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

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

Table of Contents

[1. Abstract 1](#_Toc191289181)

[2. Introduction 1](#_Toc191289182)

[2.1. Application platform conditioning 1](#_Toc191289183)

[2.2. Ansible for Automation 1](#_Toc191289184)

[2.3. Alternative solutions 2](#_Toc191289185)

[3. Implementation 3](#_Toc191289186)

[3.1. Automation idempotence 3](#_Toc191289187)

[4. Conclusions 3](#_Toc191289188)

[References 4](#_Toc191289189)

[Appendices 5](#_Toc191289190)

[Source code snippets 5](#_Toc191289191)

**Abbreviations**

# Abstract

# Introduction

## Application platform conditioning

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

Originally, I did the work in part of project work, and I had plans to add SIEM capabilities to monitor the servers and applications as well as react properly if security incidents happen. In this work I will describe the details I got done during the project and let the SIEM part happen 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

According the Ansible documentation, Ansible is an open-source automation toolset which uses simple, human-readable scripts called playbooks to automate tasks. As an administrator I declare the desired state of infrastructure in Ansible playbooks and roles. Ansible is developed to ensure that system remains in that state (Ansible project contributors, 2025).

Ansible playbooks is usually written using YAML or JSON markup languages. During the work I was using YAML formatted playbooks.

## Alternative solutions

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

# Implementation

## Automation idempotence

When working with original requirements, the idempotence was a quite hard problem because the needs to randomize the OpenSSH server port and creation of password for newly created users or existing users.

Idempotence means that if playbook has been running multiple times the state of infrastructure should remain unchanged (Behl, 2023). Random port allocation for the OpenSSH server was easy to get done but idempotence required the task being optional (Figure 1)

Another problem with OpenSSH server port randomization was the fact that automation was done using OpenSSH connection between Ansible controller and the leaf server. I had to verify that server firewall was allowing connection for randomized port (Figure 2).

# Conclusions

References

Ansible project contributors, 2025. *Introduction to Ansible.* [Online]   
Available at: https://docs.ansible.com/ansible/latest/getting\_started/introduction.html  
[Accessed 22 02 2025].

Behl, A., 2023. *Introducing Ansible Molecule with Ansible Automation Platform.* [Online]   
Available at: https://developers.redhat.com/articles/2023/09/13/introducing-ansible-molecule-ansible-automation-platform  
[Accessed 17 02 2025].

Meijer, B., Hochstein, L. & Moser, R., 2022. In: *Ansible: Up and Running, 3rd Edition.* s.l.:O'Reilly Media, Inc.

Venezia, P., 2013. *Review: Puppet vs. Chef vs. Ansible vs. Salt.* [Online]   
Available at: https://www.infoworld.com/article/2186089/data-center-review-puppet-vs-chef-vs-ansible-vs-salt.html  
[Accessed 17 02 2025].

Appendices

Source code snippets

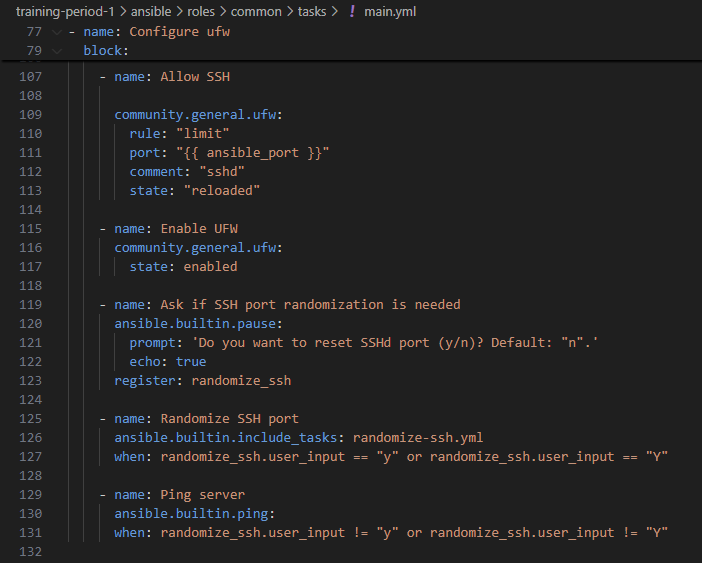


Figure 1: Conditional SSH randomization for idempotence.

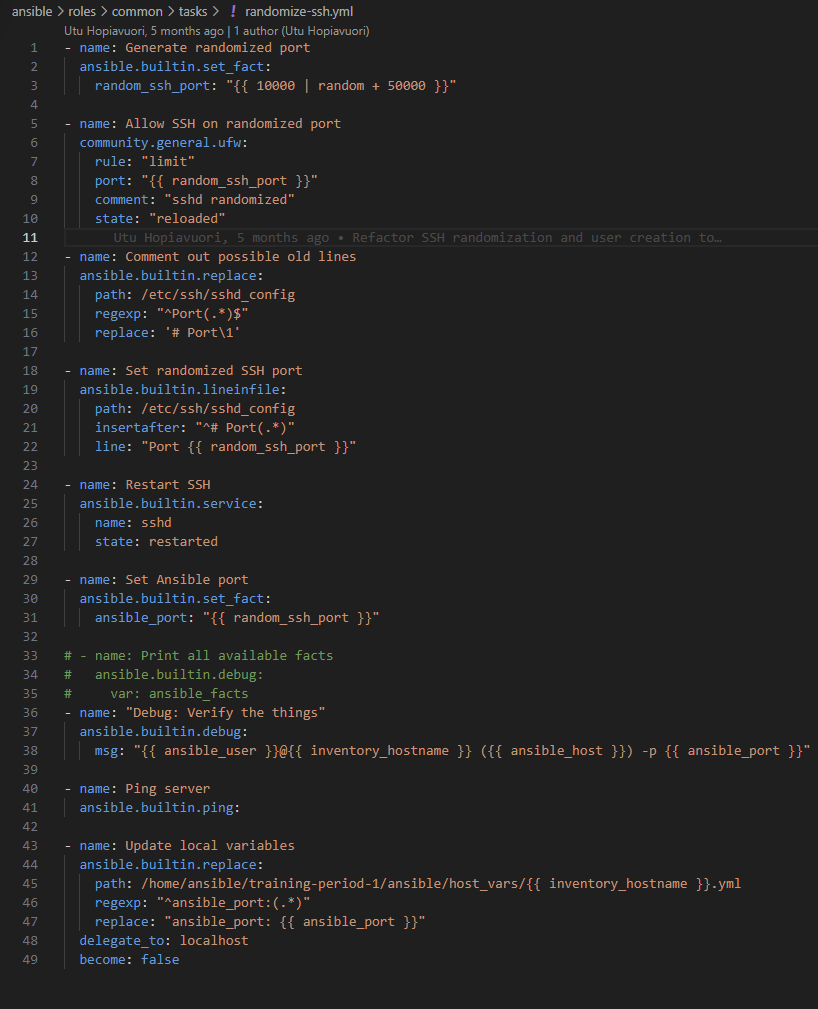


Figure 2 Source code for SSH port randomization.