ORWELL

Monitorization Platform for a 5G Testbed

Research Group

IT - ATNoG

5GASP Project

Team

Rafael Direito

rafael.neves.direito@ua.pt

Diogo Gomes

dgomes@ua.pt

Team

Pedro Duarte
Team Manager

pedro.dld@ua.pt

Gonçalo Leal
Product Owner

goncalolealsilva@ua.pt

Alexandre Serras
Architect

alexandreserras@ua.pt

Vasco Regal
Devops

vascoregal24@ua.pt

Context

- Testbeds of 5G applications
- Work with 5G applications, networks and infrastructures
- Metrics of each VNF/CNF can be obtained through different tools
- Each tool has a way to store metrics
- Current monitoring tools are typically intrusive, limited and not centralized.

In our scenario:

- Intrusive means that the monitoring tool needs to log into the VNF, i.e. have access to it in order to run the metrics collection mechanisms
- Non-intrusive monitoring tools do not need access to the VNF

Problems

- Monitor network and netapps in the 5G testbed, giving the possibility of non-intrusive harvesting
- Analysis of metrics, creating logs, reports and alarms
- Different tools store metrics in different formats, we will harvest those different metrics and store them in an uniform storage
- Guarantee safety mechanisms inside of the testbed
- Present to app owners their respective reports as well as alert when irregular situations happen

Goals

- Support intrusive and non-intrusive metrics harvesting and different source tools
- Given that extraction, store those metrics in Prometheus - Required by the 5GASP consortium, in order to standardize the monitoring mechanisms
- Setup a Grafana instance for data visualization with privacy in mind, creating authentication and authorization mechanisms
- Integration of cybersecurity tools so as to detect possible malicious activities
- Guarantee availability and performance of the system given its high data volume

Calendar and Tasks

29/3

Explore Useful Tools
Prometheus, Grafana, Netdata, Telegraf, perfSONAR, ntop, SNMP, ...

Design Architecture

05/4

Choose Implementation Technologies

Initialize Development Environment

Create Service Discovery API

Explore Testbed/Network-level Metrics

Explore 5G Core-level Metrics

12/4

Create OS Cloud Images With Monitoring Tools

Explore Message Queues For Message Aggregation

Intrusive Tools via Cloud-init

Intrusive Tools via Juju Charms

19/4

Middleware Demo
Monitor 1 Prometheus instance via Push

Decouple Middleware's Data Source For Intrusive Tools

Implement Message Queues

Support Push and Pull Approaches

26/4

Explore Non-Intrusive APIs

Explore Network APIs

03/5

Accept Network And Non-Intrusive Data in Middleware

Explore Security Mechanisms

Calendar and Tasks

10/5

Implement Security Mechanisms

Implement Data Exporting

Create Development Environment For Frontends

17/5

Create Service Discovery Frontends

Create Requirements For Reports

24/5

Explore Alarms' Requirements

Explore Authorization Mechanisms For Grafana

Expore Middleware Minification

31/5

Create Technical Report

Implement Authorization In Grafana

Implement Alarms' Plugin

14/6

Rework Technical Report

Prepare students@deti Material

14/06

Prepare Final Presentation

Communication

GitHub Repository

Slack Channel

ClickUp

To keep up with our work visit our website: www.orwell.io

Expected Results

- End-to-end monitoring platform
- Support different metrics and collectors
- Possibility of adding new tools with ease
- Assure testbed’s security
- Guarantee the alarmistic of problematic situations

Expected Results

- ➔ End-to-end monitoring platform
- ➔ Support different metrics and collectors
- ➔ Possibility of adding new tools with ease
- ➔ Assure testbed’s security
- ➔ Guarantee the alarmistic of problematic situations

Related Work

A. Wolke and D. Srivastav, "Monitoring and Controlling Research Experiments in Cloud Testbeds," 2013 IEEE Sixth International Conference on Cloud Computing, 2013, pp. 962-963, doi: 10.1109/CLOUD.2013.97.

E. Magana, A. Astorga, J. Serrat and R. Valle, "Monitoring of a virtual infrastructure testbed," 2009 IEEE Latin-American Conference on Communications, 2009, pp. 1-6, doi: 10.1109/LATINCOM.2009.5305030.

M. Shirali, M. Sharafi, M. Ghassemian and F. Fotouhi-Ghazvini, "A Testbed Evaluation for a Privacy-Aware Monitoring System in Smart Home," 2018 IEEE Global Humanitarian Technology Conference (GHTC), 2018, pp. 1-7, doi: 10.1109/GHTC.2018.8601929.

Divneet Kaur, Bashir Mohammed and Mariam Kiran, “NetGraf: A Collaborative Network Monitoring Stack for Network Experimental Testbeds” arXiv:2105.10326v1 [cs.DC] 18 Mar 2021

Related Work

- A. Wolke and D. Srivastav, "Monitoring and Controlling Research Experiments in Cloud Testbeds," 2013 IEEE Sixth International Conference on Cloud Computing, 2013, pp. 962-963, doi: 10.1109/CLOUD.2013.97.
- E. Magana, A. Astorga, J. Serrat and R. Valle, "Monitoring of a virtual infrastructure testbed," 2009 IEEE Latin-American Conference on Communications, 2009, pp. 1-6, doi: 10.1109/LATINCOM.2009.5305030.
- M. Shirali, M. Sharafi, M. Ghassemian and F. Fotouhi-Ghazvini, "A Testbed Evaluation for a Privacy-Aware Monitoring System in Smart Home," 2018 IEEE Global Humanitarian Technology Conference (GHTC), 2018, pp. 1-7, doi: 10.1109/GHTC.2018.8601929.
- Divneet Kaur, Bashir Mohammed and Mariam Kiran, “NetGraf: A Collaborative Network Monitoring Stack for Network Experimental Testbeds” arXiv:2105.10326v1 [cs.DC] 18 Mar 2021

State of The Art - Exporters

Tool	Advantages	Disadvantages	Push/ Pull	Communication channel	Integration with Prometheus
SNMP	Standard and well known. Native to IP networks.	Poor large retrievals performance. Complex implementation on our use case.	Both	UDP	✔
Telegraf			Push	TCP	
Prometheus	Own time series database. Exporter to allow metric's pushing.	Has to be installed in the machine.	Both	HTTP	✔