

PLUTI (Prophet) installation on SymbioNode - USB storage device



MEIS d.o.o.

Authors: Sašo Vrbinc

Postal address: MEIS d.o.o., Šmarska 40, SI-1291 Škofljica, Slovenia

Email: info@meis.si, Web: www.meis.si

June 2012

EMPTY PAGE

DOCUMENT: eLearningDTN-SymbionodeInstallation_v01.doc

DOCUMENT HISTORY

VERSION	DESCRIPTION	DATE	AUTHORS
01	Creation of the document	20.06.2012	Sašo Vrbinc

EMPTY PAGE

TABLE OF CONTENTS

1. INSTALLATION ON LINUX	7
1.1 CONFIGURATION OF AUTOMATIC MOUNT STORAGE DEVICE	7
1.2 SYSTEM “AUTORUN” CONFIGURATION	8
1.2.1 <i>Autorun PLUTI Script using Udev</i>	8
1.2.2 <i>Autorun PLUTI script with Hotplug2</i>	9
2. INSTALLATION ON WINDOWS.....	10
2.1 CYGWIN EMULATOR.....	10
2.1.1 <i>prophet.sh</i>	10
2.2 CONFIGURATION OF SYMBIONODE ON STORAGE DEVICE	11
2.2.1 <i>prophet.ini</i>	11

EMPTY PAGE

1. Installation on Linux

Installation of SymbioNode on Linux platform takes the following steps:

- Configure automatic mount of USB storage device
- PLUTI script installation
- Auto start of PLUTI script

1.1 Configuration of automatic mount storage device

Linux have special configuration file for mounting storage devices. Most configured devices in the file are mounted on boot. The most evident example is 'root' partition.

First we want to add additional mount point for our storage device. Linux assigns a name for all supported devices and partitions. The standard 'fstab' names notation for storage devices are generally in form 'sd[a-z]' or 'hd[a-z]', and are located in in '/dev' directory.

Removable disks don't get the same name every time when inserted, so it's safer to address partitions by ID in order to avoid complications. Most Linux distributions provide partition addressing by ID and it's recommended to use it. The usual location containing the links to all storage devices and its belonging partitions is in /dev/disk/by-id/. Every device has its own unique ID and unlikely any conflicts will ever occur.

Mounting the device to appropriate location is crucial for SymbioNode to work correctly. When standard fstab notation is used for addressing device name, an incorrect mount point can be assigned by the kernel if mount point already exists. This will lead to inconsistency with PLUTI configuration, and it results in broken PLUTI application.

Follow the steps described below to configure mount point:

- 1) Login to the console as root user
- 2) Create a directory for tor storage device mount point
- 3) Insert the device to the USB interface
- 4) Find correct label of the storage device

```
# dmesg | tail
sd 6:0:0:0: [sdb] 3907584 512-byte hardware sectors: (2.00 GB/1.86 GiB)
sd 6:0:0:0: [sdb] Write Protect is off
sd 6:0:0:0: [sdb] Mode Sense: 23 00 00 00
sd 6:0:0:0: [sdb] Assuming drive cache: write through
sd 6:0:0:0: [sdb] 3907584 512-byte hardware sectors: (2.00 GB/1.86 GiB)
sd 6:0:0:0: [sdb] Write Protect is off
sd 6:0:0:0: [sdb] Mode Sense: 23 00 00 00
sd 6:0:0:0: [sdb] Assuming drive cache: write through
```

In our example sdb is the device label

5) Find the device ID

```
# ls -l /dev/disk/by-id/
usb-Verbatim_STORE_N_GO_07980809E465303B-0:0 -> ../../sdb
usb-Verbatim_STORE_N_GO_07980809E465303B-0:0-part1 -> ../../sdb1
```

The second line is the device first and only partition. We'll use the label in our configuration.

6) Append the following line into file '/etc/fstab'

/dev/disk/by-id/usb-Verbatim_STORE_N_GO_07980809E465303B-0:0-part1	/Leonardo
ext3 rw, sync 0 0	

Note: Replace ext3 with appropriate file system type

1.2 System "Autorun" configuration

Most usual autorun systems found on Linux are udev and Hotplug2. They both offer execution of custom actions on certain events invoked by physical device. Events are device specific, and in the case of removable storage device, two types of events (attached/detached) can be generated. In general, every removable device produce the same type of events, but event information is specific.

1.2.1 Autorun PLUTI Script using Udev

Udev gives a convenient way for selective running of custom scripts based on two criteria:

- device and
- event type.

According to both criteria an appropriate rule needs to be formulated to handle attached/detached storage devices. Rules default location is '/etc/udev/rules.d/', and are in form:

```
<number[0-99]>-<name>.rules
```

Custom rules usually have higher numbers, so PLUTI rule file should look like "88-prophet.rules". Rules syntax will not be described here since many good references exists on the web.

SymbioNode can be configured for various types of storage devices (USB key, USB hdd, CF, SD cards,...). For USB based storage devices Udev rule should look like this:

- 1) Copy prophet.sh to location '/lib/udev/'
- 2) Create file '/etc/udev/rules.d/20-symbionode.rules', and add the rule as described bellow

```
KERNEL=="sd[b-z][1-9]", ACTION=="add|remove", RUN+="/lib/udev/prophet.sh %k"
```

File: /etc/udev/rules.d/20-symbionode.rules

KERNEL – Name of a device defined by the Kernel. Usually one storage device is already present as sda or hda, and the rule should be properly adjusted. USB storage devices are represented in the system in form “sd[a-z][1-9]”. The rule above applies for all added storage devices on SCSI interface except for the existing 'sda' device which is in our case a system disk.

ACTION – With this condition it's possible to catch only certain events.

RUN - Path to the application or script that executes operations for the attached device. The operator %k is evaluated by the kernel, and stands as a device name which is passed to the script by Udev.

- 3) Reboot the computer

1.2.2 Autorun PLUTI script with Hotplug2

“Hotplug2” system is predecessor of Udev, and therefore slightly more rigid. It can still be found on some Linux distributions, mostly intended for embedded devices.

Primarily, “Hotplug2” doesn't have a strong rule based system, and therefore target device needs to be found with custom program, usually a bash script. System only separates Kernel events based on device subsystem type. Each subtype has a belonging directory in /etc/hotplug.d/, such as usb, that contains scripts for handling newly added/removed device. Nevertheless, script name paradigm and calling routines are same as in “udev”.

- 1) Copy prophet.sh to location '/etc/hotplug.d/usb/'
- 2) Restart the computer

2. Installation on Windows

SymbioNode functionality was primarily implemented for Linux based platforms in shell script language Bash. Compatibility of PLUTI script among various Unix based platforms is assured by Bash interpreter, as long POSIX standard syntax and commands are used. Windows has similar variant of shell scripting tool called Batch, but unfortunately it's incompatible with Bash. In order to avoid another implementation of SymbioNode for Windows, we've rather used Cygwin emulator that provides a Unix-like environment. Therefore, it's possible to run the existing SymbioNode script without any modification.

Each SymbioNode instance has Autorun capabilities if the host Windows machine supports it. Some configurations of Windows don't allow Autorun applications due to security reasons and therefore manual execution of the script is required.

In latest developments we simplified the execution of PLUTI Instance ran by SymbioNode. Older versions of SymbioNode supported only two platforms(x86, Cambria). Required binaries were physically located on every SymbioNode storage device, which was a major drawback, since the ability of supporting additional platforms was highly difficult. The newest version of the script removes the drawback and runs PLUTI binary from the native machine with the configuration (prophet.ini) provided by SymbioNode.

2.1 Cygwin emulator

The emulator is freely available on www.cygwin.com/setup.exe. Download setup.exe and run the program. Follow the installation steps and do not change any default settings. Most important is that the installation path is left on default location (C:/cygwin) so that the configuration is consistent with the SymbioNode.

The installation takes a while, since all selected packages must be downloaded first. If the installation was successful then it's ready for installation and configuration of SymbioNode.

As described in introduction section, SybioNode is a shell script written in Bash. The script relies on PLUTI that needs to be present on host computer, therefore make sure you have it installed. In this text the installation path of PLUTI is c:\prophet\.

Make sure you copy both script "prophet.sh" and "locking.sh" to your local disk c:\cygwin\bin\.

Autorun script located on storage device of SymbioNode calls prophet.sh in cygwin emulated shell in directory c:\cygwin\bin\prophet.sh.

2.1.1 prophet.sh

PROPHET_LOCAL_DIR	This parameter needs to be modified according to the installation location of the prophet. By default the path is C:/prophet/
-------------------	---

WATCHDOG_DELAY

This parameter defines how long the SymbioNode prophet instance will be running. (300 seconds by default)

2.2 Configuration of SymbioNode on storage device

To create a SymbioNode you need a storage device, such as USB pen, CF card or SD memory card. Format it with FAT file system in order to use the key on various platforms. If it's planned to use the SymbioNode only on Windows, it may be formatted with NTFS.

Afterwards the following steps must be taken:

- 1) Create directory /AutoExe/prophet/ and create or copy an existing prophet.ini to that folder
- 2) Copy autorun.bat & autorun.inf on root directory

2.2.1 prophet.ini

The configuration file prophet.ini is located in AutoExe/prophet/prophet.ini, on storage device of SymbioNode. The prophet.sh script searches prophet.ini only in that directory. If it's not found the prophet process will not start! The following table shows a list of parameters that needs to be modified in order to make SymbioNode working:

NODEIP	SymbioNode's Prophet relies on localhost network interface, therefore it has to be set on 127.0.0.1
NODEIP2	Should be equal as NODEIP (127.0.0.1)
DTNHOSTNAME	Should be equal as NODEIP (127.0.0.1)
STORAGEPATH LOGPATH DFTPPATH	It's recommended to use relative paths. However, if absolute path with specified drive is defined (D:\some\path\to\storage) the data will not be written on SymbioNode storage device.