COGSI - Configuração e Gestão de Sistemas

Mestrado em Engenharia Informática, Ramo Sistemas Computacionais

Class Assignment

P1 - Monitoring Networks and Systems

Alexandre Bragança atb@isep.ipp.pt

Dep. de Engenharia Informática – ISEP

2022/2023

P1: Overview

- **Start Date:** 1, March
- **End Date:** 15, March
- **Development Repository:** Your individual repository
  - Create Issue(s) for your sprint
  - Expected several commits (at least 1 for each lab class!)
  - You should commit to the repository only files that you created or edited (e.g., do not commit the nagios directory!)
  - Documentation should be provided **only in the readme.md file related to the assignment!**
- **Assignment Presentation and Review:**
  - Lectures: 15, March
- **If you are selected for presentation (sprint review)** then you should **pull request** the final version of your work into the class shared repository (`https://bitbucket.org/mei-isep/cogsi-22-23-class-rep/`) before the deadline.

P1: Topic

- The topic of this sprint is **Monitoring Networks and Systems**
- Your solution should be based on **Nagios**
- The overall objective is to simulate a simple monitoring scenario where:
  - There should be one virtual machine running the monitoring server (i.e., Nagios)
  - There should be at least one more virtual machine (a monitored machine)
  - The goal is to monitor several properties/services (e.g., current load, current users, total processes, free space, etc.) as well as the HTTP service
  - We will also want to include in the scenario the notification of contacts when some faults or recoveries occur
  - Also, the system should try to automatically recover the HTTP service
  - You should use Vagrant to support virtualization
    - You should aim at including the necessary installation and configuration steps in the section **provision** of the Vagrantfile.
    - Therefore, your solution should be based on virtual machines with a base "clean" linux distribution (e.g., envimation/ubuntu-xenial) that are customized using the **provision** feature of Vagrant in order to complete the assignment.

P1: Specific Concerns

Remote Monitoring

- The monitoring of some properties (e.g., disk free space) may require the installation of software in the monitored machine
- With Nagios one of the solutions for this issue is to use NRPE
- The monitoring can be done by "polling" or by "pushing". These may also be known as active or passive checks

Automatic Recovery

- You should install Tomcat in the monitored machine
- The Monitoring Server should try to automatically recover the Tomcat server when it is down. In Nagios, please refer to "event handlers" to support your solution
- Contact(s) should be notified by email when the service changes states (e.g., up, down, etc,). In Nagios you may use the **sendemail**.

Customization

- How the monitoring tool can be customized (e.g., using a different database, adding new features/plugins, setting different compiling options for optimization or security purposes, for instance, regarding NRPE)

P1: Alternative

The alternative for this assignment is to use other monitoring tool (to substitute Nagios).

There are several other alternatives that you can study, for instance:

- **Zabbix** (`https://www.zabbix.com`)
- **Icinga** (`https://icinga.com`)
- **Prometheus** (`https://prometheus.io`)

You should describe and compare the alternative with Nagios and implement some particular aspect of the requirements with the alternative monitoring tool.

P1: Technical Report

You should produce a technical report documenting the assignment.

- The technical report **must be produced** in the **readme.md** file located in the repository folder related to P1 (e.g., 1133224-maria-ferreira/p1/)
- The report should include:
  - The Analysis of the Problem
  - The Design of your Solution
    - Present an overview of the tools (e.g., software used, major concepts, major processes, architecture of the tools)
    - Present an overview of the solution (e.g., the architecture and major configurations required)
  - The Steps required to Reproduce your Solution (it should include references/links to configuration files, scripts or code included in the same folder of the repository)

You may also include:

- Justification of Design Options
- Analysis of the Alternative
- The Steps required to Reproduce the Alternative (i.e., implement the alternative)

P1: How to Submit to the Class Shared Repository

If you have been selected to make a presentation for this component you must share your work with the class using the shared repository.

- The shared repository is located in `https://bitbucket.org/mei-isep/cogsi-22-23-class-rep/`.
- You should make a fork of this repository.
- You should then clone the forked repository into your local computer.
- Copy to this repository only the folder where you developed P1 (e.g., 1133224-maria-ferreira/p1/).
- Commit and push the changes to the forked repository.
- In Bitbucket do a pull request against the original shared class repository in `https://bitbucket.org/mei-isep/cogsi-22-23-class-rep/`.
- The teacher will review your pull request and, once accepted, it will become available to all other students.