Utu Hopiavuori

Supporting cybersecurity in web application infrastructure using Ansible

Utu Hopiavuori

Insinööri (AMK), Bachelor of Engineering

Degree Programme in Information and Communication Technology

Thesis

12.2.2025

**Abstract**

Author(s): Utu Hopiavuori

Title: Supporting cybersecurity in web application infrastructure using Ansible and ELK stack

Number of Pages: xx pages + x appendices

Date: 28 August 2020

Degree: Insinööri (AMK), Bachelor of Engineering

Degree Programme: Degree Programme in Information and Communication Technology

Specialisation option: Name of the specialisation option

Instructor(s): First name Last name, Title (e.g., Project Manager)

First name Last name, Title (e.g., Principal Lecturer)

The text section of the abstract is written to fit the space used on the page. The text section uses the **Body Text No Spacing** style.

Keywords: Keyword

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

The originality of this thesis has been checked using Turnitin Originality Check service.

**Contents**

[1 Abstract 1](#_heading=h.gjdgxs)

[2 Introduction 1](#_heading=h.30j0zll)

[2.1 Application platform serialization 1](#_heading=h.1fob9te)

[2.2 Ansible for Automation 1](#_heading=h.3znysh7)

[2.3 Alternative solutions 3](#_heading=h.2et92p0)

[3 Implementation 3](#_heading=h.tyjcwt)

[4 Conclusions 4](#_heading=h.3dy6vkm)

[References 5](#_heading=h.1t3h5sf)

[Appendices 6](#_heading=h.4d34og8)

[Title of the Appendix 6](#_heading=h.2s8eyo1)

**Abbreviations**

# Abstract

# Introduction

## Application platform serialization

I got a project to develop an infrastructure automation script set to verify that a web application platform would be complete to serve the application components: backend and frontend. There were a couple of requirements to validate if the job gets done.

1. OS level updates are applied
2. required software is installed and configured properly
   1. apache + php on frontend servers
   2. apache + php + mysql on backend servers
   3. apache modsecurity is installed, confugred and in use
3. logreader and general admin users are created, and proper access are granted
   1. passwords should be reset if needed and emailed to admin user of company
4. SSHd port is randomized and could be re-randomize if needed

We had plans to add SIEM capabilities to monitor the servers and applications and react if security incidents happen. In this work I will describe the details I got done during the project and let the SIEM part happen sometimes in the undefined future. I could conceptualize a bit how the job would get done using Ansible skills I have learned so far.

## Ansible for Automation

## Alternative solutions

The main competitors or alternative approach to configuration management, infrastructure automation and services orchestration are Puppet, Chef and Salt (Venezia, 2013).

# Implementation

Ansible playbook quality is often described by Molecule test library sequence. I didn’t implement full automation including instance creation and old infrastructure destroy but trying to be careful with idempotence and side effects.

A screenshot of a computer

Description automatically generated

Figure 1 Molecule test life cycle (not implemented fully in work)

Syntax check was done by ansible-lint. It seems to have problems with secure variables reporting errors when the variable is defined using multi-line syntax of YAML.

# Conclusions

**References**

Harvard-tyylinen lähdeviittaus

Hochstein, L., Meijer, B. & Moser, R. 2022. Ansible: Up and Running, 3rd Edition.

Anshul Behl 2023. Introducing Ansible Molecule with Ansible Automation Platform   
<https://developers.redhat.com/articles/2023/09/13/introducing-ansible-molecule-ansible-automation-platform#molecule_scenarios>

Venezia, Paul 2013. Review: Puppet vs. Chef vs. Ansible vs.Salt  
<https://www.infoworld.com/article/2186089/data-center-review-puppet-vs-chef-vs-ansible-vs-salt.html>

**Appendices**

**Title of the Appendix**