openSysMon plugin for openLuup

Version : beta (0.4)
Date : March 16, 2016
Author : logread (LV999), contact by PM on http://forum.micasaverde.com/

Special thanks to akbooer for his support and for developing the openLuup environment !

Acknowledgement to Chris Jackson for the original System Monitor plugin on which openSysMon is based.

Introduction :

This plug-in is intended to run under the "openLuup" emulation of a Vera system. It should work on a "real" Vera, but there is no point of doing so (the original System Monitor plugin by Chris Jackson was designed for that and is available on the MiOS app store).
It is intended to capture select system data from a running Vera system and to display/monitor these on an openLuup system. The ideas that lead to openSysMon are:

1. By construction, the original System Monitor plugin surveys the system on which it is running, not a “remote” one (by “remote” I mean another system running on the same LAN)
2. The openLuup environment allows to migrate as much as possible highly resource consuming processes (such as plugins) from the Vera to a remote system. This means that by using openSysMon, the original System Monitor plugin can be removed from the Vera… it is replaced by a dedicated Lua module running in the background, that uses far less resources than a stand-alone plugin.

Architecture:

openSysMon is comprised of two components:

1. Server side: A plugin running on the openLuup system that will process and monitor the system parameters from the “remote” Vera.
2. Client side: A dedicated backend Lua module running on the “remote” Vera that will poll the local system, package the data and send it to the plugin on the openLuup system.

Limitation: in its current version, only one instance of the plugin can run, that is only one “remote” Vera can be monitored. In the future, if the need/opportunity arises, a multiple instances configuration could be easily developed.

Requires:

1. A system running openLuup (or a Vera home automation controller, not tested) and the AltUI interface. For background, please see the http://forum.micasaverde.com/ forum.
2. Lua library “socket.http” installed (should already be in an openLuup environment (please refer to openLuup documentation).
3. Reasonable technical knowledge of the Vera environment, including how to use SSH/SCP to manage files. **The openSysmon code is beta software and you install and use it at your own risk.**

Installation:

1. On the openLupp system:
    a. Using your favorite remote file manager (e.g. WinSCP), upload all plugin files (“D_openSysMon.json”, “D_openSysMon.xml”, “I_ openSysMon.xml”, “L_ openSysMon.lua” and “S_openSysmon.xml”) to the “/etc/cmh-ludl/” folder.
    b. Copy the content of the openSysMon_create.lua file and paste it in the “run test lua” page of AltUI. Click “Save” and take note of the installation report.

2. On the Vera system:
    a. using your favorite remote file manager (e.g. WinSCP), upload the "SysMonModule.lua" file to the /etc/cmh-ludl/" folder.
    b. Edit the "Luup Startup code" and add the following lines (preferably at the beginning of the code):

    ```
    SysMon = require("SysMonModule")
    SysMon.config("targetIP", <period>)
    ```

    where:
    "targetIP" is the IP address of the openLuup machine on the LAN in string format (for example "192.168.90.10")
    <period> is the interval in seconds between consecutive system polls, as a number (without brackets or delimiters). That parameter is optional, the default value is 300 (i.e. 5 minutes). The module will reject inputs of less than 20s and more than 3600s.

    c. Reload the Luup engine… the module will not run until this happens !

Use:

The openSysMon plugin on the openLuup system behaves exactly like the original System Monitor plugin, with all the monitored system parameters appearing as variables inside the plugin. Variable watches and triggers for scenes/alerts should work (not fully tested !) as in the original plugins.

At any time the `config()` function can be called within the Vera Luup environment to change the IP and sample period parameters. The function can also be called without parameters, in which case it makes no change to the configuration and just returns the current IP and sample period settings. For example, with the parameters set as in the installation earlier in this document, the following Lua code:

```
do
	local currentIP, currentperiod = SysMon.config()
	print(currentIP, currentperiod)
end
```

will output: `192.168.90.10	300`

Notice: