer ES40 machines.
#
#-----------------------------------------------------------------------------

set session hw_model = AlphaServer_ES40

#=============================================================================
#
# Choose a name for the instance, if needed, to differentiate it among other
# instances running on the same host.
#
#-----------------------------------------------------------------------------

include configuration_name.icfg

#=============================================================================
#
# Use the following commands to disable the rotating LOG files and enable
# a single LOG file. Select either append or overwrite (for each time the
# instance starts) and specify desired log path and file name.
#
#-----------------------------------------------------------------------------

set session log_method = append
#set session log_method = overwrite
#set session log = AlphaServer_ES40.log
```

Alpha model specification

The first configuration statement is the specification of the exact Alpha hardware model to emulate, for example:

```
set session hw_model = AlphaServer_ES40
```

This line must not be changed.

⚠ If you create the CHARON VM configuration file from scratch, it must be the very first uncommented line in the configuration file.

Configuration name

The next configuration statement is the "Configuration name" option. If the virtual machine has been created using an existing template, the configuration name is defined in the configuration_name.icfg file otherwise it is defined directly in the configuration file using the "set session configuration_name = <...>" syntax:

```
include configuration_name.icfg
```

Notes:

- The configuration name is reported in the log file and is used to set the log file name for rotating log (see explanations further).
- Changing the configuration name in the "configuration_name.icfg" file does not change the name of the virtual machine at Virtual Machine Manager level.
- The "configuration_name.icfg" file can be found in the home directiory of the VM. Select the target VM, open "VM Configuration" tab and press "Explore" button.

The configuration name can be any label that is meaningful.

Example:

```
set session configuration_name = My_ES40
```

It is possible to specify a configuration name containing spaces, in this case use quotation marks:

```
set session configuration_name = "My ES40"
```

Log file parameters

The execution of a CHARON VM creates one log file or a set of log files reflecting the progress of its start-up and ongoing operations: start and end time of execution, system information, license and configuration details, warnings, reports on problems that may occur, etc. In case of problems with either the running CHARON VM or the emulated system configuration (such as the absence or malfunction of certain devices), the log file is the primary source to be analyzed for troubleshooting.

If it becomes necessary to contact Stromasys for support, the configuration and log files, plus the license number, will be requested to begin the problem resolution.

CHARON-AXP log file example (part1)

```
20200524:160742:INFO :0:00000249:ethane.cxx(9026): Logging started.
20200524:160742:INFO :0:000003A5:ethane.cxx( 147): session: loading built-in configuration
"AlphaServer_ES40"...
20200524:160742:INFO :0:000003A6:ethane.cxx( 179): session: ... done loading built-in configuration
"AlphaServer_ES40"
20200524:160742:INFO :0:000003A7:ethane.cxx( 337): session: loading service configuration "My_ES40"...
20200524:160742:INFO :0:000003A8:ethane.cxx( 372): session: ... done loading service configuration "My_ES40"
20200524:160742:INFO :0:000003AA:ethane.cxx( 420): session: loading configuration file "es40.cfg"...
20200524:160742:INFO :0:000003AB:ethane.cxx( 456): session: ... done loading configuration file "es40.cfg"
20200524:160742:INFO :0:0000032B:ethane.cxx(2707): Start request received.
20200524:160742:INFO :0:000003AC:ethane.cxx( 818): session: process affinity is 00000000000000FF, system
affinity is 00000000000000FF
20200524:160742:INFO :0:000003D1:ethane.cxx(1476): session: I/O domain affinity is 0000000000000003, CPU
domain affinity is 00000000000000FC
20200524:160742:INFO :0:0000024D:licenseman(1823): Checking the available license key "1422726238".
20200524:160742:INFO :0:0000024D:licenseman(1823): Found license key: "1422726238".
20200524:160742:INFO :0:0000024D:licenseman(1823): Checking product section 0.
20200524:160742:INFO :0:0000024D:licenseman(1823): The value "ALPHASERVER_ES40" of the required parameter
"P_HW" is not found in the list "PDP1193,PDP1194" for the product 0.
20200524:160742:INFO :0:0000024D:licenseman(1823): Checking product section 1.
20200524:160743:INFO :0:0000024D:licenseman(1823): HASP clock: 24-May-2020 17:08:01.
20200524:160743:INFO :0:0000024D:licenseman(1823): Host clock: 24-May-2020 16:07:43.
20200524:160743:INFO :0:0000024D:licenseman(1823): License number: "003.msc.test.center.kirill".
20200524:160743:INFO :0:0000024D:licenseman(1823): Product License number: "p.axp".
20200524:160743:INFO :0:0000024D:licenseman(1823): CHARON product code: "CHAXP-4xxxx-WI-LI".
20200524:160743:INFO :0:0000024D:licenseman(1823): Unlimited license.
20200524:160743:INFO :0:0000024D:licenseman(1823): Feature 2 check interval 1 hour(s).
20200524:160743:INFO :0:0000024D:licenseman(1823): Concurrency info:
20200524:160743:INFO :0:0000024D:licenseman(1823): There are 4 instances allowed.
```

```
20200524:160743:INFO :0:0000024D:ethane.cxx(2838): STROMASYS SA, (C) 2009-2020
20200524:160743:INFO :0:00000408:ethane.cxx(2880): CHARON-AXP (AlphaServer ES40), V 4.11 B 20404, May 24
2020 / 003.msc.test.center.kirill / 1422726238
20200524:160743:INFO :0:00000336:ethane.cxx(2921): The end user of this software has agreed to STROMASYS'
Terms and Conditions for Software License and Limited Warranty, as described at: http://www.stromasys.com/pub
/doc/30-17-033.pdf
20200524:160743:INFO :0:00000097:ethane.cxx(3000): OS Environment: Microsoft Windows 10 Pro, 64-bit (Build
17134).
20200524:160743:INFO :0:00000098:ethane.cxx(3005): Host CPU: Intel(R) Xeon(R) CPU E31275 @ 3.40GHz x8.
20200524:160743:INFO :0:00000099:ethane.cxx(3010): Host Memory: 24320Mb
20200524:160743:INFO :0:0000041F:ethane.cxx(3236): Configuration dump::
. session:
. . configuration_name = "ES40 test"
. . log = "log.txt"

. . log_method = "overwrite"
. . log_locale = "English"
. . log_show_messages = "all"
. . log_repeat_filter = "off"
. c_chip:
. . mm_timer_correction_disable = "false"
. RAM:
. . size = "512"
. ACE:
. . num_entries = "2139"
. . num_translators = "0"
. . cache_size = "1024"
. . cache_base_size = "200"
. . host_options = " --locked-size=16"
. . enabled = "true"
. . ext_compiler = "ml64.exe"
. . ext_path = ""
. . cpu_architecture = "EV6"
. . locked_size = "16"
```

CHARON-AXP log file example (part2)

```
. axp_bus_0:
. . mm_timer_correction_disable = "false"
. cpu_0:
. . locked_size = "16"
. . wtint_idle = "true"
. axp_bus_1:
. . mm_timer_correction_disable = "false"
. cpu_1:
. . locked_size = "16"
. . wtint_idle = "true"
. axp_bus_2:
. . mm_timer_correction_disable = "false"
. cpu_2:
. . locked_size = "16"
. . wtint_idle = "true"
. axp_bus_3:
. . mm_timer_correction_disable = "false"
. cpu_3:
. . locked_size = "16"
. . wtint_idle = "true"
. ROM:
. . container = "clipper.bin"
. . dsrdb[0] = "1816"
. . dsrdb[1] = "50"
. . dsrdb[4] = "50"
. . dsrdb[11] = "1050"
. . dsrdb[12] = "1050"
. . system_name = "AlphaServer ES40 6/500"
. pci_0:
. . mm_timer_correction_disable = "false"
. pci_1:
. . mm_timer_correction_disable = "false"
. ISA:
. . clock_period = "10000"
. . mm_timer_correction_disable = "false"
. TOY:
. . sync_to_host = "as_vms, nowrite"
. COM1:
. . line = "OPA0"
. . communication = "console"
. COM2:
. . line = "(void)"
. . communication = "ascii"
. OPA0:
. . trace = "disabled"
. . port = "10003"
. . application = "putty -load OPA0"
. EWA:
. . adapter_mode = "auto"
. . interface = "EWA0"
. . rx_fifo_delay_on_overload = "false"

. EWA0:
. . interface = "(disabled)"
. . port_show_driver_statistics = "false"
. . port_enable_mac_addr_change = "true"
. . port_delay_tx_on_overload = "true"
. . suspend_msg_on_mac_change = "true"
. . legacy_mode = "false"
. PKA:
. . scsi_id = "7"
. . min_n_of_threads = "0"
. . container[0] = "C:\Dev\test\performancecomparison-axp.vdisk"
. . container[400] = "C:\Dev\test\ALPHA084.ISO"
. . container[600] = "C:\Dev\test\test_tape.vtape"
. . removable[600] = "true"
```

CHARON-AXP log file example (part3)

```
20200524:160743:INFO :0:00000418:busemul.cx( 188): ISA: MultiMedia Timer correction is enabled.
20200524:160743:INFO :0:00000420:dit_server(1038): ACE: ACE is on, Running 2 translators.
20200524:160743:INFO :0:0000034B:scsi_disk.( 485): PKA_0: open I/O handle
container = "C:\Dev\test\performancecomparison-axp.vdisk"
```

```
20200524:160743:INFO :0:0000034B:scsi_cdrom( 211): PKA_400: open I/O handle
container = "C:\Dev\test\ALPHA084.ISO"
20200524:160743:INFO :0:0000034B:scsi_tape.( 348): PKA_600: open I/O handle
container = "C:\Dev\test\test_tape.vtape"
20200524:160743:INFO :0:00000400:ethdev.cxx( 388): EWA: RX FIFO size is 512KB.
20200524:160743:INFO :0:0000032C:ethane.cxx(2743): "ES40 test" started.
20200524:160743:INFO :0:00000419:winsckio.c(2703): OPA0: Connected. Remote 127.0.0.1:65128.
```

The next group of parameters defines the name of the CHARON VM log file and how the CHARON VM will use it:

```
set session log_method = append
#set session log_method = overwrite
#set session log = "AlphaServer_ES40.log"
```

Rotating log (default)

By default CHARON utilizes a so-called "rotating log" mechanism. This means that a new default log file is created each time CHARON starts and can switch to another log file if the size of the log file exceeds 64Kb (this behavior can be changed with the "set session log_file_size" and "set session log_rotation_period" parameters; see more details in the "General Settings" chapter of this guide).

This mode is turned on if all the log parameters above are disabled (commented out) or the "session_log" parameter is pointing to a directory rather than to a file. If a directory is specified, the log files will be created in that directory.

The names of the rotating log files are composed as follows:

```
configuration_name-YYYY-MM-DD-hh-mm-ss-xxxxxxxxx.log
```

If the "Configuration name" parameter described before is omitted (commented out), the log name has the following format instead:

```
hw_model-YYYY-MM-DD-hh-mm-ss-xxxxxxxxx.log
```

Note that "xxxxxxxxx" is an increasing decimal number starting from "000000000" to separate log files with the same time of creation.

⚠ The "log" parameter, if specified, must correspond to an existing folder.

ℹ If the path is not specified, the log file is created in the "Home directory" mentioned in the VM Configuration tab of the CHARON Virtual Machine Manager.

Single log

Alternatively it is possible to use a single log file. Uncomment the "set session log" line and specify the desired log file name. Optionally, a path can be added to the log file name.

ℹ If the path is not specified, the log file is created in the "Home directory" mentioned in the VM Configuration tab of the CHARON Virtual Machine Manager.

The log file can be extended specifying "log_method = append" (*recommended for reporting issues*) or overwritten, specifying "log_method = overwrite".

Below is a specification of a CHARON VM log file located in the "`C:\CHARON logs`" directory which will be appended each time the CHARON VM starts:

```
set session log_method = append
set session log = "C:\CHARON logs\my_ES40.log"
```

CPU affinity

This setting binds the running CHARON VM CPUs to particular host CPUs.This should be used for soft partitioning host CPU resources or for isolating multiple CHARON VMs on the same host from each other. By default the emulator instance allocates as many host CPUs as possible.

“Affinity” overrides the default and allows explicit specification of which host CPUs will be used by the instance. Affinity does not reserve the CPU for exclusive use.

```
set session affinity="0, 1, 2, 3"
```

The example above directs CHARON VM to use CPU 0,1,2 and 3.

If this parameter is omitted CHARON host will allocate available CPUs automatically.

⚠ Note that the number of the specified host CPUs must correspond to the number of the emulated CPUs (one host CPU for one emulated CPU; this value is specific for each Alpha model) and number of CPUs needed for CHARON application itself ("n_of_io_cpus").

Number of host CPUs dedicated to CHARON I/O

This setting reserves host CPUs (of those specified by “affinity” parameter, if any) for use by CHARON VM for I/O handling. By default CHARON VM reserves one third of available host CPUs for I/O processing (round down, at least one).

The “n_of_io_cpus” overrides the default by specifying the number of I/O host CPUs explicitly

Example:

```
set session n_of_io_cpus=2
```

The example above directs CHARON VM to use 2 host CPUs for CHARON I/O operations.

⚠ Note that the number of the specified CPUs dedicated to CHARON VM I/O operations must correspond to the total number of available for CHARON CPUs (restricted by "affinity" parameter if needed) and the number of the virtual Alpha CPUs to be emulated.

Setting a specific Alpha model

CHARON-AXP allows to specify an exact model of Alpha.

For example for AlphaServer ES40 family the template configuration file contains the following options:

```
#============================================================================
#
# AlphaServer ES40 6/500
#
#----------------------------------------------------------------------------


#set ace cpu_architecture = EV6
#set rom dsrdb[0] = 1816 system_name = "AlphaServer ES40 6/500"
#set rom version[1] = 1.98-4 version[2] = 1.92-5

#============================================================================
#
# AlphaServer ES40 6/667
#
#----------------------------------------------------------------------------

set ace cpu_architecture = EV67
set rom dsrdb[0] = 1820 system_name = "AlphaServer ES40 6/667"
```

Just uncomment the provided lines to apply a certain model (It is "AlphaServer ES40 6/667" in the example above).

The full description of the parameters and other models that can be also configured is available in the "Configuration details" chapter of this User's Guide.

Reducing number of emulated CPUs

If the CHARON host does not contain enough CPUs to emulate full range of the CPUs provided by a certain Alpha model, it is possible to direct the CHARON VM to reduce the number of emulated Alpha CPUs:

```
set session n_of_cpus=1
```

This parameter can also be used to avoid warning messages in the log if the number of CPUs allowed by the license is less than the default number of CPUs of the emulated Alpha model.

Setting system serial number

If necessary, a specific system serial number instead of the default one:

```
set rom system_serial_number = SN01234567
```

TOY and ROM containers

The TOY and ROM containers have to be configured. Their presence depends on the Alpha model. It is always recommended to enable them. If a container file of the given name does not exist, starting the CHARON VM will create it.

TOY means "Time of Year". Its container records time, date and some console parameters while the CHARON VM is not running. It is highly recommended to define and activate this container:

```
set toy container="clipper.dat"
```

The ROM container stores an intermediate state of the Flash ROM and some console parameters. It is highly recommended to define and activate this container:

```
set rom container="clipper.bin"
```

Emulated memory (RAM) size

The next parameter defines the amount of host memory the CHARON VM reserves for the emulation:

```
#set ram size=4096
set ram size=32768
```

The amount of RAM is specified in MB. It cannot exceed or be lower than certain values specific for each Alpha model. It is very important to keep the listed predefined increment between possible memory values.

The following table shows all the parameters:

Hardware Model	RAM size (in MB)			
	Min	Max	Default	Increment
AlphaServer 400	64	1024	512	64
AlphaServer 800	256	8192	512	256
AlphaServer 1000	256	1024	512	256
AlphaServer 1000A	256	1024	512	256
AlphaServer 1200	256	32768	512	256
AlphaServer 2000	64	2048	512	64
AlphaServer 2100	64	2048	512	64
AlphaServer 4000	64	32768	512	64
AlphaServer 4100	64	32768	512	64
AlphaServer DS10	64	32768	512	64
AlphaServer DS10L	64	32768	512	64
AlphaServer DS15	64	32768	512	64
AlphaServer DS20	64	32768	512	64
AlphaServer DS25	64	32768	512	64
AlphaServer ES40	64	32768	512	64
AlphaServer ES45	64	32768	512	64
AlphaServer GS80	256	65536	512	256
AlphaServer GS160	512	131072	512	512
AlphaServer GS320	1024	262144	1024	1024

It is possible to leave the RAM line commented out. In this case the model's default RAM amount is used.

Note that in some particular orders your license may restrict the maximum RAM amount of each Alpha model.

Console

Mapping to system resources

The next step is the specification of the Alpha console (OPA0) serial line.

Example:

```
#set COM1 alias = OPA0 line = "COM1:"
#set COM1 alias = OPA0 port = 10003
#set COM1 alias = OPA0 port = 10003 application = "opa0.ht"
set COM1 alias = OPA0 port = 10003 application = "putty -load OPA0 -P 10003"
#set COM1 alias = OPA0 port = 10003 application = "c:\kea\user\opa0.ktc"
```

The goal of this configuration step is to tell CHARON-AXP what host device to use as the virtual system console. The following options are available:

Option	Description
line	Mapping to host serial line, "COM<n>:".
port	Mapping to an IP port of the CHARON host. Using this mapping it is possible to connect to the CHARON console and disconnect from it at any time.
application	Starting some application (typically terminal emulator) with its specific options and switches to communicate to CHARON using the IP port defined by the "port" parameter (see above)
alias	Define some meaningful name for "COM1" and "COM2". Usually it is "OPA0" for "COM1" and "TTA0" for "COM2" (see below)

The second console line "TTA0" can be also optionally configured (for 1 CPU models such as AlphaServer 400, AlphaServer 800, AlphaServer 1000, AlphaServer 1000A, AlphaServer DS10, AlphaServer DS10L and AlphaServer DS15):

```
#set COM2 alias = TTA0 line = "COM2:"
#set COM2 alias = TTA0 port = 10000
#set COM2 alias = TTA0 port = 10000 application = "tta0.ht"
set COM2 alias = TTA0 port = 10000 application = "putty -load TTA0 -P 10000"
#set COM2 alias = TTA0 port = 10000 application = "c:\kea\user\tta0.ktc"
```

Note that additional parameters for the CHARON VM serial lines configuration can be added. Follow this link for details.

Exit on pressing F6 button

The following directive configures the emulator to exit immediately when the F6 key sequence is sent to the emulator console:

```
set OPA0 stop_on = F6
```

This means that if the F6 key is pressed in the terminal emulator being used to connect to the console (PuTTY, for example), the emulator will exit immediately.

Please note that the emulator can be stopped when at the SRM prompt ("P00>>>") by type **`power off`**, as well as from the Virtual Machine Manager. To shut down the guest safely, Stromasys recommends the following procedure:

- Shut down the guest OS (OpenVMS/Tru64 UNIX)
- Wait for the SRM prompt ("P00>>>")
- Type **`power off`** at the SRM prompt
- The connection to the console will be lost when the emulator exits, which means the terminal emulator window will probaly disappear.
- At this point you can stop or restart the emulator host safely

Stromasys does not recommend enabling this feature. It means that any person (or software) which connects to the console port on the emulator host can stop the emulator and thus "pull the virtual power cable" from the OpenVMS or Tru64 UNIX guest. In particular, many organizations have deployed automated port scanners in their datacenters to discover open ports on the systems in the network. Many such port-scanning solutions, if they are able to connect to an open port, will next try to send data to the port. In many cases this data can be interpreted as the stop sequence and cause the emulator to exit.

If you must allow incoming connections to the console port, you should configure the Windows firewall with a "whitelist" of IP addresses from which such connections are allowed, and disallow all other addresses.

Improve granularity of emulated timer

The next configuration option can be applied for improving granularity of emulated CHARON-AXP timer:

```
#set isa clock_period=1000
```

Do not uncomment this parameter unless there are some problems with the system time or the system clock intervals in the guest OS.

ATAPI CD/DVD-ROM configuration

If the sample configuration file provides this parameter it is possible to map this particular CHARON VM emulator's "DQA0" CD-ROM to the host CD/DVD-ROM with the following setting:

```
set ide container="\\.\CdRom0"
```

Networking

CHARON-AXP supports DE435, DE450, DE500AA, DE500BA, DE602 and DE602AA virtual network adapters.

All of them are configured in a similar way:

```
load DE500BA/dec21x4x EWA interface=EWA0
load packet_port/chnetwrk EWA0 interface="connection:Charon"
```

```
load DE602/i8255x EIA interface=EIA0
load packet_port/chnetwrk EIA0 interface="connection:Charon"
```

In the examples above the first line loads DE500BA/DE602 virtual adapter with a name "EWA"/"EIA" (note that "/i8255x" syntax must be used only with DE602 and DE602AA adapters); the following line maps it to the host network interface having a name "Charon" ("connection" is a key word). Note that the mapping is performed in 2 steps:

1. A mapping object "packet_port" with a name "EWA0"/"EIA0" is loaded and connected to the host interface named "Charon", so the CHARON VM will use this interface for its networking

2. The loaded DE500BA/DE602 virtual adapter "EWA"/"EIA" is connected to the "packet_port" object "EWA0"/"EIA0".

It is possible to load several DE435, DE450, DE500AA, DE500BA or DE602 controllers. For example (for DE500BA):

```
load DE500BA/dec21x4x EWA interface=EWA0
load packet_port/chnetwrk EWA0 interface="connection:Charon1"
load DE500BA/dec21x4x EWB interface=EWB0
load packet_port/chnetwrk EWB0 interface="connection:Charon2"
```

Some network adapters available in CHARON-AXP are preloaded (for example, AlphaServer DS15 contains 2 preloaded adapters EWA and EWB), so their configuration in CHARON VM is even more simple:

```
load packet_port/chnetwrk EWA0 interface = "connection:Charon"
```

Please note:

- CHARON supports VLAN adapters. If used, proceed with their installation and configuration according to the network adapter vendor User's Guide and then use the resulting VLAN interface the same way as the regular network interface.
- The AlphaServer DS15 and DS25 contain two built-in PCI Ethernet adapters. Models and names (EI* or EW*) of them depend on configuration addon. Choose one of the two or none, but not both. The first instantiates onboard network interfaces as EIA and EWA. While the second - EWA and EWB (enabled by default for backward compatibility)

  Example:

  ```
  #include ds25-onboard-nics.icfg
  include ds25-onboard-nics-ew.icfg
  ```

Follow this link for more details of CHARON-AXP network controllers configuration.

Disk/tape subsystem

The disk and tapes subsystems and the mapping to the system resources can be done using the samples given in the template configuration files.

CHARON-AXP supports KZPBA and KGPSA-CA adapters.

KZPBA PCI SCSI disk/tape controller

Below is the typical configuration options for the KZPBA PCI SCSI disk/tape controller:

```
load KZPBA PKA scsi_id = 7

#set PKA container[0] = "<file-name>.vdisk"

#set PKA container[100]="\\.\PhysicalDrive0"
#set PKA container[101]="\\.\PhysicalDrive(DevID=XXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX)"
#set PKA container[102]="\\.\PhysicalDrive(iScsiTarget = <iSCSI target>, LUN = <LUN number>)"

#set PKA container[200]="\\.\ScsiN:X:Y:Z"

#set PKA container[300]="\\.\CdRom0"
#set PKA container[300]="\\.\CdRom<N>"

#set PKA container[400] = "<file-name>.iso"

#set PKA container[500]="\\.\Tape0"
#set PKA container[500]="\\.\Tape<N>"

#set PKA container[600] = "<file-name>.vtape"

#set PKA container[600]="\\.\A:"
#set PKA media_type[600]="RX23"
```

The first line ("load KZPBA PKA") loads a disk controller KZPBA with name "PKA", followed by 8 groups of lines showing different ways of mapping to the host resources:

- **File representing a physical disk of the Alpha system (disk image)**
  - "<file-name>.vdisk"
    These files can be created from scratch with "MkDisk" utility. Data and OS disks backups are transferred from the original system via tapes or network and restored into these container files.

Mapping may also include the full path, for example: "C:\My disks\my_boot_disk.vdisk". If the path is not specified, the disk images are expected to be in the CHARON VM home directory.

Important note: using compressed folders to store virtual disks and tapes is not supported

- **Physical disk**
  - "\\.\PhysicalDrive<N>"
    ⚠️ Be careful not to destroy all the information from the disk dedicated to CHARON VM by mistake.

    These disks must not be formatted by the host OS.

- **Physical disk by its WWID**
  - "\\.\PhysicalDrive(DevID =XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX)"
    ⚠️ Be careful not to destroy all the information from the disk dedicated to CHARON VM by mistake.

    These disks must not be formatted by the host OS.

    DevID addresses the target physical disk by its WWID (hexadecimal 128-bit identifier assigned to the disk drive by its manufacturer/originator).

    Example:

    ```
    set PKA container[100]="\\.\PhysicalDrive(DevID= 6008-05F3-0005-2950-BF8E-0B86-A0C7-0001)"
    ```

- **iSCSI disks**
  - "\\.\PhysicalDrive(iScsiTarget = <iSCSI target>, LUN = <LUN number>)"

Parameter	Description
iScsiTarget	Addresses the disk by its iSCSI target name.
LUN	Specifies the LUN on the connected iSCSI disk.

Example:

```
set PKA container[200]="\\.\PhysicalDrive(iScsiTarget= iqn.2008-04:iscsi.charon-target-test1, LUN= 1)"
```

- **SCSI device unknown to Windows for direct mapping, for example, a SCSI disk or tape reader**
  - "\\.\Scsi<N>:<X>:<Y>:<Z>"
    The values of N, X, Y and Z can be collected using special utility "Host Device Check" included in the CHARON distributive - or manually by investigation of the devices connected to CHARON host in the "Device Manager" applet.

Parameter	Description
N	A logical number assigned by host operating system (Microsoft Windows) to logical or host's physical storage resource such as physical SCSI HBA
X	An internal SCSI bus number (usually 0) on host's physical SCSI HBA
Y	A SCSI ID of physical SCSI target device attached to host's physical SCSI HBA
Z	A logical unit number inside physical SCSI target device attached to host's physical SCSI HBA

- **CD-ROM device**
  - "\\.\CdRom<N>"

- **ISO file for reading distribution CD-ROM image**
  - "<file-name>.iso"
    Mapping may also include the full path, for example: "C:\My disks\vms_distributive.iso". If the path is not specified, the CD-ROM images are expected to be in the CHARON VM home directory.

- **Host tape device**
  - "\\.\Tape<N>"
- **File representing the tape (tape image)**
  - "<file-name>.vtape"
    These files are created automatically.
    Mapping may also include a full path, for example: "C:\My tapes\backup.vtape". If the path is not specified, the tape images are expected to be in the CHARON VM home directory.

    Important note: using compressed folders to store virtual disks and tapes is not supported.
- **Floppy drive**
  - "\\.\A:"
- **Other type of drive, for example magneto-optical drive**
  - "\\.\<N>:"

Additionally it is possible to specify a parameter "media_type" to assign the type of the attached media explicitly.

Example:

```
set PKA media_type[600]="RX23"
```

The numbers in the square brackets represent the SCSI addresses and LUNs associated with each container of the KZBPA controller. They have the following structure:

[XXYY], where:

Parameter	Range	Description
XX	0...15	Stands for SCSI ID of each connected unit. ℹ Note that the KZPBA itself has some ID associated with it. By default it is 7 but it can be changed in the following way: `load KZPBA PKA scsi_id = 0` In this example an instance "PKA" of KZPBA controller is assigned with SCSI ID 0.
YY	00...07	Stands for LUN.

It is possible to load several KZPBA controllers: DKB, DKC, etc. by configuring specific placement for them on the PCI bus. It is discussed in details in the "Configuration details" chapter of this Guide.

Some Alpha systems emulated by CHARON-AXP have already one or two KZPBA controllers pre-loaded. If the system has only one preloaded controller, the template configuration file usually provides some sample line on how to add another one. For example:

```
load KZPBA PKA bus=pci_1 device=1 function=0 irq_bus=isa irq=24
```

Follow this link for details on the KZPBA controllers configuration.

KGPSA-CA PCI FC disk controller

Optionally it is possible to configure KGPSA-CA FC disk controllers.

They can be configured in 3 modes:

- Direct mapping to the host resources
- Usage of "presentation mode" of connected or external storage controllers
- Pass Through mode

Below is an example of a KGPSA-CA controller loading:

```
load KGPSA FGA
```

Optionally another KGPSA-CA adapter can be loaded in a similar way:

```
load KGPSA FGB
```

Follow this link for details on the KGPSA-CA controllers configuration.

KGPSA-CA mapping to the host resources

Below is the typical configuration options for a KGPSA-CA PCI FC disk controller, mapped to the host resources:

```
load KGPSA FGA

#set FGA container[0] = "<file-name>.vdisk"
#set FGA container[100]="\\.\PhysicalDrive0"
#set FGA container[200]="\\.\PhysicalDrive(DevID=XXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX)"
#set FGA container[300]="\\.\PhysicalDrive(iScsiTarget = <iSCSI target>, LUN = <LUN number>)"
```

The first line ("load KGPSA FGA") loads a disk controller KGPSA named "FGA" followed by 2 groups of lines showing different ways of mapping to the host resources:

- **File representing a physical disk of the Alpha system (disk image)**
  - "<file-name>.vdisk"
    These files can be created from scratch with "MkDisk" utility. Data and OS disks backups are transferred from the original system via tapes or network and restored into these container files.
    Mapping may also include the full path, for example: "C:\My disks\my_boot_disk.vdisk". If the path is not specified, the disk images are expected to be in the CHARON VM home directory.

    Important note: using compressed folders to store virtual disks and tapes is not supported.

- **Physical disk**
  - "\\.\PhysicalDriveN"
    ⚠ Be careful not to destroy all the information from the disk dedicated to CHARON VM by mistake.

    These disks must not be formatted by the host OS.

- **Physical disk by its WWID**
  - "\\.\PhysicalDrive(DevID =XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX)"
    ⚠ Be careful not to destroy all the information from the disk dedicated to CHARON VM by mistake.

    These disks must not be formatted by the host OS.

    DevID addresses the target physical disk by its WWID (hexadecimal 128-bit identifier assigned to the disk drive by its manufacturer/originator).

    Example:

    ```
    set PKA container[100]="\\.\PhysicalDrive(DevID= 6008-05F3-0005-2950-BF8E-0B86-A0C7-0001)"
    ```

- **iSCSI disks**
  - "\\.\PhysicalDrive(iScsiTarget = <iSCSI target>, LUN = <LUN number>)"
    iScsiTarget addresses the disk by its iSCSI target name. LUN specifies LUN on the connected iSCSI disk.

    Example:

    ```
    set PKA container[200]="\\.\PhysicalDrive(iScsiTarget= iqn.2008-04:iscsi.charon-target-test1, LUN= 1)"
    ```

The numbers in the square brackets represent the KGPSA-CA units. They can be in the range of 0 to 32766 but no more than 255 units can be configured on a single controller.

KGPSA-CA mapping to a storage controller using its "presentation" mode

Some storage controllers allows CHARON VM to use their resources using so called "presentation" mode.

In this type of mapping the CHARON VM automatically creates a set of virtual FC devices for each of the units provided by the storage controller and connects to them through its KGPSA-CA FC adapter.

The main benefit in this type of mapping is a flexible way of the virtual disks management depending on the mapped storage controller configuration. For example if an extra disk is added to the storage controller, it automatically appears as a new disk unit on the corresponding KGPSA-CA virtual adapter mapped to that storage controller.

Below is an example of KGPSA-CA PCI FC disk controller mapped to some storage controller (for example SAN) using its "presentation" mode:

```
load KGPSA FGA storage_controller_path_id = 5008-05F3-0005-2950-5008-05F3-0005-2951
```

This line loads an instance of KGPSA-CA controller and maps it to some external controller having ID "5008-05F3-0005-2950-5008-05F3-0005-2951".

Type of mapping	Description
storage_controller_path_id = <Storage controller path ID>	"Storage controller path ID" is a storage (for example SAN) controller path ID. This ID can be obtained from the special utility "Device check tool" (accessible in the "Host Information and Utilities" section of the CHARON Virtual Machine Manager). Once specified, all the disks attached to the storage are automatically mapped as disk units to the CHARON VM.

KGPSA-CA pass through mode

It is also possible to use the emulated KGPSA-CA in "pass through" mode to address a physical EMULEX LightPulse PCI/PCI-X/PCIe FC adapter plugged into the host’s PCI/PCI-X/PCIe slot.

The sample configuration file provides a template for this type of mapping:

```
#set FGA host_bus_location = "PCI bus X, device Y, function Z"
#set FGB host_bus_location = "PCI bus A, device B, function C"
```

The "host_bus_location" parameter addresses the host EMULEX LightPulse PCI/PCI-X/PCIe FC adapter in the following way:

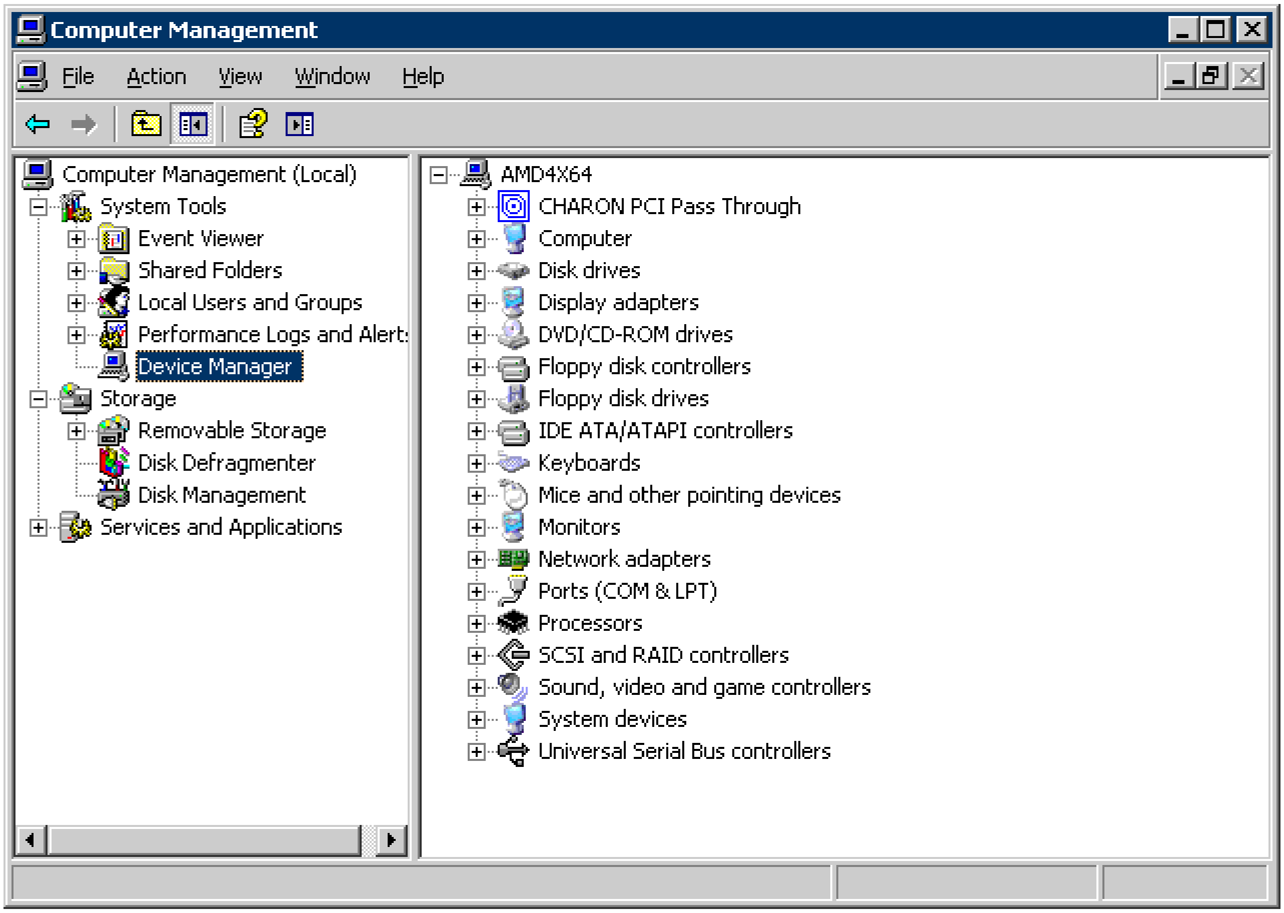
Parameters	Description
"PCI bus X"	PCI bus number of the host EMULEX LightPulse PCI/PCI-X/PCIe FC adapter
"device Y"	PCI bus device number of the host EMULEX LightPulse PCI/PCI-X/PCIe FC adapter
"function Z"	The "function" parameter of the the host EMULEX LightPulse PCI/PCI-X/PCIe FC adapter

To establish "pass through" mode do the following:

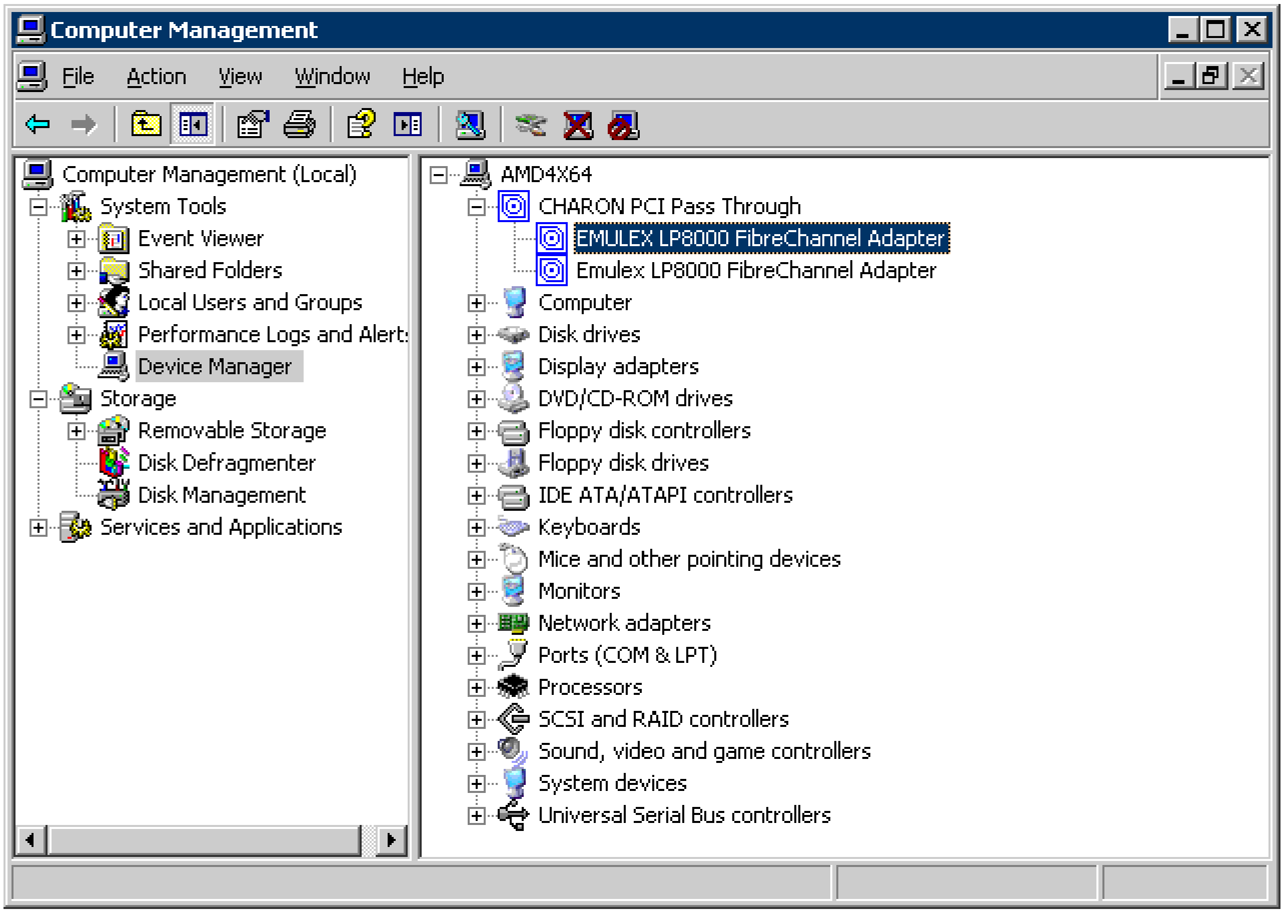
1. Install the EMULEX LightPulse PCI/PCI-X/PCIe FC adapter (see below for a list of supported models) to some spare PCI/PCI-X/PCIe slot of the host system
2. Boot a Windows operating system
3. Install the EMULEX LightPulse PCI/PCI-X/PCIe FC adapter driver from the following directory "C:\Program Files\CHARON\Drivers\EMULEX_X.X.0.XXXXX" by choosing the "Install from a list or specific location (Advanced)" option and then selecting the "emulex_lp_ppt_amd64.inf" file.
4. Reboot the host

Now it is possible to collect the parameters for CHARON VM mapping to the EMULEX LightPulse PCI/PCI-X/PCIe FC adapter.

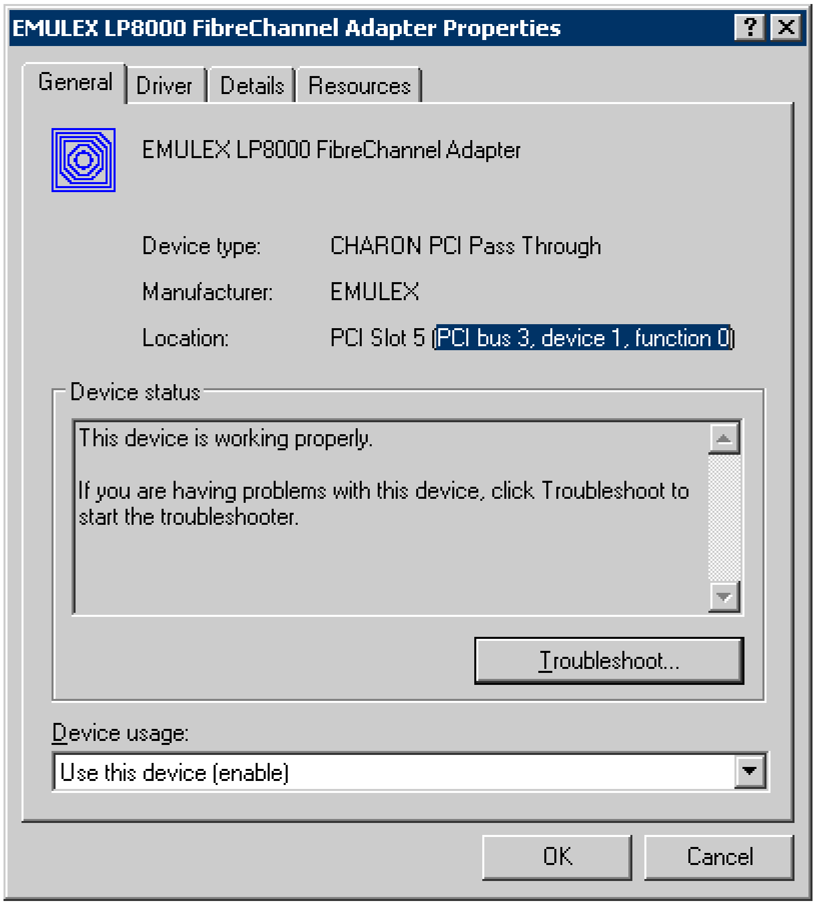
Open the “Computer Management” application and select “Device Manager”:



On the right panel select the desired physical EMULEX LP FibreChannel adapter under "CHARON PCI Pass Through":



Open the properties sheet by double-clicking on the selected adapter:



EMULEX LP8000 FibreChannel Adapter

Device type: CHARON PCI Pass Through

Manufacturer: EMULEX

Location: PCI Slot 5 (PCI bus 3, device 1, function 0)

Device status

This device is working properly.

If you are having problems with this device, click Troubleshoot to start the troubleshooter.

Device usage:

Use this device (enable)

The “Location:” on the above picture gives X, Y, and Z for the "host_bus_location" parameter. For example:

```
set FGA host_bus_location = "PCI bus 3, device 1, function 0"
```

⚠ Non-US-EN installations of Windows may present “Location:” string in local language, but "host_bus_location" parameter requires English notation, so the words “PCI”, “bus”, “device”, and “function” must be specified in English.

The following is the list of EMULEX LightPulse PCI/PCI-X/PCIe FC adapters supported by the CHARON-AXP PCI Pass Through driver and suitable for the emulation of a KGPSA-CA PCI FC adapter in CHARON PCI Pass Through mode:

Supported	Not Supported	Not tested
LP8000 LP9000 LP9002 LP9802 LP10000 LP10000DC LP10000-S LPX1000 LP11002 LPe11002 (FC2242SR, A8003A) LPe1105	LPe1150 (FC2142SR, A8002A)	LPe11000

FDDI support via DEFPA PCI FDDI controller in "pass through" mode

Optionally it is possible to configure a DEFPA PCI FDDI controller in "pass through" mode, mapped to a physical DEFPA FDDI adapter installed on the host:

```
load defpa FDDI host_bus_location = "PCI bus X, device Y, function Z"
set FDDI bus=pci_1 device=1 function=0 irq=24 irq_bus=isa
```

Pay attention to the proper placement of the emulated DEFPA adapter on the virtual Alpha PCI bus (it is controlled by "bus", "device", "function", "irq" and "irq_bus" parameters). Refer to this chapter of this Guide for more information.

The "host_bus_location" parameter addresses the host DEFPA FDDI adapter in the following way:

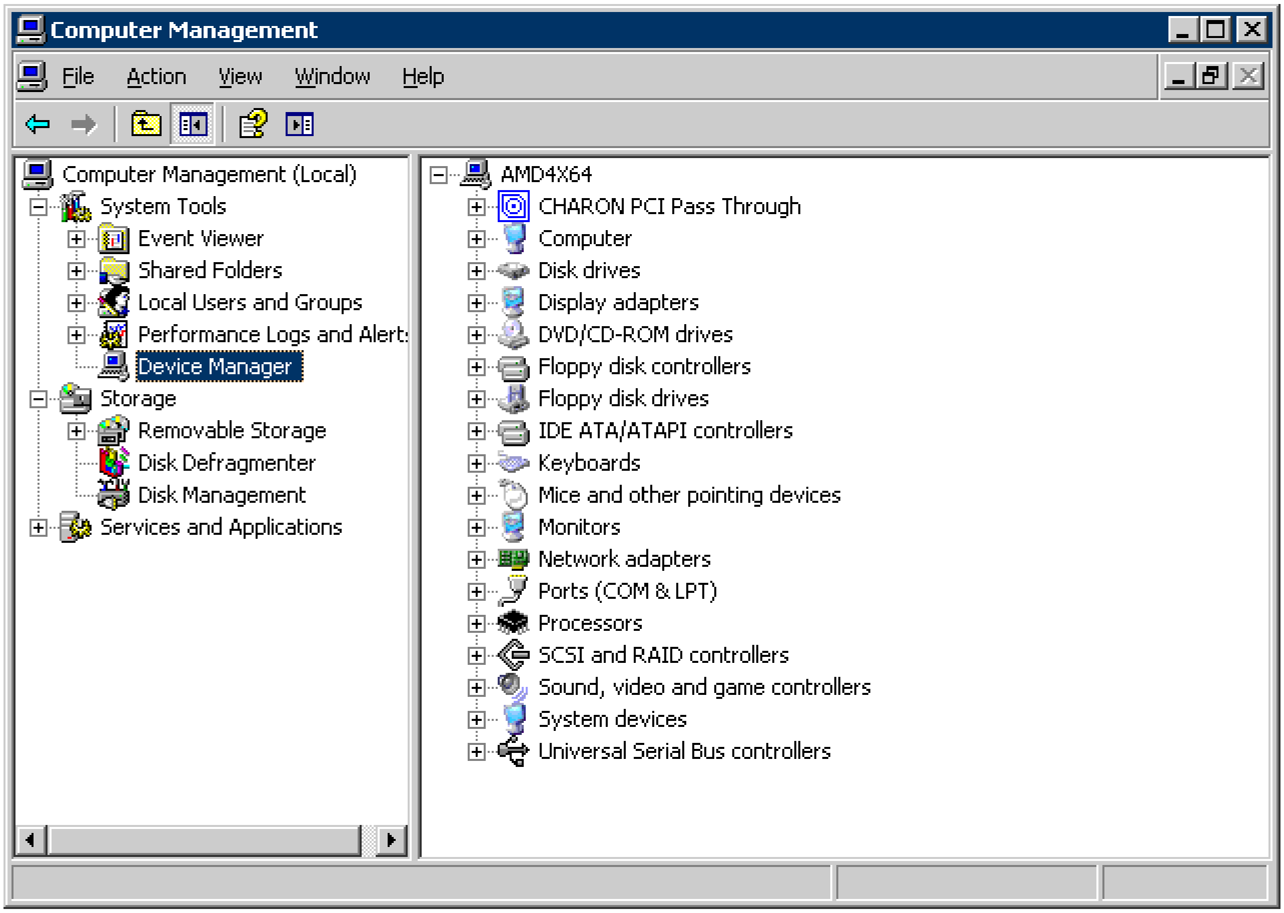
Parameters	Description
"PCI bus X"	PCI bus number of the host DEFPA FDDI adapter
"device Y"	PCI bus device number of the host DEFPA FDDI adapter
"function Z"	The "function" parameter of the the host DEFPA FDDI adapter

To establish the "pass through" mode do the following:

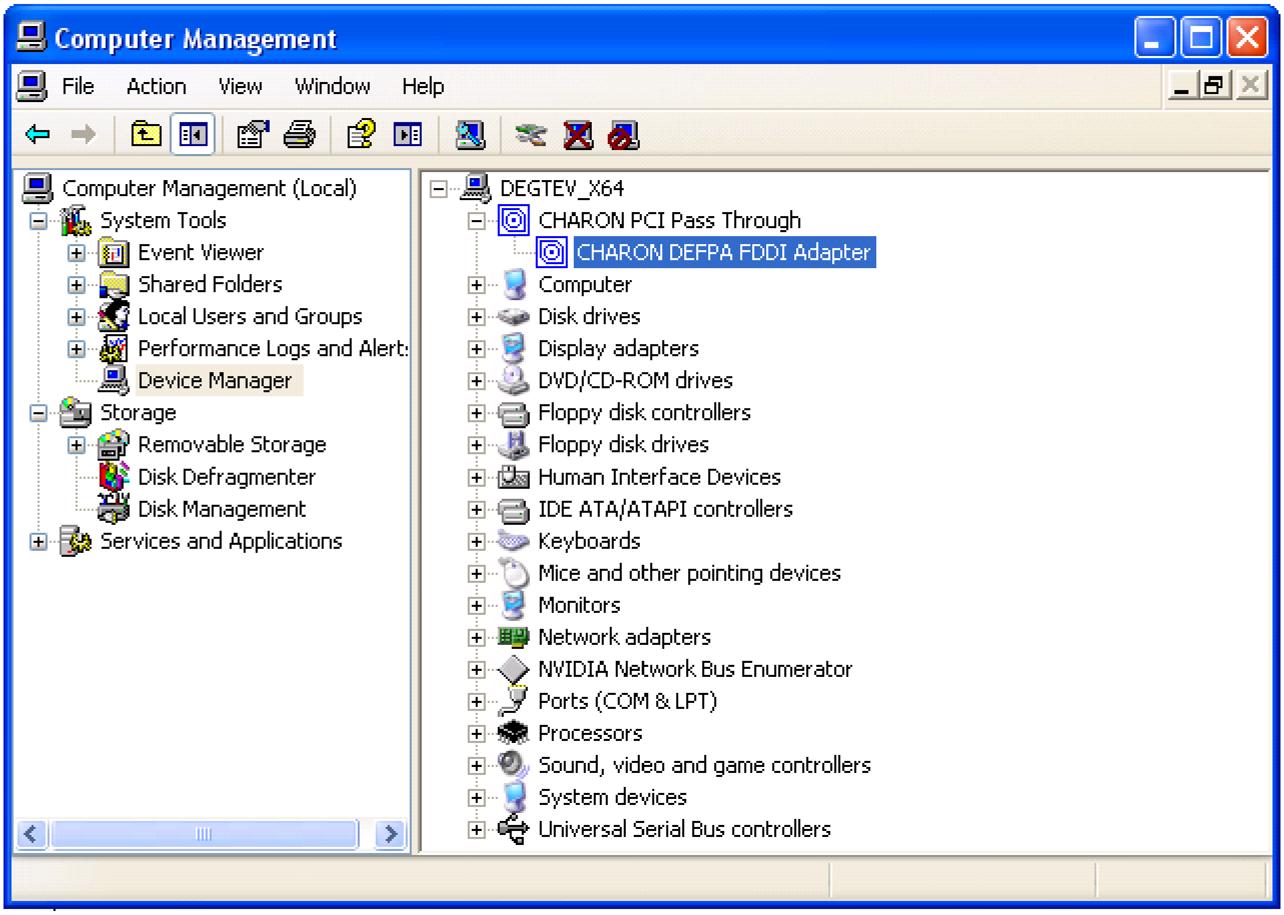
1. Install the DEFPA FDDI adapter to some spare PCI slot of the host system.
   ⚠ Note that PCIe and PCI-X are not supported by the DEFPA FDDI adapter.
2. Boot a Windows operating system
3. Install the DEFPA FDDI adapter driver from the following directory: "C:\Program Files\CHARON\Drivers\DEFPA_X.X.X.XXXXX" by choosing the "Install from a list or specific location (Advanced)" option and then selecting the "defpa_ppt_amd64.inf" file.
4. Reboot the host

Now it is possible to collect the parameters for CHARON VM mapping to the DEFPA FDDI host adapter.

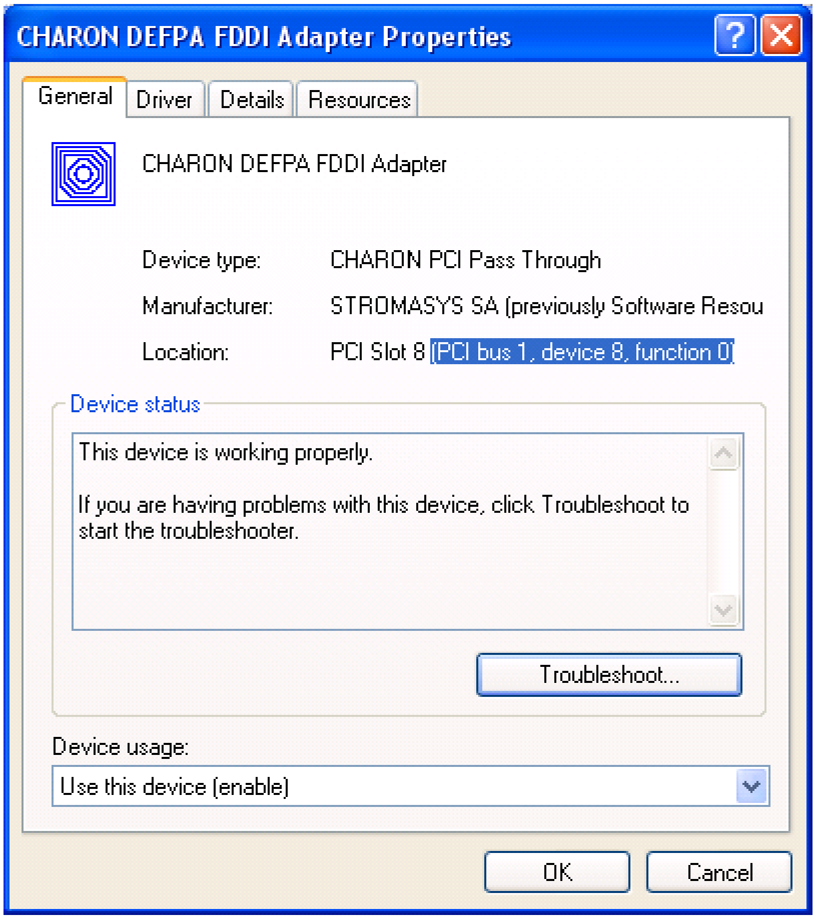
Open the “Computer Management” application and select “Device Manager”:



On the right panel select the installed DEFPA adapter:



Open the properties sheet by double-clicking on the selected adapter:



CHARON DEFPA FDDI Adapter Properties

CHARON DEFPA FDDI Adapter

Device type: CHARON PCI Pass Through

Manufacturer: STROMASYS SA (previously Software Resou

Location: PCI Slot 8 (PCI bus 1, device 8, function 0)

Device status

This device is working properly.

If you are having problems with this device, click Troubleshoot to start the troubleshooter.

Device usage: Use this device (enable)

The “Location:” on the above picture gives X, Y, and Z for the "host_bus_location" parameter. For example:

```
set FDDI host_bus_location = "PCI bus 1, device 9, function 0"
```

⚠ Non-US-EN installations of Windows may present “Location:” string in local language, but "host_bus_location" parameter requires English notation, so the words “PCI”, “bus”, “device”, and “function” must be specified in English.

Serial lines support via emulated PBXDA-xx family PCI controllers

Optionally it is possible to configure the following models of PBXDA-xx family controllers.

VIrtual PBXDA-xx

Syntax for loading PBXDA-xx family serial lines adapters:

```
#load PBXDA/DIGI <name>
#load PBXDA_BA/DIGI <name>
#load PBXDA_BB/DIGI <name>
#load PBXDA_AC/DIGI <name>
```

Example:

```
load PBXDA_AC/DIGI TXA
```

The adapter instance name ("TXA" in the example above) is used then for parametrization, for example:

```
set TXA line[2]="COM1:"
set TXA port[3]=1706
```

The numbers in the square brackets represent line number on the virtual PBXDA adapter starting from 0.

Controller type	Maximum number of lines
PBXDA	2
PBXDA_BA	4
PBXDA_BB	8
PBXDA_AC	16

It is possible to specify either physical port ("COM<n>") or an IP port for connecting the virtual PBXDA-xx to system resources. No other parameters are applicable in this type of emulation.

PBXDA-xx support in "pass through" mode

In "pass through" mode PBXDA-xx family controllers are mapped to specific models of the physical DIGI serial lines adapters installed on the CHARON VM host:

DEC PBXDA-xx adapter	Name of the device to map to	Controller	Vendor ID	Device ID
PBXDA-BA	DIGI AccelePort 4r 920	ASIC PCI	114Fh	0026h
PBXDA-BB	DIGI AccelePort 8r 920	ASIC PCI	114Fh	0027h
PBXDA-AC	DIGI AccelePort Xem	ASIC PCI	114Fh	0004h
PBXDA-AC	DIGI AccelePort Xem	ASIC PCI	114Fh	0008h

Below is an example of mapping to a physical DIGI adapter installed on the host:

```
load digi PBXDA host_bus_location=“PCI bus 3,device 1,function 0”
set PBXDA bus=pci_1 device=1 function=0 irq=24 irq_bus=isa
```

Pay attention to the proper placement of the emulated PBXDA-xx adapter on the virtual Alpha PCI bus (it is controlled by "bus", "device", "function", "irq" and "irq_bus" parameters). Refer to this chapter of this Guide for more information.

The "host_bus_location" parameter addresses the host DIGI adapter in the following way:

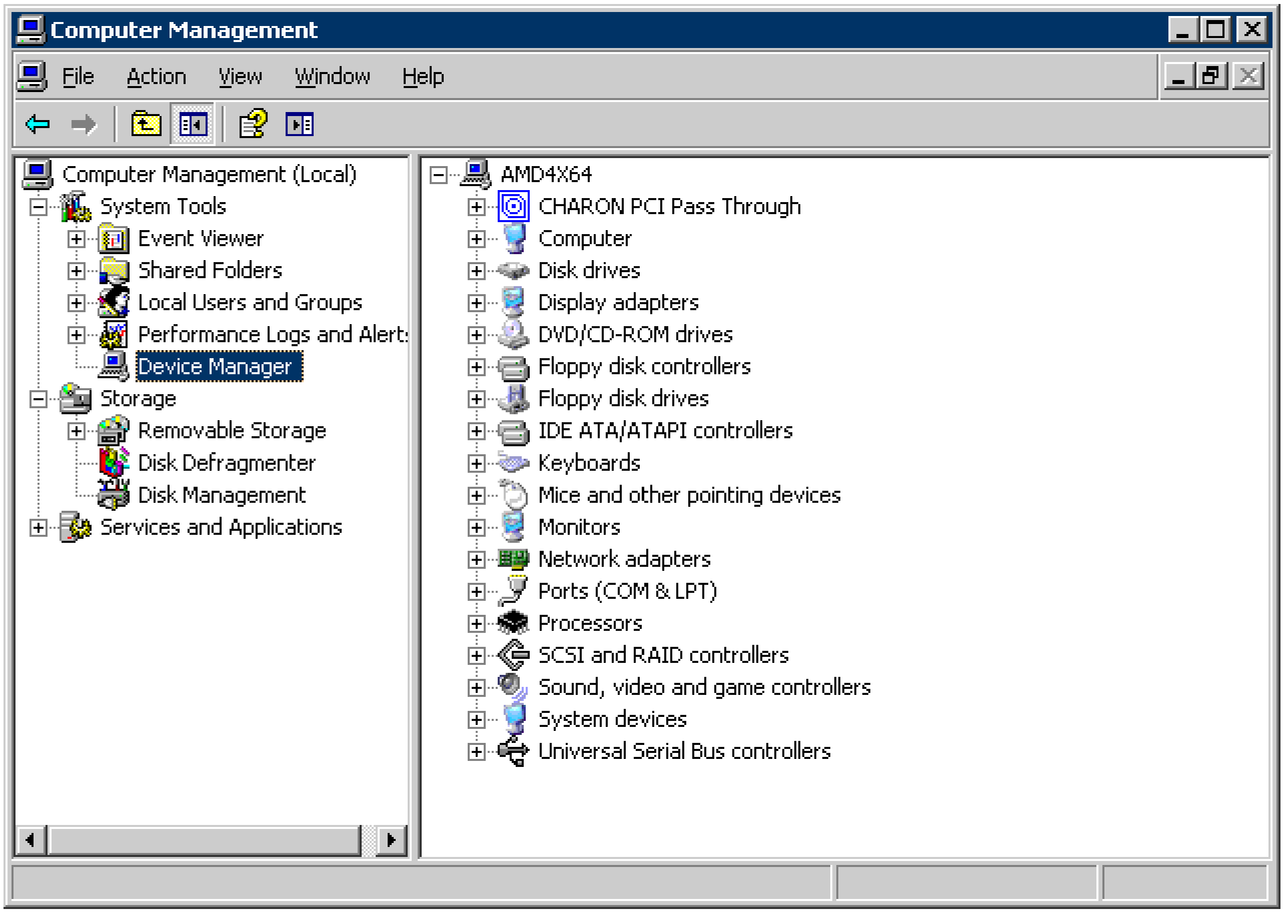
Parameters	Description
"bus X"	PCI bus number of the host DIGI adapter
"device Y"	PCI bus device number of the host DIGI adapter
"function Z"	The "function" parameter of the the host DIGI adapter

To establish the "pass through" mode do the following:

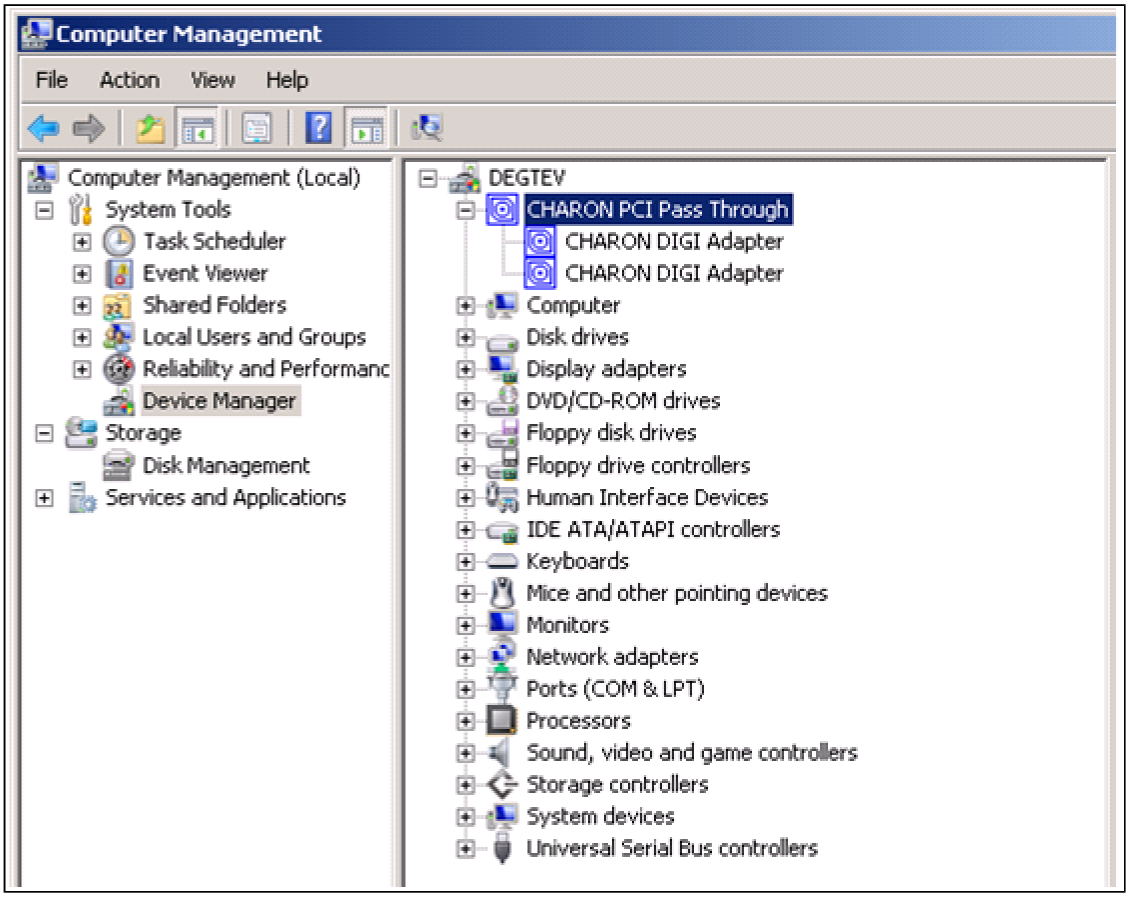
1. Install the DIGI adapter of the required type to some spare PCI/PCI-X/PCIe slot of the host system.
2. Boot a Windows operating system
3. Install the DIGI adapter driver from the following directory: "C:\Program Files\CHARON\Drivers\DIGI_X.X.X.XXXXX" by choosing "Install from a list or specific location (Advanced)" option and then selecting the "digi_ppt_amd64.inf" file.
4. Reboot the host

Now it is possible to collect the parameters for CHARON VM mapping to the DIGI host adapter.

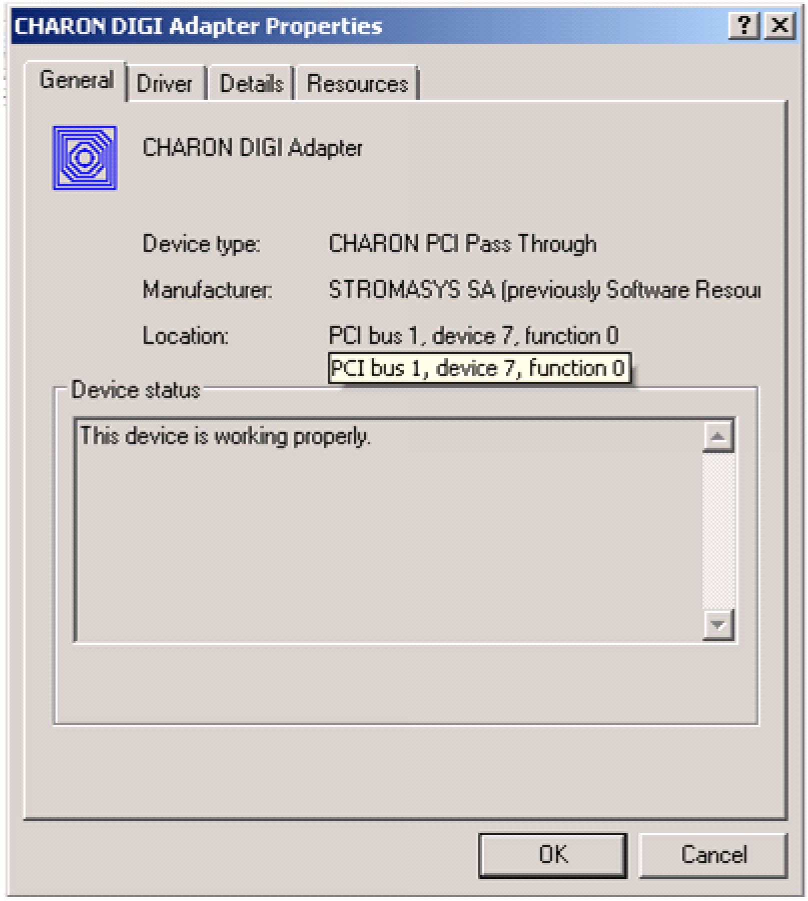
Open the “Computer Management” application and select “Device Manager”:



On the right panel select proper physical DIGI adapter:



Open te properties sheet by double-clicking on the selected adapter:



CHARON DIGI Adapter

Device type: CHARON PCI Pass Through

Manufacturer: STROMASYS SA (previously Software Resou

Location: PCI bus 1, device 7, function 0

PCI bus 1, device 7, function 0

Device status

This device is working properly.

The “Location:” on the above picture gives X, Y, and Z for the "host_bus_location" parameter. For example:

```
set PBXDA host_bus_location = "PCI bus 1, device 7, function 0"
```

⚠ Non-US-EN installations of Windows may present “Location:” string in local language, but "host_bus_location" parameter requires English notation, so the words “PCI”, “bus”, “device”, and “function” must be specified in English.

Auto boot

Charon-AXP can be configured to automatically boot an operating system immediately after it has started by specifying the default boot device and setting the 'auto_action' parameter to 'restart' from the console.

Example: dka0 is defined as the default boot device

```
P00>>>set bootdef_dev dka0
P00>>>set auto_action restart
```

Network boot

Charon-AXP can be configured to boot on network if the legacy operating system allows it via MOP. MOPv3 (DECnet) and MOPv4 (LANCP) are supported.

Usage/example:

```
P00>>>boot [-flags ...] [-file ...] ewa0
```

Both EW and EI adapters are supported for MOP network boot.

Migration OpenVMS to CHARON-AXP for Windows

Table of Contents

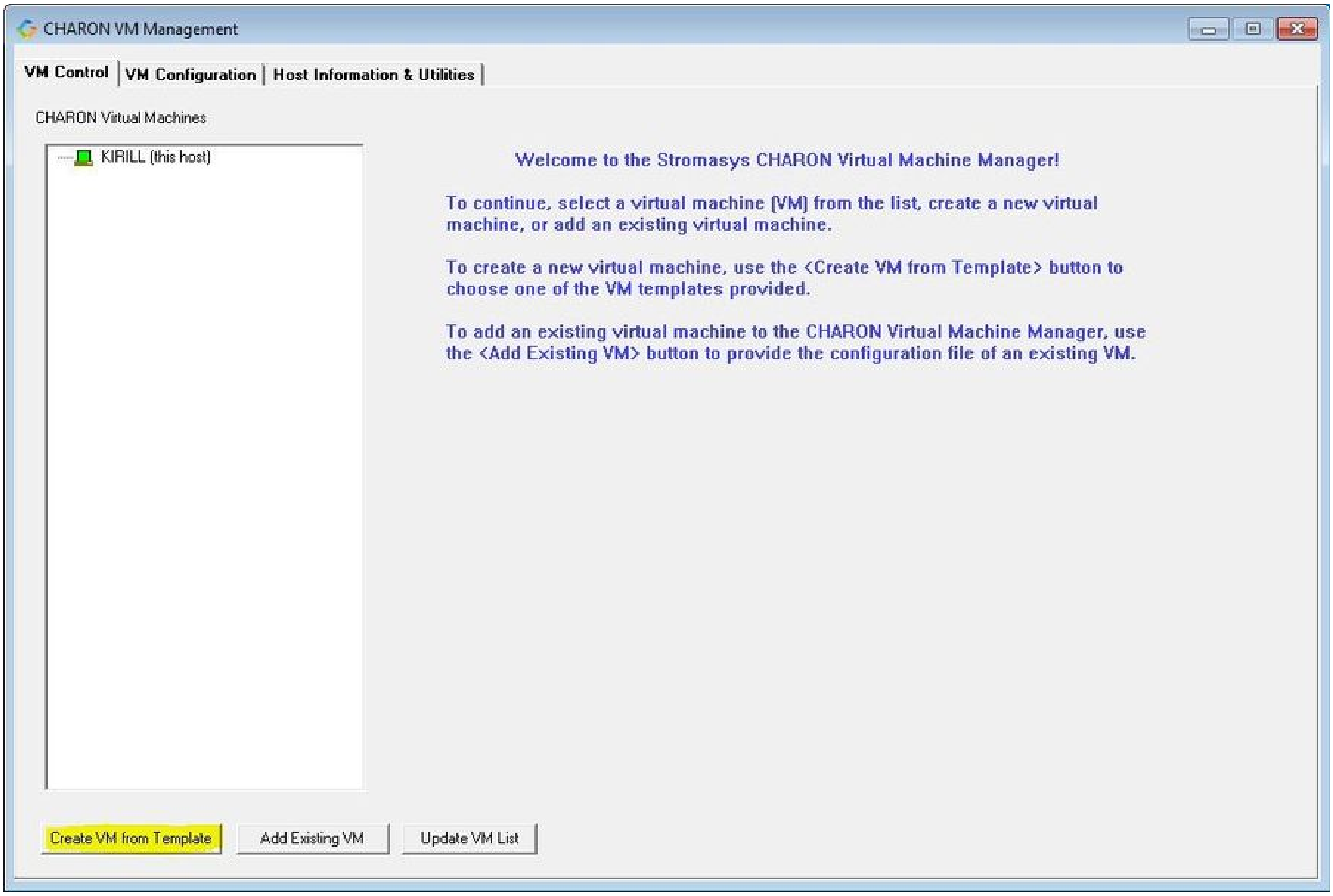
Introduction

This section describes how to migrate your Alpha system to CHARON-AXP. We will use a sample AlphaServer ES40 system running OpenVMS to demonstrate the migration procedure. The process is similar for all CHARON-AXP models.

Creating CHARON Virtual Machine

As a first step it is required to create a CHARON Virtual Machine (VM) using the CHARON Virtual Machine Manager in the following way:

Press the "Create VM from Template" button:



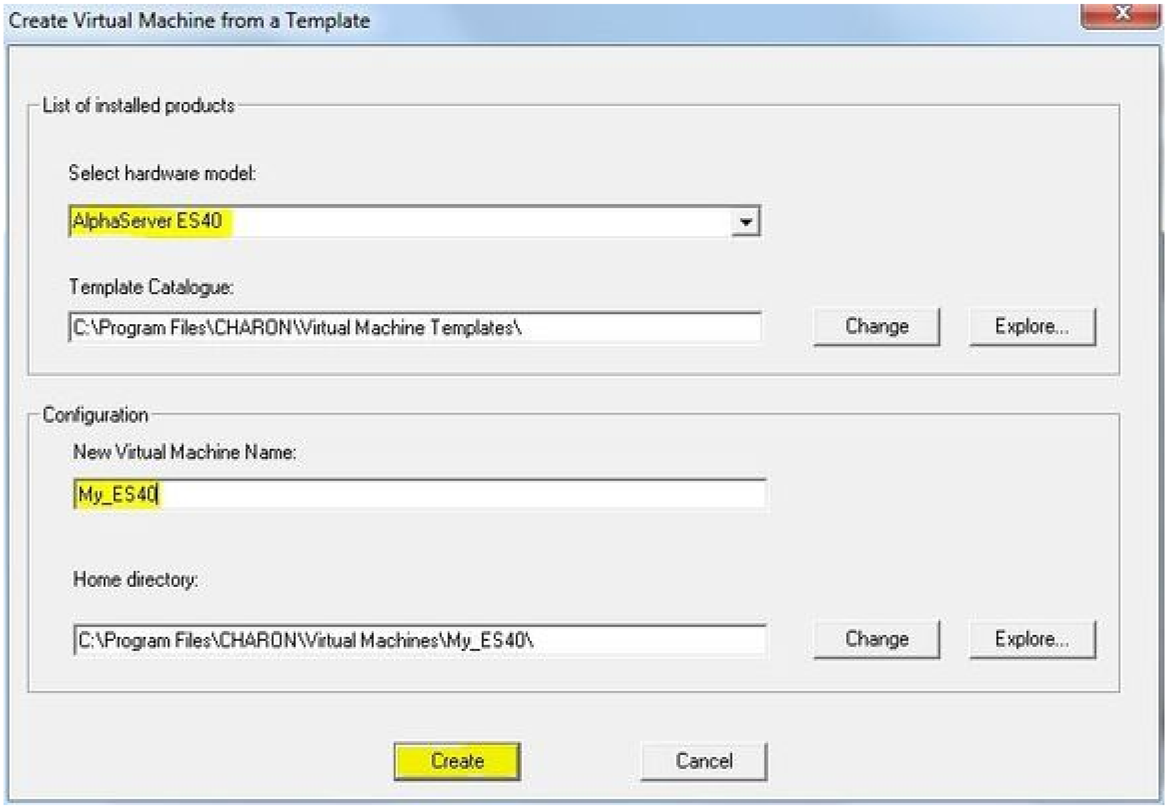
Welcome to the Stromasys CHARON Virtual Machine Manager!

To continue, select a virtual machine (VM) from the list, create a new virtual machine, or add an existing virtual machine.

To create a new virtual machine, use the <Create VM from Template> button to choose one of the VM templates provided.

To add an existing virtual machine to the CHARON Virtual Machine Manager, use the <Add Existing VM> button to provide the configuration file of an existing VM.

Specify the Alpha model to be emulated and the name of the VM then press the "Create" button:


Create Virtual Machine from a Template
List of installed products
Configuration
Change
Explore...
Change
Explore...
Create
Cancel

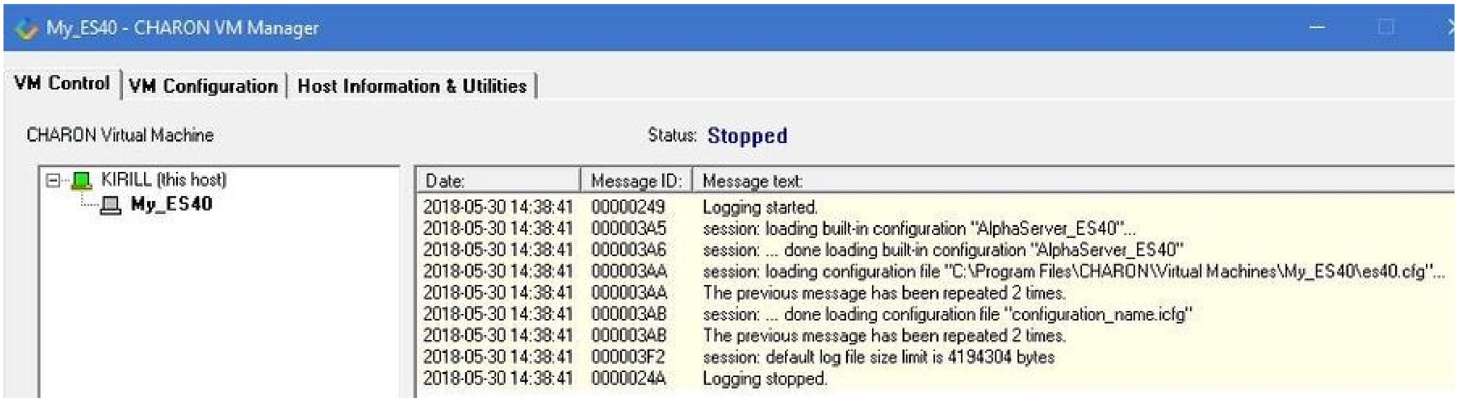
Select hardware model: AlphaServer ES40

Template Catalogue: C:\Program Files\CHARON\Virtual Machine Templates\

New Virtual Machine Name: My_ES40

Home directory: C:\Program Files\CHARON\Virtual Machines\My_ES40\

The VM will be created:



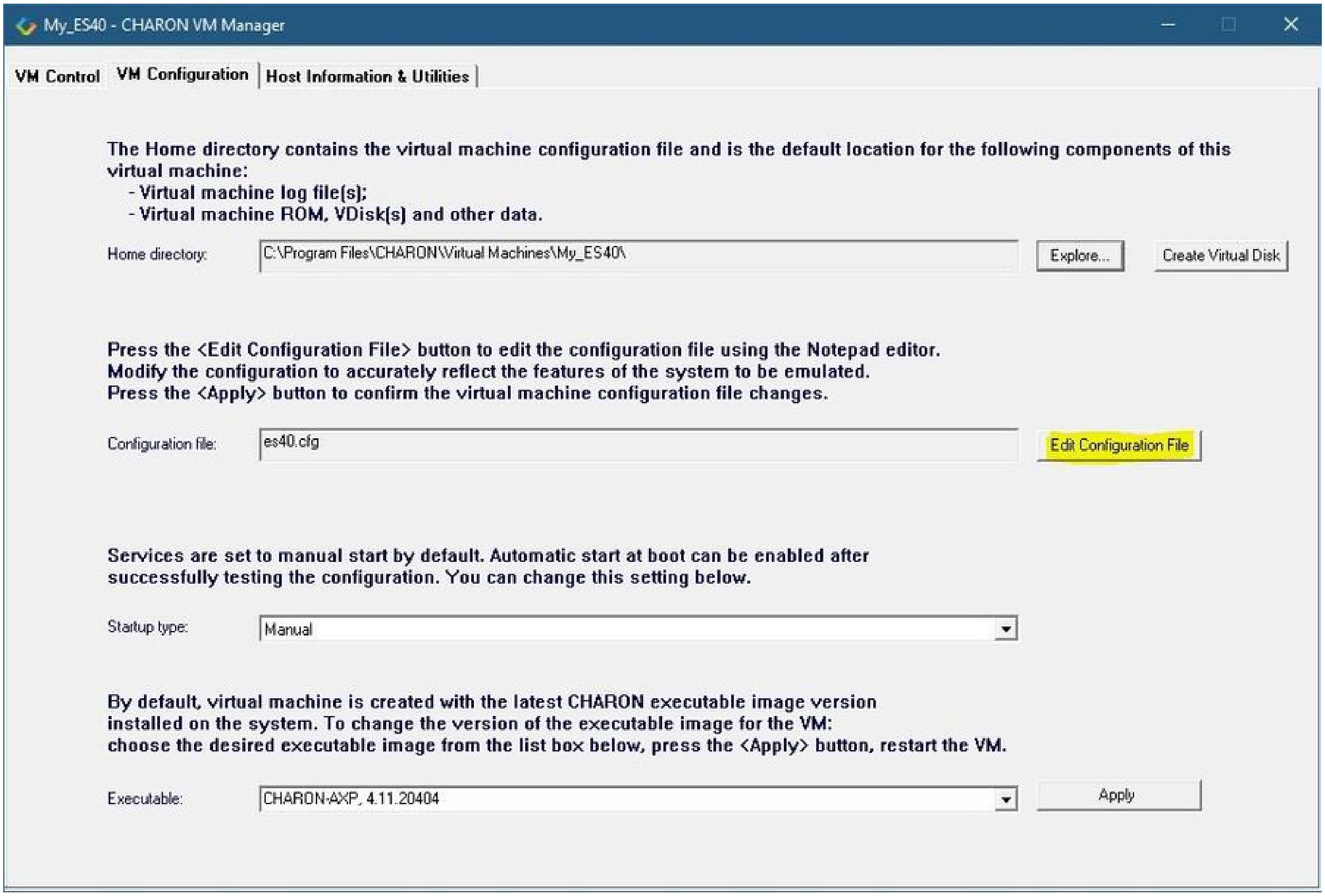
CHARON Virtual Machine

Status: Stopped

```
Date:               Message ID:  Message text:
2018-05-30 14:38:41 00000249     Logging started.
2018-05-30 14:38:41 000003A5     session: loading built-in configuration "AlphaServer_ES40"...
2018-05-30 14:38:41 000003A6     session: ... done loading built-in configuration "AlphaServer_ES40"
2018-05-30 14:38:41 000003AA     session: loading configuration file "C:\Program Files\CHARON\Virtual Machines\My_ES40\es40.cfg"...
2018-05-30 14:38:41 000003AA     The previous message has been repeated 2 times.
2018-05-30 14:38:41 000003AB     session: ... done loading configuration file "configuration_name.icfg"
2018-05-30 14:38:41 000003AB     The previous message has been repeated 2 times.
2018-05-30 14:38:41 000003F2     session: default log file size limit is 4194304 bytes
2018-05-30 14:38:41 0000024A     Logging stopped.
```

Select the VM from the left pane and switch to the "VM Configuration" tab.

In order to reproduce the target Alpha configuration, press the "Edit Configuration file" button:



The Home directory contains the virtual machine configuration file and is the default location for the following components of this virtual machine:
- Virtual machine log file(s);
- Virtual machine ROM, VDisk(s) and other data.

Home directory: C:\Program Files\CHARON\Virtual Machines\My_ES40\

Press the <Edit Configuration File> button to edit the configuration file using the Notepad editor.
Modify the configuration to accurately reflect the features of the system to be emulated.
Press the <Apply> button to confirm the virtual machine configuration file changes.

Configuration file: es40.cfg

Services are set to manual start by default. Automatic start at boot can be enabled after successfully testing the configuration. You can change this setting below.

Startup type: Manual

By default, virtual machine is created with the latest CHARON executable image version installed on the system. To change the version of the executable image for the VM: choose the desired executable image from the list box below, press the <Apply> button, restart the VM.

Executable: CHARON-AXP, 4.11.20404

ℹ Alternatively it is possible to select the target CHARON VM and press the "Edit Configuration" button.

Collecting information about the source OpenVMS system

The purpose of this step is to collect the exact configuration of the Alpha hardware in order to create the CHARON VM configuration file.

Turn on the source Alpha system. At the ">>>" prompt, issue the "`show device`" command:

```
>>>show device

sys0.0.0.0.0 SYS0 System ROOT Device
ewa0.0.0.1.1 EWA0 F8-D1-11-00-67-E6
pka0.0.0.2.1 PKA0 Q-Logic/ISP PCI SCSI HBA
pga0.0.0.3.1 PGA0 WWN 1000-0000-0248-C550
pqa0.0.0.15.0 PQA0 ALi 1553C Integrated IDE Controller
pqb0.0.1.15.0 PQB0 ALi 1553C Integrated IDE Controller
dqa0.0.0.15.0 DQA0 TSSTcorpCDDVDW SH-222BB
dka0.0.0.2.1 DKA0 DEC RZ28 (C)DEC
dka100.1.0.2.1 DKA100 DEC RZ22 (C)DEC
dka200.2.0.2.1 DKA200 DEC RZ23 (C)DEC
mka600.6.0.2.1 MKA600 Virtual SCSI Tape

>>>
```

To get more detailed information, boot OpenVMS and issue the "show device /full" command:

```
$ show device /full

Disk PFCAXP$DKA0:, device type RZ28, is online, mounted, file-oriented device,
shareable, available to cluster, error logging is enabled.
...

Disk PFCAXP$DKA100:, device type RZ22, is online, file-oriented device,
shareable, available to cluster, error logging is enabled.
...

Disk PFCAXP$DKA200:, device type RZ23, is online, file-oriented device,
shareable, available to cluster, error logging is enabled.
...

Disk PFCAXP$DQA0:, device type TSSTcorpCDDVDW SH-222BB, is online, file-oriented
device, shareable, available to cluster, error logging is enabled.
...

Disk $1$DGA0: (PFCAXP), device type RZ24, is online, file-oriented device,
shareable, available to cluster, error logging is enabled.
...

Magtape PFCAXP$MKA600:, device type Virtual SCSI Tape, is online, file-oriented
device, available to cluster, error logging is enabled, device supports
fastskip (per_io).
...

Terminal OPA0:, device type VT102, is online, record-oriented device, carriage
control.
...

Device EWA0:, device type DE500, is online, network device, device is a template
only.
...

Device FGA0:, device type KGPSA Fibre Channel, is online, shareable, error
logging is enabled.
...

Device PGA0:, device type SCSI FCP, is online, error logging is enabled.

...

Device PKA0:, device type Qlogic ISP1020 SCSI port, is online, error logging is
enabled.
...

Device $1$GGA32767:, device type Generic SCSI device, is online, shareable.

$
```

If Tru64 UNIX V5 is running on the host system, it is recommended to use the following commands to get information on the host configuration:

Command	Description
`#/sbin/hwmgr view devices`	Get detailed information about the host hardware configuration
`#/sbin/hwmgr show scsi`	Get specific information about the host SCSI controllers and attached disks
`#/sbin/hwmgr view hierarchy`	Get information about the host controllers

Please refer to the Tru64 UNIX User's Guide for more details.

The source Alpha peripheral configuration in this example is:

Controller	Devices on controller	Description
KZPBA	-DKA0 (RZ28) -DKA100 (RZ22) -DKA200 (RZ23) -MKA600 (tape)	SCSI disk/tape controller
KGPSA-CA	-DGA0 (RZ24)	FC disk controller
OPA0		System console
TSSTcorpCDDVDW SH-222BB	-DQA0	IDE CD-ROM controller
EWA0		Network interface, MAC address: F8-D1-11-00-67-E6

Now collect some general information about the AlphaServer ES40 system:

```
>>>show cpu /full

System: PFCAXP, AlphaServer ES40 6/667

SMP execlet = 3 : Enabled : Streamlined.
Config tree = None
Primary CPU = 0
HWRPB CPUs = 4
Page Size = 8192
Revision Code =
Serial Number = SN01234567
Default CPU Capabilities:
System: QUORUM RUN
Default Process Capabilities:
System: QUORUM RUN

....

>>>
```

```
>>>show mem

System Memory Resources on 5-FEB-2018 09:29:16.42

Physical Memory Usage (pages): Total Free In Use Modified
Main Memory (512.00MB) 65536 56496 8610 430

...

>>>
```

So the collected information about the AlphaServer ES40 system is:

Component	Value
System Type	AlphaServer ES40 6/667
Serial Number	SN01234567
Number of CPUs	4
System memory	512 Mb

In some particular situations it is also important to know the exact placement of all the peripheral devices on the Alpha PCI bus. To do so, issue the "show config" command at the SRM prompt (>>>) on the Alpha console. For example:

```
>>>show config

...

PCI Bus

Bus 00 Slot 03: DECchip 21142 Network Controller
ewa0.0.0.3.0 00-00-F8-03-9A-6D

Bus 00 Slot 07: Cypress PCI Peripheral Controller

Bus 00 Slot 07: Function 1: PCI IDE

Bus 00 Slot 07: Function 2: PCI IDE

Bus 00 Slot 07: PCI USB

Bus 00 Slot 08: DECchip 21052 PCI to PCI Bridge

Bus 01 Slot 08: ISP1040 Scsi Controller
pka0.7.0.1008.0 SCSI Bus ID 7
dka0.0.0.1008.0 RZ2DD-KS
dka400.4.0.1008.0 RRD45

>>>
```

The "show config" command collects the following information of placement of peripheral devices on the PCI bus:

- Bus number
- Slot number
- Function number

To find out the exact types of controllers and other useful information refer to the source Alpha system documentation.

Creation of the CHARON-AXP configuration file

Using the above information, the following parameters and values can be set in the configuration file:

```
#
# AlphaServer model: AlphaServer ES40 6/667
#

set session hw_model = AlphaServer_ES40
set ace cpu_architecture = EV67
set rom dsrdb[0] = 1820 system_name = "AlphaServer ES40 6/667"

...

#
# Override default System Serial Number, set it to "SN01234567"
#

set rom system_serial_number = SN01234567

#
# Specify RAM size: 512 Mb
#

set ram size=512

#
# Map OPA0 console to "Putty" terminal emulator (included in CHARON-AXP kit)
#

set COM1 alias = OPA0 port = 10003 application = "putty -load OPA0 -P 10003"

#
# Ð¡onnect the emulator's DQA0 to the host's ATAPI CD/DVD-ROM drive.
#

set ide container="\\.\CdRom0"

#
# Load optional DE500BA PCI Ethernet Adapter (EWA0) and map it to the "Charon" host network interface
#

load DE500BA/dec21x4x EWA interface=EWA0
load packet_port/chnetwrk EWA0 interface="connection:Charon"

#
# Load DEC-KZPBA SCSI controller and map it to 3 disk containers and 1 tape container
#

load KZPBA PKA scsi_id = 7

set PKA container[0] = "C:\My disks\bootable.vdisk"
set PKA container[100] = "C:\My disks\RZ22.vdisk"
set PKA container[200] = "C:\My disks\RZ23.vdisk"

set PKA container[600] = "C:\My tapes\my_tape.vtape"

#
# Load DEC-KGPSA-CA PCI FC adapter and map it to a disk container
#

load KGPSA FGA

set FGA container[0] = "C:\My disks\RZ24.vdisk"

...
```

Making disk images

In our example, the mapping of the KZPBA SCSI controller include disk and tape images. The tape images have not to be manually created whereas the disk images have to be created as described below.

Our example creates disk images of the original physical type. In reality, this step is the best opportunity in the migration to provision larger disks to get extra storage space.

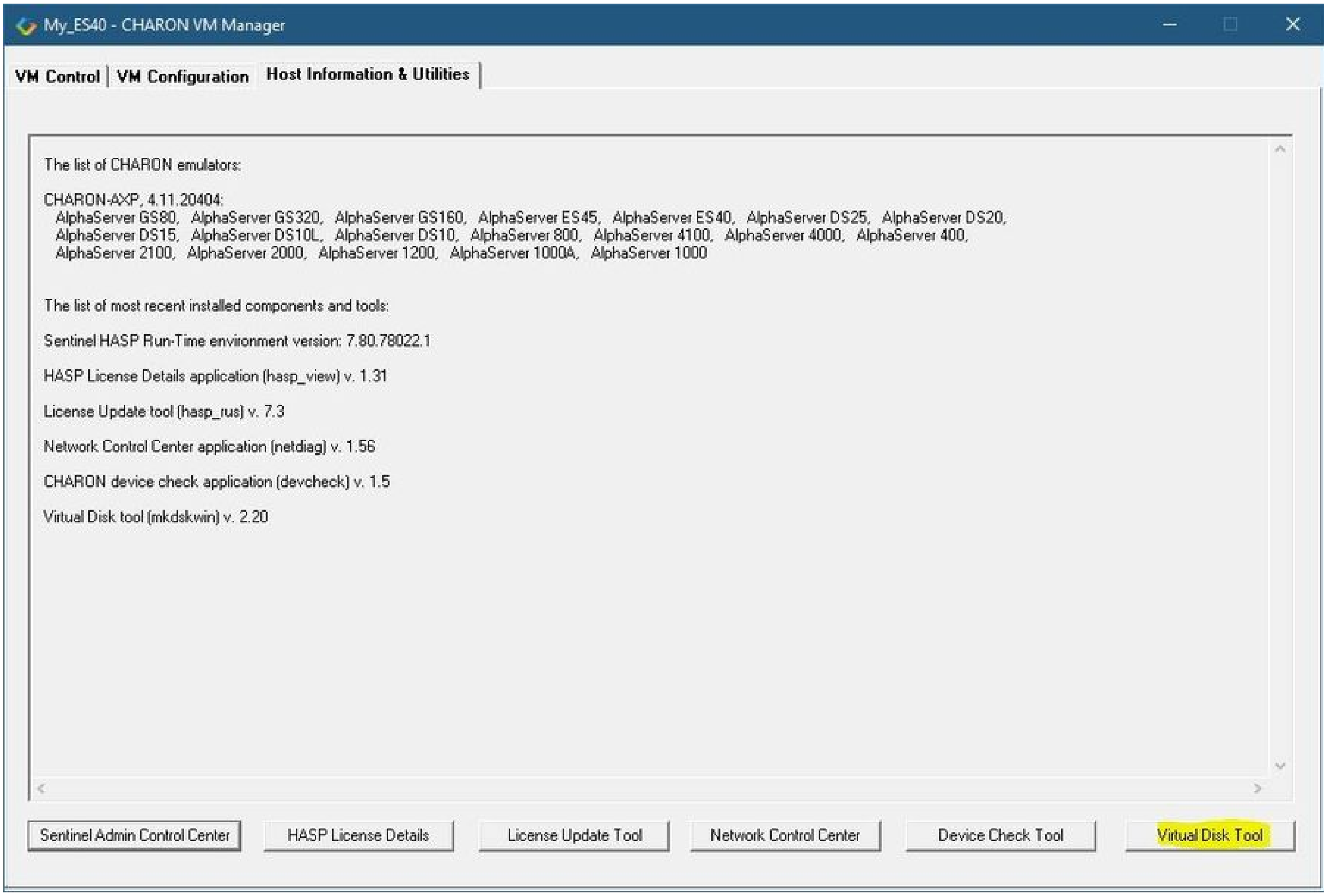
Create special directories for storing the disk and tape images. The created directories are referenced in the sample configuration file above.

```
...> cd C:\
C:\> mkdir "My disks"
C:\> mkdir "My tapes"
```

This operation can also be done using Windows Explorer.

Alternatively it is possible to put the tape and disk images in the CHARON VM Home Directory.

Start the "MkDisk" utility by pressing the "Virtual Disk Tool" button in the "Host Information & Utilities" tab of the CHARON Virtual Machine Manager:



The list of CHARON emulators:

CHARON-AXP, 4.11.20404:
  AlphaServer GS80, AlphaServer GS320, AlphaServer GS160, AlphaServer ES45, AlphaServer ES40, AlphaServer DS25, AlphaServer DS20,
  AlphaServer DS15, AlphaServer DS10L, AlphaServer DS10, AlphaServer 800, AlphaServer 4100, AlphaServer 4000, AlphaServer 400,
  AlphaServer 2100, AlphaServer 2000, AlphaServer 1200, AlphaServer 1000A, AlphaServer 1000

The list of most recent installed components and tools:

Sentinel HASP Run-Time environment version: 7.80.78022.1

HASP License Details application (hasp_view) v. 1.31

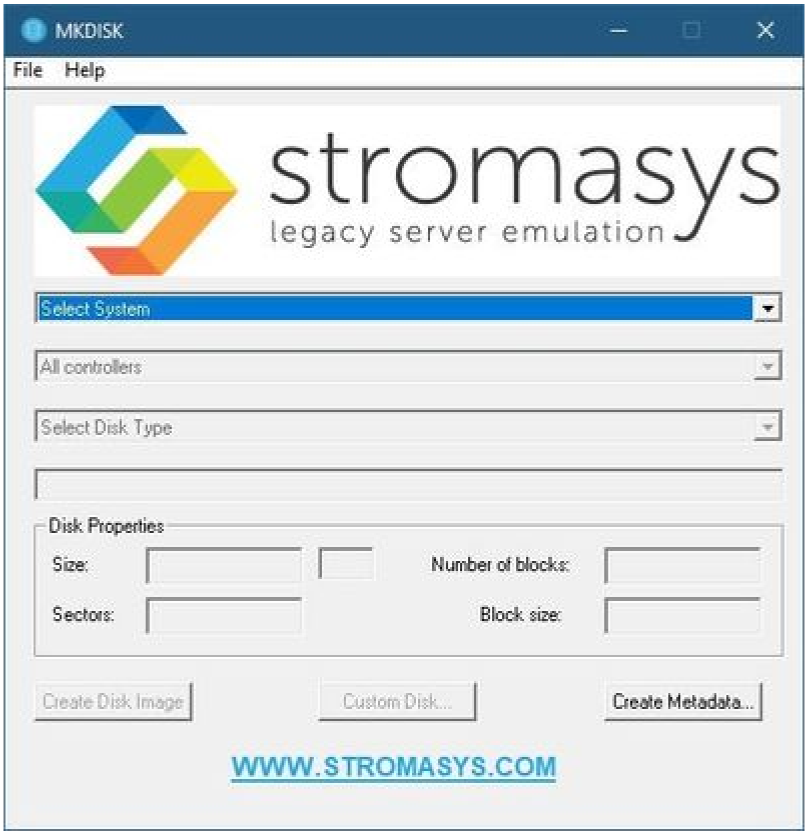
License Update tool (hasp_rus) v. 7.3

Network Control Center application (netdiag) v. 1.56

CHARON device check application (devcheck) v. 1.5

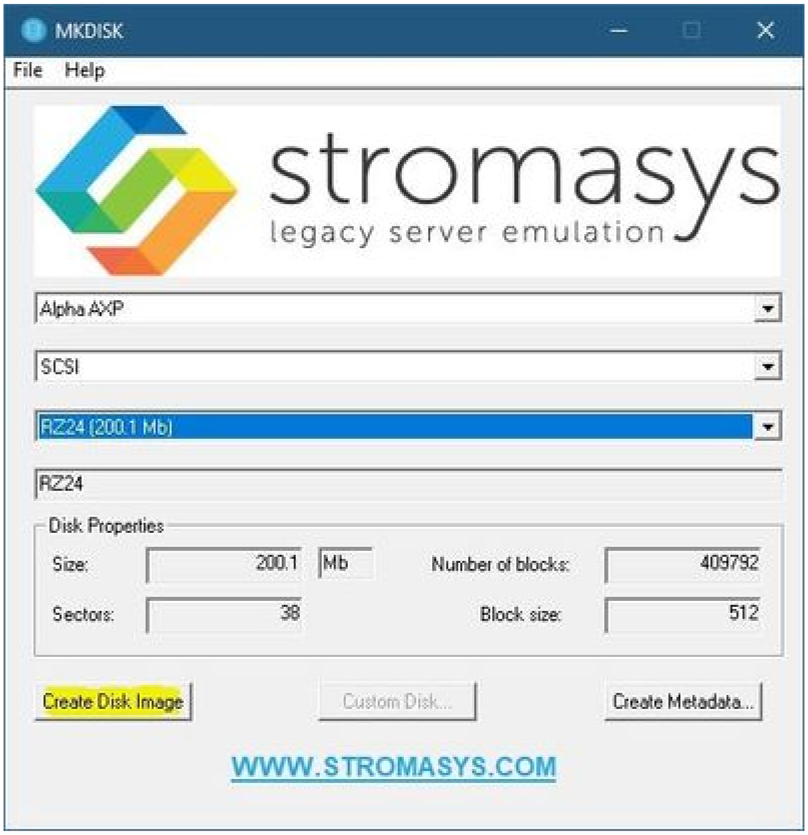
Virtual Disk tool (mkdskwin) v. 2.20

The following dialog will appear:



Select "Alpha AXP" in the "Select System" drop-down menu, "SCSI" in the "All Controllers" drop-down menu and "RZ24" (for example) in the "Select Disk Type" drop-down menu.

Press then the "Create Disk Image" button:



Alpha AXP

SCSI

RZ24 (200.1 Mb)

RZ24

Disk Properties

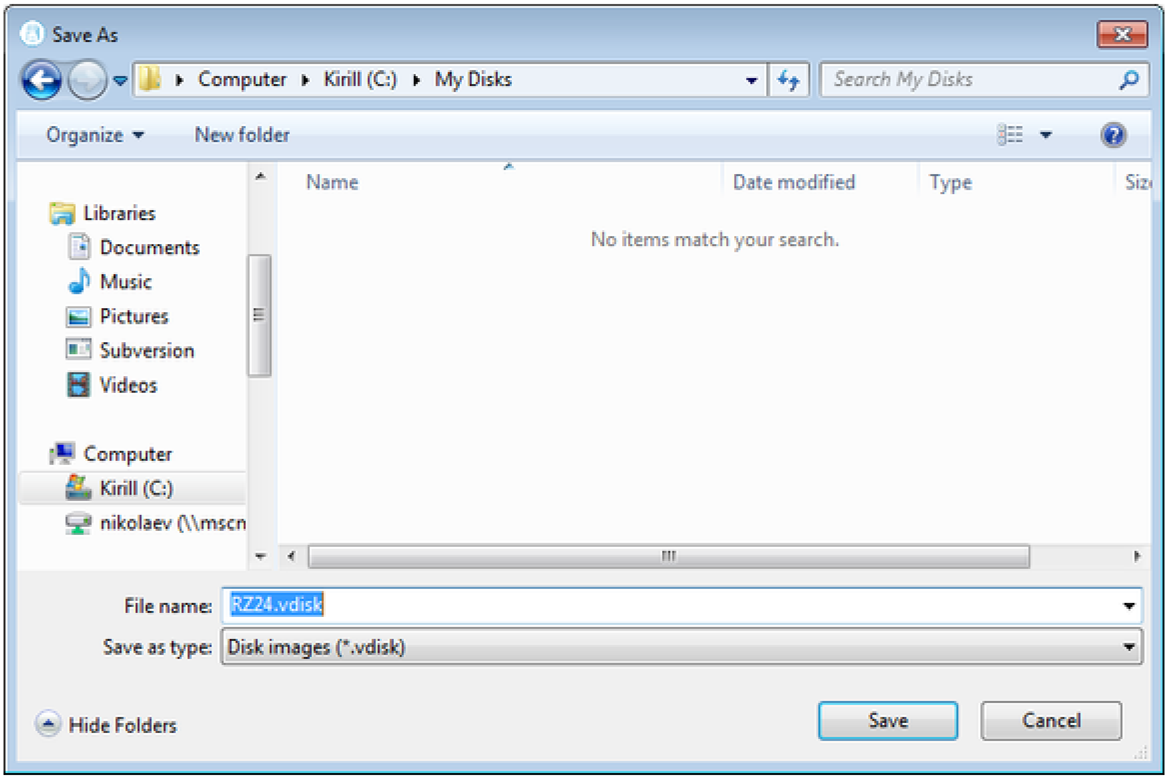
Size: 200.1 Mb

Number of blocks: 409792

Sectors: 38

Block size: 512

The utility will ask you to specify the output file name. Select the "C:\My disks" directory that was created earlier, or create it directly in the dialog, and press the "Save" button:

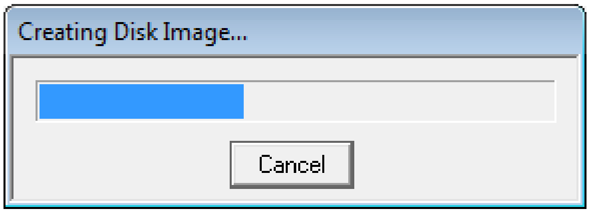


No items match your search.

File name: RZ24.vdisk

Save as type: Disk images (*.vdisk)

The "MkDisk" utility will create the requested disk image:



Repeat this sequence for the disks "user-data.vdisk", "application-data.vdisk" and "application-binaries.vdisk". Place them in the same folder.

HINT: Avoid the temptation to name the vdisks according to the disk-model used to create them (example: "RZ56.vdisk"). Such a name tells you absolutely nothing about the purpose of the vdisk. Instead, assign a name to the vdisk file which tells you about its content or purpose, for example "openvms-system", "user-data", "db-data-1", etc.

Installation of Alpha operating system

The next step is to transfer the data from the source Alpha system to CHARON-AXP. The easiest way to do this is via backup over the network but for this operation a bootable network-enabled operating system on a CHARON-AXP disk image or physical disk is needed.

The example configures the CHARON-AXP AlphaServer ES40 system for installation of OpenVMS from a distribution CD-ROM (usually it is "\\.\CdRom0" if the host has only one CD-ROM drive):

```
#
# DEC-KZPBA SCSI controller is mapped to 5 disk containers; one of them (DKA300) - for migration purposes;
# another one (DKA400) - for installation of fresh OpenVMS system from distributive
#

load KZPBA PKA scsi_id = 7

set PKA container[0] = "C:\My disks\bootable.vdisk"
set PKA container[100] = "C:\My disks\user-data.vdisk"
set PKA container[200] = "C:\My disks\application-data.vdisk"
set PKA container[300] = "C:\My disks\migration.vdisk"
set PKA container[400] = "C:\My disks\fresh_openvms.vdisk"

#
# CD-ROM for OpenVMS installation (DQA0)
#

set ide container="\\.\CdRom0"
```

⚠ DKA300 will be the disk where all the source disks will be copied so its size needs to be large enough to store all the disk backup images.

Create an empty disk image for installation of OpenVMS and another one for storing the backups from the source Alpha system as it is shown in the section above.

Run the CHARON VM and boot from the CDROM named "dqa0" ("migration.cfg" is the configuration file we use in this example):

```
CHARON-AXP/ES40 for Windows x64 (AlphaServer ES40 6/667), Version 4.11.20404
(C) 2009-2020 STROMASYS SA.
All rights reserved.

P00>>>boot dqa0
```

Install Alpha/VMS including DECnet on "dka400". The DECnet address must belong to the same area as the source Alpha system.

Login to the newly installed OpenVMS system and initialize the disk that will be used to store the backups. Let's assume its prompt is "`newvms$` "

```
newvms$ INIT DKA300: SCRATCH
newvms$ MOUNT/SYSTEM/NOASSIST DKA300: SCRATCH
```

Making remote backups

Now we are ready to create the disk backups of the source Alpha system on the CHARON VM.

Boot the CHARON VM and make sure that the source Alpha system is available via DECnet.

Login to the source Alpha system. Stop all the batch queues, kick off the users, stop all the applications and close the databases if there are. The commands listed in SYS$MANAGER:SYSHUTDWN.COM may be helpful. The goal is to close as many files as possible. The system disk will have several files opened (pagefile, swapfile, etc.) and this is a normal situation.

ℹ The use of the "SHOW DEVICE /FILES" command would be of help to list opened files on a disk.

In this example, the CHARON VM system is node 1.400.

Issue the following commands from the source Alpha. Let's assume its prompt is "`source$` ":

```
source$ BACKUP/IMAGE/IGNORE=INTERLOCK DKA0: 1.400"username password"::DKA300:[000000]DKA0.BCK/SAVE
source$ BACKUP/IMAGE/IGNORE=INTERLOCK DKA100: 1.400"username password"::DKA300:[000000]DKA100.BCK/SAVE
source$ BACKUP/IMAGE/IGNORE=INTERLOCK DKA200: 1.400"username password"::DKA300:[000000]DKA200.BCK/SAVE
```

When the backup operation will be completed, the disk "DKA300" of the CHARON VM will contain 3 savesets: "DKA0.BCK", "DKA100.BCK" and "DKA200.BCK".

Restore backups to CHARON-AXP disks

The savesets have now to be restored on their corresponding virtual disks. Login to the CHARON VM guest OS and issue this sequence of commands:

```
newvms$ MOUNT/FOREIGN DKA0:
newvms$ BACKUP/IMAGE DKA300:[000000]DKA0.BCK/SAVE DKA0:
newvms$ DISMOUNT DKA0:
newvms$ MOUNT/FOREIGN DKA100:
newvms$ BACKUP/IMAGE DKA300:[000000]DKA100.BCK/SAVE DKA100:
newvms$ DISMOUNT DKA100:
newvms$ MOUNT/FOREIGN DKA200:
newvms$ BACKUP/IMAGE DKA300:[000000]DKA200.BCK/SAVE DKA200:
newvms$ DISMOUNT DKA200:
```

If you are going to have the CHARON VM and the original physical Alpha on the network at the same time, you must change the network identity of one system, usually the CHARON VM.

The easiest way is to boot the CHARON VM on the restored system disk with the network disabled and to configure new addresses then enable the network and reboot.

ℹ The NIC can be disabled with a `interface="(disabled)"` statement in the CHARON VM configuration file.

Alternative ways of data transfer

Some alternative methods of data transfer are also possible. For example:

- Connect a SCSI tape drive to the CHARON-AXP host via a PCI card
    - Map the tape drive in the CHARON VM configuration file
        a. Restore the source Alpha system backups from tape to disk images via OpenVMS running on the CHARON VM.
        b. Boot from standalone backups and restore the content to the CHARON VM virtual disks
    - Dump the source Alpha system backups to tape images using the "mtd" utility and:
        a. Boot from the freshly installed OpenVMS system and restore the tape images to the CHARON VM virtual disks
        b. Boot from the standalone backup and restore the content to the CHARON VM virtual disks
- Create a network cluster between the source Alpha system and the CHARON VM (it is possible to use the source system as a boot server); then perform simple backups from one disk to another:

    ```
    $ BACKUP/IMAGE/IGNORE=INTERLOCK REAL$DKA0: DKA0:
    ```

CHARON-AXP for Windows licensing

Please refer to the other sections of the *Charon Licensing Handbook* and your product documentation for any topics not covered in this overview guide and for more detailed information on the topics discussed here.

This page provides an overview of the basic steps to set up the licensing environment on a system running Charon-AXP or Charon-VAX for Windows:

General Description

A Charon emulator product will not run without a valid license. For Sentinel HASP licenses (the topic of this chapter) this license can be provided on a **USB hardware dongle** (a Sentinel HASP key) or as a **software license** bound to the hardware of the host system or the network license server. A loss or defect of a license will cause the emulator to stop.

For extra protection, Stromasys recommends the use of a backup license key (purchased separately) that can replace the main license key for a limited period of time. It is possible to specify the backup license in the Charon configuration file to prevent the emulator from stopping in case its main license is no longer accessible.

To check the validity of a license, Charon performs the following steps:

- The Charon emulator checks the text block of the license to determine if it is allowed to run (if there is a product section which corresponds to the emulator).
- Then Charon checks the feature associated with the product to see if the feature is valid - that is, it is not expired, the available runtime hours are not exhausted, and if there are still some instance-slots available to run the emulator.
- If there is no matching product section or if the feature is not valid, Charon reports the problem in the log and shuts down.

Charon License Checks

Charon products **check the availability of a valid license** at different points in time:

1. At startup:

If no valid license is found, an error message will be written to the emulator log file and the emulator will not start.

Charon-AXP/VAX has a configuration parameter to define the number of retries (`license_key_lookup_retry`). Please refer to the product configuration documentation for more details.

2. At regular intervals during the runtime of the emulator (the default license check period of 1 hour can be changed by Stromasys using the appropriate license parameters):

- If the previously used valid license has been removed, has disappeared, is defect, or has become invalid, the emulator will report the loss of the license in the log file and continue operation for a limited amount of time as described below.
- If there is another valid license, for example a backup license defined in the configuration file, it will be used.
- *Starting with Charon-AXP/VAX 4.9*, Charon allows for a grace period of 12 hours during which the software checks for the presence of a valid license every 10 minutes until a valid license is found. If no valid license is found after the grace period has expired, the emulator will stop.
- *Earlier versions of the above products and other emulator products*: should there still be no valid license at the next regular license check (this default interval may be changed by individual Charon products), the emulator will stop.
- If a time-restricted license is used and it expires, the Charon instance tries to find its replacement automatically and, if found, proceeds using the replacement license.

Important note for Windows installations: any running emulators will be stopped if the license service is forced to stop despite existing service dependencies (i.e., running emulators). So any running guest systems must be cleanly shutdown before restarting the license service in order to avoid data loss. This is the behavior at the time of writing. It may change in future versions.

Charon License Updates

Under certain circumstances existing licenses must be updated.

License updates are required, for example, in the following cases:

- Charon product version upgrade
- Product change
- License validity (time) extension

License updates, for example a new expiration date or a new execution counter, can be performed without interrupting operation, that is, without shutting down active Charon instances, **provided that the resulting license is valid for the running Charon instance**. If any parameters of the updated license are not valid for the running Charon instance (for example, the wrong model or version), the Charon instance will stop running.

Licenses are **backwards compatible to some extent**. A **license update to a more recent version** will allow some previous Charon versions to run as well. The details depend on the Charon product.

The license check on **Charon-AXP/VAX/PDP starting with version 4.4** will be successful for a newer license if the major version (for example, the 4 in 4.7) is the same. In **version 4.3**, the emulator will start, but there are inconsistencies and incompatibilities especially in the associated tools (e.g., hasp_view). Product **versions 4.2 and earlier** are incompatible with the current license schema.

License Content

The license data on a license key, independent of whether it is a hardware or software license key, consists of **several parts:** the **license key text block** contains the **license section** and **one or more product sections**. The **license features** describe additional conditions and limitations for the emulator products contained on the license.

A) License Key Text Block

1) License Section (one per license key)

The **license section** contains general license information related to ownership and administration.
There is **one license section** per license key.
Please note that in rare cases, you may receive an **empty USB dongle** for which command outputs will show that **no license information was detected**. In such cases, request a license update from Stromasys as described in *Updating a License*.

Relevant **license section parameters**:

- **License type**: type of the license, i.e., hardware or software license, network-enabled or local (added in newer Charon product versions, e.g., Charon-AXP/VAX 4.9)
- **License number**: assigned by Stromasys
- **License key ID*:*** unique**;** pre-programmed; unmodifiable; on physical keys same value as the Physical Key ID
- **Master key ID**: the software vendor ID used by Stromasys for issuing licenses
- **Release date**: the day on which the license was issued
- **Release time**: the time at which the license was issued
- **Update number**: the number of license updates that have been applied
- **End user name**: the name of the person to whom the license was issued
- **Purchasing company name**: usually the company to which the license was issued

2) Product Section (one or more per license key)

The text block also contains **product sections** that contain product specific licensing information.

- Several products can be combined on one license key. Stromasys may place restrictions on what products can be combined on a single license key.
- The product section also may include information about the support contract and the end of support. It may contain hardware specific limitations related to the types of devices that can be connected (e.g., serial interfaces).
- A checksum is inserted after the last product section of a license. This checksum is not visible in the show command output. Charon products/versions requiring such a checksum will not start if the license does not contain a checksum.
- Each product section is associated with a **feature defining additional conditions** for the product.

The following list shows the most important **parameters of a product section**.

Please note that not all of these parameters may be present for a particular Charon product.

- **Product License Number:** identification of specific product section (optional; new starting with Charon-AXP/VAX 4.9 and Charon-PAR 1.10). Corresponds to configuration parameter **license_id**.
- **Virtual Hardware**: range of virtual models available for running
- **Product Name**: commercial product name
- **Product Code**: commercial product code
- **Major Version, Minor Version, Maximum Build, Minimum Build:** commercial product version and range of build numbers suitable for running with this license. The parameters **Maximum** and **Minimum Build** are no longer displayed starting with Charon-AXP/VAX 4.9 and Charon-PAR 1.10.
  Please note that, depending on the product and product version, licenses have a limited backward compatibility (i.e., when the version on the license is newer than the product version).
  See *Operational Procedures Overview* in the Sentinel HASP Licensing Documentation for more information.
- **Host CPU supported** or **32bit and 64bit OS requirements:** type of host CPU required (for example, x64)
- **Host Operating System required**: Linux or Windows
- **CPUs allowed:** number of virtual CPUs enabled for virtual multiprocessor systems
- **Maximum virtual memory:** Maximum memory emulated. If not present the value defaults to the maximum memory possible for the particular virtual system. Note that the maximum memory may not be available to the virtual system if the host computer has insufficient physical memory.
- **CHAPI enabled:** whether or not CHAPI (Charon-AXP API) can be used with this product
- **Instances allowed:** maximum number of Charon emulator instances of this program/hardware model that are allowed to run concurrently on the local host. Relevant for local license keys that do not support the process count parameter. **See also** the parameter *Number of processes* in the feature section.

- **CPU Speed**: optional parameter to reduce the speed at which the program can run to a fraction of the maximum (e.g., to 25%)
- **HPSUSAN**: optional parameter. Specific to HP3000, overrides physical key ID. The HPSUSAN (HP System Unique Serially Assigned Number) is used for licensing purposes by HP. The parameter may be needed if a license on the physical system is tied to the original HPSUSAN.
- **Feature number:** number of the feature defining further limits/restrictions for the product license. Each product can use a separate feature, or the same feature can be used by several products.
  If using a **network license** and the same feature is shared by several products, the number of instances allowed by the feature is shared between the products (e.g., 2 allowed instances enable one Charon-AXP and one Charon-SSP instance to run concurrently). If using a **local license dongle** and the same feature is shared by several products, the number of allowed instances is counted separately for each product.
  The parameters defined in a feature are described below in the *Features* section.

B) License Feature Section (one or more per license key)

Features define **product specific conditions**, for example expiration date and number of concurrent instances.

The following list describes these conditions:

- **Released product expiration date:** for time-based licenses. Note that at the time of writing the license of Charon-AXP/VAX products will expire at the end of the day of the expiration date, the license of Charon-SSP in the early morning of the expiration day. The exact behavior of the different versions can change without prior warning. Not supported for HL-MAX dongles.
- **Runtime counter:** number of license checks after which the license will be disabled. Default check interval is one hour. The counter will be decreased at the start of the emulator and then at every license check interval. Once the counter has been depleted, the license becomes invalid. This license type is typically used on licenses intended as backup licenses.
- **Login interval** or **Check interval:** frequency of license checking during emulator execution (default 1 hour). This parameter is not visible in the show command output in older Charon product version. It is visible in the show command output of newer versions, starting with Charon-AXP/VAX 4.9 and Charon-PAR 1.10.
- **Network license:** indicates that a physical license is a network license
- **Number of processes:** maximum overall number of Charon instances that are allowed to run concurrently.
  This parameter is set by Stromasys to the value matching the respective customer contract. For network dongles, the maximum possible value of this parameter is determined by the type of network dongle. For example: on a NetTime10 dongle this parameter can be set to a value between 1 and 10 (depending on the customer contract). The type of dongle can be viewed via Sentinel ACC.

Example with Time-Based Parameters

The parameters below describe the following limitations to the emulator:

- License permits running up to 2 instances of Charon at the same time.
- License expires on May 31, 2020, **or** after 360 license checks (with an interval of 60 minutes between the checks).

```
Expiration date: 31/5/2020
Runtime counter: 360
Login interval: 60 minutes
Instances allowed: 2
```

If any of the conditions are no longer met (too many instances, past expiration date, all checks exhausted), the feature becomes invalid and will not allow an emulator instance (or an additional instance) to run.

License Type Overview

Depending on customer requirements, Stromasys can provide several different license types.

Local Hardware License

Local hardware licenses are USB dongles and work on the system they are physically connected to. For Windows systems supporting AnywereUSB®, hardware licenses can also be connected over the network (similar products are available for Linux). They can easily be moved to a different system, if required. The content of the dongle can be updated if a change to the license or an extension of a time-based license is required. The necessary steps are described later in this document.

Please note:

- Hardware dongles require the Sentinel HASP run-time (driver) installation before the dongle can be connected to and used by the system.
- Hardware dongles, apart from HL-MAX dongles, are equipped with a **battery** and a clock, which makes them independent of the host clock. The battery is not rechargeable. However, the dongle can use the power provided by the host system while it is plugged in. By doing this, the depletion of the battery can be slowed down. Check the dongle at regular intervals if it is not permanently connected to a system. If the battery becomes completely depleted, the dongle will be permanently unusable and must be replaced. See also: How long does the license USB dongle battery last upon a full charge.
- Hardware dongles are not suited for running Charon emulator products in **cloud environments**. Stromasys provides other licensing solutions adapted to virtual environments (Charon VE licenses) for selected products. See Virtual Environment (VE) License Server Documentation.

Software License

A software license is a "virtual" key with functionality very similar to a HASP network-enabled hardware dongle.

A software license does not require any special hardware but it still **requires installation of the Sentinel runtime environment**.

Important information:

- To avoid unexpected problems, do not use any Sentinel runtime software that was not provided by Stromasys without being advised to do so by your Stromasys representative.
- Software licenses are best suited for stable environments, because their correct function depends on certain characteristics of the host system. **Changing these characteristics will invalidate the license**.
  - If the Charon host runs on **real hardware**, software licenses are by default tightly bound to the hardware for which they were issued. If major hardware characteristics of the system are changed, the license will be disabled.
  - If the Charon host runs in a **virtual environment (e.g., VMware)**, software licenses are normally bound to the virtual machine ID and a set of additional characteristics of the virtual machine. If any of these parameters are changed, the license will be disabled.
  - For a more detailed description of the restrictions, please refer to Software Licensing restrictions or contact your Stromasys representative.
- Software licenses are very sensitive to even small changes on the host system. Therefore, it is especially important to provide for a backup license that will ensure continued operation should there be a problem with the software license. See *Handling Multiple License Keys* for details.
- Standard software licenses are not suited for running Charon emulator products in **cloud environments**. Stromasys provides other licensing solutions adapted to virtual environments (Charon VE licenses) for selected products. See Virtual Environment (VE) License Server Documentation.

Network License

The network Sentinel HASP key (**red USB dongle**) can be shared between several hosts running a Charon emulator product (including the host on which the network license is installed).

Please note: all software licenses are also network licenses.

If the Charon emulator product is installed on the host where the network license is connected, no additional steps are required. The Sentinel driver is installed as part of the Charon product installation. If the host does not have a Charon emulator product installed, the host can still distribute the connected network license to emulator instances running on other hosts.

The Sentinel HASP runtime software must be installed on such a license server and on the client system. For details regarding the installation, please refer to the software installation section in this document. Once both the Sentinel runtime software and the network license are installed, the Charon emulator product can be started on any appropriate client host on the LAN, provided access to the license is enabled.

- The network license will be visible to all hosts that can access the license server over IP. Access to the license server must be possible on **port 1947** via **UDP** (discovery process) and **TCP** (actual access to license). Further information (e.g., use of **additional UDP ports**) can be found in the section *Firewall Considerations*.
- The license server and the client must both allow access to the network license using the appropriate management tools.

The maximum **number of concurrently active** Charon instances is determined by the parameters of the license.

Please note: Charon-AXP/VAX version 4.7 build 171-01 introduced a change with respect to network licenses. In previous versions of Charon-AXP/VAX a network license controlled the maximum number of client host systems and Charon instances per host system (station/instance mode). In the current Charon-AXP/VAX versions a network license controls the maximum overall number of active instances, which can be distributed across client host systems according to the preference of the customer.

Overview of Initial License Installation Steps

Each Charon emulator product requires a valid license to run. To access a Sentinel HASP license, the emulator needs the Sentinel runtime software. This software is part of the Charon-AXP/Charon-VAX for Windows installation kits. Refer to the software installation section of the Charon product user's guide for software installation instructions.

After the installation of this software, the license can be installed on the system. The following steps are described in more detail below:

1. Add a password for the Sentinel Admin Control Center (ACC).
2. If you purchased a **hardware license**, you can simply plug the dongle into a free USB port on the system.
3. If you purchased a **software license**, you need to create a fingerprint file in C2V (customer-to-vendor) format containing the system characteristics. Use this file to request a license for your Charon product from Stromasys.
4. If your license is a **network license** served by a license server, make sure that the access of the client system to the license server is not blocked by the configuration or a firewall.
5. Optionally, define how **multiple licenses** will be used (selecting primary/backup license, defining license priorities).
6. If you have an **existing license** that needs to be updated, you need to create a customer-to-vendor (C2V) file and use this file to request a license update from Stromasys.

Overview of License Management Tools

The following list shows the main tools used to manage licenses on Windows:

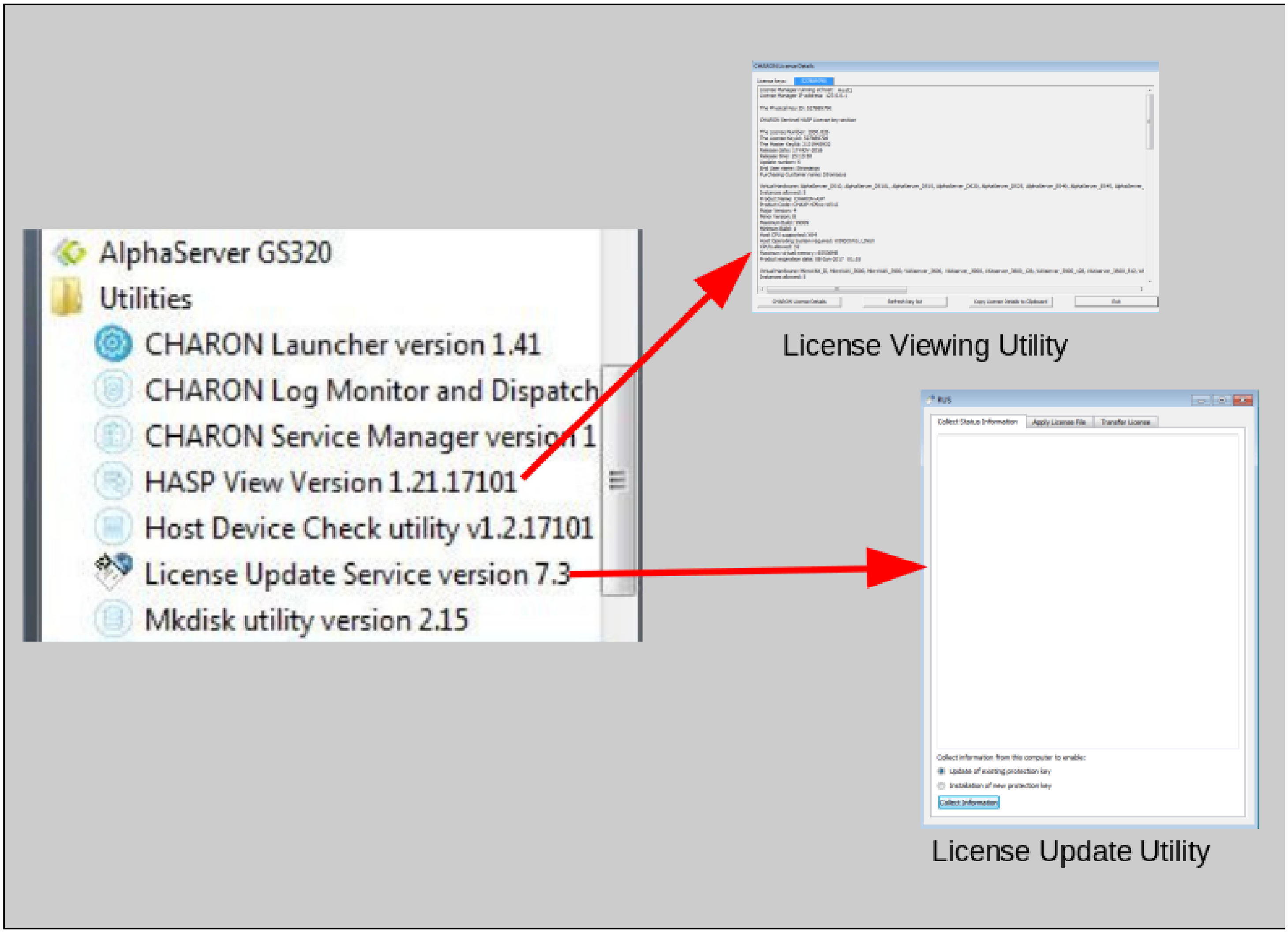
- **Sentinel Admin Control Center (ACC)**: A web-based interface providing important configuration options with respect to licenses.
- The **license viewing utility**: A GUI-based tool to display the detailed license contents. Cannot be used over a remote connection when using local hardware licenses.
- **The license update utility**: A GUI-based tool to extract the information required for creating and updating licenses, and to install the resulting license data.

Accessing the License Management Tools on Charon Version 4.7 and older

The **Sentinel ACC** on the local system is accessed by starting a web-browser and pointing it to the URL: **http://localhost:1947**.

The **license viewing utility** and the **license update utility** are accessed via

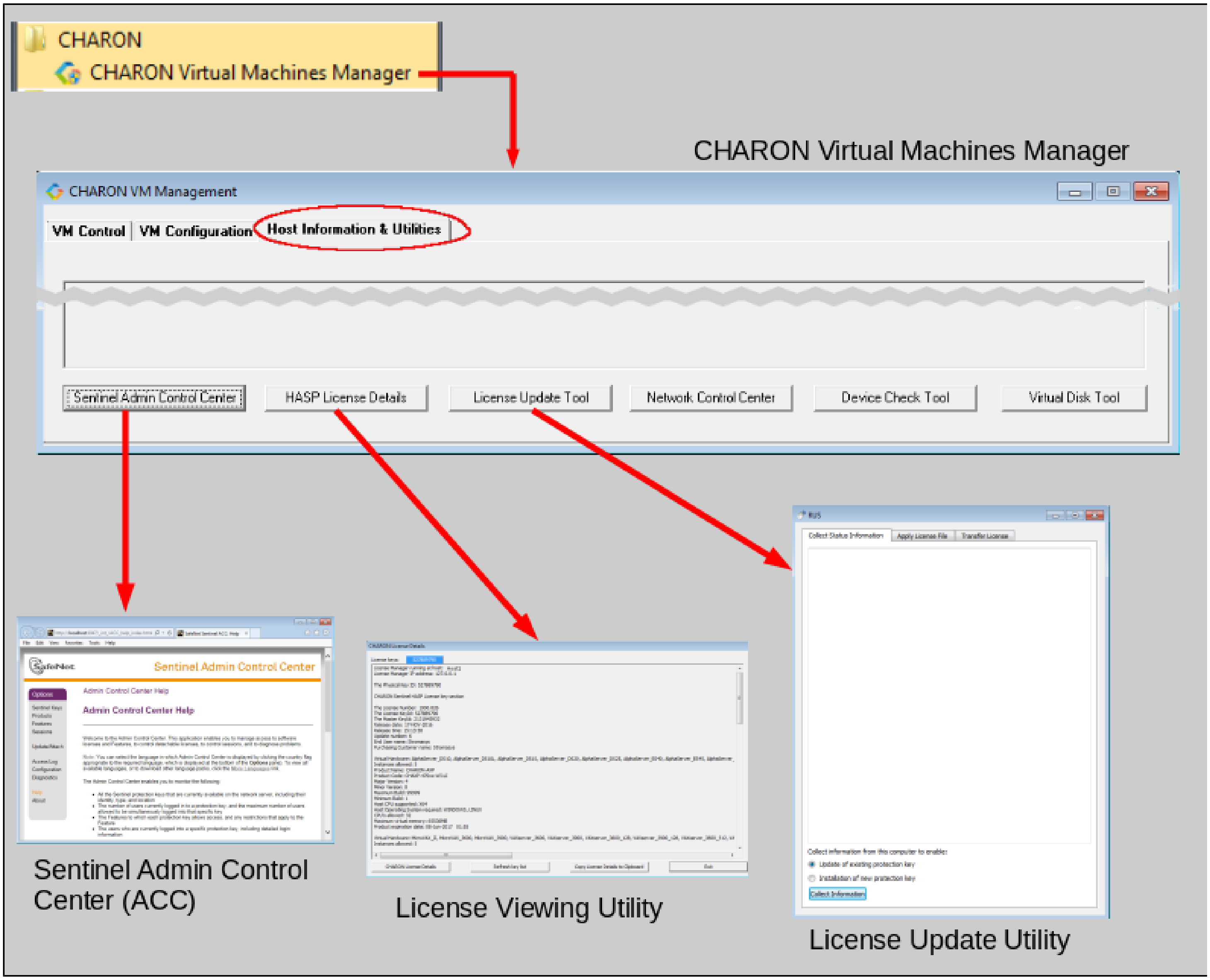
Start > All programs > Charon > Charon product folder > Utilities as shown in the image below.


AlphaServer GS320
Utilities
CHARON Launcher version 1.41
CHARON Log Monitor and Dispatch
CHARON Service Manager version 1
HASP View Version 1.21.17101
Host Device Check utility v1.2.17101
License Update Service version 7.3
Mkdisk utility version 2.15
License Viewing Utility
RUS
Collect Status Information
Apply License File
Transfer License
Collect information from this computer to enable:
○ Update of existing protection key
○ Installation of new protection key
Collect Information
License Update Utility

Accessing the License Management Tools on Charon Version 4.8 and Higher

All license management tools are accessed via the Charon Virtual Machines Manager (VMM) as shown in the image below.

The CHARON VMM is started via **Start > Charon > CHARON Virtual Machines Manager** or by clicking on the associated desktop icon.


CHARON
CHARON Virtual Machines Manager
CHARON Virtual Machines Manager
CHARON VM Management
VM Control
VM Configuration
Host Information & Utilities
Sentinel Admin Control Center
HASP License Details
License Update Tool
Network Control Center
Device Check Tool
Virtual Disk Tool
Sentinel Admin Control Center
Admin Control Center Help
Collect Status Information
Apply License File
Transfer License
Collect information from this computer to enable:
Update of existing protection key
Installation of new protection key
Collect Information
Sentinel Admin Control Center (ACC)
License Viewing Utility
License Update Utility

Define a Password for the Sentinel Admin Control Center (ACC)

Please note: Stromasys strongly recommends performing the steps described below to reduce the risk of unauthorized access to the Sentinel ACC.

By default, anyone on the local system with access to port 1947 can access the GUI. If remote access is enabled, users on the network with access to port 1947 can also access the GUI. To protect access to the GUI with a password, perform the following steps:

Step	Description
1	Open a web browser and navigate to **http://localhost:1947/.**
2	**Click** on the left-hand menu item labeled **Configuration**.
3	**Click** on the **Basic Settings** tab.
4	Under the entry labeled **Password Protection, click** the **Change Password** button.
5	At the **Change Password** window: • Leave the **Current Admin Password** field blank (there is no password set by default). • Enter the desired password into the **New Admin Password** field. • Repeat the desired password in the **Re-enter new Admin Password** field**.** • **Click** the **Submit** button**.**
6	Back at the **Basic Settings** tab: • Under the section labeled **Password Protection,** select the **All ACC Pages** radio button**.** • **Click** the **Submit** button to save this change.
7	Optional: 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 **Submit** button**.** **Additional information for remote access to the Sentinel HASP GUI:** • The firewall must permit access to port 1947/TCP, and • network visibility on the ACC network configuration tab must be set to **All Network Adapters.** If required, remote access can also be enabled on Linux by editing the file */etc/hasplm/hasplm.ini* and setting the parameter **ACCremote** to **1** . Should the file not yet exist, refer to How-to enable remote connection to Sentinel Admin Control Center without GUI. The parameter to enable network visibility is **bind_local_only**. It must be set to **0** to enable all adapters. Changed settings are recognized automatically by aksusbd.

View Existing Licenses

It is important to know which licenses are visible on a system. For example, the user can

- determine if the correct license is installed,
- identify the expiration date on time-based licenses,
- identify the remaining hours of run-time on backup licenses,
- identify the license ID of primary and backup license which are needed for the emulator configuration file.

Licenses can be viewed using two tools:

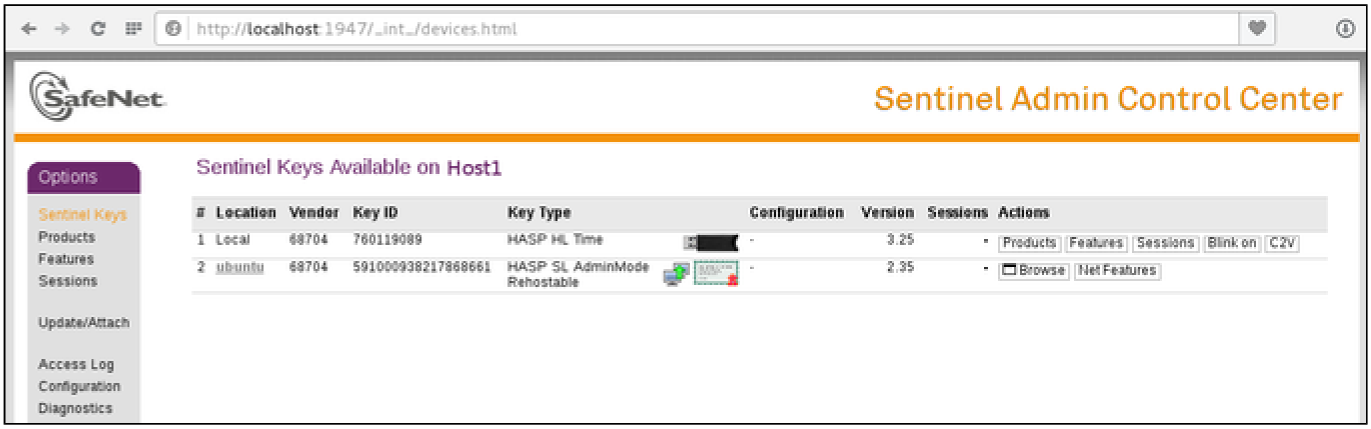
- Sentinel ACC: shows important information, but not the product specific license parameters.
- License viewing utility: shows all product details contained on the license. Can only be run from a local connection for local hardware licenses.

Viewing a License with Sentinel ACC

To view available licenses using Sentinel ACC, start the web interface as described above.

Please note: starting with Charon-AXP/VAX 4.9 a new Sentinel ACC version (>= 7.60) has been included in the product. The functionality is very similar but the appearance is somewhat different.

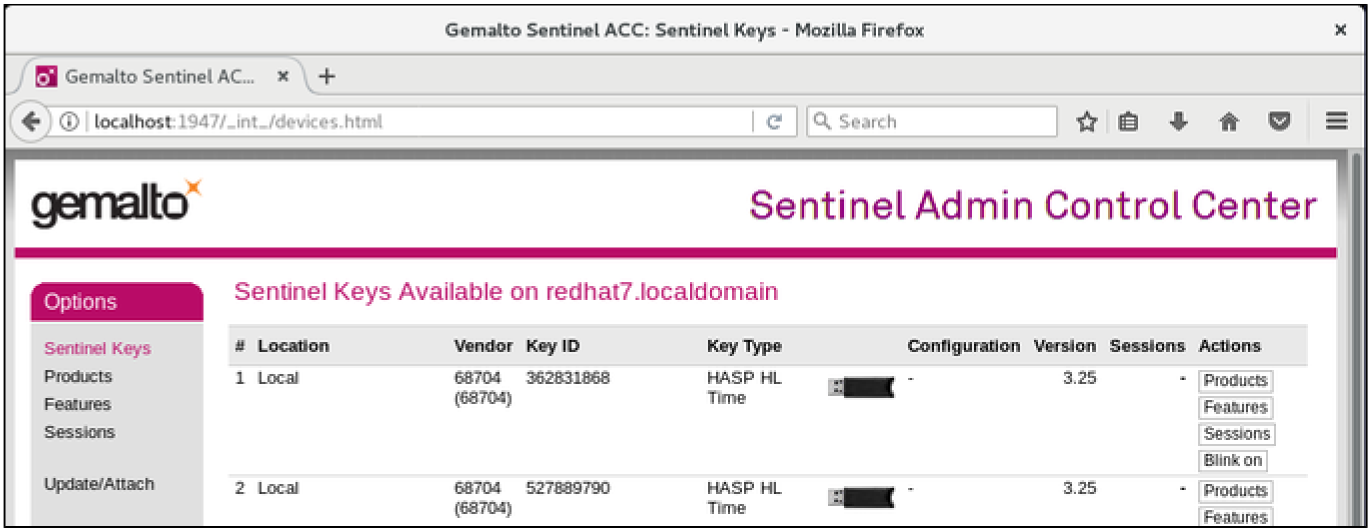
o get to the **Sentinel Keys** screen, click on the corresponding menu item or access the URL **http://localhost:1947/_int_/devices.html** directly. A screen similar to the following opens and displays the available license keys:



Sentinel Keys Available on Host1

#	Location	Vendor	Key ID	Key Type	Configuration	Version	Sessions	Actions
1	Local	68704	760119089	HASP HL Time	-	3.25	-	Products Features Sessions Blink on C2V
2	ubuntu	68704	591000938217868661	HASP SL AdminMode Rehostable	-	2.35	-	Browse Net Features

For Sentinel drivers version 7.60 and above, this screen looks similar to the following:



Gemalto Sentinel ACC: Sentinel Keys - Mozilla Firefox
Gemalto Sentinel AC...
localhost:1947/_int_/devices.html
Search
gemalto
Sentinel Admin Control Center
Options
Sentinel Keys
Products
Features
Sessions
Update/Attach

Sentinel Keys Available on redhat7.localdomain

#	Location	Vendor	Key ID	Key Type	Configuration	Version	Sessions	Actions
1	Local	68704 (68704)	362831868	HASP HL Time	-	3.25	-	Products Features Sessions Blink on
2	Local	68704 (68704)	527889790	HASP HL Time	-	3.25	-	Products Features

This page displays important information about the available licenses, including

- **Location**: Shows whether the license is local or remote. If the license is a network license, the hostname of the remote system is specified. You can access the remote license manager by clicking on the hostname, provided that access is permitted.
- **Key ID**: The unique identification of the license.
- **Key Type**: Hardware keys are marked by the abbreviation HL (hardware license) and a picture of the license dongle. Software licenses are marked by the abbreviation SL (software license).
- **Sessions**: Shows the number of active sessions opened for the specific key.

The buttons on the right-hand side can be used to retrieve more information about the license or to extract the C2V file for a license update. The **Browse** button shown for network licenses will connect to the remote license manager to show the license features.

A C2V file can only be extracted if the license in question is local to the current license manager, i.e., the license manager to which the web browser is connected. You can connect to a remote license manager by clicking on the hostname in the **Location** column (if connections are allowed).

Please note:

- The option to create a C2V file is not available in older versions of the license software.
- Starting with ACC version 7.60, the **option to create C2V files** for USB dongles is only available if it has been **enabled in the basic configuration** section (under the *Configuration* menu item).
- Newer versions of ACC also offer the option to create a fingerprint for a new SW license.

- Some newer versions of ACC show the option on this screen to create a DEMOMA fingerprint. This is a demo key for development purposes and not relevant for Charon products.
- Some newer versions of ACC offer to create a fingerprint for a software license if such a license does not exist yet. Do not use this option. The resulting file has a format that is incompatible with the Stromasys license creation tools. Instead use the tools described in *Installing the License*.

The menu options *Products*, *Features* and *Sessions* on the left-hand side provide the same information as the buttons. However, they show the information for all licenses.

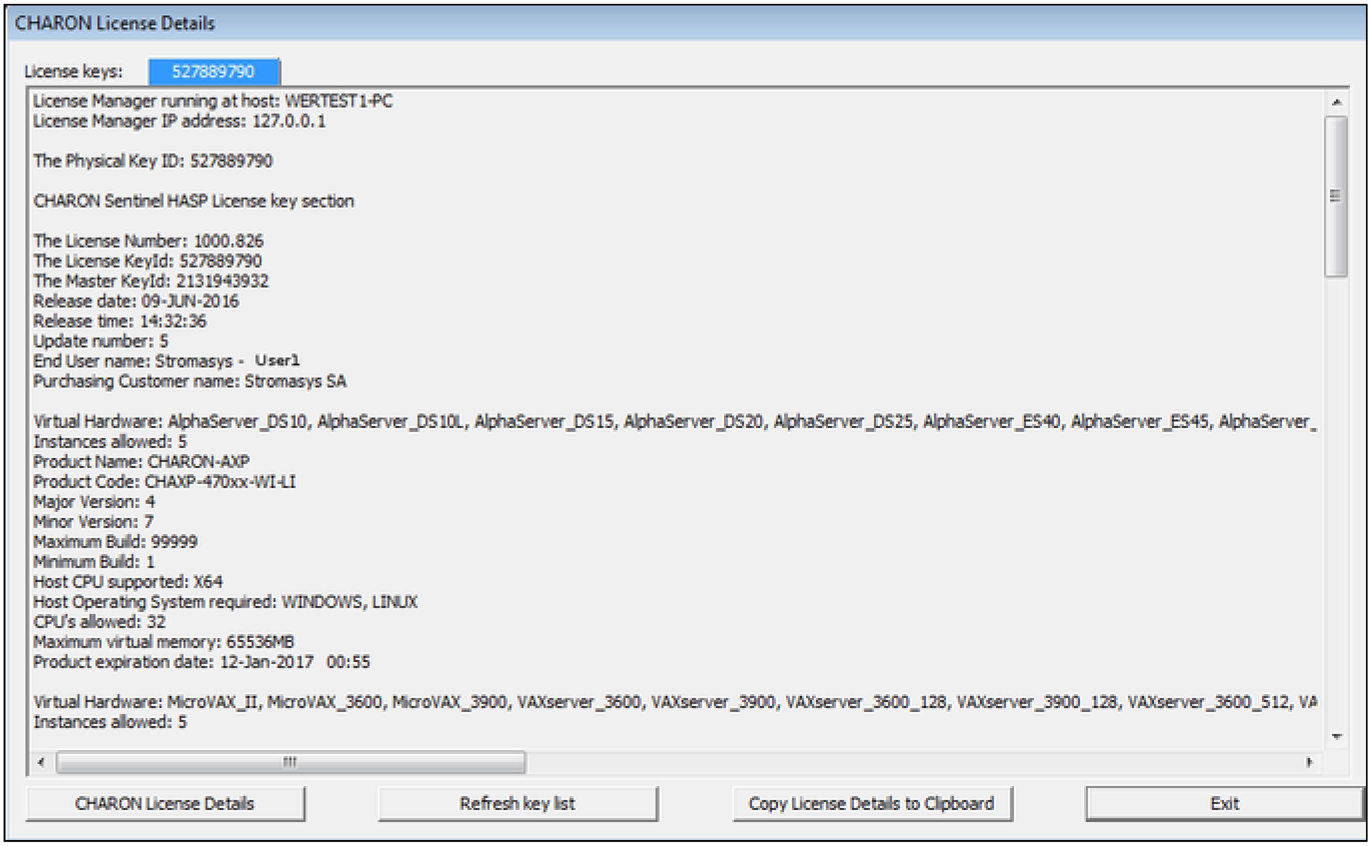
Viewing a License with the License Viewing Utility

To view available licenses using the license viewing utility, start the program as described above.

Please note: a local hardware key cannot be read via a remote connection (e.g., via RDP). One possible workaround is described in detail in the document How to read a HASP key over RDP on Windows. It shows, how to setup a scheduled task for this purpose.

HASP View Utility Window

The following image shows sample output:



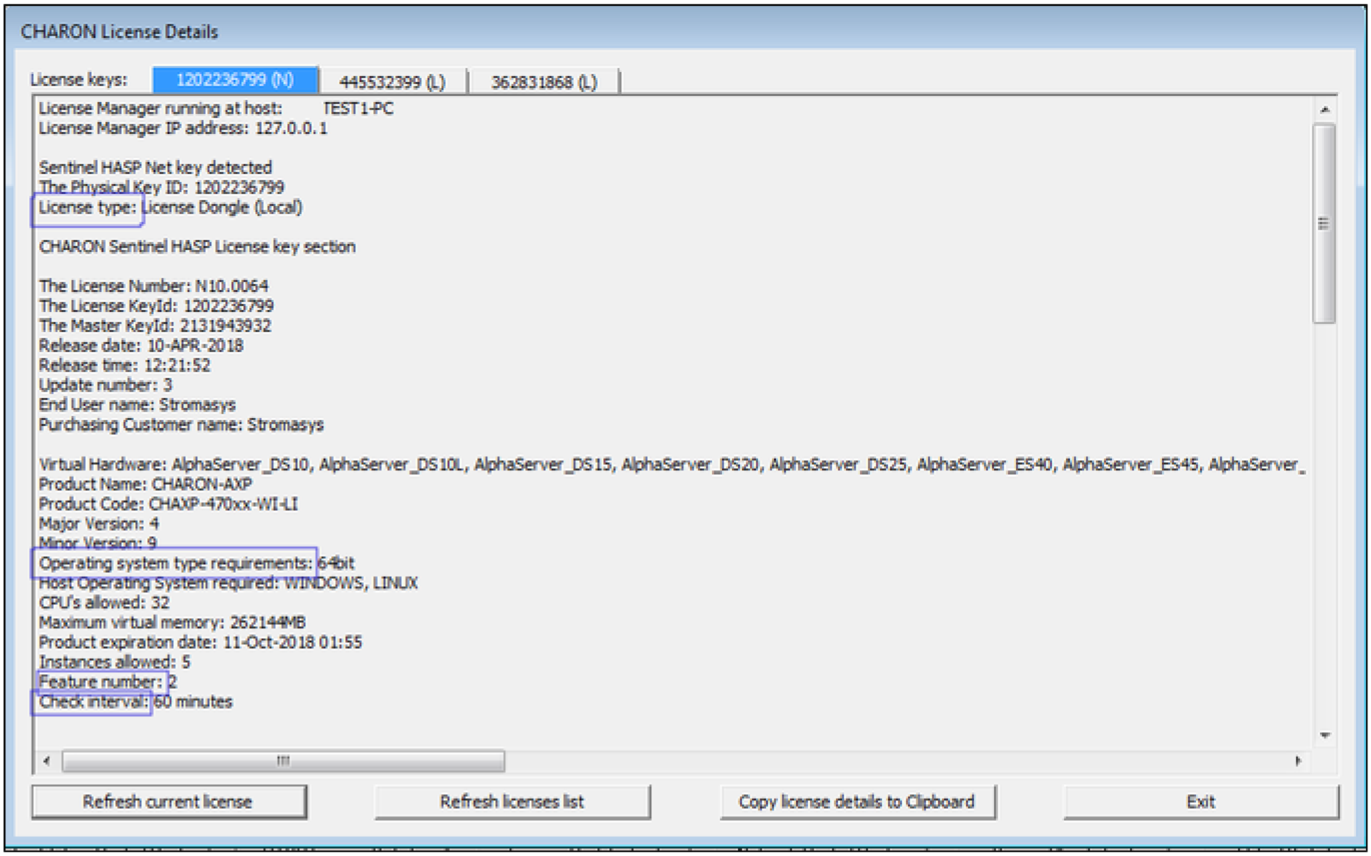
```
License Manager running at host: WERTEST1-PC
License Manager IP address: 127.0.0.1

The Physical Key ID: 527889790

CHARON Sentinel HASP License key section

The License Number: 1000.826
The License KeyId: 527889790
The Master KeyId: 2131943932
Release date: 09-JUN-2016
Release time: 14:32:36
Update number: 5
End User name: Stromasys -  User1
Purchasing Customer name: Stromasys SA

Virtual Hardware: AlphaServer_DS10, AlphaServer_DS10L, AlphaServer_DS15, AlphaServer_DS20, AlphaServer_DS25, AlphaServer_ES40, AlphaServer_ES45, AlphaServer_
Instances allowed: 5
Product Name: CHARON-AXP
Product Code: CHAXP-470xx-WI-LI
Major Version: 4
Minor Version: 7
Maximum Build: 99999
Minimum Build: 1
Host CPU supported: X64
Host Operating System required: WINDOWS, LINUX
CPU's allowed: 32
Maximum virtual memory: 65536MB
Product expiration date: 12-Jan-2017  00:55

Virtual Hardware: MicroVAX_II, MicroVAX_3600, MicroVAX_3900, VAXserver_3600, VAXserver_3900, VAXserver_3600_128, VAXserver_3900_128, VAXserver_3600_512, VA
Instances allowed: 5
```

The buttons at the bottom of the window provide additional functions:

- **Charon License Details** (re-)queries the selected license key (the highlighted tab) and displays the contents**.**
- **Refresh** the key list. This is useful if a key has just been added or removed.
- Copy license details to **clipboard**, for example to paste into a support e-mail.
- **Exit** from the utility.

If there is more than one license, the data for each license is shown on a separate tab.

Please note: starting with Charon-AXP/VAX version 4.9, the button **Charon License Details** has been renamed to **Refresh current license**. The names of some parameters have changed, new parameters were added, and other parameters removed. See the following sample output:



```
License Manager running at host:      TEST1-PC
License Manager IP address: 127.0.0.1

Sentinel HASP Net key detected
The Physical Key ID: 1202236799
License type: License Dongle (Local)

CHARON Sentinel HASP License key section

The License Number: N10.0064
The License KeyId: 1202236799
The Master KeyId: 2131943932
Release date: 10-APR-2018
Release time: 12:21:52
Update number: 3
End User name: Stromasys
Purchasing Customer name: Stromasys

Virtual Hardware: AlphaServer_DS10, AlphaServer_DS10L, AlphaServer_DS15, AlphaServer_DS20, AlphaServer_DS25, AlphaServer_ES40, AlphaServer_ES45, AlphaServer_
Product Name: CHARON-AXP
Product Code: CHAXP-470xx-WI-LI
Major Version: 4
Minor Version: 9
Operating system type requirements: 64bit
Host Operating System required: WINDOWS, LINUX
CPU's allowed: 32
Maximum virtual memory: 262144MB
Product expiration date: 11-Oct-2018 01:55
Instances allowed: 5
Feature number: 2
Check interval: 60 minutes
```

Installing the License

This section provides a short overview of the initial license installation. For more in-depth information about licensing, please refer to the licensing handbook.

Preparation

Before a license can be installed, some preparatory steps are required.

1. Ensure that the correct Sentinel Runtime software is installed. Please refer to the **installation section of your Charon emulator user's guide** for details.
2. If a conflicting or obsolete license is available on the system it may be necessary to **(temporarily) remove or disable** it. If you need to remove a hardware license, simply unplug it. If you need to remove a software license or disable access to a network license, please refer to the relevant chapter in the licensing handbook. Before removing a license or disabling access to it ensure that it is not required by another currently active product.

Installing a Local Hardware License

A local hardware license (USB dongle) is installed by inserting the USB license key into a free USB port of the host system after the Sentinel runtime software has been installed.

After this step, verify that the license is visible to the system by following the steps for viewing a license as described above. Please bear in mind that a local hardware license cannot be read when connected to the system via a remote connection (e.g., via RDP).

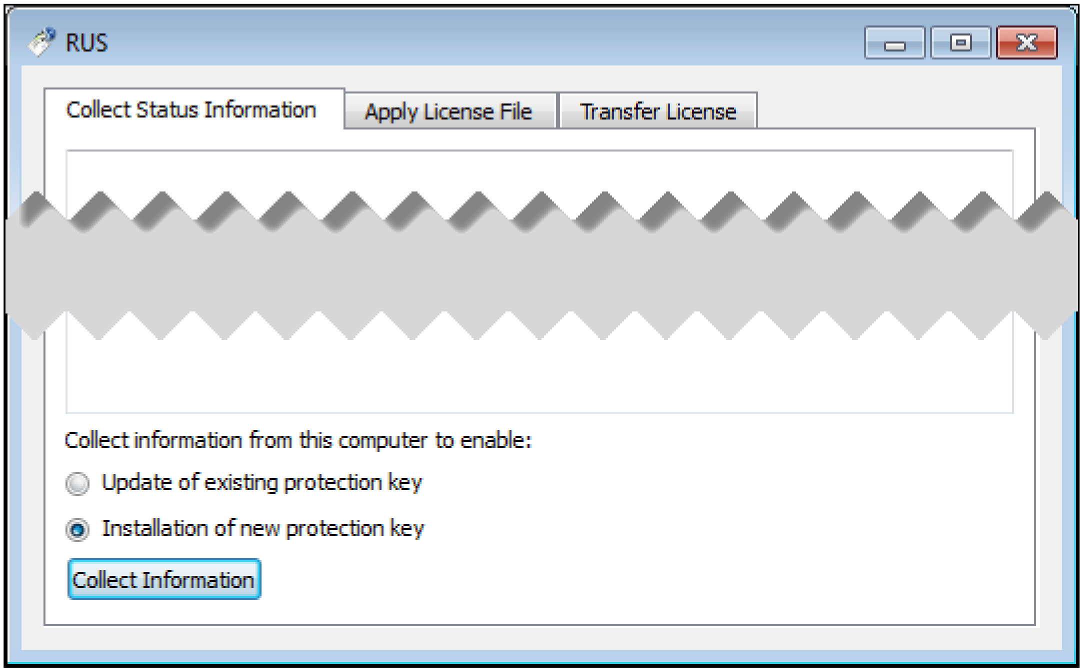
Installing a Software License

To install a new software license, perform the following steps:

Step 1: Start the license update utility as described above and create a fingerprint file:

The license update utility (HASPRUS) looks the same, independent of the product variant used.

The following image shows the **necessary options** for creating a fingerprint file: select the tab **Collect Status Information** and the option **Installation of new protection key**.


RUS
Collect Status Information
Apply License File
Transfer License
Collect information from this computer to enable:
○ Update of existing protection key
● Installation of new protection key
Collect Information

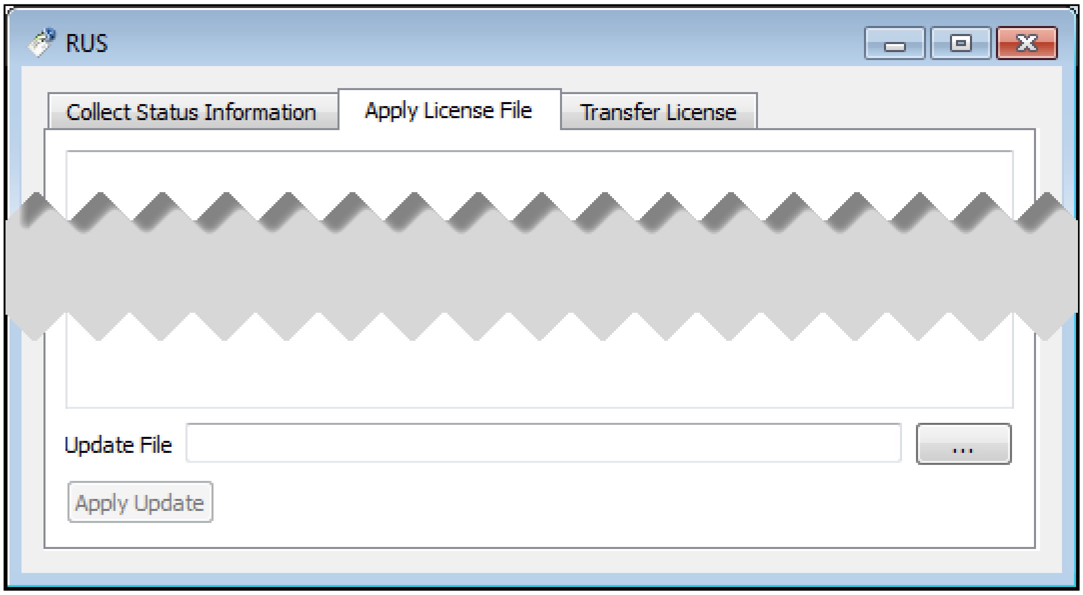
After pressing the **Collect Information** button, you will be prompted to enter a location for storing the fingerprint file. Select a location and filename and save the file.

Once a software license has been installed, current versions of the license update utility do not allow you to create another fingerprint without (temporarily) removing the installed software license. Therefore, save the created fingerprint file in case you need to create an additional software license for the same system (this will be the exception because in most cases the existing license can be updated with the required information). If you don't have the fingerprint file anymore and cannot create a second fingerprint because there is already a valid software license installed, please contact your Stromasys representative or VAR. If you have an obsolete/invalid software license on your system, you can remove it as described in *Removing a License*.

Step 2: Send the resulting fingerprint file to Stromasys orders administration using the email address that Stromasys will provide to you.

Step 3: After receiving the V2C file from Stromasys, copy the file to the system where the license needs to be installed, start the license update utility as described above, and install the new license:

1. To install the V2C file(s) for hardware **and** software licenses, select the tab **Apply License File**.



2. Press the **"..."** button next to the **Update File** field. This will open a file browser.

3. Select the V2C file to be applied.

4. Click on **Apply Update** to complete the operation.

Please note: This section describes the installation of a new software license. However, the installation procedure for a V2C file is identical when updating a hardware license.

Important caveat: When updating a hardware license, you will in most cases receive two V2C files, a ***_fmt.v2c** file and a ***.v2c** file. The *_fmt.v2c file formats the dongle and the *.v2c file contains the updated license data. In such cases the ***_fmt.v2c file** must be applied **first**.

Installing a Network License

For a network license to be provided to a client host on the network, a license server must have been set up either with a network-enabled hardware license (red dongle) or a software license (software licenses are always network enabled).

If the license server also runs a Charon emulator product, follow the software installation steps in the user's guide. Additionally, the licensing handbook contains additional information about setting up a dedicated license server without installing the full product kit using the license-specific packages of a product kit.

For the client to access a license on a license server the following steps are required:

- Any firewall between license server and client must permit the necessary communication.
- The license server must be configured to allow access from the client.
- The client must be configured to allow access to the license server.

Firewall Considerations

The following ports are used for the communication between license server and client hosts:

- On the **server side** (where network license has been installed), port 1947 must be open for incoming TCP and UDP traffic to allow client access to the license.
- On the **client side**, traffic is initiated using ports 30000 through 65535 as the source ports and port 1947 as the target port. If broadcast search for remote licenses is to be used, the client must also permit UDP traffic initiated from port 1947 of the license server to ports 30000 through 65535 of the client.

If a host on the network cannot find the license server even though the server is operational, you can **temporarily** disable the firewall to determine whether it blocks the traffic.

For details on how to configure the firewall in your network, please consult your operating system documentation and make sure to adhere to your company's security policies.

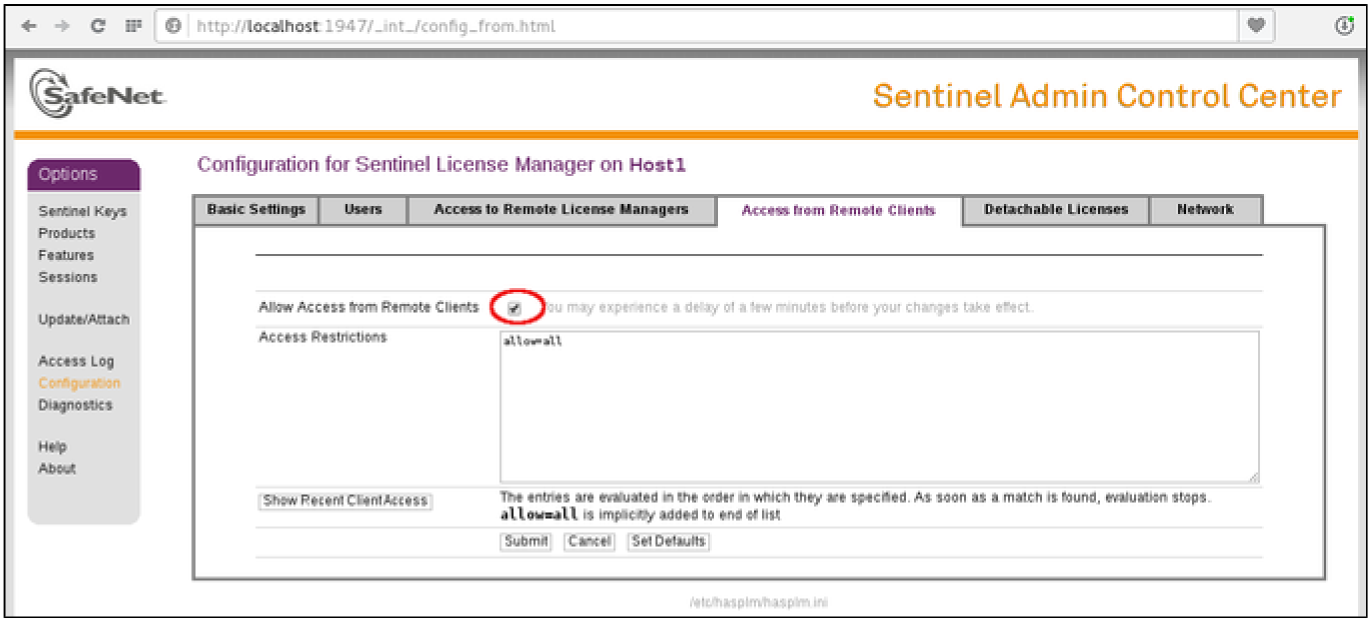
Allowing Client Access on the License Server

The Sentinel license manager on the license server can be configured to **allow** or **disallow** access from remote clients to the network licenses installed on the license server. To access this configuration option, perform the following steps **on the license server**:

1. Open a web-browser and go to the URL **http://localhost:1947/_int_/config_from.html** (option: Configuration / Access from Remote Clients).

2. This opens a page similar to the following. **Please note**: newer Charon emulator products (e.g., Charon-AXP/VAX version 4.9) have newer versions of Sentinel ACC. The pages of these versions look different, but the functionality remains mostly the same.

Old ACC version:



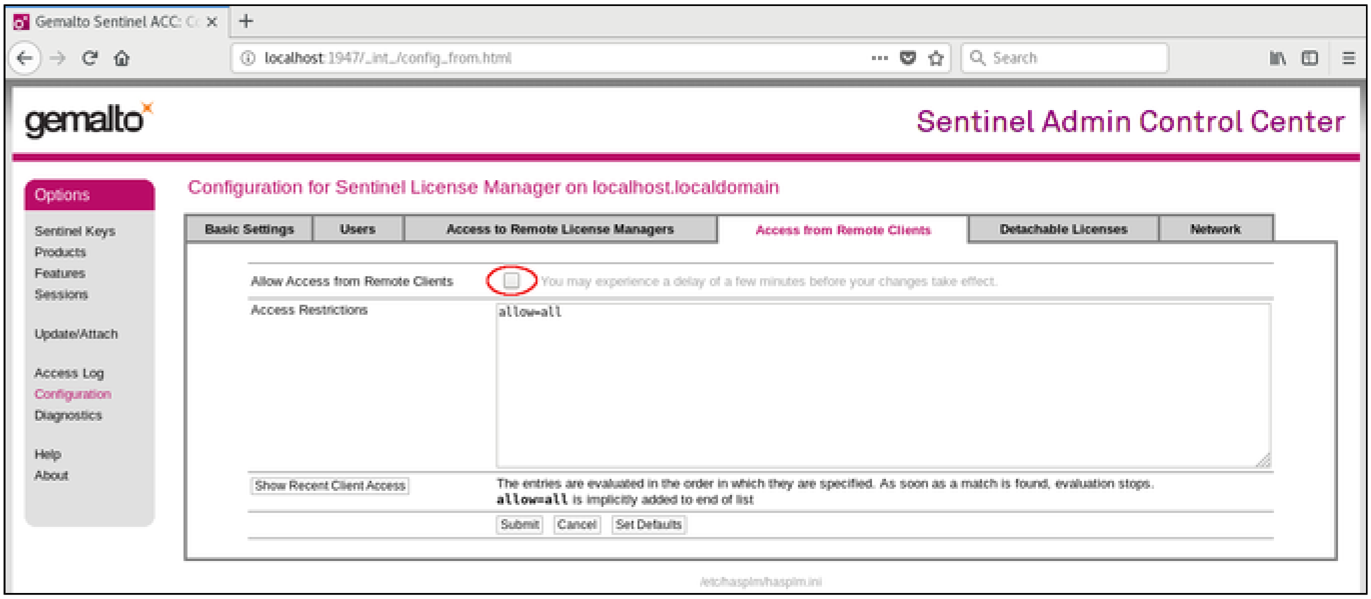
Configuration for Sentinel License Manager on Host1

Allow Access from Remote Clients ☑ You may experience a delay of a few minutes before your changes take effect.

Access Restrictions: allow=all

The entries are evaluated in the order in which they are specified. As soon as a match is found, evaluation stops. allow=all is implicitly added to end of list

New ACC version:



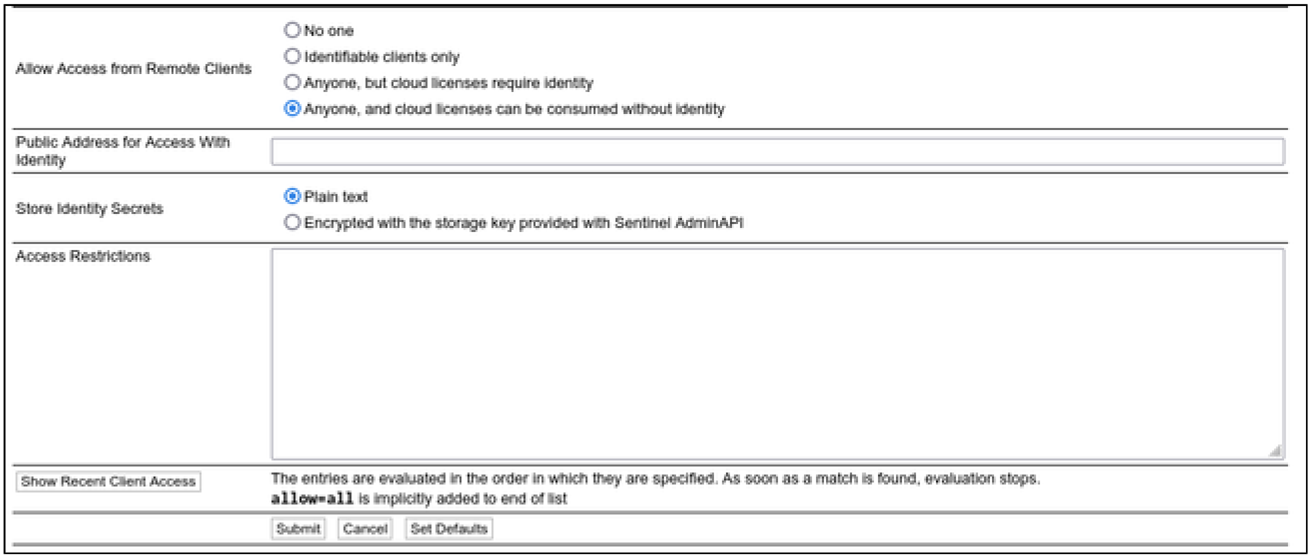
Configuration for Sentinel License Manager on localhost.localdomain

Allow Access from Remote Clients ☐ You may experience a delay of a few minutes before your changes take effect.

Access Restrictions: allow=all

The entries are evaluated in the order in which they are specified. As soon as a match is found, evaluation stops. allow=all is implicitly added to end of list

Or for versions starting with 8.x:



Allow Access from Remote Clients	○ No one ○ Identifiable clients only ○ Anyone, but cloud licenses require identity ● Anyone, and cloud licenses can be consumed without identity
Public Address for Access With Identity	
Store Identity Secrets	● Plain text ○ Encrypted with the storage key provided with Sentinel AdminAPI
Access Restrictions	
Show Recent Client Access	The entries are evaluated in the order in which they are specified. As soon as a match is found, evaluation stops. `allow=all` is implicitly added to end of list

3. Possible actions:

- To **allow access** from remote clients, *activate* the check-box next to the field **Allow Access from Remote Clients** or **Anyone and cloud licenses can be consumed without identity** and press **Submit** at the bottom of the page.
  To allow access from remote clients, network visibility on the "Network" tab must be set to **All Network Adapters.**
- To **refuse access** from remote clients, *clear* the check-box next to the field **Allow Access from Remote Clients** or select **No one**, and press **Submit**.
- **Access Restrictions** allow refining access rules, e.g., by specifying IP addresses. Please refer to Sentinel ACC help for details.

Sentinel ACC versions 8.x and higher have additional configuration options on this screen (mostly cloud related). These options are not relevant to Charon emulator products.

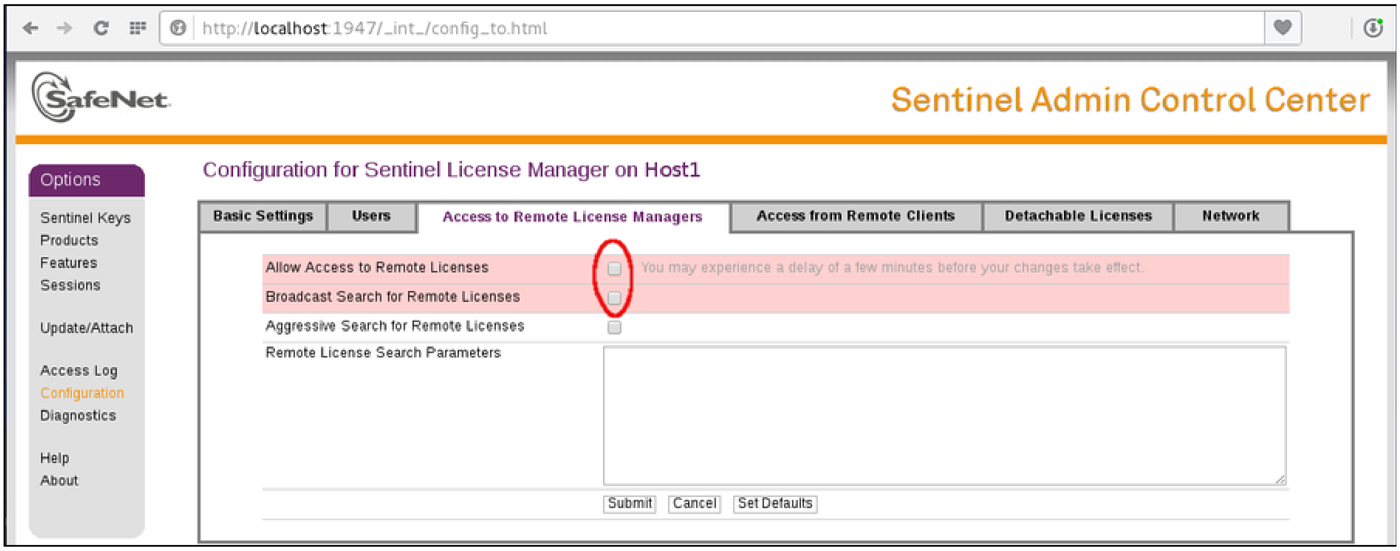
Allowing Access to a License Server on the Client

The Sentinel ACC can be configured to enable or prevent that the client host discovers network licenses and to change the options used to discover and access network licenses provided by a license server.

1. Open a web-browser on the **client host** and go to the URL **http://localhost:1947/_int_/config_to.html** (option: Configuration / Access to Remote License Managers).

2. This will open a configuration page similar to the following. **Please note**: newer Charon emulator products (e.g., Charon-AXP/VAX version 4.9) have newer versions of the Sentinel license drivers. The Sentinel ACC pages of these versions look different, but the functionality remains mostly the same

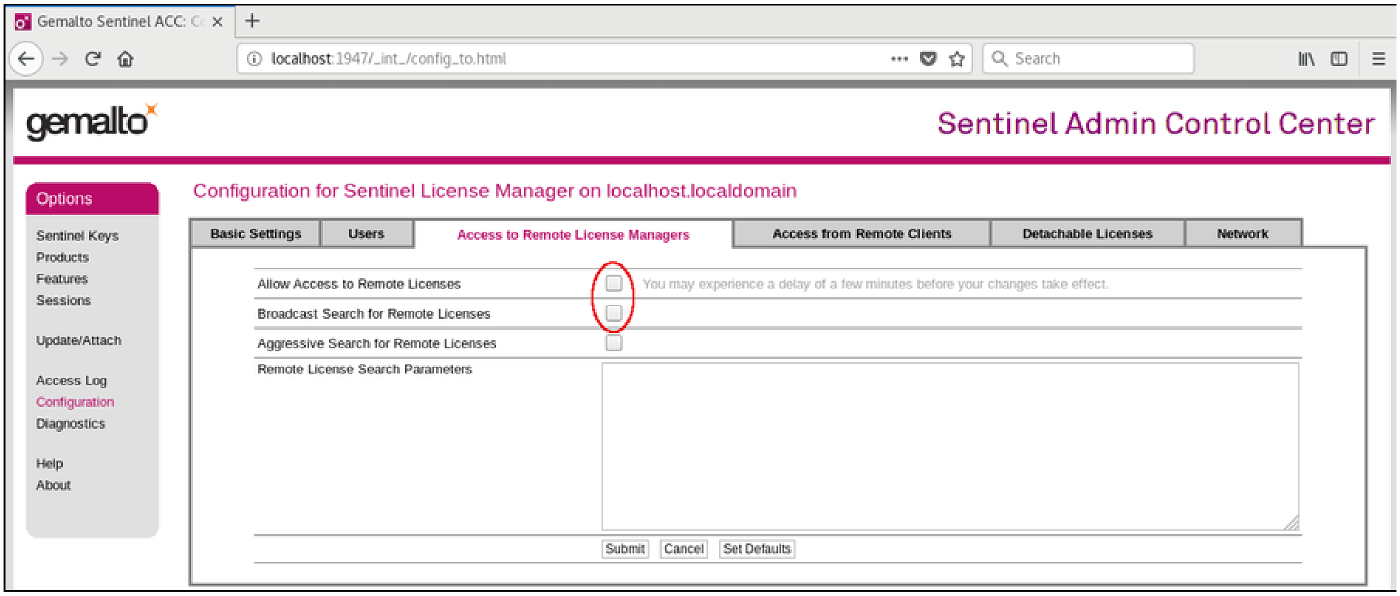
Old ACC version:



Configuration for Sentinel License Manager on Host1

Allow Access to Remote Licenses □ You may experience a delay of a few minutes before your changes take effect.
Broadcast Search for Remote Licenses □
Aggressive Search for Remote Licenses □
Remote License Search Parameters

New ACC version:



Configuration for Sentinel License Manager on localhost.localdomain

Allow Access to Remote Licenses □ You may experience a delay of a few minutes before your changes take effect.
Broadcast Search for Remote Licenses □
Aggressive Search for Remote Licenses □
Remote License Search Parameters

3. Possible actions:

- *Activate* the check-box next to the field **Allow Access to Remote Licenses** to **enable** access to license servers. Press **Submit** to save the setting.
- *Clear* the check-box next to the field **Allow Access to Remote Licenses** to **disable** access to license servers. Press **Submit** to save the setting.
- The option **Broadcast Search for Remote Licenses**, when activated, enables a broadcast search for license servers on the local network without having to enter the address of a license server.

Please note:

- If the option **Broadcast Search for Remote Licenses** is not enabled or cannot be used in the customer specific setting, you can enter specific IP addresses or host names that should be queried for network licenses in the **Remote License Search Parameters** field. Please refer to the Sentinel ACC help function for more information.
- To allow access to remote license managers, network visibility on the "Network" tab must be set to **All Network Adapters**.
- Starting with Charon-AXP/VAX 4.9 for Linux and Charon-AXP/VAX version 4.8 for Windows, these Charon emulator products do not follow the settings in the Sentinel ACC with respect to querying remote license servers and network visibility. They perform a **broadcast search** for network licenses even if this has been disabled in the local Sentinel ACC configuration. If this behavior has to be prevented for specific reasons, the access of the system to the license server has to be temporarily restricted or disabled, for example by blocking the relevant traffic in a firewall. Another possibility would be to block access to the network license at the license server side. Note that such methods can negatively impact other functions of the system or, in the case of blocking access to a network license on the server, even the functions on other license clients.

Verifying License Availability

After installing a license on the system, verify the availability of the license as described in the section *View Existing Licenses*.

Check if the license shows the correct product, expiration date etc.

Handling Multiple Keys

A Charon host system can have access to several local and remote license keys. Each license key can contain one or more product licenses.

The Charon emulator products can only use one **active** license at one time. Without additional configuration, they cannot make decisions about which product license or license key to use. If there is more than one available license key, the default license key will be used. The default license key is determined by the Sentinel software. As this can lead to undesirable effects, newer Charon emulator products have configuration parameters that allow the definition of a **primary** (or production) and a **secondary** (or backup) key, or the creation of a **list of license keys or product licenses sorted** by priority.

The following sections describe the options available to achieve a more deterministic license selection. If only one license is available, either locally installed or via the network, this section does not apply.

Please note: for Charon-AXP/VAX before version 4.9 and other older Charon products: the parameters described in this section can **only** be used to define a primary key and a backup key, or to specify the correct key if there is a conflicting license for a different Charon product on the same system. It is **not possible** with these versions of Charon to combine a local license (black dongle) with other licenses (local or network) to increase the number of concurrent instances of the same Charon product on one host system. Newer versions allow a more flexible use of multiple licenses. Both variants are described in the following sections.

Backup License Characteristics

It is strongly recommended to order a backup key to recover immediately from damage or loss of the main license key. A backup license key can also help in situations where the Charon host hardware fails and the software must be moved to a different system, thus invalidating the original software license.

Backup keys typically use a counter (integer) value programmed in the key. This integer value corresponds to a number of hours the Charon software is allowed to run. Each time the Charon software checks the license (at start and then every hour), the value is decreased (by 1 hour). Please note that backup keys have may have additional restrictions:

- The runtime is typically limited to 720 hours (30 days). This is the time available to get a replacement license from Stromasys.
- A backup license key may be valid only until a certain date.

Please note: if you start and stop the emulator frequently (e.g., frequent runs with a duration of under one hour), the runtime may be significantly less than 30 days, because the license check during the start of an emulator will reduce the counter by one.

Primary and Backup License for Charon-AXP/VAX prior to Version 4.9

If more than one key is visible to the system, you can define which is the primary and which the backup key. To do this, add the following parameters to the configuration file of the Charon instance:

```
set session license_key_id[0]=<primary-key-id>

set session license_key_id[1]=<secondary-key-id>
```

The entry with index 0 denotes the primary key, the entry with index 1 the backup key.

To identify the relevant key IDs, display the available licenses as described in section *View Existing Licenses*.

.

To get information about modifying the Charon configuration files, refer to the appropriate sections in the user's guide.

Prioritized List of License Keys or License IDs for Charon-AXP/VAX 4.9 and Higher

Prioritized List of License Keys

If more than one key is visible to the system, you can define a prioritized list of license keys. To do this, add the following parameter to the configuration file of the Charon instance:

```
set session license_key_id = "<key-id prio1>, <key-id prio2>,...[,any]"
```

The **first entry** in the list has the highest priority. The emulator instance will try to use this key first. **If it is not available**, it will try the second key in the list and so on. If the parameter **any** is added to the end of the list, the emulator instance will search for any keys not part of the list should none of the listed keys be available. If the parameter is not defined, an unqualified search for a suitable license will be performed and the first one found (if any) will be used. If no usable license is found, the emulator will stop.

To identify the relevant key IDs, display the available licenses as described in section *View Existing Licenses*.

To get information about modifying the Charon configuration files, refer to the appropriate sections in the user's guide.

Prioritized List of Product License IDs

Every license key can contain one or more product sections. Newer licenses can contain a product license ID identifying each product section. A prioritized list of product license IDs can be added to the emulator instance configuration file to specify which product sections to use and which should have the highest priority. To do this, add the following parameter to the configuration file of the Charon instance:

```
set session license_id = "<lic-id prio1>, <lic-id prio2>,..."
```

The emulator instance will scan the available license keys for the listed product license IDs. Then it will try to use the first defined product section. If it is not available, it will try the next and so on. If none of the listed product license IDs are found, the emulator will stop.

To identify the relevant product license IDs, display the available licenses as described in section *View Existing Licenses*.

To get information about modifying the Charon configuration files, refer to the appropriate sections in the user's guide.

Update an Existing License

If you need to update an **existing hardware or software** license, for example because the time limit on the license has expired or to upgrade to a new product version, perform the following tasks:

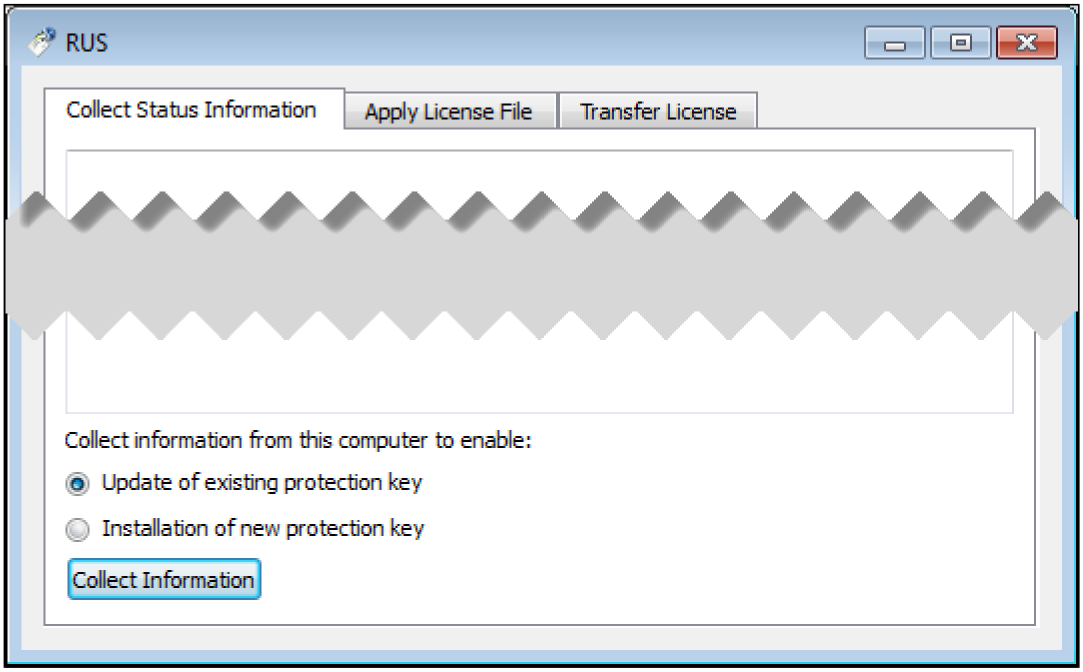
1. Generate the C2V file for the existing license. This step is the same for a hardware license or an existing software license. This Customer-to-Vendor (C2V) file contains the license characteristics necessary for creating the license update.
2. Send the C2V file to Stromasys. Stromasys will use the data to create the necessary license update. You will receive one (software license) or two (most hardware license updates) V2C files (the Vendor-to-Customer file).
3. Apply the license data from the V2C file(s). This will install and activate the update for your license.

These tasks are described below:

Step 1: Start the license update utility as described above and generate the C2V file:

The License Update Utility (HASPRUS) looks the same, independent of the product variant used.

The following image shows the **necessary options** for generating a C2V file for an existing license: select the tab **Collect Status Information** and the option **Update of existing protection key**.



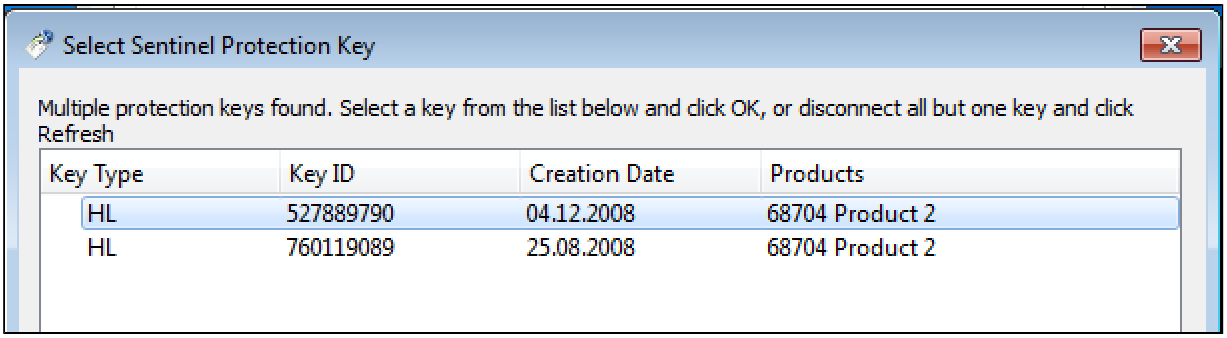
Collect information from this computer to enable:

● Update of existing protection key

○ Installation of new protection key

After pressing the **Collect Information** button, you will be prompted to enter a location for storing the C2V file. Select a location and filename and save the file.

If there is more than one license key available on the system, the tool will allow you to select the appropriate key to update as shown in the screenshot below:



Multiple protection keys found. Select a key from the list below and click OK, or disconnect all but one key and click Refresh

Key Type	Key ID	Creation Date	Products
HL	527889790	04.12.2008	68704 Product 2
HL	760119089	25.08.2008	68704 Product 2

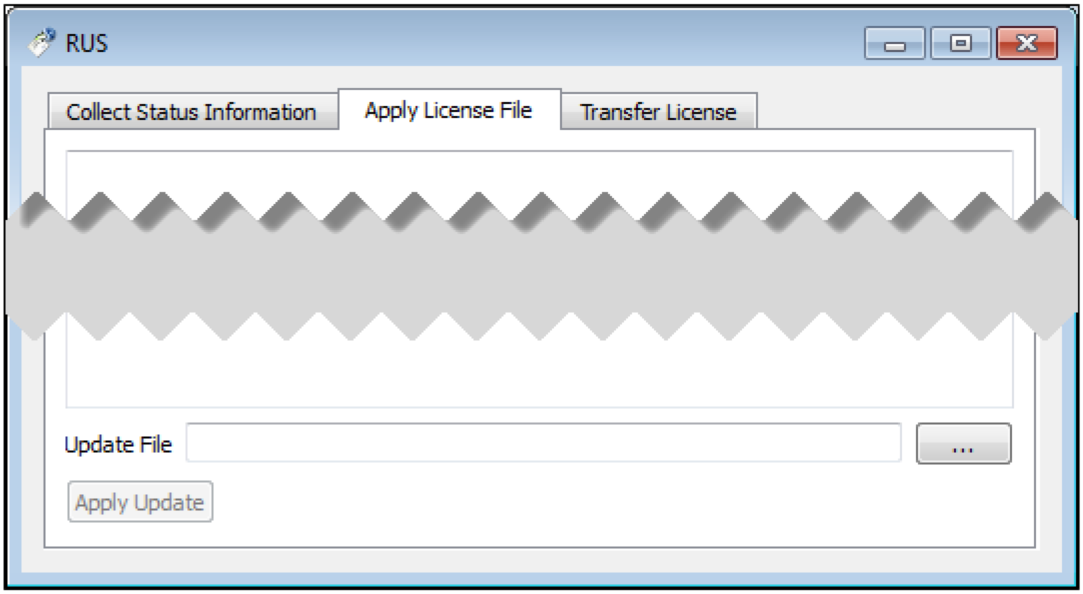
Step 2: Send the resulting C2V file to Stromasys orders administration using the email address that Stromasys will provide to you.

Step 3: After receiving the V2C file(s) from Stromasys, copy the file(s) to the system where the license needs to be installed, start the license update utility as described above, and install the new license:

Important caveat: if a hardware USB key is to be updated, you will in most cases have received **two** files: a ***_fmt.v2c** file and a ***.v2c** file. The *_fmt.v2c file formats the dongle and the *.v2c file contains the updated license data. In such cases the ***_fmt.v2c** file must be applied **first**.

Important caveat: if a hardware USB key is to be updated, you will in most cases have received **two** files: a ***_fmt.v2c** file and a ***.v2c** file. The *_fmt.v2c file formats the dongle and the *.v2c file contains the updated license data. In such cases the ***_fmt.v2c** file must be applied **first**.

1. To install the V2C file(s) for hardware **and** software licenses, select the tab **Apply License File**.



2. Press the **"..."** button next to the **Update File** field. This will open a file browser.

3. Select the V2C file to be applied.

4. Click on **Apply Update** to complete the operation.

If you received two V2C files, perform these steps for both files (remember to **first** apply the ***_fmt.v2c** file).

Alternatively, you can use the Sentinel ACC to apply V2C files (use the section Update/Attach). Refer to the license handbook for more details.

Removing a License

If a license needs to be **removed** from a system or the access to the license needs to be **disabled**, the steps are different for the different license types and are described in the next sections:

- Removing a hardware license from a system
- Removing a software license from a system
- Removing access to a network license from a system

Important: before removing a license from a system or disabling access to a license, take to following points into consideration:

- Make sure that there is a valid replacement license for all emulator instances depending on the license to be removed, or make sure the emulated systems have been cleanly shut down,
- Make sure that this step will not negatively impact your production environment.

Removing a Hardware License from a System

To remove a hardware license from a system, simply unplug it from the system.

Removing a Software License from a System

Please note:

> The old method for removing a software license from a system depended on the license transfer functionality. **This functionality is no longer available in the current versions of the Charon products.** The only way to remove a non-transferable obsolete software license is the method described below. On Microsoft Windows, you can use the **Charon License Utility** (see CHARON on Windows - Charon License Utility) to perform the necessary steps.

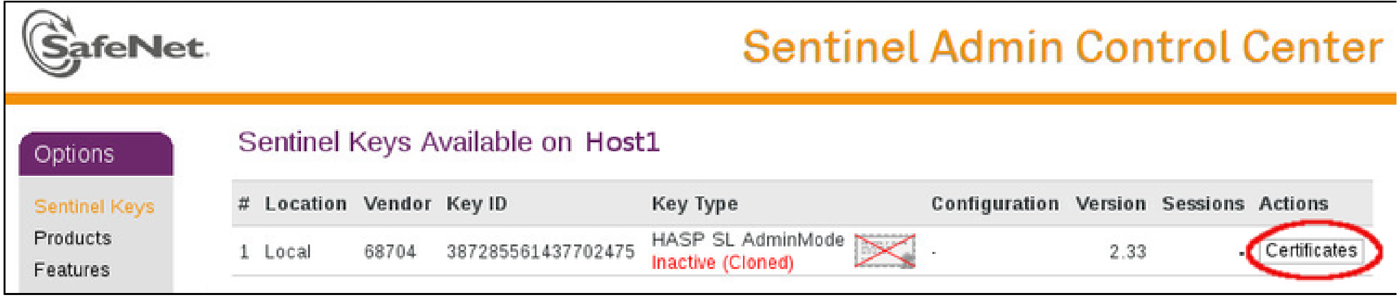
To remove an **obsolete software license or a software license in cloned state** (a software license goes into a cloned state, for example, if a VM is cloned to a different Virtual Machine UUID) from the system, perform the following basic steps:

- Start the Sentinel Admin Control Center (ACC) (for local system: URL **http://localhost:1947**).
- Identify the software license that is to be removed.
- Identify the name of the certificate file and the path of the base directory where it is stored.
- Remove the certificate file(s).
- Restart the Sentinel runtime service.

The following example shows the required steps in detail:

1. Go to **http://localhost:1947/** to access the Sentinel HASP ACC and select **Sentinel Keys** on the left.
2. In the Sentinel Admin Control Center (ACC), go to **Sentinel Keys** and locate the target "Sentinel SL AdminMode" license to be removed.
3. Click on the **Certificates** button at the right side of the SL description:

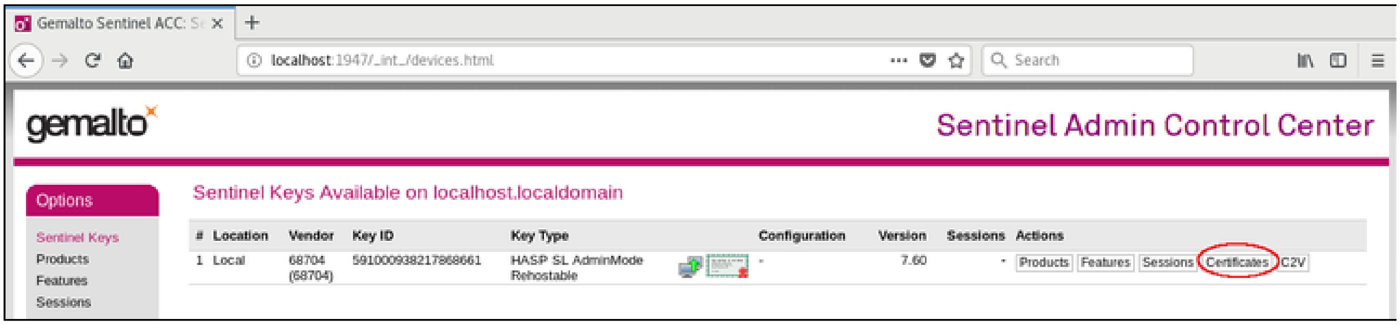
Old version (showing a software license in cloned state):



Sentinel Keys Available on Host1

#	Location	Vendor	Key ID	Key Type	Configuration	Version	Sessions	Actions
1	Local	68704	387285561437702475	HASP SL AdminMode Inactive (Cloned)	-	2.33	-	Certificates

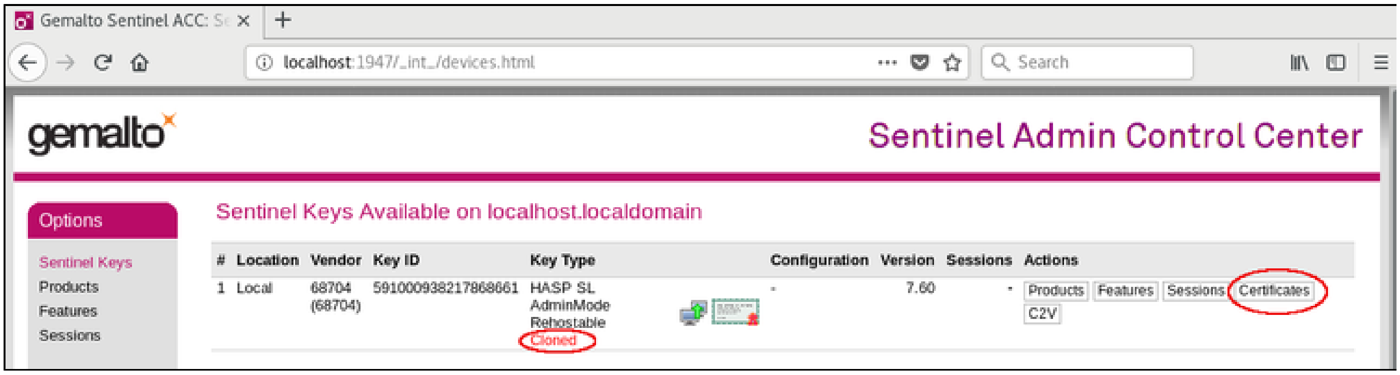
New version (showing an active software license **and** a cloned software license):



Sentinel Admin Control Center

Sentinel Keys Available on localhost.localdomain

#	Location	Vendor	Key ID	Key Type	Configuration	Version	Sessions	Actions
1	Local	68704 (68704)	591000938217868661	HASP SL AdminMode Rehostable	-	7.60	-	Products Features Sessions Certificates C2V



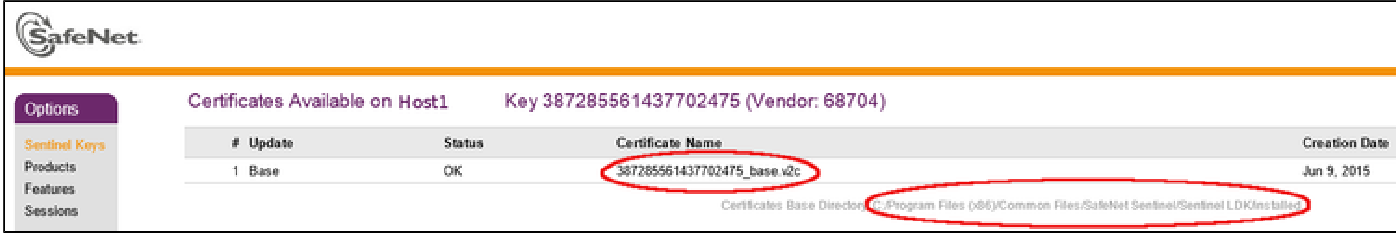
Sentinel Admin Control Center

Sentinel Keys Available on localhost.localdomain

#	Location	Vendor	Key ID	Key Type	Configuration	Version	Sessions	Actions
1	Local	68704 (68704)	591000938217868661	HASP SL AdminMode Rehostable Cloned	-	7.60	-	Products Features Sessions Certificates C2V

4. Note the name of the corresponding certificate file and the path to the certificates base directory as shown in the "Certificates" section:

Example using old ACC version:

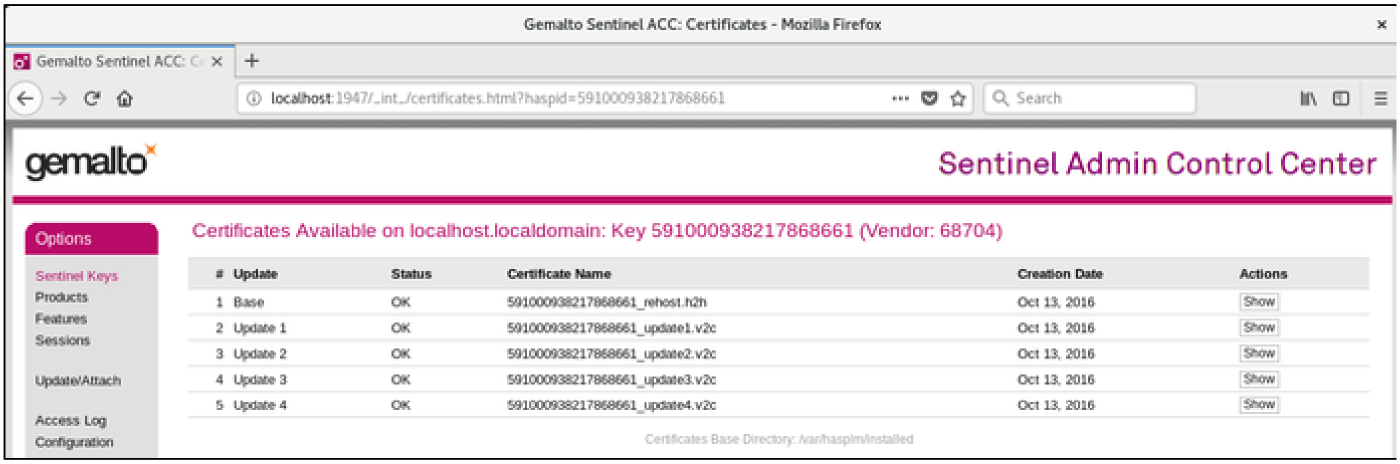


Certificates Available on Host1 Key 387285561437702475 (Vendor: 68704)

#	Update	Status	Certificate Name	Creation Date
1	Base	OK	387285561437702475_base.v2c	Jun 9, 2015

Certificates Base Directory: C:/Program Files (x86)/Common Files/SafeNet Sentinel/Sentinel LDK/installed

Please note: the certificate display screen can look slightly different depending on the type of software license and the actions performed on this license in the past. The following example shows the **certificates of a software license after several updates**. The screenshot is based on Sentinel ACC version equal or higher than version 7.60.



Gemalto Sentinel ACC: Certificates - Mozilla Firefox

localhost:1947/_int_/certificates.html?haspid=591000938217868661

Sentinel Admin Control Center

Certificates Available on localhost.localdomain: Key 591000938217868661 (Vendor: 68704)

#	Update	Status	Certificate Name	Creation Date	Actions
1	Base	OK	591000938217868661_rehost.h2h	Oct 13, 2016	Show
2	Update 1	OK	591000938217868661_update1.v2c	Oct 13, 2016	Show
3	Update 2	OK	591000938217868661_update2.v2c	Oct 13, 2016	Show
4	Update 3	OK	591000938217868661_update3.v2c	Oct 13, 2016	Show
5	Update 4	OK	591000938217868661_update4.v2c	Oct 13, 2016	Show

Certificates Base Directory: /var/hasplm/installed

5. **Remove the certificate file(s)** using operating system tools.

   a. In the first example above (Windows), there is one file to be removed. It is in *C:\Program Files (x86)\Common Files\SafeNet Sentinel\Sentinel LDK\installed\68704*.
   b. In the second example (Linux) there are several files created by a series of license updates. On Linux, the certificate files are typically stored in */var/hasplm/installed/68704/*.
   c. **All certificate files in the identified directory must be removed**.

6. After removing the certificate file, **restart the Sentinel license service** or reboot the system. To restart the service, proceed as follows:

   a. On Windows, restart the "**Sentinel LDK License Manager**" service (hasplms) either using the `services.msc` program or via a command line (as administrator) then enter the "**`sc stop hasplms`**" command followed by "**`sc start hasplms`**".
   Important: any running emulators will be stopped if the license service is forced to stop despite existing service dependencies (i.e., running emulators). So any running guest systems **must be cleanly shutdown** before restarting the license service in order to avoid data loss.

   b. On Red Hat Enterprise Linux 7.x and 8.x (and compatible Linux systems) restart the aksusbd service: `#` **`systemctl restart aksusbd`**
   This also will automatically restart the **hasplmd** service (separate service for this daemon was introduced in newer runtime driver versions).

   c. On Red Hat Enterprise Linux 6.x, restart the aksusbd service: `#` **`service aksusbd restart`**

7. Start the Sentinel ACC again to verify that the software license has been removed.

Removing Access to Network Licenses from a System

There are three different ways to remove a network license. They have different implications and are described below:

Full Removal of Network License

To completely remove a network license, follow the steps above for hardware or software licenses respectively. The steps must be performed on the license server.

Please note: this will remove the license for all emulator hosts on the network using this license and also for the local host. Make sure that this is in fact intended.

Block Access of Remote Emulator Hosts to Network License

You can use the Sentinel Admin Control Center on the license server

- to **allow** access to a network license for all or specific hosts on the network, or
- to **block** access from remote hosts completely.

Please refer to the section *Controlling Access to Network License on Server Side* in *Installing a Network License* for more information.

Block Access to Network Licenses on Client System

If access of the local system to network licenses is not desired, this can be disabled in the Sentinel ACC for certain products and product versions. Blocking access at the server side is the safer way.

Please refer to the section *Controlling Access to Network License on Client Side* in *Installing a Network License* for more information.

Manual Sentinel HASP License Driver Installation

This is normally not required as the license drivers are part of the Charon emulator product installation kits. Also, Stromasys provides a license update utility that simplifies this task.

New License Driver Installation

Before installing the license make sure that the Sentinel run-time (driver) to be installed is collected from your Charon kit or provided by Stromasys on request for your particular product. Do not update the Sentinel run-time (driver) from online or any other sources, including the Sentinel web site.

In case of several Charon kits containing different versions of Sentinel run-time (driver), the last one, having the most recent version, must be installed. The Charon installation performs this operation automatically.

At the moment Charon supports Sentinel HASP keys, Sentinel HL and Sentinel Software Licenses (SL).

Installation of a Charon regular or network license consists of:

1. Installation of the Sentinel run-time environment on the Charon emulator host (regular and network keys) or on the host that will distribute Charon licenses over a local network segment (network key only). The Sentinel software is installed automatically by the Charon emulator products for Windows.
2. Physical connection of the HASP license dongle to the Charon host or to the host distributing the Charon license over the local network segment.
3. Collecting the system fingerprint (*.c2v file), sending it to Stromasys and applying the update (*.v2c file) in case of software license (and in case of updating HASP dongle license). See the details below.

When a manual installation of the Sentinel run-time is required (in the case of the network license server that does not have a Charon emulator product installed):

1. Login as Administrator
2. Open the Charon kit folder
3. Switch to the `hasp_install` subfolder
4. Unzip the archive `haspdinst.zip` located in this folder
5. Open **cmd.exe** and switch to the folder where the files were unzipped
6. Execute the following commands:

```
...> haspdinst.exe -fr -kp -nomsg

...> haspdinst.exe -install -cm
```

7. Extract the contents of this archive `hasplib.zip` to the same directory.
8. Copy the file `haspvlib_68704.dll`
   to `C:\Program Files (x86)\Common Files\Aladdin Shared\HASP`" (in case of a x64 host) or
   to `C:\Program Files\Common Files\Aladdin Shared\HASP` (in case of a x86 host)

Note that the following operations have to be performed when installing a network license:

- *On the server side (where the network license will reside)*: open port 1947 for both TCP and UDP
- *On the client side*, if broadcast search for remote licenses is to be used, UDP traffic from port 1947 of the license server to ports 30000-65535 of the client must be permitted.
- *Both on server and client sides*: setup default gateway

Please consult with your Windows User's Guide on details.

If stricter firewall rules are required, it is possible to open the ports 30000-65535 and 1947 only for the *Sentinel HASP License Manager* (hasplms.exe) service (it will be installed as part of the Charon emulator kit).

Replacing the Currently installed License Drivers

Replacement of the currently installed Sentinel Run-time can be needed in case of installation of specific run-time provided by Stromasys.
To proceed, remove the current version of Sentinel run-time:

- Login as Administrator
- Extract the content of the unique ZIP file residing in **hasp_install** directory of the Charon distribution to the same folder.
- Run **cmd.exe** from the Start menu of host computer
- **cd** to the **hasp_install** directory
- Issue the following command:
  ...> **`haspdinst.exe-fr -kp -nomsg`**

Install the new Sentinel run-time:

- **cd** to the directory where the target run-time resides
- Issue the following command:

  ...> **`haspdinst.exe -install -cm`**
- **cd** to the **hasp_install** directory
- Extract the contents of this archive `hasplib.zip` to the same directory.
- Copy the file `haspvlib_68704.dll`
  to `C:\Program Files (x86)\Common Files\Aladdin Shared\HASP` (in case of a x64 host) or
  `to C:\Program Files\Common Files\Aladdin Shared\HASP`" (in case of a x86 host).

CHARON-AXP for Windows utilities

General description

CHARON-AXP provides the following set of utilities:

Utility	Description
Host Device Check	Used to review system resources that can be mapped to CHARON.
Network Control Center	Used to configure a CHARON network.
MkDisk	GUI-based utility used to create custom or standard CHARON virtual disk containers.
MkDskCmd	Command line utility used to create custom or standard CHARON virtual disk containers. This utility also may be used to transfer virtual disks of one type to virtual disks of another type.
HASP View	Used to display the CHARON license content.
License Update Service	Used to manage CHARON licenses, collect the host system fingerprint and to transfer software licenses from one host to another.
License Expiration Check	Used to warn user about expiration of CHARON license
mtd	Used to create CHARON tape images from physical tapes and to write tape images back to physical tapes.
CHARON Log Monitor and Dispatcher	Used to trace CHARON log files and run a specific program on certain conditions.
HOSTPrint	Used to print CHARON output to Windows printers.
CHARON Guest Utilities for OpenVMS	Used to manage virtual tapes and CHARON performance.

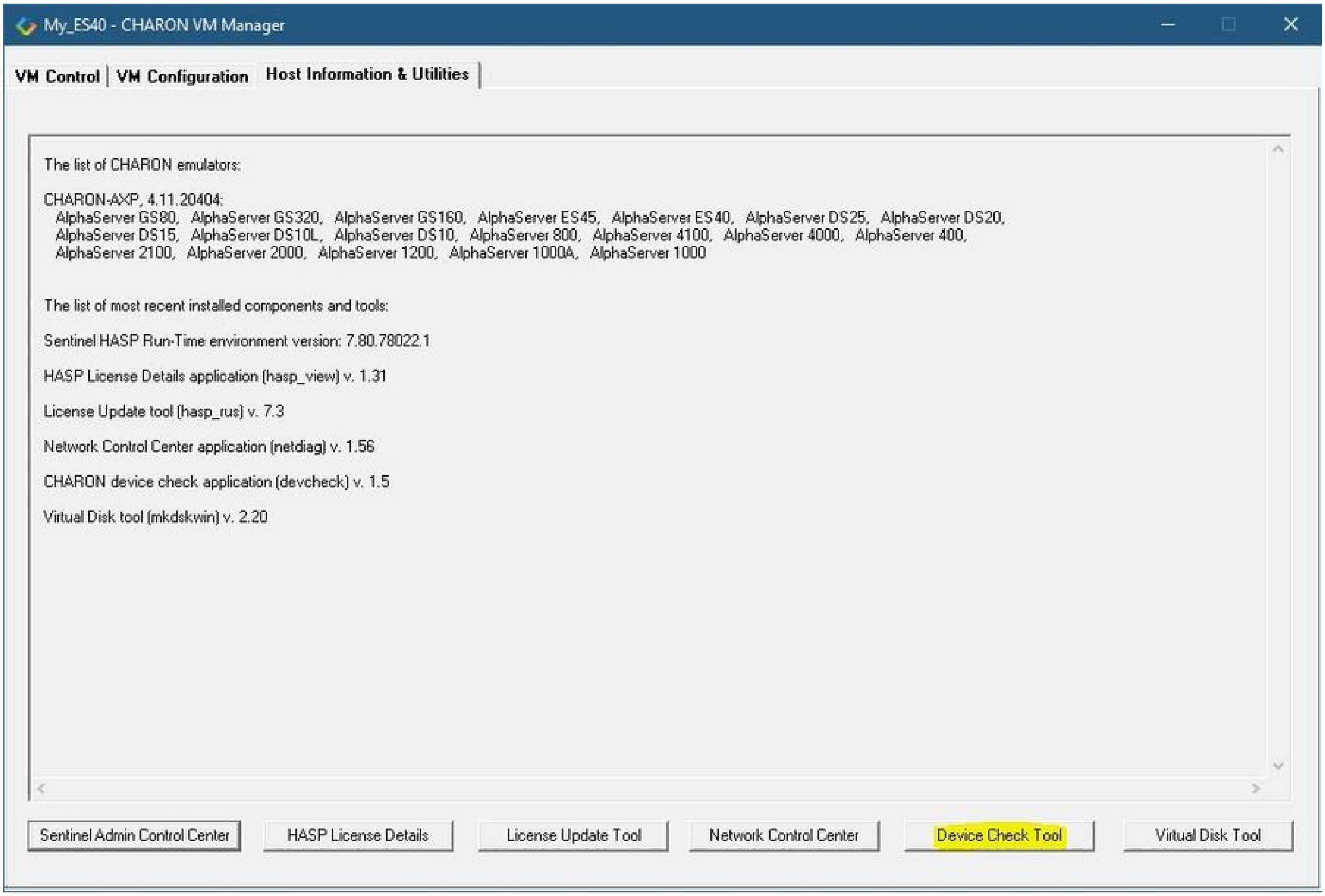
Host Device Check

Table of Contents

General Description

The "Host Device Check" utility is used to review system resources that can be mapped to CHARON.

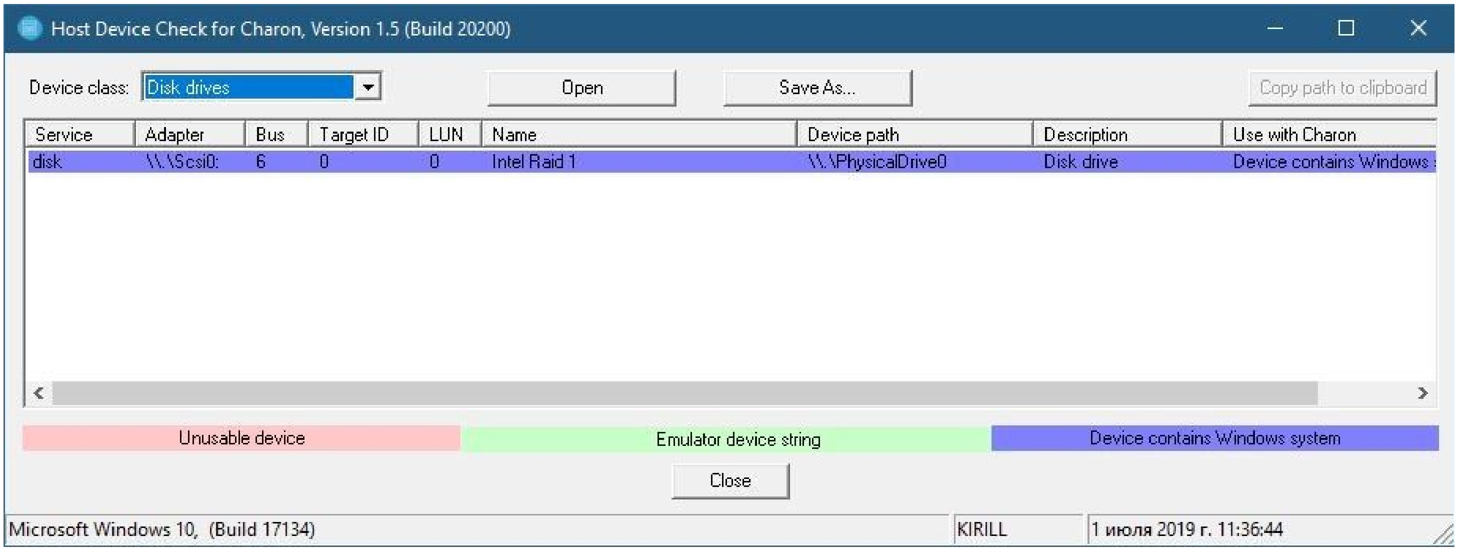
Open the CHARON Virtual Machine Manager, switch to the "Host Information & Utilities" tab and press the "Device Check Tool" button:



```
The list of CHARON emulators:

CHARON-AXP, 4.11.20404:
  AlphaServer GS80,  AlphaServer GS320,  AlphaServer GS160,  AlphaServer ES45,  AlphaServer ES40,  AlphaServer DS25,  AlphaServer DS20,
  AlphaServer DS15,  AlphaServer DS10L,  AlphaServer DS10,  AlphaServer 800,  AlphaServer 4100,  AlphaServer 4000,  AlphaServer 400,
  AlphaServer 2100,  AlphaServer 2000,  AlphaServer 1200,  AlphaServer 1000A,  AlphaServer 1000


The list of most recent installed components and tools:

Sentinel HASP Run-Time environment version: 7.80.78022.1

HASP License Details application (hasp_view) v. 1.31

License Update tool (hasp_rus) v. 7.3

Network Control Center application (netdiag) v. 1.56

CHARON device check application (devcheck) v. 1.5

Virtual Disk tool (mkdskwin) v. 2.20
```

Reviewing available physical disks

By default the "Host Device Check" utility reports the available physical disks:



Device class: Disk drives

Service	Adapter	Bus	Target ID	LUN	Name	Device path	Description	Use with Charon
disk	\\.\Scsi0:	6	0	0	Intel Raid 1	\\.\PhysicalDrive0	Disk drive	Device contains Windows :

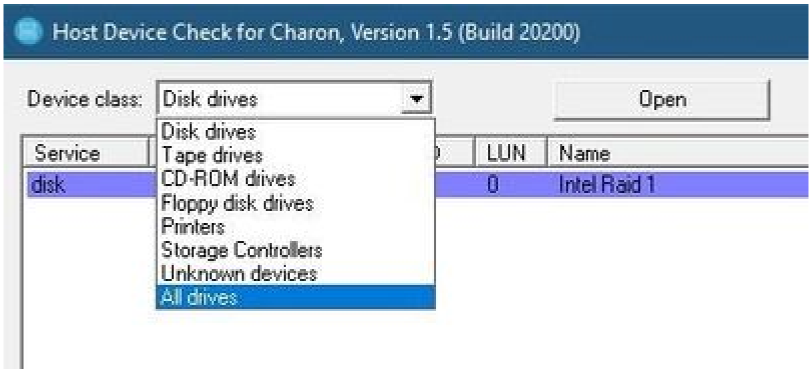
Unusable device | Emulator device string | Device contains Windows system

Microsoft Windows 10, (Build 17134) | KIRILL | 1 июля 2019 г. 11:36:44

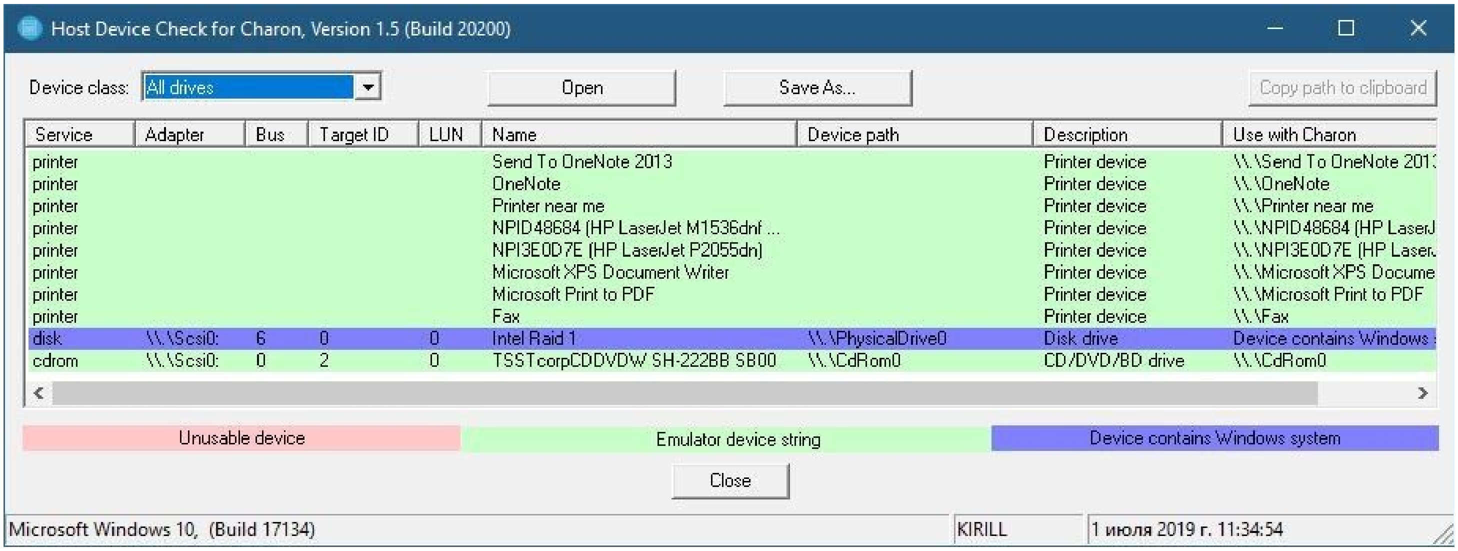
⚠ Please note that the unusable devices and the devices containing Windows system (see corresponding colors above) **must not** be mapped to CHARON!

Reviewing all the available host resources

Select "All drives" in the "Device class" drop-down list:



The "Host Device Check" utility will display all the host resources:


Host Device Check for Charon, Version 1.5 (Build 20200)
Open
Save As...
Copy path to clipboard
Unusable device
Emulator device string
Device contains Windows system
Close

Device class: All drives

Service	Adapter	Bus	Target ID	LUN	Name	Device path	Description	Use with Charon
printer					Send To OneNote 2013		Printer device	\\.\Send To OneNote 201
printer					OneNote		Printer device	\\.\OneNote
printer					Printer near me		Printer device	\\.\Printer near me
printer					NPID48684 (HP LaserJet M1536dnf ...		Printer device	\\.\NPID48684 (HP LaserJ
printer					NPI3E0D7E (HP LaserJet P2055dn)		Printer device	\\.\NPI3E0D7E (HP Laser
printer					Microsoft XPS Document Writer		Printer device	\\.\Microsoft XPS Docume
printer					Microsoft Print to PDF		Printer device	\\.\Microsoft Print to PDF
printer					Fax		Printer device	\\.\Fax
disk	\\.\Scsi0:	6	0	0	Intel Raid 1	\\.\PhysicalDrive0	Disk drive	Device contains Windows
cdrom	\\.\Scsi0:	0	2	0	TSSTcorpCDDVDW SH-222BB SB00	\\.\CdRom0	CD/DVD/BD drive	\\.\CdRom0

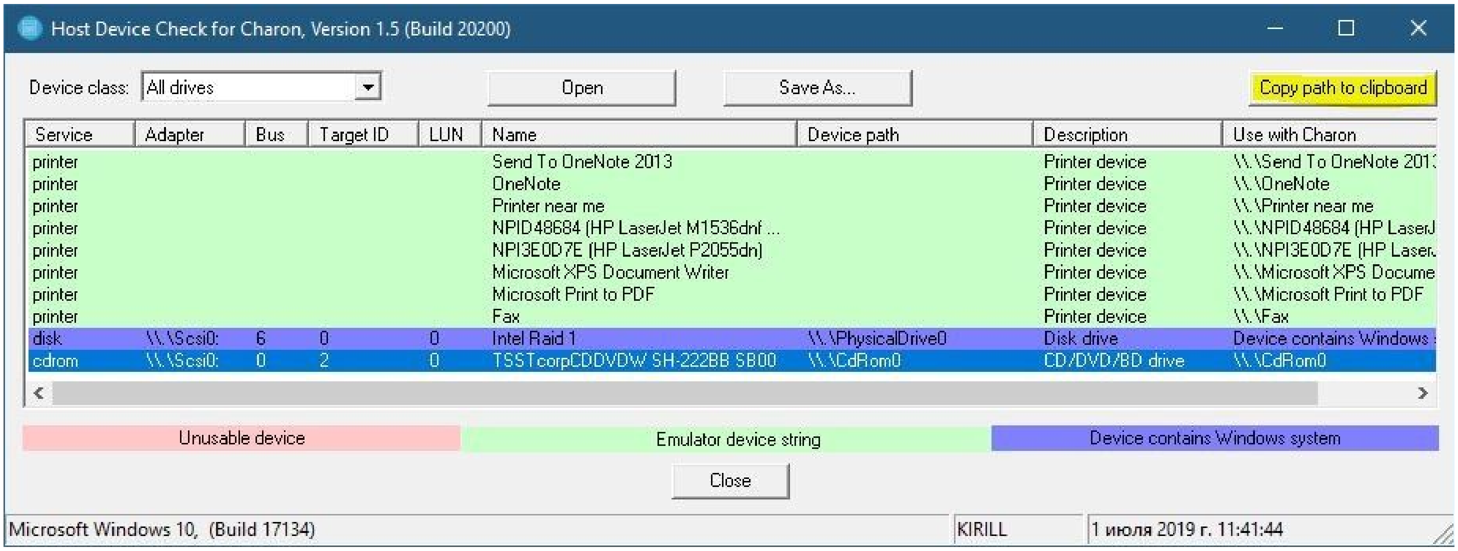
Microsoft Windows 10, (Build 17134) | KIRILL | 1 июля 2019 г. 11:34:54

It is also possible to choose all the other categories to narrow the possible mapping options list.

Note the "Use with CHARON" column contains the actual configuration options for each available device to be inserted, if needed, in the CHARON configuration file (see below).

Collecting the configuration strings

To collect the actual configuration strings to be used in the CHARON configuration file, select the target device and press the "Copy path to clipboard" button:


Host Device Check for Charon, Version 1.5 (Build 20200)
Open
Save As...
Copy path to clipboard
Unusable device
Emulator device string
Device contains Windows system
Close

Device class: All drives

Service	Adapter	Bus	Target ID	LUN	Name	Device path	Description	Use with Charon
printer					Send To OneNote 2013		Printer device	\\.\Send To OneNote 201
printer					OneNote		Printer device	\\.\OneNote
printer					Printer near me		Printer device	\\.\Printer near me
printer					NPID48684 (HP LaserJet M1536dnf ...		Printer device	\\.\NPID48684 (HP LaserJ
printer					NPI3E0D7E (HP LaserJet P2055dn)		Printer device	\\.\NPI3E0D7E (HP Laser
printer					Microsoft XPS Document Writer		Printer device	\\.\Microsoft XPS Docume
printer					Microsoft Print to PDF		Printer device	\\.\Microsoft Print to PDF
printer					Fax		Printer device	\\.\Fax
disk	\\.\Scsi0:	6	0	0	Intel Raid 1	\\.\PhysicalDrive0	Disk drive	Device contains Windows
cdrom	\\.\Scsi0:	0	2	0	TSSTcorpCDDVDW SH-222BB SB00	\\.\CdRom0	CD/DVD/BD drive	\\.\CdRom0

Microsoft Windows 10, (Build 17134) | KIRILL | 1 июля 2019 г. 11:41:44

The selected configuration string will be copied to the clipboard, it can then be pasted to the CHARON configuration file (using the "CTRL-V" keys combination for example).

The buttons "Open" and "Save As..." help to open up the displayed options in form of text file and save this text file under some given name.

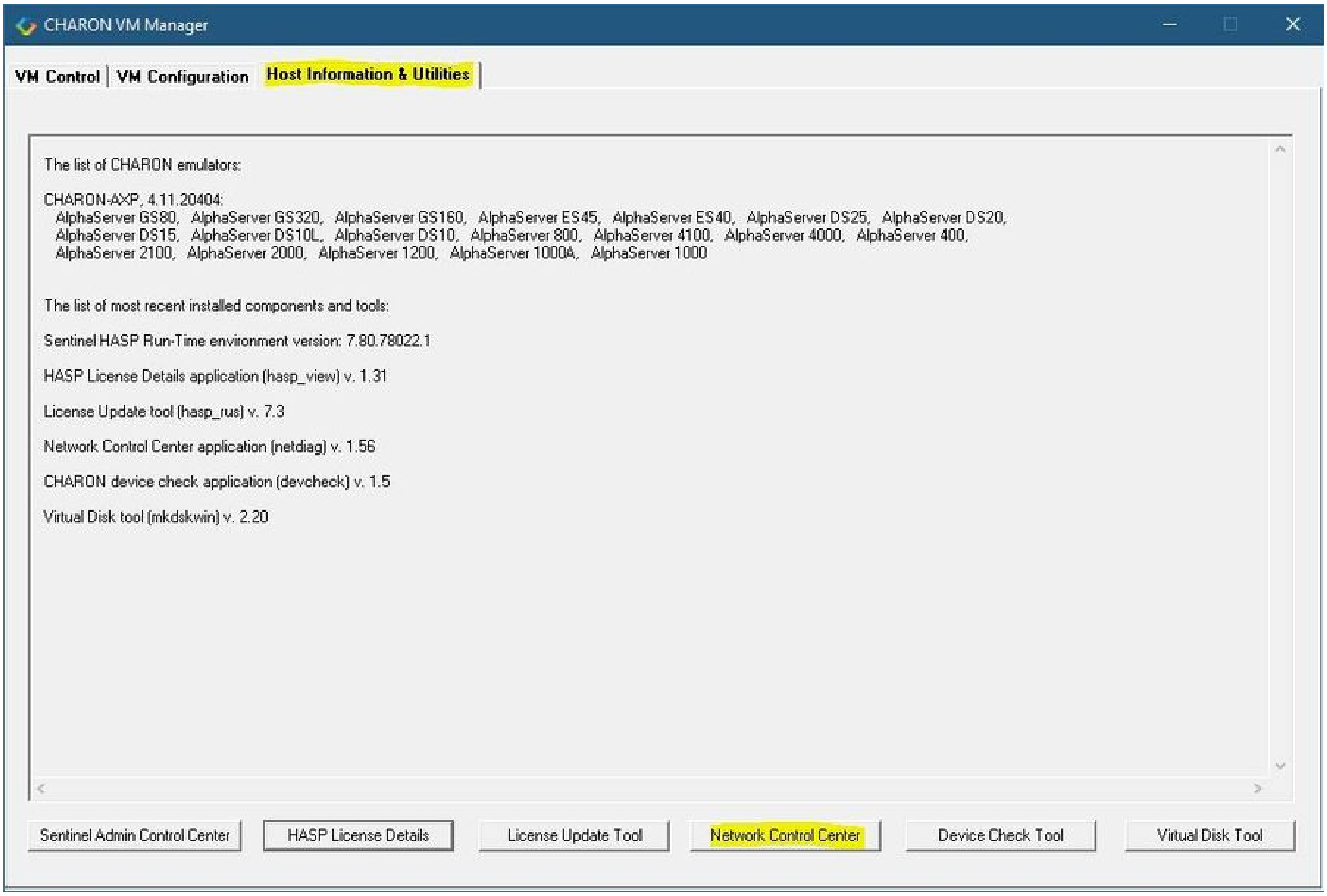
Network Control Center

Table of Contents

General Description

The "Network Control Center" utility is used to configure, verify dedicated network interfaces and trace the network activity for a CHARON network.

Open the CHARON Virtual Machine Manager, switch to the "Host Information & Utilities" tab and press the "Network Control Center" button:



The list of CHARON emulators:

CHARON-AXP, 4.11.20404:
AlphaServer GS80, AlphaServer GS320, AlphaServer GS160, AlphaServer ES45, AlphaServer ES40, AlphaServer DS25, AlphaServer DS20, AlphaServer DS15, AlphaServer DS10L, AlphaServer DS10, AlphaServer 800, AlphaServer 4100, AlphaServer 4000, AlphaServer 400, AlphaServer 2100, AlphaServer 2000, AlphaServer 1200, AlphaServer 1000A, AlphaServer 1000

The list of most recent installed components and tools:

Sentinel HASP Run-Time environment version: 7.80.78022.1

HASP License Details application (hasp_view) v. 1.31

License Update tool (hasp_rus) v. 7.3

Network Control Center application (netdiag) v. 1.56

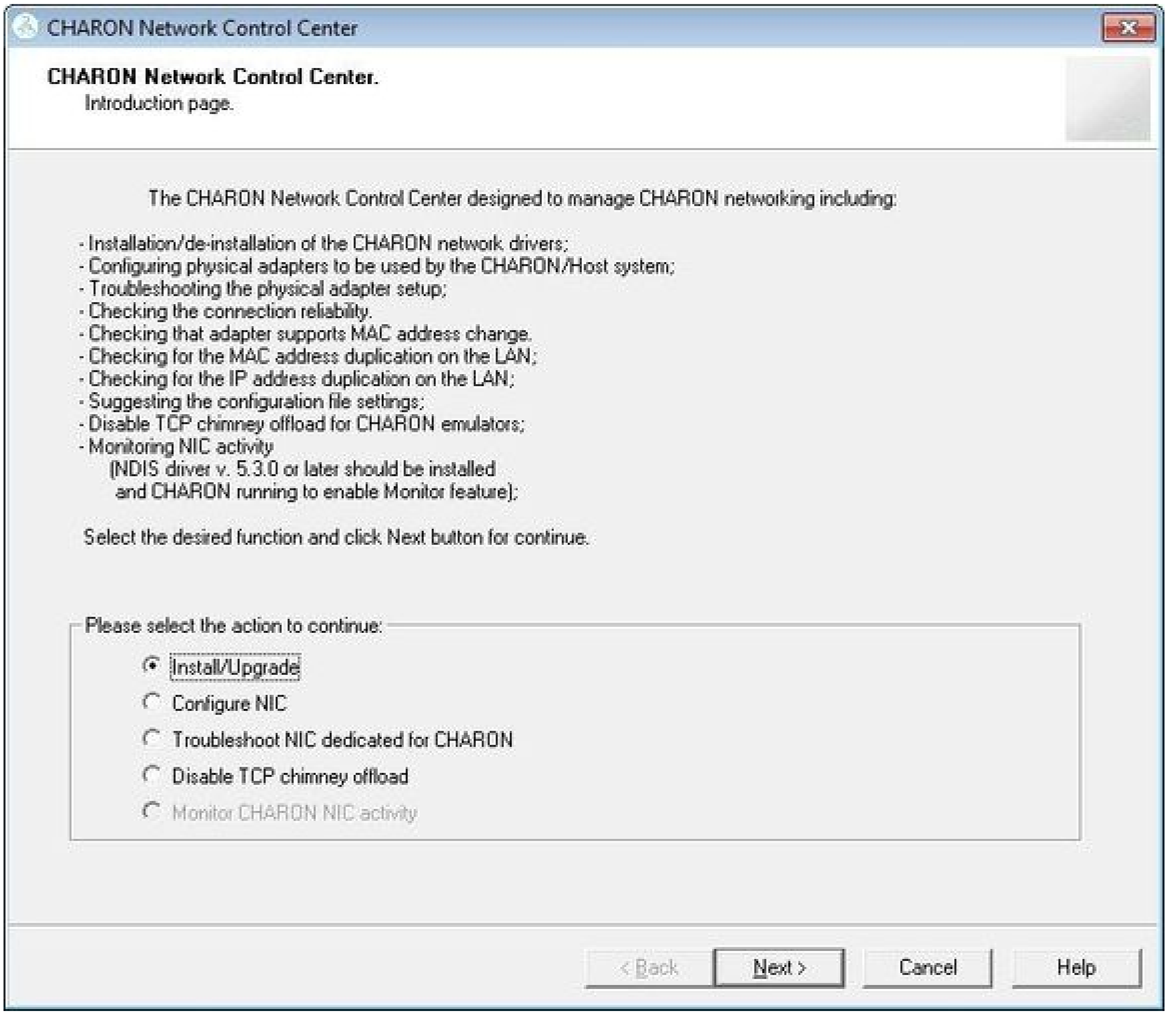
CHARON device check application (devcheck) v. 1.5

Virtual Disk tool (mkdskwin) v. 2.20

Installation of CHARON network drivers

By default a CHARON network driver is automatically installed by the CHARON installation procedure.

When it is needed to install a modified driver, use the following procedure. Select "Install/Upgrade" and press the "Next" button:



CHARON Network Control Center.

Introduction page.

The CHARON Network Control Center designed to manage CHARON networking including:

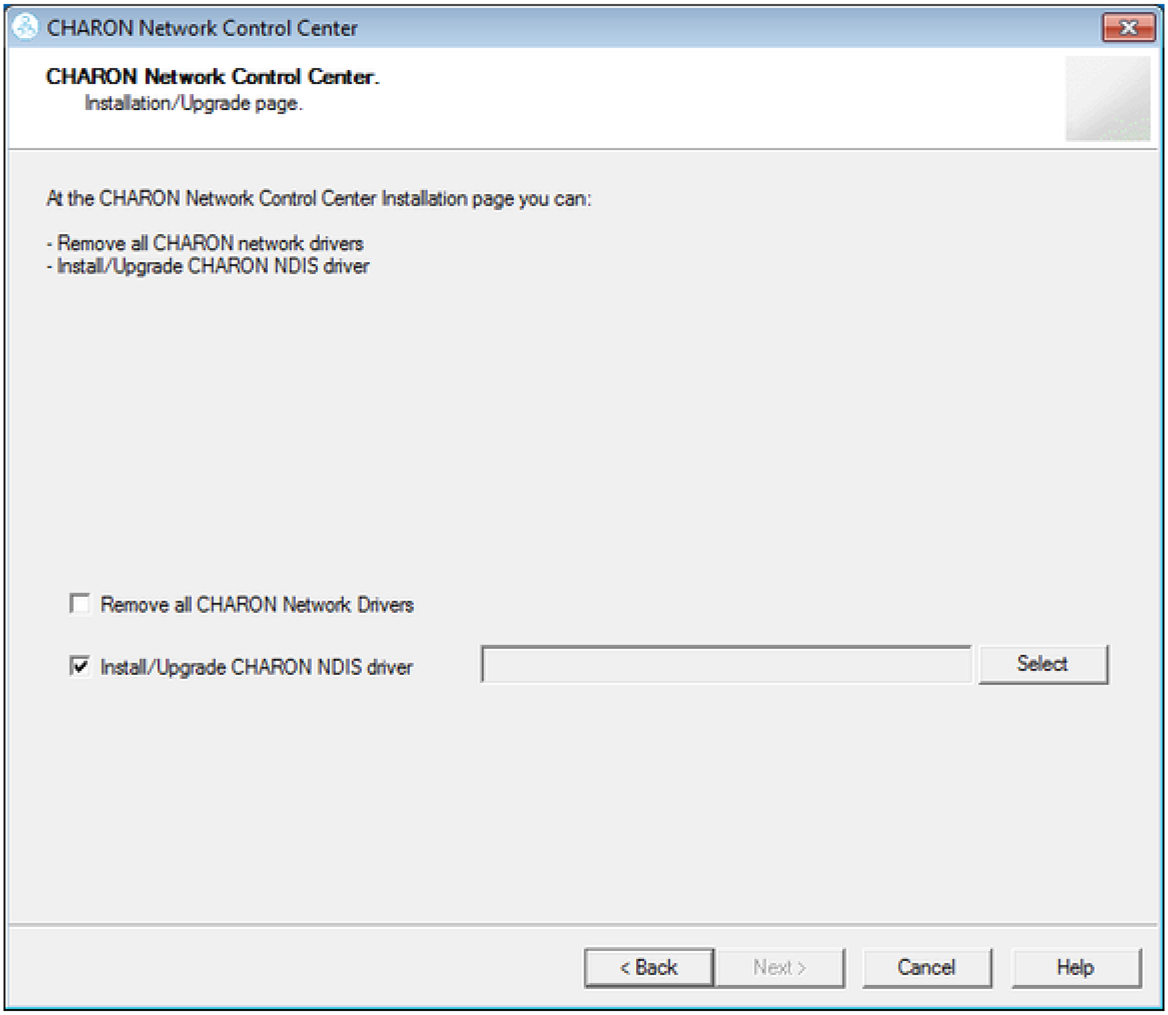
- Installation/de-installation of the CHARON network drivers;
- Configuring physical adapters to be used by the CHARON/Host system;
- Troubleshooting the physical adapter setup;
- Checking the connection reliability.
- Checking that adapter supports MAC address change.
- Checking for the MAC address duplication on the LAN;
- Checking for the IP address duplication on the LAN;
- Suggesting the configuration file settings;
- Disable TCP chimney offload for CHARON emulators;
- Monitoring NIC activity
  (NDIS driver v. 5.3.0 or later should be installed
  and CHARON running to enable Monitor feature);

Select the desired function and click Next button for continue.

Please select the action to continue:

● Install/Upgrade
○ Configure NIC
○ Troubleshoot NIC dedicated for CHARON
○ Disable TCP chimney offload
○ Monitor CHARON NIC activity

Select "Install/Upgrade CHARON NDIS driver" checkbox and press the "Select" button:



CHARON Network Control Center

CHARON Network Control Center.
Installation/Upgrade page.

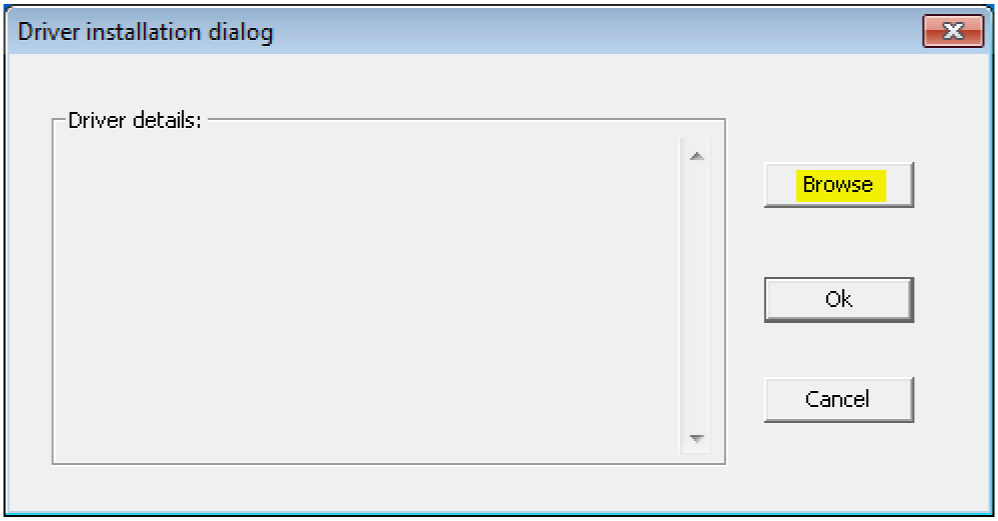
At the CHARON Network Control Center Installation page you can:

- Remove all CHARON network drivers
- Install/Upgrade CHARON NDIS driver

☐ Remove all CHARON Network Drivers

☑ Install/Upgrade CHARON NDIS driver

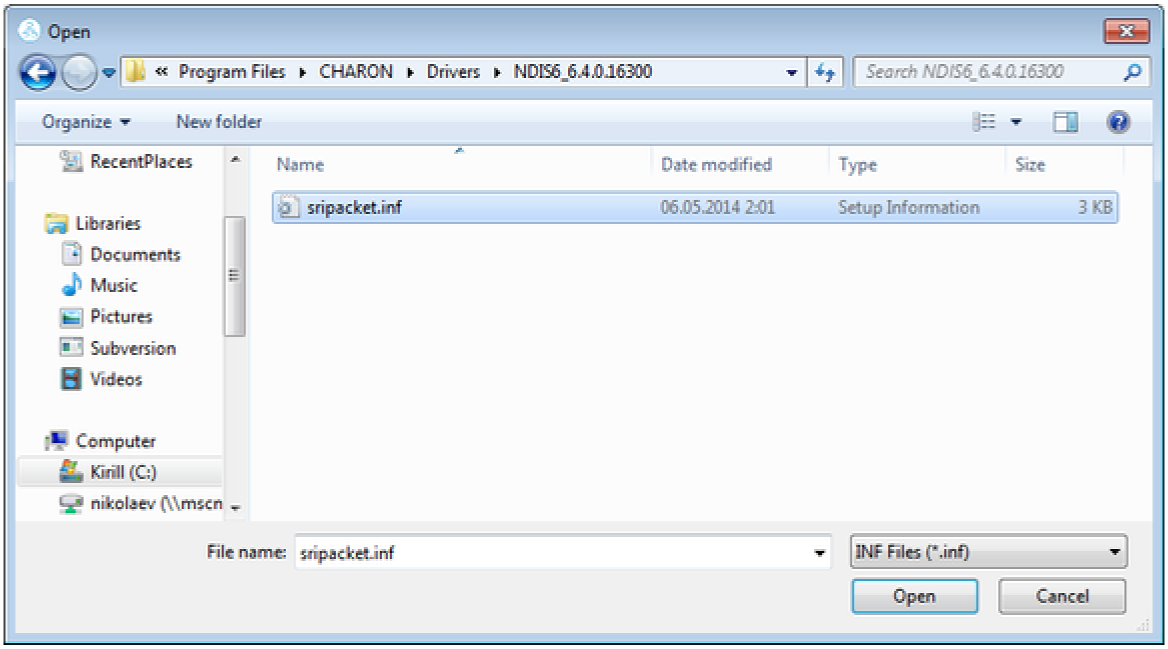
Press the "Browse" button:



Driver installation dialog

Driver details:

Browse for the target "`sripacket.inf`" file which is by default located in the "`C:\Program Files\CHARON\Drivers\NDIS6_`*`X.X.X.XXXXX`*" directory, select it and press the "Open" button:



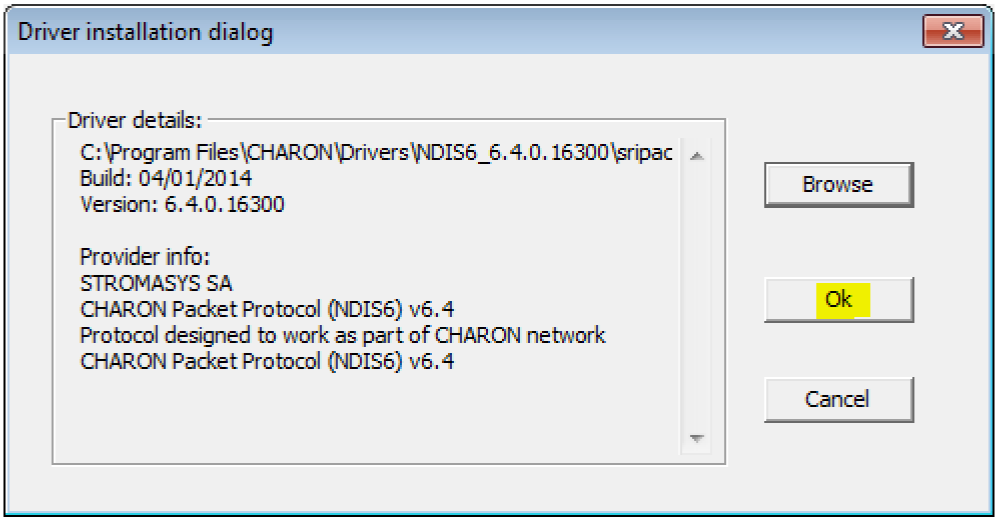
Name	Date modified	Type	Size
sripacket.inf	06.05.2014 2:01	Setup Information	3 KB

File name: sripacket.inf

INF Files (*.inf)

If the CHARON network driver has been acquired directly from STROMASYS, put it in a temporary directory and choose the "`sripacket.inf`" file from this directory in the dialog above.

Review the version of the driver and, if it is correct, press the "Ok" button:

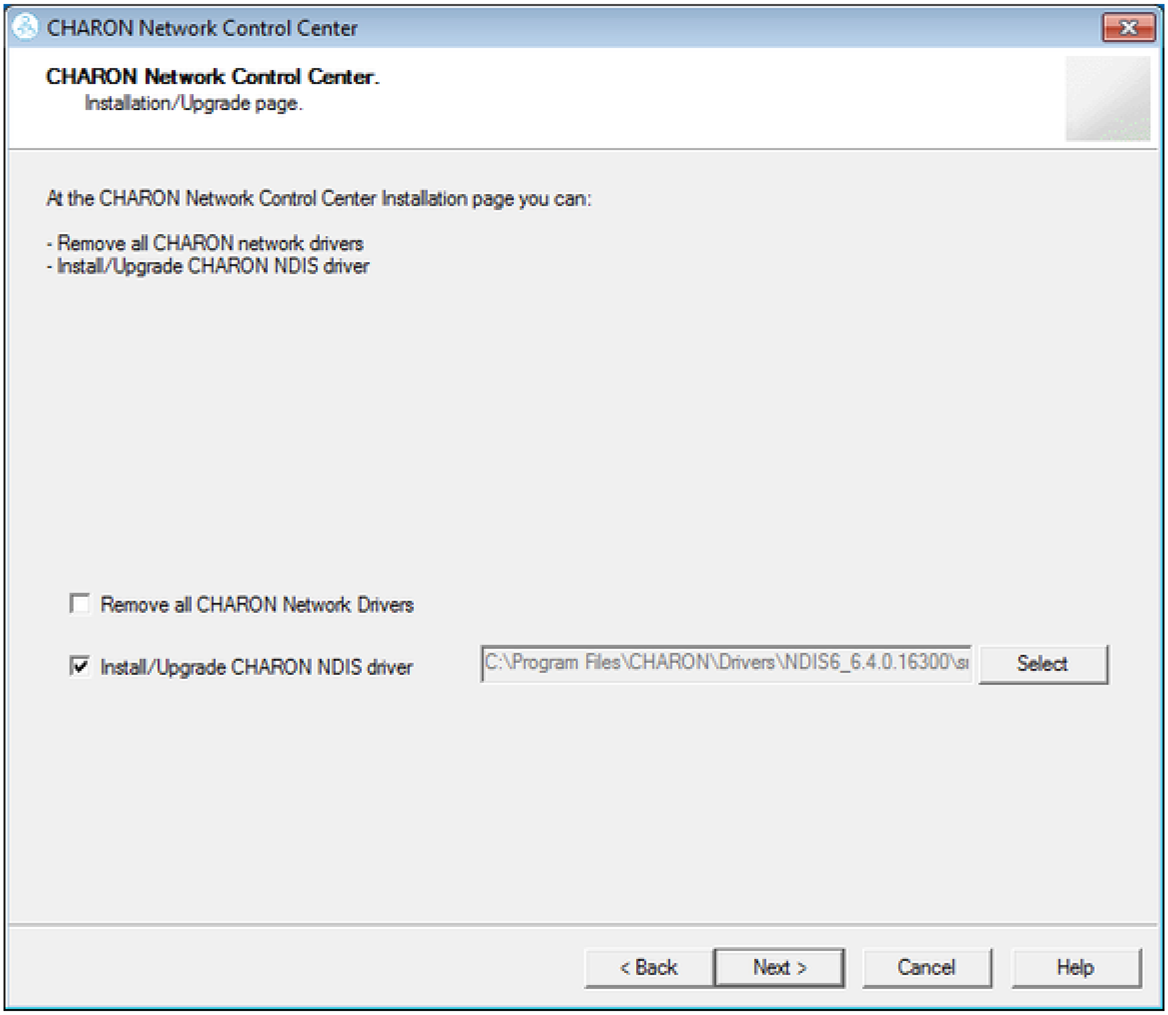


Driver details:

C:\Program Files\CHARON\Drivers\NDIS6_6.4.0.16300\sripac
Build: 04/01/2014
Version: 6.4.0.16300

Provider info:
STROMASYS SA
CHARON Packet Protocol (NDIS6) v6.4
Protocol designed to work as part of CHARON network
CHARON Packet Protocol (NDIS6) v6.4

The Network Control Center will display the previous dialog with the path to the CHARON network driver. Press the "Next" button in the popup below to continue:



CHARON Network Control Center

CHARON Network Control Center.
Installation/Upgrade page.

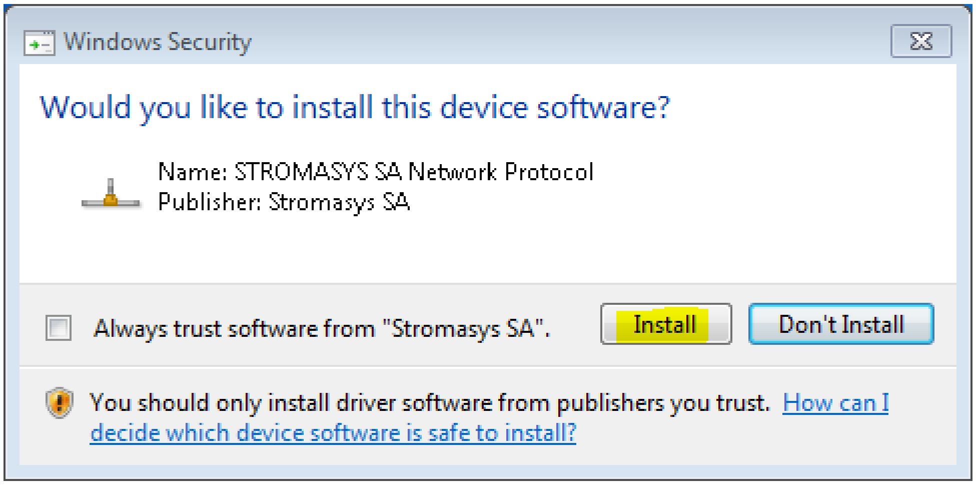
At the CHARON Network Control Center Installation page you can:

- Remove all CHARON network drivers
- Install/Upgrade CHARON NDIS driver

☐ Remove all CHARON Network Drivers

☑ Install/Upgrade CHARON NDIS driver   C:\Program Files\CHARON\Drivers\NDIS6_6.4.0.16300\s

The utility will begin the CHARON network driver installation. If Windows Security asks you to confirm the driver installation, press the "Install" button:



Windows Security

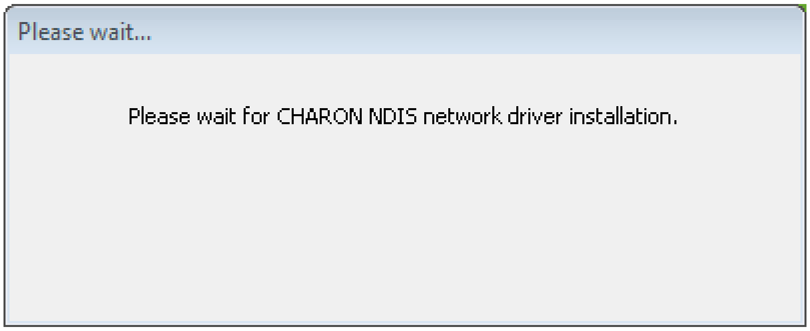
Would you like to install this device software?

Name: STROMASYS SA Network Protocol
Publisher: Stromasys SA

☐ Always trust software from "Stromasys SA".

You should only install driver software from publishers you trust. How can I decide which device software is safe to install?

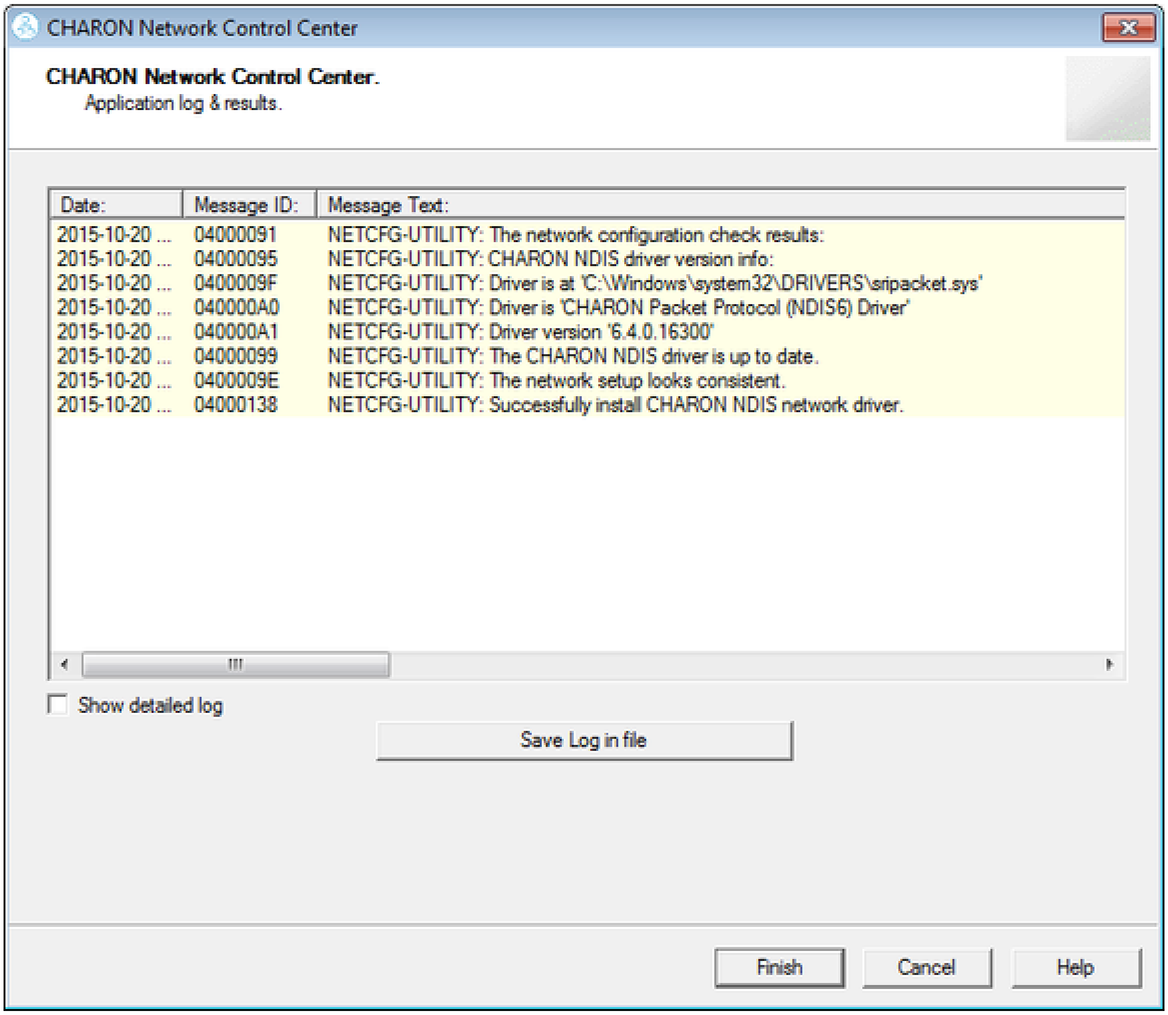
The driver installation will resume:



Please wait...

Please wait for CHARON NDIS network driver installation.

At the completion of the network installation procedure, the utility will display the following log:



CHARON Network Control Center
CHARON Network Control Center.
Application log & results.

```
Date:           Message ID:   Message Text:
2015-10-20 ...  04000091      NETCFG-UTILITY: The network configuration check results:
2015-10-20 ...  04000095      NETCFG-UTILITY: CHARON NDIS driver version info:
2015-10-20 ...  0400009F      NETCFG-UTILITY: Driver is at 'C:\Windows\system32\DRIVERS\sripacket.sys'
2015-10-20 ...  040000A0      NETCFG-UTILITY: Driver is 'CHARON Packet Protocol (NDIS6) Driver'
2015-10-20 ...  040000A1      NETCFG-UTILITY: Driver version '6.4.0.16300'
2015-10-20 ...  04000099      NETCFG-UTILITY: The CHARON NDIS driver is up to date.
2015-10-20 ...  0400009E      NETCFG-UTILITY: The network setup looks consistent.
2015-10-20 ...  04000138      NETCFG-UTILITY: Successfully install CHARON NDIS network driver.
```

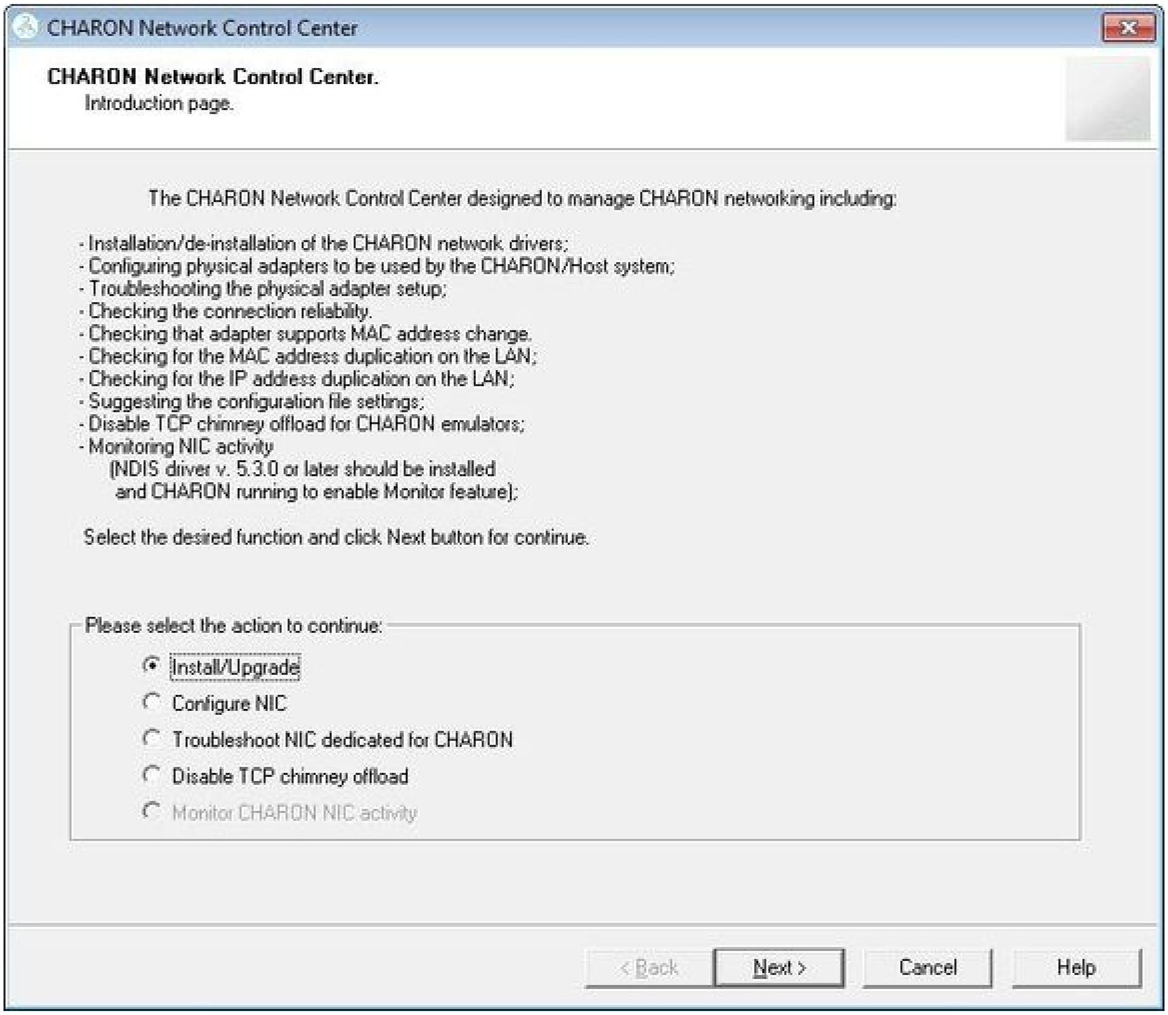
Review the log, make sure it is correct. You can save the log to a file by pressing the "Save Log in file" button in case the log file has to be sent to Stromasys Customer Support.

Select "Show detailed log" to display more detail.

Press the "Finish" button to exit.

Deinstallation of the CHARON network driver

Start the utility and select "Install/Upgrade"; press the "Next" button:



CHARON Network Control Center.
Introduction page.

The CHARON Network Control Center designed to manage CHARON networking including:

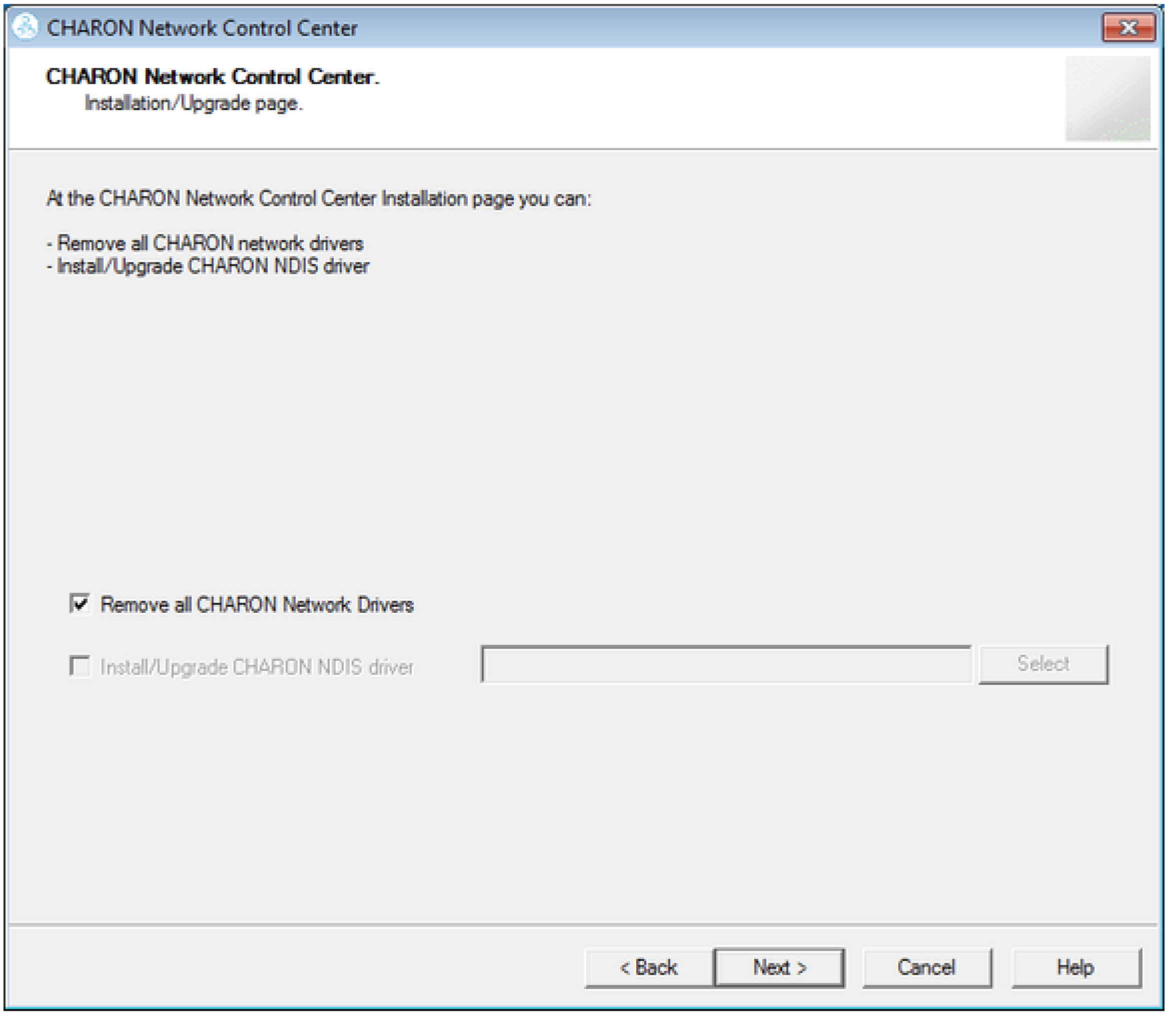
- Installation/de-installation of the CHARON network drivers;
- Configuring physical adapters to be used by the CHARON/Host system;
- Troubleshooting the physical adapter setup;
- Checking the connection reliability.
- Checking that adapter supports MAC address change.
- Checking for the MAC address duplication on the LAN;
- Checking for the IP address duplication on the LAN;
- Suggesting the configuration file settings;
- Disable TCP chimney offload for CHARON emulators;
- Monitoring NIC activity
  (NDIS driver v. 5.3.0 or later should be installed
  and CHARON running to enable Monitor feature);

Select the desired function and click Next button for continue.

Please select the action to continue:

● Install/Upgrade
○ Configure NIC
○ Troubleshoot NIC dedicated for CHARON
○ Disable TCP chimney offload
○ Monitor CHARON NIC activity

Select "Remove all CHARON Network Drivers" and press the "Next" button:



CHARON Network Control Center

CHARON Network Control Center.
Installation/Upgrade page.

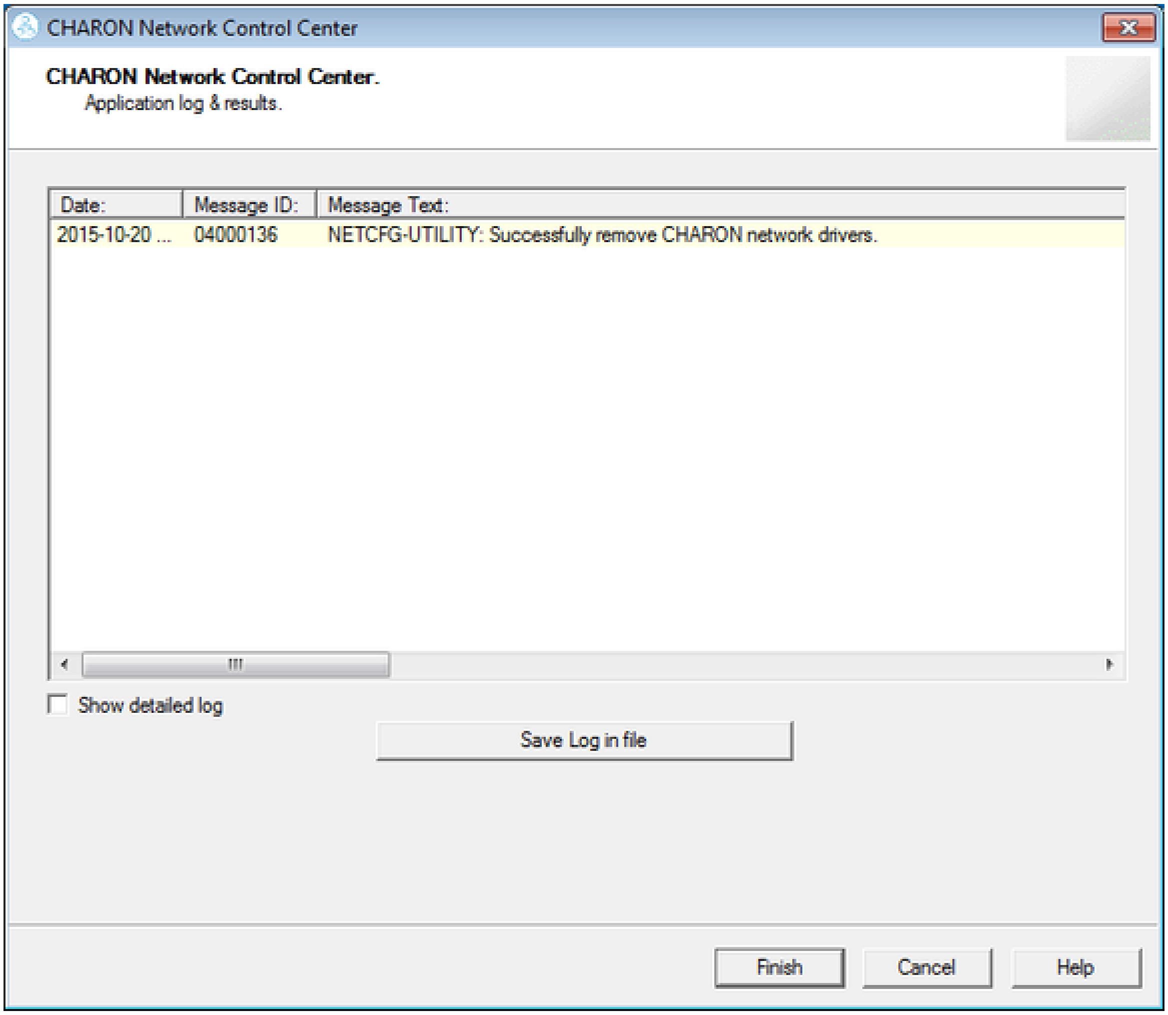
At the CHARON Network Control Center Installation page you can:

- Remove all CHARON network drivers
- Install/Upgrade CHARON NDIS driver

☑ Remove all CHARON Network Drivers

☐ Install/Upgrade CHARON NDIS driver

The utility will report that the CHARON drivers have been removed:

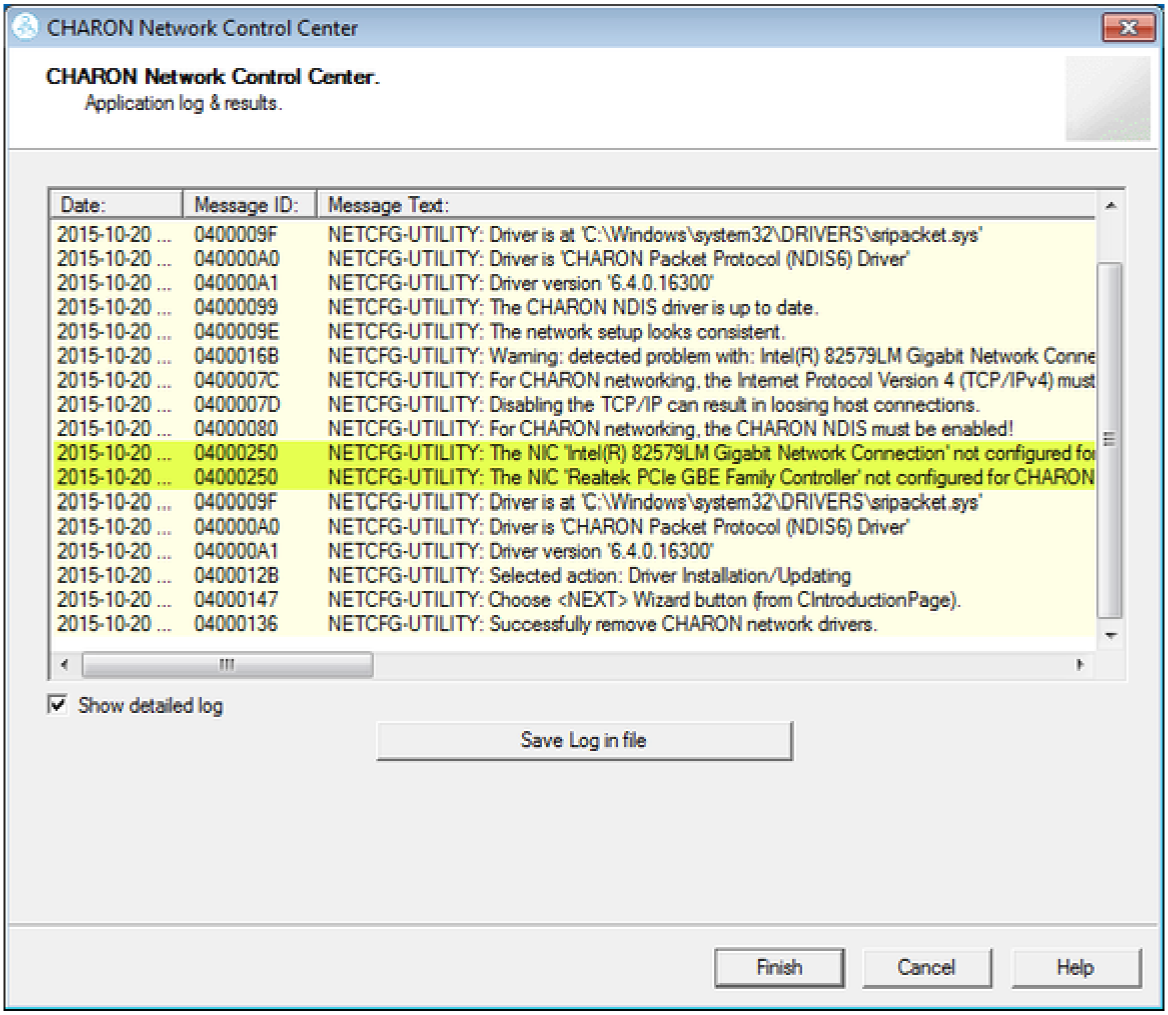

CHARON Network Control Center
CHARON Network Control Center.
Application log & results.
Show detailed log
Save Log in file
Finish
Cancel
Help

Date:	Message ID:	Message Text:
2015-10-20 ...	04000136	NETCFG-UTILITY: Successfully remove CHARON network drivers.

□ Show detailed log

Review the log for errors. You can save the log to a file by pressing the "Save Log in file" button in case the log file has to be sent to Stromasys Customer Support.

Select "Show detailed log" to display more detail.



CHARON Network Control Center

CHARON Network Control Center.
Application log & results.

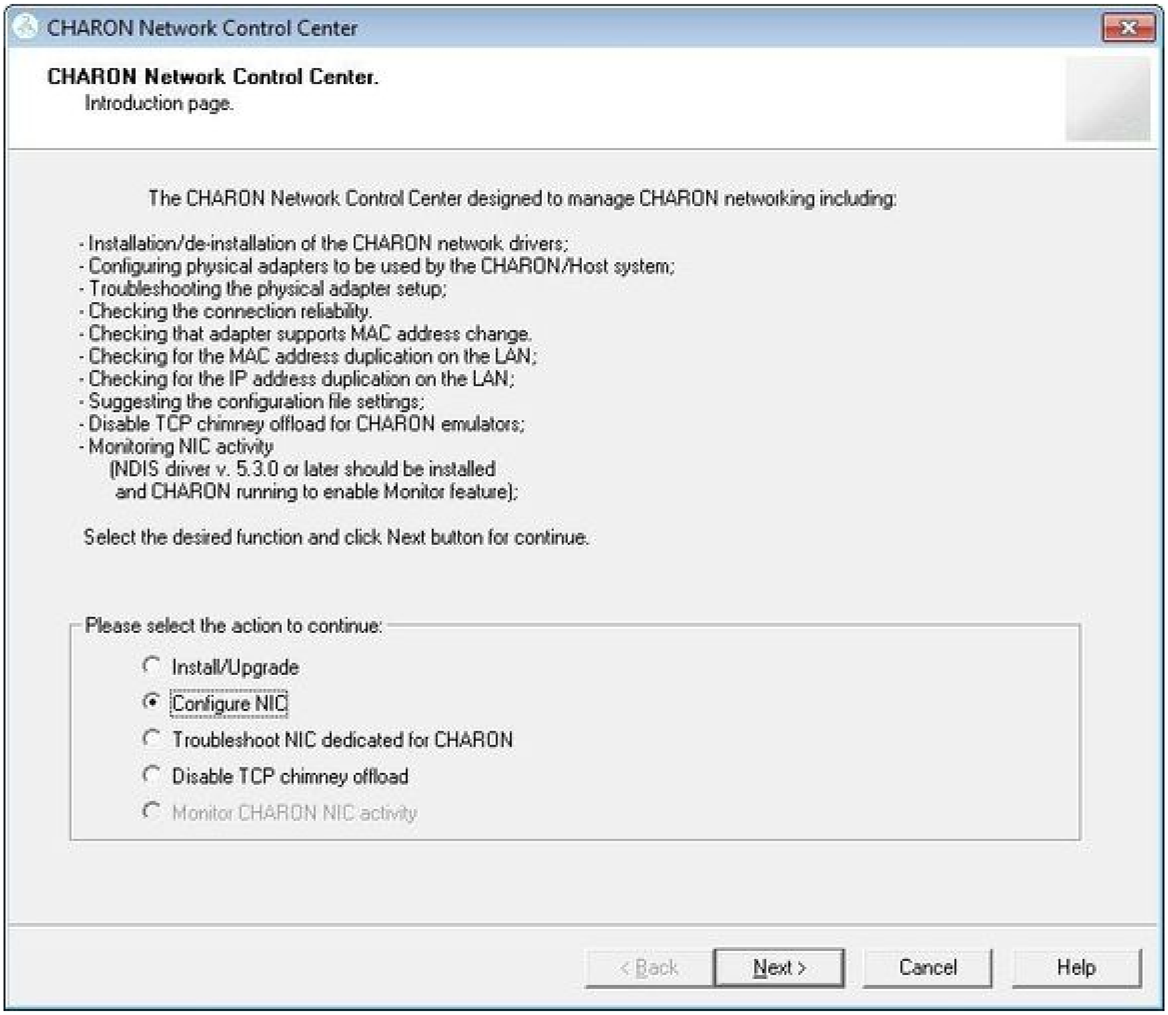
```
Date:            Message ID:   Message Text:
2015-10-20 ...   0400009F      NETCFG-UTILITY: Driver is at 'C:\Windows\system32\DRIVERS\sripacket.sys'
2015-10-20 ...   040000A0      NETCFG-UTILITY: Driver is 'CHARON Packet Protocol (NDIS6) Driver'
2015-10-20 ...   040000A1      NETCFG-UTILITY: Driver version '6.4.0.16300'
2015-10-20 ...   04000099      NETCFG-UTILITY: The CHARON NDIS driver is up to date.
2015-10-20 ...   0400009E      NETCFG-UTILITY: The network setup looks consistent.
2015-10-20 ...   0400016B      NETCFG-UTILITY: Warning: detected problem with: Intel(R) 82579LM Gigabit Network Conne
2015-10-20 ...   0400007C      NETCFG-UTILITY: For CHARON networking, the Internet Protocol Version 4 (TCP/IPv4) must
2015-10-20 ...   0400007D      NETCFG-UTILITY: Disabling the TCP/IP can result in loosing host connections.
2015-10-20 ...   04000080      NETCFG-UTILITY: For CHARON networking, the CHARON NDIS must be enabled!
2015-10-20 ...   04000250      NETCFG-UTILITY: The NIC 'Intel(R) 82579LM Gigabit Network Connection' not configured for
2015-10-20 ...   04000250      NETCFG-UTILITY: The NIC 'Realtek PCIe GBE Family Controller' not configured for CHARON
2015-10-20 ...   0400009F      NETCFG-UTILITY: Driver is at 'C:\Windows\system32\DRIVERS\sripacket.sys'
2015-10-20 ...   040000A0      NETCFG-UTILITY: Driver is 'CHARON Packet Protocol (NDIS6) Driver'
2015-10-20 ...   040000A1      NETCFG-UTILITY: Driver version '6.4.0.16300'
2015-10-20 ...   0400012B      NETCFG-UTILITY: Selected action: Driver Installation/Updating
2015-10-20 ...   04000147      NETCFG-UTILITY: Choose <NEXT> Wizard button (from CIntroductionPage).
2015-10-20 ...   04000136      NETCFG-UTILITY: Successfully remove CHARON network drivers.
```

☑ Show detailed log

Press the "Finish" button to exit.

Configuring the host network interfaces for CHARON

Start the utility, select "Configure NIC" and press the "Next" button:



CHARON Network Control Center.
Introduction page.

The CHARON Network Control Center designed to manage CHARON networking including:

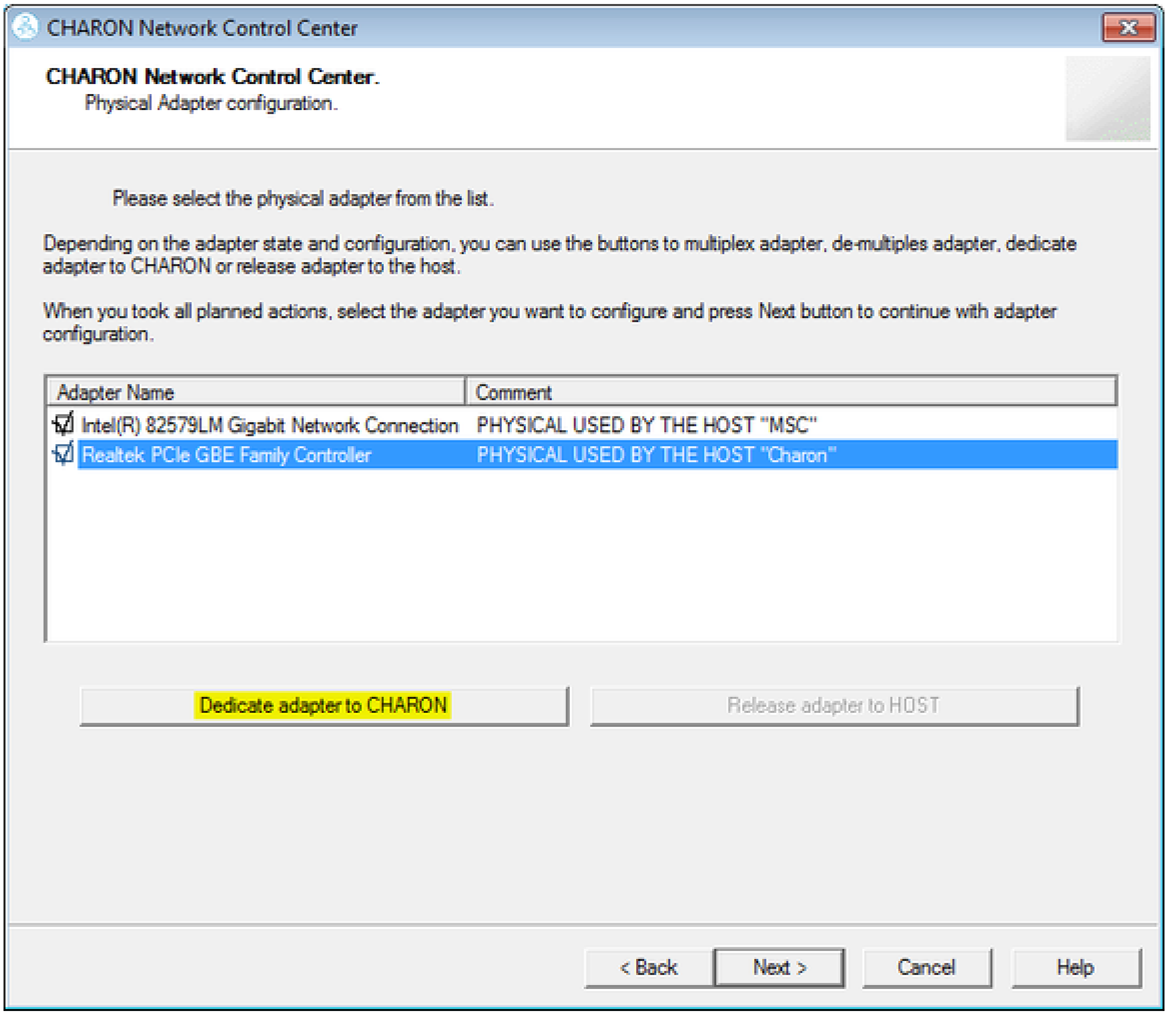
- Installation/de-installation of the CHARON network drivers;
- Configuring physical adapters to be used by the CHARON/Host system;
- Troubleshooting the physical adapter setup;
- Checking the connection reliability.
- Checking that adapter supports MAC address change.
- Checking for the MAC address duplication on the LAN;
- Checking for the IP address duplication on the LAN;
- Suggesting the configuration file settings;
- Disable TCP chimney offload for CHARON emulators;
- Monitoring NIC activity
  (NDIS driver v. 5.3.0 or later should be installed
  and CHARON running to enable Monitor feature);

Select the desired function and click Next button for continue.

Please select the action to continue:

○ Install/Upgrade
● Configure NIC
○ Troubleshoot NIC dedicated for CHARON
○ Disable TCP chimney offload
○ Monitor CHARON NIC activity

Select the host interface to be dedicated to CHARON (in the example below its name is "Charon"), press the "Dedicate adapter to CHARON" button:



CHARON Network Control Center

CHARON Network Control Center.

Physical Adapter configuration.

Please select the physical adapter from the list.

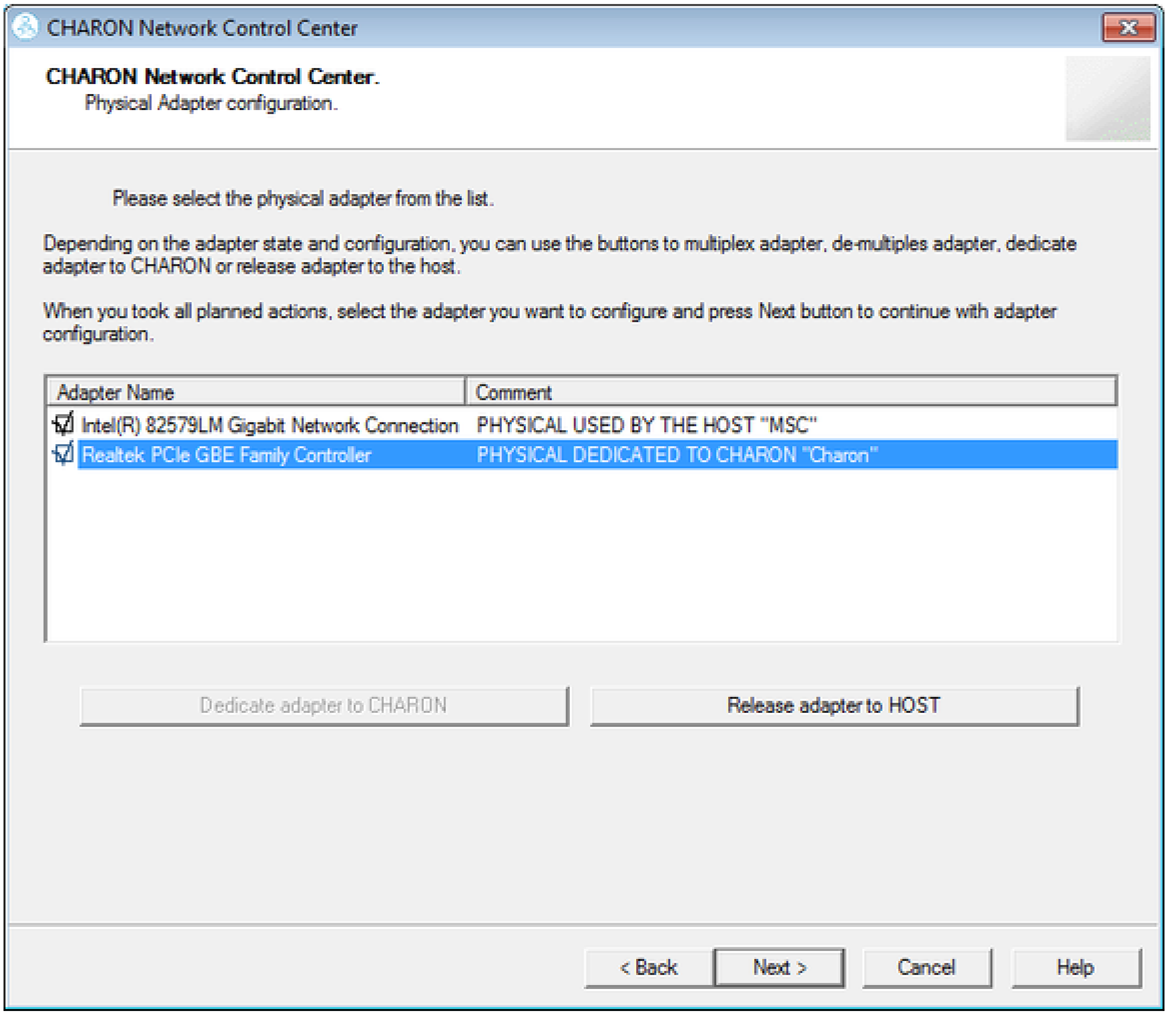
Depending on the adapter state and configuration, you can use the buttons to multiplex adapter, de-multiples adapter, dedicate adapter to CHARON or release adapter to the host.

When you took all planned actions, select the adapter you want to configure and press Next button to continue with adapter configuration.

Adapter Name	Comment
☑ Intel(R) 82579LM Gigabit Network Connection	PHYSICAL USED BY THE HOST "MSC"
☑ Realtek PCIe GBE Family Controller	PHYSICAL USED BY THE HOST "Charon"

Install and configure VLAN adapters, according to the vendor's User's Guide, if required. Select the VLAN adapter in the dialog box in the above example. A VLAN adapter is not configured differently, the same procedure should be followed.

The Network Control Center will dedicate the selected adapter to CHARON. Press the "Next" button.



CHARON Network Control Center.
Physical Adapter configuration.

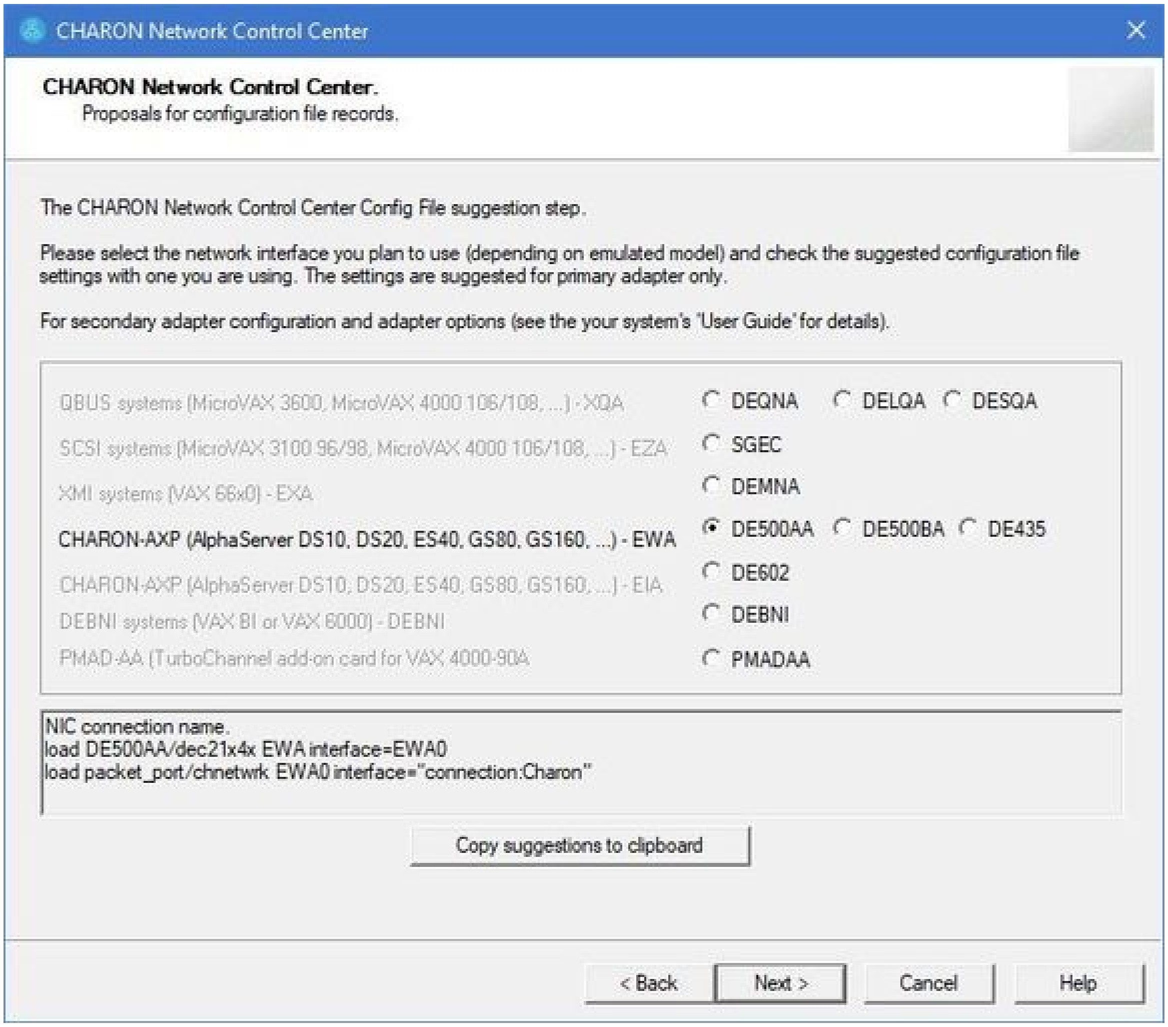
Please select the physical adapter from the list.

Depending on the adapter state and configuration, you can use the buttons to multiplex adapter, de-multiples adapter, dedicate adapter to CHARON or release adapter to the host.

When you took all planned actions, select the adapter you want to configure and press Next button to continue with adapter configuration.

Adapter Name	Comment
☑ Intel(R) 82579LM Gigabit Network Connection	PHYSICAL USED BY THE HOST "MSC"
☑ Realtek PCIe GBE Family Controller	PHYSICAL DEDICATED TO CHARON "Charon"

The next dialog helps to define the correct lines for the CHARON configuration file. Select the target emulated network adapter and press the "Copy suggestions to clipboard" button:



CHARON Network Control Center

CHARON Network Control Center.

Proposals for configuration file records.

The CHARON Network Control Center Config File suggestion step.

Please select the network interface you plan to use (depending on emulated model) and check the suggested configuration file settings with one you are using. The settings are suggested for primary adapter only.

For secondary adapter configuration and adapter options (see the your system's 'User Guide' for details).

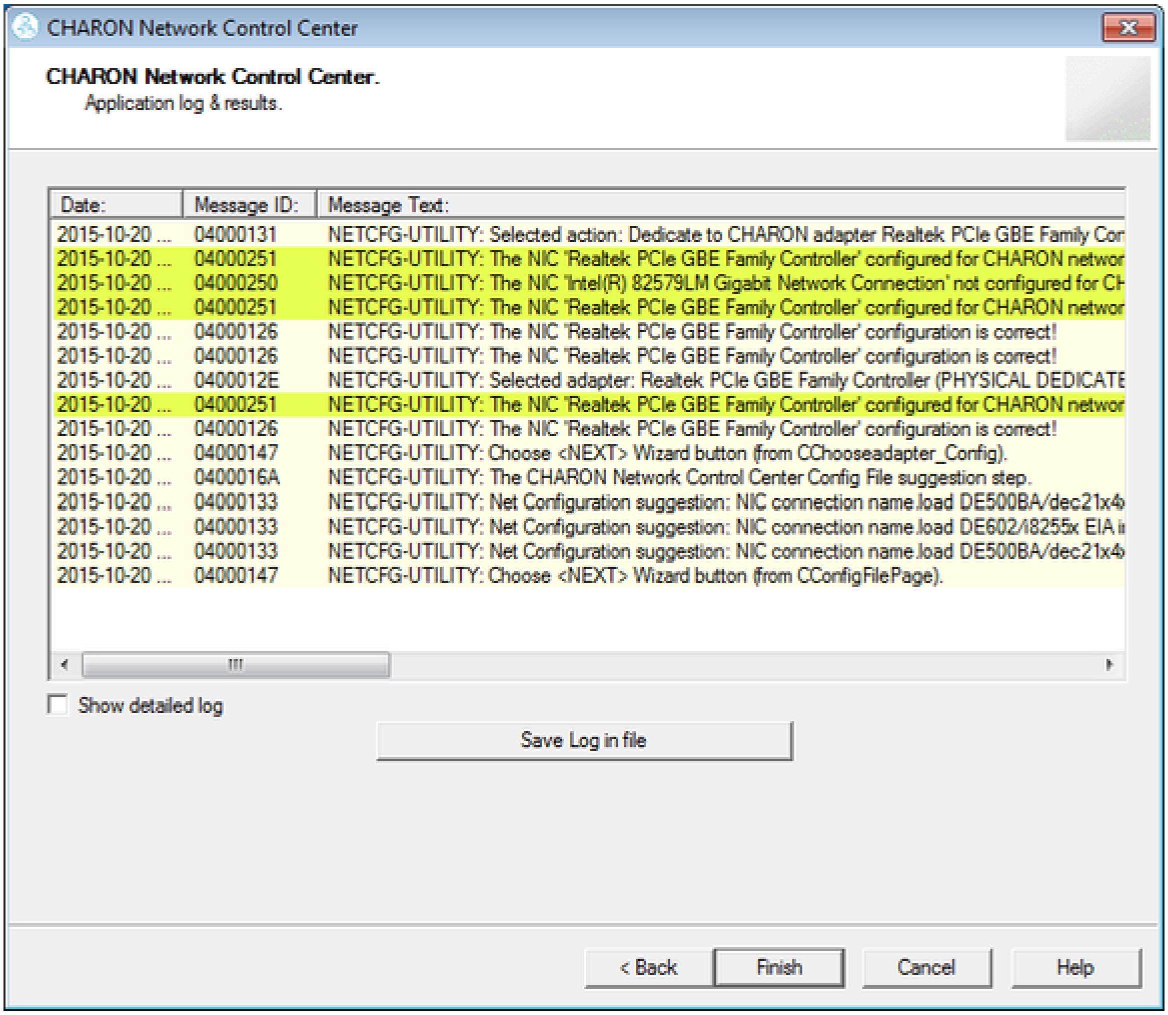
- QBUS systems (MicroVAX 3600, MicroVAX 4000 106/108, ...) - XQA ○ DEQNA ○ DELQA ○ DESQA
- SCSI systems (MicroVAX 3100 96/98, MicroVAX 4000 106/108, ...) - EZA ○ SGEC
- XMI systems (VAX 66x0) - EXA ○ DEMNA
- CHARON-AXP (AlphaServer DS10, DS20, ES40, GS80, GS160, ...) - EWA ● DE500AA ○ DE500BA ○ DE435
- CHARON-AXP (AlphaServer DS10, DS20, ES40, GS80, GS160, ...) - EIA ○ DE602
- DEBNI systems (VAX BI or VAX 6000) - DEBNI ○ DEBNI
- PMAD-AA (TurboChannel add-on card for VAX 4000-90A ○ PMADAA

```
NIC connection name.
load DE500AA/dec21x4x EWA interface=EWA0
load packet_port/chnetwrk EWA0 interface="connection:Charon"
```

It is possible to paste the content of the clipboard to the CHARON configuration file as shown in the example below:

```
...

load DE500BA/dec21x4x EWA interface=EWA0
load packet_port/chnetwrk EWA0 interface="(disabled)"
load packet_port/chnetwrk EWA0 interface="connection:Charon"

...
```

Press the "Next" button to see the log file:



CHARON Network Control Center

CHARON Network Control Center.
Application log & results.

Date:	Message ID:	Message Text:
2015-10-20 ...	04000131	NETCFG-UTILITY: Selected action: Dedicate to CHARON adapter Realtek PCIe GBE Family Cor
2015-10-20 ...	04000251	NETCFG-UTILITY: The NIC 'Realtek PCIe GBE Family Controller' configured for CHARON networ
2015-10-20 ...	04000250	NETCFG-UTILITY: The NIC 'Intel(R) 82579LM Gigabit Network Connection' not configured for CH
2015-10-20 ...	04000251	NETCFG-UTILITY: The NIC 'Realtek PCIe GBE Family Controller' configured for CHARON networ
2015-10-20 ...	04000126	NETCFG-UTILITY: The NIC 'Realtek PCIe GBE Family Controller' configuration is correct!
2015-10-20 ...	04000126	NETCFG-UTILITY: The NIC 'Realtek PCIe GBE Family Controller' configuration is correct!
2015-10-20 ...	0400012E	NETCFG-UTILITY: Selected adapter: Realtek PCIe GBE Family Controller (PHYSICAL DEDICATE
2015-10-20 ...	04000251	NETCFG-UTILITY: The NIC 'Realtek PCIe GBE Family Controller' configured for CHARON networ
2015-10-20 ...	04000126	NETCFG-UTILITY: The NIC 'Realtek PCIe GBE Family Controller' configuration is correct!
2015-10-20 ...	04000147	NETCFG-UTILITY: Choose <NEXT> Wizard button (from CChooseadapter_Config).
2015-10-20 ...	0400016A	NETCFG-UTILITY: The CHARON Network Control Center Config File suggestion step.
2015-10-20 ...	04000133	NETCFG-UTILITY: Net Configuration suggestion: NIC connection name load DE500BA/dec21x4x
2015-10-20 ...	04000133	NETCFG-UTILITY: Net Configuration suggestion: NIC connection name load DE602/i8255x EIA i
2015-10-20 ...	04000133	NETCFG-UTILITY: Net Configuration suggestion: NIC connection name load DE500BA/dec21x4x
2015-10-20 ...	04000147	NETCFG-UTILITY: Choose <NEXT> Wizard button (from CConfigFilePage).

□ Show detailed log

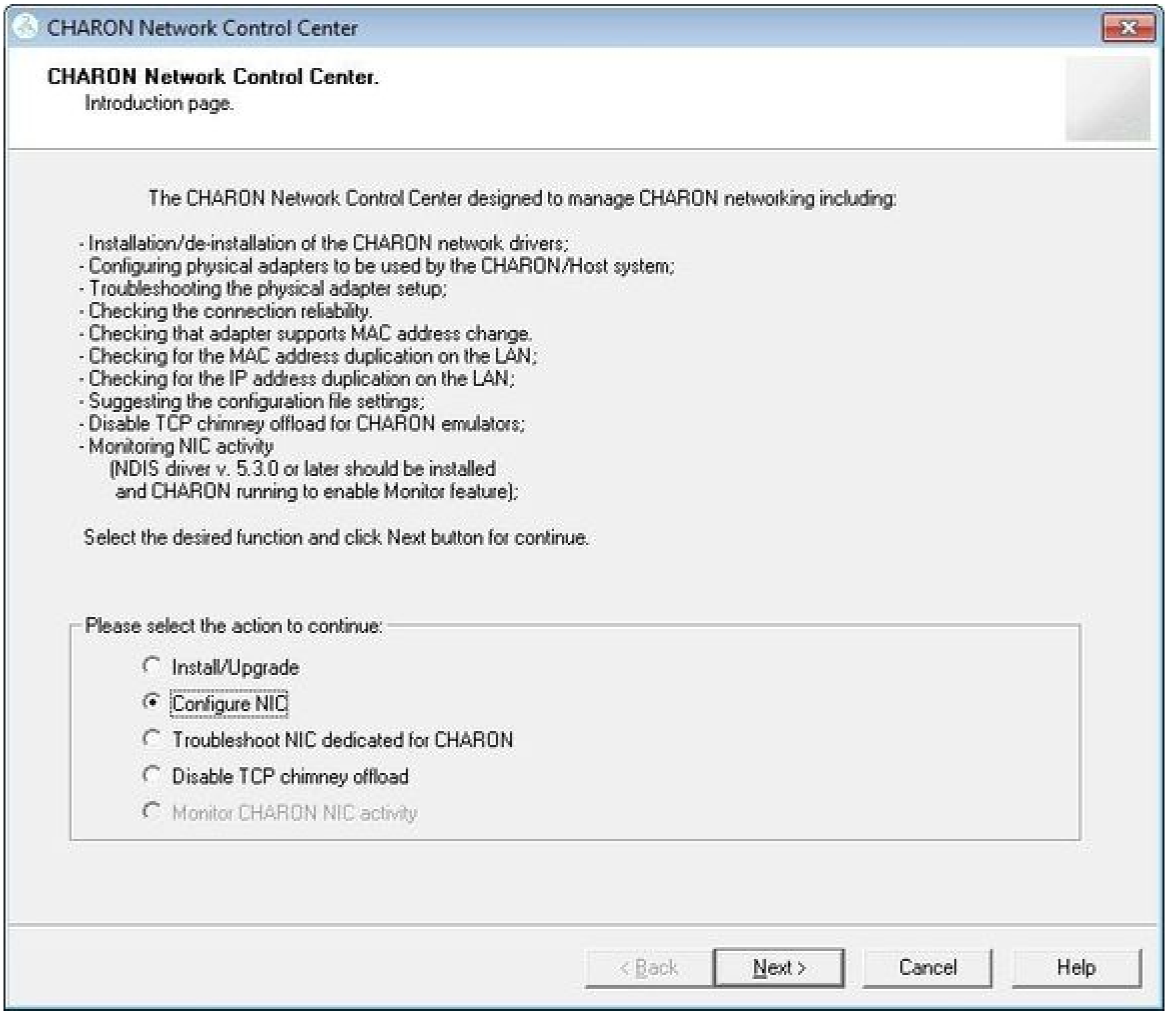
Review the log for errors. You can save the log to a file by pressing the "Save Log in file" button in case the log file has to be sent to Stromasys Customer Support.

Select "Show detailed log" to display more detail.

Press the "Finish" button to exit.

Release of the host network interfaces

Start the utility, select "Configure NIC" and press the "Next" button:



CHARON Network Control Center.
Introduction page.

The CHARON Network Control Center designed to manage CHARON networking including:

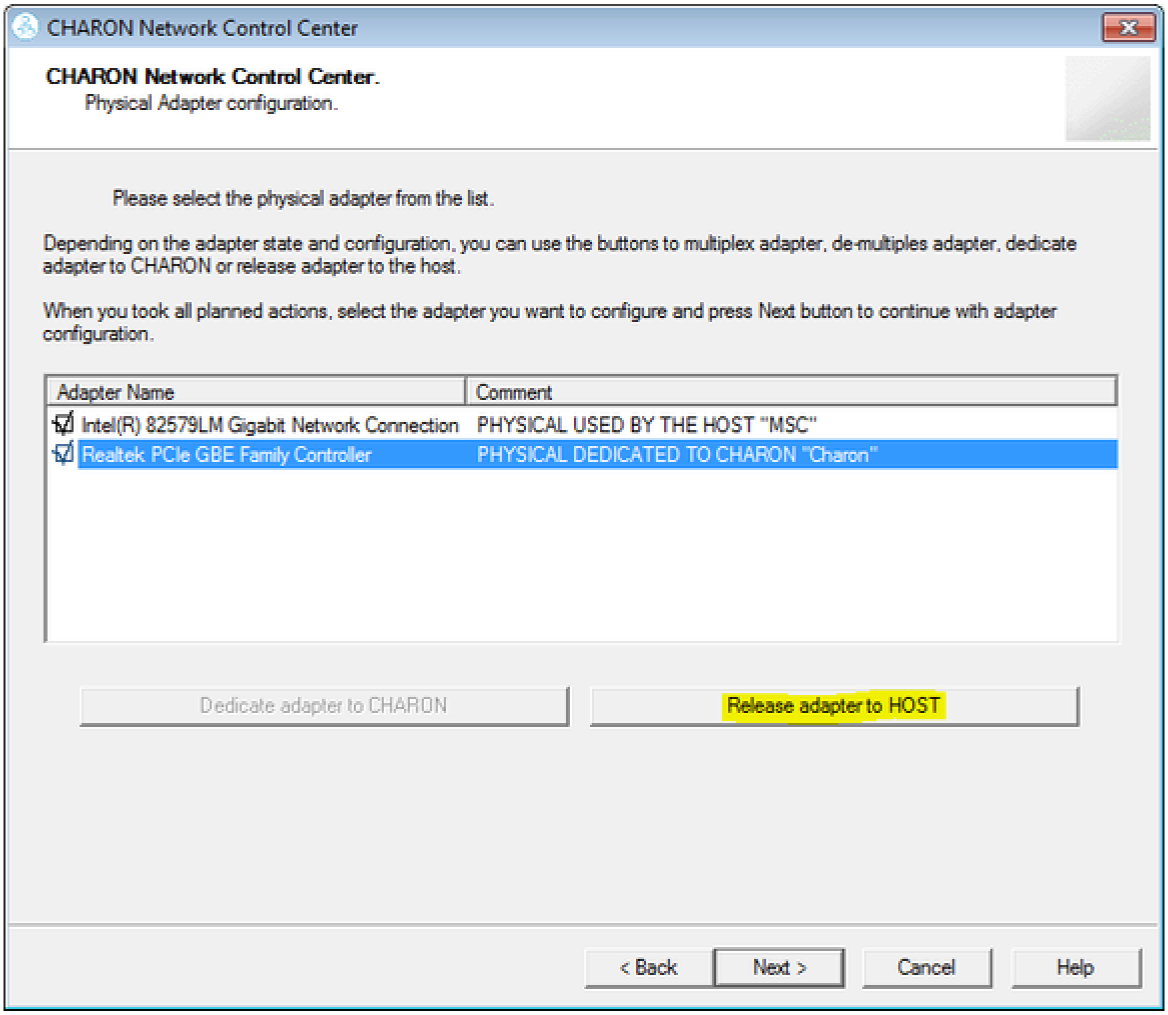
- Installation/de-installation of the CHARON network drivers;
- Configuring physical adapters to be used by the CHARON/Host system;
- Troubleshooting the physical adapter setup;
- Checking the connection reliability.
- Checking that adapter supports MAC address change.
- Checking for the MAC address duplication on the LAN;
- Checking for the IP address duplication on the LAN;
- Suggesting the configuration file settings;
- Disable TCP chimney offload for CHARON emulators;
- Monitoring NIC activity
  (NDIS driver v. 5.3.0 or later should be installed
  and CHARON running to enable Monitor feature);

Select the desired function and click Next button for continue.

Please select the action to continue:

○ Install/Upgrade
● Configure NIC
○ Troubleshoot NIC dedicated for CHARON
○ Disable TCP chimney offload
○ Monitor CHARON NIC activity

Select the interface to be released back to the host (in the example below its name is "Charon"), press the "Release adapter to HOST" button:



CHARON Network Control Center

CHARON Network Control Center.
Physical Adapter configuration.

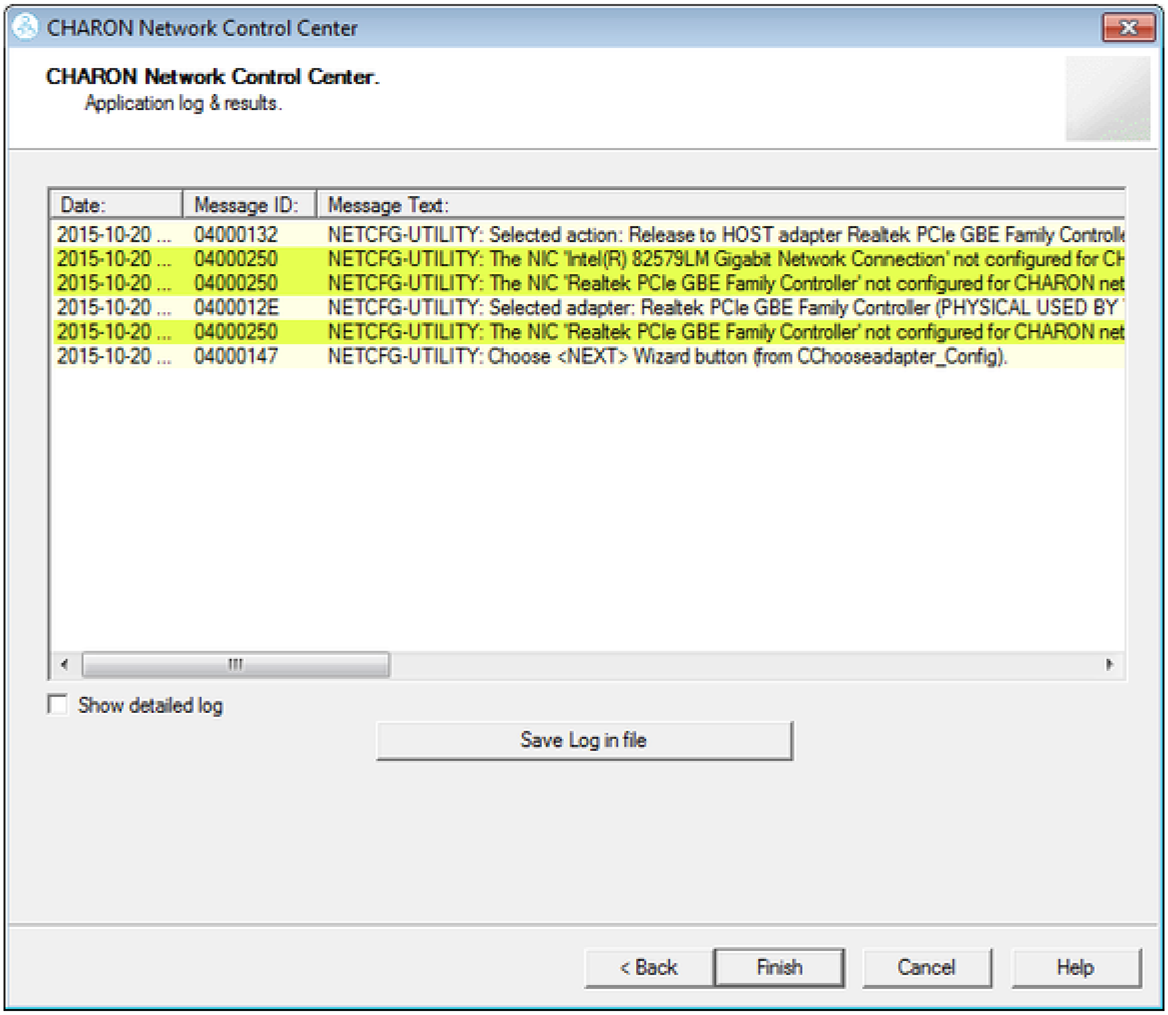
Please select the physical adapter from the list.

Depending on the adapter state and configuration, you can use the buttons to multiplex adapter, de-multiples adapter, dedicate adapter to CHARON or release adapter to the host.

When you took all planned actions, select the adapter you want to configure and press Next button to continue with adapter configuration.

Adapter Name	Comment
☑ Intel(R) 82579LM Gigabit Network Connection	PHYSICAL USED BY THE HOST "MSC"
☑ Realtek PCIe GBE Family Controller	PHYSICAL DEDICATED TO CHARON "Charon"

Press the "Next" button and review the log of this operation:



CHARON Network Control Center
CHARON Network Control Center.
Application log & results.
Show detailed log
Save Log in file
< Back
Finish
Cancel
Help

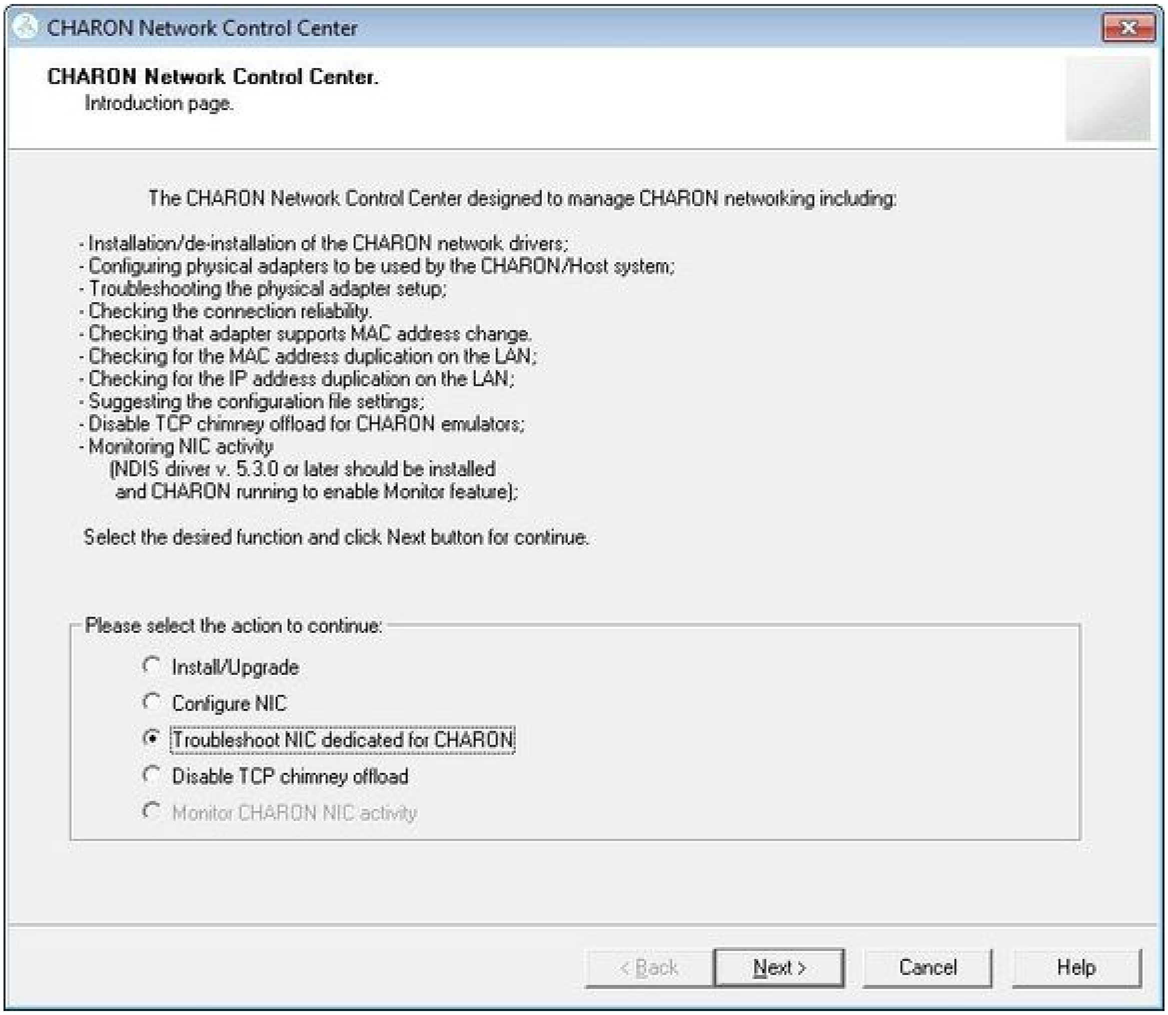
```
Date:           Message ID:   Message Text:
2015-10-20 ...  04000132      NETCFG-UTILITY: Selected action: Release to HOST adapter Realtek PCIe GBE Family Controll
2015-10-20 ...  04000250      NETCFG-UTILITY: The NIC 'Intel(R) 82579LM Gigabit Network Connection' not configured for CH
2015-10-20 ...  04000250      NETCFG-UTILITY: The NIC 'Realtek PCIe GBE Family Controller' not configured for CHARON net
2015-10-20 ...  0400012E      NETCFG-UTILITY: Selected adapter: Realtek PCIe GBE Family Controller (PHYSICAL USED BY
2015-10-20 ...  04000250      NETCFG-UTILITY: The NIC 'Realtek PCIe GBE Family Controller' not configured for CHARON net
2015-10-20 ...  04000147      NETCFG-UTILITY: Choose <NEXT> Wizard button (from CChooseadapter_Config).
```

Select "Show detailed log" to display more detail.

Press the "Finish" button to exit.

Troubleshooting the CHARON network interfaces configuration

Start the utility, select "Troubleshoot NIC dedicated for CHARON" and press the "Next" button:



CHARON Network Control Center.
Introduction page.

The CHARON Network Control Center designed to manage CHARON networking including:

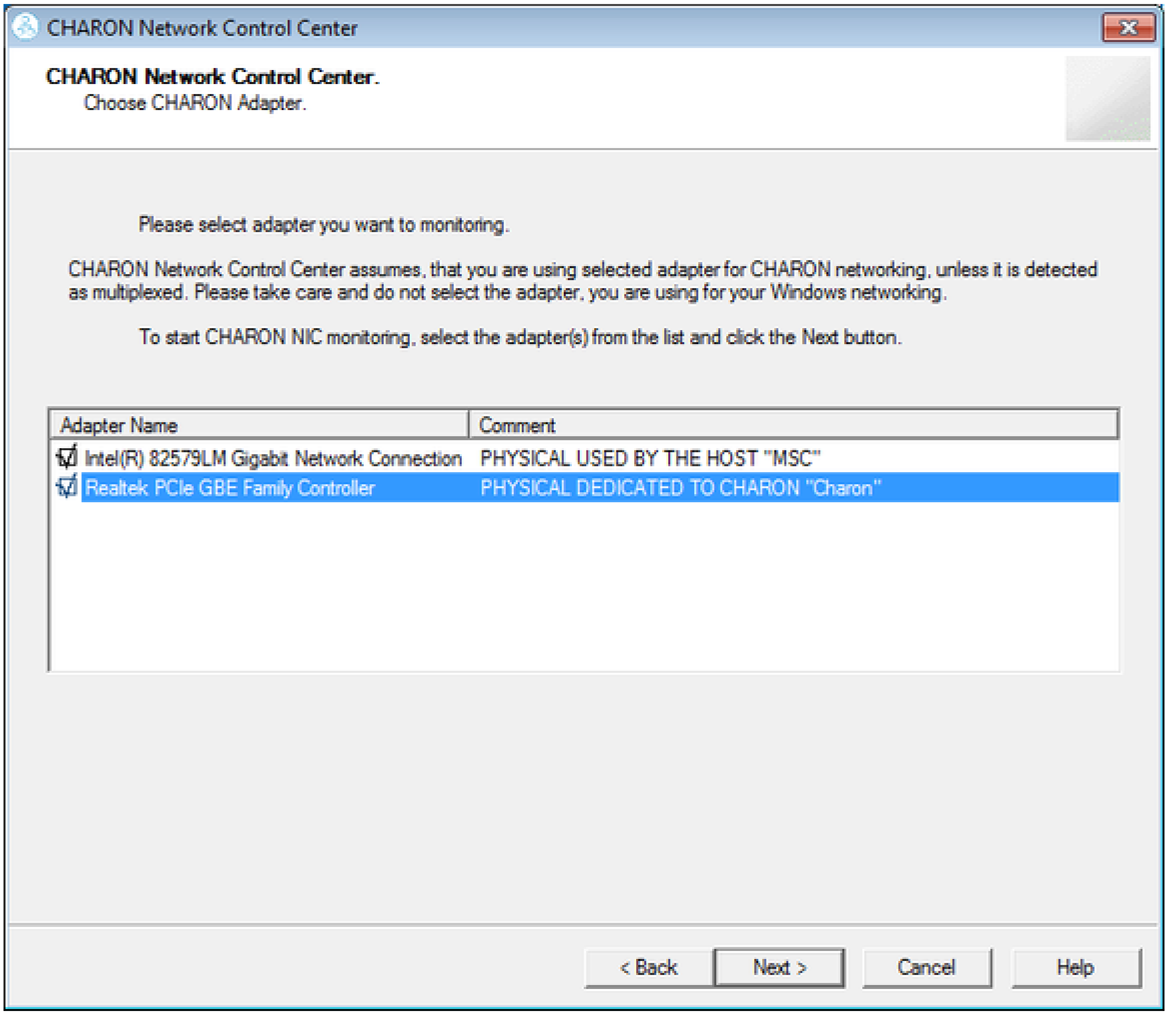
- Installation/de-installation of the CHARON network drivers;
- Configuring physical adapters to be used by the CHARON/Host system;
- Troubleshooting the physical adapter setup;
- Checking the connection reliability.
- Checking that adapter supports MAC address change.
- Checking for the MAC address duplication on the LAN;
- Checking for the IP address duplication on the LAN;
- Suggesting the configuration file settings;
- Disable TCP chimney offload for CHARON emulators;
- Monitoring NIC activity
  (NDIS driver v. 5.3.0 or later should be installed
  and CHARON running to enable Monitor feature);

Select the desired function and click Next button for continue.

Please select the action to continue:

○ Install/Upgrade
○ Configure NIC
● Troubleshoot NIC dedicated for CHARON
○ Disable TCP chimney offload
○ Monitor CHARON NIC activity

Select the target interface and press the "Next" button:



CHARON Network Control Center.

Choose CHARON Adapter.

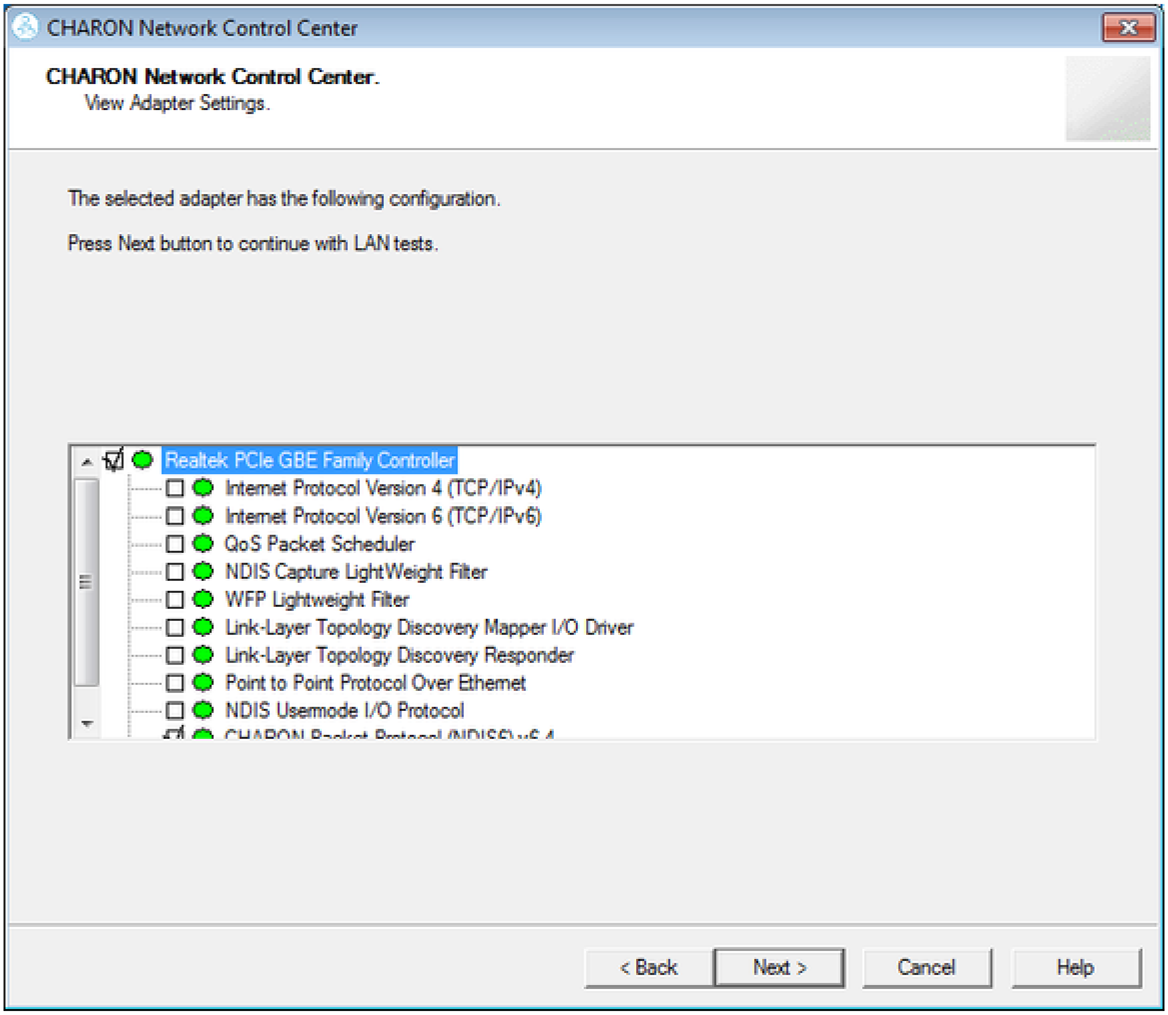
Please select adapter you want to monitoring.

CHARON Network Control Center assumes, that you are using selected adapter for CHARON networking, unless it is detected as multiplexed. Please take care and do not select the adapter, you are using for your Windows networking.

To start CHARON NIC monitoring, select the adapter(s) from the list and click the Next button.

Adapter Name	Comment
☑ Intel(R) 82579LM Gigabit Network Connection	PHYSICAL USED BY THE HOST "MSC"
☑ Realtek PCIe GBE Family Controller	PHYSICAL DEDICATED TO CHARON "Charon"

Review the status of the interface:



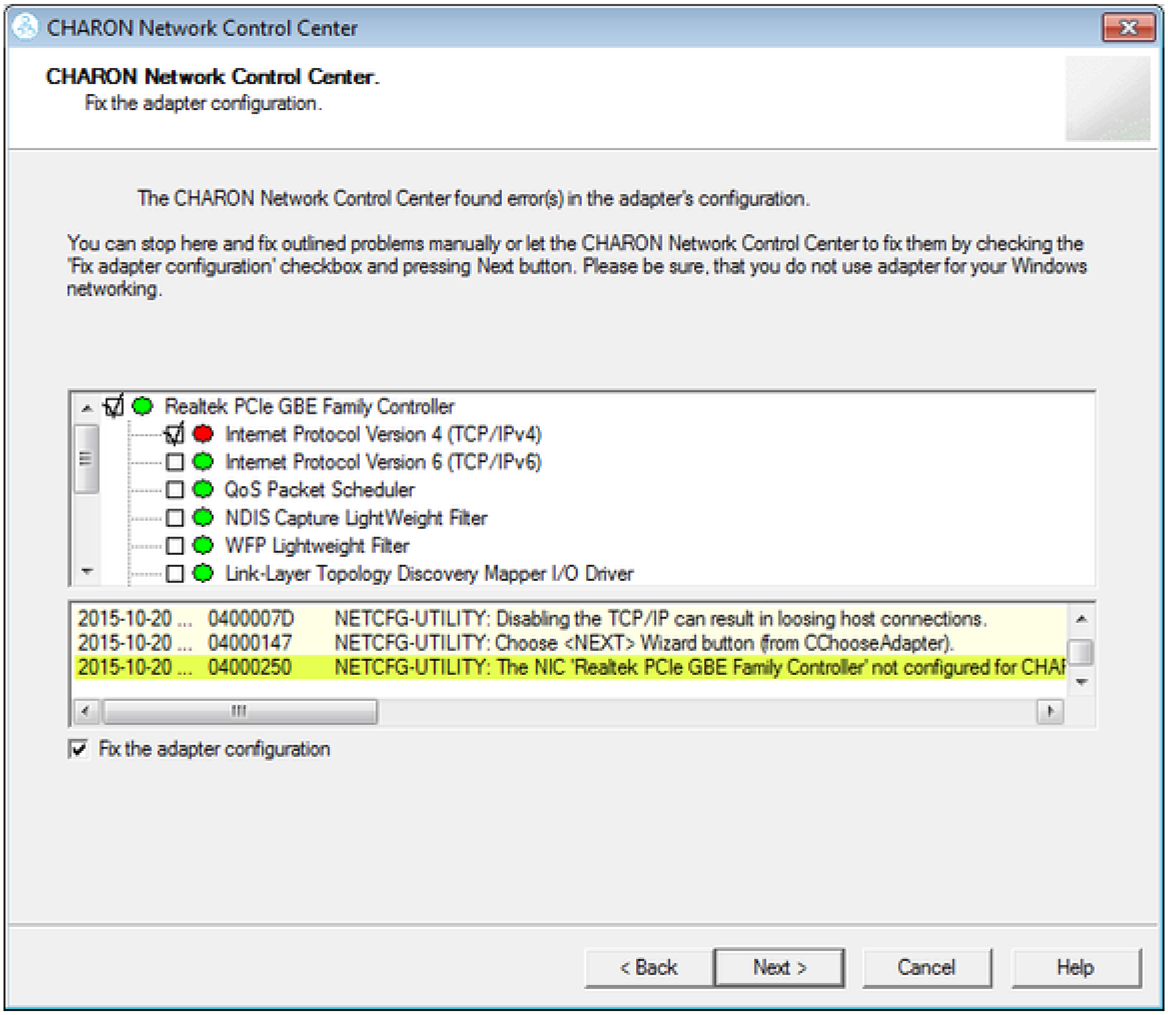
CHARON Network Control Center.
View Adapter Settings.

The selected adapter has the following configuration.

Press Next button to continue with LAN tests.

- ☑ ● Realtek PCIe GBE Family Controller
  - ☐ ● Internet Protocol Version 4 (TCP/IPv4)
  - ☐ ● Internet Protocol Version 6 (TCP/IPv6)
  - ☐ ● QoS Packet Scheduler
  - ☐ ● NDIS Capture LightWeight Filter
  - ☐ ● WFP Lightweight Filter
  - ☐ ● Link-Layer Topology Discovery Mapper I/O Driver
  - ☐ ● Link-Layer Topology Discovery Responder
  - ☐ ● Point to Point Protocol Over Ethernet
  - ☐ ● NDIS Usermode I/O Protocol
  - ☑ ● CHARON Packet Protocol (NDIS6) v6.4

In the example below, there is an issue with the host TCP/IP being enabled on the CHARON interface. Review the problem description, select "Fix the adapter configuration" then press the "Next" button.



CHARON Network Control Center.
Fix the adapter configuration.

The CHARON Network Control Center found error(s) in the adapter's configuration.

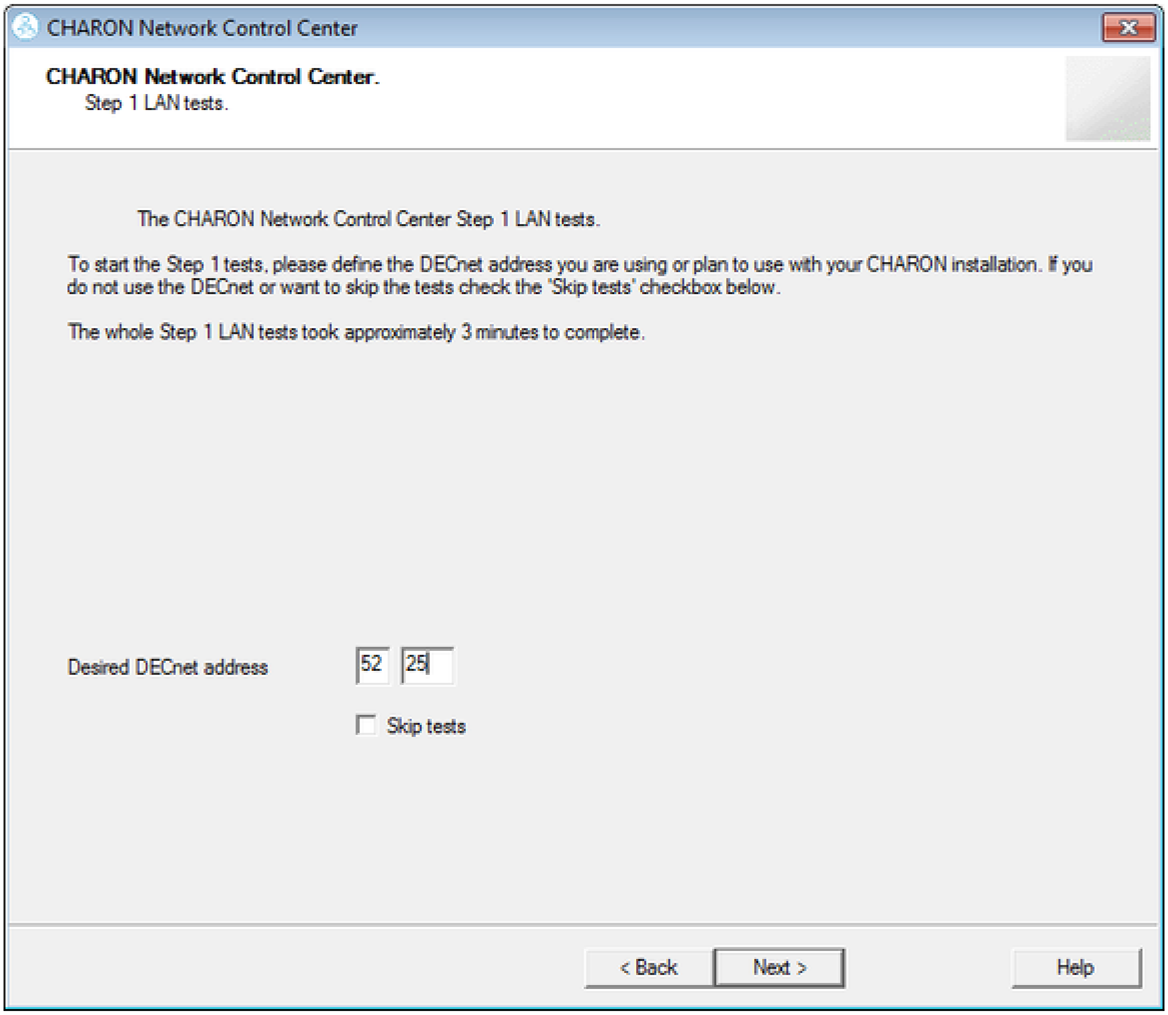
You can stop here and fix outlined problems manually or let the CHARON Network Control Center to fix them by checking the 'Fix adapter configuration' checkbox and pressing Next button. Please be sure, that you do not use adapter for your Windows networking.

- ☑ 🟢 Realtek PCIe GBE Family Controller
  - ☑ 🔴 Internet Protocol Version 4 (TCP/IPv4)
  - ☐ 🟢 Internet Protocol Version 6 (TCP/IPv6)
  - ☐ 🟢 QoS Packet Scheduler
  - ☐ 🟢 NDIS Capture LightWeight Filter
  - ☐ 🟢 WFP Lightweight Filter
  - ☐ 🟢 Link-Layer Topology Discovery Mapper I/O Driver

```
2015-10-20 ...  0400007D   NETCFG-UTILITY: Disabling the TCP/IP can result in loosing host connections.
2015-10-20 ...  04000147   NETCFG-UTILITY: Choose <NEXT> Wizard button (from CChooseAdapter).
2015-10-20 ...  04000250   NETCFG-UTILITY: The NIC 'Realtek PCIe GBE Family Controller' not configured for CHAI
```

☑ Fix the adapter configuration

The utility will fix issues and report a good status for the interface. Press the "Next" button, the following dialog will appear:



CHARON Network Control Center.
Step 1 LAN tests.

The CHARON Network Control Center Step 1 LAN tests.

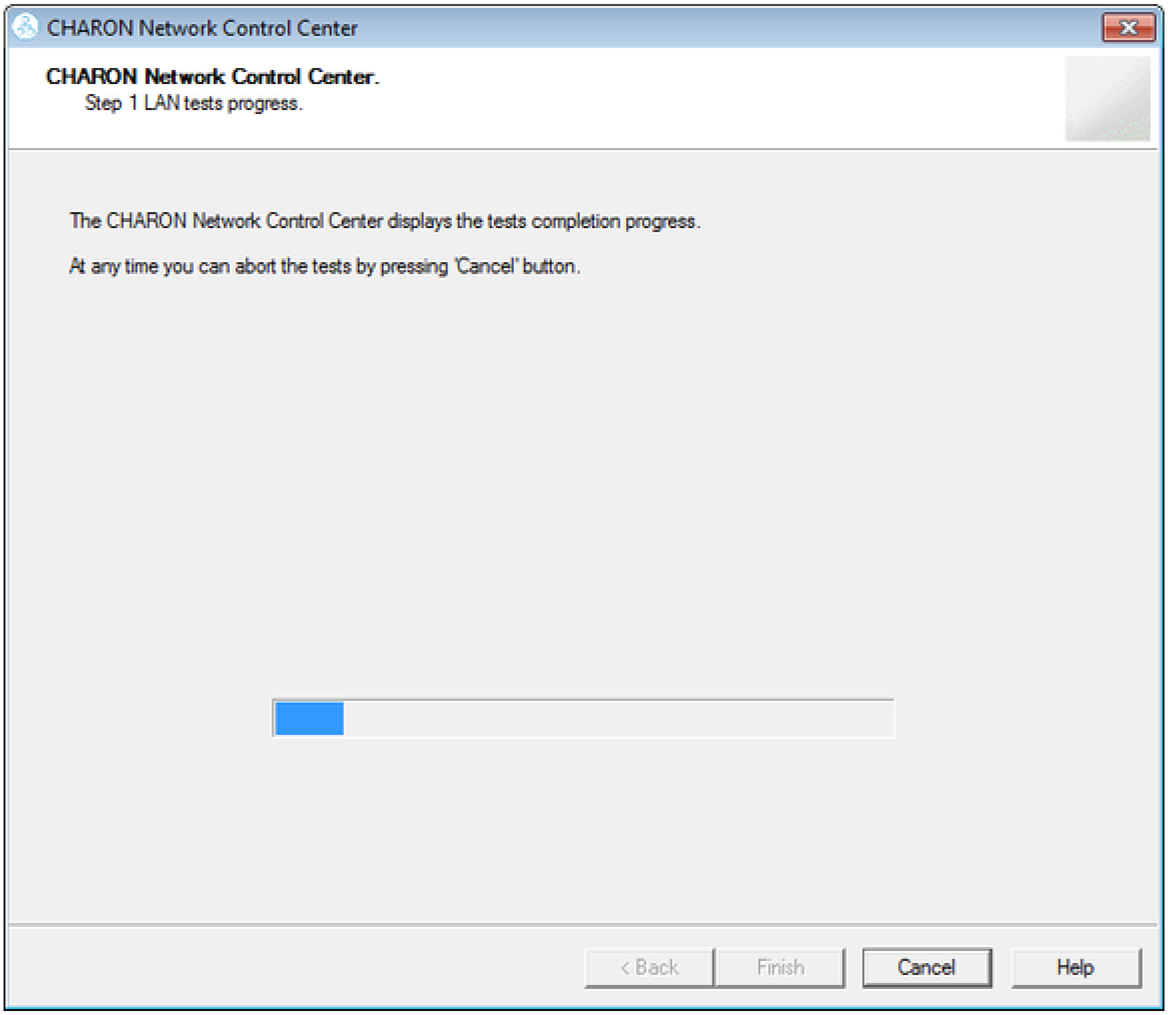
To start the Step 1 tests, please define the DECnet address you are using or plan to use with your CHARON installation. If you do not use the DECnet or want to skip the tests check the 'Skip tests' checkbox below.

The whole Step 1 LAN tests took approximately 3 minutes to complete.

Desired DECnet address 52 25

□ Skip tests

The Network Control Center offers to check whether the DECnet address, to be used by CHARON, is unique on the network in this step. Enter the desired DECnet address, for the CHARON guest, and press the "Next" button:

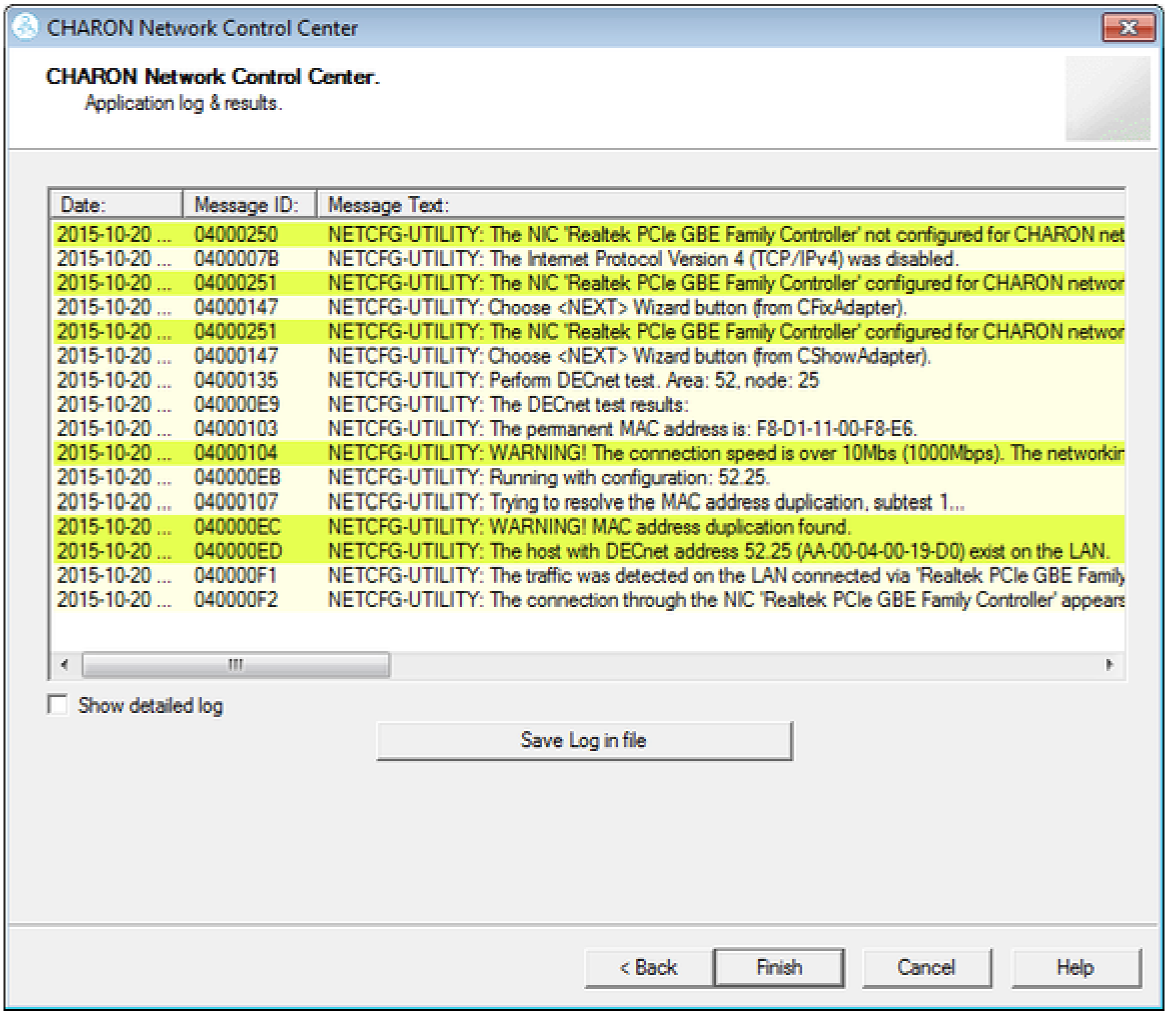


CHARON Network Control Center.
Step 1 LAN tests progress.

The CHARON Network Control Center displays the tests completion progress.

At any time you can abort the tests by pressing 'Cancel' button.

Once the check is completed, if there are issues, the utility will abort the test and display a log containing information about the issues found:

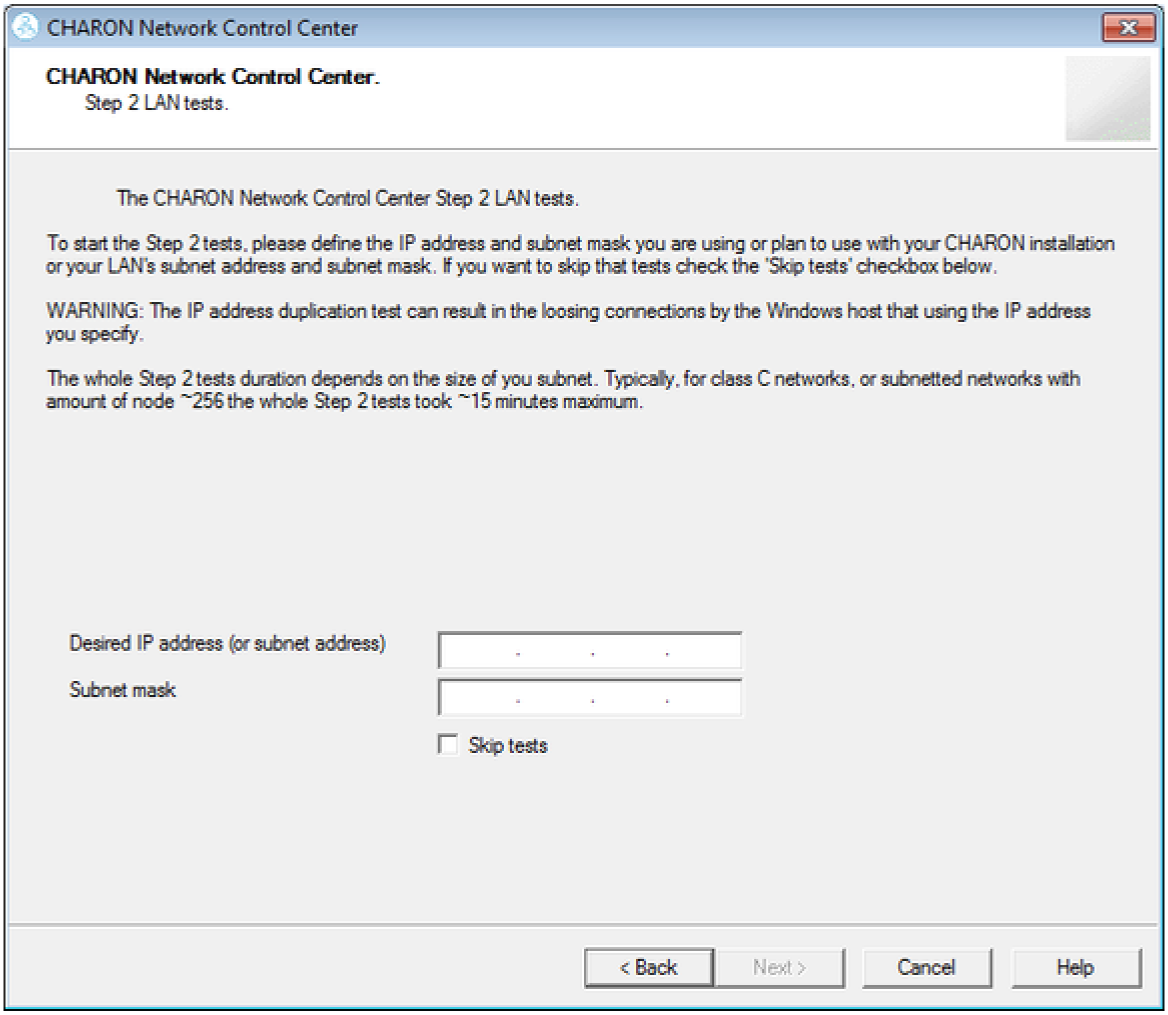


CHARON Network Control Center.

Application log & results.

Date:	Message ID:	Message Text:
2015-10-20 ...	04000250	NETCFG-UTILITY: The NIC 'Realtek PCIe GBE Family Controller' not configured for CHARON net
2015-10-20 ...	0400007B	NETCFG-UTILITY: The Internet Protocol Version 4 (TCP/IPv4) was disabled.
2015-10-20 ...	04000251	NETCFG-UTILITY: The NIC 'Realtek PCIe GBE Family Controller' configured for CHARON networ
2015-10-20 ...	04000147	NETCFG-UTILITY: Choose <NEXT> Wizard button (from CFixAdapter).
2015-10-20 ...	04000251	NETCFG-UTILITY: The NIC 'Realtek PCIe GBE Family Controller' configured for CHARON networ
2015-10-20 ...	04000147	NETCFG-UTILITY: Choose <NEXT> Wizard button (from CShowAdapter).
2015-10-20 ...	04000135	NETCFG-UTILITY: Perform DECnet test. Area: 52, node: 25
2015-10-20 ...	040000E9	NETCFG-UTILITY: The DECnet test results:
2015-10-20 ...	04000103	NETCFG-UTILITY: The permanent MAC address is: F8-D1-11-00-F8-E6.
2015-10-20 ...	04000104	NETCFG-UTILITY: WARNING! The connection speed is over 10Mbs (1000Mbps). The networkir
2015-10-20 ...	040000EB	NETCFG-UTILITY: Running with configuration: 52.25.
2015-10-20 ...	04000107	NETCFG-UTILITY: Trying to resolve the MAC address duplication, subtest 1...
2015-10-20 ...	040000EC	NETCFG-UTILITY: WARNING! MAC address duplication found.
2015-10-20 ...	040000ED	NETCFG-UTILITY: The host with DECnet address 52.25 (AA-00-04-00-19-D0) exist on the LAN.
2015-10-20 ...	040000F1	NETCFG-UTILITY: The traffic was detected on the LAN connected via 'Realtek PCIe GBE Family
2015-10-20 ...	040000F2	NETCFG-UTILITY: The connection through the NIC 'Realtek PCIe GBE Family Controller' appears

If no issues are found, the utility will offer to check the CHARON TCP/IP networking the same way:



CHARON Network Control Center

CHARON Network Control Center.
Step 2 LAN tests.

The CHARON Network Control Center Step 2 LAN tests.

To start the Step 2 tests, please define the IP address and subnet mask you are using or plan to use with your CHARON installation or your LAN's subnet address and subnet mask. If you want to skip that tests check the 'Skip tests' checkbox below.

WARNING: The IP address duplication test can result in the loosing connections by the Windows host that using the IP address you specify.

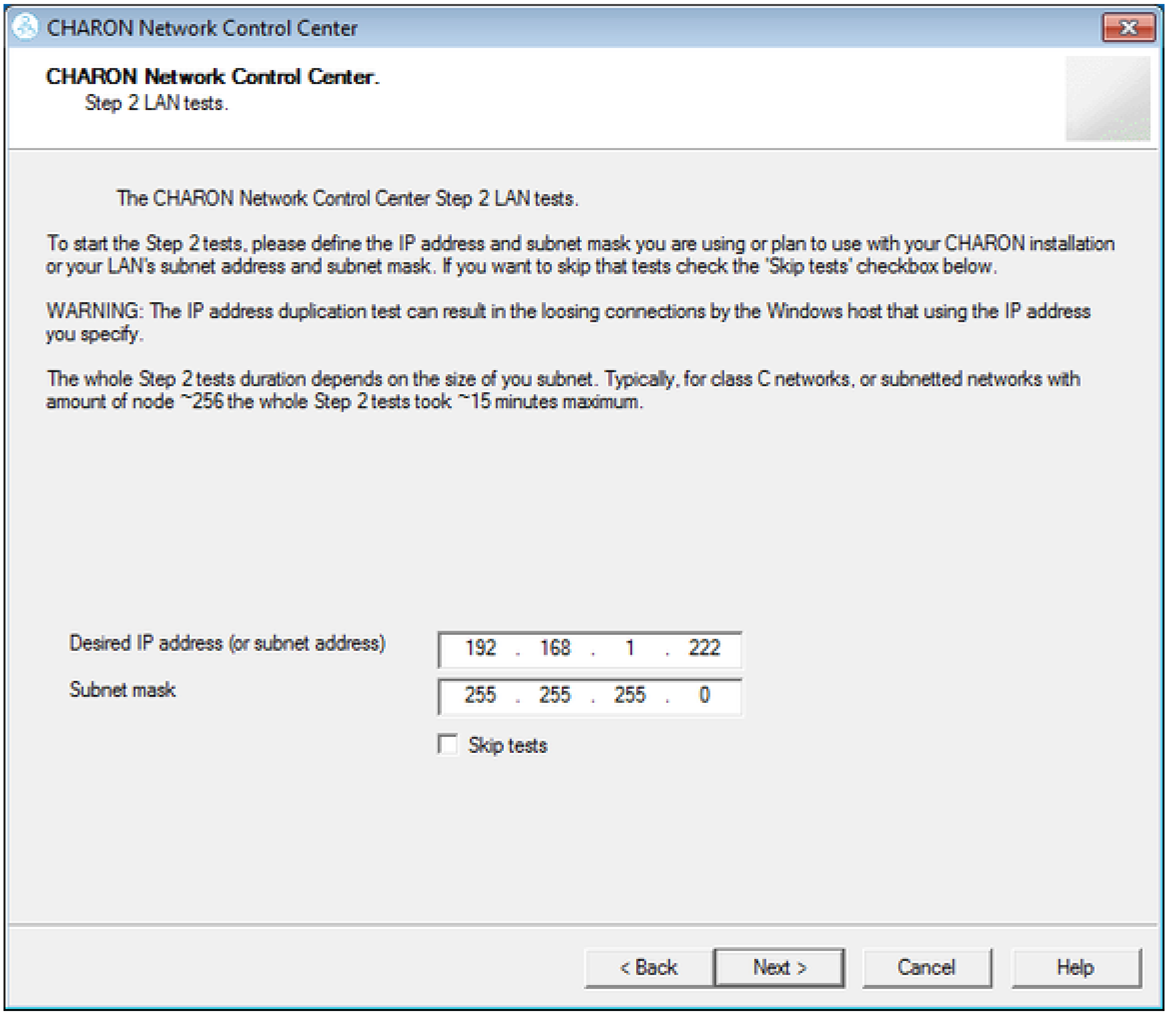
The whole Step 2 tests duration depends on the size of you subnet. Typically, for class C networks, or subnetted networks with amount of node ~256 the whole Step 2 tests took ~15 minutes maximum.

Desired IP address (or subnet address) [. . .]

Subnet mask [. . .]

□ Skip tests

Enter the IP address and subnet mask to be used by the CHARON guest then press the "Next" button:



CHARON Network Control Center.
Step 2 LAN tests.

The CHARON Network Control Center Step 2 LAN tests.

To start the Step 2 tests, please define the IP address and subnet mask you are using or plan to use with your CHARON installation or your LAN's subnet address and subnet mask. If you want to skip that tests check the 'Skip tests' checkbox below.

WARNING: The IP address duplication test can result in the loosing connections by the Windows host that using the IP address you specify.

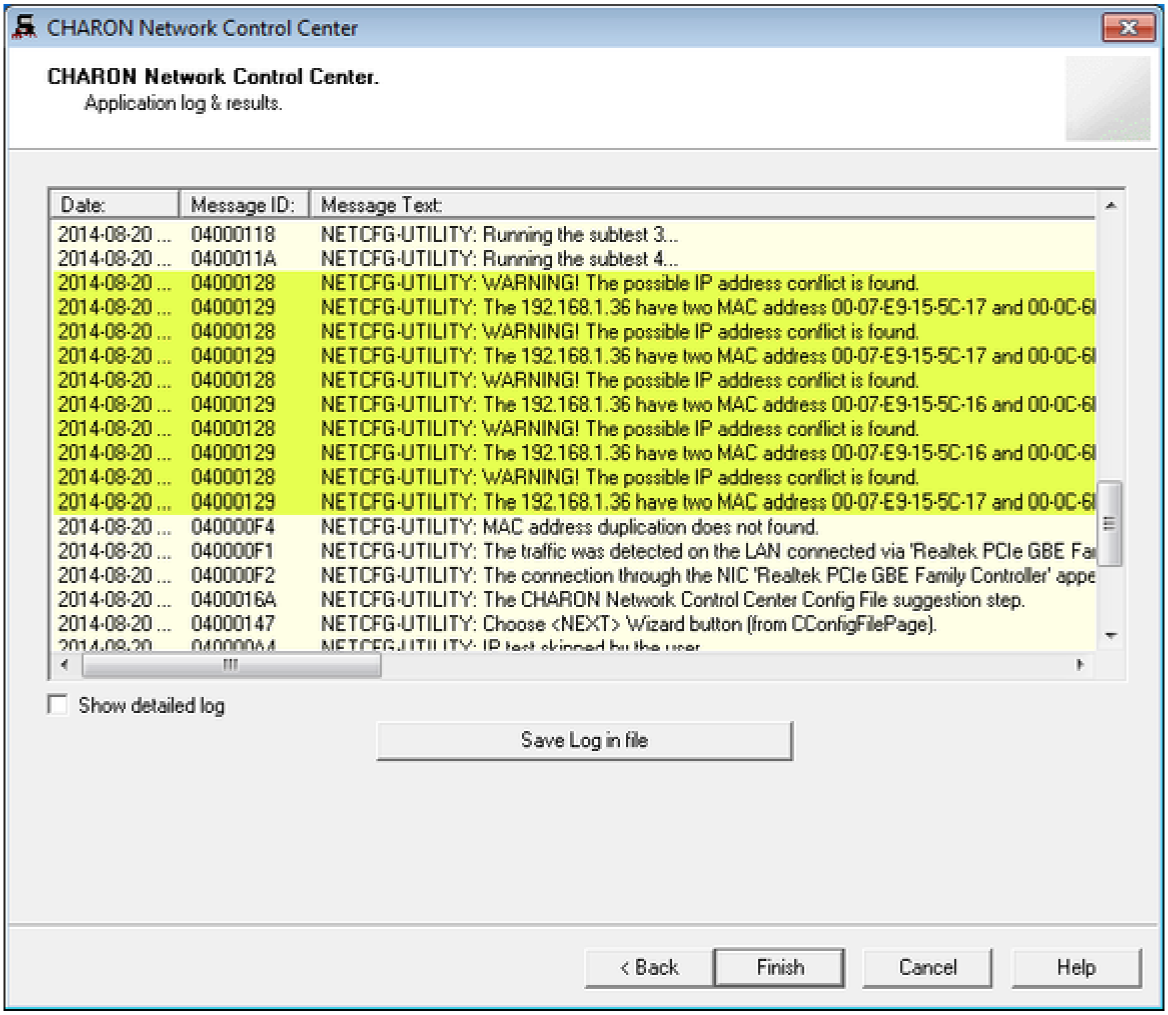
The whole Step 2 tests duration depends on the size of you subnet. Typically, for class C networks, or subnetted networks with amount of node ~256 the whole Step 2 tests took ~15 minutes maximum.

Desired IP address (or subnet address) 192 . 168 . 1 . 222

Subnet mask 255 . 255 . 255 . 0

□ Skip tests

The Network Control Center will display the log of the performed operations:



CHARON Network Control Center
CHARON Network Control Center.
Application log & results.
□ Show detailed log
Save Log in file
< Back
Finish
Cancel
Help

Date:	Message ID:	Message Text:
2014-08-20 ...	04000118	NETCFG-UTILITY: Running the subtest 3...
2014-08-20 ...	0400011A	NETCFG-UTILITY: Running the subtest 4...
2014-08-20 ...	04000128	NETCFG-UTILITY: WARNING! The possible IP address conflict is found.
2014-08-20 ...	04000129	NETCFG-UTILITY: The 192.168.1.36 have two MAC address 00-07-E9-15-5C-17 and 00-0C-6I
2014-08-20 ...	04000128	NETCFG-UTILITY: WARNING! The possible IP address conflict is found.
2014-08-20 ...	04000129	NETCFG-UTILITY: The 192.168.1.36 have two MAC address 00-07-E9-15-5C-17 and 00-0C-6I
2014-08-20 ...	04000128	NETCFG-UTILITY: WARNING! The possible IP address conflict is found.
2014-08-20 ...	04000129	NETCFG-UTILITY: The 192.168.1.36 have two MAC address 00-07-E9-15-5C-16 and 00-0C-6I
2014-08-20 ...	04000128	NETCFG-UTILITY: WARNING! The possible IP address conflict is found.
2014-08-20 ...	04000129	NETCFG-UTILITY: The 192.168.1.36 have two MAC address 00-07-E9-15-5C-16 and 00-0C-6I
2014-08-20 ...	04000128	NETCFG-UTILITY: WARNING! The possible IP address conflict is found.
2014-08-20 ...	04000129	NETCFG-UTILITY: The 192.168.1.36 have two MAC address 00-07-E9-15-5C-17 and 00-0C-6I
2014-08-20 ...	040000F4	NETCFG-UTILITY: MAC address duplication does not found.
2014-08-20 ...	040000F1	NETCFG-UTILITY: The traffic was detected on the LAN connected via 'Realtek PCIe GBE Fa
2014-08-20 ...	040000F2	NETCFG-UTILITY: The connection through the NIC 'Realtek PCIe GBE Family Controller' appe
2014-08-20 ...	0400016A	NETCFG-UTILITY: The CHARON Network Control Center Config File suggestion step.
2014-08-20 ...	04000147	NETCFG-UTILITY: Choose <NEXT> Wizard button (from CConfigFilePage).
2014-08-20 ...	040000A4	NETCFG-UTILITY: IP test skipped by the user.

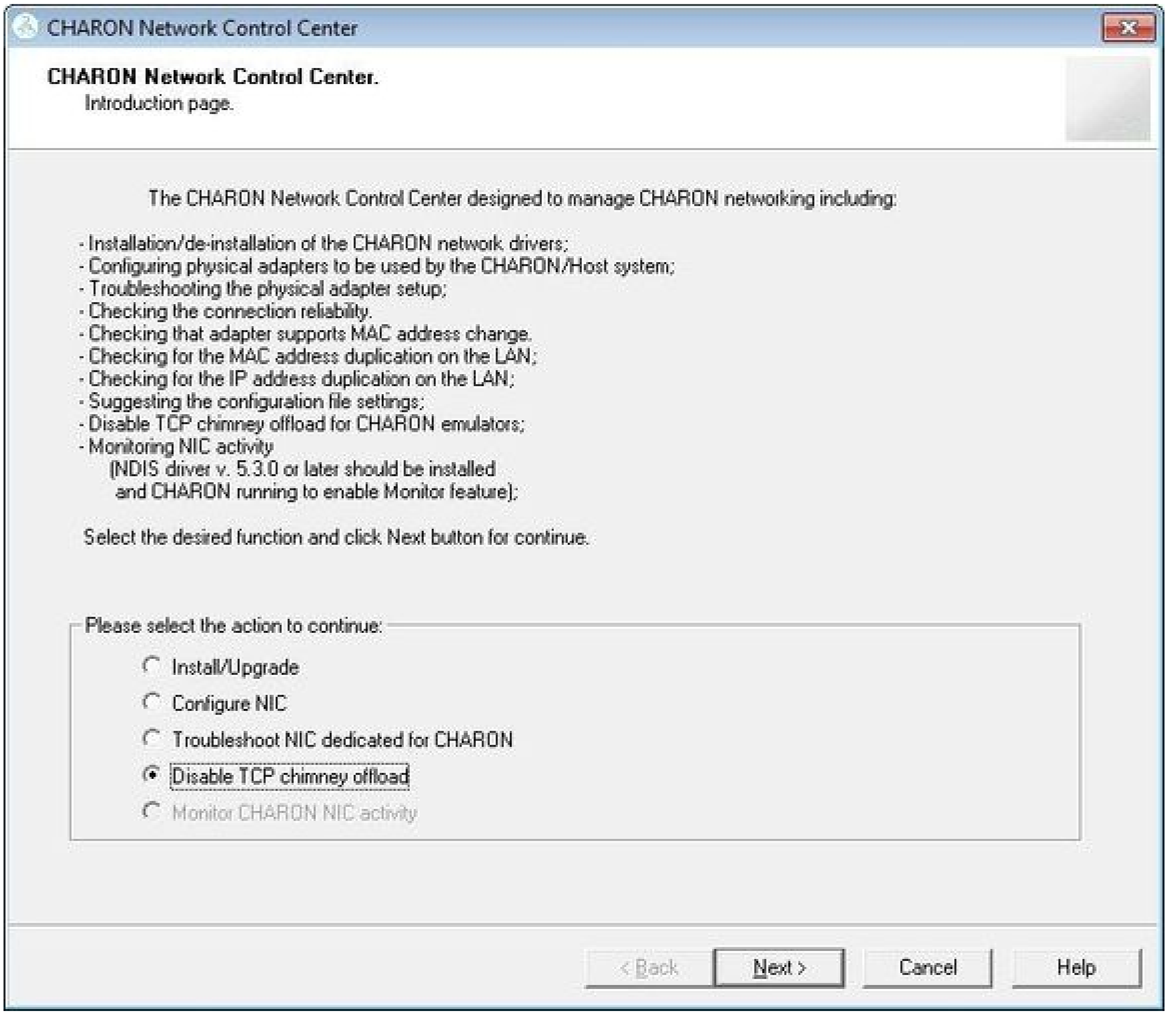
Select "Show detailed log" for more detail.

Press the "Finish" button to exit.

Disable "offload" parameters

The Network Control Center is able to disable "offload" parameters for CHARON. It is highly recommended to disable them to avoid problems with TCP /IP networking.

Start the utility, select "Disable TCP chimney offload for CHARON" and press the "Next" button:



CHARON Network Control Center

CHARON Network Control Center.
Introduction page.

The CHARON Network Control Center designed to manage CHARON networking including:

- Installation/de-installation of the CHARON network drivers;
- Configuring physical adapters to be used by the CHARON/Host system;
- Troubleshooting the physical adapter setup;
- Checking the connection reliability.
- Checking that adapter supports MAC address change.
- Checking for the MAC address duplication on the LAN;
- Checking for the IP address duplication on the LAN;
- Suggesting the configuration file settings;
- Disable TCP chimney offload for CHARON emulators;
- Monitoring NIC activity
  (NDIS driver v. 5.3.0 or later should be installed
  and CHARON running to enable Monitor feature);

Select the desired function and click Next button for continue.

Please select the action to continue:

○ Install/Upgrade
○ Configure NIC
○ Troubleshoot NIC dedicated for CHARON
● Disable TCP chimney offload
○ Monitor CHARON NIC activity

The "offload" parameters will be reset for all CHARON emulators (executable files).

Press the "Finish" button to exit.

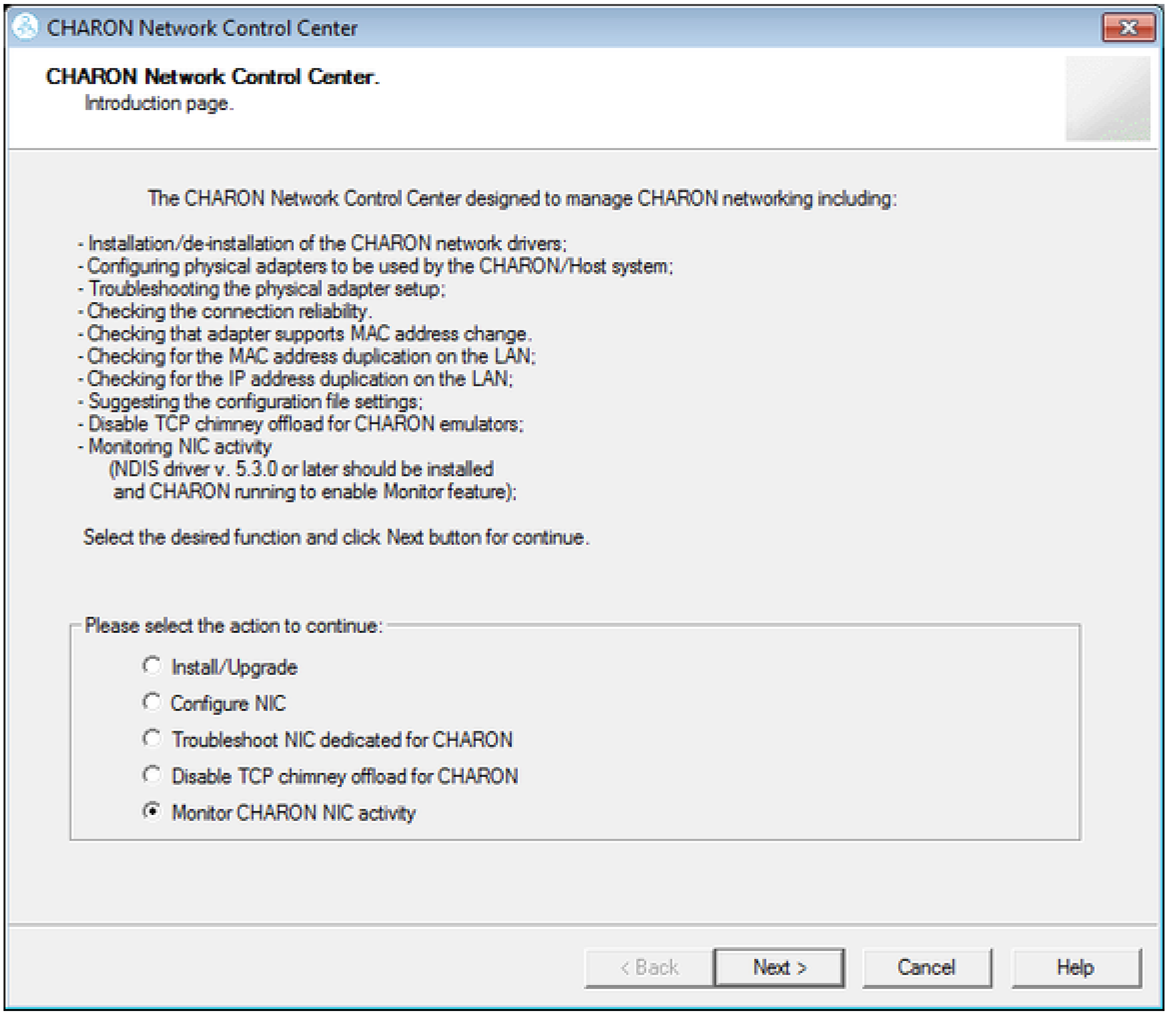
⚠ Please note to restart the CHARON host to enable this settings.

For **additional information regarding disabling this and other offload parameters**, please refer to in chapter Charon-VAX/PDP for Windows installation > Network configuration.

Monitoring the CHARON network activity

The Network Control Center is able to monitor CHARON network activity. This function is available only if CHARON is running.

Start the utility, select "Monitor CHARON NIC activity" and press the "Next" button:



CHARON Network Control Center

CHARON Network Control Center.
Introduction page.

The CHARON Network Control Center designed to manage CHARON networking including:

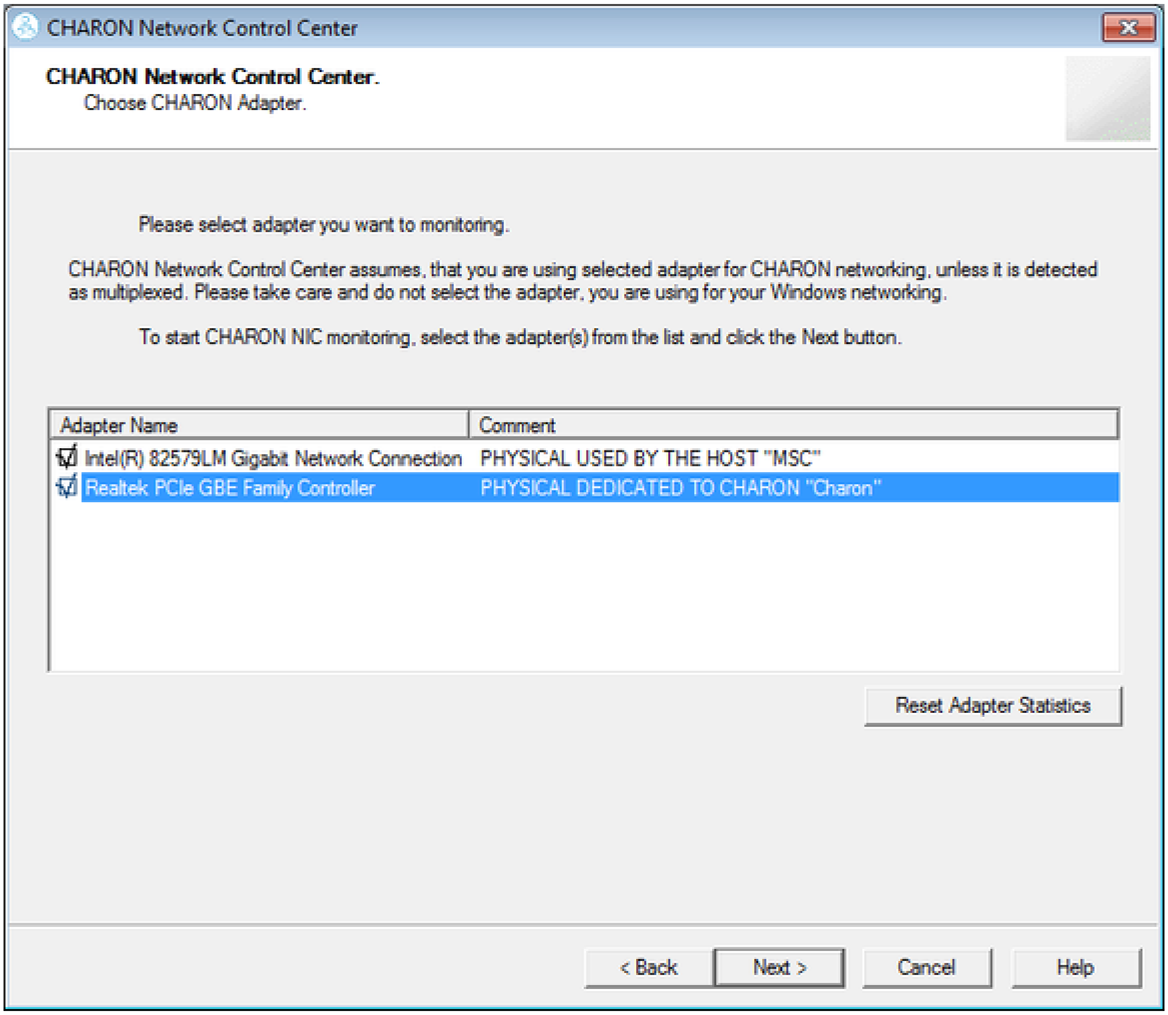
- Installation/de-installation of the CHARON network drivers;
- Configuring physical adapters to be used by the CHARON/Host system;
- Troubleshooting the physical adapter setup;
- Checking the connection reliability.
- Checking that adapter supports MAC address change.
- Checking for the MAC address duplication on the LAN;
- Checking for the IP address duplication on the LAN;
- Suggesting the configuration file settings;
- Disable TCP chimney offload for CHARON emulators;
- Monitoring NIC activity
  (NDIS driver v. 5.3.0 or later should be installed
  and CHARON running to enable Monitor feature);

Select the desired function and click Next button for continue.

Please select the action to continue:

○ Install/Upgrade
○ Configure NIC
○ Troubleshoot NIC dedicated for CHARON
○ Disable TCP chimney offload for CHARON
● Monitor CHARON NIC activity

Select the network interface to monitor (it must be dedicated to CHARON) and press the "Next" button:



CHARON Network Control Center

CHARON Network Control Center.

Choose CHARON Adapter.

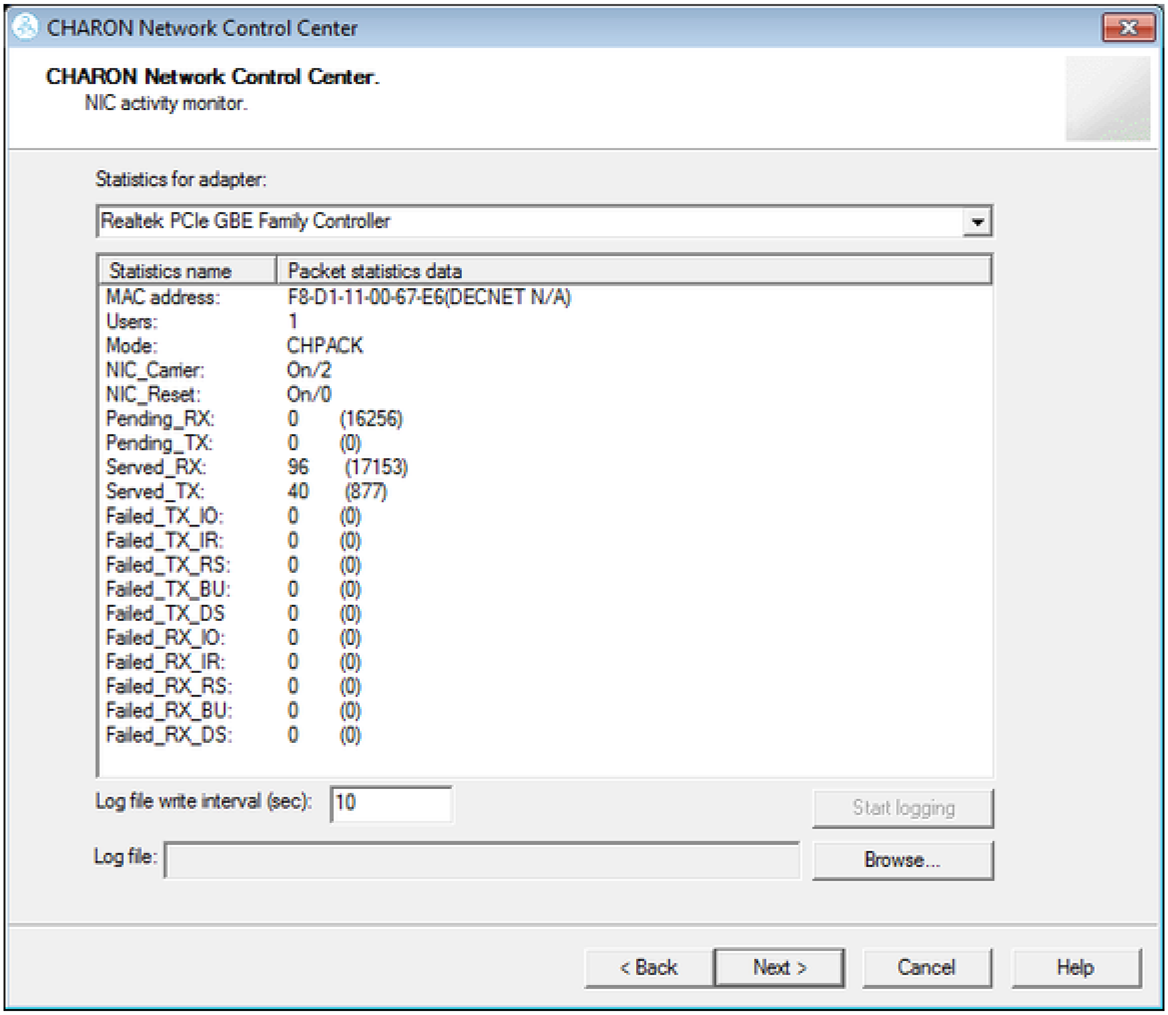
Please select adapter you want to monitoring.

CHARON Network Control Center assumes, that you are using selected adapter for CHARON networking, unless it is detected as multiplexed. Please take care and do not select the adapter, you are using for your Windows networking.

To start CHARON NIC monitoring, select the adapter(s) from the list and click the Next button.

Adapter Name	Comment
☑ Intel(R) 82579LM Gigabit Network Connection	PHYSICAL USED BY THE HOST "MSC"
☑ Realtek PCIe GBE Family Controller	PHYSICAL DEDICATED TO CHARON "Charon"

The utility will display statistics updated in a real time. Note it is possible to record the statistics in a log file with a selected write interval.



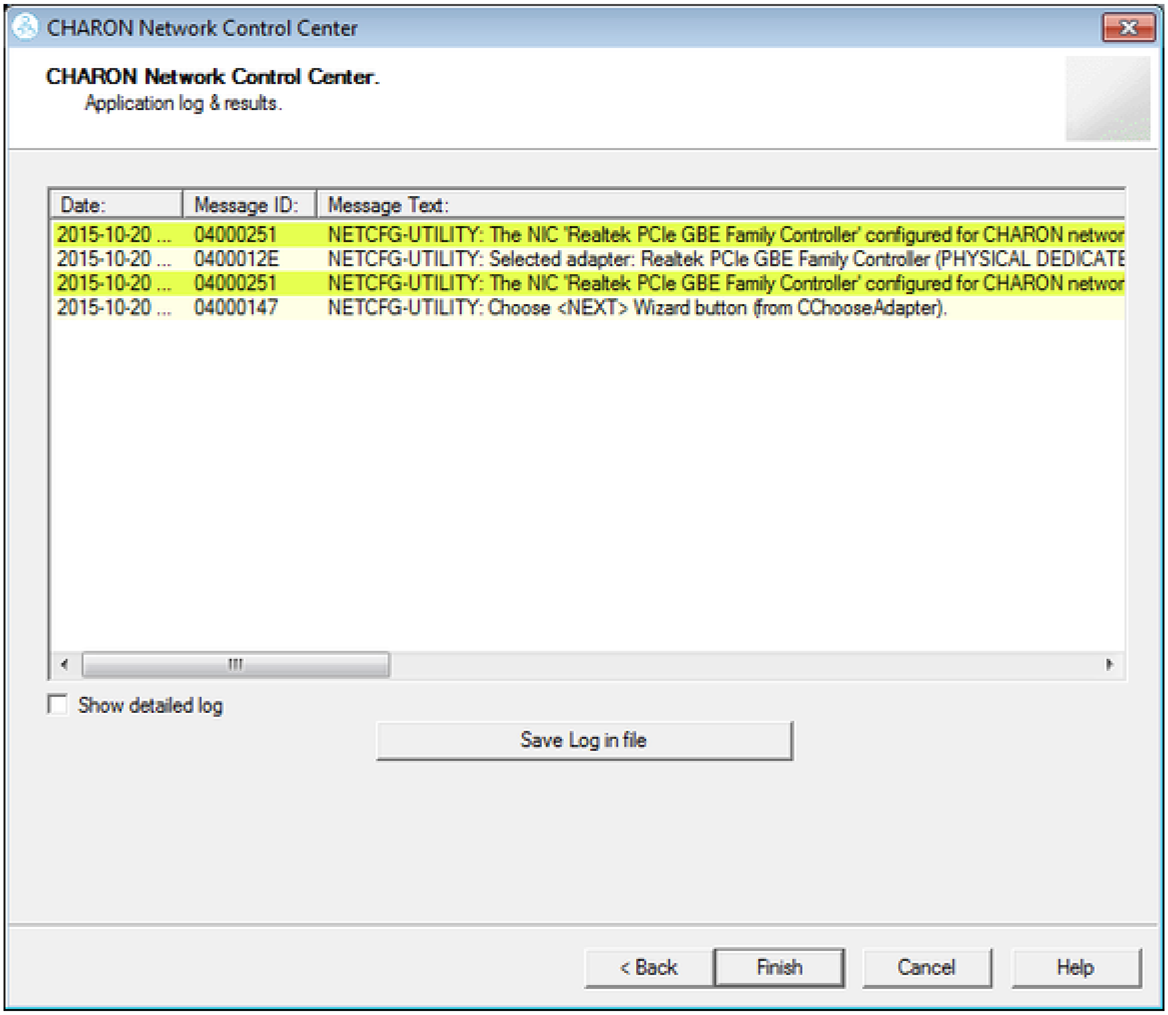
Statistics for adapter: Realtek PCIe GBE Family Controller

Statistics name	Packet statistics data
MAC address:	F8-D1-11-00-67-E6(DECNET N/A)
Users:	1
Mode:	CHPACK
NIC_Carrier:	On/2
NIC_Reset:	On/0
Pending_RX:	0 (16256)
Pending_TX:	0 (0)
Served_RX:	96 (17153)
Served_TX:	40 (877)
Failed_TX_IO:	0 (0)
Failed_TX_IR:	0 (0)
Failed_TX_RS:	0 (0)
Failed_TX_BU:	0 (0)
Failed_TX_DS	0 (0)
Failed_RX_IO:	0 (0)
Failed_RX_IR:	0 (0)
Failed_RX_RS:	0 (0)
Failed_RX_BU:	0 (0)
Failed_RX_DS:	0 (0)

Log file write interval (sec): 10

Log file:

Press the "Next" button to stop the recording and to see the log of this operation:


CHARON Network Control Center
CHARON Network Control Center.
Application log & results.
Save Log in file
< Back
Finish
Cancel
Help

Date:	Message ID:	Message Text:
2015-10-20 ...	04000251	NETCFG-UTILITY: The NIC 'Realtek PCIe GBE Family Controller' configured for CHARON networ
2015-10-20 ...	0400012E	NETCFG-UTILITY: Selected adapter: Realtek PCIe GBE Family Controller (PHYSICAL DEDICATE
2015-10-20 ...	04000251	NETCFG-UTILITY: The NIC 'Realtek PCIe GBE Family Controller' configured for CHARON networ
2015-10-20 ...	04000147	NETCFG-UTILITY: Choose <NEXT> Wizard button (from CChooseAdapter).

☐ Show detailed log

Select "Show detailed log" for more detail.

Press the "Finish" button to exit.

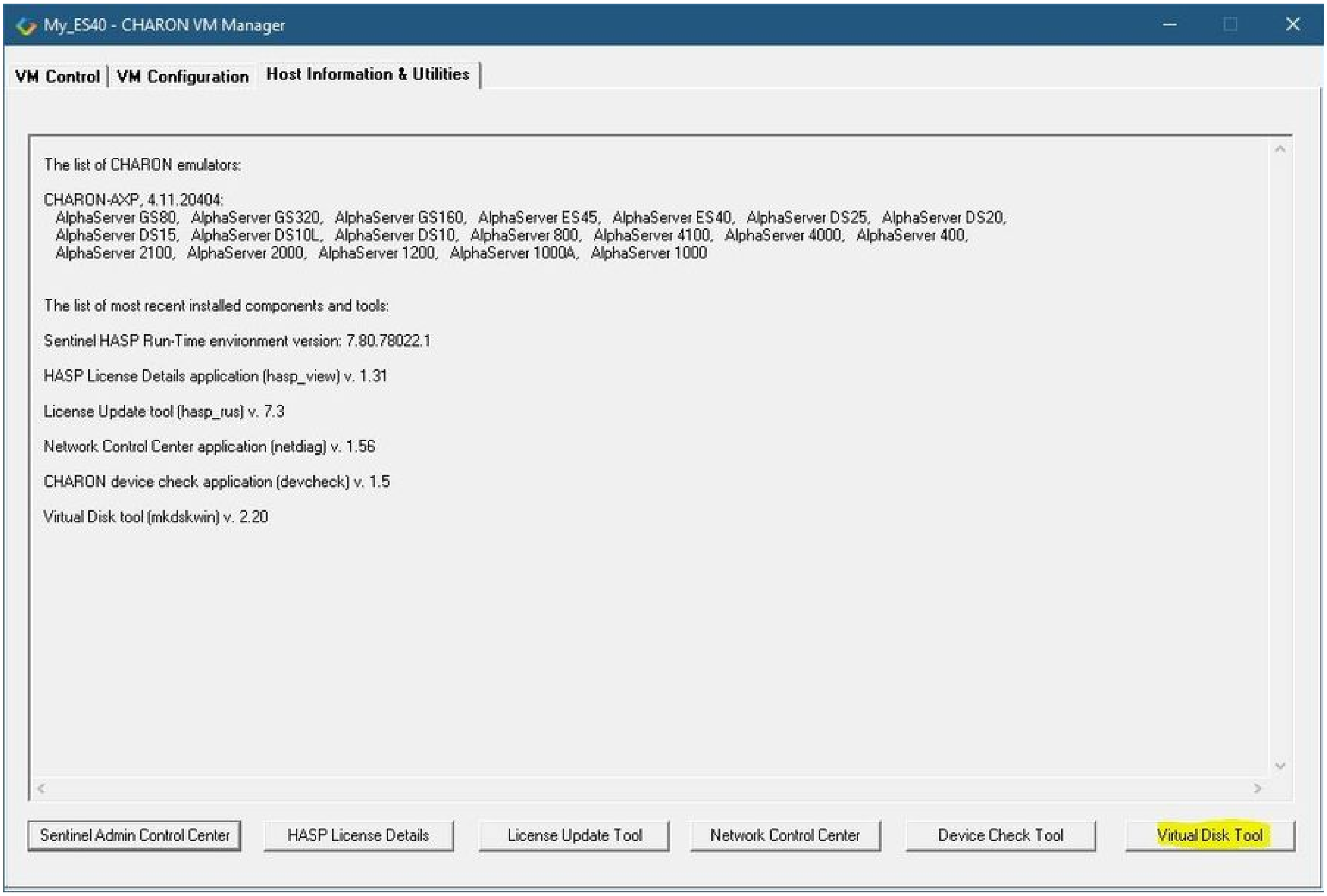
MkDisk

Table of Contents

General Description

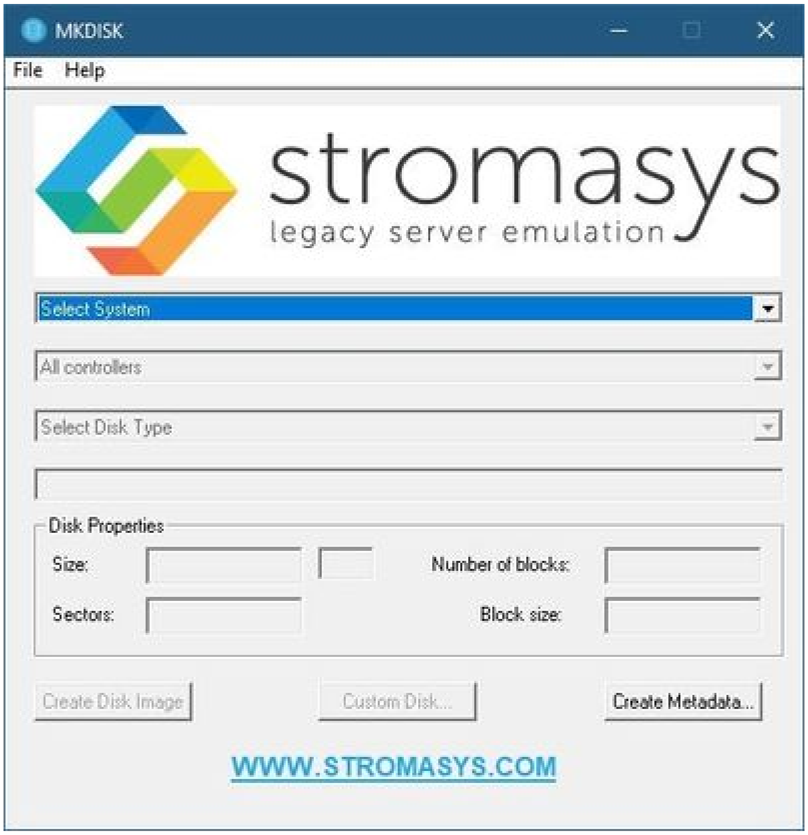
The "MkDisk" utility is used to create CHARON standard or custom empty disk containers (disk images).

Open the CHARON Virtual Machine Manager, switch to the "Host Information & Utilities" tab and press the "Virtual Disk Tool" button:



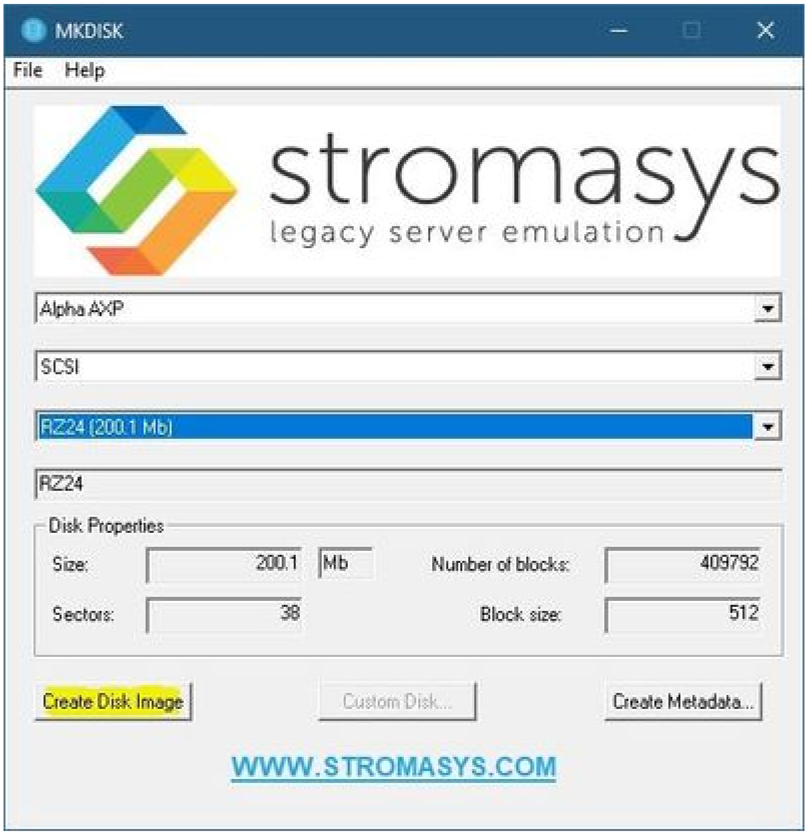
```
The list of CHARON emulators:

CHARON-AXP, 4.11.20404:
  AlphaServer GS80,  AlphaServer GS320,  AlphaServer GS160,  AlphaServer ES45,  AlphaServer ES40,  AlphaServer DS25,  AlphaServer DS20,
  AlphaServer DS15,  AlphaServer DS10L,  AlphaServer DS10,  AlphaServer 800,  AlphaServer 4100,  AlphaServer 4000,  AlphaServer 400,
  AlphaServer 2100,  AlphaServer 2000,  AlphaServer 1200,  AlphaServer 1000A,  AlphaServer 1000


The list of most recent installed components and tools:

Sentinel HASP Run-Time environment version: 7.80.78022.1

HASP License Details application (hasp_view) v. 1.31

License Update tool (hasp_rus) v. 7.3

Network Control Center application (netdiag) v. 1.56

CHARON device check application (devcheck) v. 1.5

Virtual Disk tool (mkdskwin) v. 2.20
```

The main dialog of the utility will appear:



Creating empty disk container

Select "Alpha AXP" in the "Select System" drop-down list, "SCSI" in the "All Controllers" drop-down one and choose the desired disk type in the "Select Disk Type" drop-down list to create an empty disk container:



Alpha AXP

SCSI

RZ24 (200.1 Mb)

RZ24

Disk Properties

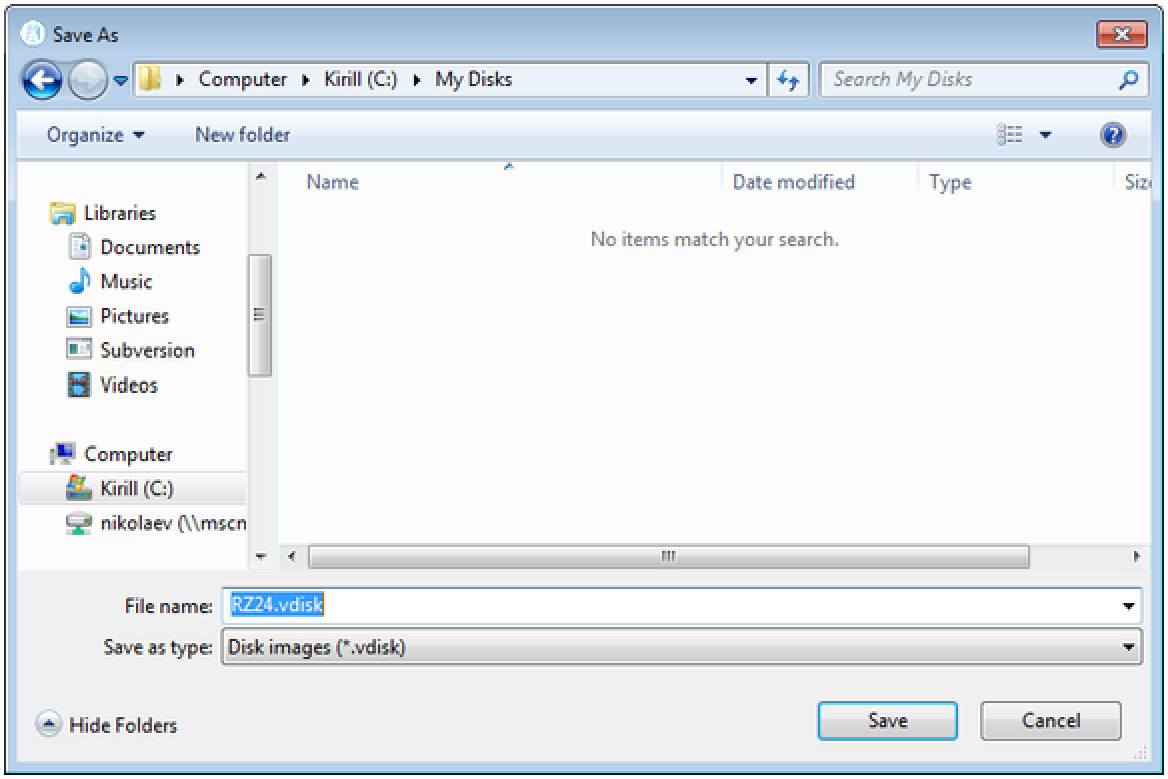
Size: 200.1 Mb

Number of blocks: 409792

Sectors: 38

Block size: 512

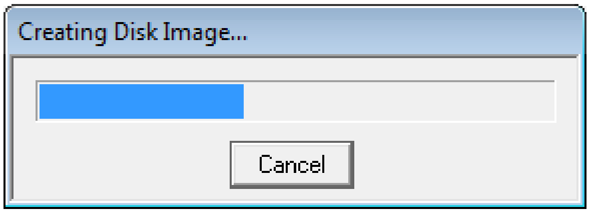
Press the "Create Disk Image" button to proceed. A dialog asking to specify the name of the disk image will appear:



File name: RZ24.vdisk

Save as type: Disk images (*.vdisk)

Browse to the target directory, specify the name of the disk image and press the "Save" button. The process of creating the disk container will start:



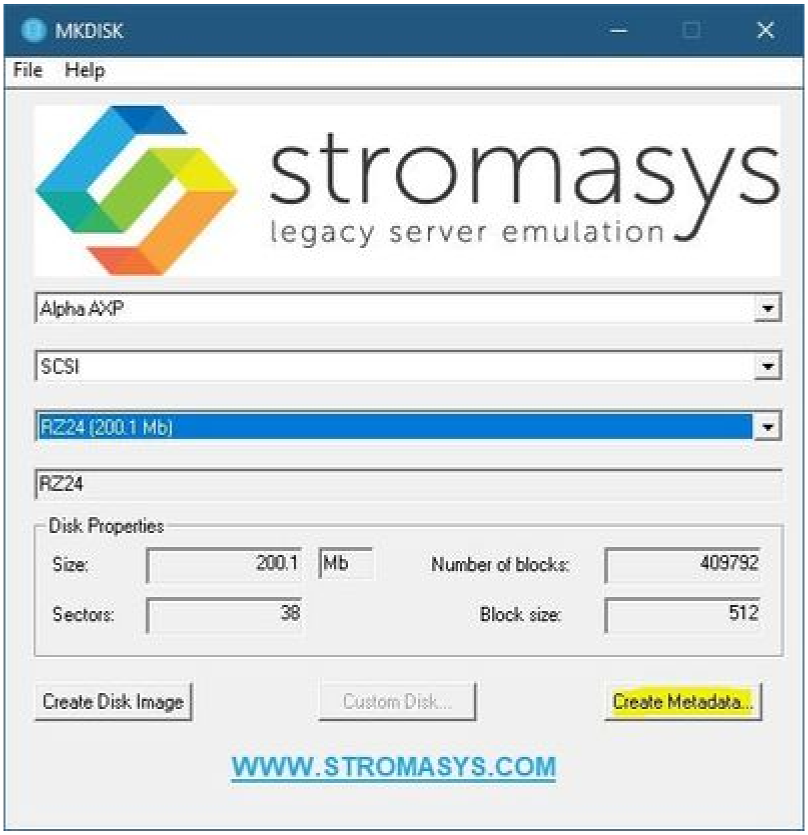
The target disk image is created - along with its metadata file (see below) once the process completes.

Creating disk metadata

The disk image metadata file contains important information about the target disk image structure - it is used by CHARON, automatically created with the same name with a different extension.

This functionality is needed if you have a disk image of a certain type and you want to generate a metadata file for it.

To create a metatada file press the "Create Metadata..." button:



Alpha AXP

SCSI

RZ24 (200.1 Mb)

RZ24

Disk Properties

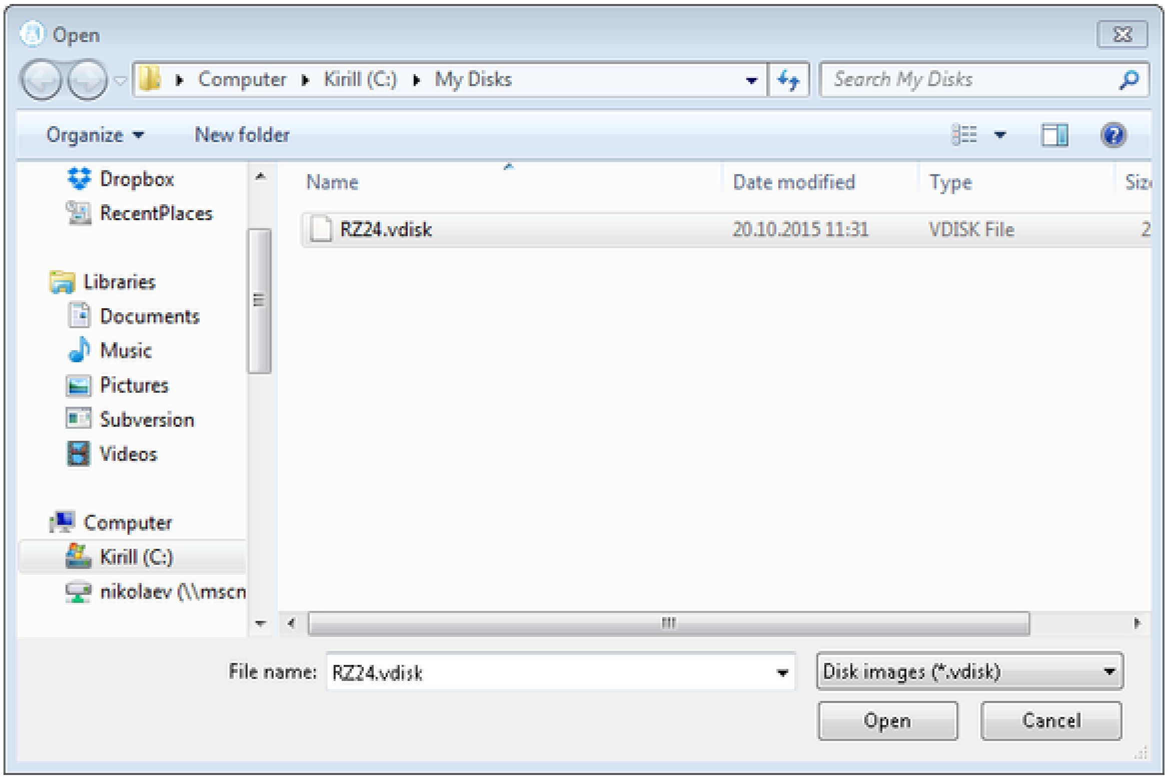
Size: 200.1 Mb

Number of blocks: 409792

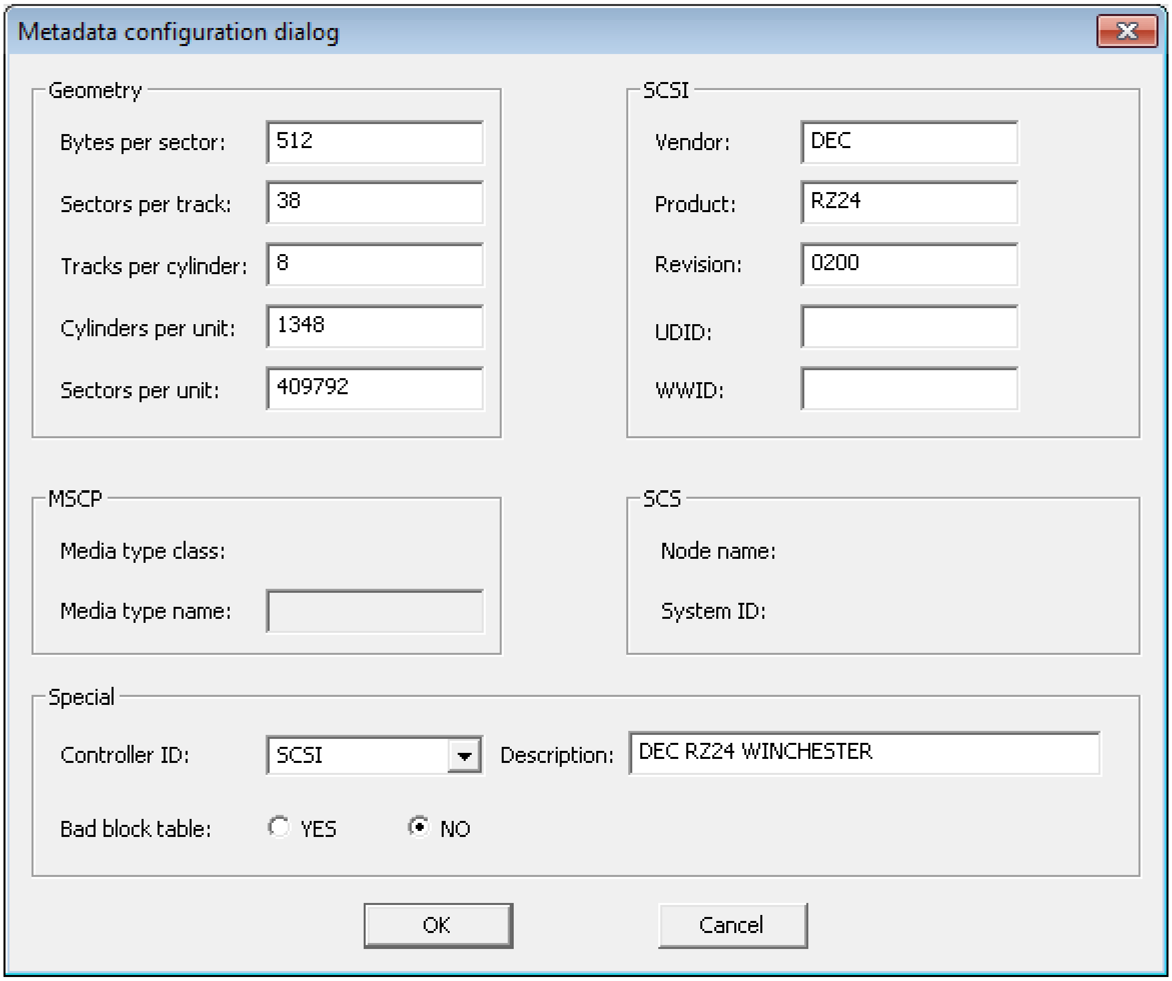
Sectors: 38

Block size: 512

The "MkDisk" utility will ask you to select a disk container for which to create the metadata file:



Select the target disk image and press the "Open" button. The utility will display a dialog for adjusting/correcting the disk image parameters:



Geometry

Bytes per sector: 512

Sectors per track: 38

Tracks per cylinder: 8

Cylinders per unit: 1348

Sectors per unit: 409792

SCSI

Vendor: DEC

Product: RZ24

Revision: 0200

UDID:

WWID:

MSCP

Media type class:

Media type name:

SCS

Node name:

System ID:

Special

Controller ID: SCSI

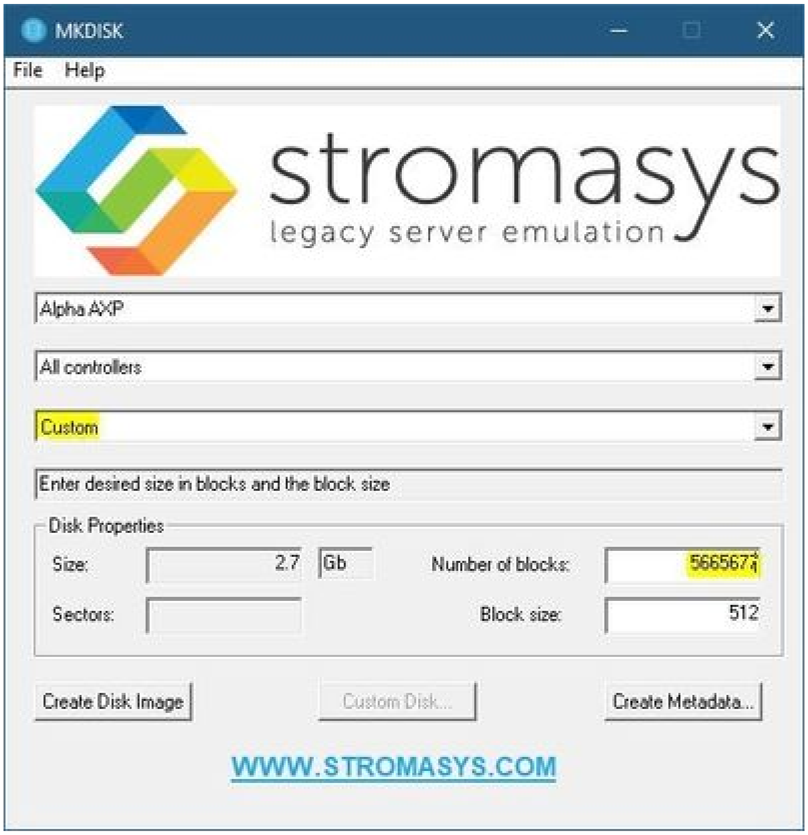
Description: DEC RZ24 WINCHESTER

Bad block table: ○ YES ● NO

Correct the desired parameters if needed and press the "Ok" button. The "MkDisk" utility will create a metadata file, having the same name as the target disk container, with extension ".avdisk"

Creating custom disk image

Select "Custom" in the "Select System" drop-down list, enter the desired number of blocks in the "Number of blocks" input box, specify the block size (if it is different from the default value of 512) in the "Block size" input box and press the "Create Disk Image" button to proceed:



Alpha AXP

All controllers

Custom

Enter desired size in blocks and the block size

Disk Properties

Size: 2.7 Gb

Number of blocks: 5665671

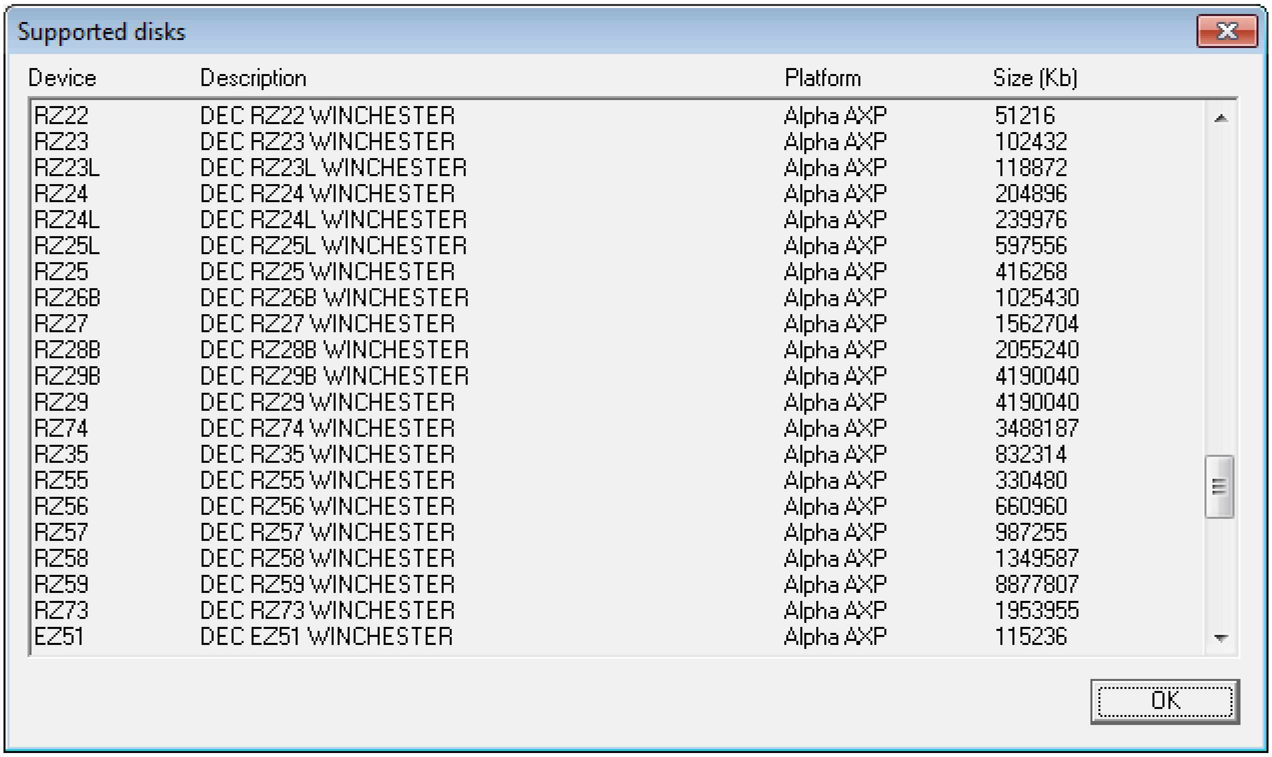
Sectors:

Block size: 512

Specify the location and name of the target disk container and press the "Save" button. The utility will create the disk image and will open the "Metadata Configuration dialog" window (for the .avdisk) → click OK to continue.

Getting information about available disk types

Information about the available disk types the "MkDisk" utility is able to create can be obtained by selecting the "Help->Disk Table" menu item. The following information dialog will appear:



Device	Description	Platform	Size (Kb)
RZ22	DEC RZ22 WINCHESTER	Alpha AXP	51216
RZ23	DEC RZ23 WINCHESTER	Alpha AXP	102432
RZ23L	DEC RZ23L WINCHESTER	Alpha AXP	118872
RZ24	DEC RZ24 WINCHESTER	Alpha AXP	204896
RZ24L	DEC RZ24L WINCHESTER	Alpha AXP	239976
RZ25L	DEC RZ25L WINCHESTER	Alpha AXP	597556
RZ25	DEC RZ25 WINCHESTER	Alpha AXP	416268
RZ26B	DEC RZ26B WINCHESTER	Alpha AXP	1025430
RZ27	DEC RZ27 WINCHESTER	Alpha AXP	1562704
RZ28B	DEC RZ28B WINCHESTER	Alpha AXP	2055240
RZ29B	DEC RZ29B WINCHESTER	Alpha AXP	4190040
RZ29	DEC RZ29 WINCHESTER	Alpha AXP	4190040
RZ74	DEC RZ74 WINCHESTER	Alpha AXP	3488187
RZ35	DEC RZ35 WINCHESTER	Alpha AXP	832314
RZ55	DEC RZ55 WINCHESTER	Alpha AXP	330480
RZ56	DEC RZ56 WINCHESTER	Alpha AXP	660960
RZ57	DEC RZ57 WINCHESTER	Alpha AXP	987255
RZ58	DEC RZ58 WINCHESTER	Alpha AXP	1349587
RZ59	DEC RZ59 WINCHESTER	Alpha AXP	8877807
RZ73	DEC RZ73 WINCHESTER	Alpha AXP	1953955
EZ51	DEC EZ51 WINCHESTER	Alpha AXP	115236

MkDskCmd

Table of Contents

General Description

The "`MkDskCmd`" utility:

- Creates empty disk images of a given standard or custom disk type or a custom disk size
- Transfers existing disk images of one type to disk images of another type.

To start the utility open "`cmd.exe`" from Administrator in the Start menu and switch to the CHARON installation folder then the "`\Build_XXX\xXX`" child folder where the utility is located.

Creating disk images

The first step is to obtain the disk type of the disk that needs to be created:

```
...> mkdskcmd -list
```

This command results in a list of all the supported disk types.

Choose the desired disk type (for example "RZ22") then use the "`mkdskcmd`" command to create the virtual disk image as shown below:

```
...> mkdskcmd -disk rz22 -output rz22.vdisk
```

A disk container "rz22.vdisk" will be created in the current directory.

A file "rz22.avdisk" will also be created. This file helps CHARON to accurately recognize a specific disk image type. It is recommended to put the ".avdisk" file in the same directory as the created disk image.

It is also possible to create custom disk images using the "`-blcount`" (blocks count) and "`-blsize`" (blocks size) switches.

For a list of all available parameters use the "–help" switch:

```
...> mkdskcmd -help

mkdisk for CHARON utility v. 1.17
Copyright: STROMASYS, 2020

Usage:
     mkdskcmd [Options]

Options:
    -h, -help - display help screen

    -o, -output <file> - specify output file name

    -d, -disk <name> - specify the disk name from Disk table

    -z, -blsize <value> - specify the block size in bytes (custom disk image)

    -c, -blcount <value> - specify number of the blocks (custom disk image)

    -a, -avtable <file> - specify AVDISK table file

    -r, -resize <file> [<disk-name>]
           - resize the disk image
       <file> - file name of the disk image to be resized
       <disk-name> - name of the disk from the Disk table

      <file> will either have the specified number of blocks added to the
      end or be truncated at the new smaller size.

      To specify a custom disk size, use the following parameters:
      -resize <file> -blsize <value> -blcount <value>

   -s, -shrink - mandatory parameter when resizing to smaller disk

   -l, -list - to display AVDISK table

   -q, -quiet - run in quiet mode

Return value:
   0 - Success
   Non zero - Failure

Examples:
mkdskcmd -help
mkdskcmd -list
mkdskcmd -avtable mkdsk.vtable -output rk07.vdisk -disk rk07
mkdskcmd -output disks\custom.vdisk -blsize 512 -blcount 16384
mkdskcmd -r E:\disks\rz22.vdisk rz59 -a "C:\Program Files\CHARON\disks\mkdsk.vtable"
```

The "`-avtable`" parameter is used to work with an alternative disk specification database or to point to the standard database ("`mkdsk.vtable`") if it is in a location other than the current directory.

The "`-blcount`" (blocks count) and "`-blsize`" (blocks size) switches are used to create custom disk images.

Resizing disk images

The "`MkDskCmd`" utility is able to resize (copy) disk images of one type to a disk image of another type.

This operation is needed, for example, to obtain more free space on a disk image that already contains data.

Notes:

- It is not possible to add more free space dynamically. The virtual machine must be stopped before performing this operation.
- Resizing a disk image requires the operating system running on the Charon virtual machine to be able to handle Dynamic Volume Expansion. Please refer to the documentation of your operating system version. If this is not supported, please create a new virtual disk then backup and restore the existing data.

ⓘ If a source disk image is larger than the target disk image, the extra data is lost. If the source disk image is smaller, it will be extended and padded with null bytes ('\0').

An example of the syntax follows:

```
...> mkdskcmd -resize <source disk file name> <source disk parameters> [-shrink] [-k]
```

where:

- <source disk file name> - the file name of the disk image to be resized
- <source disk parameters> - the disk type taken from the list of available disk types displayed by the "`mkdskcmd -list`" command or the disk geometry specification (see below).
- `-shrink` or `-k` - used to force the shrink when the target disk size is smaller than the source disk size.

Example:

```
...> mkdskcmd -resize "C:\My Disks\rz22.vdisk" rz25
```

It is also possible to specify the disk parameters manually with "`-blcount / -c`" (blocks count) and "`-blsize / -z`" (blocks size) switches:

```
...> mkdskcmd -resize <source disk file name> -blsize <number> -blcount <number>
```

Example:

```
...> mkdskcmd -r "C:\My Disks\custom.vdisk" -z 512 -c 262134
```

ⓘ There is a certain delay between the moment when the utility reports that a disk image has been resized and its actual availability to CHARON. This delay can reach up to several minutes in the case of very large disk resizes. This is because the host operating system needs time for the actual allocation of the enlarged file on HDD.

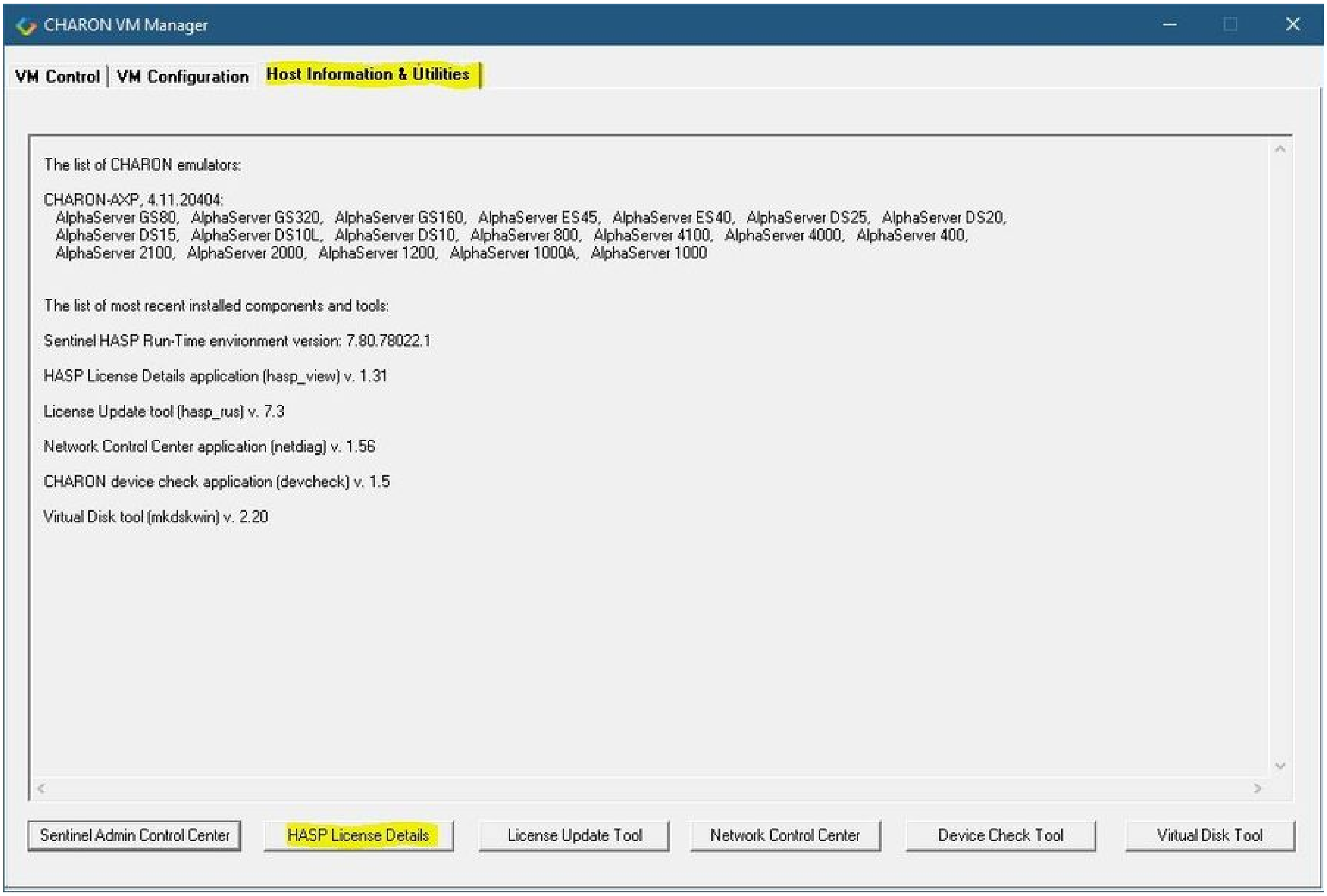
HASP View

Table of Contents

General Description

The "HASP View" utility is used to display the CHARON license content.

Open the CHARON Virtual Machine Manager, switch to the "Host Information & Utilities" tab and press the "HASP License Details" button:

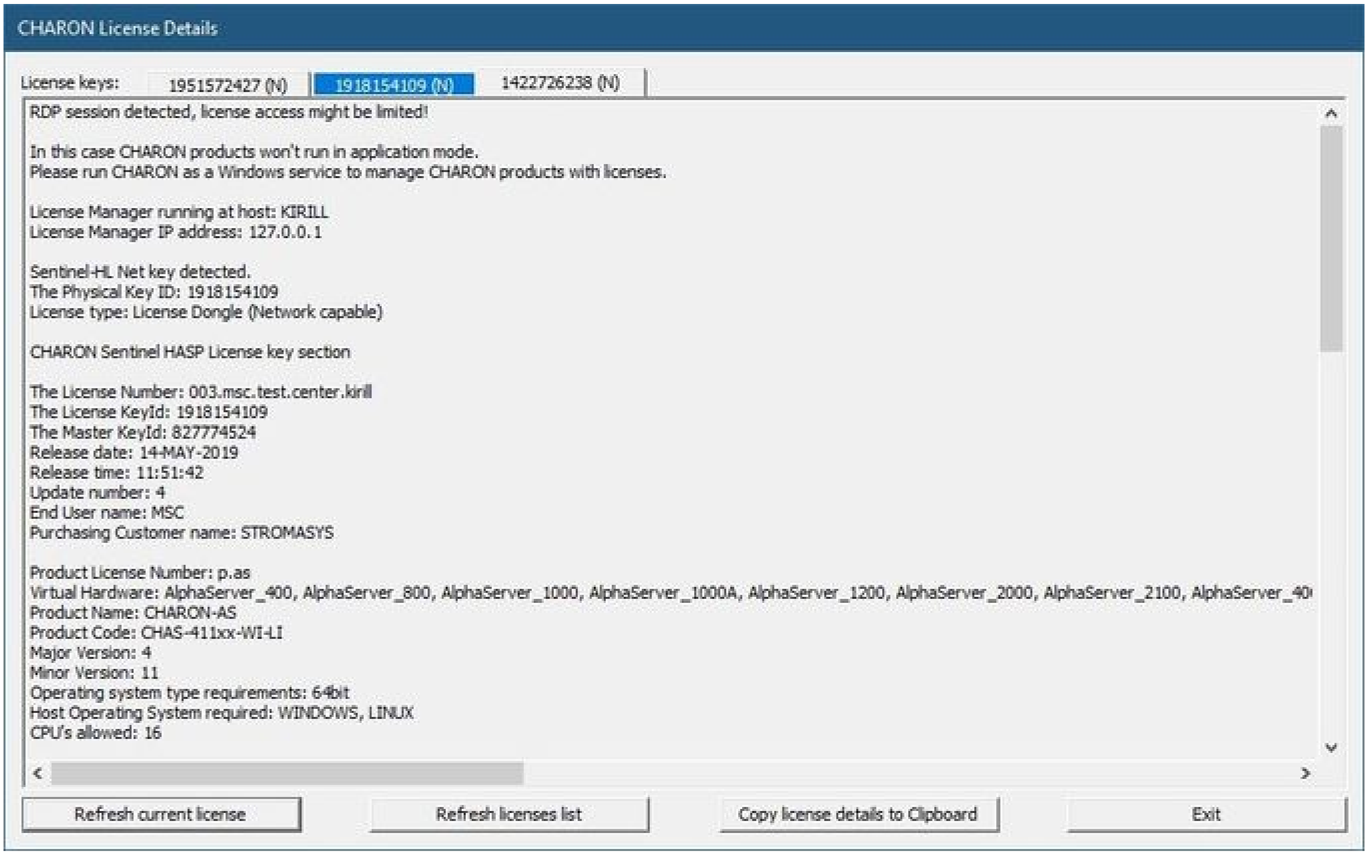


```
The list of CHARON emulators:

CHARON-AXP, 4.11.20404:
  AlphaServer GS80,  AlphaServer GS320,  AlphaServer GS160,  AlphaServer ES45,  AlphaServer ES40,  AlphaServer DS25,  AlphaServer DS20,
  AlphaServer DS15,  AlphaServer DS10L,  AlphaServer DS10,  AlphaServer 800,  AlphaServer 4100,  AlphaServer 4000,  AlphaServer 400,
  AlphaServer 2100,  AlphaServer 2000,  AlphaServer 1200,  AlphaServer 1000A,  AlphaServer 1000


The list of most recent installed components and tools:

Sentinel HASP Run-Time environment version: 7.80.78022.1

HASP License Details application (hasp_view) v. 1.31

License Update tool (hasp_rus) v. 7.3

Network Control Center application (netdiag) v. 1.56

CHARON device check application (devcheck) v. 1.5

Virtual Disk tool (mkdskwin) v. 2.20
```

Getting CHARON licenses content

The "HASP View" utility displays the content of the licenses to be used by CHARON.

Example:



```
RDP session detected, license access might be limited!

In this case CHARON products won't run in application mode.
Please run CHARON as a Windows service to manage CHARON products with licenses.

License Manager running at host: KIRILL
License Manager IP address: 127.0.0.1

Sentinel-HL Net key detected.
The Physical Key ID: 1918154109
License type: License Dongle (Network capable)

CHARON Sentinel HASP License key section

The License Number: 003.msc.test.center.kirill
The License KeyId: 1918154109
The Master KeyId: 827774524
Release date: 14-MAY-2019
Release time: 11:51:42
Update number: 4
End User name: MSC
Purchasing Customer name: STROMASYS

Product License Number: p.as
Virtual Hardware: AlphaServer_400, AlphaServer_800, AlphaServer_1000, AlphaServer_1000A, AlphaServer_1200, AlphaServer_2000, AlphaServer_2100, AlphaServer_40
Product Name: CHARON-AS
Product Code: CHAS-411xx-WI-LI
Major Version: 4
Minor Version: 11
Operating system type requirements: 64bit
Host Operating System required: WINDOWS, LINUX
CPU's allowed: 16
```

Use the "License keys" tabs to view the content of the CHARON licenses found.

Press the "Refresh key list" button to look for newly added and available licenses.

Note: collecting the license content may take some time. Use the buttons located at the bottom of the window to refresh the license content and to copy the license details to the clipboard if needed.

Press the "Exit" button to exit from the utility.

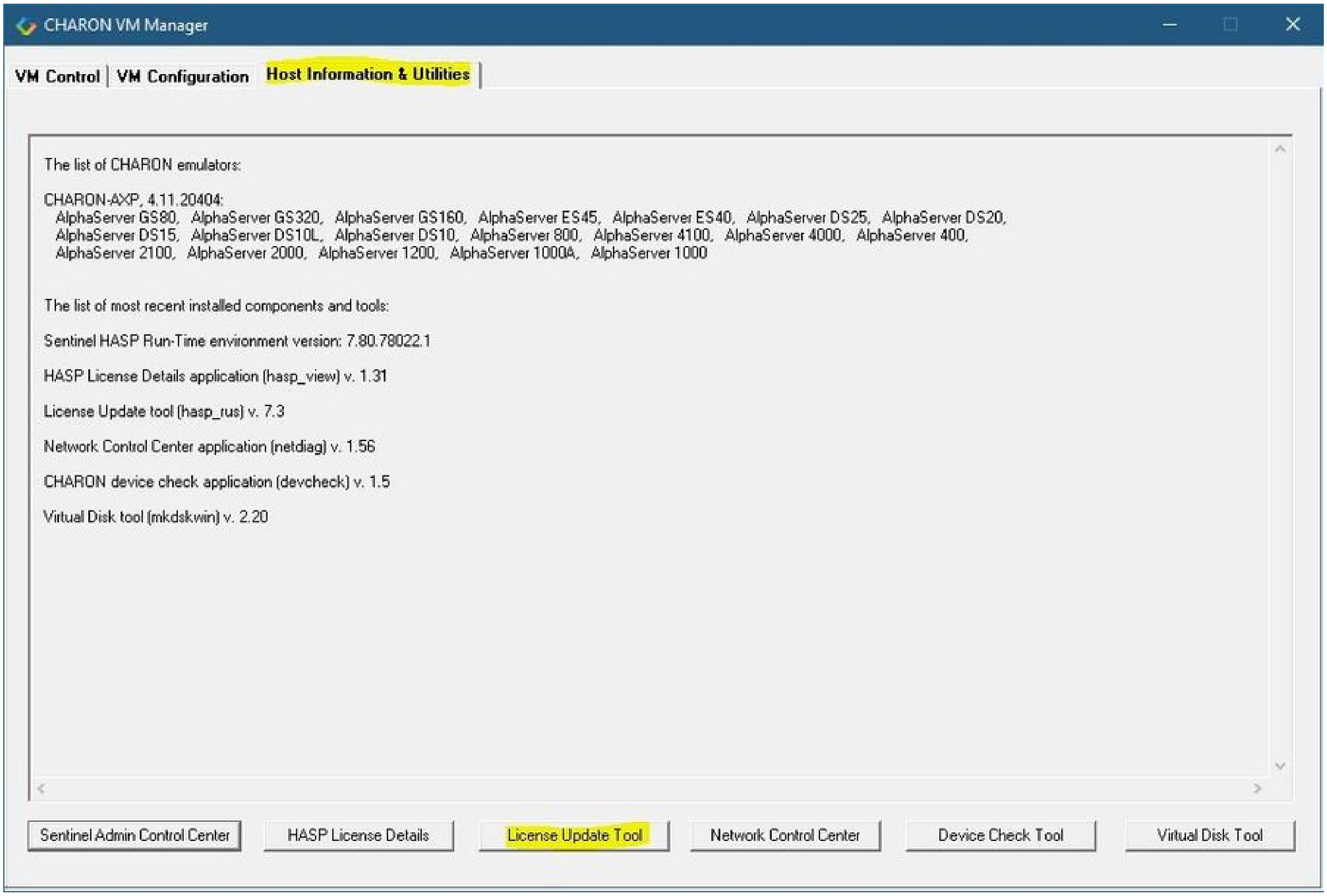
License Update Service

Table of Contents

General Description

The "License Update Service" utility is used to manage CHARON licenses, collect the host system fingerprint.

Open the CHARON Virtual Machine Manager, switch to the "Host Information & Utilities" tab and press the "License Update Tool" button:

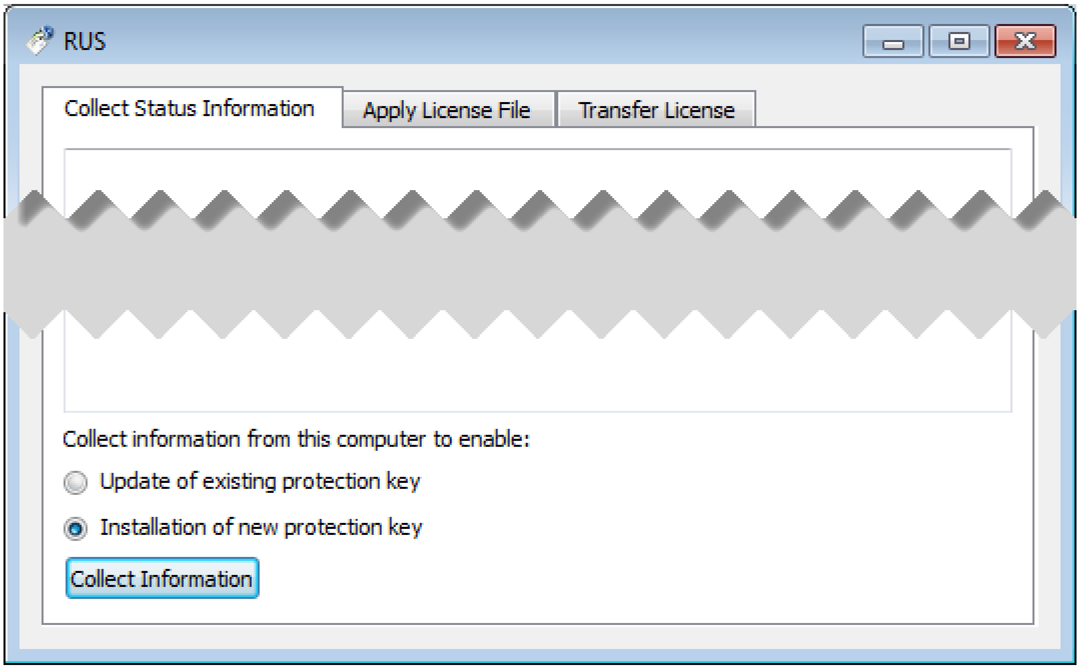


```
The list of CHARON emulators:

CHARON-AXP, 4.11.20404:
  AlphaServer GS80,  AlphaServer GS320,  AlphaServer GS160,  AlphaServer ES45,  AlphaServer ES40,  AlphaServer DS25,  AlphaServer DS20,
  AlphaServer DS15,  AlphaServer DS10L,  AlphaServer DS10,  AlphaServer 800,  AlphaServer 4100,  AlphaServer 4000,  AlphaServer 400,
  AlphaServer 2100,  AlphaServer 2000,  AlphaServer 1200,  AlphaServer 1000A,  AlphaServer 1000


The list of most recent installed components and tools:

Sentinel HASP Run-Time environment version: 7.80.78022.1

HASP License Details application (hasp_view) v. 1.31

License Update tool (hasp_rus) v. 7.3

Network Control Center application (netdiag) v. 1.56

CHARON device check application (devcheck) v. 1.5

Virtual Disk tool (mkdskwin) v. 2.20
```

Collecting the host fingerprint and information on an existing license

Open the "Collect Status Information" tab.

Select either "Update of existing protection key" to acquire information on the current hardware of software license or "Installation of new protection key" to get information on the host system.

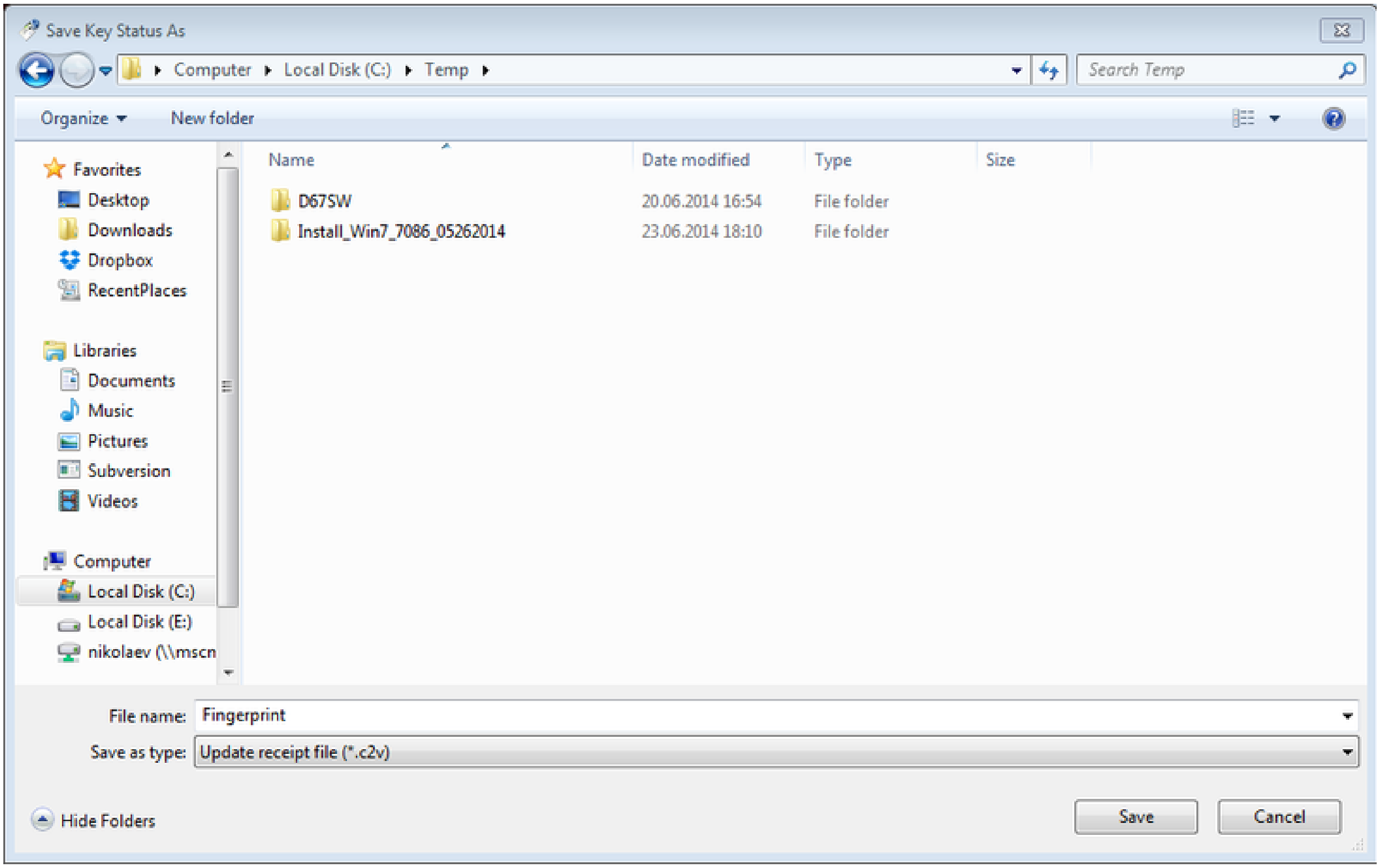


Collect information from this computer to enable:

○ Update of existing protection key

● Installation of new protection key

Press the "Collect Information" button and save the fingerprint or the information on the current license to a "*.c2v" file:



Name	Date modified	Type	Size
D67SW	20.06.2014 16:54	File folder	
Install_Win7_7086_05262014	23.06.2014 18:10	File folder	

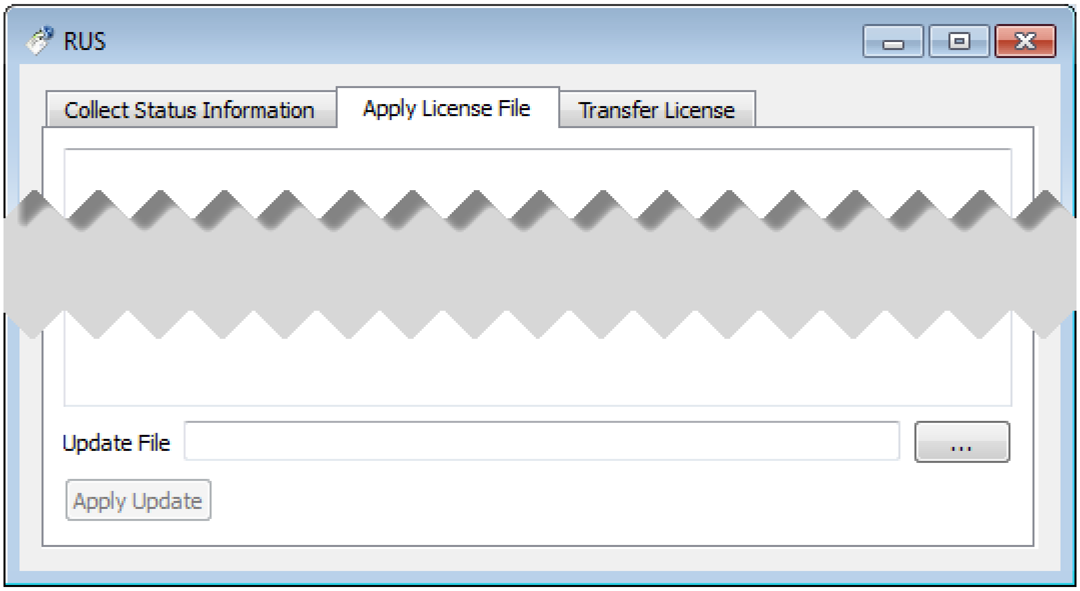
File name: Fingerprint

Save as type: Update receipt file (*.c2v)

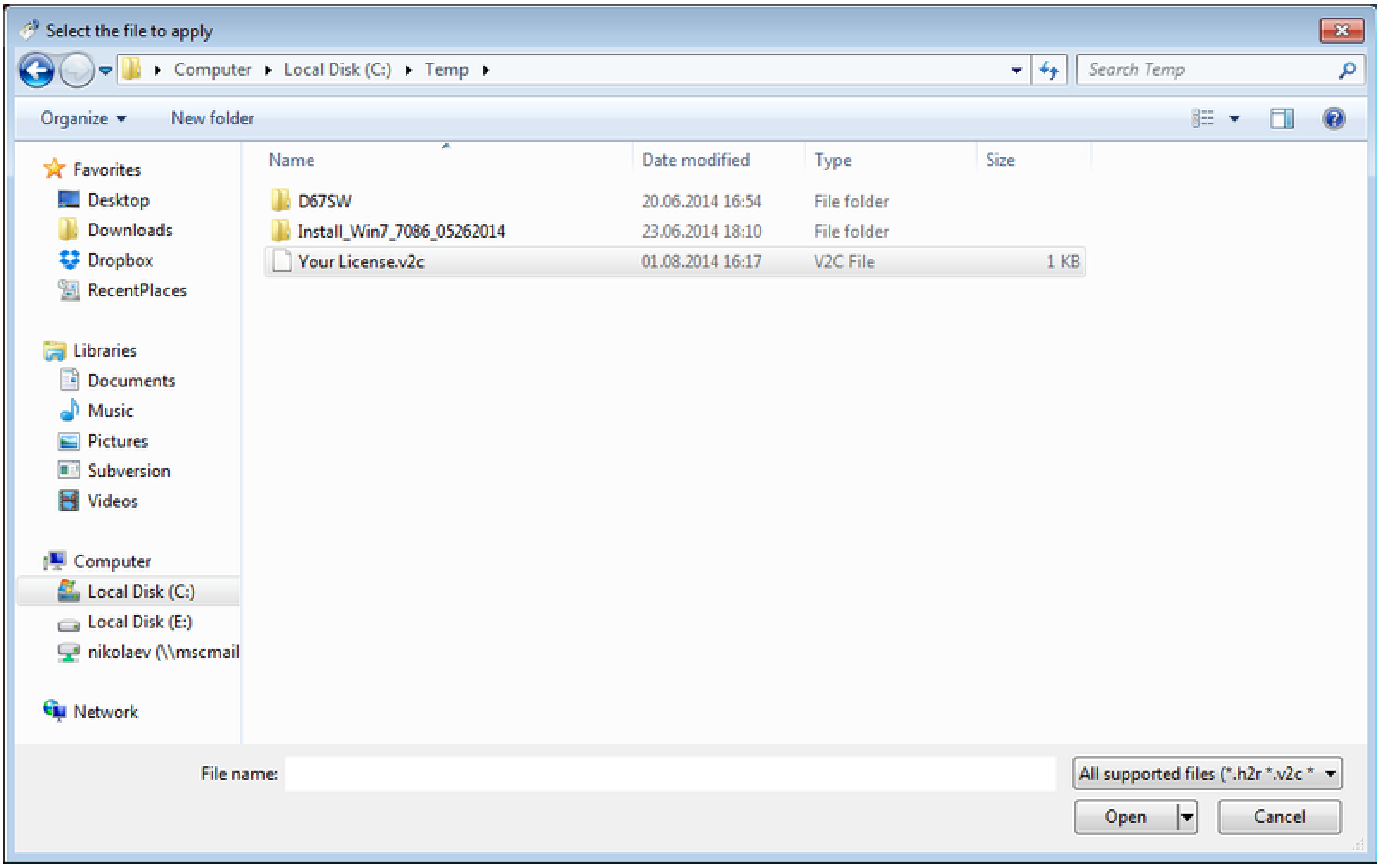
Press the "Save" button to create the "*.c2v" file and send this file to STROMASYS to receive an update of the current license or a new license.

Installation and update of a CHARON license

Open the "Apply License File" tab and press the "..." button:



Select the license "*.v2c" file received from STROMASYS:



Name	Date modified	Type	Size
D67SW	20.06.2014 16:54	File folder	
Install_Win7_7086_05262014	23.06.2014 18:10	File folder	
Your License.v2c	01.08.2014 16:17	V2C File	1 KB

File name:

All supported files (*.h2r *.v2c *

Press the "Open" button and then the "Apply Update" button in the main dialog box to apply the new license or to update an existing one.

ℹ In case you receive 2 files to update the license, install the "_fmt" (format) one first then the second one.

License expiration check

Introduction

Once setup, this utility will execute once a day at 09:00 AM and will send Windows Application Events 15 days before expiration. No other action will be taken unless you customize the `expiredlicense.bat` script. The utility can be found in the Charon installation folder under the "Redistributables\Stromasys" child folder.

The license expiration check utility is not provided with the CHARON-AXP V4.12 kit. If you need it, please ask us for the download link.

The installation process is described below. It is recommended not to update the example files given with the kit as they could be overwritten by a patch installation or product upgrade. It is preferable to copy them in a dedicated folder.

Version: 1.18 - **Published**: 17 Jan 2023

Table of contents

Description

The goal of this Powershell script is to read the license content and send alerts 15 days (by default) before expiration. This document is related to the kit versions 1.9 to 1.17.

This utility is replaced by CHARON on Windows - Charon License Utility that provides a graphical interface providing the Windows server has PowerShell 5.1 and .NET Framework 4.6 minimum installed (4.8 recommended) installed.

Alerts consist in:

- Adding a Windows Event entry in the Application log
- Executing a customized script named "`expiredlicense.bat`" that can execute actions at your convenience: send an email, send an event to a monitoring software, etc...

Alert levels:

Days before expiration (date limited license)	Hours before expiration (time limited license)	Alert level
Between 7 and number of days defined (15 by default)	Between 73 and 96 hours	Informational
Between 3 and 7	Between 49 and 72 hours	Warning
Less than 3 and expired	Less than 49 and expired	Error

Related products:

Charon-AXP and Charon-VAX/PDP V4.5 and above

Operating systems:

Windows 7, Windows 8.1, Windows 10 Professional, Windows Server 2008 R2, Windows Server 2012 (R2), Windows Server 2016, Windows Server 2019

Step-by-step guide

Installation

- Create a folder dedicated to store the scripts, for example: "`C:\Charon`"
- Download the kit from our SFTP server (please ask us connection credentials if you have no access) and extract all files in this folder.
  The kit contains some files, their usage is described further in this document.
- Optionally create the "`expiredlicense.bat`" file in the same folder and customize it at your own (to send emails for example).

  If the script does not exist, a warning event will be posted but it will not prevent the script from running. In this case only Windows Application events will be sent.

  Below is a simple example to append entries in a log file. You will find further in this article an example on how to send an email using powershell.

  ```
  echo %date% %time% %1 %2 >> C:\Charon\expiredlicense.log
  ```

  Notes:

  - "%1" corresponds to the severity level and can be "informational", "warning" or "error"
  - "%2" corresponds to the alert message containing the license number, the product name and the expiration status

Interactive check

- Open a command line window
- Move to the folder you created, example: `C:\Charon`
- Execute the powershell script. The following parameters can be passed to the script:
  - "`-folder`" is used to specify where the scripts (.ps1 and .bat) are stored. If not specified, default is "`C:\Charon`"
  - "`-nbdays`" is used to specify the number of days before expiration to start sending alerts, default is 15
  - "`-nbhours`" is used to specify the number of hours before expiration of a backup/spare license (time limited) to start sending alerts, default is 96. Must not be lower than 72.
  - "`-usevanilla`" is used in case the `hasp_view.exe` program is not able to provide the expiration date in the output file (bug found in builds 18304 and 18305 with `hasp_view.exe` version 1.26). In this case, the script will use the `hasp_view.exe` program located in the "`xxxxx_ORIGINALS_xxx_PATCH_xxx`" folder (example: `B18302_ORIGINALS_AXP_PATCH_X64`). The bug is solved with `hasp_view.exe` version 1.27 and later.
  - "`-haspview <hasp_view.exe location>`" is used to specify the location of the `hasp_view.exe` file. If this parameter is not specified then the script searches for the location itself depending on Charon installation folder. It is used in case a license server performs the checks instead of the server hosting the Charon virtual machines. The "`-usevanilla`" parameter is ignored if this parameter is defined.
  - "`-ini <file>`" is used to specify a file where the `-folder`, `-nbdays`, `-nbhours`, `-usevanilla`, `-haspview` and `-tick` parameters can be stored. Priority goes to the parameters passed in the command line then those in the specified file. The file is used to facilitate updates in the parameters passed in the tasks scheduler (update the file instead of the task).
  - "`-tick`" parameter is used to execute external script (`expiredlicense.bat`) even if no alert is detected (counter alert for monitoring software). If used in the .ini file, "1", "y", "yes" or "true" value activates it, other values will invalidate (**Example**: `tick=true`)

To run PowerShell scripts (files that end with .ps1), you must first set the execution policy to Unrestricted (This operation has to be done once).

To do so, open a command line window (cmd.exe) as an Administrator and use the following command:

```
c:\Charon>powershell -command "Set-ExecutionPolicy Unrestricted"
```

The ExecutionPolicy can also be set to "`RemoteSigned`". In this case the `.ps1` script files will have to be unblocked as described below.

If you are still prompted to allow for execution of the script, please run the following command to unblock the downloaded `charon_expchk.ps1` file:

```
c:\Charon>powershell -command "Unblock-File -path c:\charon\charon_expchk.ps1"
```

Example:

```
c:\Charon>powershell -file charon_expchk.ps1 -folder c:\charon -nbdays 20
```

If the license is not readable (dongle not connected for example), the script will end with an "Unable to generate the license view from hasp_view.exe!" error, an entry will be added to the Windows Event Application log and the `expiredlicense.bat` script will be executed.

If no script error is detected, it is now possible to create a new scheduled task.

Scheduled task creation

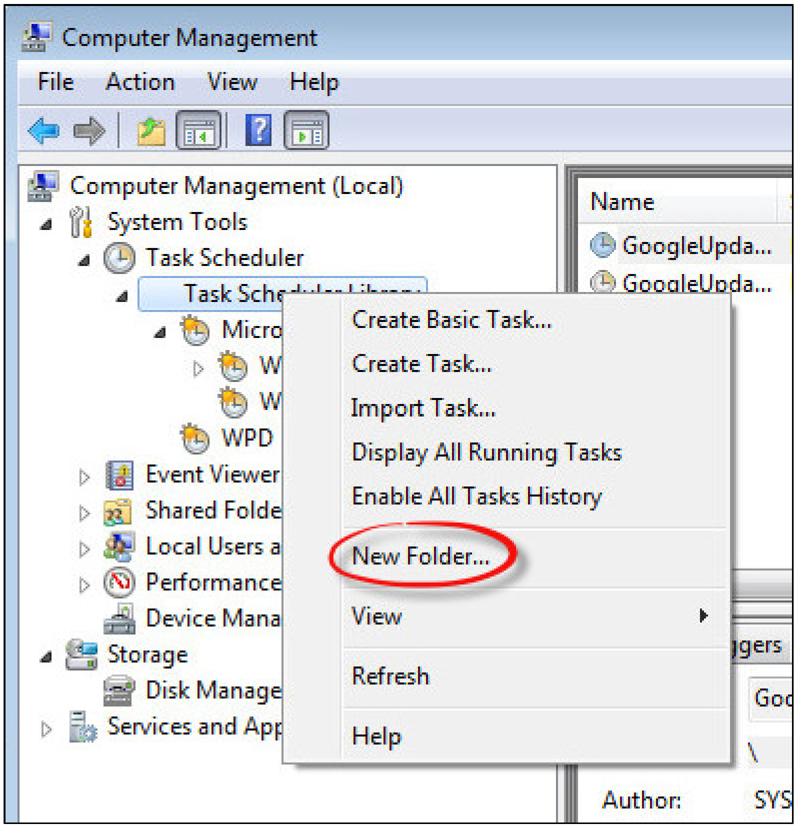
Scheduled task creation using the Graphical User Interface

To set this,

- either open the Administrative tools under Control Panel and select "**Task Scheduler**" or
- execute `taskschd.msc` from the Windows run command (press Windows key + R)

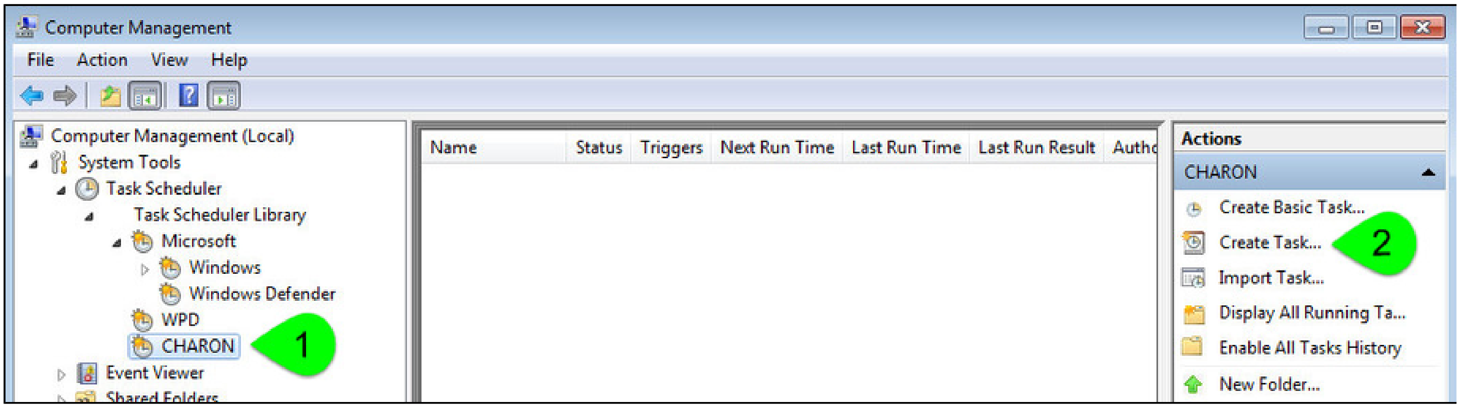
Examples are given based on a Windows 2008 R2 server

Right click on **Task Scheduler Library**, select New folder then enter **CHARON**.

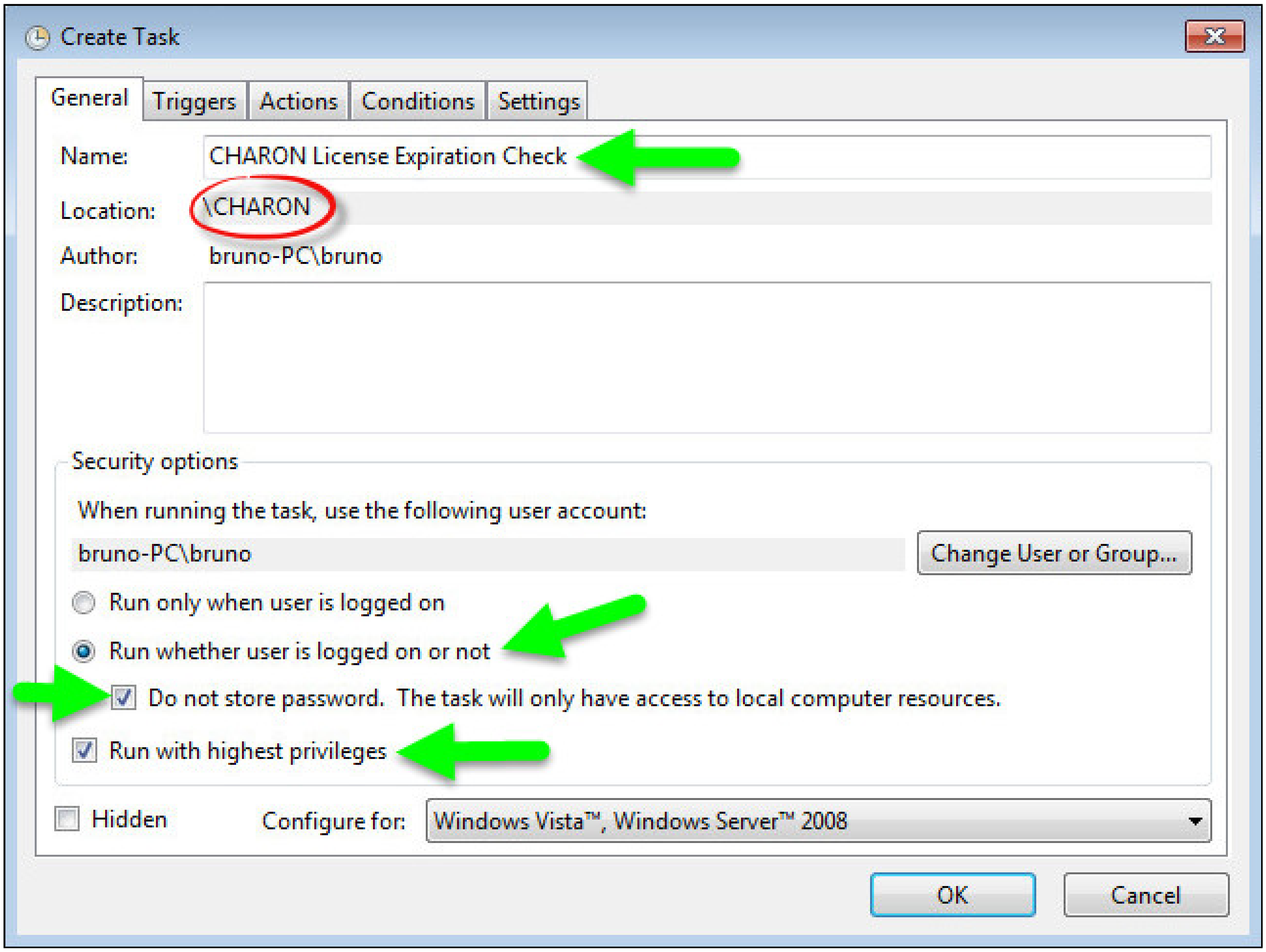


Select the newly created folder (**1**) and on the right pane, select "**Create Task...**" (**2**).

Do not use "Create Basic Task..." some of the options required aren't available in that wizard.



Fill the **General** tab as below:



Name: CHARON License Expiration Check

Location: \CHARON

Author: bruno-PC\bruno

Description:

Security options

When running the task, use the following user account:

bruno-PC\bruno

○ Run only when user is logged on

● Run whether user is logged on or not

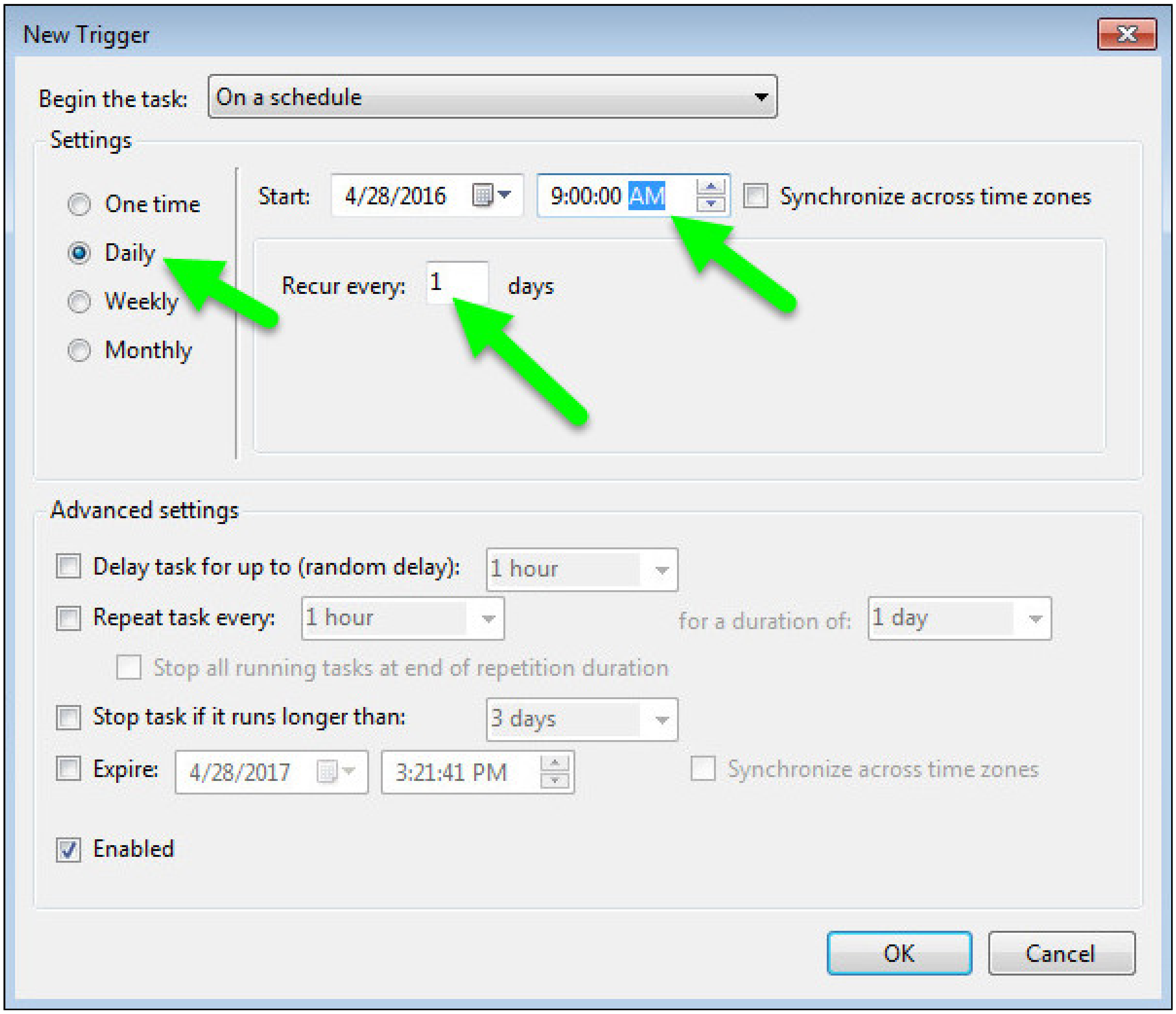
☑ Do not store password. The task will only have access to local computer resources.

☑ Run with highest privileges

□ Hidden

Configure for: Windows Vista™, Windows Server™ 2008

Select the **Triggers** tab and set the check interval, for example here: everyday at 09:00 AM



New Trigger

Begin the task: On a schedule

Settings

○ One time
● Daily
○ Weekly
○ Monthly

Start: 4/28/2016 9:00:00 AM □ Synchronize across time zones

Recur every: 1 days

Advanced settings

□ Delay task for up to (random delay): 1 hour

□ Repeat task every: 1 hour for a duration of: 1 day

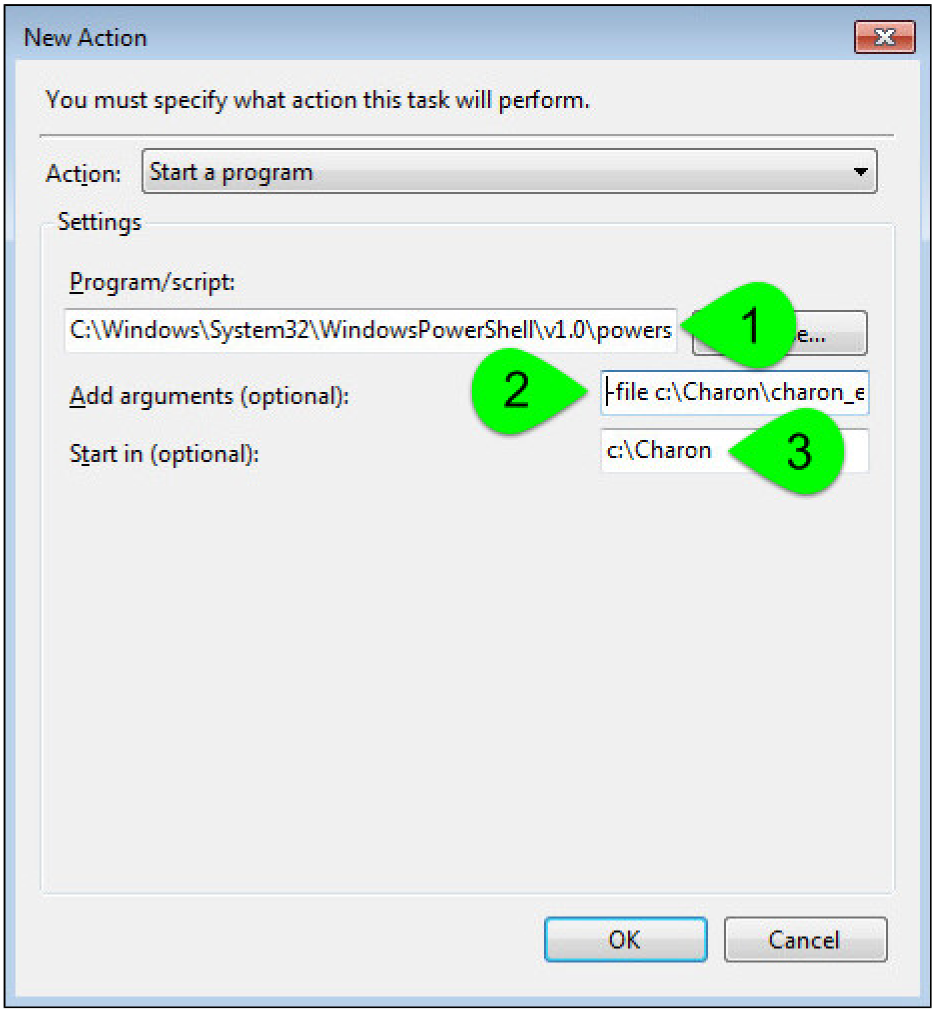
□ Stop all running tasks at end of repetition duration

□ Stop task if it runs longer than: 3 days

□ Expire: 4/28/2017 3:21:41 PM □ Synchronize across time zones

☑ Enabled

Select the **Actions** tab and configure powershell to be executed with the following parameters:



New Action

You must specify what action this task will perform.

Action: Start a program

Settings

Program/script: C:\Windows\System32\WindowsPowerShell\v1.0\powers

Add arguments (optional): -file c:\Charon\charon_e

Start in (optional): c:\Charon

(**1**) : Powershell location here is "`C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe`"

(**2**) : Define the powershell arguments to be added: `-file c:\Charon\charon_expchk.ps1 -folder c:\Charon` (the folder parameter is optional here as it is the same as the default folder)

(**3**) : Set the folder to be the one where the scripts reside

ℹ Once the task is created, it is recommended to right click on the task then "Run". To verify the script ran correctly, please check the existence of the "`charon_expchk.txt`" file in the installation folder (it should contain the license view, if the license is present) and check for events in the Windows Event Application log if any.

Scheduled task creation using the Character User Interface

Open a command line window (cmd.exe) and execute the following command:

```
c:\Charon>schtasks /create /SC DAILY /ST 09:00 /NP /RL HIGHEST /TN "CHARON\CHARON License Expiration Check" ^
More? /TR "C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -file c:\Charon\charon_expchk.ps1 -folder c:\Charon"
```

💡 Update the schedule type (/SC) and time (/ST) to your desired settings

Start the task as shown below:

```
c:\Charon>schtasks /run /TN "CHARON\CHARON License Expiration Check"
```

To verify the script ran correctly, please check the existence of the "`charon_expchk.txt`" file in the installation folder (it should contain the license view, if the license is present) and check for events in the Windows Event Application log if any.

Configuration file example

In the following example, the parameters/values are stored in the `c:\charon\charon_expchk.ini` file.

Example:

```
#Folder where the scripts are located
folder=C:\Charon

# Number of days before expiration alert
nbdays=21

# Location of the hasp_view.exe file if Charon is not installed
haspview=C:\Charon

# Send an alert even if no license is about to or has expired
tick=true
```

To execute the script with the .ini file, use the following command:

```
c:\Charon>powershell -file c:\charon\charon_expchk.ps1 -ini c:\charon\charon_expchk.ini
```

To create the scheduled task, open a command line window (cmd.exe) and execute the following command:

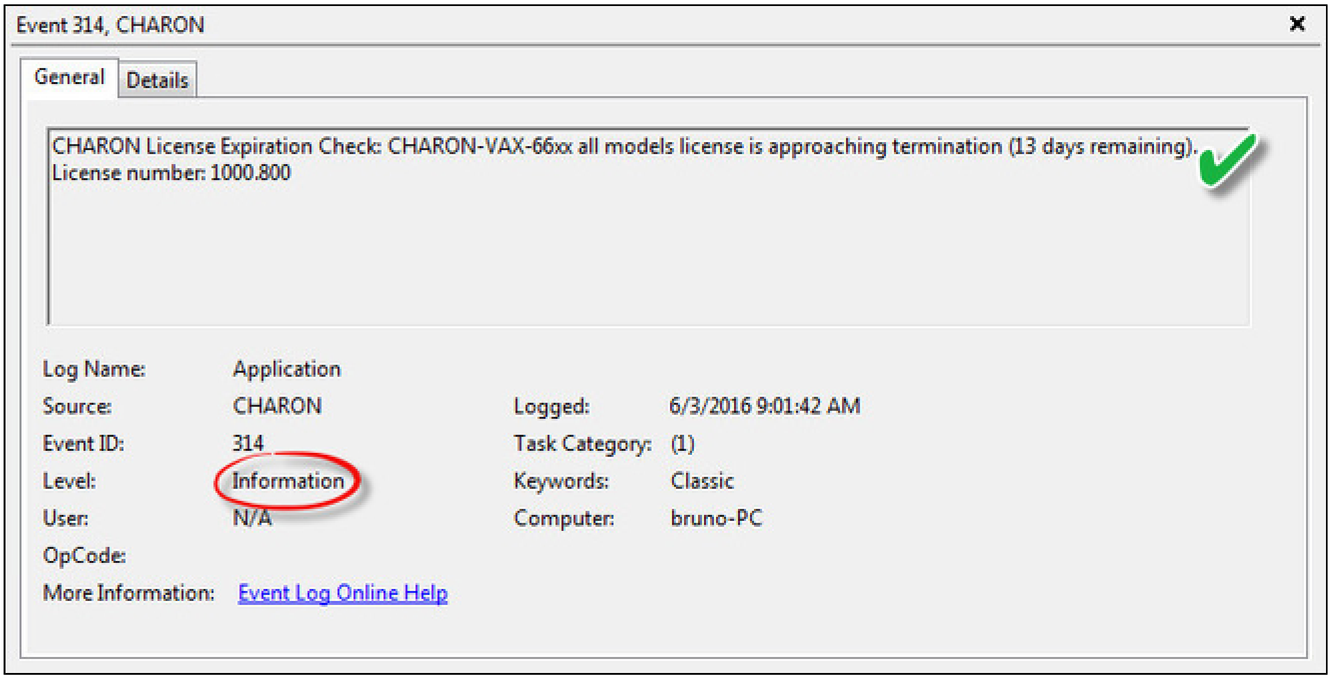
```
c:\Charon>schtasks /create /SC DAILY /ST 09:00 /NP /RL HIGHEST /TN "CHARON\CHARON License Expiration Check" ^
More? /TR "C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -file c:\Charon\charon_expchk.ps1 -ini c:\Charon\charon_expchk.ini"
```

Hints

💡 To verify the script is correctly working and to force the creation of a Windows event, and eventually the execution of the "`expiredlicense.bat`" file, you can:

- either play with the removal of the dongle (if there is, that is no Software License)
- or with the number of days (if the license is not unlimited)
- or you can specify a folder that does not exist.

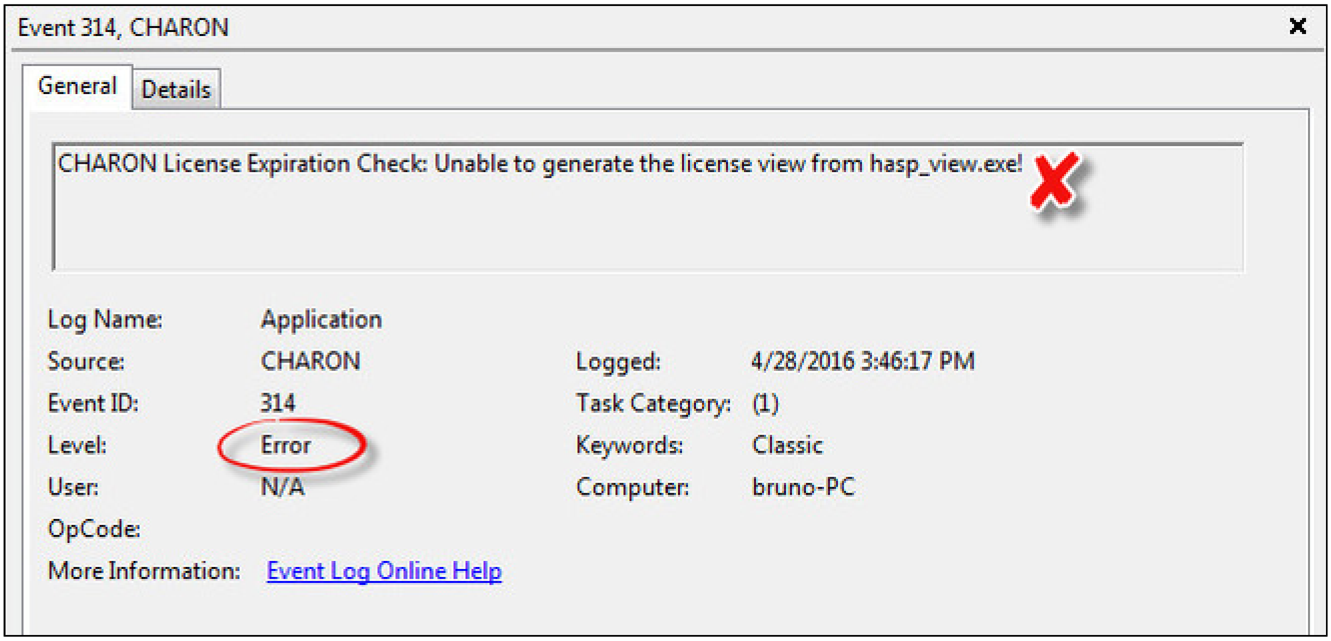
Windows Event Application log example - Information alert:



CHARON License Expiration Check: CHARON-VAX-66xx all models license is approaching termination (13 days remaining).
License number: 1000.800

Log Name:	Application		
Source:	CHARON	Logged:	6/3/2016 9:01:42 AM
Event ID:	314	Task Category:	(1)
Level:	Information	Keywords:	Classic
User:	N/A	Computer:	bruno-PC
OpCode:			
More Information:	Event Log Online Help		

Windows Event Application log example - License dongle not connected:



CHARON License Expiration Check: Unable to generate the license view from hasp_view.exe!

Log Name:	Application		
Source:	CHARON	Logged:	4/28/2016 3:46:17 PM
Event ID:	314	Task Category:	(1)
Level:	Error	Keywords:	Classic
User:	N/A	Computer:	bruno-PC
OpCode:			
More Information:	Event Log Online Help		

Sending alerts via email - example

Using Powershell we can send alerts via email. You will find below an example with configuration steps.

Configuration details:

- Windows 2008 R2 server
- Powershell V4.0 installed.
  - ⚠ This version is required to have the "`-port`" option available from the "`send-mailmessage`" powershell command.
  - To determine which version of Powershell is installed and upgrade if necessary, see Powershell version, upgrade, enabling scripts execution, tips and tricks
- Email account hosted on Office365

Email account password encryption

Due to the email account settings, to be able to send an email we need to generate a CliXML file that will store the encrypted password as described below.

ℹ This file will only be valid for the current user and machine. The following operations must be performed using the same user as the author of the scheduled task.

❓ The "<monitoring-account>@<somewhere>" value must be adapted to your configuration

❓ When you will execute the "`get-credential`" command, you will be prompted to store the password of the corresponding email account, that is the "sender".

```
C:\Charon>powershell
Windows PowerShell
Copyright (C) 2014 Microsoft Corporation. All rights reserved.

PS C:\Charon> $From="<monitoring-account>@<somewhere>"
PS C:\Charon> $Creds=(get-credential -credential $From)
PS C:\Charon> $Creds | Export-CliXML C:\Charon\creds.clixml
PS C:\Charon> exit
C:\Charon>exit
```

⚠ If the sender's email address or its password has to be changed, the `.clixml` file will have to be recreated.

Powershell script used to send emails

Create a Powershell script named "`sendmail.ps1`" in the "`c:\Charon`" folder for example.

The sender ($From) and recipient ($To) values must be adapted to your configuration

```
Param(
  [string]$lvl = "Information",
  [string]$msg = "no message"
)

$Hostname = (Get-Item env:\COMPUTERNAME).value
$Creds = Import-clixml C:\Charon\creds.clixml

#--- Customize the values below: "From" and "To" ----------------------------
$From = "<monitoring-account>@<somewhere>"
$To = "<someone>@<somewhere>"
#---------------------------------------------------------------------------

send-mailmessage -to $To -from $From -subject "CHARON License Expiration Check for $Hostname ($lvl)" -body "$msg" -credential
$Creds -smtpserver smtp.office365.com -usessl -port 587 -delivery none
```

If you have to specify multiple recipients, an array has to be used as described below.

Example1:

```
$To = @( "James T. Kirk <kirk@uss-enterprise.fed>", "Spock <spock@uss-enterprise.fed>" )
```

Example2:

```
$To = @( "kirk@uss-enterprise.fed", "spock@uss-enterprise.fed", "uhura@uss-enterprise.fed" )
```

The example shown above and another one given with embedded html code and colored email body are part of the kit downloaded previously.

Office365 email account is given here as an example but you can use "gmail" also and create a free dedicated account to send alerts. In this case the last line of the script(s) above has to be updated: the "`-smtpserver`" parameter has its "`smtp.office365.com`" value to be replaced by "`smtp.gmail.com`" (no other value to update)

Google may block sign-in attempts when using powershell and send-mailmessage. In this case the sender will receive a "Sign in attempt prevented" alert email. To allow emails to be sent:

1. Create a dedicated gmail account
2. Allow less secure apps to access your account. See this article: https://support.google.com/accounts/answer/6010255?hl=en

expiredlicense.bat file settings

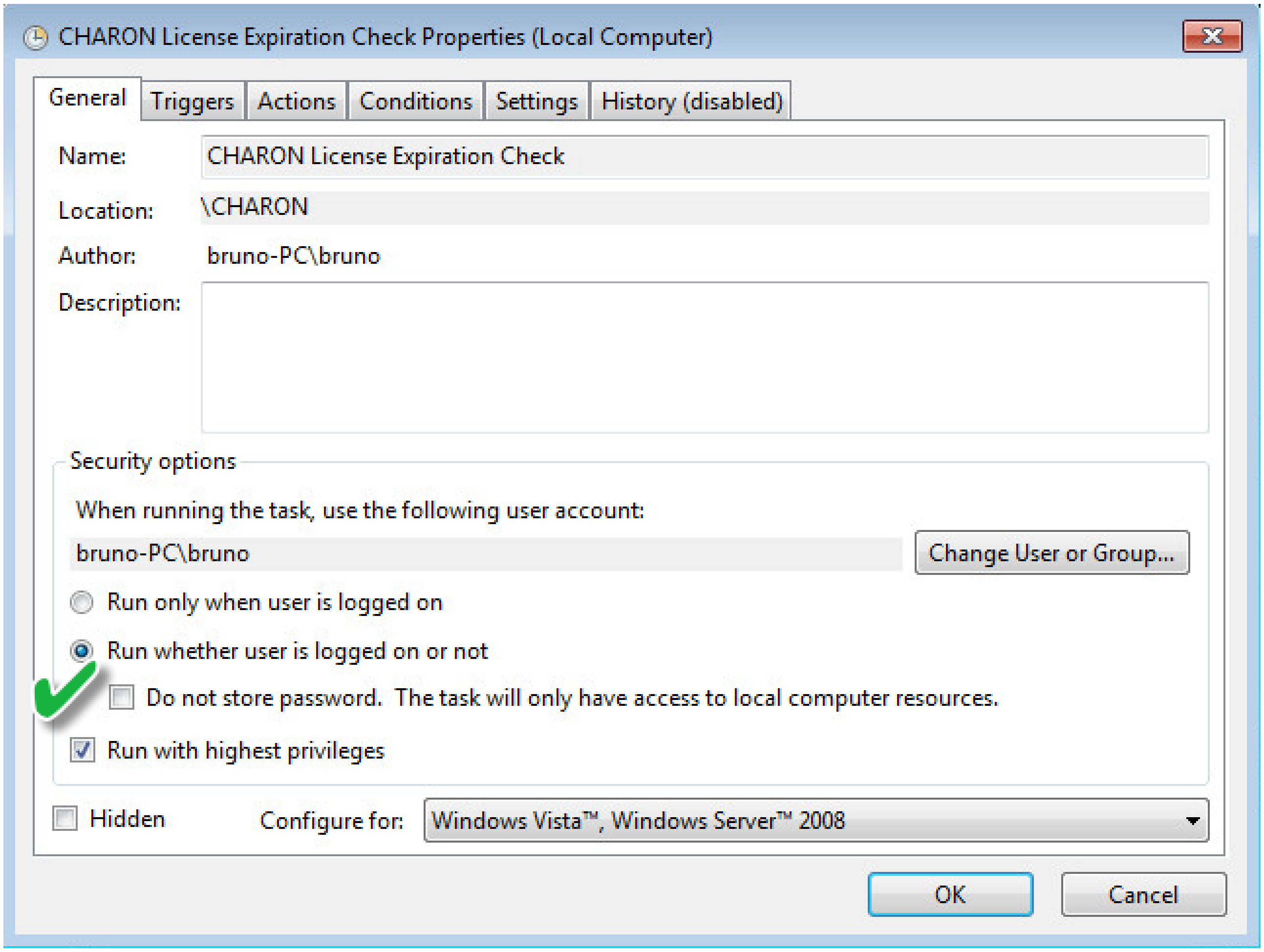
Add the following line in the "`expiredlicense.bat`" file:

```
powershell -NonInteractive -file c:\charon\sendmail.ps1 -lvl %1 -msg %2
```

This .bat file can be found in the kit downloaded previously.

Scheduled task settings update

This time, the **General** tab of the Scheduled task has to be updated in order to store the password and to be able to send the email:



CHARON License Expiration Check Properties (Local Computer)

Name: CHARON License Expiration Check

Location: \CHARON

Author: bruno-PC\bruno

Description:

Security options

When running the task, use the following user account:

bruno-PC\bruno

○ Run only when user is logged on

● Run whether user is logged on or not

□ Do not store password. The task will only have access to local computer resources.

☑ Run with highest privileges

□ Hidden

Configure for: Windows Vista™, Windows Server™ 2008

You will be prompted to store the password of the author.

This can also be done using the command line:

```
c:\Charon>schtasks /change /TN "CHARON\CHARON License Expiration Check" /rp <password>
```

Related articles

- CHARON on Windows - Automated License Expiration Check
- How-to install Charon license expiration check on a Linux server
- CHARON on Windows - Charon License Utility
- CHARON on Windows - Charon License Utility - Release Notes
- How to read a HASP key over RDP on Windows

mtd

Table of Contents

General Description

"`mtd`" is a command line utility to:

- Create a CHARON tape image from a physical tape.
- Write a tape image to a physical tape.
- Convert tape images formats.
- Test a tape container integrity.

To start the utility open "`cmd.exe`" in the Start menu and switch to the CHARON installation folder then the "`\Build_`*`XXX`*`\x`*`XX`*" child folder where the utility is located.

Usage

The following are examples of the usage syntax:

Dump tape content to file:

```
...> mtd [options] <tape device name> <tape container name>
```

Restore dump to tape:

```
...> mtd [options] <tape container name> <tape device name>
```

Convert formats:

```
...> mtd [options] <tape container name> <tape container name>
```

Examine tape dump and check its integrity:

```
...> mtd [options] <tape container name>
```

Options:

Parameter	Description
`-l <file name>`	Creates a log file. The name is "file name".
`-r <number>`	Specifies a number of attempts to read a damaged data block
`-i`	Ignore bad blocks and continue processing w/o interruption. It implies "-r 0"
`-n`	Do not rewind tape
`-p`	Disable progress reporting
`-v`	Enable verbose tracing of data transfer (implies "-p")

Parameter(cont.)	Description
-s	Write tape image in Charon-AXP/SMA (named SMA further) format
-g	Gather statistics and print upon completion
-a	Do not print logo

Example 1:

```
...> mtd -l tape1.txt -r 10 \\.\Tape0 "C:\Charon\Tapes\tape1.vtape"
```

Example 2:

```
...> mtd "C:\Charon\Tapes\tape1.vtape" \\.\Tape0
```

Tape container formats transfer

Use the following syntax to transfer the CHARON-AXP/SMA tape container format to the CHARON-AXP/VAX/PDP one:

```
...> mtd <SMA tape container name> <AXP/VAX/PDP tape container name>
```

Example:

```
...> mtd C:\charon\tapes\sma_tape.vtape C:\charon\tapes\axp_tape.vtape
```

Use the following syntax to transfer the CHARON-AXP/VAX/PDP tape container format to the CHARON-AXP/SMA one:

```
...> mtd -s <AXP/VAX/PDP tape container name> <SMA tape container name>
```

Example:

```
...> mtd -s C:\charon\tapes\axp_tape.vtape C:\charon\tapes\sma_tape.vtape
```

CHARON Log Monitor and Dispatcher

Table of Contents

General Description

The Log Monitor & Dispatcher is a special program which monitors a Charon-AXP emulator log file and executes a custom script when it detects the disappearance of a license.

It runs in the background (as a program or as a service) and periodically scans a specified LOG file. It detects the following messages and executes the associated scripts (created manually):

Error code	Description	Script executed
`00000424`	Detected removal of a license.	`nolicense.bat`
`0000040B`	License has changed. License detected and online.	`license_changed.bat`

The Log Monitor & Dispatcher service is installed as EmulatorLogMonitor. By default it is installed in such a way that requires explicit actions to be started (either through a command line interface or using the standard ways of service management). For unattended execution, change the service’s configuration so that Windows starts the service automatically.

ℹ The tool requires the specific files "`nolicense.bat`" and "`license_changed.bat`" containing specific instructions to be taken in situation of license absence or change. It is recommended you create these files in the folder (presumably) containing the LOG file.

When it is invoked by the “Log Monitor & Dispatcher”, the current directory of the batch process is set to the same folder from which the “Log Monitor & Dispatcher” was previously installed as a service or from which it was started as a background process. This means that the user action file may, in principle, operate with relative paths and relocate (as part of the whole configuration, i.e. together with accompanying LOG file).

⚠ The user action file will not invoke interactive applications as it may run in an environment where interactive services do not work, for example: when “Log Monitor & Dispatcher” is installed as a service.

Starting in background mode

In order to **start** the Log Monitor & Dispatcher as a background application:

1. Open "cmd.exe" from the "Start" menu.
2. Change current directory to the folder (presumably) containing the CHARON configuration file.
3. Start the Log Monitor & Dispatcher using the following command line as an example:

   For single log file:

```
...> “C:\Program Files\CHARON\Build_XXX\x64\logmond” C:\my_charon.log
```

   For rotating log file:

```
...> “C:\Program Files\CHARON\Build_XXX\x64\logmond” -l <log-directory> -p <log-prefix>
```

where:

- <log-directory> is the directory where the rotating log files are stored
- <log-prefix> is the same as the "configuration_name" value in corresponding CHARON configuration file (or "hw_model", if "configuration_name" is not specified).

Example:

```
...> “C:\Program Files\CHARON\Build_XXX\x64\logmond” -l "C:\My CHARON logs" -p "MY_VAX"
```

To stop the Log Monitor & Dispatcher application, open the Task Manager, find the “logmond.exe” process and terminate it.

Installing as a service

In order to install the Log Monitor & Dispatcher as a background application:

1. Open "cmd.exe" from the "Start" menu in "Run as Administrator" mode.
2. Install Log Monitor & Dispatcher service using the following command line as an example:

```
...> “C:\Program Files\CHARON\Build_XXX\x64\logmond” -r
...> “C:\Program Files\CHARON\Build_XXX\x64\logmond” -i
```

As result the "logmond" service has "Manual" service start type. Use Windows service management tools to change this mode to "Automatic" if it is required.

⚠

- There will have one `logmond` father process that will scan the virtual machines services and one `logmond` process per log monitored.
- The `EmulatorLogMonitor` service running the `logmond` processes will discover already installed CHARON virtual machines services. If a new virtual machine service is added, it's log file will be automatically discovered and monitored.
- The "`nolicense.bat`" and "`license_changed.bat`" files have to be placed in the virtual machine "Home directory". This directory corresponds to the configuration file folder when the service is installed, however it will not change if the configuration file is relocated and the service is updated. To find this "Home directory", open the "CHARON Service Manager" utility, right click on the corresponding service and select "View configuration"

Starting the Log Monitor and Dispatcher service

In order to start the Log Monitor & Dispatcher as a background application:

1. Open "cmd.exe" from the "Start" menu in "Run as Administrator" mode.
2. Start the Log Monitor & Dispatcher service using the following command line as an example:

```
...> “C:\Program Files\CHARON\Build_XXX\x64\logmond” -u
```

It is also possible to use Windows service management tools to start up the service.

As soon as "logmond" service is installed this way, and started, it monitors all CHARON instances (provided that these are installed as services).

It does NOT monitor emulator instances started from Launcher or manually from command line prompt, these cases can still be monitored with "logmond" invoked from command line prompt with name of the log file as an argument (see above).

Stopping the Log Monitor and Dispatcher service

In order to stop the Log Monitor & Dispatcher as a background application:

1. Open "cmd.exe" from the "Start" menu in "Run as Administrator" mode.
2. Stop the Log Monitor & Dispatcher service using the following command line as an example

```
...> “C:\Program Files\CHARON\Build_XXX\x64\logmond” -d
```

It is also possible to use Windows service management tools to stop the service.

Uninstalling the Log Monitor and Dispatcher service

In order to uninstall the Log Monitor & Dispatcher as a background application:

1. Open "cmd.exe" from the "Start" menu in "Run as Administrator" mode.
2. Uninstall the Log Monitor & Dispatcher service using the following command line as an example

```
...> “C:\Program Files\CHARON\Build_XXX\x64\logmond” -r
```

Log Monitor and Dispatcher best practice

Refer to this article in the Appendixes for proper configuration of the "nolicense.bat" file to establish needed actions on the dongle removal.

HOSTPrint

Table of Contents

General Description

The HOSTPrint utility is used to print CHARON output to Windows printers. Typically it receives data from COM2 port of emulated Alpha model via a TCP /IP socket and prints the data received on the default Windows printer (if no printer is specified at the utility command line) on the host computer.

Note that the second console line "TTA0" (COM2) is available only for 1 CPU models such as:

- AlphaServer 400
- AlphaServer 800
- AlphaServer 1000
- AlphaServer 1000A
- AlphaServer DS10
- AlphaServer DS10L
- AlphaServer DS15

Usage

HOSTPrint calling is specified in the CHARON configuration file. The call implements one or more of the following parameters:

Parameter	Description
-host=<*hostname*>	Name of the host - source of the printing data.
-port=<*connection port number*>	Port on the host to get the information to print from.
-delay=<*delay for automatically buffer flush in seconds*>	Flushing delay, 0 - wait infinite, 5..10800 - timeout for flush

-printer=[*PrinterDeviceName*]	Host name for the printer used.
	Example 1:
	`-printer=[\\.\Microsoft Office Document Image Writer]`
	Example 2:
	`-printer=[\\print_server\MSCLPS]`

-font=<*default font name*>	Default font name.
-fontsize=<*default font size*>	Default font size.

The two last parameters are only used for compatibility with older versions of the utility (HOSTprint allows changing font settings from a popup menu).

It is **strongly recommended** to use fixed-size fonts (by default the "Courier" font is used) to avoid any problems relevant to proper calculation of the printing line length.

Example of CHARON-AXP configuration file for the HOSTPrint usage:

```
set COM2 alias = TTA0 port = 10000
set COM2 application = "hostprint.exe -port=10000 -printer=[\\print_server\MSCPS2] -font=\Courier New\ -fontsize=10"
```

After initialization, HOSTprint creates an icon in the Windows tray. The icon can have two colors:

Color	Description
GREEN	IDLE (or Ready) state
YELLOW	BUSY (processing) mode

*: 💡 To access the HOSTprint application popup menu, point the mouse cursor at its icon in the system tray menu and click the right button, preview the last page content then change the utility mode via the flush buffer delay or change the default font setting.

CHARON Guest Utilities for OpenVMS

Table of Contents

General Description

The "CHARON Guest Utilities for OpenVMS" (CHARONCP) package contains several utilities for managing virtual tapes, changing the emulator speed and creating useful definitions for that operations.

This set of utilities is located in the "`charoncp013.vdisk`" disk file in the "`C:\Program Files\CHARON\Virtual Disk Images\Tape utilities package`" folder.

Supported OpenVMS versions: OpenVMS 6.1 and above.

In case of OpenVMS upgrade, CHARONCP will have to be re-installed.

Installation

Make this virtual disk file recognized by the virtual machine in the CHARON configuration file, boot from the system disk and mount the disk with the following OpenVMS command:

```
$ MOUNT <device name> /OVERRIDE=IDENTIFICATION
```

Issue the following commands to install the package (example given for OpenVMS V8.4):

```
$ @SYS$UPDATE:VMSINSTAL

...

* Are you satisfied with the backup of your system disk [YES]? YES
* Where will the distribution volumes be mounted: <device name>:[CHARONCP013.KIT]

Enter the products to be processed from the first distribution volume set.
* Products: CHARONCP013

* Enter installation options you wish to use (none): <press enter>

...

Do you want to install this product [NO]? YES

...

* Where should the CHARONCP root directory be located ? [SYS$SYSDEVICE:[CHARONCP]]: <press enter>

...

* Do you want to purge files replaced by this installation [YES]? <press enter>
```

Select all the components included in the package:

```
                              Component Selection

Select the CHARONCP components you wish to install from the menu below.
An asterisk appears next to the packages that have already been
selected. You can remove a package from the list by selecting it
again. You may enter more than one selection by separating your
choices with commas.

1. [*] CHARONCP Guest Utility (REQUIRED)
2. [*] Compatability Utilities
3. [*] Install DCL Commands & Help

4. Exit

* Your choice [4]: 1,2,3

...

* Your choice [4]: 4

...

* Is this correct [YES]: <press enter>

...

* Products:<press enter>

        VMSINSTAL procedure done at hh:mn

$
```

Proceed with installation using all the default options.

Once the installation is completed, add the following line to the "`SYS$STARTUP:SYSTARTUP_VMS.COM`" file for the package to be loaded automatically at system startup:

```
$ @SYS$STARTUP:CHARONCP_STARTUP
```

Performance optimization

CHARON takes 100% of host CPU even in case of idle state of guest OpenVMS operating system. To get rid of such resources consumption there is a specific option provided by CHARON Guest Utilities: "idle" mode.

To load the OpenVMS idle loop detection software, use:

```
$ CHARONCP SET IDLE /ENABLE
```

This allows CHARON to detect when the emulated CPU(s) are idle and use the host power saving instructions to reduce power usage.

To unload the OpenVMS idle loop detection software, use:

```
$ CHARONCP SET IDLE /DISABLE
```

Virtual tapes management

Specify mapping to tape container in the following way in the CHARON configuration file:

```
set <adapter name> container[<unit name>] = ".vtape" removable[<unit name>] = true
```

- It is mandatory to set the "`removable`" parameter to "`true`"
- The container name must be "`.vtape`", no name and path must be specified, only extension.

Example:

```
set PKA container[600] = ".vtape" removable[600] = true
```

Once it is done using the following commands it is possible to manage virtual tapes attached to CHARON.

To create the specified host-file (if it does not already exist) and attach it to the specified virtual tape device, use:

```
$ CHARONCP SET MAGTAPE <device> /LOAD="<filename>.vtape"
```

The container name specified in the /LOAD parameter must not be more than 255 characters

Example:

```
$ CHARONCP SET MAGTAPE MKA600: /LOAD="E:\Charon\Tapes\backup_01.vtape"
```

To detach any file currently attached to te specified virtual tape device, use:

```
$ CHARONCP SET MAGTAPE <device> /UNLOAD
```

Example:

```
$ CHARONCP SET MAGTAPE MKA600: /UNLOAD
```

Possible error	Description
BADFILENAME	The filename specified as a value to the qualifier `/LOAD` was either too long or does not have a file extension of ".vtape".
DEVNOTDISM	Attempting to execute a `SET MAGTAPE/LOAD` command when a file is already attached. Perform a `SET MAGTAPE/UNLOAD` first. If a `SET MAGTAPE/LOAD` command has not previously been executed then the CHARON configuration container specification for the tape device may contain a full path. Doing this will create and attach and initial tape container file. To avoid this, remove the file name from the specification (leaving only a file extension of ".vtape" and optional directory).

If some tape container has been already specified in the CHARON configuration file, use the command "`CHARONCP SET MAGTAPE <device> /UNLOAD`" to unload it first.

Defining keys

It is possible to define certain keys on the terminal keyboard for fast access to the CHARONCP functionality while you are in CHARONCP.

To define an equivalence string and a set of attributes with a key on the terminal keyboard, use:

```
$ CHARONCP
CHARONCP> DEFINE /KEY <key-name> <equivalence-string>
```

You can have a set of keys defined automatically for use with the CHARONCP utility by placing `DEFINE/KEY` commands in the `SYS$LOGIN:CHARONCP_KEYDEFS.INI` file.

Example:

```
$ CHARONCP
CHARONCP> DEFINE /KEYF1 "SET MAGTAPE MKA600: /UNLOAD"
```

To display key definitions created with the `DEFINE/KEY` command, use:

```
$ CHARONCP
CHARONCP> SHOW KEY <key-name>
```

Refer to the DCL help entry for `SHOW KEY` for further information.

Example:

```
$ CHARONCP
CHARONCP> SHOW KEY F1
DEFAULT key state definitions:
F1 = "set magtape mka600: /unload"
CHARONCP>
```

ℹ For more information refer to the OpenVMS DCL Dictionary (`DEFINE/KEY` section).

Displaying version

To display the CHARONCP package version number and architecture, use:

```
$ CHARONCP SHOW VERSION
```

This can be useful for customers reporting issues with the CHARONCP software.

Example:

```
$ CHARONCP SHOW VERSION
CHARONCP version id is: V1.3
```

Charon Report Utility

Description

The Charon Report Utility creates an HTML report file with useful information for the Stromasys support team, and a zip file that contains this HTML file and the configuration and log files of the virtual machines defined on the Windows server.

Installation

The Charon Report Utility is provided as a package in the Charon product kit and as an executable in case installing a package requires authorizations. It is not installed by default. The package is located in the "`CharonReportUtility`" folder of the kit. It is also possible that Stromasys will provide a download link to get the latest version from our file server.

Install the package (.msi file), this will create a shortcut on the desktop and an entry in the Windows menu.

Prerequisites

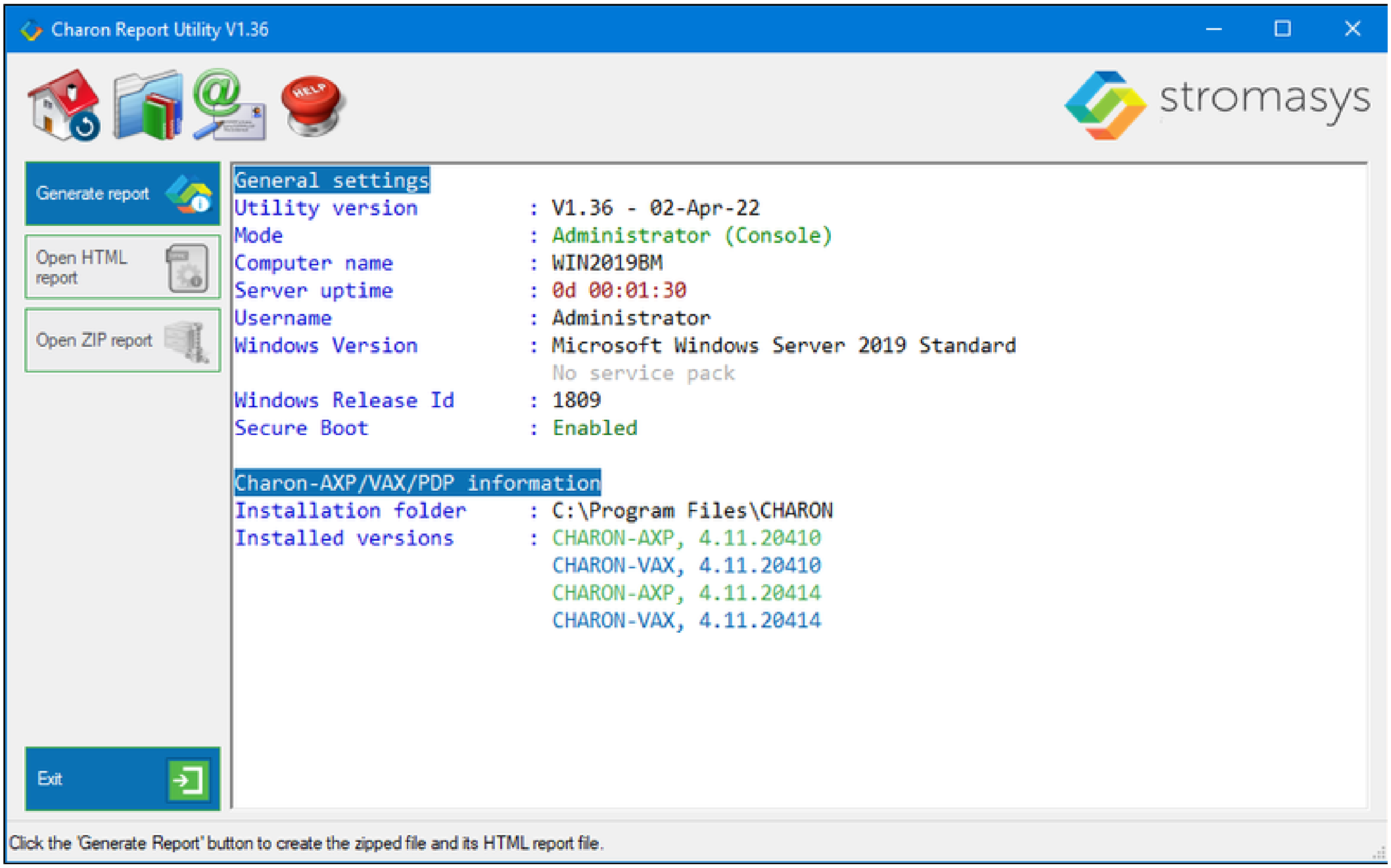
- PowerShell 4.0, PowerShell 5.1 recommended.
  - Powershell 4.0 can work but with limitations, some functions will not be operational
  - If you installed PowerShell 7, please ensure PowerShell 5.1 is installed (not 4.0)
  - To upgrade, please see: https://docs.microsoft.com/en-us/powershell/scripting/windows-powershell/install/installing-windows-powershell
- .NET Framework 4.6 minimum, 4.8 recommended.
  - Installed by default on Windows 10, 11, 2016, 2019 and 2022.
  - For older versions of Windows, this has to be installed manually. For more, see: https://docs.microsoft.com/en-us/dotnet/framework/migration-guide/versions-and-dependencies

Usage

Open the executable file and click the "Generate report" button (the operation can take some time). Once completed, the resulting .zip file has to be sent to our support team.

⚠ The utility must be executed "as administrator"

Example



```
General settings
Utility version          : V1.36 - 02-Apr-22
Mode                     : Administrator (Console)
Computer name            : WIN2019BM
Server uptime            : 0d 00:01:30
Username                 : Administrator
Windows Version          : Microsoft Windows Server 2019 Standard
                           No service pack
Windows Release Id       : 1809
Secure Boot              : Enabled

Charon-AXP/VAX/PDP information
Installation folder      : C:\Program Files\CHARON
Installed versions       : CHARON-AXP, 4.11.20410
                           CHARON-VAX, 4.11.20410
                           CHARON-AXP, 4.11.20414
                           CHARON-VAX, 4.11.20414
```

Click the 'Generate Report' button to create the zipped file and its HTML report file.

CHARON-AXP for Windows configuration details

Introduction

This chapter describes, in detail, the configuration parameters of the devices emulated by CHARON-AXP for Windows, with corresponding examples and parameters.

The emulated devices are loaded with the "load" command (if a device has not been already loaded) and the parameters are made active with the "set" command. These parameters can be specified directly in the "load" command.

Example:

```
load KZPBA DKA
set DKA container[0]="C:\Charon\Disks\BootDisk.vdisk"
```

In this example, an instance of a KZPBA controller is loaded with the name "DKA". Its first unit, "container[0]", is mapped to the "C:\Charon\Disks\BootDisk.vdisk" disk image.

ⓘ The Controller name is accompanied with a "/<module name>". The module name is a CHARON-AXP component that specifies the controller load module. Its name can be the same as the loaded controller, however this is not mandatory. Once a module name is specified, there is no need to specify it again for additional references of the same controller.

Details of CHARON-AXP configuration

General Settings

Table of Contents

Session

General settings that control the execution of CHARON Virtual Machine (VM) belong to an object called the "session". It is a preloaded object; therefore, only "set" commands are necessary.

Example:

```
set session <parameter>=<value>
```

The following table describes all available "session" parameters, their meaning and examples of their usage:

hw_model

Parameter	hw_model
Type	Text string, case sensitive
Value	Virtual Alpha system hardware model to be emulated. Use a default configuration template for each particular model as a starting point for a custom configuration. This would ensure that the parameter is set correctly. **Example**: `set session hw_model="AlphaServer_ES40"` **Available models are**: • AlphaServer_AS400 • AlphaServer_AS800 • AlphaServer_AS1000 • AlphaServer_AS1000A • AlphaServer_AS1200 • AlphaServer_AS2000 • AlphaServer_AS2100 • AlphaServer_AS4000 • AlphaServer_AS4100 • AlphaServer_DS10 • AlphaServer_DS10L • AlphaServer_DS15 • AlphaServer_DS20 • AlphaServer_DS25 • AlphaServer_ES40 • AlphaServer_ES45 • AlphaServer_GS80 • AlphaServer_GS160 • AlphaServer_GS320 ℹ Refer to this section to find how to set a particular HP Alpha model supported by CHARON-AXP.

configuration_name

Parameter	configuration_name
Type	Text string

Value	Name of the CHARON VM (it must be unique):
	`set session configuration_name="MSCDV1"`
	⚠ In provided templates this parameter is specified in the included configuration file "configuration_name.icfg".
	The value of this parameter is used as a prefix to the event log file name (see below).
	From the example above, the CHARON VM log file will have the following name:
	`MSCDV1-YYYY-MM-DD-hh-mm-ss-xxxxxxxxx.log`
	xxxxxxxxx is an increasing decimal number starting from 000000000 to separate log files with the same time of creation (in case the log is being written faster than one log file per second).
	⚠ It is strictly recommended to use the "configuration_name" parameter if more than one CHARON VM runs on the same server.

log

Parameter	log
Type	Text string

Value	The log file or directory name is where the log file for this CHARON-AXP session is stored.
	Log specified as a file name
	Example:
	`set session log="C:\Charon\es40prod.log"`
	Log specified as a directory
	CHARON-AXP automatically creates individual log files for each CHARON-AXP execution session. If the log parameter is omitted, CHARON-AXP creates a log file for each CHARON-AXP execution session in the directory where the emulator was started. In these two cases, the log rotation mode is enabled, meaning a new log file is created each time the virtual machine is started and when the log file size exceeds the one specified (see General Settings#log_file_size) and/or when the log file is older than a specified number of days (see General Settings#log_rotation_period).
	ℹ A shortcut located in the same directory will be created, pointing to the active log file. Its name is based on the hw_model parameter or the configuration_name parameter if specified.
	If the "configuration_name" parameter of the sessionis specified, the log file name is composed as follows:
	`<configuration_name>-YYYY-MM-DD-hh-mm-ss-xxxxxxxxx.log`
	If the "configuration_name" parameter is omitted, the log file name will have the following format:
	`<hw_model>-YYYY-MM-DD-hh-mm-ss-xxxxxxxxx.log`
	where "`xxxxxxxxx`" is an increasing decimal integer, starting from 000000000 to separate log files with the same time of creation (in case the log is being created faster than one log file per second).
	⚠ Only existing directory can be specified. If the directory specified does not exist, this will be considered as a flat file. No trailing backslash character is allowed.
	Example:
	`set session configuration_name="es40prod"`
	`set session log="C:\Charon\Logs"`
	The execution of the virtual machine will create a log file, named `C:\Charon\Logs\es40prod-2016-10-13-10-00-00-000000000.log` (for example) and a shortcut named `C:\Charon\Logs\es40prod.log` pointing to this file. The shortcut will be updated when the log rotation will occur.
	ℹ Starting with version 4.11 build 204-11, log file numbering is implemented when log rotating is used: when starting the Charon emulator the first log file name will be like `<configuration_name>-YYYY-MM-DD-hh-mm-ss-xxxxxxxxx.log` where `xxxxxxxxx` will be 000000000, this number will increase with each log rotation to make it easier to find a log file set (1st log file at start + all its rotated files)

log_method

Parameter	log_method
Type	Text string
Value	• "append" (default) • "overwrite" Determines if the previous log information is maintained or overwritten. ⓘ This parameter must be specified only in addition to the “log” parameter and on the same line. This parameter is applicable only if the CHARON VM log is stored to a file that is specified explicitly with the "log" parameter. **Example**: `set session log="log.txt" log_method="append"`

log_show_messages

Parameter	log_show_messages
Type	Text string
Value	• "all" (default) • "info" • "warning" • "error" Defines the message types to be shown. The parameter is a string of comma delimited words. **Example**: `set session log_show_messages="error, warning"`

log_file_size

Parameter	log_file_size
Type	Text string

Value	If log rotation is enabled, the log_file_size parameter determines the log file size threshold at which a new log is automatically created. Rotating log file size is multiple of 64K
	- "unlimited" or "0" (default) - the feature is disabled
	- "default" - default size is used (4Mb)
	- <size>[KMG] - size of the current log file in bytes with additional multipliers:
	  - K - Kilobyte
	  - M - Megabyte
	  - G - Gigabyte
	Example 1:
	`set session log_file_size="default"`
	Example 2:
	`set session log_file_size=10M`
	ℹ Minimum LOG File size is 64K, maximum is 1G. Setting a size less than 64K effectively makes the log file size unlimited.

log_rotation_period

Parameter	log_rotation_period
Type	Text string

Value	- "default" - default value, 7 days. This value is used even if the "log_rotation_period" is not specified.
	- "daily" or "1"
	- "weekly" or "7"
	- "never"
	- <N> - in N days where N is greater than 0
	If the rotation log mode is enabled, this parameter controls switching to the next log file based on a period of time. If enabled, the switching to the next log file occurs at midnight
	Example 1:
	`set session log_rotation_period="weekly"`
	Example 2:
	`set session log_rotation_period=14`

log_repeat_filter

Parameter	log_repeat_filter
Type	Text string
Value	Specifies if repeated messages should be filtered or not. Possible values are "on" and "off" (default). If the value is "on", immediately following messages with the same identifier and system error code are not listed in the log but they are counted. When a different log message is generated, the repeat count of the earlier log message is reported with the entry "The previous message has been repeated *N* times." and the counter is cleared. ⚠ Stromasys highly recommends to let this value to "off" otherwise important information will not be reported in the log file. **Example**: `set session log_repeat_filter="off"`

log_flush_period

Parameter	log_flush_period
Type	Numeric
Value	• <period-in-seconds> Defines a period of flushing log to disk. Default period is 60 seconds (it means that every minute log file is flushed to disk) **Examples**: `set session log_flush_period=30`

license_key_id

Parameter	license_key_id
Type	Text string
Value	A set of Sentinel Key IDs that specifies the license keys to be used by CHARON. Setting to "**any**" and CHARON will look for a suitable license in all available keys. **Example**: `set session license_key_id = "1877752571,354850588,any"` Based on the presence of this parameter in the configuration file, CHARON behaves as follows: 1. **No keys are specified (the parameter is absent)** CHARON performs an unqualified search for any suitable key in unspecified order. If no key is found, CHARON exits. 2. **One or many keys are specified** CHARON performs a qualified search for a regular license key in the specified order. If it is not found, CHARON exits (if the keyword "any" is not set). If the keyword "any" is specified then if no valid license has been found in the keys with specified ID’s all other available keys are examined for valid license as well. ⚠ The order in which keys are specified is very important. If a valid license was found in the key which ID was not the first one specified in configuration file, then available keys are periodically re-scanned and if the key with the ID earlier in the list than the current one is found CHARON tries to find a valid license there and in case of success switches to that key.

license_id

Parameter	license_id
Type	Text string
Value	A set of license identifiers that specifies the licenses to be used by CHARON. This parameter is applicable only to licenses on which Stromasys placed restrictions on what products can be combined on a single license key. Please contact your Stromasys representative or VAR for more information. **Example**: `set session license_id = "2718281828,314159265"` If this parameter is set, Charon considers for validation only the available licenses with license ID parameter set and equal to one of the license ID's specified in the configuration. This prioritized list corresponds to the "Product License Number" line in the Product section of the license.

license_key_lookup_retry

Parameter	license_key_lookup_retry
Type	Text String

Value	In case the CHARON-AXP license connection is not present when the guest starts up, this parameter specifies how many times CHARON-AXP will try to establish the connection and, optionally, a period of time between retries.
	Syntax:
	`set session license_key_lookup_retry = "N [, T]"`
	Options:
	• N - Number of retries to look for license keys.
	• T - Time between retries in seconds. If not specified 60 seconds are used
	Example 1:
	`set session license_key_lookup_retry = 1`
	If license key is not found during initial scan, do only one more attempt after 60 seconds.
	Example 2:
	`set session license_key_lookup_retry = "1,30"`
	Same as above but retry in 30 seconds.
	Example 3:
	`set session license_key_lookup_retry = "3,10"`
	If license key is not found during initial scan, do 3 more attempts waiting 10 seconds between them.
	Example 4:
	`set session license_key_lookup_retry = "5"`
	If license key is not found during the initial scan, do 5 more attempts waiting 60 seconds between them.

affinity

Parameter	affinity
Type	Text string
Value	Overrides any initial process affinity mask provided by the host operating system. Once specified it binds the running instance of the emulator to particular host CPUs. Used for soft partitioning of the host CPU resources and/or for isolating host CPUs for other applications. By default the CHARON VM allocates as many host CPUs as possible. The “affinity” parameter overrides that and allows explicit specification on which host CPU the instance must run on. ⚠ The "affinity" parameter defines the total number of host CPUs to be used both for emulated Alpha CPUs and for the CHARON VM itself (including the CPUs to be used for I/O - they are controlled by the "n_of_io_cpus" parameter described below). Host CPUs are enumerated as a comma separated list of host system assigned CPU numbers: `set session affinity="0, 2, 4, 6"`

n_of_cpus

Parameter	n_of_cpus
Type	Numeric
Value	Limits the number of emulated CPUs. **Example**: `set session n_of_cpus=3` The maximum number of CPUs enabled by CHARON VM is specified by the license key. It cannot exceed the original hardware restrictions. See table below.

Alpha Model	MAX Number of emulated CPUs
AlphaServer_AS400	1
AlphaServer_AS800	1
AlphaServer_AS1000	1
AlphaServer_AS1000A	1
AlphaServer_AS1200	2
AlphaServer_AS2000	2
AlphaServer_AS2100	4
AlphaServer_AS4000	2
AlphaServer_AS4100	4
AlphaServer_DS10	1
AlphaServer_DS15	1
AlphaServer_DS20	2
AlphaServer_DS25	2
AlphaServer_ES40	4
AlphaServer_ES45	4
AlphaServer_GS80	8
AlphaServer_GS160	16
AlphaServer_GS320	32

n_of_io_cpus

Parameter	n_of_io_cpus
Type	Numeric
Value	This parameter specifies how many host CPUs CHARON-AXP will use for I/O handling. Use of the “affinity” parameter may limit the number of CPUs available. By default the CHARON-AXP instance reserves one third of all available host CPUs for I/O processing (round down, at least one). The “n_of_io_cpus” parameter overrides that by specifying the number of CHARON I/O CPUs explicitly. **Example**: `set session n_of_io_cpus=2`

File inclusion

It is possible to include a configuration file into an existing one using the "`include`" command. The file extension is usually `.icfg`.

Format:

```
include "file.icfg"
```

Example:

```
include "c:\charon\commonpart.icfg"
```

The configuration template files are defined to use the "configuration_name.icfg" configuration file which is used by the CHARON Virtual Machines Manager. If you use an existing configuration file, including this file is not necessary.

Core Devices

Table of Contents

CPU

The CHARON-AXP CPU can be calibrated with "set ace" directive and the following parameters:

enabled

Parameter	enabled
Type	Boolean
Value	A CHARON-AXP emulated CPU is configured with the "enabled" command enabling the high performance Advanced CPU Emulation mode ("ACE"). The ACE option optimizes the Alpha instruction interpretation and significantly improves performance. It also requires approximately twice the amount of host memory allocated by CHARON instance itself to store the optimized code (Note that 2Gb of host memory + the amount of Alpha memory emulated per each CHARON instance is required). ACE optimization is performed dynamically during execution. It does not need to write optimized code back to disk, ACE provides its full capability instantly. The optimization does not compromise the Alpha instruction decoding; CHARON-AXP remains fully Alpha hardware compatible and completely transparent to the Alpha operating systems and applications. This configuration setting enables the ACE mode if the CHARON-AXP license permits it. If this configuration setting is omitted from the CHARON VM configuration file and the license permits it, "true" is the default, otherwise "false" is the default. **Example**: `set ace enabled = false` Please note that "`set ace enabled=true`" is ignored when the license does not permit ACE operation. The ACE mode is disabled when the host system does not meet the minimum physical requirements for this operation. If the emulator appears not to run at its normal performance, check the log file for a change in the ACE mode and verify that sufficient host resources, especially memory, are available.

cpu_architecture

Parameter	cpu_architecture
Type	Text String
Value	Specifies the architecture of the virtual Alpha CPU. Can be one of the following: EV4, EV45, EV5, EV56, EV6, EV67, EV68 **Example**: `set ace cpu_architecture = EV6` Refer to Core Devices#Setting of a particular Alpha model to find an appropriate value of the Alpha architecture per each Alpha model supported by CHARON-AXP.

cache_size

Parameter	cache_size
Type	Value
Value	⚠ This parameter may affect Charon performance. Do not change this parameter until advised by Stromasys "cache_size" defines the amount of memory in megabytes allocated to the ACE cache. Default value is 1GB (1024 MB). **Example**: `set ace cache_size = 2048` This parameter may be changed for performance optimization in case of some guest OS specific tasks (see the section below).

num_translators

Parameter	num_translators
Type	Value
Value	⚠ This parameter may affect Charon performance. Do not change this parameter until advised by Stromasys "num_translators" defines the number of ACE translators. Default value in most situations is number of I/O CPUs dedicated to CHARON (defined by "n_of_io_cpus" parameter) minus 1, but in some specific situations it may be set to some other value for better performance. **Example**: `set ace num_translators = 4`

host_options

Parameter	host_options
Type	Text String
Value	⚠ This parameter may affect Charon performance. Do not change this parameter until advised by Stromasys "host_options" defines options of ACE (DIT) translator and code-generator. Those options affect Charon performance. Default settings are set to optimize performance for most geust OS (VMS and Tru64) usage profiles. However there are some profiles (for example OpenVMS compilation tasks) where the default settings do not provide optimal perfromance. The following switches are available to user: 1. *--fixed-variant=[X]* The value X can be one of three options: [-1, 0, 1]. This value defines the desired translation variant. Set -1 for dynamic (default) or 0 or 1 for the fixed number implemented by the translator. 2. *--x64-optimize or --x64-nooptimize* This switch enables translation optimizations (the default is to optimize). ⚠ **Default parameters have been changed in Charon version 4.9 compared to previous versions (4.8 and below). If Charon system demonstrates lower performance after upgrade to version 4.9, please test the system with host_options switched to default 4.8 settings:** `set ace host_options = "--fixed-variant=0 --x64-nooptimize"` Note that in this case the parameter "Core Devices#num_translators" must be set to the number of I/O CPUs dedicated to CHARON (see the "Core Devices#n_of_io_cpus" parameter). Also note that changing parameter "Core Devices#cache_size" can be an alternative solution too (see the section below).

Enabling the old style performance optimization

Despite the fact that CHARON is already optimized for wide range of the guest OS tasks, there may be some situations (for example OpenVMS compilation tasks) when performance degradation may reach about 50% of the version 4.8.

In this case the following solutions can be applied:

Enlarging ACE cache size

Try this solution first. It is recommended to enlarge the ACE cache size at least in 2 times as it is shown in the following example:

```
set ace cache_size = 2048
```

If this solution does not work for the specific guest OS tasks, apply also the next solution.

Setting specific ACE host options

Set the following set of options in the configuration file:

```
set session n_of_io_cpus = <N>
set ace num_translators = <N>
set ace host_options = "--fixed-variant=0 --x64-nooptimize"
```

where <N> is number of the I/O CPUs dedicated to CHARON. If this value is not set (the default value is used) it is recommended to specify it explicitly.

RAM

The CHARON VM memory subsystem is permanently loaded and has the logical name "ram"

size

Parameter	size
Type	Numeric
Value	Size of the emulated memory in MB.

Example:

```
set ram size = 2048
```

The amount of memory is capped at a maximum, this is defined in the CHARON license key. If the host system cannot allocate enough memory to map the requested emulated memory, CHARON VM generates an error message in the log file and reduces its effective memory size.

The following table lists the values of emulated RAM for various hardware models of virtual Alpha systems:

Hardware Model	RAM size (in MB)			
	Min	**Max**	**Default**	**Increment**
AlphaServer 400	64	1024	512	64
AlphaServer 800	256	8192	512	256
AlphaServer 1000	256	1024	512	256
AlphaServer 1000A	256	1024	512	256
AlphaServer 1200	256	32768	512	256
AlphaServer 2000	64	2048	512	64
AlphaServer 2100	64	2048	512	64
AlphaServer 4000	64	32768	512	64
AlphaServer 4100	64	32768	512	64
AlphaServer DS10, DS10L	64	32768	512	64
AlphaServer DS15	64	32768	512	64
AlphaServer DS20	64	32768	512	64
AlphaServer DS25	64	32768	512	64
AlphaServer ES40	64	32768	512	64
AlphaServer ES45	64	32768	512	64
AlphaServer GS80	256	65536	512	256
AlphaServer GS160	512	131072	512	512
AlphaServer GS320	1024	262144	1024	1024

TOY

CHARON-AXP maintains its time and date using the "toy" (time-of-year) component. In order to preserve the time and date while a virtual system is not running, the TOY component uses a binary file on the host system to store the date and time relevant data. The name of the file is specified by the “container” option of the "toy" component.

container

Parameter	container
Type	Text string
Value	Specifies a name for the file in which CHARON-AXP preserves the time and date during its “offline” period. This file also keeps some console parameters (such as the default boot device). By default it is left unspecified. ⓘ it is recommended to specify the full path to the TOY file.

sync_to_host

<table>
<tr><td><b>Parameter</b></td><td>sync_to_host</td></tr>
<tr><td><b>Type</b></td><td>Text string</td></tr>
<tr><td><b>Value</b></td><td>Specifies whether and how the guest OS time is synchronized with the CHARON host time.
<b>Syntax</b>:
<code>set TOY sync_to_host = "{as_vms | as_tru64 | as_is}"</code>
where:
■ <code>as_vms</code>: if the guest OS is OpenVMS/AXP and its date and time must be set to the host's date and time each time it boots.
■ <code>as_tru64</code>: if the guest OS is Tru64 UNIX and its date and time must be set to the host's date and time each time it boots.
■ <code>as_is</code>: if the TOY date and time must be set to the host's UTC date and time
<b>Example</b>:
<code>set TOY sync_to_host = "as_vms"</code>
To synchronize the guest OS with TOY, use the following commands (from "SYSTEM"/"root" account):
■ <b>On OpenVMS/AXP</b>, use: $ <b><code>set time</code></b>
■ <b>On Tru64 UNIX</b>, use: # <b><code>date -u `consvar -g date | cut -f 3 -d ' '`</code></b>
The default value is "not specified" - it means that by default CHARON does not synchronize its guest OS time with the CHARON host time but collects date and time from the file specified with "container" parameter.
⚠ If "sync_to_host" parameter is specified there is no need to specify "container" parameter in addition.</td></tr>
</table>

Example

```
set toy container="C:\Charon\my_virtual_system.dat"
```

The CHARON VM time zone may be different from that of the host system. Correct CHARON time relies on the correctness of the host system time to calculate the duration of any CHARON “offline” periods. (i.e. while the virtual system is not running). Every time CHARON comes on line it calculates a Delta time (the system time is used if there is no TOY file). Therefore, if the host system time is changed while CHARON VM is not running, the CHARON time may be incorrect when CHARON VM is restarted and the CHARON time must be set manually.

ROM

The System Flash ROM file conserves specific parameters between reboots.

container

Parameter	container
Type	Text string
Value	Specifies the name of a file in which CHARON-AXP stores an intermediate state of its Flash ROM. This state includes most of the console parameters. By default it is left unspecified. ⓘ it is recommended to specify the full path to this file **Example**: `set rom container="C:\Charon\my_virtual_system.rom"`

system_name

Parameter	system_name
Type	Text string
Value	Allows changing the system name, for example: `set rom system_name="Alpha Server 1000 4/200"` Refer to Core Devices#Setting of a particular Alpha model to find an appropriate value of the Alpha system name per each Alpha model supported by CHARON-AXP

system_serial_number

Parameter	system_serial_number
Type	Text string
Value	Allows changing the system serial number, for example: `set rom system_serial_number = NY12345678` Any sequence of characters can be used as a serial number. Sequences longer than 16 characters are truncated to 16 characters. Serial Numbers should be according to DEC standard: 10 characters. First two characters are capital letters, remaining 8 characters are decimal digits. By default it is set to SN01234567

dsrdb

Parameter	dsrdb[n]
Type	Numeric
Value	DSRDB - Dynamic System Recognition Data Block. These parameters allow changing the emulated hardware model type. dsrdb[0] stands for SMM - System Marketing Model. **Example**: `set rom dsrdb[0]=1090` Core Devices#Setting of a particular Alpha model describes connection between "dsrdb" parameter and the rest of the parameters defining an exact Alpha model - including SMM.

version

Parameter	version
Type	Text string

Value	Sets Console and PAL code versions in the following way:
	■ **Set SRM Console version to X.Y-Z**:
	`set rom version[0] = x.y-z`
	■ **Set OpenVMS PAL code version to X.Y-Z**:
	`set rom version[1] = x.y-z`
	■ **Set Tru64 UNIX PAL code version to X.Y-Z**:
	`set rom version[2] = x.y-z`
	Example:
	`set rom version[0] = 7.3-1 version[1] = 1.98-104 version[2] = 1.92-105`

Virtual Alpha interval timer

The CHARON-AXP provides interval timer interrupts to virtual Alpha CPU(s) at frequency 100Hz (100 interrupts a second).

This is default behavior which may be changed through “clock_period” configuration parameter of virtual ISA or EISA bus, depending on emulated hardware model of virtual Alpha system.

Value of the parameter is interval timer period in microseconds. By default it is set to 10000. By changing it to 1000 frequency of virtual interval timer interrupts may be increased to 1000Hz (1000 interrupts per second).

clock_period

Parameter	clock_period
Type	Numeric
Value	Specifies period of interval timer, in microseconds. Only two values are supported: ● 10000 (which corresponds to 100Hz interval timer) ● 1000 (which corresponds to 1000Hz interval timer) By default it is set to 10000.

Example for AlphaServer 400, DS, ES, GS

```
set ISA clock_period=1000
```

Example for AlphaServer 800, 1000, 1000A, 1200, 2000, 2100, 4000, 4100

```
set EISA clock_period=1000
```

⚠ Higher interval timer frequency creates higher load for virtual Alpha CPU which may cause degradation of overall virtual system performance.

Setting of a particular Alpha model

It is important to have the "system_name", "hw_model", "cpu_architecture" and "dsrdb[n]" (DSRDB - Dynamic System Recognition Data Block) parameters in sync. (see above for details) to configure CHARON VM for emulation of a particular Alpha model.

The following tables illustrate how to synchronize those values:

AlphaStation 200 - 400

set session hw_model=	set rom system_name=	set ace cpu_architecture=	set rom dsrdb[0]=
AlphaServer_400	AlphaStation 200 4/100	EV4	1156
AlphaServer_400	AlphaStation 200 4/133	EV4	1088
AlphaServer_400	AlphaStation 205 4/133	EV4	1250
AlphaServer_400	AlphaStation 255 4/133	EV4	1257
AlphaServer_400	AlphaStation 200 4/166	EV4	1087
AlphaServer_400	AlphaStation 205 4/166	EV4	1251
AlphaServer_400	AlphaStation 255 4/166	EV4	1258
AlphaServer_400	AlphaStation 400 4/166	EV4	1086
AlphaServer_400	AlphaStation 205 4/200	EV4	1252
AlphaServer_400	AlphaStation 255 4/200	EV4	1259
AlphaServer_400	AlphaStation 200 4/233	EV45	1151
AlphaServer_400	AlphaStation 205 4/233	EV45	1253
AlphaServer_400	AlphaStation 255 4/233	EV45	1260
AlphaServer_400	AlphaStation 400 4/233	EV45	1152
AlphaServer_400	AlphaStation 205 4/266	EV45	1254
AlphaServer_400	AlphaStation 255 4/266	EV45	1261
AlphaServer_400	AlphaServer 300 4/266	EV45	1593
AlphaServer_400	AlphaStation 400 4/266	EV45	1153
AlphaServer_400	AlphaStation 400 4/266	EV45	1154
AlphaServer_400	AlphaStation 200 4/300	EV45	1157
AlphaServer_400	AlphaStation 205 4/300	EV45	1255
AlphaServer_400	AlphaStation 255 4/300	EV45	1262
AlphaServer_400	AlphaStation 400 4/300	EV45	1160
AlphaServer_400	AlphaStation 205 4/333	EV45	1256
AlphaServer_400	AlphaStation 255 4/333	EV45	1263

AlphaServer 600 - 800

set session hw_model=	set rom system_name=	set ace cpu_architecture=	set rom dsrdb[0]=
AlphaServer_800	AlphaServer 600 5/333	EV56	1310
AlphaServer_800	AlphaServer 800 5/333	EV56	1310
AlphaServer_800	AlphaServer 800 5/400	EV56	1584
AlphaServer_800	AlphaStation 600A 5/500	EV56	1590
AlphaServer_800	AlphaServer 800 5/500	EV56	1585

AlphaServer 1000

set session hw_model=	set rom system_name=	set ace cpu_architecture=	set rom dsrdb[0]=
AlphaServer_1000	AlphaServer 1000 4/200	EV4	1090
AlphaServer_1000	AlphaServer 1000 4/233	EV45	1091
AlphaServer_1000	AlphaServer 1000 4/266	EV45	1264

AlphaServer 1000A

set session hw_model=	set rom system_name=	set ace cpu_architecture=	set rom dsrdb[0]=
AlphaServer_1000A	AlphaServer 1000A 4/266	EV45	1265

AlphaServer 1200 and AlphaStation 1200

set session hw_model=	set rom system_name=	set ace cpu_architecture=	set rom dsrdb[0]=
AlphaServer_1200	AlphaServer 1200 5/300	EV5	1722
AlphaServer_1200	AlphaServer 1200 5/300	EV5	1724
AlphaServer_1200	AlphaServer 1200 5/400	EV56	1726
AlphaServer_1200	AlphaServer 1200 5/400	EV56	1728
AlphaServer_1200	AlphaStation 1200 5/400	EV56	1758
AlphaServer_1200	AlphaStation 1200 5/400	EV56	1760
AlphaServer_1200	AlphaServer 1200 5/466	EV56	1730
AlphaServer_1200	AlphaServer 1200 5/466	EV56	1732
AlphaServer_1200	AlphaStation 1200 5/466	EV56	1762
AlphaServer_1200	AlphaStation 1200 5/466	EV56	1764
AlphaServer_1200	AlphaServer 1200 5/533	EV56	1734
AlphaServer_1200	AlphaServer 1200 5/533	EV56	1736
AlphaServer_1200	AlphaServer 1200 5/533	EV56	1746
AlphaServer_1200	AlphaServer 1200 5/533	EV56	1748
AlphaServer_1200	AlphaStation 1200 5/533	EV56	1766
AlphaServer_1200	AlphaStation 1200 5/533	EV56	1768
AlphaServer_1200	AlphaStation 1200 5/533	EV56	1778
AlphaServer_1200	AlphaStation 1200 5/533	EV56	1780
AlphaServer_1200	AlphaServer 1200 5/600	EV56	1738
AlphaServer_1200	AlphaServer 1200 5/600	EV56	1740
AlphaServer_1200	AlphaServer 1200 5/600	EV56	1750
AlphaServer_1200	AlphaStation 1200 5/600	EV56	1752
AlphaServer_1200	AlphaStation 1200 5/600	EV56	1770
AlphaServer_1200	AlphaStation 1200 5/600	EV56	1772
AlphaServer_1200	AlphaStation 1200 5/600	EV56	1782
AlphaServer_1200	AlphaStation 1200 5/600	EV56	1784

AlphaServer_1200	AlphaServer 1200 5/666	EV56	1742
AlphaServer_1200	AlphaServer 1200 5/666	EV56	1744
AlphaServer_1200	AlphaServer 1200 5/666	EV56	1754
AlphaServer_1200	AlphaServer 1200 5/666	EV56	1756
AlphaServer_1200	AlphaStation 1200 5/666	EV56	1774
AlphaServer_1200	AlphaStation 1200 5/666	EV56	1776
AlphaServer_1200	AlphaStation 1200 5/666	EV56	1786
AlphaServer_1200	AlphaStation 1200 5/666	EV56	1788

AlphaServer 2000

set session hw_model=	set rom system_name=	set ace cpu_architecture=	set rom dsrdb[0]=
AlphaServer_2000	AlphaServer 2000 4/200	EV4	1123
AlphaServer_2000	AlphaServer 2000 4/233	EV45	1171
AlphaServer_2000	AlphaServer 2000 4/275	EV45	1127

AlphaServer 2100

set session hw_model=	set rom system_name=	set ace cpu_architecture=	set rom dsrdb[0]=
AlphaServer_2100	AlphaServer 2100 4/200	EV4	1059
AlphaServer_2100	AlphaServer 2100 4/200	EV4	1135
AlphaServer_2100	AlphaServer 2100 4/233	EV45	1179
AlphaServer_2100	AlphaServer 2100 4/233	EV45	1187
AlphaServer_2100	AlphaServer 2100 4/275	EV45	1115
AlphaServer_2100	AlphaServer 2100 4/275	EV45	1139

AlphaServer 4000

set session hw_model=	set rom system_name=	set ace cpu_architecture=	set rom dsrdb[0]=
AlphaServer_4000	AlphaServer 4000 5/266	EV5	1409
AlphaServer_4000	AlphaServer 4000 5/266	EV5	1411
AlphaServer_4000	AlphaServer 4000 5/266	EV5	1421
AlphaServer_4000	AlphaServer 4000 5/266	EV5	1423
AlphaServer_4000	AlphaServer 4000 5/266	EV5	1433
AlphaServer_4000	AlphaServer 4000 5/266	EV5	1435
AlphaServer_4000	AlphaServer 4000 5/266	EV5	1445
AlphaServer_4000	AlphaServer 4000 5/266	EV5	1447
AlphaServer_4000	AlphaServer 4000 5/300	EV5	1413
AlphaServer_4000	AlphaServer 4000 5/300	EV5	1415
AlphaServer_4000	AlphaServer 4000 5/300	EV5	1425
AlphaServer_4000	AlphaServer 4000 5/300	EV5	1427
AlphaServer_4000	AlphaServer 4000 5/300	EV5	1437
AlphaServer_4000	AlphaServer 4000 5/300	EV5	1439

AlphaServer_4000	AlphaServer 4000 5/300	EV5	1449
AlphaServer_4000	AlphaServer 4000 5/300	EV5	1451
AlphaServer_4000	AlphaServer 4000 5/400	EV56	1417
AlphaServer_4000	AlphaServer 4000 5/400	EV56	1419
AlphaServer_4000	AlphaServer 4000 5/400	EV56	1429
AlphaServer_4000	AlphaServer 4000 5/400	EV56	1431
AlphaServer_4000	AlphaServer 4000 5/400	EV56	1441
AlphaServer_4000	AlphaServer 4000 5/400	EV56	1443
AlphaServer_4000	AlphaServer 4000 5/400	EV56	1453
AlphaServer_4000	AlphaServer 4000 5/400	EV56	1455
AlphaServer_4000	AlphaServer 4000 5/466	EV56	1634
AlphaServer_4000	AlphaServer 4000 5/466	EV56	1636
AlphaServer_4000	AlphaServer 4000 5/466	EV56	1654
AlphaServer_4000	AlphaServer 4000 5/466	EV56	1656
AlphaServer_4000	AlphaServer 4000 5/533	EV56	1638
AlphaServer_4000	AlphaServer 4000 5/533	EV56	1640
AlphaServer_4000	AlphaServer 4000 5/533	EV56	1642
AlphaServer_4000	AlphaServer 4000 5/533	EV56	1644
AlphaServer_4000	AlphaServer 4000 5/533	EV56	1658
AlphaServer_4000	AlphaServer 4000 5/533	EV56	1660
AlphaServer_4000	AlphaServer 4000 5/533	EV56	1662
AlphaServer_4000	AlphaServer 4000 5/533	EV56	1664
AlphaServer_4000	AlphaServer 4000 5/600	EV56	1646
AlphaServer_4000	AlphaServer 4000 5/600	EV56	1648
AlphaServer_4000	AlphaServer 4000 5/600	EV56	1666
AlphaServer_4000	AlphaServer 4000 5/600	EV56	1668
AlphaServer_4000	AlphaServer 4000 5/666	EV56	1650
AlphaServer_4000	AlphaServer 4000 5/666	EV56	1652
AlphaServer_4000	AlphaServer 4000 5/666	EV56	1670
AlphaServer_4000	AlphaServer 4000 5/666	EV56	1672

AlphaServer 4100

set session hw_model=	set rom system_name=	set ace cpu_architecture=	set rom dsrdb[0]=
AlphaServer_4100	AlphaServer 4100 5/266	EV5	1313
AlphaServer_4100	AlphaServer 4100 5/266	EV5	1317
AlphaServer_4100	AlphaServer 4100 5/266	EV5	1337
AlphaServer_4100	AlphaServer 4100 5/266	EV5	1341
AlphaServer_4100	AlphaServer 4100 5/266	EV5	1361
AlphaServer_4100	AlphaServer 4100 5/266	EV5	1365
AlphaServer_4100	AlphaServer 4100 5/266	EV5	1385
AlphaServer_4100	AlphaServer 4100 5/266	EV5	1389
AlphaServer_4100	AlphaServer 4100 5/300	EV5	1321
AlphaServer_4100	AlphaServer 4100 5/300	EV5	1325
AlphaServer_4100	AlphaServer 4100 5/300	EV5	1345
AlphaServer_4100	AlphaServer 4100 5/300	EV5	1349
AlphaServer_4100	AlphaServer 4100 5/300	EV5	1369
AlphaServer_4100	AlphaServer 4100 5/300	EV5	1373
AlphaServer_4100	AlphaServer 4100 5/300	EV5	1393
AlphaServer_4100	AlphaServer 4100 5/300	EV5	1397
AlphaServer_4100	AlphaServer 4100 5/400	EV56	1329
AlphaServer_4100	AlphaServer 4100 5/400	EV56	1333
AlphaServer_4100	AlphaServer 4000 5/400	EV56	1353
AlphaServer_4100	AlphaServer 4000 5/400	EV56	1357
AlphaServer_4100	AlphaServer 4000 5/400	EV56	1377
AlphaServer_4100	AlphaServer 4100 5/400	EV56	1381
AlphaServer_4100	AlphaServer 4100 5/400	EV56	1401
AlphaServer_4100	AlphaServer 4100 5/400	EV56	1405
AlphaServer_4100	AlphaServer 4100 5/466	EV56	1594
AlphaServer_4100	AlphaServer 4100 5/466	EV56	1598
AlphaServer_4100	AlphaServer 4100 5/533	EV56	1602
AlphaServer_4100	AlphaServer 4100 5/533	EV56	1606
AlphaServer_4100	AlphaServer 4100 5/533	EV56	1610
AlphaServer_4100	AlphaServer 4100 5/533	EV56	1614
AlphaServer_4100	AlphaServer 4100 5/600	EV56	1618
AlphaServer_4100	AlphaServer 4100 5/600	EV56	1622
AlphaServer_4100	AlphaServer 4100 5/666	EV56	1626
AlphaServer_4100	AlphaServer 4100 5/666	EV56	1630

AlphaServer/AlphaStation DS10 and AlphaServer DS10L

set session hw_model=	set rom system_name=	set ace cpu_architecture=	set rom dsrdb[0]=
AlphaServer_DS10	AlphaServer DS10 6/466	EV6	1839
AlphaServer_DS10	AlphaStation DS10 6/466	EV6	1879
AlphaServer_DS10	AlphaStation XP900 6/466	EV6	1879
AlphaServer_DS10L	AlphaServer DS10L 6/466	EV6	1961
AlphaServer_DS10L	AlphaServer DS10L 67/616	EV67	1962
AlphaServer_DS10	AlphaStation DS10 67/616	EV67	1962
AlphaServer_DS10	AlphaServer DS10 67/616	EV67	1970

AlphaServer DS15 and AlphaStation DS15

set session hw_model=	set rom system_name=	set ace cpu_architecture=	set rom dsrdb[0]=
AlphaServer_DS15	AlphaServer DS15 68CB/1000	EV68	2047
AlphaServer_DS15	AlphaStation DS15 68CB/1000	EV68	2048
AlphaServer_DS15	AlphaServer TS15 68CB/1000	EV68	2049

AlphaServer DS20 and AlphaStation DS20

set session hw_model=	set rom system_name=	set ace cpu_architecture=	set rom dsrdb[0]=
AlphaServer_DS20	AlphaServer DS20 6/500	EV6	1838
AlphaServer_DS20	AlphaServer DS20E 6/500	EV6	1840
AlphaServer_DS20	AlphaServer DS20 6/500	EV6	1920
AlphaServer_DS20	AlphaServer DS20 6/500	EV6	1921
AlphaServer_DS20	AlphaServer DS20E 67/667	EV67	1939
AlphaServer_DS20	AlphaStation DS20E 6/500	EV6	1941
AlphaServer_DS20	AlphaStation DS20E 67/667	EV57	1943
AlphaServer_DS20	AlphaServer DS20E 68A/833	EV68	1964
AlphaServer_DS20	AlphaServer DS20E 68A/833	EV68	1982
AlphaServer_DS20	AlphaServer DS20L 68A/833	EV68	2006

AlphaServer DS25 and AlphaStation DS25

set session hw_model=	set rom system_name=	set ace cpu_architecture=	set rom dsrdb[0]=
AlphaServer_DS25	AlphaServer DS25 68CB/1000	EV68	1994
AlphaServer_DS25	AlphaStation DS25 68CB/1000	EV68	1995

AlphaServer ES40 and AlphaStation ES40

set session hw_model=	set rom system_name=	set ace cpu_architecture=	set rom dsrdb[0]=
AlphaServer_ES40	AlphaServer ES40 6/500	EV6	1813
AlphaServer_ES40	AlphaServer ES40 6/500	EV6	1861
AlphaServer_ES40	AlphaServer ES40 6/500	EV6	1869
AlphaServer_ES40	AlphaServer ES40 6/500	EV6	1923
AlphaServer_ES40	AlphaServer ES40 6/500	EV6	1931
AlphaServer_ES40	AlphaServer ES40 6/667	EV6	1817
AlphaServer_ES40	AlphaServer ES40 6/667	EV6	1865
AlphaServer_ES40	AlphaServer ES40 6/667	EV6	1873
AlphaServer_ES40	AlphaServer ES40 6/667	EV6	1927
AlphaServer_ES40	AlphaServer ES40 6/667	EV6	1935
AlphaServer_ES40	AlphaStation ES40 67/667	EV67	1949
AlphaServer_ES40	AlphaStation ES40 67/667	EV67	1957
AlphaServer_ES40	AlphaStation ES40 68/833	EV68	1984
AlphaServer_ES40	AlphaStation ES40 68/833	EV68	1988

AlphaServer ES45

set session hw_model=	set rom system_name=	set ace cpu_architecture=	set rom dsrdb[0]=
AlphaServer_ES45	AlphaServer ES45/3B 68CB/1000	EV68	1971
AlphaServer_ES45	AlphaServer ES45/2 68CB/1000	EV68	1975
AlphaServer_ES45	AlphaServer ES45/2B 68CB/1000	EV68	1975
AlphaServer_ES45	AlphaServer ES45/1B 68CB/1000	EV68	2002
AlphaServer_ES45	AlphaServer ES45/3B 68CB/1250	EV68	2013
AlphaServer_ES45	AlphaServer ES45/2 68CB/1250	EV68	2017
AlphaServer_ES45	AlphaServer ES45/2B 68CB/1250	EV68	2017
AlphaServer_ES45	AlphaServer ES45/1B 68CB/1250	EV68	2021

AlphaServer GS80

set session hw_model=	set rom system_name=	set ace cpu_architecture=	set rom dsrdb[0]=	set rom dsrdb[1]=	set rom dsrdb[4]=
AlphaServer_GS80	AlphaServer GS80 67/728	EV67	1967		
AlphaServer_GS80	AlphaServer GS1280	EV67	2038	50	3050

AlphaServer GS160

set session hw_model=	set rom system_name=	set ace cpu_architecture=	set rom dsrdb[0]=	set rom dsrdb[1]=	set rom dsrdb[4]=
AlphaServer_GS160	AlphaServer GS160 67/728	EV67	1968		
AlphaServer_GS160	AlphaServer GS1280	EV67	2039	50	3050

AlphaServer GS320

set session hw_model=	set rom system_name=	set ace cpu_architecture=	set rom dsrdb[0]=	set rom dsrdb[1]=	set rom dsrdb[4]=
AlphaServer_GS320	AlphaServer GS320 67/728	EV67	1969		
AlphaServer_GS320	AlphaServer GS1280	EV67	2040	50	3050

Auto Boot

CHARON VMs can be configured to boot the operating system automatically at start up to behave like hardware.

auto_action restart

Parameter	auto_action restart
Type	Text string

Value	Determines whether CHARON-AXP boots automatically if the correct boot flags are set (and saved in the Alpha console files).
	Example:
	`>>>`**`set bootdef_dev`**` dka0`
	`>>>`**`set auto_action restart`**

Setting System Marketing Model (SMM)

CHARON-AXP allows to set an exact System Marketing Model (SMM) for a given model of Alpha, for example:

```
set rom dsrdb[0]=1090
```

Refer to Core Devices#Setting of a particular Alpha model to find allowed values of SMM per each Alpha model supported by CHARON-AXP.

Console

Table of Contents

General Description

CHARON-AXP offers two-port serial console on all supported AXP models.

Example for OPA0 console ("COM1" port):

```
set COM1 alias=OPA0
```

ℹ Starting with Charon-AXP V4.12 build 204-14, the line above is optional. OPA0 is mapped by default to COM1.

Example when using the TTA0 console ("COM2" port):

```
set COM2 alias=TTA0
```

Refer to Console#Mapping Serial line controllers to system resources for details of mapping.

The CHARON emulated console supports only 80 symbols in one line.

General parameters

CHARON-AXP console lines COM1 and COM2 have the following parameters:

ℹ All the values in the following tables are case insensitive.

alias

Parameter	alias
Type	Identifier
Value	This parameter is used to set an useful name for COM1 or COM2 ports. It can be any name, for example "Console1", but usually it is "OPA0" for COM1 and "TTA0" for COM2. This name is logged in CHARON log file, it can also be used for parametrization in CHARON configuration file along with "COM1" and "COM2" identifiers. The main purpose of this parameter is migration from old CHARON systems (which do not have the described implementation of consoles) to the current design, since it allows retaining the original name used for parametrization, since the rest of the parameters stay the same in both implementations. ⚠ **This note applies to builds before 204-13** **Example**: `set COM2 alias=TTA0`

communication

Parameter	communication
Type	Text string
Value	- "ascii" - for connection to terminals (default) - "binary" - for binary (packet) protocols, which are used mainly for communicating with PLCs

baud

Parameter	baud
Type	Numeric
Value	Forces the baud rate of the corresponding COM port to be a specified value. The variety of supported values depends on the underlying physical communication resource (COM port). The most widely used values are: 300, 1200, 9600, 19200, 38400. **Example**: `set OPA0 baud=38400`

break_on

Parameter	break_on
Type	Text string

Value	Specifies what byte sequences received over the physical serial line will trigger a HALT command.
	This parameter works only for the console line.
	Specify the following values: "Ctrl-P", "Break" or "none" ("none" disables triggering a HALT condition).
	Please note:
	If your guest operating system is OpenVMS in addition to "none" setting you have to set a specific console parameter "controlp" to "off" in the following way:
	`>>> set controlp off`
	`>>> power off`
	The second line is to preserve the ROM settings.
	Example:
	`set OPA0 break_on="Ctrl-P"`
	The default value is "Break".

stop_on

Parameter	stop_on
Type	Text string
Value	Specifies what byte sequences received over the physical serial line will trigger a STOP condition. The STOP condition causes CHARON Virtual Machine (VM) to exit. Specify the one of the following values: “F6” or “none” ("none" disables triggering a STOP condition). **Example**: `set OPA0 stop_on="F6"` The default value is "none". Setting "F6" triggers the STOP condition upon receipt of the "`<ESC>[17~`" sequence. Terminals usually send these sequences by pressing the **F6** button ⚠ This parameter can be specified only for COM1 (OPA0) console

log

Parameter	log
Type	Text string

Value	A string specifying a file name to store the content of the console sessions or a directory where the log files for each individual session will be stored.
	If an existing directory is specified, CHARON-AXP automatically enables creation of individual log files, one for each session using the same scheme as used for the generation of the rotating log files. If the "log" parameter is omitted, CHARON-AXP does not create a console log.
	Example 1:
	`set OPA0 log="log.txt"`
	Example 2:
	`set OPA0 log="C:\Charon\Logs"`
	For rotating logs:
	• Specify only an existing directory
	• If you want to store the OPA0 log file in the Charon virtual machine home directory (default for virtual machine log file), use: `set OPA0 log=".\."`

log_file_size

Parameter	log_file_size
Type	Text string

Value	If log rotation is enabled, the log_file_size parameter determines the log file size threshold at which a new log is automatically created.
	• "unlimited" or "0" (default) - the feature is disabled
	• "default" - default size is used (4Mb)
	• <size>[KMG] - size of the current log file in bytes with additional multipliers:
	  • K - Kilobyte - multiply by 1024
	  • M - Megabyte - multiply by 1024*1024
	  • G - Gigabyte - multiply by 1024*1024*1024
	Example 1:
	`set OPA0 log_file_size="default"`
	Example 2:
	`set OPA0 log_file_size=10M`
	⚠ Minimum log file size is 64K, maximum is 1G. Setting size less then 64K effectively makes the log file unlimited.

log_flush_period

Parameter	log_flush_period
Type	Numeric
Value	• <period-in-seconds> Defines a period of flushing log to disk. Default period is 60 seconds (it means that every minute log file is flushed to disk) **Example**: `set OPA0 log_flush_period=30`

port

Parameter	port
Type	Numeric
Value	The TCP/IP port number for the virtual serial line. A virtual serial line always listens on this port for incoming connection requests. If multiple virtualized machines are running on a server, ensure the port number is unique across the platform.

application

Parameter	application
Type	Text string

Value	An application (a terminal emulator is assumed in most cases) to be started on initialization of this serial line emulation. The specified application startup string may contain all required parameters
	Example:
	`set OPA0 application = "putty -load CHTERM-VT100 -title OPA0@XYZ -P 10003"`
	In this example the terminal emulator application "putty" is started with the parameters:
	■ "-load `CHTERM-VT100`" telling it to load a specific saved session named "`CHTERM-VT100`" (created separately), from the host registry
	■ "-title" specifies the name of the given instance of putty terminal emulator
	■ "-P" stands for communication port used
	The "application" parameter is often combined with a "port" parameter:
	`set TTA0 port = 10004 application = "putty -load CHTERM-VT100 -title TTA0@XYZ -P 10004"`
	`The` **`CHTERM-VT100 settings`** `in form of ".reg" file can be found under Charon installation location in the subfolder "Build_xxxx\x64"`

connection_override

Parameter	connection_override
Type	text string

Value	"enable"
	Allows new connection to override existing connection, if any. Enabled connection override on OPA0 allows to intercept virtual serial console.
	When emulator detects new connection request on the port (10003 for the below example), it closes old connection, if any, and switches to the new one.
	Example:
	`set COM1 alias = OPA0`
	`set OPA0 port = 10003 connection_override = enable`
	⚠ This is implemented only for serial lines using the new syntax, not for lines using the legacy syntax (`load virtual_serial_line ...`).

access_control

ℹ **Available only since build 204-14**

Parameter	access_control
Type	text string
Value	"disable" Since build 204-13, Incoming connection requests are by default filtered for virtual serial lines and then allowed only for the localhost. This is to avoid security scanners that can block the port. **Example**: `set OPA0 access_control = disable` ⚠ This is implemented only for serial lines using the new syntax, not for lines using the legacy syntax (`load virtual_serial_line ...`).

Mapping Serial line controllers to system resources

line

Parameter	line
Type	Text string
Value	A defined COM port on a host system in the form of "\\.\COMn" (or "COM<n>:") **Example**: `set OPA0 line="\\.\COM1"`

OPA0 console configuration examples

Using legacy syntax (not recommended)

This example maps OPA0 to port 10003, enable F6 key (emulator stop) and logs the console input/output to a rotating log file in its home directory:

```
load virtual_serial_line OPA0

set OPA0 port=10003

set OPA0 stop_on="F6"

set OPA0 log=".\."
```

Using new syntax (recommended)

This example maps OPA0 to port 10003, enable F6 key (emulator stop), logs the console input/output to a rotating log file in its home directory and enables the connection_override feature:

```
set COM1 alias=OPA0 (optional starting with build 204-14)

set OPA0 port=10003

set OPA0 stop_on="F6"

set OPA0 log=".\."

set OPA0 connection_override=enable
```

Remote Management Console (RMC)

Table of Contents

General Description

The purpose of the Remote Management Console is to let the emulator trigger actions on the Charon host to properly unload and save a vtape container while the guest operating system is running.

Originally, the tape **offline** state in Charon-VAX and Charon-AXP was not persistent. An automatic, periodic loading function will bring the tape device **online** again using the same container. That means that subsequent operations might overwrite the tape container.

To improve this behavior, an initially persistent **offline** state was introduced that could be configured by providing an "empty tape drive" using a name of **.vtape**. With this improvement, a virtual tape device will remain **offline**, until a valid name is provided during the runtime of the emulator. This name can be provided in two ways:

- CHARONCP: SCSI devices on OpenVMS only (see "Charon Guest Utilities for OpenVMS" chapter)
- Remote Management Console (this section)

However, this first improvement did not allow to return to an "empty tape drive" once a valid name had been provided. Hence, the automatic loading process would once again create the problem of possibly overwriting an existing container. This problem triggered an improvement that will allow a **persistent offline state**. The persistent offline state is configured on platforms where it is supported by adding a new keyword to the removable parameter as shown in the example below from a configuration file:

```
set PKA container[0] = ".vtape" removable = "noauto"
```

With this configuration, a tape device in Charon behaves as follows. When the tape device is put into status **offline**, it clears the runtime value of the associated container turning it into an "empty tape drive". Any further operations on this tape device are only possible after changing value of the container to a meaningful value and loading the tape. This behavior makes the runtime **offline** state persistent.

To load the Remote Management Console use the following syntax:

```
load remote_management_console RMC
```

ℹ Only one Remote Management Console can be set per Charon instance.

Parameter

Parameter	port
Type	Numeric
Value	The TCP/IP port number for the remote management console. If multiple Charon instances are running on a server, ensure the port number is unique across the platform.

Examples

Example 1:

```
load remote_management_console RMC port=13000
```

Example 2:

```
load remote_management_console RMC
set RMC port=13000
```

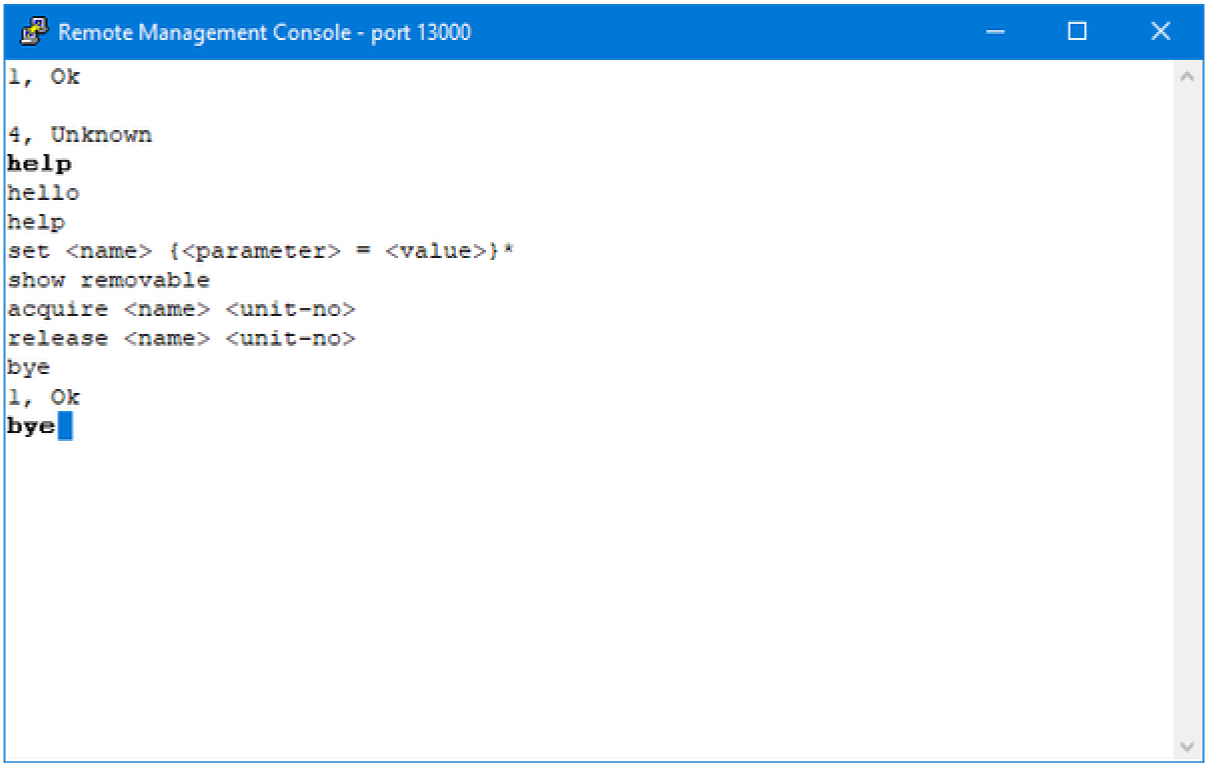
Connection

The RMC is reachable via the configured TCP port using (for example) a **telnet** client.

Notes:

- Line mode has to be used (default) and not character mode during telnet connection.
- If you use 'putty' to access the RMC, use "Raw" connection type and not "Telnet" and ensure the "Implicit CR in every LF" option in the "Terminal" settings is checked.
- If you access from a remote location to the RMC port, please define appropriate firewall rules if necessary.

Example:



```
1, Ok

4, Unknown
help
hello
help
set <name> {<parameter> = <value>}*
show removable
acquire <name> <unit-no>
release <name> <unit-no>
bye
1, Ok
bye
```

Available commands

Short description of the available commands:

Command	Description
help	Usage information as shown in the example above
hello	Currently no user-relevant function
set *<name>* **{***<parameter>* **=** *<value>***}**	The set command is used to set a meaningful name for a vtape container. **Example**: `set PUA container[11] = mytape1.vtape`
show removable	Lists the removable devices configured for the specific emulator instance.
acquire *<name> <unit-no>*	Brings a vtape **online**. **Example**: `acquire pua 11`
release *<name> <unit-no>*	Brings a vtape **offline**. If **noauto** is configured, an empty container name is set. **Example**: `release pua 11`
show virtual *<serial line>*	List configured serial lines. If run without parameters, all serial lines will be listed.
disconnect *<name> <unit-no>*	Disconnect virtual serial line. **Example**: **disconnect com2 0**
bye	Disconnect from RMC

Commands cannot be shortened

Example:

```
1, Ok
show removable
PUA, 11, offline, ".vtape"
PUA, 12, offline, ".vtape"
1, Ok
set PUA container[11] = xxx.vtape
1, Ok
show removable
PUA, 11, offline, "xxx.vtape"
PUA, 12, offline, ".vtape"
1, Ok
acquire pua 11
1, Ok
show removable
PUA, 11, available, "xxx.vtape"
PUA, 12, offline, ".vtape"
1, Ok
release pua 11
1, Ok
show removable
PUA, 11, offline, "xxx.vtape"
PUA, 12, offline, ".vtape"
1, Ok
bye
```

Future versions may contain additional features to support the automation of the functionality that the Remote Management Console provides. Currently, if you need support for an automation project, please contact your Stromasys representative or your Stromasys VAR.

Current connection can be closed disregard to connection_override settings.

Placement of peripheral devices on PCI bus

Table of Contents

General Description

Each peripheral device of CHARON Virtual Machine (VM) connects to CHARON-AXP emulated PCI bus with the following configuration parameters:

bus

Parameter	bus
Type	Text string

Value	When specified, the bus configuration parameter tells the CHARON-AXP software the virtual PIC bus to which virtual Alpha system shall connect a certain virtual PCI adapter.
	If the bus configuration parameter is not specified, Charon-AXP software connects the virtual PCI adapter to the first available virtual PCI bus. This is the default.
	■ AlphaServer 400-4100, DS, ES use: `"pci_<X>"`
	■ AlphaServer GS use: `"qbb_<X>_pca_<Y>_pci_<Z>"`
	Example (AlphaServer ES40):
	`load KZPBA PKA bus=pci_1`
	Example (AlphaServer GS80):
	`load KZPBA PKA bus=qbb_1_pca_1_pci_0`

device

Parameter	device
Type	Numeric
Value	When specified, the device configuration parameter specifies position of a virtual PCI adapter on virtual PCI bus. If the device configuration parameter is not specified, the CHARON VM software connects the virtual PCI adapter at the first available position of the virtual PCI bus. This is the default. **Example**: `load KZPBA PKA device=2`

function

Parameter	function
Type	Numeric
Value	When specified, the function configuration parameter specifies position of a virtual PCI adapter on virtual PCI bus. If the function configuration parameter is not specified, the CHARON VM software connects the virtual PCI adapter at the first available position of the virtual PCI bus. This is the default. **Example**: `load KZPBA PKA function=0`

irq_bus

Parameter	irq_bus
Type	Text string
Value	When specified, the "irq_bus" configuration parameter specifies virtual bus routing interrupt requests from virtual PCI adapter to CHARON-AXP virtual Alpha CPUs. By default the "irq_bus" configuration parameter is not specified. The "irq_bus" configuration parameter must be set to “isa” for AlphaServer 400. For Alpha systems other than AlphaServer 400 the "irq_bus" configuration parameter must be left as is (i.e. not specified). **Example**: `load KZPBA PKA irq_bus=isa`

irq

Parameter	irq
Type	Numeric
Value	When specified, the "irq" configuration parameter assigns interrupt request to the virtual PCI adapter in Alpha system. If the irq configuration parameter is not specified, the CHARON VM software uses the correct values depending on the selected PCI position of a virtual PCI adapter. Thisis the default. **Example**: `load KZPBA PKA irq=24`

Note that typically all or some of those parameters are specified on loading of some PCI controller in the following way:

```
load KZPBA PKA bus=pci_1 device=1 function=0 irq_bus=isa irq=24
```

Available PCI slots per each Alpha model emulated by CHARON-AXP

The tables below specifies a map of preloaded devices and available slots for each Alpha models emulated by CHARON-AXP.

AlphaServer 400 (3 PCI slots)

In addition to 3 PCI vacant slots there are 2 PCI positions occupied by on-board devices. All 5 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	pci_<N>	device	function	irq	Description	Preloaded Name
PCI0 (bus=pci_0)						
-	0	6	0	11	NCR 53C810 PCI SCSI Adapter	PKA
-	0	7	0	-	Intel i82378 PCI ISA Bridge (SATURN)	
0	0	11	0	10	<option>	
1	0	12	0	15	<option>	
2	0	13	0	9	<option>	

The IRQ stands for ISA IRQ Number because all interrupts are routed through the Intel i82378 PCI ISA Bridge (SATURN) resident cascade of Intel i8259 interrupt controllers.

So far the CHARON-AXP VMs do not support virtual NCR 53C810 PCI SCSI adapter. Instead, virtual QLOGIC ISP1040B PCI SCSI adapter is used.

ℹ No support for Multi-Function PCI devices in AlphaServer 400.

Example: Loading DE435 into slot 0

```
load DE435/dec21x4x EWA bus=pci_0 device=11 function=0 irq_bus=isa
```

ℹ The "irq_bus=isa" setting is specific to AlphaServer 400 only and must be present.

AlphaServer 800 (4 PCI slots)

In addition to 4 PCI vacant slots there are 3 PCI positions occupied by on-board devices. All 7 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	pci_<N>	device	function	irq	Description	Preloaded Name
PCI0 (bus=pci_0)						
-	0	5	0	0	QLOGIC ISP1020 PCI SCSI Adapter	PKA
-	0	6	0	0	S3 Trio32/64 Display Adapter	
-	0	7	0	-	Intel i82375 PCI EISA Bridge (MERCURY)	
0	0	11	0	1	<option>	
			1	2	<option>, function 1	
			2	17	<option>, function 2	
			3	18	<option>, function 3	
1	0	12	0	3	<option>	
			1	4	<option>, function 1	
			2	19	<option>, function 2	
			3	20	<option>, function 3	
2	0	13	0	5	<option>	
			1	6	<option>, function 1	
			2	21	<option>, function 2	
			3	22	<option>, function 3	
3	0	14	0	7	<option>	
			1	8	<option>, function 1	
			2	23	<option>, function 2	
			3	24	<option>, function 3	

The IRQ stands for input line of ASIC interrupt controllers. It has nothing to do with "EISA" style interrupts. So far, the CHARON-AXP VMs do not emulate S3 Trio32/64 Display Adapter. So position of the device 6, function 0 on the PCI 0 remains empty.

Example 1: Loading DE500BA into slot 0

```
load DE500BA/dec21x4x EWA bus=pci_0 device=11 function=0
```

Example 2: Loading multiple DE500BA's into slot 3, populating all 4 functions (gives 4 Ethernet ports)

```
load DE500BA/dec21x4x EWA bus=pci_0 device=14 function=0
load DE500BA/dec21x4x EWB bus=pci_0 device=14 function=1
load DE500BA/dec21x4x EWC bus=pci_0 device=14 function=2
load DE500BA/dec21x4x EWD bus=pci_0 device=14 function=3
```

Example 3: Loading mixture of KZPBA and DE500BA into slot 1, populating 2 functions out of 4

```
load KZPBA PKB bus=pci_0 device=12 function=0
load DE500BA/dec21x4x EWA bus=pci_0 device=12 function=1
```

In the above example device name is PKB as there is a built-in PK-like PCI SCSI Adapter located “closer” to CPU and therefore assigned name PKA.

AlphaServer 1000 (3 PCI slots)

In addition to 3 PCI vacant slots there are 2 PCI positions occupied by on-board devices. All 5 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	pci_<N>	device	function	irq	Description	Preloaded Name
PCI0 (bus=pci_0)						
-	0	6	0	12	NCR 53C810 PCI SCSI Adapter	PKA
-	0	7	0	-	Intel i82375 PCI EISA Bridge (MERCURY)	
0	0	11	0	0	<option>	
			1	1	<option>, function 1	
			2	2	<option>, function 2	
			3	3	<option>, function 3	
1	0	12	0	4	<option>	
			1	5	<option>, function 1	
			2	6	<option>, function 2	
			3	7	<option>, function 3	
2	0	13	0	8	<option>	
			1	9	<option>, function 1	
			2	10	<option>, function 2	
			3	11	<option>, function 3	

The IRQ stands for input line of ASIC interrupt controllers. It has nothing to do with "EISA" style interrupts. So far, the CHARON-AXP VMs do not emulate NCR 53C810 PCI SCSI adapter. Instead, emulation of QLOGIC ISP1040B PCI SCSI adapter is used.

Example 1: Loading DE500BA into slot 0

```
load DE500BA/dec21x4x EWA bus=pci_0 device=11 function=0
```

Example 2: Loading multiple DE500BA's into slot 0, populating all 4 functions (gives 4 Ethernet ports)

```
load DE500BA/dec21x4x EWA bus=pci_0 device=11 function=0
load DE500BA/dec21x4x EWB bus=pci_0 device=11 function=1
load DE500BA/dec21x4x EWC bus=pci_0 device=11 function=2
load DE500BA/dec21x4x EWD bus=pci_0 device=11 function=3
```

Example 3: Loading mixture of KZPBA and DE500BA into slot 2, populating 2 functions out of 4

```
load KZPBA PKB bus=pci_0 device=13 function=0
load DE500BA/dec21x4x EWA bus=pci_0 device=13 function=1
```

In the above example device name is PKB as there is a built-in PK-like PCI SCSI Adapter located “closer” to CPU and therefore assigned name PKA.

AlphaServer 1000A (7 PCI slots)

In addition to 7 PCI vacant slots there are 3 PCI positions occupied by on-board devices. All 10 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	pci_<N>	device	function	irq	Description	Preloaded Name
PCI0 (bus=pci_0)						
-	0	6	0	-	Intel i82375 PCI EISA Bridge (MERCURY)	
-	0	7	0	-	DECchip 21050 PCI-to-PCI Bridge)	
0	0	11	0	1	<option>	
			1	2	<option>, function 1	
			2	17	<option>, function 2	
			3	18	<option>, function 3	
1	0	12	0	2	<option>	
			1	3	<option>, function 1	
			2	19	<option>, function 2	
			3	20	<option>, function 3	
2	0	13	0	3	<option>	
			1	4	<option>, function 1	
			2	21	<option>, function 2	
			3	22	<option>, function 3	
PCI1 (bus=pci_1)						
-	1	0	0	0	NCR 53C810 PCI SCSI Adapter	PKA
3	1	1	0	7	<option>	
			1	8	<option>, function 1	
			2	23	<option>, function 2	
			3	24	<option>, function 3	
4	1	2	0	9	<option>	
			1	10	<option>, function 1	
			2	25	<option>, function 2	
			3	26	<option>, function 3	
5	1	3	0	11	<option>	
			1	12	<option>, function 1	
			2	27	<option>, function 2	
			3	28	<option>, function 3	
6	1	4	0	13	<option>	
			1	14	<option>, function 1	
			2	29	<option>, function 2	
			3	30	<option>, function 3	

The IRQ stands for input line of ASIC interrupt controllers. It has nothing to do with "EISA" style interrupts. So far, the CHARON-AXP VMs do not emulate NCR 53C810 PCI SCSI adapter. Instead, emulation of QLOGIC ISP1040B PCI SCSI adapter is used.

Example 1: Loading DE500BA into slot 0

```
load DE500BA EWA bus=pci_0 device=11 function=0
```

Example 2: Loading multiple DE500BA’s into slot 0, populating all 4 functions (gives 4 Ethernet ports)

```
load DE500BA EWA bus=pci_0 device=11 function=0
load DE500BA EWB bus=pci_0 device=11 function=1
load DE500BA EWC bus=pci_0 device=11 function=2
load DE500BA EWD bus=pci_0 device=11 function=3
```

Example 3: Loading mixture of KZPBA and DE500BA into slot 3, populating 2 functions out of 4

```
load KZPBA PKB bus=pci_1 device=1 function=0
load DE500BA EWA bus=pci_1 device=1 function=1
```

ℹ In the above example device name is PKB as there is a built-in PK-like PCI SCSI Adapter located “closer” to CPU and therefore assigned name PKA.

AlphaServer 1200 (6 PCI slots)

In addition to 6 PCI vacant slots there are 2 PCI positions occupied by on-board devices. All 8 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	pci_<N>	device	function	irq	Description	Preloaded Name
PCI1 (bus=pci_1)						
-	1	1	0	4	NCR 53C810 PCI SCSI Adapter	PKA
0	1	2	0	8	<option>	
			1	9	<option>, function 1	
			2	10	<option>, function 2	
			3	11	<option>, function 3	
1	1	3	0	12	<option>	
			1	13	<option>, function 1	
			2	14	<option>, function 2	
			3	16	<option>, function 3	
2	1	4	0	16	<option>	
			1	17	<option>, function 1	
			2	18	<option>, function 2	
			3	19	<option>, function 3	
PCI0 (bus=pci_0)						
-	0	1	0	-	Intel i82375 PCI EISA Bridge (MERCURY)	
4	0	2	0	8	<option>	
			1	9	<option>, function 1	
			2	10	<option>, function 2	
			3	11	<option>, function 3	
5	0	3	0	12	<option>	
			1	13	<option>, function 1	
			2	14	<option>, function 2	
			3	15	<option>, function 3	
6	0	4	0	16	<option>	
			1	17	<option>, function 1	
			2	18	<option>, function 2	
			3	19	<option>, function 3	

So far, the CHARON-AXP VMs do not emulate NCR 53C810 PCI SCSI adapter. Instead, emulation of QLOGIC ISP1040B PCI SCSI adapter is used.

Example 1: Loading DE500BA into slot 4

```
load DE500BA/dec21x4x EWA bus=pci_0 device=2 function=0
```

Example 2: Loading multiple DE500BA’s into slot 4, populating all 4 functions (gives 4 Ethernet ports)

```
load DE500BA/dec21x4x EWA bus=pci_0 device=2 function=0
load DE500BA/dec21x4x EWB bus=pci_0 device=2 function=1
load DE500BA/dec21x4x EWC bus=pci_0 device=2 function=2
load DE500BA/dec21x4x EWD bus=pci_0 device=2 function=3
```

Example 3: Loading mixture of KZPBA and DE500BA into slot 1, populating 2 functions out of 4

```
load KZPBA PKB bus=pci_1 device=2 function=0
load DE500BA/dec21x4x EWA bus=pci_1 device=2 function=1
```

ℹ In the above example device name is PKB as there is a built-in PK-like PCI SCSI Adapter located “closer” to CPU and therefore assigned name PKA.

AlphaServer 2000 (3 PCI slots)

In addition to 3 PCI vacant slots there are 3 PCI positions occupied by on-board devices. All 6 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	pci_<N>	device	function	irq	Description	Preloaded Name
PCI0 (bus=pci_0)						
-	0	0	0	2	DEC TULIP PCI Ethernet adapter	EWA
-	0	1	0	1	NCR 53C810 PCI SCSI Adapter	PKA
-	0	2	0	-	Intel i82375 PCI EISA Bridge (MERCURY)	
0	0	6	0	0	<option>	
			1	24	<option>, function 1	
			2	26	<option>, function 2	
			3	29	<option>, function 3	
1	0	7	0	4	<option>	
			1	25	<option>, function 1	
			2	27	<option>, function 2	
			3	30	<option>, function 3	
2	0	8	0	5	<option>	
			1	20	<option>, function 1	
			2	28	<option>, function 2	
			3	31	<option>, function 3	

The IRQ stands for input line of T2 resident cascade of Intel i8259 interrupt controllers. It has nothing to do with “EISA” style interrupts.

So far the CHARON-AXP emulators do not support virtual NCR 53C810 PCI SCSI adapter. Instead, virtual QLOGIC ISP1040B PCI SCSI adapter is used.

Example 1: Loading DE500BA into slot 0

```
load DE500BA/dec21x4x EWB bus=pci_0 device=6 function=0
```

Example 2: Loading multiple DE500BA’s into slot 0, populating all 4 functions (gives 4 Ethernet ports)

```
load DE500BA/dec21x4x EWB bus=pci_0 device=6 function=0
load DE500BA/dec21x4x EWC bus=pci_0 device=6 function=1
load DE500BA/dec21x4x EWD bus=pci_0 device=6 function=2
load DE500BA/dec21x4x EWE bus=pci_0 device=6 function=3
```

ℹ In the above examples device name is EWB as there is a built-in EW-like PCI Ethernet Adapter located “closer” to CPU and therefore assigned name EWA.

Example 3: Loading mixture of KZPBA and DE500BA into slot 1, populating 2 functions out of 4

```
load KZPBA PKB bus=pci_0 device=7 function=0
load DE500BA/dec21x4x EWB bus=pci_0 device=7 function=1
```

ℹ In the above example device name is PKB as there is a built-in PK-like PCI SCSI Adapter located “closer” to CPU and therefore assigned name PKA, and device name is EWB as there is a built-in EW-like PCI Ethernet Adapter located “closer” to CPU and therefore assigned name EWA.

AlphaServer 2100 (3 PCI slots)

In addition to 3 PCI vacant slots there are 3 PCI positions occupied by on-board devices. All 6 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	pci_<N>	device	function	irq	Description	Preloaded Name
PCI0 (bus=pci_0)						
-	0	0	0	2	DEC TULIP PCI Ethernet adapter	EWA
-	0	1	0	1	NCR 53C810 PCI SCSI Adapter	PKA
-	0	2	0	-	Intel i82375 PCI EISA Bridge (MERCURY)	
0	0	6	0	0	<option>	
			1	24	<option>, function 1	
			2	26	<option>, function 2	
			3	29	<option>, function 3	
1	0	7	0	4	<option>	
			1	25	<option>, function 1	
			2	27	<option>, function 2	
			3	30	<option>, function 3	
2	0	8	0	5	<option>	
			1	20	<option>, function 1	
			2	28	<option>, function 2	
			3	31	<option>, function 3	

The IRQ stands for input line of T2 resident cascade of Intel i8259 interrupt controllers. It has nothing to do with “EISA” style interrupts.

So far the CHARON-AXP VMs do not support virtual NCR 53C810 PCI SCSI adapter. Instead, virtual QLOGIC ISP1040B PCI SCSI adapter is used.

Example 1: Loading DE500BA into slot 0

```
load DE500BA/dec21x4x EWB bus=pci_0 device=6 function=0
```

Example 2: Loading multiple DE500BA’s into slot 0, populating all 4 functions (gives 4 Ethernet ports)

```
load DE500BA/dec21x4x EWB bus=pci_0 device=6 function=0
load DE500BA/dec21x4x EWC bus=pci_0 device=6 function=1
load DE500BA/dec21x4x EWD bus=pci_0 device=6 function=2
load DE500BA/dec21x4x EWE bus=pci_0 device=6 function=3
```

ℹ In the above examples device name is EWB as there is a built-in EW-like PCI Ethernet Adapter located “closer” to CPU and therefore assigned name EWA.

Example 3: Loading mixture of KZPBA and DE500BA into slot 1, populating 2 functions out of 4

```
load KZPBA PKB bus=pci_0 device=7 function=0
load DE500BA/dec21x4x EWB bus=pci_0 device=7 function=1
```

ℹ In the above example device name is PKB as there is a built-in PK-like PCI SCSI Adapter located “closer” to CPU and therefore assigned name PKA, and device name is EWB as there is a built-in EW-like PCI Ethernet Adapter located “closer” to CPU and therefore assigned name EWA.

AlphaServer 4000 (16 PCI slots)

In addition to 16 PCI vacant slots there are 2 PCI positions occupied by on-board devices. All 18 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	pci_<N>	device	function	irq	Description	Preloaded Name
PCI1 (bus=pci_1)						
-	1	1	0	4	NCR 53C810 PCI SCSI Adapter	PKA
1	1	2	0	8	<option>	
			1	9	<option>, function 1	
			2	10	<option>, function 2	
			3	11	<option>, function 3	
2	1	3	0	12	<option>	
			1	13	<option>, function 1	
			2	14	<option>, function 2	
			3	15	<option>, function 3	
3	1	4	0	16	<option>	
			1	17	<option>, function 1	
			2	18	<option>, function 2	
			3	19	<option>, function 3	
4	1	5	0	20	<option>	
			1	21	<option>, function 1	
			2	22	<option>, function 2	
			3	23	<option>, function 3	
PCI0 (bus=pci_0)						
-	0	1	0	-	Intel i82375 PCI EISA Bridge (MERCURY)	
5	0	2	0	8	<option>	
			1	9	<option>, function 1	
			2	10	<option>, function 2	
			3	11	<option>, function 3	
6	0	3	0	12	<option>	
			1	13	<option>, function 1	
			2	14	<option>, function 2	
			3	15	<option>, function 3	
7	0	4	0	16	<option>	
			1	17	<option>, function 1	
			2	18	<option>, function 2	
			3	19	<option>, function 3	
8	0	5	0	20	<option>	
			1	21	<option>, function 1	
			2	22	<option>, function 2	
			3	23	<option>, function 3	

PCI3 (bus=pci_3)						
9	3	2	0	8	<option>	
			1	9	<option>, function 1	
			2	10	<option>, function 2	
			3	11	<option>, function 3	
10	3	3	0	12	<option>	
			1	13	<option>, function 1	
			2	14	<option>, function 2	
			3	15	<option>, function 3	
11	3	4	0	16	<option>	
			1	17	<option>, function 1	
			2	18	<option>, function 2	
			3	19	<option>, function 3	
12	3	5	0	20	<option>	
			1	21	<option>, function 1	
			2	22	<option>, function 2	
			3	23	<option>, function 3	
PCI2 (bus=pci_2)						
13	2	2	0	8	<option>	
			1	9	<option>, function 1	
			2	10	<option>, function 2	
			3	11	<option>, function 3	
14	2	3	0	12	<option>	
			1	13	<option>, function 1	
			2	14	<option>, function 2	
			3	15	<option>, function 3	
15	2	4	0	16	<option>	
			1	17	<option>, function 1	
			2	18	<option>, function 2	
			3	19	<option>, function 3	
16	2	5	0	20	<option>	
			1	21	<option>, function 1	
			2	22	<option>, function 2	
			3	23	<option>, function 3	

So far the CHARON-AXP VMs do not support virtual NCR 53C810 PCI SCSI adapter. Instead, virtual QLOGIC ISP1040B PCI SCSI adapter is used.

Example 1: Loading DE500BA into slot 4

```
load DE500BA/dec21x4x EWA bus=pci_1 device=5 function=0
```

Example 2: Loading multiple DE500BA's into slot 4, populating all 4 functions (gives 4 Ethernet ports)

```
load DE500BA/dec21x4x EWA bus=pci_1 device=5 function=0
load DE500BA/dec21x4x EWB bus=pci_1 device=5 function=1
load DE500BA/dec21x4x EWC bus=pci_1 device=5 function=2
load DE500BA/dec21x4x EWD bus=pci_1 device=5 function=3
```

Example 3: Loading mixture of KZPBA and DE500BA into slot 1, populating 2 functions out of 4

```
load KZPBA PKB bus=pci_1 device=2 function=0
load DE500BA/dec21x4x EWA bus=pci_1 device=2 function=1
```

ℹ In the above example device name is PKB as there is a built-in PK-like PCI SCSI Adapter located “closer” to CPU and therefore assigned name PKA.

AlphaServer 4100 (8 PCI slots)

In addition to 8 PCI vacant slots there are 2 PCI positions occupied by on-board devices. All 10 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	pci_<N>	device	function	irq	Description	Preloaded Name
PCI1 (bus=pci_1)						
-	1	1	0	4	NCR 53C810 PCI SCSI Adapter	PKA
1	1	2	0	8	<option>	
			1	9	<option>, function 1	
			2	10	<option>, function 2	
			3	11	<option>, function 3	
2	1	3	0	12	<option>	
			1	13	<option>, function 1	
			2	14	<option>, function 2	
			3	15	<option>, function 3	
3	1	4	0	16	<option>	
			1	17	<option>, function 1	
			2	18	<option>, function 2	
			3	19	<option>, function 3	
4	1	5	0	20	<option>	
			1	21	<option>, function 1	
			2	22	<option>, function 2	
			3	23	<option>, function 3	
PCI0 (bus=pci_0)						
-	0	1	0	-	Intel i82375 PCI EISA Bridge (MERCURY)	
5	0	2	0	8	<option>	
			1	9	<option>, function 1	
			2	10	<option>, function 2	
			3	11	<option>, function 3	
6	0	3	0	12	<option>	
			1	13	<option>, function 1	
			2	14	<option>, function 2	
			3	15	<option>, function 3	
7	0	4	0	16	<option>	
			1	17	<option>, function 1	
			2	18	<option>, function 2	
			3	19	<option>, function 3	
8	0	5	0	20	<option>	
			1	21	<option>, function 1	
			2	22	<option>, function 2	
			3	23	<option>, function 3	

So far the CHARON-AXP VMs do not support virtual NCR 53C810 PCI SCSI adapter. Instead, virtual QLOGIC ISP1040B PCI SCSI adapter is used.

Example 1: Loading DE500BA into slot 4

```
load DE500BA/dec21x4x EWA bus=pci_1 device=5 function=0
```

Example 2: Loading multiple DE500BA’s into slot 4, populating all 4 functions (gives 4 Ethernet ports)

```
load DE500BA/dec21x4x EWA bus=pci_1 device=5 function=0
load DE500BA/dec21x4x EWB bus=pci_1 device=5 function=1
load DE500BA/dec21x4x EWC bus=pci_1 device=5 function=2
load DE500BA/dec21x4x EWD bus=pci_1 device=5 function=3
```

Example 3: Loading mixture of KZPBA and DE500BA into slot 1, populating 2 functions out of 4

```
load KZPBA PKB bus=pci_1 device=2 function=0
load DE500BA/dec21x4x EWA bus=pci_1 device=2 function=1
```

ℹ In the above example device name is PKB as there is a built-in PK-like PCI SCSI Adapter located “closer” to CPU and therefore assigned name PKA.

AlphaServer DS10 (4 PCI slots)

In addition to 4 PCI vacant slots there are 5 PCI positions occupied by on-board devices. All 9 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	pci_<N>	device	function	irq	Description	Preloaded Name
PCI1 (bus=pci_0)						
-	0	7	0	-	ALi M1543C PCI ISA bridge	
-	0	9	0	29	DECchip 21143 PCI Ethernet Adapter	EWA
-	0	11	0	30	DECchip 21143 PCI Ethernet Adapter	EWB
-	0	13	0	-	ALi M1543C PCI IDE/ATAPI controller	DQA, DQB
1	0	14	0	35	<option>	
			1	34	<option>, function 1	
			2	33	<option>, function 2	
			3	32	<option>, function 3	
2	0	15	0	39	<option>	
			1	38	<option>, function 1	
			2	37	<option>, function 2	
			3	36	<option>, function 3	
3	0	16	0	43	<option>	
			1	42	<option>, function 1	
			2	41	<option>, function 2	
			3	40	<option>, function 3	
4	0	17	0	47	<option>	
			1	46	<option>, function 1	
			2	45	<option>, function 2	
			3	44	<option>, function 3	
-	0	19	0	11	ALi M1543C PCI USB adapter	

Example 1: Loading DE500BA into slot 1

```
load DE500BA/dec21x4x EWC bus=pci_0 device=14 function=0
```

Example 2: Loading multiple DE500BA’s into slot 1, populating all 4 functions (gives 4 Ethernet ports)

```
load DE500BA/dec21x4x EWC bus=pci_0 device=14 function=0
load DE500BA/dec21x4x EWD bus=pci_0 device=14 function=1
load DE500BA/dec21x4x EWE bus=pci_0 device=14 function=2
load DE500BA/dec21x4x EWF bus=pci_0 device=14 function=3
```

ℹ In the above examples device name is EWC as there are built-in EW-like PCI Ethernet Adapters located “closer” to CPU and therefore assigned names EWA and EWB.

Example 3: Loading mixture of KZPBA and DE500BA into slot 1, populating 2 functions out of 4

```
load KZPBA PKB bus=pci_0 device=14 function=0
load DE500BA/dec21x4x EWC bus=pci_0 device=14 function=1
```

ℹ In the above example device name is PKB as there is a built-in PK-like PCI SCSI Adapter located “closer” to CPU and therefore assigned name PKA, as there are two built-in EW-like PCI Ethernet Adapters located “closer” to CPU and therefore assigned names EWA and EWB.

AlphaServer DS10L (1 PCI slot)

In addition to 1 PCI vacant slots there are 5 PCI positions occupied by on-board devices. All 6 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	pci_<N>	device	function	irq	Description	Preloaded Name
PCI1 (bus=pci_0)						
-	0	7	0	-	ALi M1543C PCI ISA bridge	
-	0	9	0	29	DECchip 21143 PCI Ethernet Adapter	EWA
-	0	11	0	30	DECchip 21143 PCI Ethernet Adapter	EWB
-	0	13	0	-	ALi M1543C PCI IDE/ATAPI controller	DQA, DQB
1	0	17	0	47	<option>	
			1	46	<option>, function 1	
			2	45	<option>, function 2	
			3	44	<option>, function 3	
-	0	19	0	11	ALi M1543C PCI USB adapter	

Example 1: Loading DE500BA into slot 1

```
load DE500BA/dec21x4x EWC bus=pci_0 device=17 function=0
```

Example 2: Loading multiple DE500BA's into slot 1, populating all 4 functions (gives 4 Ethernet ports)

```
load DE500BA/dec21x4x EWC bus=pci_0 device=17 function=0
load DE500BA/dec21x4x EWD bus=pci_0 device=17 function=1
load DE500BA/dec21x4x EWE bus=pci_0 device=17 function=2
load DE500BA/dec21x4x EWF bus=pci_0 device=17 function=3
```

ℹ In the above examples device name is EWC as there are built-in EW-like PCI Ethernet Adapters located "closer" to CPU and therefore assigned names EWA and EWB.

Example 3: Loading mixture of KZPBA and DE500BA into slot 1, populating 2 functions out of 4

```
load KZPBA PKB bus=pci_0 device=17 function=0
load DE500BA/dec21x4x EWC bus=pci_0 device=17 function=1
```

ℹ In the above example device name is PKB as there is a built-in PK-like PCI SCSI Adapter located "closer" to CPU and therefore assigned name PKA, as there are two built-in EW-like PCI Ethernet Adapters located "closer" to CPU and therefore assigned names EWA and EWB

AlphaServer DS15 (4 PCI slots)

In addition to 4 PCI vacant slots there are 7 PCI positions occupied by on-board devices. All 11 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	pci_<N>	device	function	irq	Description	Preloaded Name
PCI0 (bus=pci_0)						
-	0	7	0	-	ALi M1543C PCI ISA bridge	
-	0	8	0	-	Adaptec AIC-7899 (channel 0)	PKA
			1	-	Adaptec AIC-7899 (channel 1)	PKB
-	0	9	0	-	Intel i82559 PCI Ethernet Adapter	EIA (EWA)
-	0	10	0	-	Intel i82559 PCI Ethernet Adapter	EIB (EWB)
-	0	13	0	-	ALi M1543C PCI IDE/ATAPI controller	DQA, DQB
-	0	19	0	-	ALi M1543C PCI USB adapter	
PCI2 (bus=pci_2)						
1	2	7	0	40	<option>	
			1	41	<option>, function 1	
			2	42	<option>, function 2	
			3	43	<option>, function 3	
2	2	8	0	36	<option>	
			1	37	<option>, function 1	
			2	38	<option>, function 2	
			3	39	<option>, function 3	
3	2	9	0	24	<option>	
			1	25	<option>, function 1	
			2	26	<option>, function 2	
			3	27	<option>, function 3	
4	2	10	0	20	<option>	
			1	21	<option>, function 1	
			2	22	<option>, function 2	
			3	23	<option>, function 3	

The IRQ stands for bit position in DRIR of TITAN chip. It has nothing to do with “ISA” style interrupts which are routed to IRQ 55 (including ALi M1543C PCI IDE/ATAPI controller).

So far the CHARON-AXP emulators do not emulate Adaptec AIC-7899. Instead, emulation of QLOGIC ISP1040B is used.

So far the CHARON-AXP emulators do not emulate Intel i82559. Instead, emulation of DECchip 21143 is used.

So far the CHARON-AXP emulators do not emulate ALi M1543C PCI USB adapter. So position of the device 19, function 0 on the PCI 0 remains empty.

Example 1: Loading DE500BA into slot 1

```
load DE500BA/dec21x4x EWC bus=pci_2 device=7 function=0
```

Example 2: Loading multiple DE500BA's into slot 2, populating all 4 functions (gives 4 Ethernet ports)

```
load DE500BA/dec21x4x EWC bus=pci_2 device=8 function=0
load DE500BA/dec21x4x EWD bus=pci_2 device=8 function=1
load DE500BA/dec21x4x EWE bus=pci_2 device=8 function=2
load DE500BA/dec21x4x EWF bus=pci_2 device=8 function=3
```

ℹ In the above examples device name is EWC as there are built-in EW-like PCI Ethernet Adapters located “closer” to CPU and therefore assigned names EWA and EWB.

Example 3: Loading mixture of KZPBA and DE500BA into slot 3, populating 2 functions out of 4

```
load KZPBA PKC bus=pci_2 device=9 function=0
load DE500BA/dec21x4x EWC bus=pci_2 device=9 function=1
```

ℹ In the above example device name is PKC as there are 2 built-in PK-like PCI SCSI Adapter located “closer” to CPU and therefore assigned name PKA and PKB, as there are two built-in EW-like PCI Ethernet Adapters located “closer” to CPU and therefore assigned names EWA and EWB

AlphaServer DS20 (6 PCI slots)

In addition to 6 PCI vacant slots there are 5 PCI positions occupied by on-board devices. All 11 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	pci_<N>	device	function	irq	Description	Preloaded Name
PCI1 (bus=pci_1)						
4	1	7	0	47	<option>	
			1	46	<option>, function 1	
			2	45	<option>, function 2	
			3	44	<option>, function 3	
5	1	8	0	43	<option>	
			1	42	<option>, function 1	
			2	41	<option>, function 2	
			3	49	<option>, function 3	
6	1	9	0	39	<option>	
			1	38	<option>, function 1	
			2	37	<option>, function 2	
			3	36	<option>, function 3	
PCI0 (bus=pci_0)						
-	0	5	0	-	ALi M1543C PCI ISA bridge	
-	0	6	0	19	Adaptec AIC-7895 (channel 0)	PKA
			1	18	Adaptec AIC-7895 (channel 1)	PKB
-	0	15	0	-	ALi M1543C PCI IDE/ATAPI controller	DQA, DQB
-	0	19	0	-	ALi M1543C PCI USB adapter	
1	0	7	0	31	<option>	
			1	30	<option>, function 1	
			2	29	<option>, function 2	
			3	28	<option>, function 3	
2	0	8	0	27	<option>	
			1	26	<option>, function 1	
			2	25	<option>, function 2	
			3	24	<option>, function 3	
3	0	9	0	23	<option>	
			1	22	<option>, function 1	
			2	21	<option>, function 2	
			3	20	<option>, function 3	

The IRQ stands for bit position in DRIR of Tsunami/Typhoon Chip. It has nothing to do with “ISA” style interrupts which are routed to IRQ 55 (including ALi M1543C PCI IDE/ATAPI controller).

Unless SCSI option is plugged into PCI slot 4, 5, or 6, the onboard SCSI controllers appear as PKA (pka7.0.0.6.0) and PKB (pkb7.0.0.106.0) respectively.

So far the CHARON-AXP VMs do not support virtual Adaptec AIC-7895 PCI SCSI adapter. Instead, virtual QLOGIC ISP1040B PCI SCSI adapter is used.

So far the CHARON-AXP eVMs do not support virtual ALi M1543C PCI USB adapter. So position of the device 19, function 0 on the PCI 0 remains empty

Example 1: Loading DE500BA into slot 4

```
load DE500BA/dec21x4x EWA bus=pci_1 device=7 function=0
```

Example 2: Loading multiple DE500BA’s into slot 4, populating all 4 functions (gives 4 Ethernet ports)

```
load DE500BA/dec21x4x EWA bus=pci_1 device=7 function=0
load DE500BA/dec21x4x EWB bus=pci_1 device=7 function=1
load DE500BA/dec21x4x EWC bus=pci_1 device=7 function=2
load DE500BA/dec21x4x EWD bus=pci_1 device=7 function=3
```

Example 3: Loading mixture of KZPBA and DE500BA into slot 1, populating 2 functions out of 4

```
load KZPBA PKC bus=pci_0 device=7 function=0
load DE500BA/dec21x4x EWA bus=pci_0 device=7 function=1
```

ℹ In the above example device name is PKC as there are two built-in PK-like PCI SCSI Adapters located “closer” to CPU and therefore assigned names PKA and PKB.

AlphaServer DS25 (6 PCI slots)

In addition to 6 PCI vacant slots there are 7 PCI positions occupied by on-board devices. All 13 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	pci_<N>	device	function	irq	Description	Preloaded Name
PCI0 (bus=pci_0)						
-	0	7	0	-	ALi M1543C PCI ISA bridge	
-	0	8	0	-	Intel i82559 PCI Ethernet Adapter	EIA (EWA)
1	0	9	0	24	<option>	
			1	25	<option>, function 1	
			2	26	<option>, function 2	
			3	27	<option>, function 3	
2	0	10	0	12	<option>	
			1	13	<option>, function 1	
			2	14	<option>, function 2	
			3	15	<option>, function 3	
-	0	16	0	-	ALi M1543C PCI IDE/ATAPI controller	DQA, DQB
-	0	19	1	-	ALi M1543C PCI USB adapter	
PCI1 (bus=pci_1)						
3	1	1	0	28	<option>	
			1	29	<option>, function 1	
			2	30	<option>, function 2	
			3	31	<option>, function 3	
4	1	2	0	32	<option>	
			1	33	<option>, function 1	
			2	34	<option>, function 2	
			3	35	<option>, function 3	
PCI2 (bus=pci_2)						
-	2	1	0	-	Adaptec AIC-7899 (channel 0)	PKA
			1	-	Adaptec AIC-7899 (channel 1)	PKB
-	2	5	0	-	BroadCom BCM5703 PCI Ethernet Adapter	EIB (EWB)
PCI3 (bus=pci_3)						
5	3	1	0	36	<option>	
			1	37	<option>, function 1	
			2	38	<option>, function 2	
			3	39	<option>, function 3	
6	3	2	0	40	<option>	
			1	41	<option>, function 1	
			2	42	<option>, function 2	
			3	43	<option>, function 3	

The IRQ stands for bit position in DRIR of TITAN Chip. It has nothing to do with “ISA” style interrupts which are routed to IRQ 55 (including ALi M1543C PCI IDE/ATAPI controller).

So far the CHARON-AXP VMs do not emulate Intel i82559. Instead, emulation of DECchip 21143 is used.

So far the CHARON-AXP VMs do not emulate ALi M1543C PCI USB adapter. So position of the device 19, function 0 on the PCI 0 remains empty.

Unless SCSI option is plugged into PCI slot 1, 2, 3, or 4, the onboard SCSI controllers appear as PKA (pka7.0.0.1.2) and PKB (pkb7.0.0.101.2) respectively.

So far the CHARON-AXP VMs do not emulate Adaptec AIC-7899. Instead, emulation of QLOGIC ISP1040B is used.

So far the CHARON-AXP VMs do not emulate BroadCom BCM5703. Instead, emulation of DECchip 21143 is used.

Example 1: Loading DE500BA into slot 5

```
load DE500BA/dec21x4x EWC bus=pci_3 device=1 function=0
```

Example 2: Loading multiple DE500BA’s into slot 5, populating all 4 functions (gives 4 Ethernet ports)

```
load DE500BA/dec21x4x EWC bus=pci_3 device=1 function=0
load DE500BA/dec21x4x EWD bus=pci_3 device=1 function=1
load DE500BA/dec21x4x EWE bus=pci_3 device=1 function=2
load DE500BA/dec21x4x EWF bus=pci_3 device=1 function=3
```

In the above examples device name is EWC as there are built-in EW-like PCI Ethernet Adapters located “closer” to CPU and therefore assigned names EWA and EWB.

Example 3: Loading mixture of KZPBA and DE500BA into slot 6, populating 2 functions out of 4

```
load KZPBA PKC bus=pci_3 device=2 function=0
load DE500BA/dec21x4x EWC bus=pci_3 device=2 function=1
```

In the above example device name is PKB as there is a built-in PK-like PCI SCSI Adapter located “closer” to CPU and therefore assigned name PKA, as there are two built-in EW-like PCI Ethernet Adapters located “closer” to CPU and therefore assigned names EWA and EWB

AlphaServer ES40 (10 PCI slots)

In addition to 10 PCI vacant slots there are 3 PCI positions occupied by on-board devices. All 13 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	pci_<N>	device	function	irq	Description	Preloaded Name
PCI1 (bus=pci_1)						
5	1	1	0	24	<option>	
			1	25	<option>, function 1	
			2	26	<option>, function 2	
			3	27	<option>, function 3	
6	1	2	0	28	<option>	
			1	29	<option>, function 1	
			2	30	<option>, function 2	
			3	31	<option>, function 3	
7	1	3	0	32	<option>	
			1	33	<option>, function 1	
			2	34	<option>, function 2	
			3	35	<option>, function 3	
8	1	4	0	36	<option>	
			1	37	<option>, function 1	
			2	38	<option>, function 2	
			3	39	<option>, function 3	
9	1	5	0	40	<option>	
			1	41	<option>, function 1	
			2	42	<option>, function 2	
			3	43	<option>, function 3	
10	1	6	0	44	<option>	
			1	45	<option>, function 1	
			2	46	<option>, function 2	
			3	47	<option>, function 3	
PCI0 (bus=pci_0)						
1	0	1	0	8	<option>	
			1	9	<option>, function 1	
			2	10	<option>, function 2	
			3	11	<option>, function 3	
2	0	2	0	12	<option>	
			1	13	<option>, function 1	
			2	14	<option>, function 2	
			3	15	<option>, function 3	
3	0	3	1	16	<option>	
			1	17	<option>, function 1	

			2	18	<option>, function 2	
			3	19	<option>, function 3	
4	0	4	0	20	<option>	
			1	21	<option>, function 1	
			2	22	<option>, function 2	
			3	23	<option>, function 3	
-	0	5	0	-	ALi M1543C PCI ISA bridge	
-	0	15	0	-	ALi M1543C PCI ISA bridge	DQA, DQB
-	0	19	0	-	ALi M1543C PCI USB adapter	

The IRQ stands for bit position in DRIR of Tsunami/Typhoon chip. It has nothing to do with “ISA” style interrupts which are routed to IRQ 55 (including ALi M1543C PCI IDE/ATAPI controller).

So far the CHARON-AXP VMs do not support virtual ALi M1543C PCI USB adapter. So position of the device 19, function 0 on the PCI 0 remains empty.

Example 1: Loading DE500BA into slot 5

```
load DE500BA/dec21x4x EWA bus=pci_1 device=1 function=0
```

Example 2: Loading multiple DE500BA’s into slot 5, populating all 4 functions (gives 4 Ethernet ports)

```
load DE500BA/dec21x4x EWA bus=pci_1 device=1 function=0
load DE500BA/dec21x4x EWB bus=pci_1 device=1 function=1
load DE500BA/dec21x4x EWC bus=pci_1 device=1 function=2
load DE500BA/dec21x4x EWD bus=pci_1 device=1 function=3
```

Example 3: Loading mixture of KZPBA and DE500BA into slot 1, populating 2 functions out of 4

```
load KZPBA PKA bus=pci_0 device=1 function=0
load DE500BA/dec21x4x EWA bus=pci_0 device=1 function=1
```

AlphaServer ES45 (10 PCI slots)

In addition to 10 PCI vacant slots there are 3 PCI positions occupied by on-board devices. All 13 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	pci_<N>	device	function	irq	Description	Preloaded Name
PCI0 (bus=pci_0)						
-	0	7	0	-	ALi M1543C PCI ISA bridge	
1	0	8	0	20	<option>	
			1	21	<option>, function 1	
			2	22	<option>, function 2	
			3	23	<option>, function 3	
2	0	9	0	24	<option>	
			1	25	<option>, function 1	
			2	26	<option>, function 2	
			3	27	<option>, function 3	
3	0	10	0	12	<option>	
			1	13	<option>, function 1	
			2	14	<option>, function 2	
			3	15	<option>, function 3	
4	0	11	0	16	<option>	
			1	17	<option>, function 1	
			2	18	<option>, function 2	
			3	19	<option>, function 3	
-	0	16	0	-	ALi M1543C PCI IDE/ATAPI controller	DQA, DQB
-	0	19	0	-	ALi M1543C PCI USB adapter	
PCI1 (bus=pci_1)						
5	1	1	0	28	<option>	
			1	29	<option>, function 1	
			2	30	<option>, function 2	
			3	31	<option>, function 3	
6	1	2	0	32	<option>	
			1	33	<option>, function 1	
			2	34	<option>, function 2	
			3	35	<option>, function 3	
PCI2 (bus=pci_2)						
7	2	1	0	0	<option>	
			1	1	<option>, function 1	
			2	2	<option>, function 2	
			3	3	<option>, function 3	
8	2	2	0	4	<option>	
			1	5	<option>, function 1	

			2	6	<option>, function 2	
			3	7	<option>, function 3	
PCI3 (bus=pci_3)						
9	3	1	0	36	<option>	
			1	37	<option>, function 1	
			2	38	<option>, function 2	
			3	39	<option>, function 3	
10	3	2	0	40	<option>	
			1	41	<option>, function 1	
			2	42	<option>, function 2	
			3	43	<option>, function 3	

The IRQ stands for bit position in DRIR of TITAN chip. It has nothing to do with “ISA” style interrupts which are routed to IRQ 55 (including ALi M1543C PCI IDE/ATAPI controller).

So far the CHARON-AXP VMs do not support virtual ALi M1543C PCI USB adapter. So position of the device 19, function 0 on the PCI 0 remains empty.

Example 1: Loading DE500BA into slot 5

```
load DE500BA/dec21x4x EWA bus=pci_1 device=1 function=0
```

Example 2: Loading multiple DE500BA’s into slot 5, populating all 4 functions (gives 4 Ethernet ports)

```
load DE500BA/dec21x4x EWA bus=pci_1 device=1 function=0
load DE500BA/dec21x4x EWB bus=pci_1 device=1 function=1
load DE500BA/dec21x4x EWC bus=pci_1 device=1 function=2
load DE500BA/dec21x4x EWD bus=pci_1 device=1 function=3
```

Example 3: Loading mixture of KZPBA and DE500BA into slot 1, populating 2 functions out of 4

```
load KZPBA PKA bus=pci_0 device=8 function=0
load DE500BA/dec21x4x EWA bus=pci_0 device=8 function=1
```

AlphaServer GS80 (8 PCI busses)

Slot	pci_<N>	device	function	irq	Description	Preloaded Name
QBB0.PCA0.PCI0 (bus=qbb_0_pca_0_pci_0)						
0/1	0	1	0	36	QLOGIC ISP1040B PCI SCSI Adapter	PKA
2	0	2	0	40	<option>	
3	0	3	0	44	<option>	
-	0	7	0	-	ALi M1543C PCI ISA bridge	
-	0	15	0	-	ALi M1543C PCI IDE/ATAPI controller	DQA
-	0	19	0	-	ALi M1543C PCI USB adapter	
QBB0.PCA0.PCI1 (bus=qbb_0_pca_0_pci_1)						
4	1	4	0	48	<option>	
5	1	5	0	52	<option>	
6	1	6	0	56	<option>	
7	1	7	0	60	<option>	
QBB0.PCA1.PCI0 (bus=qbb_0_pca_1_pci_0)						
0/1	2	0	0	32	<option>	
2	2	2	0	40	<option>	

3	2	3	0	44	<option>	
QBB0.PCA1.PCI1 (bus=qbb_0_pca_1_pci_1)						
4	3	4	0	48	<option>	
5	3	5	0	52	<option>	
6	3	6	0	56	<option>	
7	3	7	0	60	<option>	
QBB1.PCA0.PCI0 (bus=qbb_1_pca_0_pci_0)						
0/1	8	0	0	32	<option>	
2	8	2	0	40	<option>	
3	8	3	0	44	<option>	
QBB1.PCA0.PCI1 (bus=qbb_1_pca_0_pci_1)						
4	9	4	0	48	<option>	
5	9	5	0	52	<option>	
6	9	6	0	56	<option>	
7	9	7	0	60	<option>	
QBB1.PCA1.PCI0 (bus=qbb_1_pca_1_pci_0)						
0/1	10	0	0	32	<option>	
2	10	2	0	40	<option>	
3	10	3	0	44	<option>	
QBB1.PCA1.PCI1 (bus=qbb_1_pca_1_pci_1)						
4	11	4	0	48	<option>	
5	11	5	0	52	<option>	
6	11	6	0	56	<option>	
7	11	7	0	60	<option>	

PCI 2 and 3 on each QBB are not populated.

So far the CHARON-AXP VMs do not support virtual ALi M1543C PCI USB adapter. So position of the device 19, function 0 on the PCI 0 remains empty.

Total number of PCI devices configured through CFG file may not exceed 20.

Example: Loading DE500BA into slot 2 of QBB0.PCA0

```
load DE500BA/dec21x4x EWA bus=qbb_0_pca_0_pci_0 device=2 function=0
```

AlphaServer GS160 (16 PCI busses)

Slot	pci_<N>	device	function	irq	Description	Preloaded Name
QBB0.PCA0.PCI0 (bus=qbb_0_pca_0_pci_0)						
0/1	0	1	0	36	QLOGIC ISP1040B PCI SCSI Adapter	PKA
2	0	2	0	40	<option>	
3	0	3	0	44	<option>	
-	0	7	0	-	ALi M1543C PCI ISA bridge	
-	0	15	0	-	ALi M1543C PCI IDE/ATAPI controller	DQA
-	0	19	0	-	ALi M1543C PCI USB adapter	
QBB0.PCA0.PCI1 (bus=qbb_0_pca_0_pci_1)						
4	1	4	0	48	<option>	
5	1	5	0	52	<option>	
6	1	6	0	56	<option>	
7	1	7	0	60	<option>	
QBB0.PCA1.PCI0 (bus=qbb_0_pca_1_pci_0)						
0/1	2	0	0	32	<option>	
2	2	2	0	40	<option>	
3	2	3	0	44	<option>	
QBB0.PCA1.PCI1 (bus=qbb_0_pca_1_pci_1)						
4	3	4	0	48	<option>	
5	3	5	0	52	<option>	
6	3	6	0	56	<option>	
7	3	7	0	60	<option>	
QBB1.PCA0.PCI0 (bus=qbb_1_pca_0_pci_0)						
0/1	8	0	0	32	<option>	
2	8	2	0	40	<option>	
3	8	3	0	44	<option>	
QBB1.PCA0.PCI1 (bus=qbb_1_pca_0_pci_1)						
4	9	4	0	48	<option>	
5	9	5	0	52	<option>	
6	9	6	0	56	<option>	
7	9	7	0	60	<option>	
QBB1.PCA1.PCI0 (bus=qbb_1_pca_1_pci_0)						
0/1	10	0	0	32	<option>	
2	10	2	0	40	<option>	
3	10	3	0	44	<option>	
QBB1.PCA1.PCI1 (bus=qbb_1_pca_1_pci_1)						
4	11	4	0	48	<option>	
5	11	5	0	52	<option>	
6	11	6	0	56	<option>	

7	11	7	0	60	<option>	
QBB2.PCA0.PCI0 (bus=qbb_2_pca_0_pci_0)						
0/1	16	0	0	32	<option>	
2	16	2	0	40	<option>	
3	16	3	0	44	<option>	
QBB2.PCA0.PCI1 (bus=qbb_2_pca_0_pci_1)						
4	17	4	0	48	<option>	
5	17	5	0	52	<option>	
6	17	6	0	56	<option>	
7	17	7	0	60	<option>	
QBB2.PCA1.PCI0 (bus=qbb_2_pca_1_pci_0)						
0/1	18	0	0	32	<option>	
2	18	2	0	40	<option>	
3	18	3	0	44	<option>	
QBB2.PCA1.PCI1 (bus=qbb_2_pca_1_pci_1)						
4	19	4	0	48	<option>	
5	19	5	0	52	<option>	
6	19	6	0	56	<option>	
7	19	7	0	60	<option>	
QBB3.PCA0.PCI0 (bus=qbb_3_pca_0_pci_0)						
0/1	24	0	0	32	<option>	
2	24	2	0	40	<option>	
3	24	3	0	44	<option>	
QBB3.PCA0.PCI1 (bus=qbb_3_pca_0_pci_1)						
4	25	4	0	48	<option>	
5	25	5	0	52	<option>	
6	25	6	0	56	<option>	
7	25	7	0	60	<option>	
QBB3.PCA1.PCI0 (bus=qbb_3_pca_1_pci_0)						
0/1	26	0	0	32	<option>	
2	26	2	0	40	<option>	
3	26	3	0	44	<option>	
QBB0.PCA0.PCI1 (bus=qbb_3_pca_1_pci_1)						
4	27	4	0	48	<option>	
5	27	5	0	52	<option>	
6	27	6	0	56	<option>	
7	27	7	0	60	<option>	

PCA 2 and 3 on each QBB are not populated in emulator.

So far the CHARON-AXP VMs do not emulate ALi M1543C PCI USB adapter. So position of the device 19, function 0 on the PCI 0 on QBB 0 remains empty.

Total number of PCI devices configured through CFG file may not exceed 20.

Example: Loading DE500BA into slot 2 of QBB0.PCA0

```
load DE500BA/dec21x4x EWA bus=qbb_0_pca_0_pci_0 device=2 function=0
```

AlphaServer GS320 (32 PCI busses)

Slot	pci_<N>	device	function	irq	Description	Preloaded Name
QBB0.PCA0.PCI0 (bus=qbb_0_pca_0_pci_0)						
0/1	0	1	0	36	QLOGIC ISP1040B PCI SCSI Adapter	PKA
2	0	2	0	40	<option>	
3	0	3	0	44	<option>	
-	0	7	0	-	ALi M1543C PCI ISA bridge	
-	0	15	0	-	ALi M1543C PCI IDE/ATAPI controller	DQA
-	0	19	0	-	ALi M1543C PCI USB adapter	
QBB0.PCA0.PCI1 (bus=qbb_0_pca_0_pci_1)						
4	1	4	0	48	<option>	
5	1	5	0	52	<option>	
6	1	6	0	56	<option>	
7	1	7	0	60	<option>	
QBB0.PCA1.PCI0 (bus=qbb_0_pca_1_pci_0)						
0/1	2	0	0	32	<option>	
2	2	2	0	40	<option>	
3	2	3	0	44	<option>	
QBB0.PCA1.PCI1 (bus=qbb_0_pca_1_pci_1)						
4	3	4	0	48	<option>	
5	3	5	0	52	<option>	
6	3	6	0	56	<option>	
7	3	7	0	60	<option>	
QBB1.PCA0.PCI0 (bus=qbb_1_pca_0_pci_0)						
0/1	8	0	0	32	<option>	
2	8	2	0	40	<option>	
3	8	3	0	44	<option>	
QBB1.PCA0.PCI1 (bus=qbb_1_pca_0_pci_1)						
4	9	4	0	48	<option>	
5	9	5	0	52	<option>	
6	9	6	0	56	<option>	
7	9	7	0	60	<option>	
QBB1.PCA1.PCI0 (bus=qbb_1_pca_1_pci_0)						
0/1	10	0	0	32	<option>	
2	10	2	0	40	<option>	
3	10	3	0	44	<option>	
QBB1.PCA1.PCI1 (bus=qbb_1_pca_1_pci_1)						

4	11	4	0	48	<option>	
5	11	5	0	52	<option>	
6	11	6	0	56	<option>	
7	11	7	0	60	<option>	
QBB2.PCA0.PCI0 (bus=qbb_2_pca_0_pci_0)						
0/1	16	0	0	32	<option>	
2	16	2	0	40	<option>	
3	16	3	0	44	<option>	
QBB2.PCA0.PCI1 (bus=qbb_2_pca_0_pci_1)						
4	17	4	0	48	<option>	
5	17	5	0	52	<option>	
6	17	6	0	56	<option>	
7	17	7	0	60	<option>	
QBB2.PCA1.PCI0 (bus=qbb_2_pca_1_pci_0)						
0/1	18	0	0	32	<option>	
2	18	2	0	40	<option>	
3	18	3	0	44	<option>	
QBB2.PCA1.PCI1 (bus=qbb_2_pca_1_pci_1)						
4	19	4	0	48	<option>	
5	19	5	0	52	<option>	
6	19	6	0	56	<option>	
7	19	7	0	60	<option>	
QBB3.PCA0.PCI0 (bus=qbb_3_pca_0_pci_0)						
0/1	24	0	0	32	<option>	
2	24	2	0	40	<option>	
3	24	3	0	44	<option>	
QBB3.PCA0.PCI1 (bus=qbb_3_pca_0_pci_1)						
4	25	4	0	48	<option>	
5	25	5	0	52	<option>	
6	25	6	0	56	<option>	
7	25	7	0	60	<option>	
QBB3.PCA1.PCI0 (bus=qbb_3_pca_1_pci_0)						
0/1	26	0	0	32	<option>	
2	26	2	0	40	<option>	
3	26	3	0	44	<option>	
QBB3.PCA1.PCI1 (bus=qbb_3_pca_1_pci_1)						
4	27	4	0	48	<option>	
5	27	5	0	52	<option>	
6	27	6	0	56	<option>	
7	27	7	0	60	<option>	
QBB4.PCA0.PCI0 (bus=qbb_4_pca_0_pci_0)						

0/1	32	0	0	32	<option>	
2	32	2	0	40	<option>	
3	32	3	0	44	<option>	
QBB4.PCA0.PCI1 (bus=qbb_4_pca_0_pci_1)						
4	33	4	0	48	<option>	
5	33	5	0	52	<option>	
6	33	6	0	56	<option>	
7	33	7	0	60	<option>	
QBB4.PCA1.PCI0 (bus=qbb_4_pca_1_pci_0)						
0/1	34	0	0	32	<option>	
2	34	2	0	40	<option>	
3	34	3	0	44	<option>	
QBB4.PCA1.PCI1 (bus=qbb_4_pca_1_pci_1)						
4	35	4	0	48	<option>	
5	35	5	0	52	<option>	
6	35	6	0	56	<option>	
7	35	7	0	60	<option>	
QBB5.PCA0.PCI0 (bus=qbb_5_pca_0_pci_0)						
0/1	40	0	0	32	<option>	
2	40	2	0	40	<option>	
3	40	3	0	44	<option>	
QBB5.PCA0.PCI1 (bus=qbb_5_pca_0_pci_1)						
4	41	4	0	48	<option>	
5	41	5	0	52	<option>	
6	41	6	0	56	<option>	
7	41	7	0	60	<option>	
QBB5.PCA1.PCI0 (bus=qbb_5_pca_1_pci_0)						
0/1	42	0	0	32	<option>	
2	42	2	0	40	<option>	
3	42	3	0	44	<option>	
QBB5.PCA1.PCI1 (bus=qbb_5_pca_1_pci_1)						
4	43	4	0	48	<option>	
5	43	5	0	52	<option>	
6	43	6	0	56	<option>	
7	43	7	0	60	<option>	
QBB6.PCA0.PCI0 (bus=qbb_6_pca_0_pci_0)						
0/1	48	0	0	32	<option>	
2	48	2	0	40	<option>	
3	48	3	0	44	<option>	
QBB6.PCA0.PCI1 (bus=qbb_6_pca_0_pci_1)						
4	49	4	0	48	<option>	

5	49	5	0	52	<option>	
6	49	6	0	56	<option>	
7	49	7	0	60	<option>	
QBB6.PCA1.PCI0 (bus=qbb_6_pca_1_pci_0)						
0/1	50	0	0	32	<option>	
2	50	2	0	40	<option>	
3	50	3	0	44	<option>	
QBB6.PCA1.PCI1 (bus=qbb_6_pca_1_pci_1)						
4	51	4	0	48	<option>	
5	51	5	0	52	<option>	
6	51	6	0	56	<option>	
7	51	7	0	60	<option>	
QBB7.PCA0.PCI0 (bus=qbb_7_pca_0_pci_0)						
0/1	56	0	0	32	<option>	
2	56	2	0	40	<option>	
3	56	3	0	44	<option>	
QBB7.PCA0.PCI1 (bus=qbb_7_pca_0_pci_1)						
4	57	4	0	48	<option>	
5	57	5	0	52	<option>	
6	57	6	0	56	<option>	
7	57	7	0	60	<option>	
QBB7.PCA1.PCI0 (bus=qbb_7_pca_1_pci_0)						
0/1	58	0	0	32	<option>	
2	58	2	0	40	<option>	
3	58	3	0	44	<option>	
QBB7.PCA1.PCI1 (bus=qbb_7_pca_1_pci_1)						
4	59	4	0	48	<option>	
5	59	5	0	52	<option>	
6	59	6	0	56	<option>	
7	59	7	0	60	<option>	

PCA 2 and 3 on each QBB are not populated in emulator.

So far the MSC/AXP emulators do not emulate ALi M1543C PCI USB adapter. So position of the device 19, function 0 on the PCI 0 on QBB 0 remains empty.

Total number of PCI devices configured through CFG file may not exceed 20.

Example: Loading DE500BA into slot 2 of QBB0.PCA0

```
load DE500BA/dec21x4x EWA bus=qbb_0_pca_0_pci_0 device=2 function=0
```

Disks and tapes

Contents

KZPBA PCI SCSI adapter

Table of Contents

General description

KZPBA is a PCI SCSI adapter based on the QLogic ISP1040 Fast Wide SCSI adapter chip for HP Alpha.

In CHARON-AXP environment it supports up to 120 disks and tapes.

For systems with more than 16 heavily used units it is recommended to configure several virtual KZPBA PCI SCSI adapters and distribute the heavily loaded units evenly between the adapters.

Loading KZPBA storage adapter

Syntax for loading KZPBA storage adapter:

```
load KZPBA <controller designation>
```

Example:

```
load KZPBA PKA
```

In **AlphaStation 400 configurations** use the following syntax for KZPBA storage adapter loading:

```
load KZPBA PKB irq_bus = isa
```

The adapter instance name ("PKA" in the example above) is used then for parametrization, for example:

```
set PKA container[602]="C:\My disks\vms_distribution.vdisk"
```

The numbers in the square brackets represent SCSI ID and LUN of the devices on the virtual KZPBA SCSI bus.

They have the following format: **XXYY**, where:

Parameter	Range	Description
XX	0..15	SCSI ID
YY	00..07	LUN

By default KZPBA adapter uses first available PCI slot. If instead some particular slot is needed, refer to this section for details of specific placement of PCI peripherals on CHARON-AXP PCI bus.

By default each loaded KZPBA SCSI PCI adapter has SCSI ID=7. This setting can be changed with "scsi_id" parameter, for example:

```
set PKA scsi_id=0
```

ℹ CHARON-AXP Alpha models may have one or two KZPBA adapters preloaded.

Configuration parameters

The KZPBA PCI SCSI adapter emulation has the following configuration parameters:

scsi_id

Parameter	scsi_id
Type	Numeric
Value	Specifies SCSI ID of KZPBA PCI SCSI Adapter in a range 0..7 By default the "scsi_id" configuration parameter is set to 7. **Example**: `set PKA scsi_id=6`

host, port

Parameter	host, port
Type	Text string

Value	These parameters are used in SCSI cluster configurations.
	• Specifies remote end-point (remote host name and, optionally, TCP/IP port on remote host) of SCSI connection between this KZPBA PCI SCSI adapter and remote KZPBA PCI SCSI adapter on some host.
	• Specifies local end-point (TCP/IP port on local host) of SCSI connection between this KZPBA PCI SCSI adapter and remote KZPBA PCI SCSI adapter on some host.
	By default the "host" and "port" configuration options are not specified.
	Syntax:
	`port[connection-number]=<local port> host[connection-number]="<host-name{:tcpip-port-no}>"`
	where: connection_number = remote_scsi_id * 100 + lun_id
	Example - 2 members cluster on the same Charon server - Node 1:
	`set PKA scsi_id=6`
	`set PKA port[700]=11067 host[700]="localhost:11076"`
	Example - 2 members cluster on the same Charon server - Node 2:
	`set PKA scsi_id=7`
	`set PKA port[600]=11076 host[600]="localhost:11067"`

container

Parameter	container[N] N is "XXYY" number, where XX = SCSI ID (0..15) and YY = LUN (00..07)
Type	Text string
Value	Possible values of the parameter are strings in one of the following forms: • **Physical disk** ◦ "\\.\PhysicalDrive<X>", where X is 0, 1, … ⚠ Be careful not to destroy all the information from the disk dedicated to CHARON-AXP by mistake. These disks must not be formatted by the host OS. **Example**: `set PKA container[0]="\\.\PhysicalDrive1"`

- **Physical disk by its WWID**
  - "\\.\PhysicalDrive(DevID=XXX-...-XXXX)"

    ⚠ Be careful not to destroy all the information from the disk dedicated to CHARON-AXP by mistake.

    These disks must not be formatted by the host OS.

    DevID addresses the target physical disk by its WWID (hexadecimal 128-bit identifier assigned to the disk drive by its manufacturer/originator).

    Example:
    ```
    set PKA container[100]="\\.\PhysicalDrive(DevID=6008-05F3-0005-2950-BF8E-0B86-A0C7-0001)"
    ```

    The WWID values can be obtained from "Host Device Check" utility ("All drives" section)

    If the "Host Device Check" utility does not display any WWID, this means the target disk does not have one. Use the "\\.\PhysicalDrive<N>" mapping in this case.
- **iSCSI disks**
  - "\\.\PhysicalDrive(iScsiTarget = <iSCSI target>, LUN = <LUN number>)"

    iScsiTarget addresses the disk by its iSCSI target name.

    LUN specifies LUN on connected iSCSI disk.

    Example:
    ```
    set PKA container[200]="\\.\PhysicalDrive(iScsiTarget=iqn.2008-04:iscsi.charon-target-test1, LUN= 1)"
    ```
- **Tape device**
  - "\\.\Tape<X>", where X is 0, 1, …
    Example:
    ```
    set PKA container[600]="\\.\Tape0"
    ```
- **Tape changer**
  - "\\.\Changer<X>", where X is 0, 1, …
- **Floppy drive**
  - "\\.\A:"
  - "\\.\B:"

    Example:
    ```
    set PKA container[300]="\\.\A:"
    ```
- **CDROM device**
  - "\\.\CdRom<X>", where X is 0, 1, …

    Example:
    ```
    set PKA container[400]="\\.\CdRom0"
    ```
- **ISO file for reading distribution CD-ROM image**
  - [<drive>":\"<path-name>"\"]<file-name>[".iso”]
    Mapping may also include the full path (recommended), for example: "C:\My disks\vms_distribution.iso"

    Example:
    ```
    set PKA container[600]="C:\My disks\vms_distribution.iso"
    ```
- **File representing a physical disk of the HP Alpha system (disk image)**
  - [<drive>":\"<path-name>"\"]<file-name>[".vdisk"]
    These files can be created from scratch with "MkDisk" utility. Data and OS disks backups are transferred from the original system via tapes or network and restored into these container files.
    Mapping may also include the full path (recommended), for example: "C:\My disks\my_boot_disk.vdisk"
    Example:
    ```
    set PKA container[401]="tru64-v51-system.vdisk"
    ```

    Important note: using compressed folders to store virtual disks and tapes is not supported.
- **File representing the tape (tape image)**

- [<drive>":\"<path-name>"\"]<file-name>".vtape"
  The container-file to which an emulated tape drive is mapped is created automatically when the emulator starts, and when the device is accessed from within the guest operating system.

  We recommend specifying the full path to the container-file, for example: "C:\MyTapes\backup.vtape"

  Example:
  `set PKA container[500]="E:\Tapes\backup.vtape"`

  If the "CHARON Guest Utilities for OpenVMS" (CHARONCP) package is used, the syntax is different. Please read the corresponding chapter.

  Important note: using compressed folders to store virtual disks and tapes is not supported.

How the Emulator Maps Guest-OS Operations to the Virtual Tape Drive

Guest-OS Operation	Emulator Action
Open device for writing	Create a container file if necessary; open for writing and lock the container file
Open device for reading	Create a container file if none exists. open for reading and lock container file
Unload (eject) tape from drive	Close a container file if open and unlock it - this allows copy/move/delete operations on CHARON host

The container file associated with a virtual tape drive can be compared to the tape cartridge used in a physical tape drive. Both store the data written to the tape device by the guest OS.

The size of virtual tape container files is limited only by space available in the emulator host file system.

CHARON does not support multi-volume backup for tape images. If some mutli-volume set (in form of tape images) has to be restored it is recommended to configure several tape drives in CHARONAXP configuration file, assign each tape image to each tape drive and use them in the following way (OpenVMS example):

```
$ BACKUP MKA100:BACKUP.BCK,MKA200,MKA300,MKA4000/SAVE_SET DKA0:...
```

⚠ Prerequisite to the examples below: a virtual tape device has been configured in the CHARON configuration file and it is not in use by the guest OS.

To perform backup:

1. The tape device may be issued the "unload" command and the container-file moved/deleted to insure proper status
2. Initialize the tape device using standard guest OS procedure.
3. Perform backup.
4. Issue "unload" command to the tape device in the guest OS.
5. On the emulator host, move the *.vtape container file containing backup data for storage or further backup.

To restore from a backup:

1. The tape device may be issued the "unload" command to insure proper status.
2. On the emulator host, move or copy a *.vtape container file containing backup data onto the filename specified in the CHARON configuration file.
3. Perform restore.
4. Issue the "unload" command to the tape device in the guest OS.
5. Delete or move the container file in preparation for the next vtape operation.

- **Other type of drive, for example magneto-optical drive**
  - "\\.\<N>:"
    Example:
    `set PKA container[300]="\\.\Z:"`

ℹ This parameter is initially not set, thus creating NO storage elements on the controller.

media_type

Parameter	media_type[N] N is "XXYY" number, where XX = SCSI ID (0..15) and YY = LUN (00..07)
Type	Text string

Value	Instructs CHARON-AXP to use the supplied value as the PRODUCT field in the SCSI INQUIRY data returned to a software running on virtual Alpha system in response to SCSI INQUIRY command. If not specified, CHARON-AXP attempts to guess the SCSI INQUIRY data based on virtual SCSI device type and underlying container (which is specified in the corresponding container configuration parameter). Initially is not specified. **Example**: `set PKA media_type[0]="HSZ70"`

removable

Parameter	removable[N] N is "XXYY" number, where XX = SCSI ID (0..15) and YY - LUN (00..07)
Type	Boolean
Value	When set to "true", the removable configuration parameter instructs CHARON-AXP to report the corresponding virtual SCSI device as removable. Note to set this parameter to "true" if this disk storage element is: • Mapped to a host removable device • Mapped to a disk/tape image located on a host removable device • Mapped to a disk/tape image that has to be renamed/moved/deleted right after dismounting it in the guest OS By default the removable configuration parameter is set to "false". **Example**: `set PKA removable[400]=true` ℹ Note that virtual SCSI tapes and CD-ROM devices are always reported as removable regardless of the "removable" configuration parameter.

geometry

Parameter	geometry [N] N is "XXYY" number, where XX = SCSI ID (0..15) and YY - LUN (00..07)
Type	Text String

Value	This formatted string value specifies the explicit geometry of the disk storage element. This parameter is not applicable to tape storage elements.
	The string format is <X>”/”<Y>[“/”<Z>][“/”<B>] or <X>”,”<Y>[“,”<Z>][“,”<B>] where:
	▪ **X** corresponds to the number of sectors per track
	▪ **Y** corresponds to the number of tracks per cylinder
	▪ **Z** corresponds to the number of cylinders on the unit. If omitted, Z is calculated based on X, Y and the total number of sectors on the unit that reflects the size of the disk storage element. This is an optional parameter.
	▪ **B** corresponds to the total size of the disk (in blocks) reported to the guest OS. If omitted it is calculated automatically. This is an optional parameter.
	If this parameter is not set, CHARON-AXP will configure the geometry based on the most probable disk type.
	Initially not set.
	It is possible to specify each parameter independently of another one. The following syntax is used for that:
	`set PKA geometry[300]="*,*,*,16777210"`
	The syntax described above is applicable only to disk storage elements. If the container is a **tape image**, the following format is used instead:
	`"<image-size>[, <early-warning-zone-size>]"`
	where:
	▪ `image-size` corresponds to the tape size in MB
	▪ `early-warning-zone-size` corresponds to the size (in KB) of the space left on the tape when a warning to the OS is issued. If omitted, 64K is assumed.
	Example:
	`set PKA geometry[603] = "255/255"`

use_io_file_buffering

Parameter	use_io_file_buffering[N] N is "XXYY" number, where XX = SCSI ID (0..15) and YY = LUN (00..07)
Type	Text String
Value	Instructs CHARON-AXP to enable host operating system I/O cache on reading/writing operations. The following values are possible: - "true" or "write-back" - the "write back" caching mode is used - "write-through" - the "write through" caching mode is used - "false" - caching is switched off ℹ Note that this caching has a significant effect only in case of mapping to disk and tape containers, not physical drives. When enabled, host operating system I/O cache may significantly improve I/O performance of the virtual system. At the same time maintaining I/O cache requires additional host resources (CPU and memory) which may negatively affect overall performance of the virtual system. Initially is set to "false". **Example**: `set PKA use_io_file_buffering[603]=true`

io_queue_depth

Parameter	io_queue_depth[N] N is "XXYY" number, where: - XX - SCSI ID (0..15) - YY - LUN (00..07)
Type	Numeric
Value	Specifies KZPBA I/O requests (read or write) for a given unit in a range 2..128 Setting this parameter enables KZPBA instance to run up-to the specifiied numbers of I/O requests (read or write) for unit N in parallel, thus improving the performance. The default value set by controller is optimal for most of the cases. It may be needed to enlarge this number if guest OS I/O queue for a certain unit contains too many pending entries. In this case the value should be equal to an average size of the queue, collected statistically. Please do not set this parameter without clear understanding the purpose. By default parallel execution of I/O requests is disabled. **Example**: `set PKA io_queue_depth[603]=4`

min_n_of_threads

Parameter	min_n_of_threads
Type	Numeric
Value	Instructs KZPBA I/O to reserve a given number of working threads in a range 1..64, thus improving the performance. All units of KZPBA instance share the I/O threads. The default value is equal to number of units plus 2. For optimization it is possible to set this parameter to sum of the "io_queue_depth" parameters for each unit plus 2. This assumption seems optimal for most of the cases. Please do not set this parameter without clear understanding the purpose. **Example**: `set PKA min_n_of_threads=16`

When a tape or disk image connected to an emulated KZPBA controller is dismounted by OpenVMS, it is disconnected from CHARON-AXP and can be manipulated. It can be replaced with a different disk image if it keeps the same name. This capability may be useful when designing back-up and restore procedures. When copying CHARON-AXP disk images while CHARON-AXP is running, please take care to minimize the risk of overloading a heavily loaded CHARON host system. For example, using a sequential series of simple ftp binary copies is less resource intensive and thus less disruptive than multiple, simultaneous copies.

Empty disk images are created with the "MkDisk" utility. Tape images ("*.vtape") will be created automatically if they don't exist (no utility needed).

CHARON-AXP is able to boot from disk images of any supported OpenVMS/Alpha and Tru64 version.

The virtual KZPBA storage controller examines the **file extension** `(vdisk or vtape)` to distinguish between a disk image and a tape image.

Configured physical devices or tape/disk images that do not exist on the host system will, in general, cause OpenVMS/Alpha to report the unit offline. In some cases this will result in a VMS BUG CHECK. In this case, an error message will be written to the log file.

KGPSA-CA PCI Fibre Channel adapter

Table of Contents

General description

CHARON-AXP supports emulation of DEC-KGPSA-CA PCI Fibre Channel adapter.

Every instance of KGPSA-CA works in one of the three following modes:

- KGPSA-CA PCI Fibre Channel adapter#Fabric virtualization mode (mapping to disk images and host/SAN physical disks). This is default mode.
- Usage of "presentation mode" of external storage controllers to automatically configure KGPSA-CA disks correspondent to the LUNs provided by the external storages
- KGPSA-CA PCI Fibre Channel adapter#Pass Through mode (using a specific CHARON PCI Pass Through driver to work through the EMULEX LightPulse PCI/PCI-X/PCIe FC adapter plugged into host)

Loading KGPSA storage adapter

Syntax for loading KGPSA-CA storage adapter:

```
load KGPSA <name>
```

Example:

```
load KGPSA FGA
```

⚠ In AlphaStation 400 configuration use the following syntax for KGPSA-CA storage adapter loading:

```
load KGPSA FGA irq_bus = isa
```

The adapter instance name ("FGA" in the example above) is used then for parametrization, for example:

```
set FGA container[100]="C:\My disks\vms_distributive.vdisk"
```

Numbers in the square brackets represent KGPSA-CA units. They can be in the range 0..32766, but no more than 255 units can be configured on a single controller.

By default KGPSA-CA adapter uses first available PCI slot. If instead some particular slot is needed, refer to this section for details of specific placement of PCI peripherals on CHARON Virtual Machine (VM) PCI bus.

Configuration parameters

The KGPSA-CA PCI FC adapter emulation has the following configuration parameters:

storage_controller_path_id

Parameter	storage_controller_path_id
Type	Text String

Value	Instructs CHARON-AXP to create a set of virtual FC devices and connect to the virtual system through the virtual KGPSA FC Adapter.
	ℹ In this mode WWID and UDID values are taken from the storage controller directly. Note that some storage controllers may not support aquiring UDID, in this case this way of mapping cannot be used in case of OpenVMS (Tru64 Unix is Ok)!
	Syntax:
	`load KGPSA <controller name> storage_controller_path_id = <Storage controller path ID>`
	where the "Storage controller path ID" parameter is a storage (for example SAN) controller path ID. This ID can be obtained from the "Host Device Check" utility ("Storage Controllers" section).
	Once specified, all the disks attached to the storage are automatically mapped as disk units to CHARON-AXP.
	Example:
	`load KGPSA FGA storage_controller_path_id = 5008-05F3-0005-2950-5008-05F3-0005-2951`
	It is also possible to use wildcards for specifying the ID number:
	Example:
	`load KGPSA FGA storage_controller_path_id = 5008-05F3-0005-2950-*-05F3-0005-*`
	In this case CHARON will collect the LUNs from all the matching paths.
	This syntax works disregarding whether Windows MPIO is active or not. If it is active, CHARON-AXP uses "active" path (because it works through Windows), if Windows MPIO is OFF, CHARON-AXP selects the first matching path (and warns you in the LOG file).
	ℹ Note that LUNs obtained by usage of "storage_controller_path_id" parameter is not re-enumerated on reboot, so this mode is possible to use in case of shared disks clusters.

host_bus_location

Parameter	host_bus_location
Type	Text String

Value	Establish connection between virtual DEC-KGPSA-CA PCI FC adapter and physical EMULEX LightPulse PCI/PCI-X/PCIe FC adapter (pass through mode)
	Syntax:
	`load KGPSA <controller name> host_bus_location="PCI bus X, device Y, function Z"`
	where X, Y, and Z describe location of physical EMULEX LightPulse PCI/PCI-X/PCIe FC adapter in the host computer (see KGPSA-CA PCI Fibre Channel adapter#Pass Through mode for details).
	Example:
	`load KGPSA FGA host_bus_location="PCI bus 3, device 1, function 0"`

wwid

Parameter	wwid[N] N is 0..32766 (no more than 255 units)
Type	Text String

Value	Sets WWID for emulated KGPSA adapter unit in case if the WWID can not be obtained directly from the device CHARON-AXP is mapped to.
	For example in case of disk images or local physical drives CHARON-AXP cannot get its WWID from the the host system, so it is automaticaly introduce some generated WWID. In case if this WWID is not acceptable it can be replaced with the custom one using the "wwid" parameter.
	Syntax:
	`set <controller name> wwid[unit-number]="XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX"`
	Example:
	`set FGA wwid[2]="6008-05F3-0005-2950-BF8E-0B86-A0C7-0001"`

container

Parameter	container[N] N is 0..32766 (no more than 255 units)
Type	Text String

Value	Possible values of the parameter are strings in one of the following forms:
	■ **Physical disk**
	■ "\\.\PhysicalDrive<X>", where X is 0, 1, …
	⚠ Be careful not to destroy all the information from the disk dedicated to CHARON-AXP by mistake.
	These disks must not be formatted by the host OS.
	ℹ In this mode WWID is generated by CHARON and UDID is defined by the index of the "container" parameter.
	Example:
	`set FGA container[0]=”\\.\PhysicalDrive1"`
	⚠ In case "\\.\PhysicalDrive<N>" points to some external iSCSI or FC disk note that the disks re-enumeration may appear on CHARON host reboot, for example an external disk originally referenced as "\\.\PhysicalDrive3" may become "\\.\PhysicalDrive1" etc. Especially it is important in shared disks cluster configurations!
	So it is stricty recommended to use "Mapping by WWID" / "Mapping iSCSI disks" ways of mapping in these situations (see below). The mapping by using "storage_controller_path_id" (see above) is acceptable as well, since it is not re-enumerating LUNs on each re-boot or CHARON-AXP restart.
	■ **Physical disk by its WWID**
	■ "\\.\PhysicalDrive(DevID =XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX)"
	⚠ Be careful not to destroy all the information from the disk dedicated to CHARON-AXP by mistake.
	These disks must not be formatted by the host OS.
	DevID addresses the target physical disk by its WWID (hexadecimal 128-bit identifier assigned to the disk drive by its manufacturer/originator).
	ℹ In this mode WWID is generated by CHARON-AXP and UDID is defined by the index of the "container" parameter.
	Example:
	`set FGA container[100]="\\.\PhysicalDrive(DevID= 6008-05F3-0005-2950-BF8E-0B86-A0C7-0001)"`
	The WWID values can be obtained from "Host Device Check" utility ("All drives" section).
	⚠ If the "Host Device Check" utility does not display any WWID, this means the target disk does not have one. Use the "\\.\PhysicalDrive<N>" mapping in this case.

- **iSCSI disks**
  - "\\.\PhysicalDrive(iSCSITarget = <iSCSI target>, LUN = <LUN number>)"
    iSCSITarget addresses the disk by its iSCSI target name. LUN specifies LUN on connected iSCSI disk.
    In this mode WWID is generated by CHARON-AXP and UDID is defined by the index of the "container" parameter.
    Example:

```
set FGA container[200]="\\.\PhysicalDrive(iSCSITarget= iqn.2008-04:iSCSI.charon-target-test1, LUN= 1)"
```

- **File representing a physical disk of the HP Alpha system (disk image)**
  - [<drive>":\"<path-name>"\"]<file-name>[".vdisk"]

These files can be created from scratch with "MkDisk" utility. Data and OS disks backups are transferred from the original system via tapes or network and restored into these container files.

Mapping may also include the full path (recommended), for example: "C:\My disks\my_boot_disk.vdisk"
Example:

```
set FGA container[401]="my_dka401.vdisk"
```

Using compressed folders to store virtual disks and tapes is not supported

This parameter is initially not set, thus creating NO storage elements on the controller.

media_type

Parameter	media_type[N] N is 0..32766 (no more than 255 units)
Type	Text String
Value	Instructs CHARON-AXP to use the supplied value as the PRODUCT field in the FC INQUIRY data returned to a software running on virtual HP Alpha system in response to FC INQUIRY command. If not specified, CHARON-AXP attempts to guess the FC INQUIRY data based on virtual FC device type and underlying container (which is specified in the corresponding container configuration parameter). Initially is not specified. **Example**: `set FGA media_type[0]="HSZ70"`

removable

Parameter	removable[N] N is 0..32766 (no more than 255 units)
Type	Boolean
Value	When set to "true", the removable configuration parameter instructs CHARON-AXP to report the corresponding virtual FC device as removable. By default the removable configuration parameter is set to "false". **Example**: `set FGA removable[400]=true`

geometry

Parameter	geometry [N] N is 0..32766 (no more than 255 units)
Type	Text String

Value	This formatted string value specifies the explicit geometry of the disk storage element.
	The string format is <X>”/”<Y>[“/”<Z>][“/”<B>] or <X>”,”<Y>[“,”<Z>][“,”<B>] where:
	■ **X** corresponds to the number of sectors per track
	■ **Y** corresponds to the number of tracks per cylinder
	■ **Z** corresponds to the number of cylinders on the unit. If omitted, Z is calculated based on X, Y and the total number of sectors on the unit that reflects the size of the disk storage element. This is an optional parameter.
	■ **B** corresponds to the total size of the disk (in blocks) reported to the guest OS. If omitted it is calculated automatically. This is an optional parameter.
	If this parameter is not set, CHARON-AXP will configure the geometry based on the most probable disk type.
	Initially not set.
	Example:
	`set FGA geometry[201] = "255/255"`
	ⓘ It is possible to specify each parameter independently of another one. The following syntax is used for that:
	`set FGA geometry[300]="*,*,*,16777210"`

use_io_file_buffering

Parameter	use_io_file_buffering[N] N is 0..32766 (no more than 255 units)
Type	Text String
Value	Instructs CHARON-AXP to enable host operating system I/O cache on reading/writing operations. The following values are possible: • "true" or "write-back" - the "write back" caching mode is used • "write-through" - the "write through" caching mode is used • "false" - caching is switched off ⓘ Note this caching has a significant effect only in case of mapping to disk containers, not physical drives. When enabled, host operating system I/O cache may significantly improve I/O performance of the virtual system. At the same time maintaining I/O cache requires additional host resources (CPU and memory) which may negatively affect overall performance of the virtual system. Initially is set to "false". **Example**: `set FGA use_io_file_buffering[300]=true`

io_queue_depth

Parameter	io_queue_depth[N] N is "XXYY" number, where: • XX - SCSI ID (0..15) • YY - LUN (00..07)
Type	Numeric
Value	Specifies KZPBA I/O requests (read or write) for a given unit in a range 2..128 Setting this parameter enables KZPBA instance to run up-to the specified numbers of I/O requests (read or write) for unit N in parallel, thus improving the performance. The default value set by controller is optimal for most of the cases. It may be needed to enlarge this number if guest OS I/O queue for a certain unit contains too much pending entries. In this case the value should be equal to an average size of the queue, collected statistically. Please do not set this parameter without clear understanding the purpose. By default parallel execution of I/O requests is disabled. **Example**: `set PKA io_queue_depth[603]=4`

min_n_of_threads

Parameter	min_n_of_threads
Type	Numeric
Value	Instructs KZPBA I/O to reserve a given number of working threads in a range 1..64, thus improving the performance. All units of KZPBA instance share the I/O threads. The default value is equal to number of units plus 2. For optimization it is possible to set this parameter to sum of the "io_queue_depth" parameters for each unit plus 2. This assumption seems optimal for most of the cases. Please do not set this parameter without clear understanding the purpose. **Example**: `set PKA min_n_of_threads=16`

When a disk image connected to an emulated KGPSA-CA controller is dismounted by OpenVMS, it is disconnected from CHARON-AXP and can be manipulated. It can be replaced with a different disk image if it keeps the same name. This capability may be useful when designing back-up and restore procedures. When copying CHARON disk images while CHARON-AXP is running, please take care to minimize the risk of overloading a heavily loaded CHARON host system. For example, using a sequential series of simple ftp binary copies is less resource intensive and thus less disruptive than multiple, simultaneous copies.

Empty disk images are created with the "MkDisk" utility.

CHARON-AXP is able to boot from disk images of any OpenVMS/Alpha and Tru64 version.

Mapping to host resources

There are 3 modes of KGPSA-CA mapping to system resources:

Mode	Description	Pros	Cons
KGPSA-CA PCI Fibre Channel adapter#Fabric virtualization mode	Virtual KGPSA-CA PCI FC adapter is mapped to physical disks (both local, FC and iSCSI) and disk images ⓘ This is default mode, suitable for most cases	• High I/O performance using modern local SSD disks.	• Requires CHARON-AXP reconfiguration and restart on any change to the mapped disk images and physical drives • Disk images cannot be used in case of Tru64 cluster, but acceptable for OpenVMS cluster.
"Presentation" mode	CHARON-AXP automatically creates /removes a set of virtual FC devices on a virtual KGPSA-CA FC adapter correspondent to available external SAN LUNs ⓘ This mode is recommended in case of working with SAN that may change LUN configuration - so CHARON-AXP will reflect it on-fly w/o reconfiguration and restarting.	• No reconfiguration and restart of CHARON-AXP is needed on change to the virtual FC drives provided as LUNs by the connected SAN. • Effectively any SAN FC system could be used, that means newer, faster, larger, FC HBA controllers and device drivers supported by newer, faster host server platforms and operating systems - it offers significant performance capabilities and performance improvements.	• Target configurations are restricted to the SAN models capable to work with HW Alphas (for example aquiring UDID must be supported in case of OpenVMS). This restriction comes from the fact that different SAN controllers could have different level of SCSI-3 protocol support, and if apparently some individual SAN model does not properly support complete set of SCSI-3 commands used by OpenVMS & Tru64 KGPSA-CA drivers, then with high probability such SAN Controller will not be able to work with CHARON-AXP in "presentation" (as well as in Pass Through) Mode.
KGPSA-CA PCI Fibre Channel adapter#Pass Through mode	This mode allows connection between virtual KGPSA-CA FC adapter and physical EMULEX LightPulse PCI /PCI-X/PCIe FC adapter plugged into host's PCI /PCI-X/PCIe slot ⓘ This mode is recommended for cluster setups that include HW Alphas	• High performance • Support of shared disks and cluster configuration that includes HW Alphas	• Cannot be used in VMware environment • Obsolete PCI-X cards are not available anymore, HBA list is restricted to several EMULEX HBA models only. No way to use HBA from other vendors. • Target configurations are restricted to SAN models capable to work with HW Alphas. • As EMULEX LightPulse PCI/PCI-X/PCIe FC adapters have been never used on Alphas, formally there is a probability that some compatibility problems may be detected.

⚠ All the described modes can be used in case of a cluster including HW Alphas and shared SAN, but pay attention that all the disks must look absolutely the same to the operating systems on all the cluster nodes, including CHARON: same WWID, same UDID (in case of OpenVMS), same size, same geometry etc.

Fabric virtualization mode

In this mode KGPSA-CA PCI FC adapter can be directly mapped to physical disks (both local and iSCSI) and disk images as shown in the following example:

```
set FGA container[0]="C:\My disks\my_dka401.vdisk"
set FGA container[100]="\\.\PhysicalDrive1"
set FGA container[200]="\\.\PhysicalDrive(DevID= 6008-05F3-0005-2950-BF8E-0B86-A0C7-0001)"
set FGA container[300]="\\.\PhysicalDrive(iSCSITarget= iqn.2008-04:iSCSI.charon-target-test1, LUN= 1)"
```

⚠ Use only "Mapping by WWID" and "Mapping iSCSI disks" syntax in case of the mapped physical disk located on some external disk controller (SAN, etc) since the "\\.\PhysicalDrive<N>" re-enumeration may happen on CHARON host reboot.

ℹ Note that WWID and UDID is generated by CHARON-AXP in its mode. It may be important in case of shared disks cluster configurations.

See the "KGPSA-CA PCI Fibre Channel adapter#Containers" section for details.

Usage of "presentation mode"

CHARON-AXP can utilize so called "presentation mode" of the storage controller it maps to.

In this mode CHARON-AXP creates a set of virtual FC devices and connects to the SAN system through the virtual KGPSA FC adapter.

The main benefit of this mode is a flexible way of CHARON-AXP virtual disks management depending on the mapped storage controller configuration. LUN parameters of any available to CHARON host storage controller specified in CHARON-AXP configuration file via "storage_controller_path_id" parameter are automatically scanned and then assigned as ‘virtual’ FC drives which guest OS can see, no need to manually change/redefine/modify virtual LUN parameters in emulator.

For example if an extra disk has been added to the storage controller it will automatically appear as a new disk unit on the corresponding virtual KGPSA-CA virtual controller mapped to that storage controller.

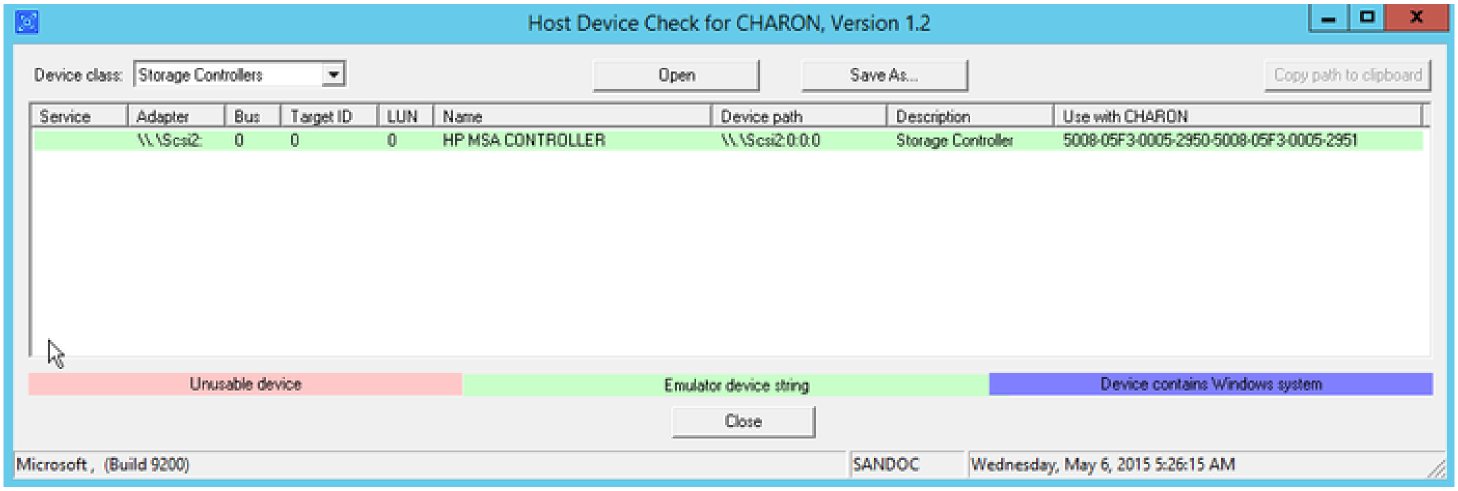
In "presentation mode" CHARON-AXP supports the following operations with FC disks (similar to ones supported in KGPSA FC Pass Through mode) w/o need of CHARON-AXP restart:

- Dynamically allocation (retranslation) of new created SAN LUN to virtual Alpha configuration
- Scan and translation of SAN LUN parameters to virtual Alpha’s KGPSA DGxx: virtual drive parameters
- Propagation of SAN LUN size change to virtual Alpha’s KGPSA virtual drive parameters
- Automatically deallocation from virtual Alpha deleted SAN LUN

Example of configuration:

```
load KGPSA FGA storage_controller_path_id = 5008-05F3-0005-2950-5008-05F3-0005-2951
```

The ID number specified in the example above is a storage (for example SAN) controller path ID. This ID can be obtained from the "Host Device Check" utility ("Storage Controllers" section):



Host Device Check for CHARON, Version 1.2

Device class: Storage Controllers

Service	Adapter	Bus	Target ID	LUN	Name	Device path	Description	Use with CHARON
	\\.\Scsi2:	0	0	0	HP MSA CONTROLLER	\\.\Scsi2:0:0:0	Storage Controller	5008-05F3-0005-2950-5008-05F3-0005-2951

🟥 Unusable device 🟩 Emulator device string 🟦 Device contains Windows system

Microsoft , (Build 9200) | SANDOC | Wednesday, May 6, 2015 5:26:15 AM

Once specified, all the disks attached to the storage are automatically mapped as disk units to CHARON-AXP.

It is also possible to use wildcards for specifying the ID number:

```
load KGPSA FGA storage_controller_path_id = 5008-05F3-0005-2950-*-05F3-0005-*
```

In this case CHARON-AXP will collect the LUNs from all the matching paths.

This syntax works disregarding whether Windows MPIO is active or not. If it is active, CHARON-AXP uses "active" path (because it works through Windows), if Windows MPIO is OFF, CHARON-AXP selects the first matching path (and warns you in the LOG file).

ⓘ Usage of LUNs in cluster configuration:

Note that PATHs (on OpenVMS level) do not affect OpenVMS cluster. What really makes sense is LUN size, geometry, WWI, etc. CHARON-AXP obtains these properties from the LUN (within the limits established by Windows), so LUNs should appear similar to all cluster members.

ⓘ Note that LUNs obtained by usage of "storage_controller_path_id" parameter is not re-enumerated on reboot, so this mode is possible to use in case of shared disks clusters.

ⓘ Note that the WWID and UDID are automatically obtained from disk server (SAN) in this mode - it is very convenient in case of shared disks cluster configurations. Also note that some storage controllers may not support aquiring UDID - in this case this way of mapping cannot be used in case of OpenVMS (Tru64 Unix is Ok)!

See the "Presentation mode" section for details.

Pass Through mode

The CHARON PCI Pass Through mode allows connection between virtual DEC-KGPSA-CA PCI FC adapter and physical EMULEX LightPulse PCI/PCI-X /PCIe FC adapter plugged into host’s PCI/PCI-X/PCIe slot.

Syntax:

```
load <controller name> host_bus_location="PCI bus X, device Y, function Z"
```

Example:

```
load KGPSA FGA host_bus_location="PCI bus 3, device 1, function 0"
```

The "host_bus_location" parameter addresses the host EMULEX LightPulse PCI/PCI-X/PCIe FC adapter in the following way:

Parameters	Description
"PCI bus X"	PCI bus number of the host EMULEX LightPulse PCI/PCI-X/PCIe FC adapter
"device Y"	PCI bus device number of the host EMULEX LightPulse PCI/PCI-X/PCIe FC adapter
"function Z"	The "function" parameter of the the host EMULEX LightPulse PCI/PCI-X/PCIe FC adapter

The following is a list of EMULEX LightPulse PCI/PCI-X/PCIe FC adapters supported by CHARON PCI Pass Through driver and suitable for emulation of KGPSA-CA PCI FC adapter in CHARON PCI Pass Through mode:

Supported	Not Supported	Not tested
LP8000 LP9000 LP9002 LP9802 LP10000 LP10000DC LP10000-S LPX1000 LP11002 LPe11002 (FC2242SR, A8003A) LPe1105 LPe12002 (AJ762B)	LPe1150 (FC2142SR, A8002A)	LPe11000

Also see the "host_bus_location" section for details

Installation of EMULEX LightPulse PCI/PCI-X/PCIe FC adapter driver

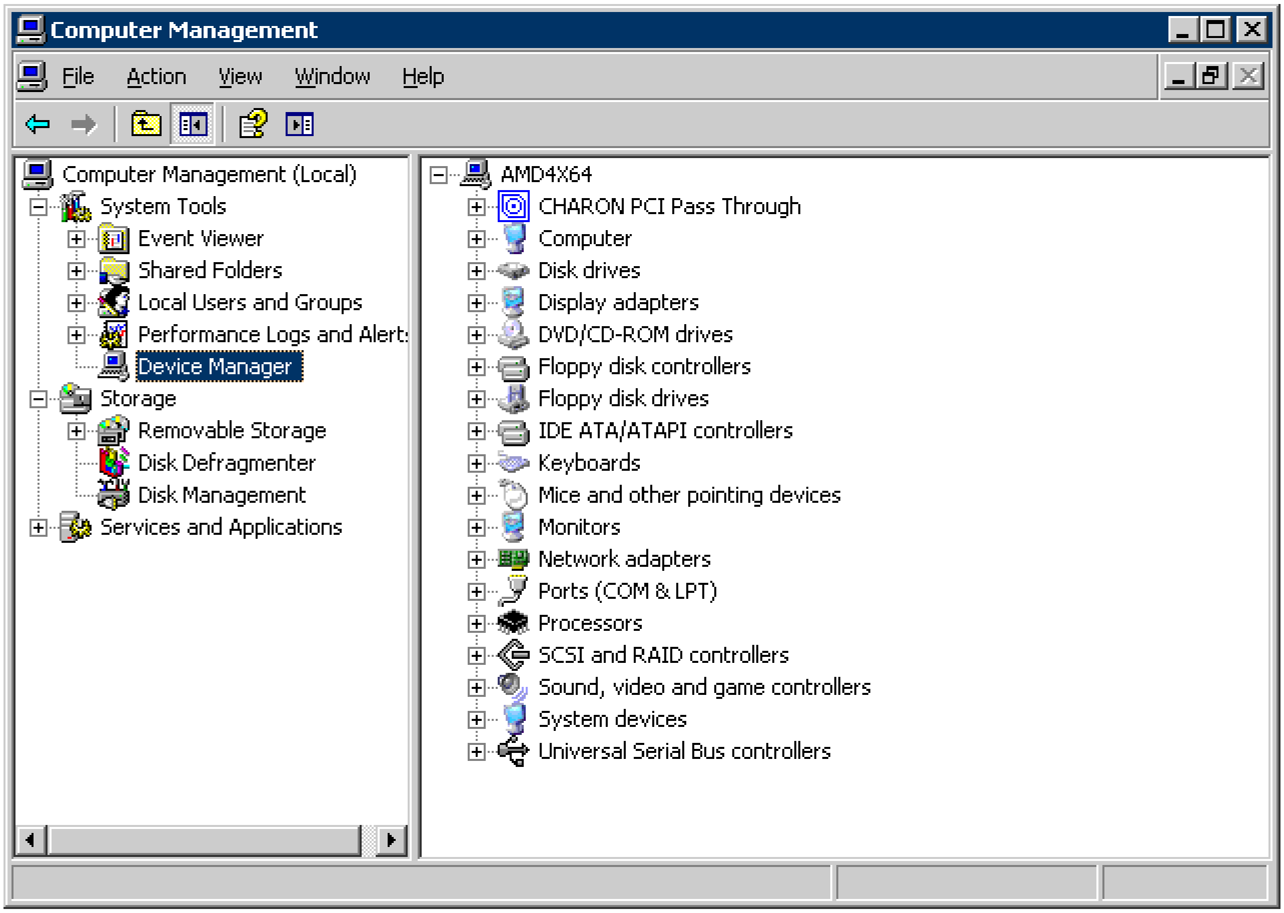
1. Install the EMULEX LightPulse PCI/PCI-X/PCIe FC adapter (see below for a list of supported models) to some spare PCI/PCI-X/PCIe slot of the host system
2. Boot a Windows operating system
3. Install the EMULEX LightPulse PCI/PCI-X/PCIe FC adapter driver from the following directory "C:\Program Files\CHARON\Drivers\EMULEX_X.X.0.XXXXX" by choosing the "Install from a list or specific location (Advanced)" option and then selecting the "emulex_lp_ppt_amd64.inf" file.
4. Direct Windows to use this custom driver instead of the default one:
   a. Type "gpedit.msc" in the "Search programs and files" field under "Start" menu, press "Enter"
   b. In the appearing applet choose: Administrative Templates -> System -> Device Installation -> Device Installation Restrictions -> Prevent Installation of Devices not described by other policy settings
   c. Change the default "Not configured" to "Disabled"; press "Ok" to apply,
5. Reboot the host

Collecting the parameters for mapping

There are 2 ways of getting the required parametrs for mapping virtual KGPSA-CA to EMULEX LightPulse PCI/PCI-X/PCIe FC adapter plugged into host's PCI/PCI-X/PCIe slot:

1. Usage of "HOST Device Check" utility. Follow the link for details.
2. Collecting the parameters directly using "Device Manager" applet

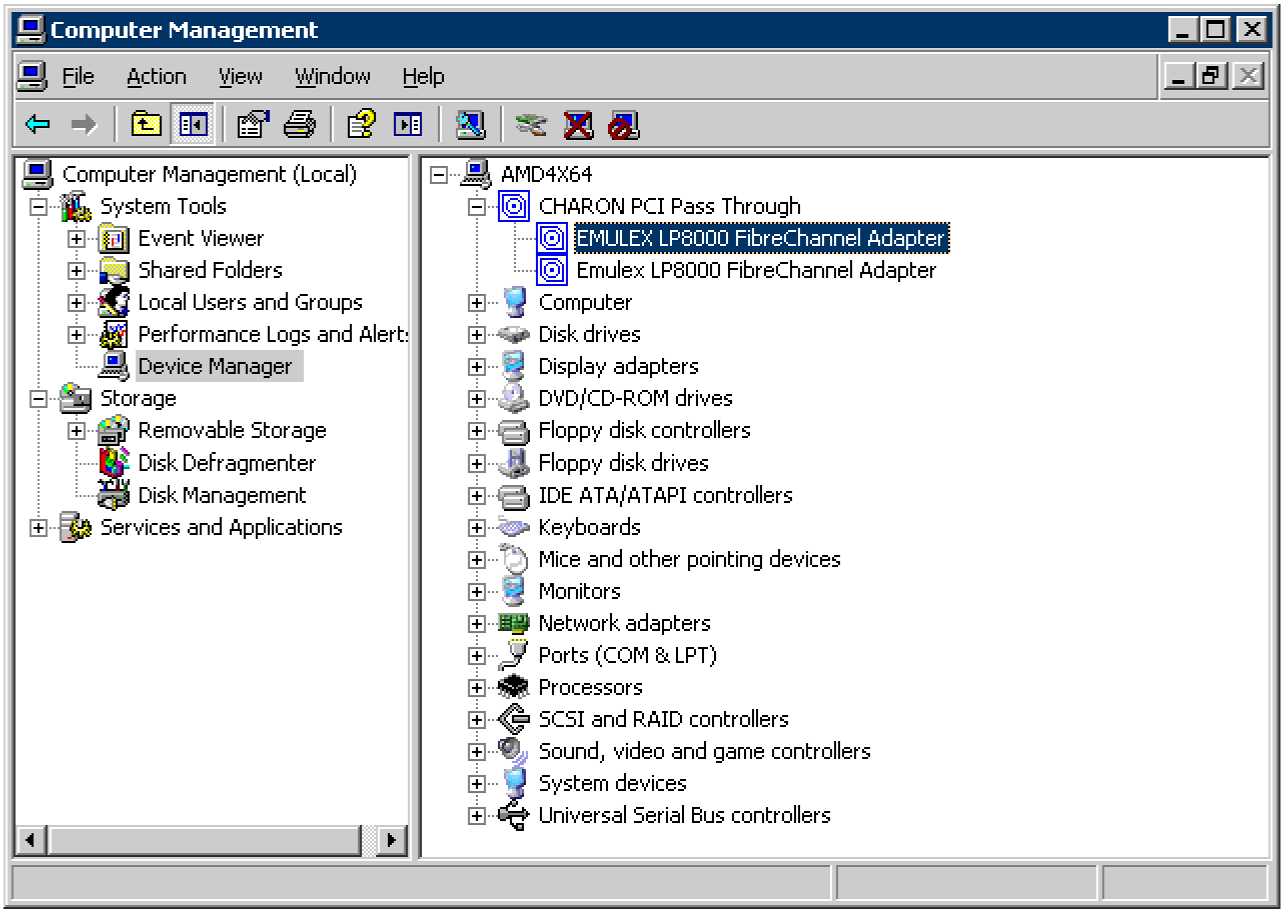
To collect the parameters directly using "Device Manager" applet, open “Computer Management” application and select “Device Manager”:



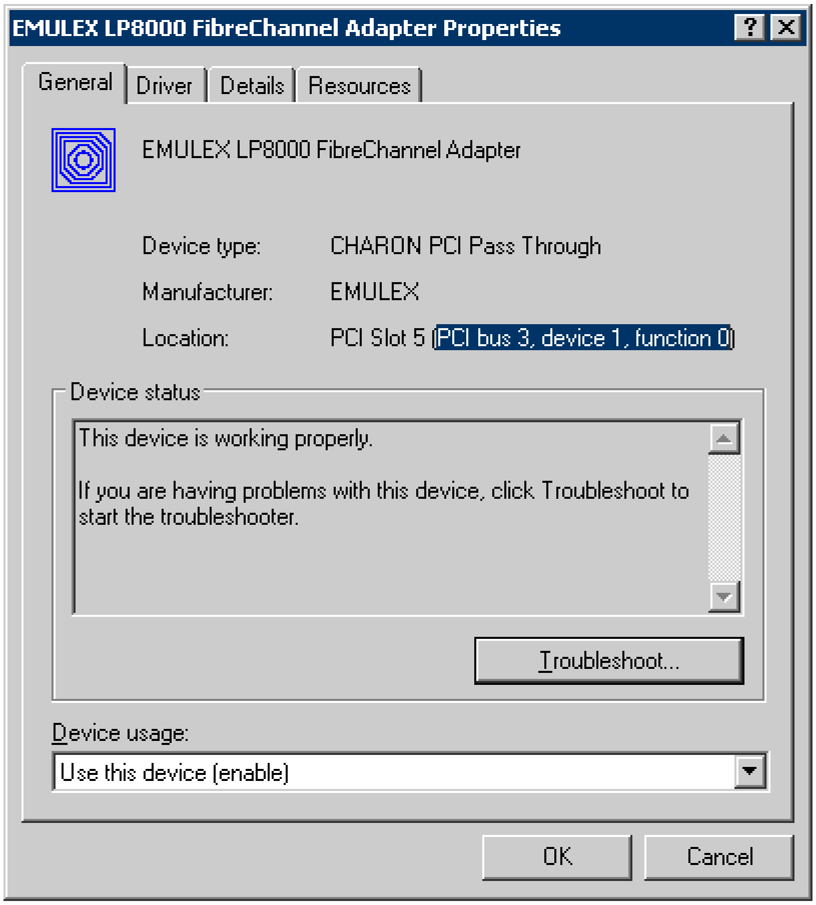
- Computer Management (Local)
  - System Tools
    - Event Viewer
    - Shared Folders
    - Local Users and Groups
    - Performance Logs and Alert:
    - Device Manager
  - Storage
    - Removable Storage
    - Disk Defragmenter
    - Disk Management
  - Services and Applications

- AMD4X64
  - CHARON PCI Pass Through
  - Computer
  - Disk drives
  - Display adapters
  - DVD/CD-ROM drives
  - Floppy disk controllers
  - Floppy disk drives
  - IDE ATA/ATAPI controllers
  - Keyboards
  - Mice and other pointing devices
  - Monitors
  - Network adapters
  - Ports (COM & LPT)
  - Processors
  - SCSI and RAID controllers
  - Sound, video and game controllers
  - System devices
  - Universal Serial Bus controllers

In the right panel select desired physical EMULEX LP FibreChannel adapter under "CHARON PCI Pass Through":



Open its properties sheet by double-clicking on the selected adapter:



EMULEX LP8000 FibreChannel Adapter Properties

EMULEX LP8000 FibreChannel Adapter

Device type: CHARON PCI Pass Through

Manufacturer: EMULEX

Location: PCI Slot 5 (PCI bus 3, device 1, function 0)

Device status

This device is working properly.

If you are having problems with this device, click Troubleshoot to start the troubleshooter.

Device usage: Use this device (enable)

The “Location:” on the above picture gives X, Y, and Z for the "host_bus_location" parameter, for example:

```
load KGPSA FGA host_bus_location = "PCI bus 3, device 1, function 0"
```

⚠ Non-US-EN installations of Windows may present “Location:” string in local language, but "host_bus_location" parameter requires English notation, so the words “PCI”, “bus”, “device”, and “function” must be specified in English.

Configuration of KGPSA-CA in pass through mode

FCMGR utility description

To configure KGPSA-CA adapter in pass through mode a special SRM console utility "FCMGR" is used (it has the same functionality as the "WWIDMGR" utility of the native HP Alpha hardware).

It provides the following functionality:

<table>
<tr><th>Command</th><th>Description</th></tr>
<tr><td><b>fc rescan {/verbose}</b></td><td>Scans connected SAN using FC adapters, discovers FC ports, storage controllers, logical units and then builds volatile FC database.
The "/verbose" qualifier enables FC communication trace on console (for diagnostic and troubleshooting).</td></tr>
<tr><td><b>fc show {adapter | port | device}</b></td><td>Displays corresponding part of volatile FC database.</td></tr>
<tr><td><b>fc set {boot | dump} udid &lt;X&gt;</b></td><td>Fills the environment variables wwid0..wwid3 and n1..n4 to identify path(s) to logical unit with the specified UDID.
These variables are later used by "INIT" to create device database entries and by OpenVMS/Tru64 to get access to boot and dump disks.
This command does NOT make any change to other environment variables.</td></tr>
<tr><td><b>fc set {boot | dump} wwid &lt;X&gt;</b></td><td>Fills the environment variables wwid0..wwid3 and n1..n4 to identify path(s) to logical unit with the specified WWID.
These variables are later used by "INIT" to create device database entries and by OpenVMS/Tru64 to get access to boot and dump disks.
This command does NOT make any change to other environment variables.
This parameter is useful if UDID is absent.
ⓘ Only right part of the displayed WWID is used for specification, for example:<pre>
P00>>>fc res
polling for units on kgpsa0, slot 2, bus 0, hose 0 ...
    pga0.0.0.2.0   PGA0   F/W Rev 2.72A2
                                               WWN 1000-0000-0263-0040
           fabric   WWN 2003-0060-0263-0040
                directory   WWN 20fc-0060-0263-0040
                port 020100   CED 8GSH 0 F88V
                                               WWN 5000-1fe0-0000-0bf1
                  lun 0000000000000000   DEC HSG80CCL V88F
             UDID:-1   WWID:01000010:6000-1fe0-0000-0bf0-3030-
3030-373f-3f3f
                  lun 0000000000000100   COMPAQ RZ1ED 0000
             UDID:-1   WWID:01000010:6000-1fe0-0000-0bf0-3030-
3030-3030-3030
P00>>>fc set boot wwid 6000-1fe0-0000-0bf0-3030-3030-3030-3030
</pre></td></tr>
<tr><td><b>fc clear</b></td><td>Clears environment variables wwid0..wwid3 and n1..n4, which automatically disable (but do NOT delete) device database entries representing FC attached devices.
This command does NOT affect volatile FC database.</td></tr>
</table>

Example of usage:

```
P00>>>fc show devices
     UDID:110 WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-0081 (ev:none)
                      via adapter     via fc_port         con
[0]                   pga0.0.0.5.1    5000-1fe1-000b-6bf1 yes (ev:none)
[1]                   pga0.0.0.5.1    5000-1fe1-000b-6bf4 yes (ev:none)
     UDID:108 WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-0039 (ev:none)
                      via adapter     via fc_port         con
[12]                  pga0.0.0.5.1    5000-1fe1-000b-6bf1 yes (ev:none)
[13]                  pga0.0.0.5.1    5000-1fe1-000b-6bf4 yes (ev:none)
```

Configuration steps using FCMGR utility

Once the configuration steps described above are done, start the CHARON-AXP and wait for the P00**>>>** prompt.

Please refer to the following example with two FC adapters PGA and PGB defined:

```
initializing ...

polling for units on kzpba0, slot 4, bus 0, hose 0 ...
   pka0.0.0.4.0         PKA0            Q-Logic/ISP PCI SCSI HBA

polling for units on kgpsa0, slot 5, bus 0, hose 0 ...
   pga0.0.0.5.0         PGA0            WWN 1000-0000-C92E-97C9
      fabric                            WWN 2003-0060-6920-4682
         directory                      WWN 20fc-0060-6920-4682
         port 021400                    WWN 5000-1fe1-000b-6bf1
            lun 0000000000000100        DEC     HSG80           V88F
         UDID:100       WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-0038
            lun 0000000000000200        DEC     HSG80           V88F
         UDID:200       WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-0074
            lun 0000000000000300        DEC     HSG80           V88F
         UDID:300       WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-007b
            lun 0000000000000400        DEC     HSG80           V88F
         UDID:400       WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-0080
            lun 0000000000006c00        DEC     HSG80           V88F
         UDID:108       WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-0039
            lun 0000000000006d00        DEC     HSG80           V88F
         UDID:208       WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-003a
            lun 0000000000000000

         port 021500                    WWN 5000-1fe1-000b-6bf4
            lun 0000000000000100        DEC     HSG80           V88F
         UDID:100       WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-0038
            lun 0000000000000200        DEC     HSG80           V88F
         UDID:200       WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-0074
            lun 0000000000000300        DEC     HSG80           V88F
         UDID:300 WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-007b
            lun 0000000000000400        DEC     HSG80           V88F
         UDID:400 WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-0080
            lun 0000000000006c00        DEC     HSG80           V88F
         UDID:108 WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-0039
            lun 0000000000006d00        DEC     HSG80           V88F
         UDID:208 WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-003a
            lun 0000000000000000

polling for units on kgpsa1, slot 6, bus 0, hose 0 ...
   pgb0.0.0.6.0         PGB0            WWN 1000-0000-C92D-8D00
      fabric                            WWN 2003-0060-6920-45ff
         directory                      WWN 20fc-0060-6920-45ff
         port 011400                    WWN 5000-1fe1-000b-6bf2
            lun 0000000000000100        DEC     HSG80           V88F
         UDID:100 WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-0038
            lun 0000000000000200        DEC     HSG80           V88F
         UDID:200 WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-0074
            lun 0000000000000300        DEC     HSG80           V88F
         UDID:300 WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-007b
            lun 0000000000000400        DEC     HSG80           V88F
         UDID:400 WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-0080
            lun 0000000000006c00        DEC     HSG80           V88F
         UDID:108 WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-0039
            lun 0000000000006d00        DEC     HSG80           V88F
         UDID:208 WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-003a
            lun 0000000000000000
         port 011500                    WWN 5000-1fe1-000b-6bf3
            lun 0000000000000100        DEC     HSG80           V88F
         UDID:100 WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-0038
            lun 0000000000000200        DEC     HSG80           V88F
         UDID:200 WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-0074
            lun 0000000000000300        DEC     HSG80           V88F
         UDID:300 WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-007b
            lun 0000000000000400        DEC     HSG80           V88F
         UDID:400 WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-0080
```

```
            lun 0000000000006c00        DEC     HSG80           V88F
         UDID:108 WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-0039
            lun 0000000000006d00        DEC     HSG80           V88F
         UDID:208 WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-003a
            lun 0000000000000000
         port 011100                    failed port login

... enter console

CHARON-AXP (AlphaServer ES40) emulator. Version 4.10
Copyright (C) 2019, STROMASYS (www.stromasys.com)

P00>>>
```

The next step is to configure paths for the FC storage:

```
P00>>>fc show devices

     UDID:100 WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-0038 (ev:none)
                        via adapter     via fc_port         con
[0]                     pga0.0.0.5.0    5000-1fe1-000b-6bf1 no (ev:none)
[1]                     pga0.0.0.5.0    5000-1fe1-000b-6bf4 yes (ev:none)
[2]                     pgb0.0.0.6.0    5000-1fe1-000b-6bf2 no (ev:none)
[3]                     pgb0.0.0.6.0    5000-1fe1-000b-6bf3 yes (ev:none)
     UDID:200 WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-0074 (ev:none)
                        via adapter     via fc_port         con
[4]                     pga0.0.0.5.0    5000-1fe1-000b-6bf1 no (ev:none)
[5]                     pga0.0.0.5.0    5000-1fe1-000b-6bf4 yes (ev:none)
[6]                     pgb0.0.0.6.0    5000-1fe1-000b-6bf2 no (ev:none)
[7]                     pgb0.0.0.6.0    5000-1fe1-000b-6bf3 yes (ev:none)
     UDID:300 WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-007b (ev:none)
                        via adapter     via fc_port         con
[8]                     pga0.0.0.5.0    5000-1fe1-000b-6bf1 no (ev:none)
[9]                     pga0.0.0.5.0    5000-1fe1-000b-6bf4 yes (ev:none)
[10]                    pgb0.0.0.6.0    5000-1fe1-000b-6bf2 no (ev:none)
[11]                    pgb0.0.0.6.0    5000-1fe1-000b-6bf3 yes (ev:none)
     UDID:400 WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-0080 (ev:none)
                        via adapter     via fc_port         con
[12]                    pga0.0.0.5.0    5000-1fe1-000b-6bf1 no (ev:none)
[13]                    pga0.0.0.5.0    5000-1fe1-000b-6bf4 yes (ev:none)
[14]                    pgb0.0.0.6.0    5000-1fe1-000b-6bf2 no (ev:none)
[15]                    pgb0.0.0.6.0    5000-1fe1-000b-6bf3 yes (ev:none)
     UDID:108 WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-0039 (ev:none)
                        via adapter     via fc_port         con
[16]                    pga0.0.0.5.0    5000-1fe1-000b-6bf1 yes (ev:none)
[17]                    pga0.0.0.5.0    5000-1fe1-000b-6bf4 no (ev:none)
[18]                    pgb0.0.0.6.0    5000-1fe1-000b-6bf2 yes (ev:none)
[19]                    pgb0.0.0.6.0    5000-1fe1-000b-6bf3 no (ev:none)
     UDID:208 WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-003a (ev:none)
                        via adapter     via fc_port         con
[20]                    pga0.0.0.5.0    5000-1fe1-000b-6bf1 yes (ev:none)
[21]                    pga0.0.0.5.0    5000-1fe1-000b-6bf4 no (ev:none)
[22]                    pgb0.0.0.6.0    5000-1fe1-000b-6bf2 yes (ev:none)
[23]                    pgb0.0.0.6.0    5000-1fe1-000b-6bf3 no (ev:none)

P00>>>fc set boot udid 400

P00>>>INIT

initializing ...

polling for units on kzpba0, slot 4, bus 0, hose 0 ...
   pka0.0.0.4.0         PKA0            Q-Logic/ISP PCI SCSI HBA

polling for units on kgpsa0, slot 5, bus 0, hose 0 ...
   pga0.0.0.5.0         PGA0            WWN 1000-0000-C92E-97C9
      fabric                            WWN 2003-0060-6920-4682
         directory                      WWN 20fc-0060-6920-4682
         port 021400                    WWN 5000-1fe1-000b-6bf1
            lun 0000000000000100        DEC     HSG80           V88F
         UDID:100       WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-0038
            lun 0000000000000200        DEC     HSG80           V88F
         UDID:200       WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-0074
            lun 0000000000000300        DEC     HSG80           V88F
         UDID:300       WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-007b
            lun 0000000000000400        DEC     HSG80           V88F
         UDID:400       WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-0080
            lun 0000000000006c00        DEC     HSG80           V88F
         UDID:108       WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-0039
            lun 0000000000006d00        DEC     HSG80           V88F
         UDID:208       WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-003a
            lun 0000000000000000
         port 021500                    WWN 5000-1fe1-000b-6bf4
            lun 0000000000000100        DEC     HSG80           V88F
         UDID:100       WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-0038
            lun 0000000000000200        DEC     HSG80           V88F
         UDID:200       WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-0074
            lun 0000000000000300        DEC     HSG80           V88F
         UDID:300 WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-007b
            lun 0000000000000400        DEC     HSG80           V88F
         UDID:400 WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-0080
            lun 0000000000006c00        DEC     HSG80           V88F
         UDID:108 WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-0039
            lun 0000000000006d00        DEC     HSG80           V88F
         UDID:208 WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-003a
            lun 0000000000000000

polling for units on kgpsa1, slot 6, bus 0, hose 0 ...
   pgb0.0.0.6.0         PGB0            WWN 1000-0000-C92D-8D00
      fabric                            WWN 2003-0060-6920-45ff
         directory                      WWN 20fc-0060-6920-45ff
         port 011400                    WWN 5000-1fe1-000b-6bf2
            lun 0000000000000100        DEC     HSG80           V88F
         UDID:100 WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-0038
            lun 0000000000000200        DEC     HSG80           V88F
         UDID:200 WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-0074
            lun 0000000000000300        DEC     HSG80           V88F
         UDID:300 WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-007b
            lun 0000000000000400        DEC     HSG80           V88F
         UDID:400 WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-0080
            lun 0000000000006c00        DEC     HSG80           V88F
         UDID:108 WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-0039
            lun 0000000000006d00        DEC     HSG80           V88F
         UDID:208 WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-003a
            lun 0000000000000000
         port 011500                    WWN 5000-1fe1-000b-6bf3
            lun 0000000000000100        DEC     HSG80           V88F
         UDID:100 WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-0038
            lun 0000000000000200        DEC     HSG80           V88F
         UDID:200 WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-0074
            lun 0000000000000300        DEC     HSG80           V88F
         UDID:300 WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-007b
            lun 0000000000000400        DEC     HSG80           V88F
         UDID:400 WWID:01000010:6000-1fe1-000b-6bf0-0009-9081-1283-0080
            lun 0000000000006c00        DEC     HSG80           V88F
         UDID:108 WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-0039
            lun 0000000000006d00        DEC     HSG80           V88F
         UDID:208 WWID:01000010:6000-1fe1-000b-6bf0-0009-0440-4014-003a
            lun 0000000000000000
         port 011100                    failed port login

... enter console

CHARON-AXP (AlphaServer ES40) emulator. Version 4.10
Copyright (C) 2019, STROMASYS (www.stromasys.com)

P00>>>SHOW DEV

sys0.0.0.0.0            SYS0          System ROOT Device
ewa0.0.0.3.0            EWA0          00-51-71-F5-8E-D8
pka0.0.0.4.0            PKA0          Q-Logic/ISP PCI SCSI HBA
pga0.0.0.5.0            PGA0          WWN 1000-0000-C92E-97C9
pgb0.0.0.6.0            PGB0          WWN 1000-0000-C92D-8D00
pqa0.0.0.15.0           PQA0          ALi 1553C Integrated IDE Controller
pqb0.0.1.15.0           PQB0          ALi 1553C Integrated IDE Controller
dqa0.0.0.15.0           DQA0          Virtual ATAPI - TEAC DW-224E-V
dka0.0.0.4.0            DKA0          Virtual SCSI Disk (C)SRI
dga400.1001.0.5.0       $1$DGA400     DEC     HSG80           V88F
dga400.1002.0.5.0       $1$DGA400     DEC     HSG80           V88F
dgb400.1003.0.6.0       $1$DGA400     DEC     HSG80           V88F
dgb400.1004.0.6.0       $1$DGA400     DEC     HSG80           V88F

P00>>>BOOT $1$DGA400

dga400.1001.0.5.0: failed to open device
(boot dga400.1002.0.5.0)
jumping to bootstrap code

    OpenVMS (TM) Alpha Operating System, Version V7.3-2
      © Copyright 1976-2003 Hewlett-Packard Development Company, L.P.
```

```
%SMP-I-CPUTRN, CPU #02 has joined the active set.
%SMP-I-CPUTRN, CPU #03 has joined the active set.
%SMP-I-CPUTRN, CPU #01 has joined the active set.
Please enter date and time (DD-MMM-YYYY HH:MM)
```

Acer Labs 1543C IDE/ATAPI CD-ROM adapter

Table of Contents

General description

CHARON-AXP supports emulation of an integrated virtual Acer Labs 1543C IDE/ATAPI controller.

Loading Acer Labs 1543C IDE/ATAPI adapter

By default the integrated virtual Acer Labs 1543C IDE/ATAPI controller is preloaded with a name "ide".

Example:

```
set ide container="\\.\CdRom0"
```

Configuration parameters

The Acer Labs 1543C IDE/ATAPI adapter emulation has only one configuration parameter:

container

Parameter	container
Type	Text String
Value	Specifies the name of ATAPI or SATA CD/DVD-ROM drive attached to the host system. The supported values are of the form “\\.\CdRomN”, where N is 0, 1, 2… In most cases 0 is the only meaningful value for N, because usually the host system has only one CD/DVD-ROM drive. By default it is left unspecified.

Example

```
set ide container="\\.\CdRom0"
```

When running HP OpenVMS/Alpha Operating System on CHARON VM the specified CD/DVD-ROM drive is available as DQA0: device.

CHARON-AXP is able to boot any OpenVMS/Alpha and Tru64 version from Acer Labs 1543C IDE/ATAPI CD-ROM.

Virtual Acer Labs 1543C IDE/ATAPI can me mapped only to physical CD-ROM drives. If a CD-ROM container or an ISO file should be used, it is required to utilize KZPBA-CA controller as it offers full support of both physical and virtual mappings to system resources.

PCI I/O Bypass controller

Table of Contents

General description

CHARON-AXP supports PCI I/O bypass controller for accessing to disk images and host physical disks. PCI I/O bypass controller requires a specific driver to be installed.

❗ PCI I/O bypass controller support is available only for OpenVMS guest operating system.

Prerequisites

This release supports VMS version V6.2-1H3 and higher. Bypass disks can not be used as a boot device in V6.2-1H3, higher versions do not have this restriction.

Make sure that the latest Bypass controller kit has been installed, especially for VMS versions before V7.3-2.

Installation

1. Open your VM configuration and attach the virtual disk "ovms_tool.vdisk" located by default in the "C:\Program Files\CHARON\Virtual Disk Images\OpenVMS tools package" directory:
   `set PKA container[400] = "C:\Program Files\CHARON\Virtual Disk Images\OpenVMS tools package\ovms_tools.vdisk"`
2. Run VM, boot guest OpenVMS operating system.
3. Mount the disk, its label is "TOOLS" (Example: `$` **`MOUNT DKA400: TOOLS`**)
4. Use the POLYCENTER Software Installation (PCSI) utility to install the Bypass Driver. The following example demonstrates the "PCSI `PRODUCT INSTALL"` command to execute and the expected output (the example assumes the utilities virtual disk image is attached as `DKA400:`):

```
$ PRODUCT INSTALL CHARON_DISK/SOURCE=DKA400:[BYPASS]
%PCSI-I-CANNOTVAL, cannot validate DKA400:[BYPASS]SRI-AXPVMS-CHARON_DISK-V0104--1.PCSI;1
-PCSI-I-NOTSIGNED, product kit is not signed and therefore has no manifest file

The following product has been selected:
    SRI AXPVMS CHARON_DISK V1.4 Layered Product

Do you want to continue? [YES] YES

Configuration phase starting ...

You will be asked to choose options, if any, for each selected product and for any products that may be
installed to satisfy software dependency requirements.

SRI AXPVMS CHARON_DISK V1.4: Charon disk driver V1.4 for OpenVMS Alpha.

    Copyright (C) 1976, 2009 Software Resources International

    CHARON_DISK was produced by Software Resources International

* This product does not have any configuration options.

Execution phase starting ...

The following product will be installed to destination:
    SRI AXPVMS CHARON_DISK V1.4 DISK$TARDISSYSTEM:[VMS$COMMON.]

Portion done: 0%...10%...20%...80%...100%

The following product has been installed:
    SRI AXPVMS CHARON_DISK V1.4 Layered Product
```

Loading PCI I/O bypass controller

Syntax for loading PCI I/O bypass storage adapter:

```
load pci_io_bypass <name>
```

The <name> can be DI<x>, DR<x> or DU<x>

where x is selected according to VMS naming scheme, i.e. A stands for the first controller of given type, B - for the second, etc.

Example:

```
load pci_io_bypass DIA
```

⚠ In AlphaStation 400 configuration use the following syntax for PCI I/O bypass storage loading:

```
load pci_io_bypass DIA irq_bus = isa
```

The adapter instance name ("DIA" in the example above) is used then for parametrization, for example:

```
set DIA container[0]="C:\My disks\vms_distribution.vdisk"
```

The numbers in the square brackets represent a number of device on PCI I/O Bypass controller.

The maximum number of I/O Bypass controller devices is 64.

By default I/O Bypass controller uses PCI slot corresponded to the <x> parameter (see above). If instead some particular slot is needed, refer to this section for details of specific placement of PCI peripherals on CHARON Virtual Machine (VM) PCI bus (note that "irq_bus" and "irq" parameters are ignored for I/O Bypass controller). In this case the <x> will be changed automatically according to custom position of I/O Bypass controller on PCI bus.

ℹ I/O Bypass controller is implemented for OpenVMS only.

Configuration parameters

The I/O Bypass controller has the following configuration parameters:

container

Parameter	container[N] N is 0..32766 (no more than 64 units)
Type	Text string

Value	Possible values of the parameter are strings in one of the following forms:
	• **Physical disk**
	◦ "\\.\PhysicalDrive<X>", where X is 0, 1, …
	⚠ Be careful not to destroy all the information from the disk dedicated to CHARON VM by mistake.
	These disks must not be formatted by the host OS.
	Example:
	`set DIA container[0]="\\.\PhysicalDrive1"`
	• **Physical disk by its WWID**
	◦ "\\.\PhysicalDrive(DevID=XXX-...-XXXX)"
	⚠ Be careful not to destroy all the information from the disk dedicated to CHARON VM by mistake.
	These disks must not be formatted by the host OS.
	DevID addresses the target physical disk by its WWID (hexadecimal 128-bit identifier assigned to the disk drive by its manufacturer/originator).
	Example:
	`set DIA container[100]="\\.\PhysicalDrive(DevID=6008-05F3-0005-2950-BF8E-0B86-A0C7-0001)"`
	• **iSCSI disks**
	◦ "\\.\PhysicalDrive(iScsiTarget = <iSCSI target>, LUN = <LUN number>)"
	iScsiTarget addresses the disk by its iSCSI target name.
	LUN specifies LUN on connected iSCSI disk.
	Example:
	`set DIA container[200]="\\.\PhysicalDrive(iScsiTarget=iqn.2008-04:iscsi.charon-target-test1, LUN= 1)"`
	• **File representing a physical disk of the HP Alpha system (disk image)**

  - [<drive>":\"<path-name>"\"]<file-name>[".vdisk"]
    These files can be created from scratch with "MkDisk" utility. Data and OS disks backups are transferred from the original system via tapes or network and restored into these container files.
    Mapping may also include the full path (recommended), for example: "C:\My disks\my_boot_disk.vdisk"
    Example:
    ```
    set DIA container[401]="tru64-v51-system.vdisk"
    ```

    ⚠ Using compressed folders to store virtual disks and tapes is not supported

- **Other type of drive, for example magneto-optical drive**
  - "\\.\<N>:"
    Example:
    ```
    set DIA container[300]="\\.\Z:"
    ```

    ⚠ Using compressed folders to store virtual disks and tapes is not supported

ℹ This parameter is initially not set, thus creating NO storage elements on the controller.

removable

Parameter	removable[N] N is 0..32766 (no more than 64 units)
Type	Boolean
Value	When set to "true", the removable configuration parameter instructs CHARON VM to report the corresponding virtual disk as removable. Note to set this parameter to "true" if this disk storage element is: • Mapped to a host removable device • Mapped to a disk image located on a host removable device • Mapped to a disk image that has to be renamed/moved/deleted right after dismounting it in the guest OS By default the removable configuration parameter is set to "false". **Example**: `set DIA removable[400]=true`

geometry

Parameter	geometry [N] N is 0..32766 (no more than 64 units)
Type	Text String
Value	This formatted string value specifies the explicit geometry of the disk storage element. This parameter is not applicable to tape storage elements. The string format is ”*/*/*/<B>" or ”*,*,*<B>" where B is the total size of the disk (in blocks) reported to the guest OS. If omitted it is calculated automatically. If this parameter is not set, CHARON VM will configure the geometry based on the most probable disk type. Initially not set.

use_io_file_buffering

Parameter	use_io_file_buffering[N] N is 0..32766 (no more than 64 units)
Type	Text String
Value	Instructs CHARON-AXP to enable host operating system I/O cache on reading/writing operations. The following values are possible: ● "true" or "write-back" - the "write back" caching mode is used ● "write-through" - the "write through" caching mode is used ● "false" - caching is switched off ℹ Note that this caching has a significant effect only in case of mapping to disk and tape containers, not physical drives. When enabled, host operating system I/O cache may significantly improve I/O performance of the virtual system. At the same time maintaining I/O cache requires additional host resources (CPU and memory) which may negatively affect overall performance of the virtual system. Initially is set to "false". **Example**: `set DIA use_io_file_buffering[603]=true`

When a disk image connected to an I/O Bypass controller is dismounted by OpenVMS, it is disconnected from CHARON-AXP and can be manipulated. It can be replaced with a different disk image if it keeps the same name. This capability may be useful when designing back-up and restore procedures. When copying CHARON-AXP disk images while CHARON-AXP is running, please take care to minimize the risk of overloading a heavily loaded CHARON host system. For example, using a sequential series of simple ftp binary copies is less resource intensive and thus less disruptive than multiple, simultaneous copies.

Empty disk images are created with the "MkDisk" utility.

CHARON-AXP is able to boot from disk images of any OpenVMS/Alpha version.

Usage example

Example using DR disk devices (disk drives connected to a SWXCR raid controller):

```
load pci_io_bypass DRA
set DRA container[1] = "d:\Vdisks\dra1.vdisk"
set DRA container[2] = "d:\Vdisks\dra2.vdisk"
set DRA container[3] = "d:\Vdisks\dra3.vdisk"
set DRA container[4] = "d:\Vdisks\dra4.vdisk"
```

Deinstallation

1. Do a conversational boot. Please refer to your OpenVMS system administration guide for instructions.
2. Set the `NOAUTOCONFIG` system parameter to `1`
3. Boot OpenVMS
4. Remove the product with the "`$ PRODUCT REMOVE CHARON_DISK`" command.
5. Set the `NOAUTOCONFIG` system parameter to `0` and reboot.

Networking

Table of Contents

General description

CHARON-AXP supports emulation of the following network adapters:

- DE435
- DE450
- DE500AA
- DE500BA
- DE602
- DE602AA

Each of them is a PCI Ethernet adapter based on the DEC21040 (DE435, DE450, DE500AA and DE500BA) and the Intel i8255x (DE602 and DE602AA) PCI Ethernet adapter chips for the Alpha.

CHARON-AXP maps the virtual adapter to a dedicated Ethernet adapter in the Windows host system.

All the emulated controllers are loaded and configured in the same way.

⚠ The Ethernet adapter in the Windows host system must support dynamic changes of its MAC address (i.e. no reboot of the host system is required to change the MAC address), which is the case with nearly all modern Ethernet adapters.

ℹ By default the PCI Ethernet adapters use first available PCI slot. If instead some particular slot is needed, refer to this section for details of specific placement of PCI peripherals on CHARON-AXP PCI bus.

Configuration steps

To configure CHARON-AXP networking, follow these 3 steps:

1. **Load network adapter (if required)**

   Use the "load" command as shown below.

   Example:

For DEC21040 adapters	For Intel i8255x adapters
`load DE500BA/dec21x4x NIC`	`load DE602/i8255x NIC`

   ℹ By default each loaded virtual network adapter uses first available PCI slot. If instead some particular slot is needed, refer to this section for details of specific placement of PCI peripherals on CHARON-AXP PCI bus.

   In **AlphaStation 400 configurations** use the following syntax for network adapter loading:

   ```
   load DE500AA/dec21x4x NIC irq_bus = isa
   ```

2. **Load "packet_port"**

   Load "packet_port" to connect network adapter to the host hardware network card (or to a virtual network interface).

   Example:

   ```
   load packet_port/chnetwrk NDIS interface = "connection:Charon"
   ```

3. **Connect the loaded "packet_port" to the loaded virtual network adapter**

   Connect the network adapter to the "packet_port" by setting the interface name.

   Example:

   ```
   set NIC interface = NDIS
   ```

Please note:

- The interface name can be either "`(disabled)`" for a disabled interface or "`connection:<`*`Windows network interface name`*`>`"

  Examples:
  ```
  load packet_port/chnetwrk NIC1 interface="(disabled)"
  load packet_port/chnetwrk NIC2 interface="connection:CHARON_NIC"
  ```

- The AlphaServer DS15 and DS25 contain two built-in PCI Ethernet adapters. Models and names (EI* or EW*) of them depend on configuration addon. Choose one of the two or none, but not both. The first instantiates onboard network interfaces as EIA and EWA. While the second - EWA and EWB (enabled by default for backward compatibility)
  Example:
  ```
  #include ds25-onboard-nics.icfg
  include ds25-onboard-nics-ew.icfg
  ```

- The specification of "chnetwrk" is not mandatory, but it is still used for compatibility.

Configuration parameters

Each virtual network controller has the following parameters that are specified with the "set" command:

interface

Parameter	interface
Type	Text string.
Value	Name of the corresponding instance of the "packet_port" component

station_address

Parameter	station_address
Type	Text String

Value	The "station_address" provides the ability to configure the adapter's permanent address. By default the adapter's permanent address is read from the host system's NIC.
	Format:
	`XX-XX-XX-XX-XX-XX`
	or
	`XX:XX:XX:XX:XX:XX`
	Example:
	`set EWA station_address="AF:01:AC:78:1B:CC"`

rx_fifo_size

Parameter	rx_fifo_size
Type	Numeric
Value	"rx_fifo_size" sets the receive FIFO size. The value is specified in Kb and, by default, is pre-calculated from the connected port's size of the receive queue. Typically, you do not need to change the "rx_fifo_size" parameter. It is available for extended tuning and debugging purposes.

adapter_mode

Parameter	adapter_mode
Type	Text String
Value	Assigns the link speed and the duplex settings of the virtual network adapter (except for DE602/DE602AA - see Networking#DE602 and DE602AA network adapters link speed and duplex settings). The values are: - "Auto" for auto-negotiate (default) - "10BaseT-HD" for 10Mbps half duplex - "10BaseT-FD" for 10Mbps full duplex - "100BaseT-HD" for 100Mbps half duplex - "100BaseT-FD" for 100Mbps full duplex **Example**: `set EWA adapter_mode="100BaseT-HD"` **Important notes:** - This parameter controls only the setting which the emulated network adapter reports to the guest operating system running on CHARON-AXP. It does not change any settings of the host adapter mapped to the virtual adapter! In case if the settings of the host adapter have to be changed please refer to the host Operating System User's Guide for details on how to do it. - In case the "Auto" value is set CHARON-AXP collects the settings from the mapped host network adapter directly, assuming that all speeds above 100Mbps are represented as 100Mbps (maximum value). The resulting duplex setting corresponds to the duplex setting of the mapped host network adapter.

Example

```
load packet_port/chnetwrk EWA0 interface = "connection:Charon"
set EWA interface = EWA0
set EWA station_address="0C:FE:35:AA:67:3B"
```

DE602 and DE602AA network adapters link speed and duplex settings

Regardless of the "adapter_mode" setting in CHARON VM configuration file (see above), DE602 and DE602AA network adapters remains in "Auto-negotiation" mode, since the EIDRIVER of OpenVMS checks for EIx0_MODE environment variable when configuring the network card.

So mode propagation is implemented in CHARON-AXP via SRM console EIx0_MODE environment variable ("x" is A, B, C... depending on CHARON-AXP configuration), for example:

```
>>>help set

usage: set <variable-name> <value>
set <variable-name> ""

set eia0_mode { Twisted | Full | Fast | FastFD | Auto* }

>>>
```

⚠ The EIx0_MODE variable name is case insensitive, while its values are case sensitive! This is feature of OpenVMS EIDRIVER.

The values are:

Parameter	Description
"Auto"	Auto-negotiate (default)
"Twisted"	10Mbps half duplex
"Full"	10Mbps full duplex
"Fast"	100Mbps half duplex
"FastFD"	100Mbps full duplex

Example:

```
>>>set eia0_mode FastFD
```

Important notes:

- Please note that EIx0_MODE environment variable controls only the setting which the emulated network adapter reports to the guest operating system running on CHARON VM. It does not change any settings of the host adapter mapped to the virtual adapter! In case if the settings of the host adapter have to be changed please refer to the host Operating System User's Guide for details on how to do it.
- In case the "Auto" value is set CHARON-AXP collects the settings from the mapped host network adapter directly, assuming that all speeds above 100Mbps are represented as 100Mbps (maximum value). The resulting duplex setting corresponds to the duplex setting of the mapped host network adapter.

Packet Port

The CHARON-specific "packet_port" interface establishes a connection between an Ethernet adapter in the Windows host system and a network adapter in the virtual HP Alpha system.

For every virtual adapter instance loaded, one dedicated host Ethernet physical adapter is required.

To create instances of the "packet_port", use the “load” command in the configuration file as follows:

```
load packet_port/chnetwrk <instance-name>
```

Example:

```
load packet_port/chnetwrk NDIS
```

"packet_port" uses several configuration parameters to control its behavior.

interface

Parameter	interface
Type	Text string

Value	This parameter identifies an Ethernet adapter of the host system dedicated to CHARON-AXP. The leading keyword "`connection:`" is mandatory except if the interface is to be disabled.
	Syntax:
	`set <name> interface="connection:<adapter>"`
	Example:
	`set NDIS interface="connection:Charon"`

port_enable_mac_addr_change

Parameter	port_enable_mac_addr_change
Type	Boolean
Value	If "true" is specified (default value), CHARON-AXP sets the appropriate Ethernet address automatically. If "false" is specified, set the Ethernet address manually. **Example**: `set NDIS port_enable_mac_addr_change=false`

port_retry_on_tx

Parameter	port_retry_on_tx
Type	Numeric
Value	The "port_retry_on_tx" parameter controls the number of times a port will attempt to transmit a packet before giving up. By default, the value is 3. Increasing this value may introduce problems in carrier loss logic, because not all NIC drivers support a carrier status query. Typically, you do not need to increase the value. **Example**: `set NDIS port_retry_on_tx=8`

port_pending_rx_number

Parameter	port_pending_rx_number
Type	Numeric
Value	The "port_pending_rx_number" parameter sets the number of pending receive buffers. The default value is 63. The maximum value allowed is 195. You may want to increase the "port_pending_rx_number" when you have very busy networking and experience problems like losing connections not related to the carrier loss. Typically, you do not need to change this parameter. **Example**: `set NDIS port_pending_rx_number=128`

port_pending_tx_number

Parameter	port_pending_tx_number
Type	Numeric
Value	The "port_pending_tx_number" parameter sets the number of buffers the port uses to transmit. The default value is 62. You may want to increase the "port_pending_tx_number" value if the log file indicates dropped TX packets due to TX queue overflow. Typically, you do not need to change this parameter. **Example**: `set NDIS port_pending_tx_number=128`

suspend_msg_on_mac_change

Parameter	suspend_msg_on_mac_change
Type	Boolean
Value	To avoid confusion arising from non critical errors during a MAC address change, logging is by default suppressed (default value is "true"). To enable tracing during a MAC address change set this parameter to "false" **Example**: `set NDIS suspend_msg_on_mac_change=false`

log

Parameter	log
Type	Text string

Value	If this parameter is set to some valid file name or a directory where the log files for each individual session will be stored CHARON-AXP logs Recv and Xmit packets at the emulated port layer.
	If an existing directory is specified, CHARON-AXP automatically enables creation of individual log files, one for each session using the same scheme as used for the generation of the rotating log files. If the "log" parameter is omitted, CHARON-AXP does not create log.
	In certain situations enabling this parameter may help to detect loss of packets.
	Example 1:
	`set NDIS log="ndis.log"`
	Example 2:
	`set NDIS log="C:\Charon\Logs"`
	Only an existing directory can be used as a value of the "log" parameter.

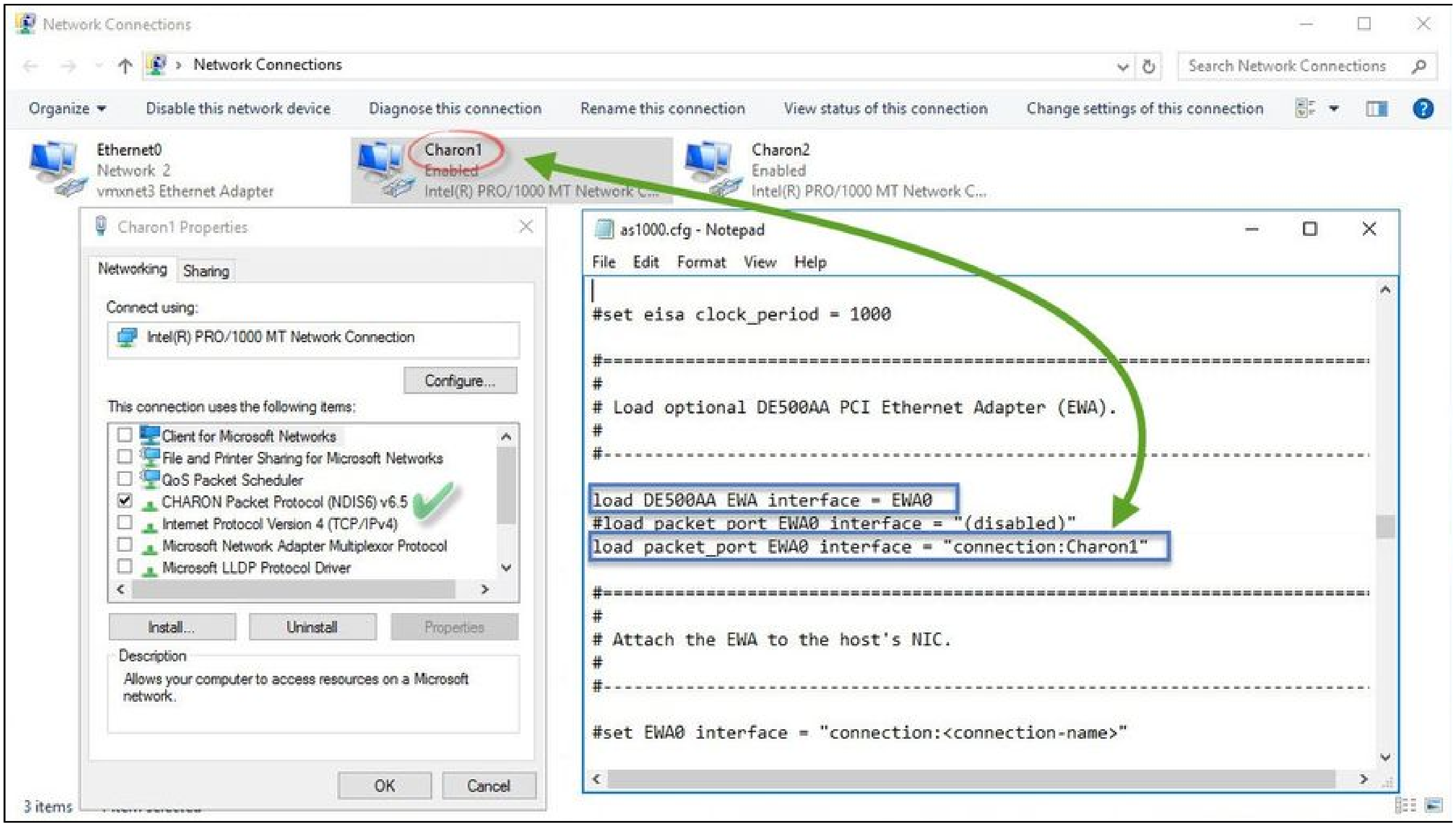
log_flush_period

Parameter	log_flush_period
Type	Numeric
Value	• <period-in-seconds> Defines a period of flushing log to disk. Default period is 60 seconds (it means that every minute, a log file is flushed to disk) **Examples**: `set NDIS log_flush_period=30`

legacy_mode

Parameter	legacy_mode
Type	Boolean
Value	If this parameter is set to "true" CHARON-AXP tries to use very first implementation of "packet_port" having certain restrictions such as inability to automatically change MAC address on given interface etc. This setting may be needed if CHARON-AXP runs in VMware environment for example and you do not want to allow promiscuous mode. In other situations please do not change its default value ("false"). **Example**: `set NDIS legacy_mode=true` **Please note**: • This parameter is required only if you do not want to allow promiscuous traffic for the VMware adapter/port. In this mode, CHARON-AXP uses legacy MAC address change logic, so use of this option is not desirable. **Thus for ESXi networking configuration**: 1. Create the port; 2. Enable the promiscuous traffic; 3. Do not use legacy_mode option in Charon's configuration file • TruCluster alias vMAC option may not operate correctly in combination with "legacy_mode=true" when host is VMware virtual machine. This is because vMAC option sets adapter to promiscuous mode which does not seem operating correctly on VMware adapters.

Example



Connect using:

Intel(R) PRO/1000 MT Network Connection

This connection uses the following items:

- ☐ Client for Microsoft Networks
- ☐ File and Printer Sharing for Microsoft Networks
- ☐ QoS Packet Scheduler
- ☑ CHARON Packet Protocol (NDIS6) v6.5
- ☐ Internet Protocol Version 4 (TCP/IPv4)
- ☐ Microsoft Network Adapter Multiplexor Protocol
- ☐ Microsoft LLDP Protocol Driver

Description

Allows your computer to access resources on a Microsoft network.

```
#set eisa clock_period = 1000

#==============================================================================
#
# Load optional DE500AA PCI Ethernet Adapter (EWA).
#
#------------------------------------------------------------------------------

load DE500AA EWA interface = EWA0
#load packet_port EWA0 interface = "(disabled)"
load packet_port EWA0 interface = "connection:Charon1"

#==============================================================================
#
# Attach the EWA to the host's NIC.
#
#------------------------------------------------------------------------------

#set EWA0 interface = "connection:<connection-name>"
```

CHARON-AXP supports VLAN adapters. If for some reasons you are going to use them, proceed with their installation and configuration according to the network adapter's vendor's User's Guide and then use the resulting VLAN interface the same way as the regular network interface.

DEFPA PCI FDDI adapter

Table of Contents

General description

CHARON-AXP supports emulation of DEFPA PCI FDDI adapter via CHARON PCI Pass Through mode (using a specific CHARON PCI Pass Through driver).

CHARON PCI Pass Through mode enables connection between the virtual DEFPA PCI FDDI adapter and the physical DEFPA PCI FDDI adapter plugged into a hosting server PCI bus.

Loading DEFPA PCI FDDI adapter

Syntax for loading DEFPA PCI FDDI adapter:

```
load defpa <name>
```

Example:

```
load defpa FDDI
```

By default DEFPA adapter uses first available PCI slot. If instead some particular slot is needed, refer to this section for details of specific placement of PCI peripherals on CHARON Virtual Machine (VM) PCI bus.

⚠ In AlphaStation 400 configuration use the following syntax for DEFPA PCI FDDI adapter loading:

```
load defpa FDDI irq_bus = isa
```

Configuration parameters

The DEFPA PCI FDDI adapter emulation has only one configuration parameter:

host_bus_location

Parameter	host_bus_location
Type	Text String

Value	Establish connection between virtual DEFPA PCI FDDI adapter and physical DEFPA PCI FDDI adapter installed on CHARON host (pass through mode)
	Syntax:
	`load defpa <controller name> host_bus_location="PCI bus X, device Y, function Z"`
	where:
	■ "PCI bus X" corresponds to the PCI bus number of the host DEFPA PCI FDDI adapter
	■ "device Y" corresponds to the PCI bus device number of the host DEFPA PCI FDDI adapter
	■ "function Z" corresponds to the "function" parameter of the the host DEFPA PCI FDDI adapter
	Example:
	`load defpa FDDI host_bus_location="PCI bus 3, device 1, function 0"`

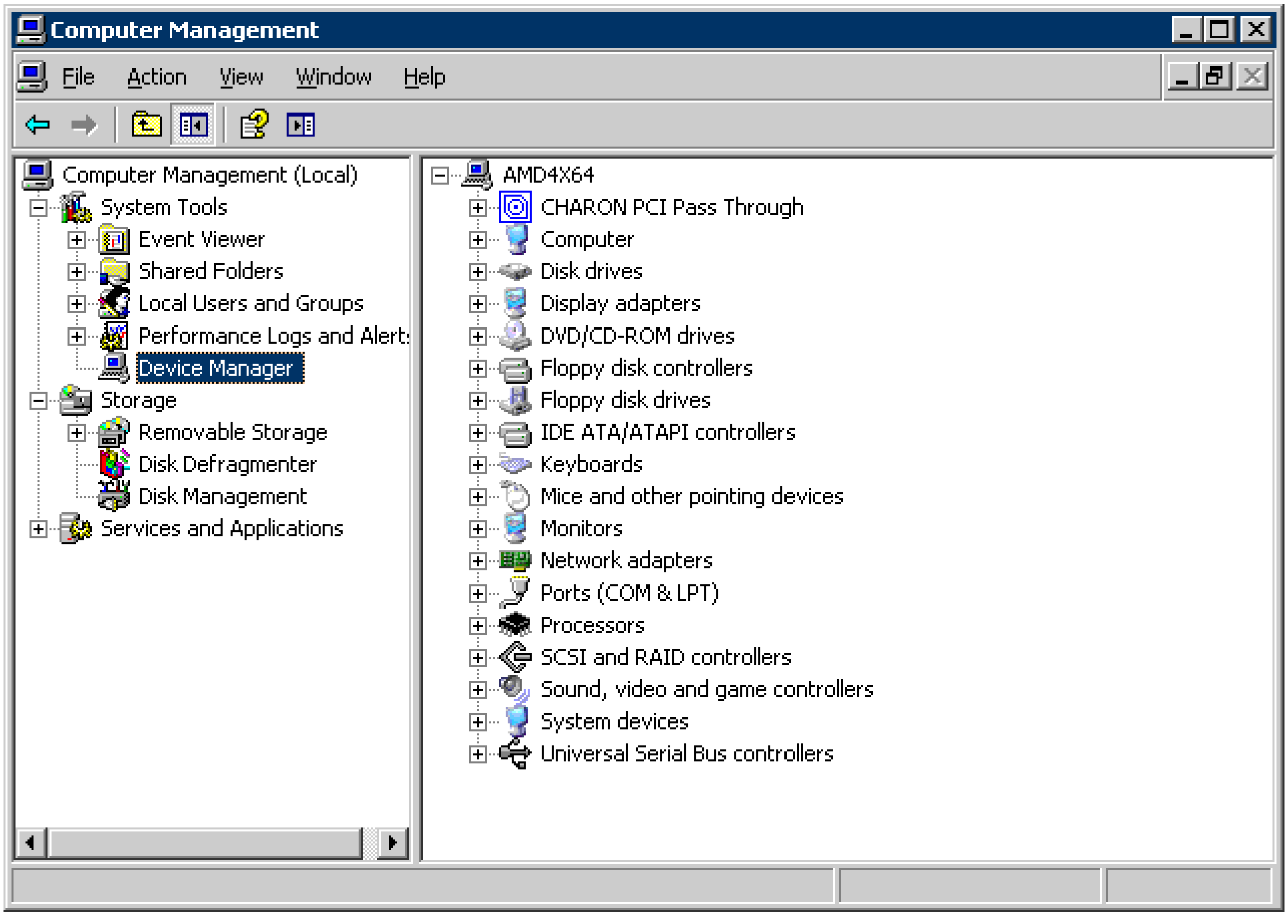
Installation of DEFPA PCI FDDI adapter

1. Install the DEFPA PCI FDDI adapter to some spare PCI slot of the host system
2. Boot a Windows operating system
3. Open “Computer Management”
4. Select “Device Manager”
5. In the right window select the desired physical FDDI adapter connected to the system, right-click the mouse button, and the corresponding menu will appear.
6. From the menu select “Update driver…”. Windows will show “Hardware Upgrade Wizard”
7. Select “No, not this time”, click “Next”
8. Select “Install from a list or specific location (Advanced)”, click “Next”
9. Select “Don’t search. I will choose the driver to install”, click “Next”
10. Click “Have Disk…”. Windows shows dialog “Install From Disk”
11. Instead of “A:\”, click “Browse” and select path to the folder in which driver’s INF file is located (typically "C:\Program Files\CHARON\Drivers\DEFPA_X.X.0.XXXXX"), select “defpa_ppt_amd64.inf”, and click “Open”
12. The “Hardware Upgrade Wizard” should have “CHARON DEFPA FDDI adapter”. Select it, and click “Next”
13. There will be one or two more dialogs, but they are usual for device driver installation.
14. Reboot the host

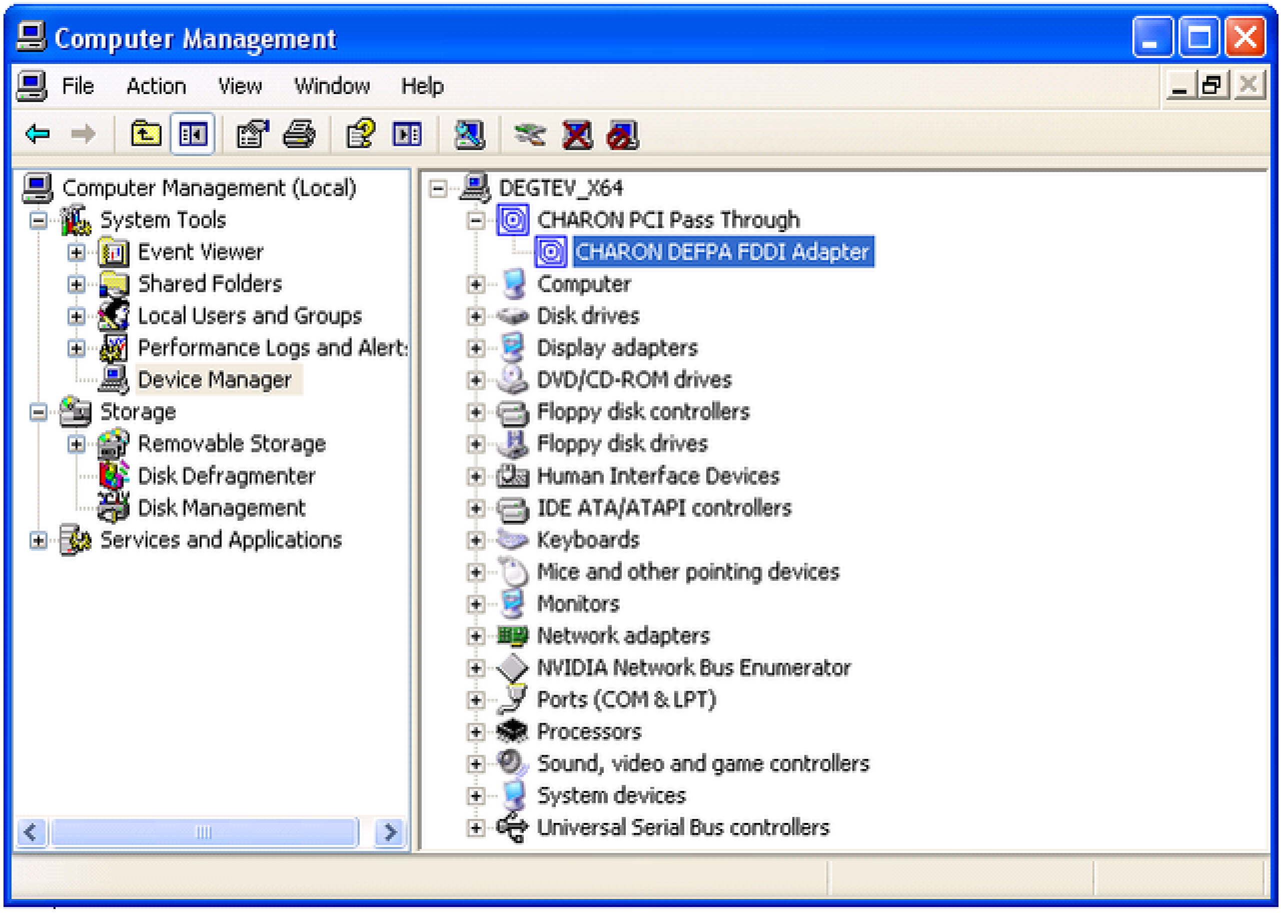
Upon completion, a new device will appear in the device manager with the CHARON logo on it.

Mapping to host DEFPA PCI FDDI adapter

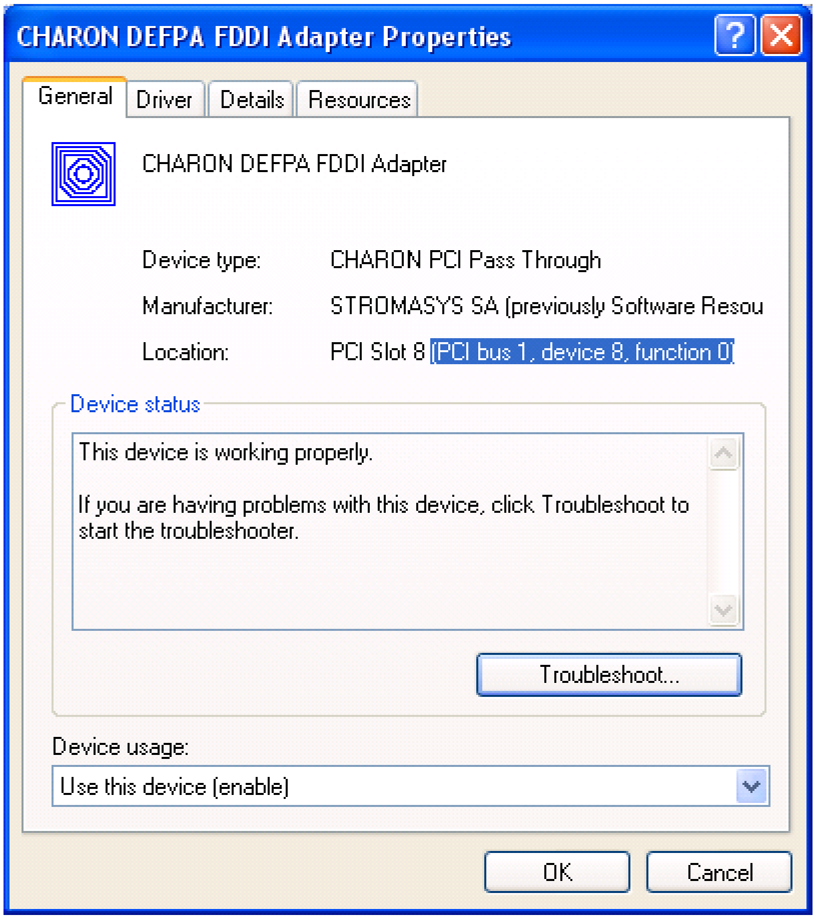
Open “Computer Management” application and select “Device Manager”:



In the right panel select the installed DEFPA PCI FDDI adapter:



Open its properties sheet by double-clicking on the selected adapter:



CHARON DEFPA FDDI Adapter

Device type: CHARON PCI Pass Through

Manufacturer: STROMASYS SA (previously Software Resou

Location: PCI Slot 8 (PCI bus 1, device 8, function 0)

Device status

This device is working properly.

If you are having problems with this device, click Troubleshoot to start the troubleshooter.

Device usage:

Use this device (enable)

The “Location:” on the above picture gives X, Y, and Z for the "host_bus_location" parameter, for example:

```
set FDDI host_bus_location = "PCI bus 1, device 8, function 0"
```

⚠ Non-US-EN installations of Windows may present “Location:” string in local language, but "host_bus_location" parameter requires English notation, so the words “PCI”, “bus”, “device”, and “function” must be specified in English.

PBXDA-xx series PCI serial adapters

Table of Contents

General description

CHARON-AXP supports emulation of PBXDA-xx family PCI serial adapter based on DIGI AccelePort serial adapters via CHARON PCI Pass Through mode (using a specific CHARON PCI Pass Through driver) and by direct virtualization.

CHARON PCI Pass Through mode enables connection between the virtual PBXDA-xx PCI serial adapter and the physical DIGI AccelePort PCI serial adapter plugged into a hosting server PCI bus whether as the direct virtualization may address the host serial port or TCP/IP port.

Virtual PBXDA-xx PCI serial adapter

Loading virtual PBXDA-xx serial lines adapter

Syntax for loading PBXDA (AccelePort 2r 920) serial lines adapter:

```
load PBXDA/DIGI <name>
```

Syntax for loading PBXDA_BA (AccelePort 4r 920) serial lines adapter:

```
load PBXDA_BA/DIGI <name>
```

Syntax for loading PBXDA_BB (AccelePort 8r 920) serial lines adapter:

```
load PBXDA_BB/DIGI <name>
```

Syntax for loading PBXDA_AC (AccelePort Xem) serial lines adapter:

```
load PBXDA_AC/DIGI <name>
```

Example:

```
load PBXDA/DIGI TXA
```

The adapter instance name ("TXA" in the example above) is used then for parametrization, for example:

```
set TXA line[2]="COM1:"
```

The numbers in the square brackets represent line number on the virtual PBXDA-xx adapter starting from 0.

Controller type	Maximum number of lines
PBXDA	2
PBXDA_BA	4
PBXDA_BB	8
PBXDA_AC	16

ⓘ All the parameters described in the "Placement of peripheral devices on PCI bus" chapter, such as "bus", "device", "function", "irq", "irq_bus" are applicable for PBXDA-xx controller.

⚠ DIGI drivers for OpenVMS and Tru64 are sensitive to PBXDA-xx location on PCI, therefore it is recommended to fix PBXDA-xx location with explicit configuration.

Example:

```
load PBXDA TXA bus=pci_1 device=4 function=0
```

Configuration parameters

The PBXDA-xx serial lines adapter emulation has the following configuration parameters:

port

Parameter	port
Type	Text String

Value	Specifies a local port for incoming telnet connections
	By default the "port" configuration option is not specified.
	Syntax:
	`port[line-number]=<local port>`
	Example:
	`set TXA port[2]=17060`

line

Parameter	line
Type	Text string
Value	A defined COM port on host system: **Example:** `set TXA line[2]="COM2:"`

log

Parameter	log
Type	Text string
Value	The optional log file or directory name is where the log file for the serial line is stored. **Example:** `set TXA log[0]="D:\Charon\txa0.log"`

Pass-through mode

Loading PBXDA-xx PCI serial adapter in pass-through mode

Note the model of DIGI AccelePort PCI serial adapter for emullation of a particular model of PBXDA-xx family PCI serial adapter:

DEC PBXDA-xx adapter	Name of the device to map to	Controller	Vendor ID	Device ID
PBXDA-BA	DIGI AccelePort 4r 920	ASIC PCI	114Fh	0026h
PBXDA-BB	DIGI AccelePort 8r 920	ASIC PCI	114Fh	0027h
PBXDA-AC	DIGI AccelePort Xem	ASIC PCI	114Fh	0004h
PBXDA-AC	DIGI AccelePort Xem	ASIC PCI	114Fh	0008h

Syntax for loading PBXDA-xx PCI serial adapter:

```
load digi <name>
```

Example:

```
load digi SERIAL_A
```

By default PBXDA-xx adapter uses first available PCI slot. If instead some particular slot is needed, refer to this section for details of specific placement of PCI peripherals on CHARON Virtual Machine (VM) PCI bus.

In AlphaStation 400 configuration use the following syntax for PBXDA-xx PCI serial adapter loading:

```
load digi SERIAL_A irq_bus = isa
```

Configuration parameters

The PBXDA-xx PCI serial adapter emulation has only one configuration parameter:

host_bus_location

Parameter	host_bus_location
Type	Text String

Value	Establish connection between virtual PBXDA-xx PCI serial adapter and physical DIGI AccelePort PCI serial adapter installed on CHARON host (pass through mode)
	Syntax:
	`load digi <controller name> host_bus_location="PCI bus X, device Y, function Z"`
	where:
	▪ "PCI bus X" corresponds to the PCI bus number of the host DIGI AccelePort PCI serial adapter
	▪ "device Y" corresponds to the PCI bus device number of the host DIGI AccelePort PCI serial adapter
	▪ "function Z" corresponds to the "function" parameter of the the host DIGI AccelePort PCI serial adapter
	Example:
	`load digi SERIAL_A host_bus_location="PCI bus 3, device 1, function 0"`

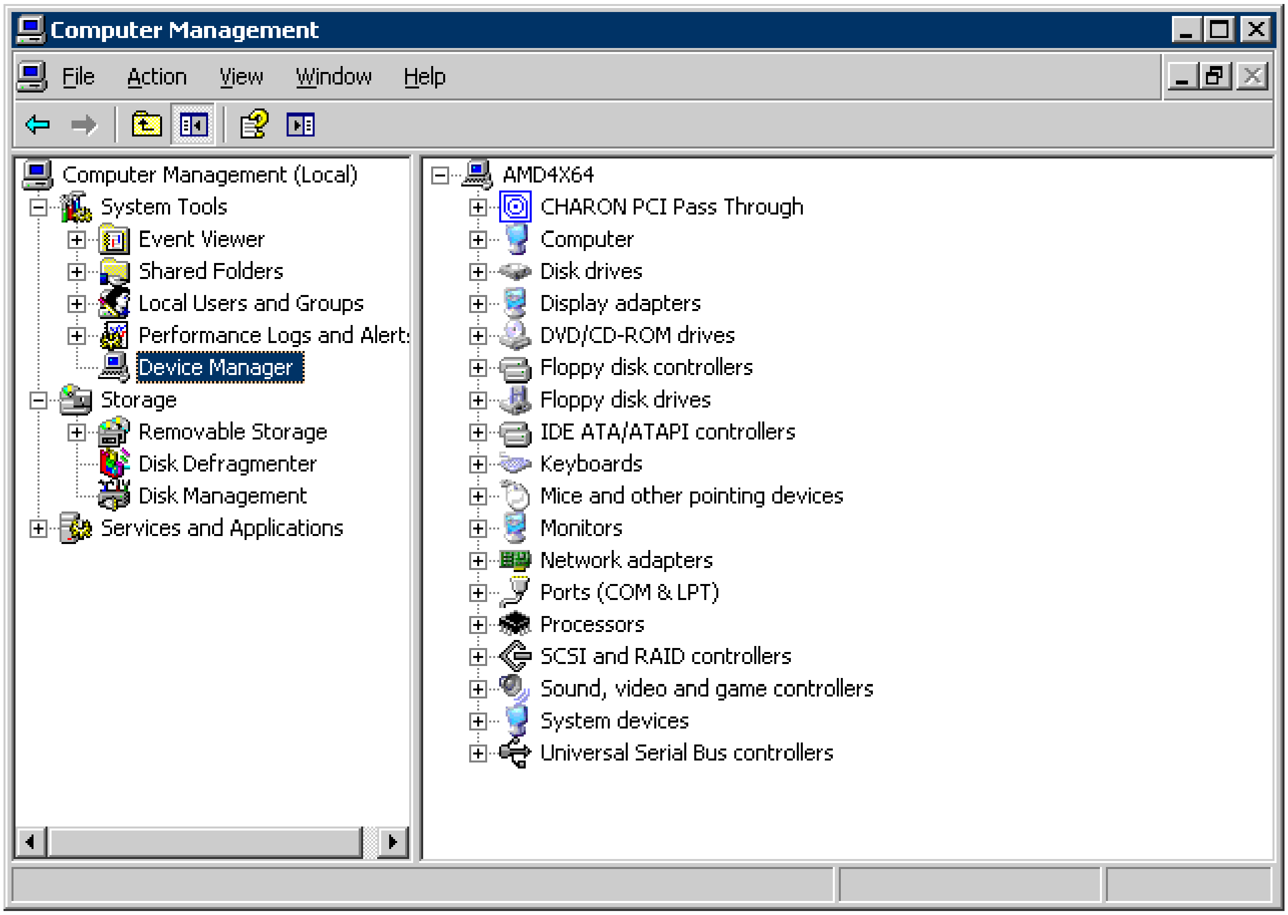
Installation of DIGI AccelePort PCI serial adapter

1. Install the DIGI AccelePort PCI serial adapter to some spare PCI slot of the host system
2. Boot a Windows operating system
3. Open “Computer Management”
4. Select “Device Manager”
5. In the right window select the desired physical serial adapter connected to the system, right-click the mouse button, and the corresponding menu will appear.
6. From the menu select “Update driver…”. Windows will show “Hardware Upgrade Wizard”
7. Select “No, not this time”, click “Next”
8. Select “Install from a list or specific location (Advanced)”, click “Next”
9. Select “Don’t search. I will choose the driver to install”, click “Next”
10. Click “Have Disk…”. Windows shows dialog “Install From Disk”
11. Instead of “A:\”, click “Browse” and select path to the folder in which driver’s INF file is located (typically "C:\Program Files\CHARON\Drivers\DIGI_X.X.0.XXXXX"), select “digi_ppt_amd64.inf”, and click “Open”
12. The “Hardware Upgrade Wizard” should have “CHARON DIGI adapter”. Select it, and click “Next”
13. There will be one or two more dialogs, but they are usual for device driver installation.
14. Reboot the host

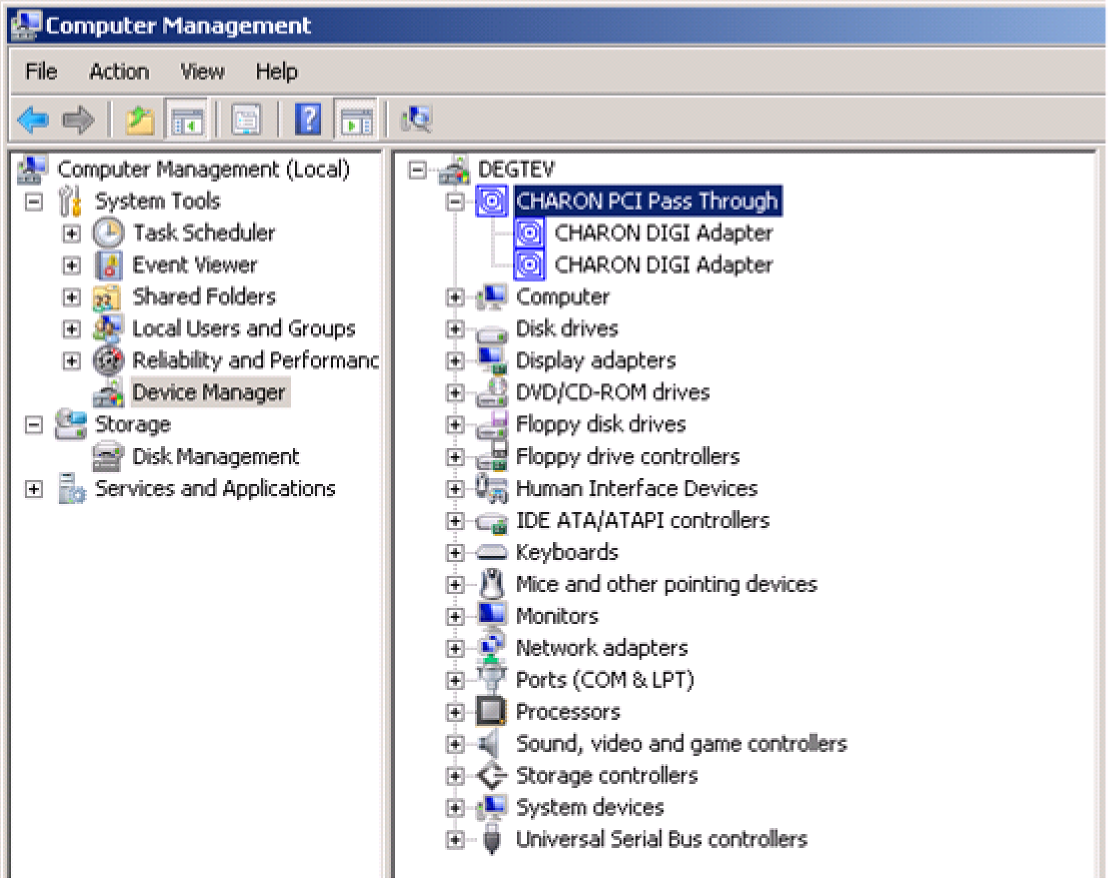
Upon completion, a new device will appear in the device manager with the CHARON logo on it.

Mapping to host DIGI AccelePort PCI serial adapter

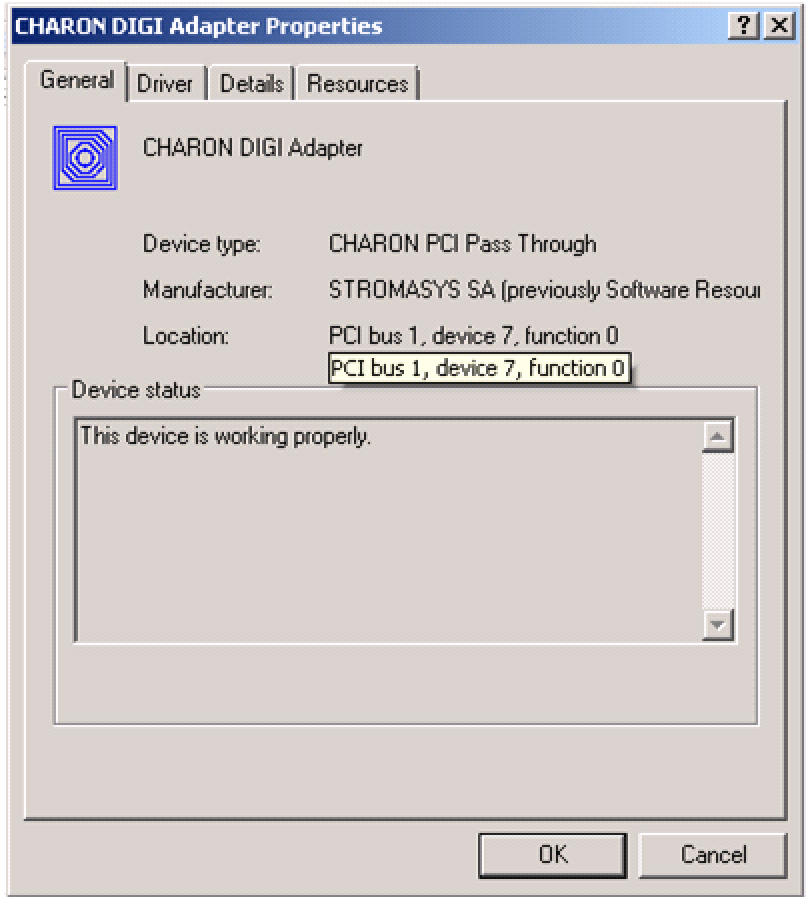
Open “Computer Management” application and select “Device Manager”:



In the right panel select proper physical DIGI adapter:



Open its properties sheet by double-clicking on the selected adapter:



CHARON DIGI Adapter

Device type: CHARON PCI Pass Through

Manufacturer: STROMASYS SA (previously Software Resou

Location: PCI bus 1, device 7, function 0

PCI bus 1, device 7, function 0

Device status

This device is working properly.

The “Location:” on the above picture gives X, Y, and Z for the "host_bus_location" parameter, for example:

```
set PBXDA host_bus_location = "PCI bus 1, device 7, function 0"
```

⚠ Non-US-EN installations of Windows may present “Location:” string in local language, but "host_bus_location" parameter requires English notation, so the words “PCI”, “bus”, “device”, and “function” must be specified in English.

PBXGA graphics card

This functionality is experimental.

Specific note for Charon-AXP version 4.11 204-10: the pbxga_tv application will start automatically with the emulator only if logged in as Administrator user. If logged in as another user, an "Access is denied" popup will appear. As a workaround, please run the pbxga_tv.exe application manually with the specified parameters from a cmd.exe window (as Administrator). For more, see the PBXGA graphics card#application and PBXGA graphics card#Using PBXGA_TV chapters below.

Note: The problem is solved starting with build 204-11 (patch for version 4.11 B204-10 and next versions)

Table of Contents

General description

Charon-AXP supports emulation of PBXGA graphics card(s) by direct virtualization and provides a virtual console for displaying graphics.

The console starts automatically upon starting the emulator. Its resolution is 800 x 600 by default.

Support for AXP graphics is provided with two components:

- emulation of PBXGA PCI graphics adapter and
- emulation of display unit (display unit is emulated with separate executable PBXGA_TV)

Emulated cards

- PBXGA_AA emulates 8bpp graphics card (256 colors, pseudo-color) with 4MB frame buffer - ZLXp-E1
- PBXGA_BA emulates 24bpp graphics card (16M colors, TrueColor) with 16MB frame buffer - ZLXp-E2

Important notes

- PBXGA graphics console cannot be used as the system console. Use OPA0 for this purpose.
- If PBXGA graphics card is going to be used, the resolution of CHARON host screen must be sufficient to provide ability to display the graphics console properly.
- If the window of the PBXGA graphics card console is closed by user or killed as a process, it will not re-appear automatically (💡 the Charon emulator will continue to work). In this situation the user has to restart the pbxga_tv program with the correct parameters, as defined in the configuration file, to restart the graphical display (see PBXGA graphics card#application chapter further).

Virtual PBXGA PCI graphics card settings

Loading virtual PBXGA graphics card

Syntax for loading PBXGA graphics card:

```
load <emulated card> <name>
```

Syntax for loading PBXGA graphics card on AlphaServer 400:

```
load <emulated card> <name> irq_bus = isa
```

Example (more examples below):

```
load PBXGA_BA GYA
```

On Charon startup a window will appear on CHARON host monitor to display graphics.

All the parameters described in the "Placement of peripheral devices on PCI bus" chapter, such as "bus", "device", "function", "irq", "irq_bus" are applicable for PBXGA graphics card.

Configuration parameters

size

Parameter	size
Type	Text string
Value	Predefine the screen size. Possible values: 640x480, 800x600 (default), 1024x768, 1152x900, 1280x1024, 1920x1080 **Example**: `set GYA size = 1024x768`

cpu_draw

Parameter	cpu_draw
Type	boolean
Value	Offloading raster operations. The default value is "true" **Example**: `set GYA cpu_draw = false`

application

Parameter	application
Type	Text string

Value	Specifies the application to open for displaying the graphical console.
	The pbxga_tv executable file is provided for this. It is located in the same folder as the Charon emulator executable file.
	Syntax:
	`pbxga_tv -c <configuration_name> -n <device name(s)>`
	ℹ When several graphics cards are loaded, the *<device name(s)>* have to be separated by a pipe character and the application has to be defined for the latest loaded PBXGA card. See examples below. Note that correct consolidation is only possible when screens are of the same depth and size.
	❓ The *<configuration_name>* must correspond to the one defined in the configuration file or the configuration_name.icfg file located in the virtual machine home directory if it has been defined from a template.
	Example:
	`set GYA application = "pbxga_tv -c DS20 -n GYA"`
	There is no need to supply display size to PBXGA_TV application.
	Using PanoramiX/XINERAMA extensions to DECwindows is also possible (topics of proper configuration OpenVMS/Tru64 are not covered here). These extensions allow to join multiple heads in one big virtual screen. The PBXGA_TV virtual display unit has special notation allowing to consolidate multiple frame buffers in single window (topology is flexible, to some extent):
	• Screens are joined horizontally when separated by "|"
	• Screens are joined vertically when separated by "/"
	Example:
	`set GYD application = "pbxga_tv -c MYCONF -n GYA|GYB/GYC|GYD"`
	💡 It is recommended to create a shortcut on the desktop for the pbxga_tv.exe application with the specified parameters in case the graphical display is closed and needs to be reopened.

Using PBXGA_TV

The PBXGA_TV display unit starts in passive mode. It displays content of frame buffer but does not react on keyboard input and/or mouse movements. In order to switch it to active mode, place mouse cursor with the window and click once. When PBXGA_TV is active, mouse cursor is stuck to PBXGA_TV window. Press CTRL+SHIFT+R to release.

The PBXGA_TV display unit starts in Window Mode. Use CTRL+SHIFT+F to toggle Full Screen Mode.

The PBXGA_TV application uses libsdl to draw its window. Copy of SDL.dll comes (version 1.2-15, unmodified) with Charon-AXP for Windows.

ⓘ Note, that using PBXGA_TV in VNC sessions is not suggested. There are known issues with mouse cursor.

Running PBXGA_TV remotely is not supported.

⚠ The PBXGA_TV can only consolidate screens belonging to the same virtual machine. It can't consolidate screens belonging to different machines.

To create a shortcut on the desktop, you can use one of the following methods:

- Navigate to the Charon installation folder and find the pbxga_tv.exe ("C:\Program Files\CHARON\Build_20410\x64" usually) then create a shortcut, open its properties and add the parameters (-c ... -n ..) in the Shortcut/Target field, **or**
- open a Powershell window and run the following commands (update the $cfgnam and $pbxga values, lines 1 and 2, for your own configuration):

```
$cfgnam = "DS20withPBXGA"
$pbxga = "GYA"
$TargetFile = "C:\Program Files\CHARON\Build_20410\x64\pbxga_tv.exe"
$ShortcutFile = "${env:USERPROFILE}\Desktop\PBXGA $cfgnam $pbxga.lnk"
$WScriptShell = New-Object -ComObject WScript.Shell
$Shortcut = $WScriptShell.CreateShortcut($ShortcutFile)
$Shortcut.Arguments = "-c $cfgnam -n $pbxga"
$Shortcut.TargetPath = $TargetFile
$Shortcut.Save()
```

Once the shortcut is created, check the "Run as administrator" box in its advanced properties.

Examples

Example 1: 8-bpp graphics on AlphaServer 400

```
...
set session hw_model = AlphaServer_400
set session configuration_name = AS400
...
load PBXGA_AA GYA irq_bus = isa size = 1024x768
set GYA application = "pbxga_tv -c AS400 -n GYA"
```

Example 2: 24-bpp graphics on AlphaServer DS20

```
...
set session hw_model = AlphaServer_DS20
set session configuration_name = DS20
...
load PBXGA_BA GYA size = 1280x1024
set GYA application = "pbxga_tv -c DS20 -n GYA"
```

Example 3: two display units, running independently

```
...
set session configuration_name = MYCONF
...
load PBXGA_BA GYA
set GYA application = "pbxga_tv -c MYCONF -n GYA"
load PBXGA_BA GYB size = 1280x1024
set GYB application = "pbxga_tv -c MYCONF -n GYB"
```

In the above example, display units have different resolutions.

Example 4: dual head display, 2x 1280x1024

```
...
set session hw_model = AlphaServer_ES45
set session configuration_name = ES45
...
load PBXGA_BA GYA size = 1280x1024
load PBXGA_BA GYB size = 1280x1024
set GYB application = "pbxga_tv -c ES45 -n GYA|GYB"
```

Example 5: triple head display, 3x 1280x1024

```
...
set session hw_model = AlphaServer_ES45
set session configuration_name = ES45
...
load PBXGA_BA GYA size = 1280x1024
load PBXGA_BA GYB size = 1280x1024
load PBXGA_BA GYC size = 1280x1024
set GYC application = "pbxga_tv -c ES45 -n GYA|GYB|GYC"
```

Example 6: Two screens consolidated in a column in one window

```
...
set session configuration_name = MYCONF
...
set session configuration_name = MYCONF
load PBXGA_BA GYA size = 1280x1024
load PBXGA_BA GYB size = 1280x1024
set GYB application = "pbxga_tv -c MYCONF -n GYA/GYB"
```

Configuration File Templates

The template files can be found after product installation in the following directory:

```
C:\Program Files\CHARON\Build_xxxxx\x64\
```

Where **`xxxxx`** is the build number of the desired Charon-AXP version.

The template file names have the form **`<Alpha-model>.cfg`**.

CHARON-AXP for Windows deinstallation

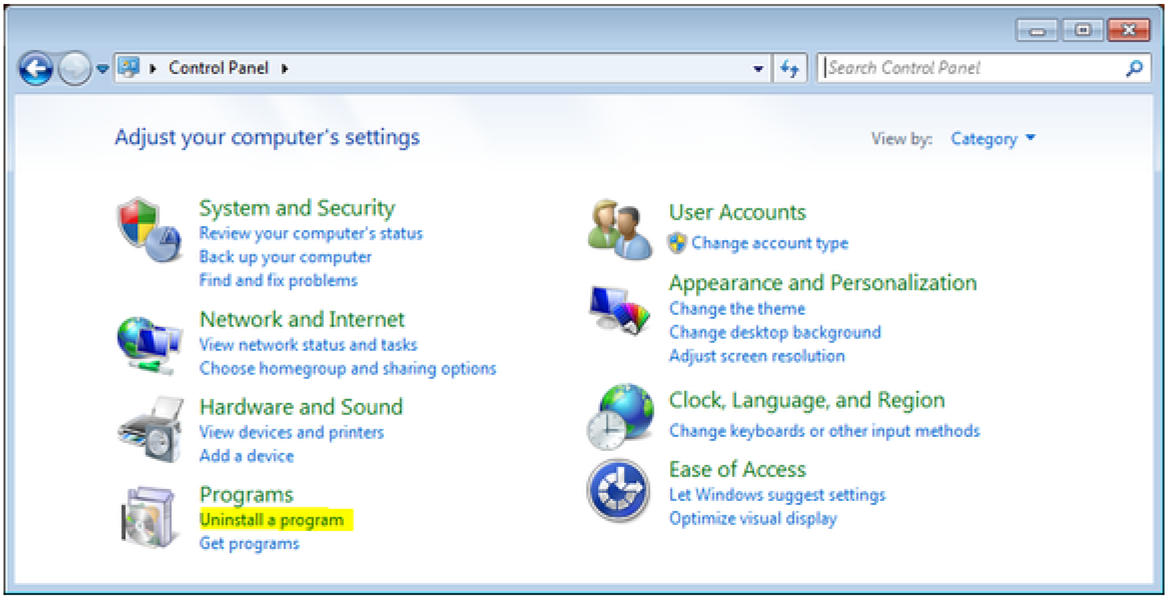
Deinstallation procedure

Perform a clean shutdown of the running guests, open the CHARON Virtual Machine Manager, stop all running CHARON-AXP virtual machines (VM) and remove them.

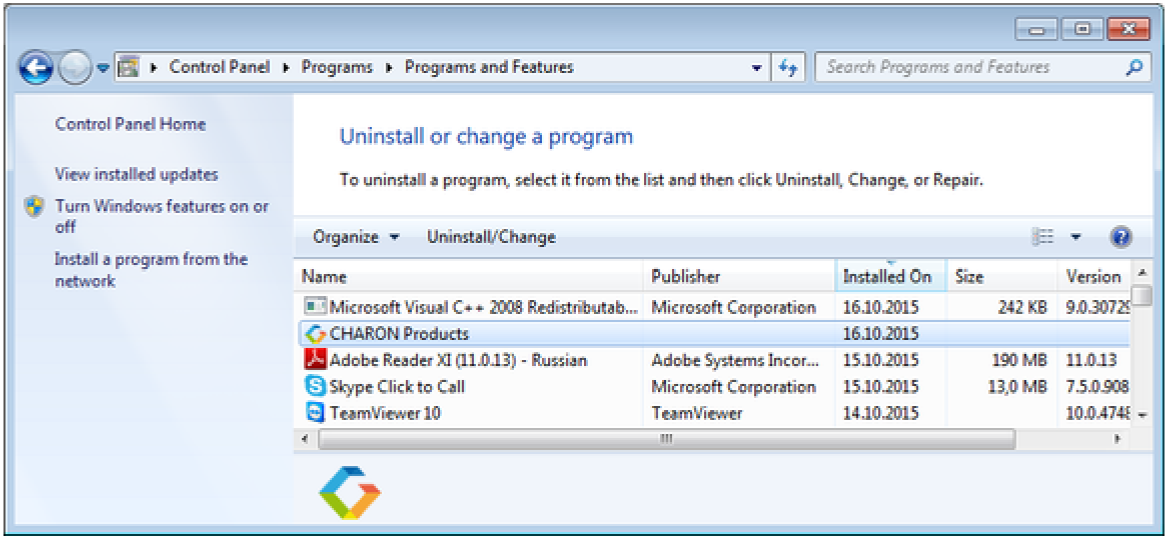
In case of a complete deinstallation:

- Remove all the CHARON virtual machines.
- Using the "Network Control Center" utility that can be opened from the "CHARON Virtual Machine Manager", release all the network interfaces back to the CHARON host.
- Remove any specific CHARON drivers installed manually (if any).

Open up the "Control Panel" and select "Uninstall a program":



In the list of applications, double click on "CHARON Products":

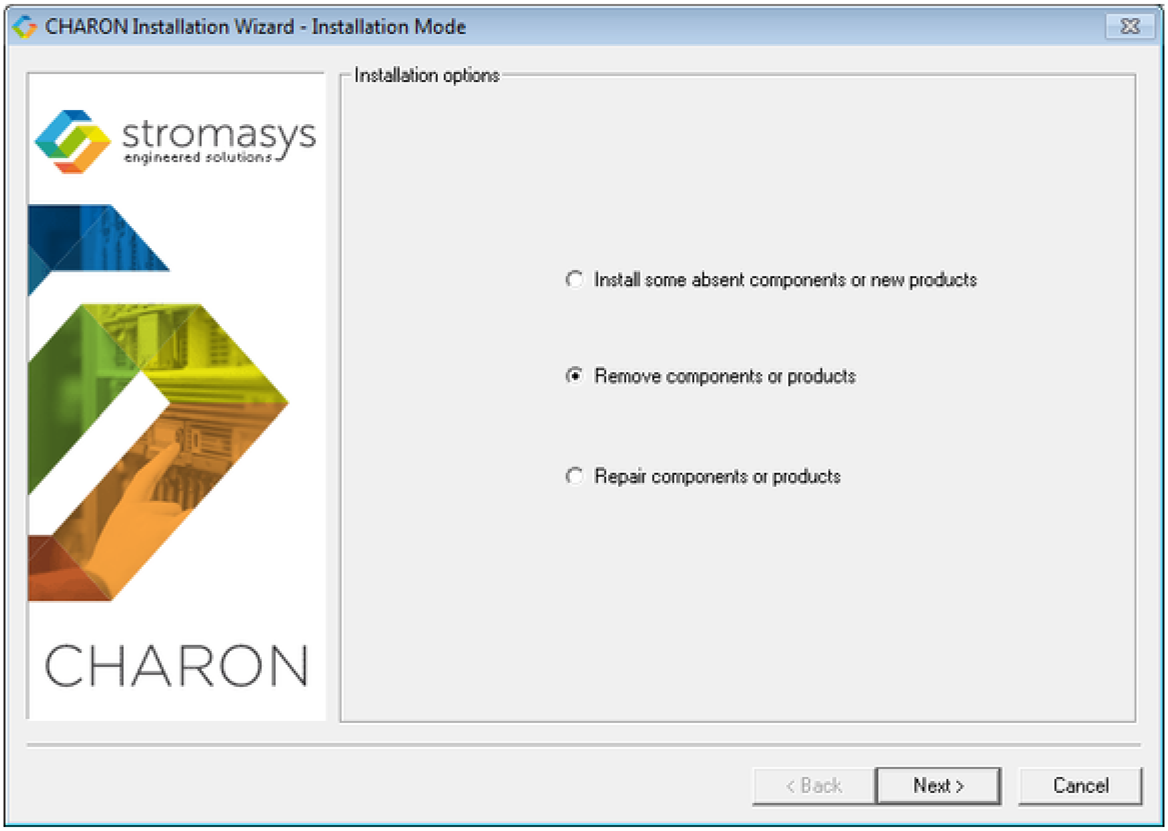

Control Panel ▸ Programs ▸ Programs and Features
Search Programs and Features
Control Panel Home
View installed updates
Turn Windows features on or off
Install a program from the network
Organize
Uninstall/Change

Uninstall or change a program

To uninstall a program, select it from the list and then click Uninstall, Change, or Repair.

Name	Publisher	Installed On	Size	Version
Microsoft Visual C++ 2008 Redistributab...	Microsoft Corporation	16.10.2015	242 KB	9.0.30729
CHARON Products		16.10.2015		
Adobe Reader XI (11.0.13) - Russian	Adobe Systems Incor...	15.10.2015	190 MB	11.0.13
Skype Click to Call	Microsoft Corporation	15.10.2015	13,0 MB	7.5.0.908
TeamViewer 10	TeamViewer	14.10.2015		10.0.474

In the appearing CHARON installation dialog, select "Remove components or products" and press the "Next" button:


CHARON Installation Wizard - Installation Mode

Installation options

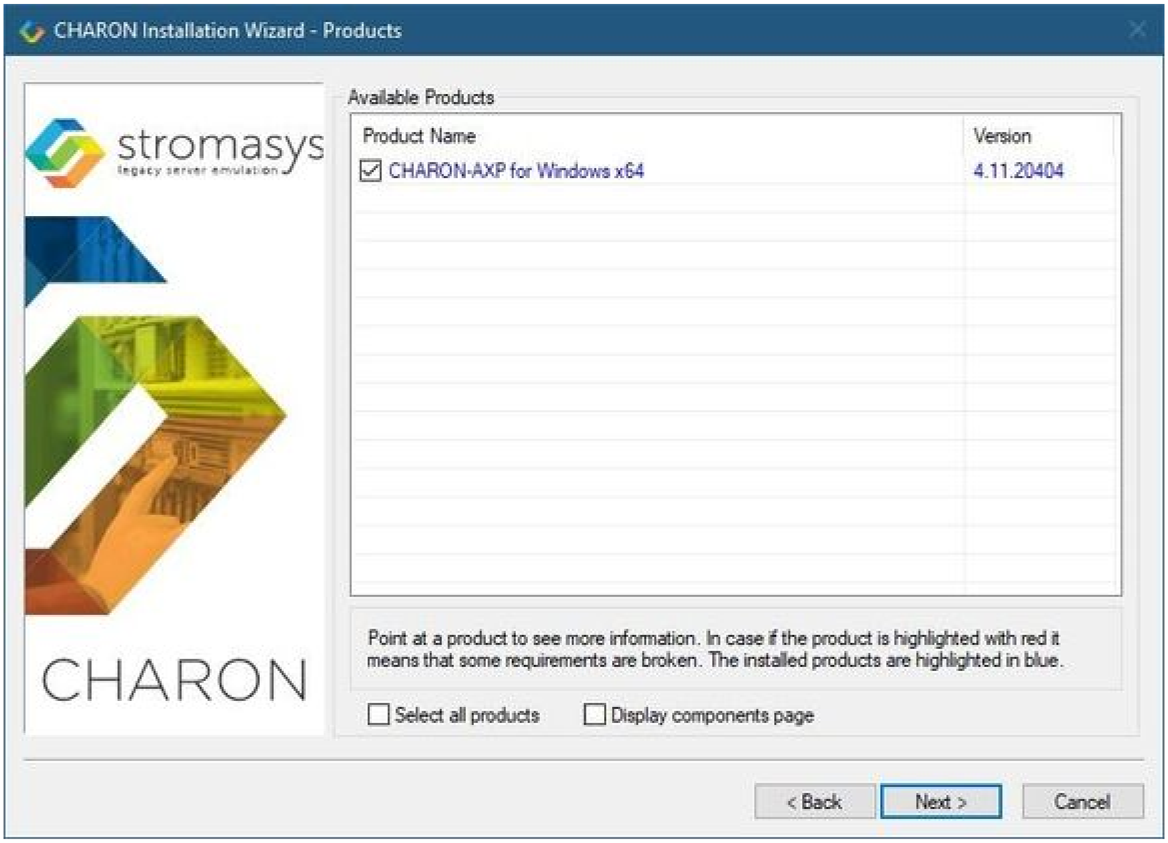
○ Install some absent components or new products

● Remove components or products

○ Repair components or products

In the following dialog, select the CHARON product to be removed (or changed). Check "Select all the products" for deinstallation (or changing) of all installed CHARON products. Do not select "Display components page" unless you need to remove just some components of the CHARON product.

Press the "Next" button:



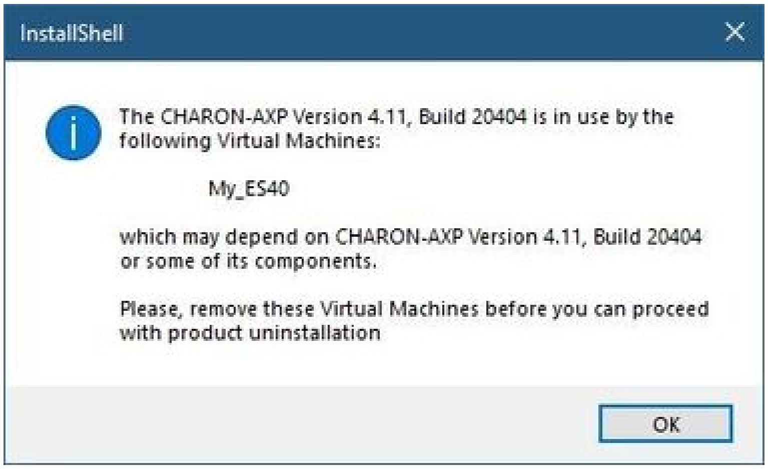
CHARON Installation Wizard - Products

Available Products

Product Name	Version
☑ CHARON-AXP for Windows x64	4.11.20404

Point at a product to see more information. In case if the product is highlighted with red it means that some requirements are broken. The installed products are highlighted in blue.

If there are some CHARON virtual machines that use this particular version of CHARON the following error message will be displayed:



InstallShell

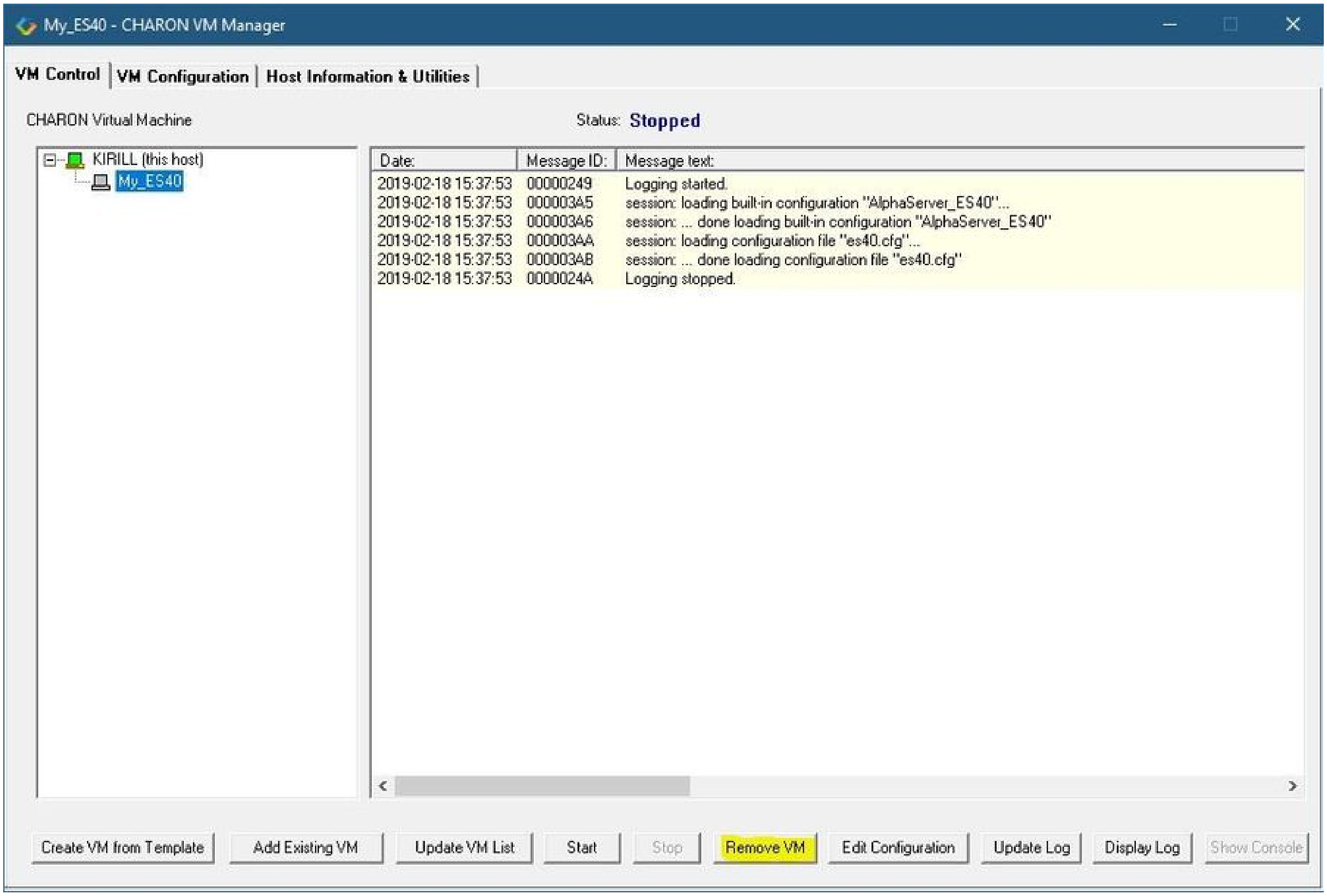
The CHARON-AXP Version 4.11, Build 20404 is in use by the following Virtual Machines:

My_ES40

which may depend on CHARON-AXP Version 4.11, Build 20404 or some of its components.

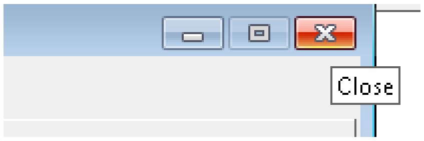
Please, remove these Virtual Machines before you can proceed with product uninstallation

In this case, open the "CHARON Virtual Machine Manager", select the CHARON Machine(s) listed above and either press the "Remove VM" button or open the VM configuration tab and set this virtual machine to use another Charon version if more than one version is installed:

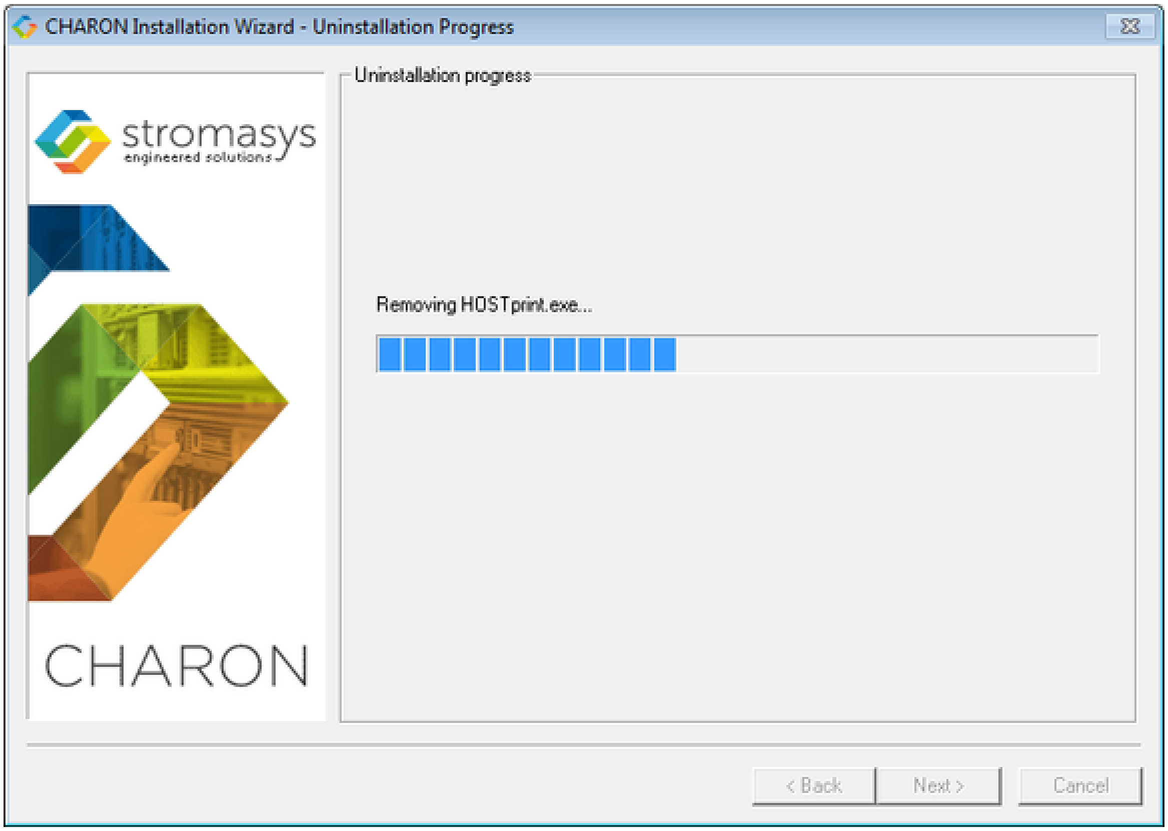


```
2019-02-18 15:37:53   00000249   Logging started.
2019-02-18 15:37:53   000003A5   session: loading built-in configuration "AlphaServer_ES40"...
2019-02-18 15:37:53   000003A6   session: ... done loading built-in configuration "AlphaServer_ES40"
2019-02-18 15:37:53   000003AA   session: loading configuration file "es40.cfg"...
2019-02-18 15:37:53   000003AB   session: ... done loading configuration file "es40.cfg"
2019-02-18 15:37:53   0000024A   Logging stopped.
```

Close the "CHARON Virtual Machine Manager" with a click on the "x" button and confirm you want to exit:



Press the "Next" button, the uninstallation process will begin:

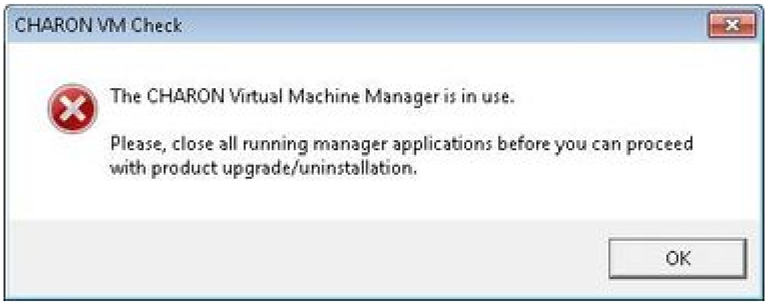


CHARON Installation Wizard - Uninstallation Progress

Uninstallation progress

Removing HOSTprint.exe...

If the "CHARON Virtual Machine Manager" is still running the following dialog will be displayed:



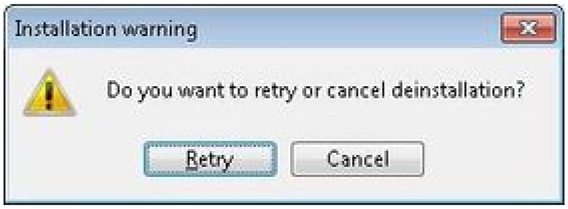
CHARON VM Check

The CHARON Virtual Machine Manager is in use.

Please, close all running manager applications before you can proceed with product upgrade/uninstallation.

Do not close the uninstallation dialog but exit from "CHARON Virtual Machine Manager" the way described above.

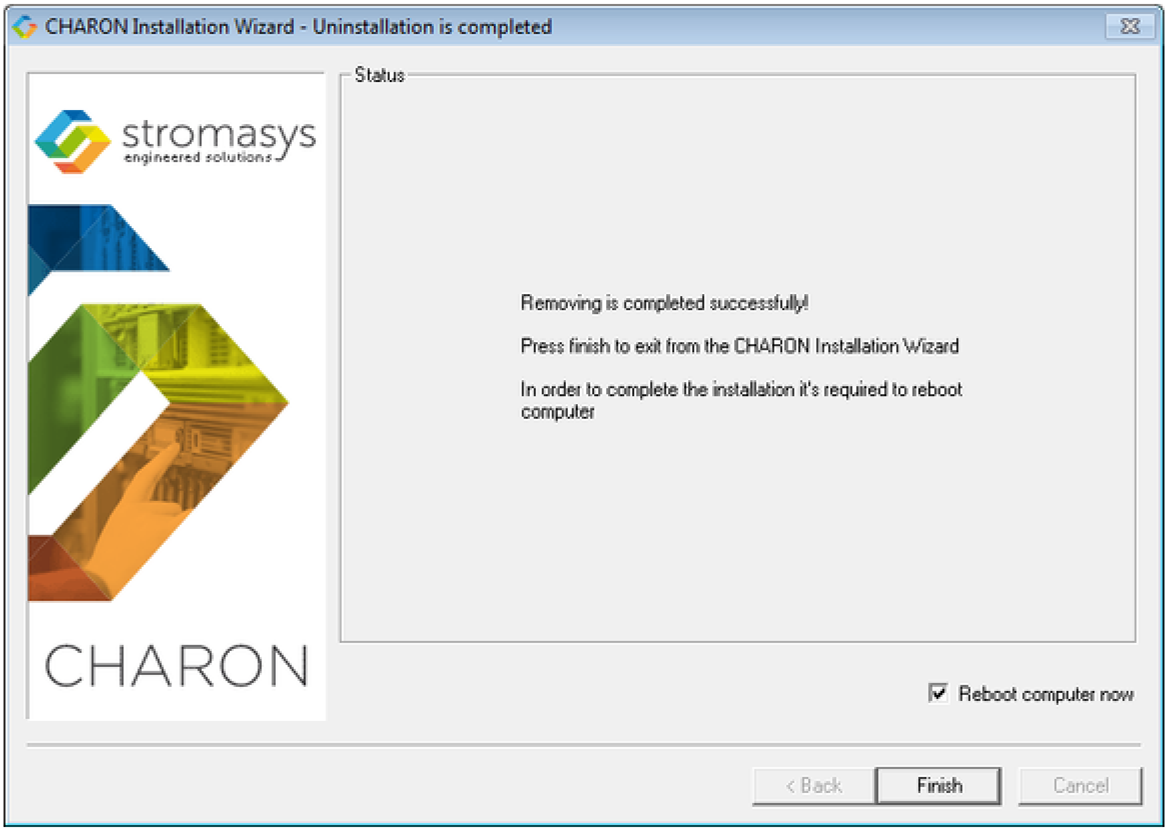
Press the "OK" button then the "Retry" one in the appearing dialog:



Installation warning

Do you want to retry or cancel deinstallation?

Once uninstallation (or changing) is done, the following dialog will be displayed:



CHARON Installation Wizard - Uninstallation is completed

Status

Removing is completed successfully!

Press finish to exit from the CHARON Installation Wizard

In order to complete the installation it's required to reboot computer

☑ Reboot computer now

Keep the "Reboot computer now" selected and press the "Finish" button to reboot the CHARON host.

Appendixes

Contents

Log monitoring - logmond best practice

Table of contents

Description

The Log Monitor & Dispatcher, known as LOGMOND, is a special program which monitors a guest LOG file produced by Charon and executes a customized script when it detects removal of a license.

It runs in the background (as a program or as a service) and periodically scans a specified LOG file. When it detects a message with the code 00000424 ("Detected removal of a license") or 0000002A or 00000351 for backward compatibility, it submits the `nolicense.bat` procedure (this BAT file must be created manually). It also invokes the `license_changed.bat` script, if it exists, when it detects a message with code 0000040B / "License has changed. License detected and online".

The Log Monitor & Dispatcher service is installed as `EmulatorLogMonitor`. By default it is installed in such a way that requires explicit actions to be started (either through a command line interface or using the standard ways of service management). For unattended execution, change the service's configuration so that Windows starts the service automatically.

The tool requires a specific file "`nolicense.bat`", and optionally a file named "`license_changed.bat`", containing some specific instructions to be taken in situation of license absence or license change. These files must be created in the virtual machine's home directory.

⚠ The user action file will not invoke interactive applications as it may run in an environment where interactive services do not work, for example: when “Log Monitor & Dispatcher” is installed as a service.

✓ **A new utility is available for download that offers more monitoring capabilites than logmond**.

Please see CHARON on Windows - Charon Log Monitor and contact us to get the download link if needed.

Recommendations

- Create a specific folder to hold the Charon configuration and log files, the `nolicense.bat` file, the `license_changed.bat` if needed, and any specific action script you would add.
- Use the rotating log files mechanism or use a single log file with "append" mode.
- Install and start the Charon log monitor service, named "`EmulatorLogMonitor`": doing so you will not have to start `logmond` manually in interactive mode and newly added virtual machines services will be automatically taken into account. Use interactive mode for tests and debugging purpose.

Notes

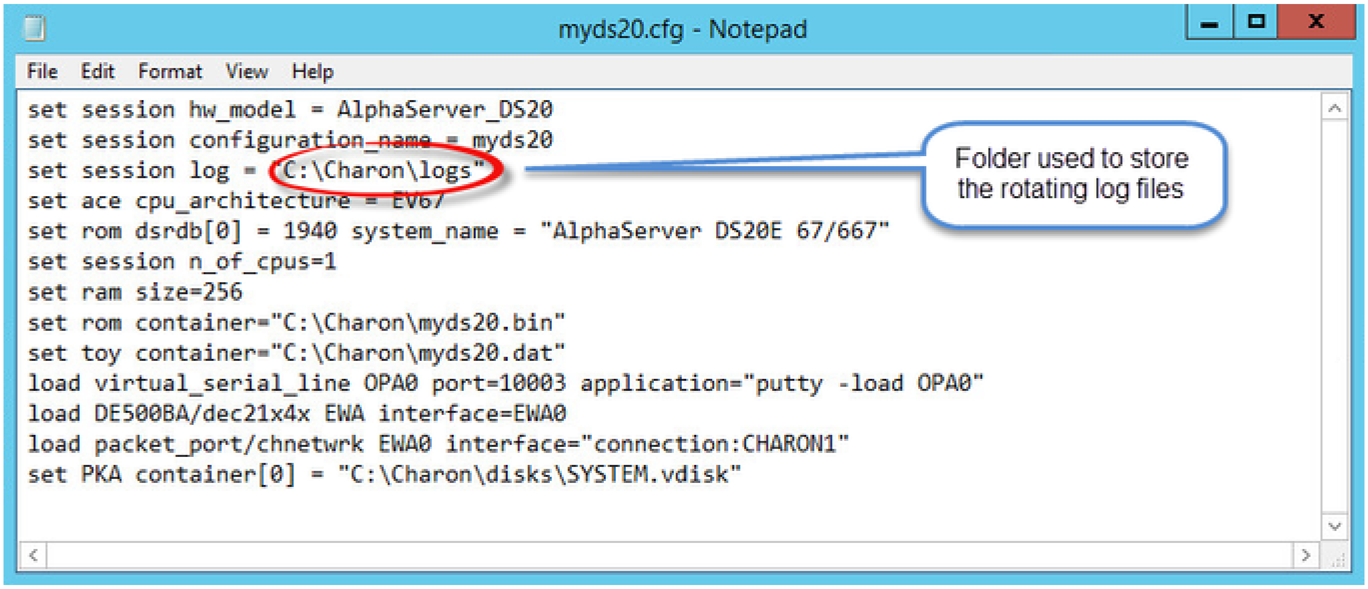
- The log monitor utility will detect the absence or change of the dongle only at a specified interval, called the license check interval and defined by the license. Its default is set to 1 hour. In case you need to perform tests, you can send us a fresh C2V file and ask for an update to reduce this interval.

- Once started the logmond utility will lock the virtual machine log file, you will not be able to move it, copy it or delete it. You will however be able to open it using the notepad utility or other log editors like `baretail` for example which will allow you to have a continuous view of the log with highlights (see: Charon Log files - Using baretail on Windows for syntax highlighting) or `vim` (see: Charon Log files - Using vim for syntax highlighting on Linux and Windows).

Service setup example

Example below is based on a Charon-AXP V4.9 B19402 running on a Windows 2012 R2 server and emulating an AlphaServer DS20. All files (configuration, bin, rom and `nolicense.bat`) are located in `C:\Charon`, rotating log files are located in `C:\Charon\logs`

The configuration file is defined as follows:



```
set session hw_model = AlphaServer_DS20
set session configuration_name = myds20
set session log = "C:\Charon\logs"
set ace cpu_architecture = EV67
set rom dsrdb[0] = 1940 system_name = "AlphaServer DS20E 67/667"
set session n_of_cpus=1
set ram size=256
set rom container="C:\Charon\myds20.bin"
set toy container="C:\Charon\myds20.dat"
load virtual_serial_line OPA0 port=10003 application="putty -load OPA0"
load DE500BA/dec21x4x EWA interface=EWA0
load packet_port/chnetwrk EWA0 interface="connection:CHARON1"
set PKA container[0] = "C:\Charon\disks\SYSTEM.vdisk"
```

Folder used to store the rotating log files

Setup and start your virtual machine.

- Start the "Charon Virtual Machines Manager"
- Create your virtual machine and start it

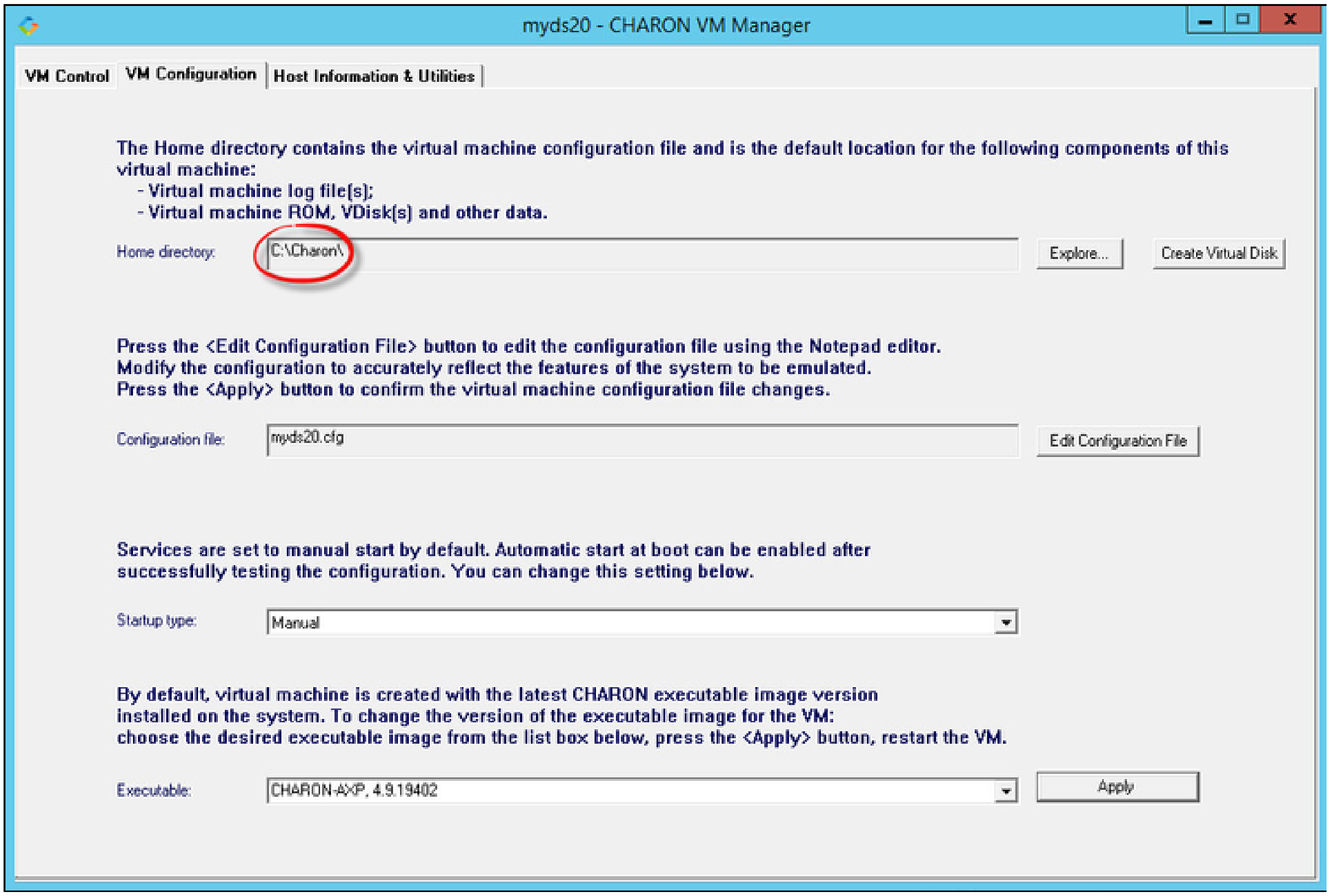
See Running and managing Charon-AXP for Windows or Running and managing Charon-VAX and Charon-PDP for Windows for details.

Install and start the log monitor service

- First, prepare a `nolicense.bat` file. See nolicense.bat considerations chapter further for more information if needed.

The `nolicense.bat` file has to be placed in the virtual machine "Home directory". To locate this "Home directory" open the "Charon Virtual Machines manager" utility, select the virtual machine and switch to the "VM Configuration" tab

Example:



The Home directory contains the virtual machine configuration file and is the default location for the following components of this virtual machine:
- Virtual machine log file(s);
- Virtual machine ROM, VDisk(s) and other data.

Home directory: C:\Charon\

Press the <Edit Configuration File> button to edit the configuration file using the Notepad editor.
Modify the configuration to accurately reflect the features of the system to be emulated.
Press the <Apply> button to confirm the virtual machine configuration file changes.

Configuration file: myds20.cfg

Services are set to manual start by default. Automatic start at boot can be enabled after successfully testing the configuration. You can change this setting below.

Startup type: Manual

By default, virtual machine is created with the latest CHARON executable image version installed on the system. To change the version of the executable image for the VM:
choose the desired executable image from the list box below, press the <Apply> button, restart the VM.

Executable: CHARON-AXP, 4.9.19402

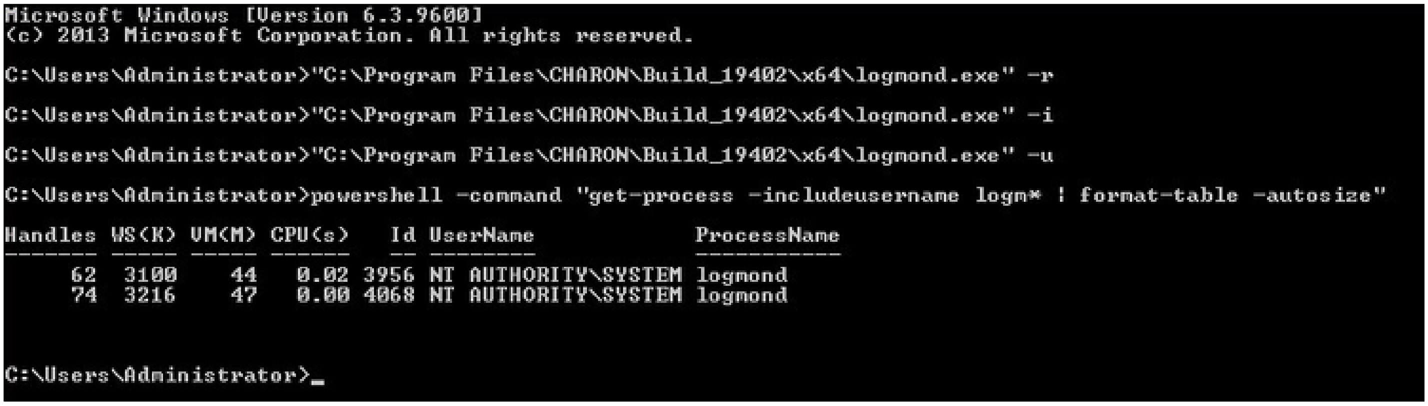
Use the notepad for example to create the `nolicense.bat` script. We will here add a line in a log file:

```
echo %date% %time% dongle removed >>C:\Charon\logmond.log
```

The `license_changed.bat` script can also contain quite the same line:

```
echo %date% %time% license has changed >>C:\Charon\logmond.log
```

- Install the log monitoring service and start it:
  ⚠ **These command lines must be executed as an administrator in order to manage the services**

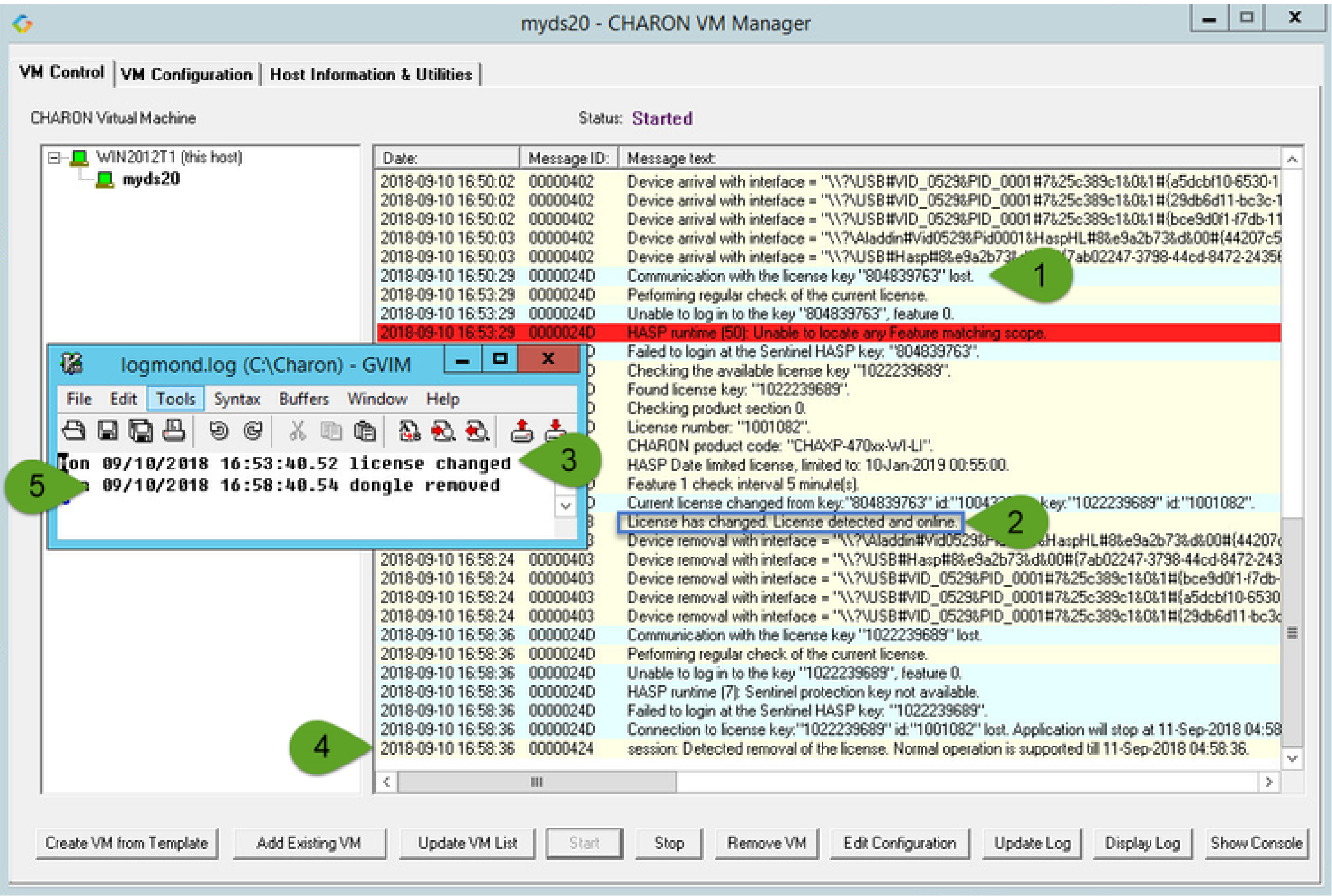


```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>"C:\Program Files\CHARON\Build_19402\x64\logmond.exe" -r

C:\Users\Administrator>"C:\Program Files\CHARON\Build_19402\x64\logmond.exe" -i

C:\Users\Administrator>"C:\Program Files\CHARON\Build_19402\x64\logmond.exe" -u

C:\Users\Administrator>powershell -command "get-process -includeusername logm* | format-table -autosize"

Handles WS(K) VM(M) CPU(s)   Id UserName            ProcessName
------- ----- ----- ------   -- --------            -----------
     62  3100    44   0.02 3956 NT AUTHORITY\SYSTEM logmond
     74  3216    47   0.00 4068 NT AUTHORITY\SYSTEM logmond


C:\Users\Administrator>_
```

- **Notes**:
  - The "`logmond -r`" command removes the service if any
  - The "`logmond -i`" command installs the service

- The "`logmond -u`" command starts the service
- The powershell commands shows 2 logmond processes, one scans for new services, the other is monitoring the log file of the current service (myds20)
- The `EmulatorLogMonitor` service running the `logmond` processes will discover already installed Charon virtual machines services. If a new virtual machine service is added, it's log file will be automatically discovered and monitored.
- The logmond.exe file is located by default in the "`C:\Program Files\Charon\Build_19402\x64`" folder for Charon-AXP/VAX/PDP V4.9, "`C:\Program Files\Charon\Build_20200\x64`" for V4.10.

- In this test, two dongles were connected, the active one has been removed (**1**) producing the license changed message (**2** and **3**) then the last active dongle has been removed (**4** and **5**).
  During the tests the license check interval was set to 5 minutes therefore, when the dongle is disconnected, a message is sent telling there are only 4 minutes remaining before the DS20 stops



```
Date:                Message ID:  Message text:
2018-09-10 16:50:02  00000402     Device arrival with interface = "\\?\USB#VID_0529&PID_0001#7&25c389c1&0&1#{a5dcbf10-6530-1
2018-09-10 16:50:02  00000402     Device arrival with interface = "\\?\USB#VID_0529&PID_0001#7&25c389c1&0&1#{29db6d11-bc3c-1
2018-09-10 16:50:02  00000402     Device arrival with interface = "\\?\USB#VID_0529&PID_0001#7&25c389c1&0&1#{bce9d0f1-f7db-11
2018-09-10 16:50:03  00000402     Device arrival with interface = "\\?\Aladdin#Vid0529&Pid0001&HaspHL#8&e9a2b73&d&00#{44207c5
2018-09-10 16:50:03  00000402     Device arrival with interface = "\\?\USB#Hasp#8&e9a2b73       7ab02247-3798-44cd-8472-2435
2018-09-10 16:50:29  0000024D     Communication with the license key "804839763" lost.
2018-09-10 16:53:29  0000024D     Performing regular check of the current license.
2018-09-10 16:53:29  0000024D     Unable to log in to the key "804839763", feature 0.
2018-09-10 16:53:29  0000024D     HASP runtime (50): Unable to locate any Feature matching scope.
                              D   Failed to login at the Sentinel HASP key: "804839763".
                              D   Checking the available license key "1022239689".
                              D   Found license key: "1022239689".
                              D   Checking product section 0.
                              D   License number: "1001082".
                              D   CHARON product code: "CHAXP-470xx-WI-LI".
                              D   HASP Date limited license, limited to: 10-Jan-2019 00:55:00.
                              D   Feature 1 check interval 5 minute(s).
                              D   Current license changed from key:"804839763" id:"10043       key:"1022239689" id:"1001082".
                              3   License has changed. License detected and online.
                              3   Device removal with interface = "\\?\Aladdin#Vid0529&P       &HaspHL#8&e9a2b73&d&00#{44207c
2018-09-10 16:58:24  00000403     Device removal with interface = "\\?\USB#Hasp#8&e9a2b73&d&00#{7ab02247-3798-44cd-8472-243
2018-09-10 16:58:24  00000403     Device removal with interface = "\\?\USB#VID_0529&PID_0001#7&25c389c1&0&1#{bce9d0f1-f7db-
2018-09-10 16:58:24  00000403     Device removal with interface = "\\?\USB#VID_0529&PID_0001#7&25c389c1&0&1#{a5dcbf10-6530
2018-09-10 16:58:24  00000403     Device removal with interface = "\\?\USB#VID_0529&PID_0001#7&25c389c1&0&1#{29db6d11-bc3c
2018-09-10 16:58:36  0000024D     Communication with the license key "1022239689" lost.
2018-09-10 16:58:36  0000024D     Performing regular check of the current license.
2018-09-10 16:58:36  0000024D     Unable to log in to the key "1022239689", feature 0.
2018-09-10 16:58:36  0000024D     HASP runtime (7): Sentinel protection key not available.
2018-09-10 16:58:36  0000024D     Failed to login at the Sentinel HASP key: "1022239689".
2018-09-10 16:58:36  0000024D     Connection to license key:"1022239689" id:"1001082" lost. Application will stop at 11-Sep-2018 04:58
2018-09-10 16:58:36  00000424     session: Detected removal of the license. Normal operation is supported till 11-Sep-2018 04:58:36.
```

```
Mon 09/10/2018 16:53:40.52 license changed
  n 09/10/2018 16:58:40.54 dongle removed
```

Running the log monitor from the command line

The "`logmond`" utility has several parameters available. Use the "`logmond -h`" command to have a complete list.

Starting in background mode

In order to **start** the Log Monitor & Dispatcher as a background application:

1. Open "cmd.exe" from the "Start" menu.
2. Change current directory to the folder (presumably) containing the Charon configuration file.
3. Start the Log Monitor & Dispatcher using the following command line as an example:

   For single log file:

   ```
   ...> "C:\Program Files\Charon\Build_19402\x64\logmond" C:\my_Charon.log
   ```

   For rotating log file:

   ```
   ...> "C:\Program Files\Charon\Build_19402\x64\logmond" -l <log-directory> -p <log-prefix>
   ```

   where:

- `<log-directory>` is the directory where the rotating log files are stored
- `<log-prefix>` is the same as the "configuration_name" value in corresponding Charon configuration file (or "hw_model", if "configuration_name" is not specified).

Example:

```
...> "C:\Program Files\Charon\Build_19402\x64\logmond" -l "C:\My Charon logs" -p "MY_VAX"
```

To stop the Log Monitor & Dispatcher application, open the Task Manager, find the "`logmond.exe`" process and terminate it.

Debugging

To diagnose LOGMOND using trace feature:

1. Stop LOGMOND service using "`--stop`" (or "`-d`" or "`--down`"). This is not absolutely necessary though, as multiple running LOGMOND utilities do not affect each other.

2. Start "almost-as-a-service" LOGMOND with diagnostic trace using two command line switches: "`-e`" and "`-t`". This starts master task of LOGMOND which enumerates all Charon services and for each of them invokes slave task of LOGMOND. The master task of LOGMOND looks for installing or removal Charon services.

   a. When Charon service is installed, master creates new slave.
   b. When Charon service is removed, slave is terminated to. The slave task monitors LOG of particular Charon service.

   ℹ The "`-t`" switch enables trace windows, so you can see the process of monitoring.

3. To terminate "almost-as-a-service" LOGMOND master and all its slaves, bring focus to master's trace window, and press "`Ctrl-C`" ([X] button closes master task, but slaves continue to run unmastered).

The LOGMOND still supports backward compatibility mode, when it does NOT monitor ALL Charon services, but only monitors a particular folder or even a particular file. So, if you want to diagnose how it monitors a file, you can use the same "`-t`" option:

```
C:\...>logmond.exe -t -l <path to Charon log file>
```

If you want to diagnose monitoring rotating log:

```
C:\...>logmond.exe -t -l <path to Charon log directory> -p <log prefix>
```

If you want to diagnose monitoring particular service:

```
C:\...>logmond.exe -t -s <service name>
```

When you install LOGMOND to run as a service it implicitly runs with "`-e`" option but without "`-t`". Alternatively you may install LOGMOND as a service with explicitly specified name of log file (for backward compatibility)

You better use "`-e`" together with "`-t`" to diagnose monitoring Charon services. Unless you run instances with Launcher, in which case you will have to invoke LOGMOND with "`-l`" with "`-t`" options.

nolicense.bat considerations

LOGMOND always looks for `nolicense.bat` and `license_changed.bat` files in the current working directory. What is current working directory for LOGMOND depends on how it is invoked.

If you run LOGMOND with just path to log file, the LOGMOND inherits current working directory from command prompt it is invoked from. So, before you run LOGMOND this way, change prompt's working directory to where you keep your NOLICENSE.BAT file.

If you run LOGMOND with name of Charon service, the LOGMOND changes its current working directory to the Charon service's Home directory. So, keep the `nolicense.bat` and `license_changed.bat` files in the virtual machine's home directory.

If you run LOGMOND with "`-e`" option, each slave receives a particular name of Charon service, and therefore changes to the service's Home directory. So, keep the `nolicense.bat` and `license_changed.bat` files in the virtual machine's home directory.

Stromasys highly recommends to use logmond as a service, installing it with the "`-i`" option then starting it with "`-u`".

Alerts customization examples

You will find below some examples of alerts that can be sent from the `nolicense.bat` file

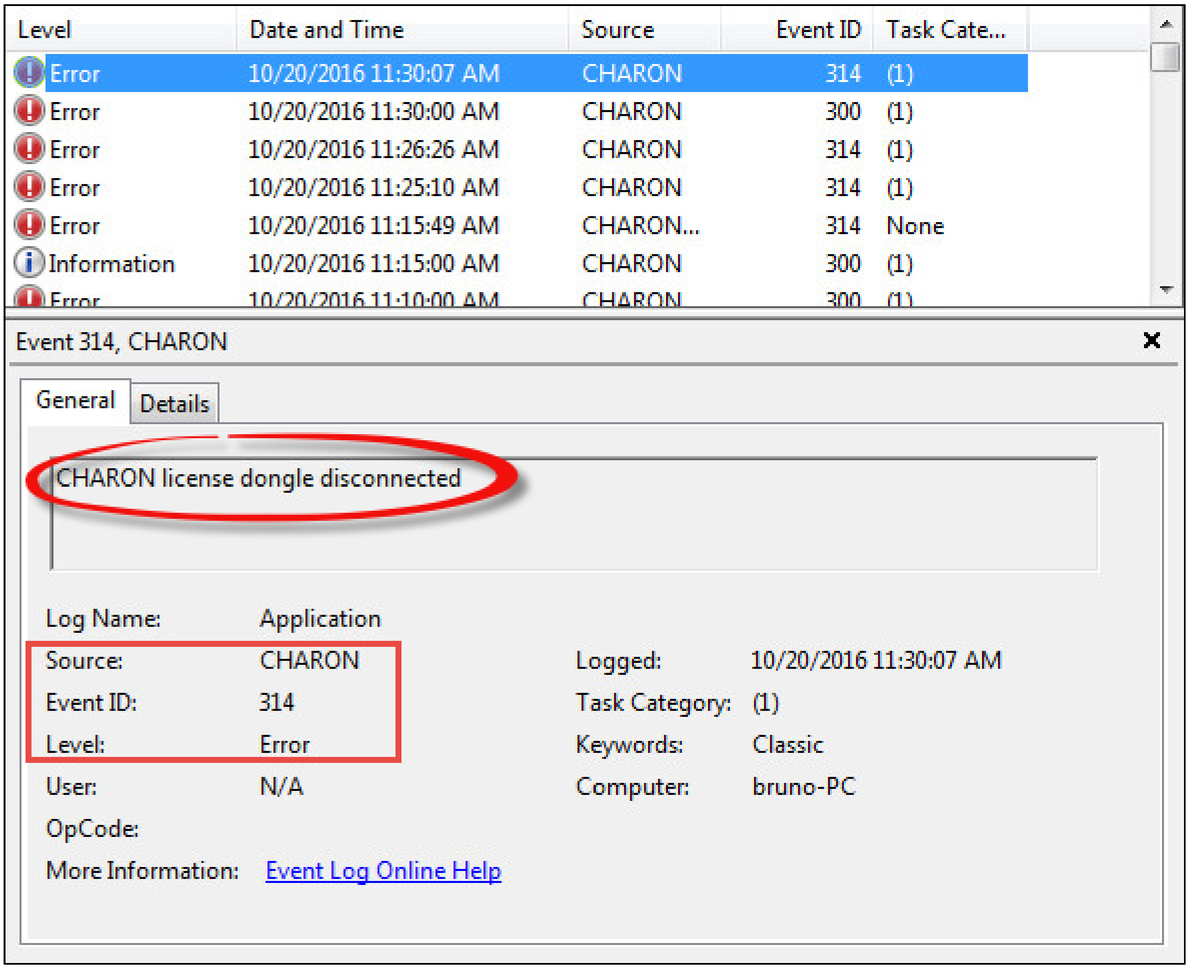
Reminder: Interactive applications are not allowed

Creating a Windows Event

```
powershell -command "New-Eventlog -Logname application -Source Charon -Erroraction silentlycontinue;Write-Eventlog -Logname
application -Source Charon -Entrytype Error -Eventid 314 -Message 'Charon license dongle disconnected'"
```

It is recommended to use the powershell command to write events instead of the "eventcreate" command line as some other events are created with powershell scripts. Using the eventcreate command in this case will fail as the source, Charon, will not be accepted.

Example:



Level	Date and Time	Source	Event ID	Task Cate...
Error	10/20/2016 11:30:07 AM	CHARON	314	(1)
Error	10/20/2016 11:30:00 AM	CHARON	300	(1)
Error	10/20/2016 11:26:26 AM	CHARON	314	(1)
Error	10/20/2016 11:25:10 AM	CHARON	314	(1)
Error	10/20/2016 11:15:49 AM	CHARON...	314	None
Information	10/20/2016 11:15:00 AM	CHARON	300	(1)
Error	10/20/2016 11:10:00 AM	CHARON	300	(1)

Event 314, CHARON

CHARON license dongle disconnected

Log Name:	Application		
Source:	CHARON	Logged:	10/20/2016 11:30:07 AM
Event ID:	314	Task Category:	(1)
Level:	Error	Keywords:	Classic
User:	N/A	Computer:	bruno-PC
OpCode:			
More Information:	Event Log Online Help		

The "Event Id" can be set at your convenience. A valid ID is any number from 1 to 65535.

Sending an email via powershell

An alert email can be sent using the "`Send-MailMessage`" powershell command however it is important to know no interactive command can be executed (for asking the sender's credentials for example).

You can either ask the `nolicense.bat` file to invoke a powershell script or pass the commands from the command line

⚠ Examples below are given with Powershell V4.0 installed on a Windows 2012 R2 server, some commands may not be appropriate to your Windows distribution.

To determine which version of Powershell is installed and upgrade if necessary, see Powershell version, upgrade, enabling scripts execution, tips and tricks

To run PowerShell scripts (files that end with .ps1), you must first set the execution policy to Unrestricted (This operation has to be done once).

To do so, open a command line window (cmd.exe) as an Administrator and use the following command:

```
c:\Charon>powershell -command "Set-ExecutionPolicy Unrestricted"
```

The ExecutionPolicy can also be set to "`RemoteSigned`". In this case the `.ps1` script files will have to be unblocked as described below.

If you are still prompted to allow for execution of the script, please run the following command to unblock the `.ps1` file you want to execute:

```
c:\Charon>powershell -command "Unblock-File -path c:\Charon\sendmail.ps1"
```

Example using an office365 account

Create a powershell script file, named `sendmail.ps1` for example and located in "`C:\Charon`" folder:

```
# Update the email addresses below:
$From = "<monitoring-account>@<somewhere>"
$To = "<someone>@<somewhere>"

# Send the email
send-mailmessage -to $To -from $From -subject "Charon detected removal of the license" -body "Please check" -smtpserver smtp.
office365.com -usessl -port 587 -delivery none
```

If you must specify multiple recipients, you can specify them using an array with comma separated fields.

Example1:

```
$To = @( "Kirk <Kirk@uss-enterprise.fed>", "Spock <spock@uss-enterprise.fed>" )
```

Example2:

```
$To = @( "Kirk@uss-enterprise.fed", "Spock@uss-enterprise.fed", "McCoy@uss-enterprise.fed" )
```

Update the `nolicense.bat` file as follows:

```
powershell -NonInteractive -File C:\Charon\sendmail.ps1
```

The problem here is the email account mentioned in the `$Myemail` variable must be able to send anonymous emails otherwise the following error can occur:

"`Client was not authenticated to send anonymous mail during MAIL FROM`"

If the account cannot send anonymous emails, you can perform an interactive test by asking for the credentials:

The "<monitoring-account>@<somewhere>" and "<recipient>@<somewhere>" values must be adapted to your configuration

```
# Update the email addresses below:
$From = "<monitoring-account>@<somewhere>"
$To = "<someone>@<somewhere>"

# The command below will open a window for you to store the password
$Creds=(get-credential -credential "$From")

# Send the email (credentials are then required each time)
send-mailmessage -to $To -from $From -subject "Charon detected removal of the license" -body "Please check" -smtpserver smtp.
office365.com -usessl -port 587 -delivery none -credential $Creds
```

This method cannot be used when executed within a service due to the popup window that will be opened to ask for the email account password.

To be able to send an email, we can then store the credentials in an encrypted and protected file. This is done using the `Export-CliXML` powershell command. As this file will be bound to the server where the command is issued and to the user account who created the file, we need to perform the operations as "`system`" user account because this user is the one running the `logmond` processes that will invoke the `nolicense.bat` file.

It is necessary then to use `psexec` from the Windows Sysinternals kit. It can be downloaded here: https://technet.microsoft.com/en-us/sysinternals/psexec.aspx

Once extracted from the zip file, to the "`C:\Charon`" folder for example, run the following command from the command line running as Administrator:

```
C:\...> C:\Charon\psexec -i -s cmd.exe
```

and generate the CliXML file as described below:

```
C:\Windows\system32>powershell
Windows PowerShell
Copyright (C) 2014 Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> $From="<monitoring-account>@<somewhere>"
PS C:\Windows\system32> $Creds=(get-credential -credential $From)
PS C:\Windows\system32> $Creds | Export-CliXML C:\Charon\creds.clixml
PS C:\Windows\system32> exit

C:\Windows\system32>exit
```

If it is no more needed, remove the "`psexec.exe`" file.

If the sender's email address or its password has to be changed, the `.clixml` file will have to be recreated.

Now update the powershell script as follows:

The "<monitoring-account>@<somewhere>" and "<recipient>@<somewhere>" values must be adapted to your configuration

```
# Update the email addresses below:
$From = "<monitoring-account>@<somewhere>"
$To = "<someone>@<somewhere>"

$Creds = Import-CliXml C:\Charon\creds.clixml

send-mailmessage -to $To -from $From -subject "Charon detected removal of the license" -body "Please check" -credential
$Creds -smtpserver smtp.office365.com -usessl -port 587 -delivery none
```

Once updated, the script will only work when executed by the "`system`" account and on the server where the `Export-CliXML` powershell command was issued.

Example using a gmail account

To send an email using a gmail account, perform the same operations as described in the Example using an office365 account chapter above and replace the `smtpserver` value by `smtp.gmail.com` as shown below:

The "<monitoring-account>@gmail.com" and "<recipient>@<somewhere>" values must be adapted to your configuration

```
# Update the email addresses below:
$From = "<monitoring-account>@gmail.com"
$To = "<someone>@<somewhere>"

$Creds = Import-CliXml C:\Charon\creds.clixml

send-mailmessage -to $To -from $From -subject "Charon detected removal of the license" -body "Please check" -credential
$Creds -smtpserver smtp.gmail.com -usessl -port 587 -delivery none
```

Only the `smtpserver` value must be changed

Google may block sign-in attempts when using powershell and send-mailmessage. In this case the sender will receive a "Sign in attempt prevented" alert email. To allow emails to be sent:

1. Create a dedicated gmail account
2. Allow less secure apps to access your account. See this article: https://support.google.com/accounts/answer/6010255?hl=en

Related articles

- CHARON on Windows - Automated License Expiration Check
- CHARON on Windows - Charon Log Monitor
- CHARON on Windows - Charon Log Monitor - Release Notes
- CHARON on Windows - Rotating log files cleaning
- CHARON on Windows - Rotating log files cleaning - Release Notes

How to implement time synchronisation between CHARON-AXP Host OS and Guest OS

Table of contents

Description

This document will explain how to implement the time synchronization feature using the "`sync_to_host`" parameter in the configuration file. This parameter allows to keep TOY time always synchronized with the host's time and disable undesirable updates to the TOY from guest OS.

Restrictions: Minimum product versions/builds required:

- **Windows**:
  - CHARON-AXP V4.4 Build 148-02 with patch 148-09 installed
  - CHARON-AXP V4.6 Build 166-03 and later
  - Note: CHARON-AXP V4.5 Build 153-03 and 153-05 (patched) are not supported
- **Linux**:
  - CHARON-AXP V4.6 Build 168-03 and later

Step-by-step guide

Configuration file settings

Update the configuration file with the following settings:

Syntax:

```
set TOY sync_to_host = "{as_vms | as_tru64 | as_is}[, nowrite]"
```

⚠ If "sync_to_host" parameter is specified there is no need to specify "container" parameter in addition.

where:

Parameter	Description
as_vms	If the guest OS is OpenVMS/AXP and its date and time must be set to the host's date and time each time it boots.
as_tru64	If the guest OS is Tru64 UNIX and its date and time must be set to the host's date and time each time it boots.
as_is	If the TOY date and time must be set to the host's UTC date and time
nowrite	Forbid updates to the TOY from the guest OS ℹ If you want guest to synchronize itself using DTSS or NTP for example, remove "nowrite"

Example:

```
set TOY sync_to_host = "as_vms, nowrite"
```

To synchronize the guest OS with TOY, use the following commands (from "SYSTEM"/"root" account):

On OpenVMS/AXP	On Tru64 UNIX
`$ set time`	`` # date -u `consvar -g date	cut -f 3 -d ' '` ``

The default value is "`not specified`" - it means that by default CHARON does not synchronize its guest OS time with the CHARON host time but collects date and time from the file specified with "`container`" parameter.

⚠ If "`sync_to_host`" parameter is specified there is no need to specify "`container`" parameter in addition.

ℹ The CHARON virtual machine must be restarted in order to take the new parameter into account

Virtual machine operating system settings

The commands mentioned above used to synchronize the guest OS with TOY are effective only when they are executed. To avoid time difference, these commands must be executed at specified intervals.

You will find below examples on how to implement scripts to perform time synchronization for OpenVMS and Tru64 UNIX.

💡 If you have NTP running on your OpenVMS or Tru64 UNIX system, you can keep it running even if `sync_to_host` is enabled

On OpenVMS/AXP

You need first to perform a manual synchronization between the CHARON server and the CHARON virtual machine using the `SET TIME=` command:

```
$ SET TIME=12:30:00
```

You can use either a batch queue or a detached process to synchronize time. The two methods are described below.

Using a batch queue

Create a simple script containing the following lines. The example below will sync time every hour:

ℹ In our example, we will create the script in the `SYS$MANAGER` folder and name it `CHARON_SYNCTIME.COM`. The OS version used is OpenVMS 7.3-2. Its content is:

```
$ SET NOON
$ SET VERIFY
$LOOP:
$ SHOW TIME
$ SET TIME
$ SHOW TIME
$ WAIT 01:00:00
$ GOTO LOOP
```

ℹ The `"$ SET VERIFY"` line is optional, just used for verifying commands are correctly executed

A batch queue will be required to create the job:

- Find an available batch queue or create a new one
- Execute the following command to view available batch queues (refer to OpenVMS documentation)

  ```
  $ SHOW QUEUE /ALL /BATCH
  ```

  If the command returns the following error message: "`%JBC-E-JOBQUEDIS, system job queue manager is not running`", you will need to initialize the queue manager:

  ```
  $ START /QUEUE /MANAGER /NEW
  %%%%%%%%%%%  OPCOM  29-MAY-2015 12:30:07.36  %%%%%%%%%%%
  Message from user SYSTEM on VMS732
  %JBC-I-CREATED, SYS$COMMON:[SYSEXE]QMAN$MASTER.DAT; created
  ```

- Create a dedicated batch queue for the synchronization job (recommended):

  ```
  $ INIT /QUEUE /BATCH /START SYS$SYNCTIME /JOB_LIMIT=1
  $ SHOW QUEUE SYS$SYNCTIME /FULL
  Batch queue SYS$SYNCTIME, idle, on VMS732::
    /BASE_PRIORITY=4 /JOB_LIMIT=1 /OWNER=[SYSTEM] /PROTECTION=(S:M,O:D,G:R,W:S)
  ```

- Submit the job:

  ```
  $ SUBMIT /QUEUE=SYS$SYNCTIME SYS$MANAGER:CHARON_SYNCTIME
  Job CHARON_SYNCTIME (queue SYS$SYNCTIME, entry 1) started on SYS$SYNCTIME
  ```

- Update the systartup script (`SYS$STARTUP:SYSTARTUP_VMS.COM`) to start the `SYS$SYNCTIME` queue and the job at system boot. The two following lines will have to be added at the very end of the script (for example):

  ```
  ...
  $ START /QUEUE SYS$SYNCTIME
  $ SUBMIT /QUEUE=SYS$SYNCTIME SYS$MANAGER:CHARON_SYNCTIME
  $
  $ EXIT
  ```

  For OpenVMS version 5, the systartup script will be named: `SYS$STARTUP:SYSTARTUP_V5.COM`

Using a detached process

Create a simple script containing the following lines. The example below will sync time every hour
In our example, we will create the script in the `SYS$MANAGER` folder and name it `CHARON_SYNCTIME.COM`. The OS version used is OpenVMS 7.3-2. Its content is:

```
$ SET NOON
$ SET VERIFY
$LOOP:
$ SHOW TIME
$ SET TIME
$ SHOW TIME
$ WAIT 01:00:00
$ GOTO LOOP
```

The `"$ SET VERIFY"` line is optional, just used for verifying commands are correctly executed. If you let it active, please replace the "`NL:`" device above by a log file name

Update the systartup script (`SYS$STARTUP:SYSTARTUP_VMS.COM`) to start the detached process at system boot. The following lines will have to be added at the very end of the script (for example):

```
...
$ RUN SYS$SYSTEM:LOGINOUT /AUTHORIZE /DETACH /UIC=[SYSTEM] -
      /PROCESS_NAME="TIME SYNC" /OUTPUT=NL: /ERROR=NL: -
      /INPUT=SYS$MANAGER:CHARON_SYNCTIME.COM
$
$ EXIT
```

To start the job manually without a reboot, just execute the line above from an interactive session.

Considerations using DECnet-Plus software

If you have DECnet-Plus software installed, you will have to disable DTSS before setting time in order to avoid errors like:

```
%SET-E-NOTSET, error modifying time
-SYSTEM-E-TIMENOTSET, time service enabled; enter a time service command to update the time
```

To disable DTSS, you will have to update the `CHARON_SYNCTIME.COM` script:

```
$ SET NOON
$! SET VERIFY
$LOOP:
$ SHOW TIME
$ RUN SYS$SYSTEM:NCL
DISABLE DTSS
DELETE DTSS
EXIT
$ SET TIME
$ SHOW TIME
$ @SYS$STARTUP:DTSS$STARTUP
$ WAIT 01:00:00
$ GOTO LOOP
```

It is recommended here not to use "`SET VERIFY`"

On Tru64 UNIX

Restrictions: The synchronization requires the "consvar" command to be available on the Tru64 operating system thus Tru64 UNIX version 4.0F minimum is required

You need first to perform a manual synchronization between the CHARON server and the CHARON virtual machine using the date command. Example:

```
# date -u 05291724
```

Create an entry in the root's crontab file using "**crontab -e**" as shown below:

```
...
# CHARON time sync_to_host
00 * * * * /sbin/date -u `/sbin/consvar -g date | cut -f 3 -d ' '`
```

Full path to date and consvar commands must be specified

The above command will be executed at minute 0 of every hour. If you want to execute this every 15 minutes for example, use the following line instead:

```
00/15 * * * * /sbin/date -u `/sbin/consvar -g date | cut -f 3 -d ' '`
```

More information at Wikipedia.org - Cron

Related articles

How to implement time synchronisation between CHARON-AXP Host OS and Guest OS (sync_to_host)