BLEKINGE TEKNISKA HÖGSKOLA

Monitoring the Performance of Virtual Machines

TEAM: 'SHIELD'

HARSHINI NEKKANTI

HIMA BINDU NUTALAPATI

JYOTHI SPANDANA PENMETSA

NAVYA UPPALAPATI

PRIYASUBHA CHUNDRU

RAYWON TEJA KARI

SAIPHANI KRISHNA PRIYANKA KOLLURI

SASANK SAI SUJAN ADAPA

SRAVANI KANCHARLA

TULASI PRIYANKA SANABOYINA

VEERAVENKATA NAGA SOMESWARA MANITEJA DARISIPUDI

Document Type: **Developer Document**

Version 1.2

CONTENTS

Preface:

This document is the developer document, intended for the developers to understand and improvise the existing software. The document provides information on the dependencies of the product.

Revised version v1.2 on 2015-06-15

-Added details on source codes.

-Added brief description on JSON format.

Revised version v1.1 on 2015-05-29

-Source code organization included.

Initial version v1.0 on 2015-05-20

-Initial release

Glossary and Abbreviations:

API: Application Program Interface

RRD: Round Robin Database

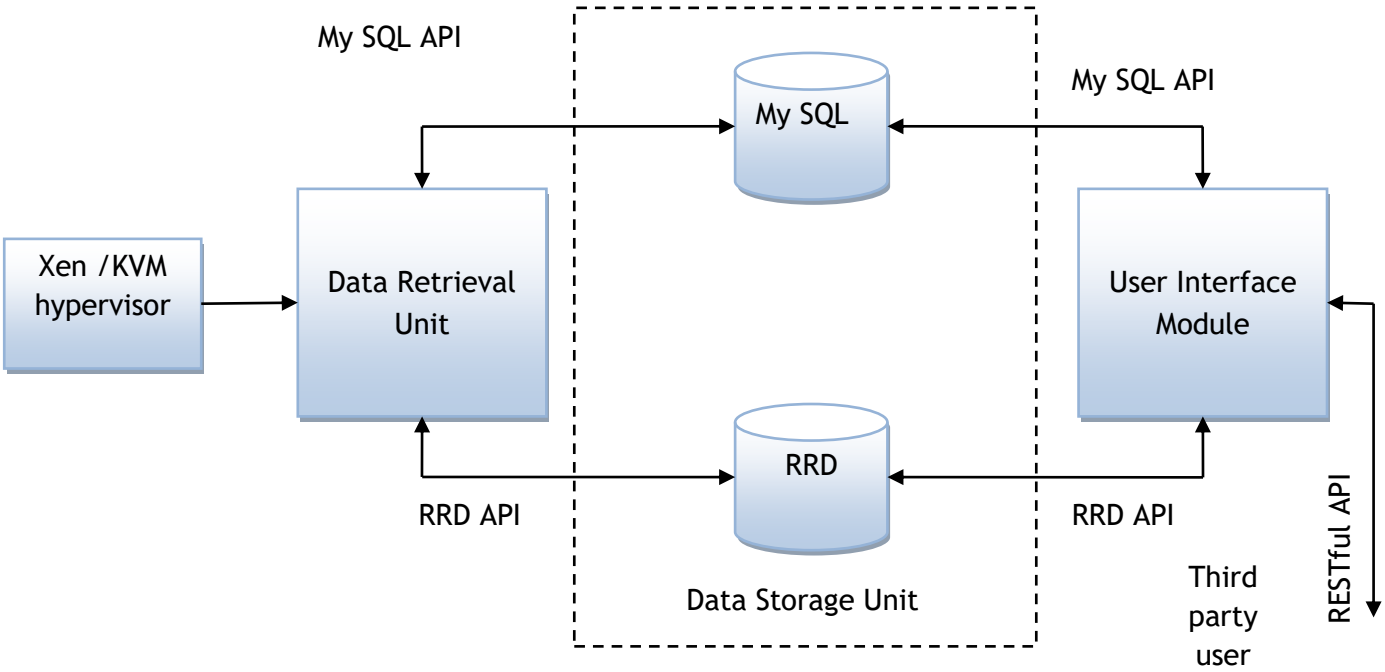
SSH: Secure Shell

VM: Virtual Machine

Description of the product:

The user interface module provides access for the user to interact with the system. The user can add devices to the monitoring list and remove devices from the monitoring list, view device statistics and utilization graphs, view status of the devices and receive email notifications when devices exceed the thresholds. It allows exporting the data shown to the user, to a third party application using RESTful API. This can be done by using URLs. The requested data will be exported in JSON format. For exporting the resource utilization data, the third party user must request the data using the URL in the format http://localhost/rest.php/?resource=<utlilizationmetric>

System Architecture:



Modules implemented:

➢ User Interface Module:

The user interface module provides access for the user to interact with the system. The user can add devices to the monitoring list and remove devices from the monitoring list, view device statistics and utilization graphs, view status of the devices and receive email notifications when devices exceed the thresholds. It allows exporting the data shown to the user, to a third party application using RESTful API. This can be done by using URLs. The requested data will be exported in JSON format. For exporting the resource utilization data, the third party user must request the data using the URL in the format http://localhost/rest.php/?resource=<utlilizationmetric>

➢ Data Storage Module:

The data storage module is a database used to store the information retrieved by the data retrieval module. The data stored in the database can be accessed by the user interface module using MySQL API and RRD API. MySQL database is used to store the details of the device and the login credentials while RRD is used to store

the utilization and historical aggregate data required in generating the utilization graphs.

➢ Data Retrieval Module

The data retrieval unit is used to retrieve the CPU load and utilization, I/O usage, network usage, memory usage and disk usage of the devices to be monitored. SSH is used to retrieve the information and the retrieved information is stored into the database. When the product is installed, the retrieval unit gets the information and stores into the database periodically.

Product dependencies:

- Apache2
- phpMyadmin
- MySQL
- php5
- RRD tool

Product Implementation:

➢ Data retrieval module:

The data retrieval module is implemented as the backend part of the product. This module is responsible for retrieving the required utilization metrics using ssh. The data retrieved is stored into MySQL and RRD databases. The backend script is written in perl. When the script is run the first time, a database with all the tables is created is it stays in loop for the device details to be added from the frontend. When the data is retrieved, it is stored into the database.

➢ Dependencies:

- Perl
- ssh
- RRD database
- MySQL database

➢ Data storage module:

This module is the database, common for both backend and frontend. MySQL and RRD databases are used in the implementation of the product. The login credentials and device details are stored into MySQL database and the device statistics that are used to generate the utilization graphs in the frontend are stored into RRD database.

- **Dependencies:**
  - MySQL
  - phpMyadmin
  - RRD

- **Database table:**
  - Table name: DEVICES
    Column parameters in the table
    - ID
    - IP
    - Username
    - password
    - VMname
    - cpu
    - memory
    - networkinput
    - networkoutput
    - disk

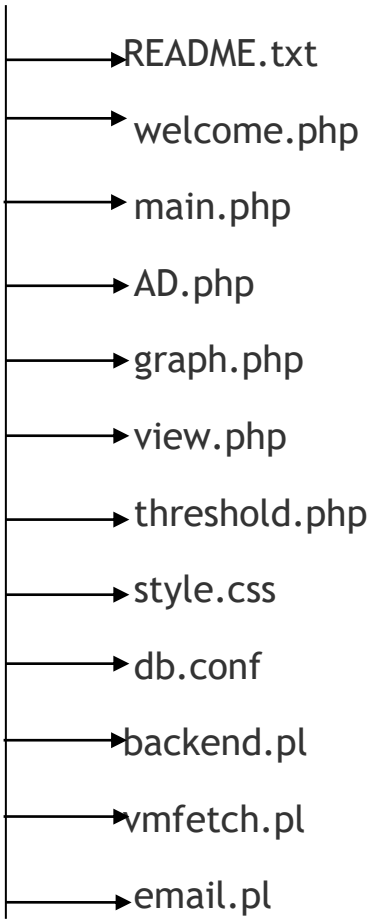
- **User interface module:**

The user interface module is implemented as the frontend part of the product. This module is responsible for providing user interaction to the product using a html page. This module can export data to third party users making use of a RESTful service.

- **Dependencies:**
  - HTML
  - PHP
  - CSS

Source code organization:

SHIELD



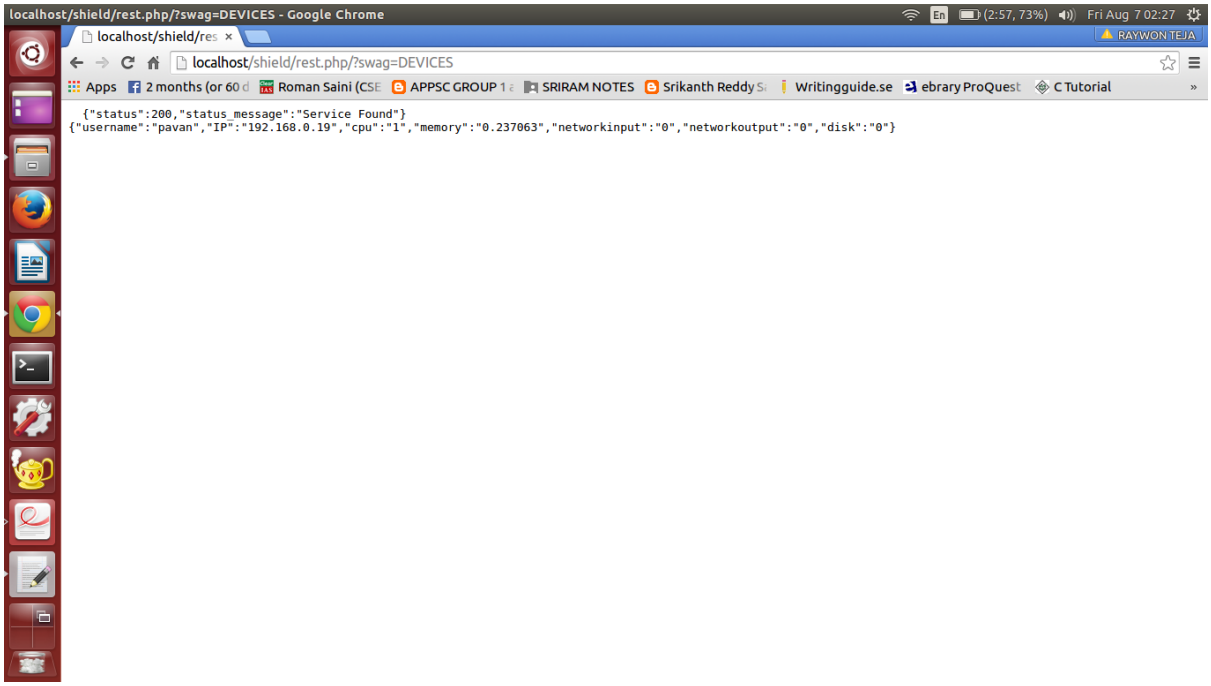
- README.txt: In the text file, following information is present:
  - System requirements
  - Prerequisites
  - Dependencies
  - Operation
  - Few instructions on how to use the tool
- welcome.php: this is the webpage through which the tool is started

- main.php: this is the script which has the logout , add delete, view graphs, threshold links
- AD.php: this script is used to add and delete devices, vms.
- graph.php: this script is used to retrieve specific data to print in the graphs
- view.php: this script is used to display respective metrics
- threshold.php: this script is used to define and display thresholds
- style.css: basic css file used to style html pages
- db.conf: mysql database credentials such as username, password etc are present.
- backend.pl: Is the main perl script, which performs necessary backend functions
- vmfetch.pl: Is used to retrieve respective virtual machines in devices
- email.pl: is used to send emails

Restful API:

The product supports exporting the data shown to the user, to a third party application using RESTful API. This can be done by using URLs. The requested data will be exported in JSON format. JSON (Java Script Object Notation) is a type of format which uses human readable text to transmit data objects containing attribute-value pairs. For exporting the resource utilization data, the third party user must request the data using the URL in the format http://localhost/rest.php/?resource=<utlilizationmetric>

Following is an example of how the JSON format looks like:



Future scope:

The existing product can be extended for the following services

- The product is intended for Ubuntu 14.04 LTS. It can be extended for Windows and Mac
- Providing notifications to the user via SMS
- Monitoring the resource utilization of the processes in the virtual machines
- Importing the data from third party users using the RESTful API

Perl modules used:

i. Net::OpenSSH module

- Install Net::OpenSSH

ii. IPC::System::Simple (qw)

- Install IPC::System::Simple (qw)

iii. DBI Module

- Install DBI module

iv. DBD::MySql module

- Install DBD::MySql

v. Data::Dumper module

- Install Data::Dumper

vi. RRD::Simple module

- Install RRD::Simple

vii. Mail::Sender module

- Install Mail::Sender

Commands in PHP to fetch data from database:

- mysql_connect(): To connect to MySQL database.
- mysql_select_db(): To select the database from MySQL.
- mysql_query(): To perform query actions select, insert, update, delete from a table.
- mysql_fetch_array(): To fetch a row from a table in MySQL.
- mysql_close(): To close the MySQL connection
- a href(): To specify the URL of the link the page goes to.
- Post method/ Get method: To pass values from one webpage to another webpage.