

UG4 Project - IT Infrastructure Monitoring System

Cameron Gray

October 19, 2015

What is the project?

To build a complete, self contained system to monitor the servers and other IT infrastructure in small to medium organisations which may not have a dedicated IT department.

What is the project?

To build a complete, self contained system to monitor the servers and other IT infrastructure in small to medium organisations which may not have a dedicated IT department.

- Provide both time series monitoring and real time alerting

What is the project?

To build a complete, self contained system to monitor the servers and other IT infrastructure in small to medium organisations which may not have a dedicated IT department.

- Provide both time series monitoring and real time alerting
- Can be managed entirely from a central interface

What is the project?

To build a complete, self contained system to monitor the servers and other IT infrastructure in small to medium organisations which may not have a dedicated IT department.

- Provide both time series monitoring and real time alerting
- Can be managed entirely from a central interface
- Support for plugins to monitor specialised hardware

System is split up into 4 main components

- Agent
- Core
- Web
- Alerting

Progress - Evaluation of Current Tools

Investigated current tools including Nagios, Icinga 2 and Munin and evaluated them against several points.

- Support for timeseries monitoring and real time alerting
- How they can be configured to monitor custom metrics
- How are alert thresholds defined
- How the user configures the system
- How dependencies are handled

Progress - Agent

Completed - Sits on machine being monitored and provides network interface to fetch data from plugins which are executed by the agent.

```
python agent.py --run-server  
(proj-agent)[cameron@barra agent (master x)]$ python agent.py --run-server  
Agent starting up...  
Creating directory at /tmp/proj  
Running!  
□
```


Progress - Agent

Completed - Sits on machine being monitored and provides network interface to fetch data from plugins which are executed by the agent.

The screenshot displays a REST client interface with the following details:

- Method:** GET
- URL:** `https://127.0.0.1:4048/get-plugin-data/?plugin-id=me.camerongray.proj.test_plugin`
- Authorization:** Basic Auth
- Headers (1):** (empty)
- Body:** (empty)
- Pre-request script:** (empty)
- Tests:** (empty)
- Status:** 200 OK
- Time:** 315 ms
- Body View:** Pretty, Raw, Preview (selected)
- JSON:** (selected)
- Response Body:**

```
1 {  
2   "message": "",  
3   "success": true,  
4   "value": 0.45  
5 }
```

Progress - Agent

Completed - Sits on machine being monitored and provides network interface to fetch data from plugins which are executed by the agent.

```
1 from plugin import PluginInterface, PluginResult
2 import os
3
4 class Plugin(PluginInterface):
5     def get_data(self):
6         (avg, _, _) = os.getloadavg()
7         return PluginResult(avg)
```

Progress - Core

Substantial Progress - Lives on machine doing the monitoring, handles scheduling of checks and sends requests for data to the agents. Multithreaded so can run checks on multiple hosts in parallel.

```
camerong@barra: ~/Uni/honours_project/application/core
(proj-core)[camerong@barra core (master x)]$ python core.py
Core starting up...
Starting scheduler
2015-10-18 23:30:32.137368: Executing Test Plugin on Database 1
2015-10-18 23:30:32.139330: Executing Test Plugin on Database 2
2015-10-18 23:30:32.141500: Executing Test Plugin on Database 3
2015-10-18 23:30:33.138538: Completed Test Plugin on Database 1
2015-10-18 23:30:33.138753: Executing Test Plugin 2 on Database 1
2015-10-18 23:30:33.140493: Completed Test Plugin on Database 2
2015-10-18 23:30:33.140664: Executing Test Plugin 2 on Database 2
2015-10-18 23:30:33.141921: Completed Test Plugin on Database 3
2015-10-18 23:30:33.142081: Executing Test Plugin 2 on Database 3
2015-10-18 23:30:34.139795: Completed Test Plugin 2 on Database 1
2015-10-18 23:30:34.141721: Completed Test Plugin 2 on Database 2
2015-10-18 23:30:34.143121: Completed Test Plugin 2 on Database 3
2015-10-18 23:30:42.023182: Executing Test Plugin on Database 1
2015-10-18 23:30:42.025250: Executing Test Plugin on Database 2
```

Partial Implementation - Allows management of plugins, hosts, checks and schedules within the system. In future will also provide management for alerting and displaying of collected data/graphs.

Prophasis Monitoring

SETTINGS

Hosts & Groups

Plugins

Checks

Scheduling

Checks Edit Check

Home Checks Edit Check

Edit Check


Name


Database Server All Plugins


Description


Member Hosts/Groups


Search


☐ Webservers


☒ Database Servers

☐ Storage Servers

☐ All Hosts

☐ Webserver 1

☐ Webserver 2

☐ ...

Plugins

Search

☒ Test Plugin

☒ Test Plugin 2

Save Check

Back...

Prophasis Monitoring

SETTINGS

Hosts & Groups

Hosts

Host Groups

Plugins

Checks

Scheduling

Host Groups

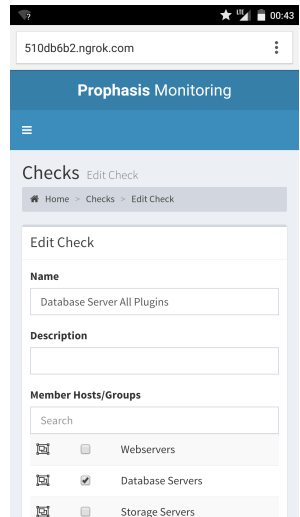
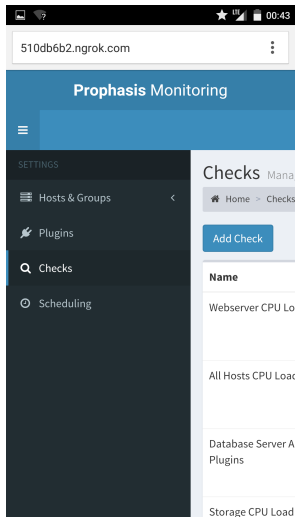
Manage Host Groups

Home > Host Groups

Add Host Group

Name	Description	# Hosts	Actions
Webservers		5	<div>Edit</div> <div>Delete</div>
Database Servers		3	<div>Edit</div> <div>Delete</div>
Storage Servers		4	<div>Edit</div> <div>Delete</div>
All Hosts		11	<div>Edit</div> <div>Delete</div>

Progress - Web



- End of Week 6 (Next week)
 - Initial integration - Core and agent communicating
 - Initial test deployment on Tardis¹ to collect some real data

¹<http://www.tardis.ed.ac.uk/>

- End of Week 6 (Next week)
 - Initial integration - Core and agent communicating
 - Initial test deployment on Tardis¹ to collect some real data
- End of Semester 1
 - Alerting complete
 - Functional interface for visualising collected data
 - Deploy on Tardis and gain feedback from users

¹<http://www.tardis.ed.ac.uk/>

- End of Week 6 (Next week)
 - Initial integration - Core and agent communicating
 - Initial test deployment on Tardis¹ to collect some real data
- End of Semester 1
 - Alerting complete
 - Functional interface for visualising collected data
 - Deploy on Tardis and gain feedback from users
- By start of Semester 2
 - Work on additional features e.g. dependency support and distributed monitoring nodes

¹<http://www.tardis.ed.ac.uk/>

- End of Week 6 (Next week)
 - Initial integration - Core and agent communicating
 - Initial test deployment on Tardis¹ to collect some real data
- End of Semester 1
 - Alerting complete
 - Functional interface for visualising collected data
 - Deploy on Tardis and gain feedback from users
- By start of Semester 2
 - Work on additional features e.g. dependency support and distributed monitoring nodes
- Semester 2
 - Finish implementation
 - Documentation
 - Report

¹<http://www.tardis.ed.ac.uk/>

Questions?