**Server Uptime Tool**

**Background**

The Rwandan Ministry of Health (MOH) has deployed OpenMRS EMR servers to over 250 health facilities across Rwanda. While many servers have been deployed little data is available on how much the servers are being used and what the greatest hurdles are to consistent usage of the EMR. As part of a grant from the Rockefeller Foundation, the MOH and IMB EMR teams would like to develop tools that will help capture data related to system usage and help monitor key factors that contribute to usability of the system.

At a base level, the EMR can only be used when the underlying server is turned on and running. Thus first tool we need is one that will determine how much a server is on/running during primary clinic hours.

**Outcome**

To determine the percentage of uptime of the Ubuntu server (which has the OpenMRS EMR application installed on it) during primary clinic hours for a period specified by the user. Optionally (as the technical details are not yet clear) the uptime of the OpenMRS technology stack (Java, Tomcat, MySQL, OpenMRS Webapp) should be measured.

**Definitions**

Uptime – The Ubuntu server is turned on and running

Start Date – the first day of the period for which “uptime” should be calculated

End Date – the last day of the period for which “uptime” should be calculated

System ID - Ideally meaningful and unique identifier of current system. (Either facility name/code, system name, or network MAC address)

Default Start Date – (run\_date – 8 days), used when the tool is set to run automatically

Default End Date – (run\_date – 1 day), used when the tool is set to run automatically

Primary Clinic Days – Monday, Tuesday, Wednesday, Thursday, Friday

Primary Clinic Hours – 08:00-17:00 on primary clinic days

**Requirements**

* Tool should have a minimal install “footprint”
* Tool must run on Ubuntu servers (versions 10.4 and 12.X)
* Tool can be set to run automatically on a day/time specified by the administrator
  + When run automatically, the tool should use the Default Start Date and Default End Date as the input parameters
  + Default Start Date and Default End Date parameters should be exposed in a way that the system administrator could change their definitions in the tool (e.g. run\_date – 15 days for running the tool every 2 weeks)
* Tool can be run “on demand” by a user
  + Steps to run the tool should be minimal enough to be completed by a user with limited computer skills/knowledge. It is assumed that this user will have a local (SSH) login to the Ubuntu system and with enough documentation is able to execute the tool via command line / shell.
  + Tool can be run for different time periods with start date and end date as parameters input by the user each time the tool is run
* Primary Clinic Days and Primary Clinic Hours have a common definition in the tool and cannot be changed by a user, but Primary Clinic Days and Primary Clinic Hours parameters should be exposed in a way that a system administrator could change them in the tool
* When run, the tool should use the Start and End Dates as bounds and calculate the percent of uptime for the Ubuntu server during Primary Clinic Hours
* The tool should output a .pdf file listing the Start Date and End Date entered by the user, the current values of the Primary Clinic Days and Primary Clinic Hours parameters, and the percent of uptime for the period between (and including) start and end date.
* The tool should also record in the output file each startup/shutdown event and date/time that was used in the calculation. (*this may not make sense depending on how Ubuntu logs the events*)
* The output file should be saved to the desktop in a folder called Server\_Uptime, with a file name of Server\_uptime\_[*date/time tool was run*].pdf.

**Usage**

**Automatic start**

Start every Sunday night (23:50) for the last week

Open: Where should the results be saved, how does the user gets access to it?

**Manual start**

Log in via SSH to the Ubuntu command line

Invoke command: system\_report.sh <start date> <end date>

Open: Where should the results be saved, how does the user gets access to it?

**Change defaults**

Change defaults for primary clinic days and hours; To be defined

**Storage:**

Instead of parsing the general Ubuntu log files, a dedicated log just for this monitoring tool will make it less dependent on the specific installation details of Ubuntu. The format of the file (which should be stored under /var/log) is as follows:

<Date & Time> <System id> <Message type> <Values>

As Message types the following values are currently defined:

STARTUP, SHUTDOWN, SYSTEM HEARTBEAT, OPENMRS HEARTBEAT

Optional additional values can be stored as <Values> (like system load average for SYSTEM HEARTBEAT)

Example:

Jan 21 13:03:39 ; HealthCenter123 ; STARTUP

Jan 21 13:08:39 ; HealthCenter123 ; SYSTEM HEARTBEAT ; 1.20

Jan 21 13:13:39 ; HealthCenter123 ; SYSTEM HEARTBEAT ; 0.20

Jan 22 10:03:39 ; HealthCenter123 ; STARTUP

Jan 22 10:08:39 ; HealthCenter123 ; SYSTEM HEARTBEAT ; 1.20

Jan 22 10:13:39 ; HealthCenter123 ; SHUTDOWN

**Output**

Based from this storage a simple PDF document will be generated as text for every run.

Current date and time: Jan 23 17:00:00

System ID: HealthCenter123

Start date (including): Jan 21

End date (including): Jan 23

Primary Clinic Days: Monday, Tuesday, Wednesday, Thursday, Friday

Primary Clinic Hours: 08:00-17:00

Percentage of system uptime (1): 90%

Percentage of OpenMRS uptime (1) (optional): 50%

Number of system starts (2) (optional): 2

Number of system starts without preceding shutdown (2)(optional): 0

Three highest average system loads (2) (optional): 1.20 2.20 1.04

(1) during clinic hours between start and end date

(2) between start and end date (incl. outside of clinic hours)

**Installation:**

Is currently unknown as this depends on system setup and abilities to change them.

**Open issues/questions:**

* How can the servers be accessed locally? E.g. is a local desktop (monitor/keyboard) attached to every OpenMRS server or is there always a local network with a SSH access from a Windows computer?
* How are the software components of the installation maintained? E.g. are there any ways to use apt-get to install additional components or a well-defined way to install/update OpenMRS modules?
* Is there a place for automated reports could be saved? E.g. a shared drive or local folder? If local folder, how can the use access it?