KTH Royal Institute Of Technology

School of Electrical Engineering and Computer Science

Automating Infrastructure Management in CTF Games with Ansible

April 27, 2023

Georgios Akkogiounoglou
gakk@kth.se

Abstract

This essay will explore the use of Ansible, which is an open-source automation tool, for managing infrastructure in the context of a Capture The Flag 'CTF' game. It will provide an overview of Ansible, its features, and the benefits of Infrastructure as Code 'IaC' and how Ansible supports IaC workflows. This essay will also dig into Ansible's uses for automating various components of a CTF game. Overall, this essay provides an completely understanding of how Ansible can be used to improve infrastructure management processes and support DevOps practices in the context of a CTF game.

Introduction

Cyber security employees are using Capture The Flag games to test and improve their skills, in a safe simulated environment. CTF games require participants to find and exploit vulnerabilities in various aspects of a simulated network or system. In that way, Cyber security employees get hands-on experience on exploiting various vulnerabilities. However, managing the infrastructure of a CTF game could be complex and time-consuming especially for large networks.[5]
To face these problems, DevOps teams developed Ansible which is an automation tool to manage their CTF infrastructure more efficiently. Ansible is an open-source automation tool used in the DevOps community on various IT tasks, incuding configuration management and application deployment. Infrastructure as Code 'IaC' has become an important value in DevOps, and Ansbile is very well-suited for IaC workflows, making it an essential tool for many organizations.[1]
This essay will explore how Ansible can automate the infrastructure management in CTF games from a DevOps perspective. It will explain how Ansible works, and the benefits of it on IaC and an overview of workflows and tools. In addition, case studies of organizations that have successfully implemented Ansible for automation and IaC in the context of CTF games will also be provided. The benefits and challenges are going to be summarized and insights will be provided into how it can be used in different types of CTF games.[6]
Finally, this essay demonstrates how Ansible is a valuable tool for any organization which is looking to improve their infrastructure management and adopt DevOps practises in their CTF games.

Overview of Ansible

Ansible is an automation tool that is very popular among DevOps community. It is an open-source tool that can be used to automate various IT tasks, including configuration management, software deployment, and orchestrate workflows. It was created by Michael DeHaan in 2012 [8]. Ansible uses a simple, human-readable language to describe automation tasks, which makes it easier for the users, to learn and use it.
Simplicity is the key feature of Ansible. It doesn't require to install agents on nodes, instead of that it uses SSH to connect remote nodes and execute tasks. As a result, the deployment and managing is becoming easier and it also reduces the risk of breaches.[9]
Ansible is designed to be idempotent. This means that its tasks can be run multiple times and it will not make any unwanted changes. A task is also analytical, which means that it describes the desired state that is wanted in a system rather than the steps to get there.

Regarding how it's working and its architecture, it is based on a client-server architecture, and its controlling node is located to the machine that Ansible is running from. The managing nodes are the machines that Ansible is managing. For the controlling of the nodes it is using SSH to connect to them, and it executes tasks on them using modules.
Basically, modules are short scripts that perform tasks on the nodes. They can be used to perform tasks like installing new software, copying files, managing users and groups etc. Those scripts are written in Python.[9]
Last but not least, Ansible is using a simple inventory system to manage nodes. The inventory is a list of managed nodes that Ansible is managing. Usually, the inventory is stored in a simple text file, or in a dynamic inventory system like AWS, Azure or Google Cloud.[9]
As the advantages of using Ansible for automation can be many. First of all, the simplicity of it, makes it easy to learn and use. It uses simple, human readable language to describe the automation of the tasks. Ansible has also very good documentation and a large community of users who can help answering questions and provide support if its needed.[2]
Nevertheless, Ansible is agent-less, which means that it doesn't require agents to be installed on the nodes. This makes it much easier to deploy and manage Ansible, and reduce the risk of security breaches.[15]
Finally, it is platform-independent, which means that it can be used to manage a wide variety of systems, including Linux, macOS, Windows etc. This makes it a flexible tool that can be used in many different environments.[9]

Infrastructure as Code (IaC)

Infrastructure as Code 'IaC' is the key to DevOps, which involves managing and testing infrastructure resources as code. IaC provides a lot of benefits, such as efficiency, quicker time-to-market, and reduction of errors. Digging into the advantages of IaC and how Ansible can be used to support IaC workflow, is going to be analyzed in manner of a Capture The Flag game.
IaC is a methodology that improves the management of infrastructure components, such as servers, networks, and databases. This can has an outcome, better control and automation of the efficient management process. IaC can also be used to configure components such as servers and networks, which can lead to optimization of management and deployment.[16]
To implement IaC, there are various tools available online, such as Terraform, Ansible, Chef, and Pulumi. Ansible could particularly be better-suited for IaC workflows in manner of a CTF game. Using Ansible in a CTF game can have several advantages, such as the minimizing of the risk to unintended changes and reducing the chances of downtime or data loss. Ansible can ensure an easy and consist management of the Cyber range, so the game can run smoothly. In addition, through Ansible easy rollbacks can be use to reset or restore servers in case of failures, reducing in that way the downtime.[12]
As a result, Iac is valuable in DevOps and it provides efficiency and consistency of management of the infrastructure resources through code.

Ansible Use Cases in CTF Games

The popularity of Capture The Flag games among the cybersecurity community has grown as it allows players to test and improve their skills and knowledge in a safe simulated environment. These games includes challenges that allow players to solve various security puzzles and complete specific objectives while defending or attacking assets of them or from other players. However, this demands a well-managed infrastructure to support such a game.

Ansible's use case in CTF game is automating the deployment of various infrastructure components such as servers, web apps, and databases. Here comes Ansible playbooks, which a desired state of the infrastructure can be defined and the automation of the deployment is going to be processed. In that way, time is saved, a lot of effort is reduced and human errors are minimized. In that way, the deployment of a new game server can be organized in just a few lines of code, reducing deployment time from days to minutes.[3]

Ansible's idempotent nature, can ensure that all changes are consistent across the infrastructure. This can assist in the roll-back feature of Ansible if a change leads to unexpected outcomes, ensuring that way the infrastructure remains stable and operational.[10]

Overall, the declarative approach and idempotent nature of Ansible are ideal for efficiently managing infrastructure in a fast-moving and constantly changing infrastructure for a CTF game. After adopting Ansible, it can improve the infrastructure management processes, increase efficiency and reduce errors in the context of CTF games

Listing 1: Ansible Playbook Example

```
---
- name: Update web servers
  hosts: webservers
  remote_user: root

  tasks:
  - name: Write the apache config file
    ansible.builtin.template:
      src: /srv/httpd.j2
      dest: /etc/httpd.conf

- name: Update db servers
  hosts: databases
  remote_user: root

  tasks:
  - name: Ensure postgresql is at the latest version
    ansible.builtin.yum:
      name: postgresql
      state: latest

  - name: Ensure that postgresql is started
    ansible.builtin.service:
      name: postgresql
      state: started
```

Ansible Best Practices for CTF Games

When using Ansible in a CTF game, there are some many best practices that can be followed to ensure the smooth and efficient management of it. Git is the best to use for version control, in order to track changes Ansible playbooks and other configuration files. This facilitates collaboration between team members who are working on the same infrastructure.[4]
Another best practice is to organize Ansible code in a logical way, by using roles and playbooks to define specific tasks. This makes it easier to manage and update the infrastructure, as well as the testing and problem troubleshooting.[7] Regarding testing, it is crucial to test Ansible playbooks thoroughly before deploying them to production.[14]
In addition, a best practice can also be to integrate Ansible with other DevOps tools, such as Jenkins. Jenkins can be used to automatically trigger Ansible playbooks in response to certain eventsm such as code changes or new commits to the repository.[11]
One of the common mistakes to avoid it, is to create complex Ansible playbooks that can cause confusion and mistakes. To prevent this, it is important to keep playbooks simple and modular, with clear documentation and comments for each task.
Another mistake to avoid when using Ansible for a CTF game, is not paying attention to the security considerations. To keep the infrastructure secure, access controls, and authentication mechanisms must be in place. This can help prevent unauthorized access to infrastructure resources and protect sensitive data.[13]
In general, using Ansible in the context of a CTF can greatly improve the infrastructure management processes, increasing efficiency and reducing errors. Best practices such as version control, code organization, testing and integration with other DevOps tools, can make the most out of Ansible's capabilities. However, it's necessary to be aware of common mistakes and take appropriate security measures to ensure safe and effective use of it.

Listing 2: Ansible Roles Example

```
roles/
    common/               # this hierarchy represents a "role"
        tasks/            #
            main.yml      #  <-- tasks file can include smaller files if warranted
        handlers/         #
            main.yml      #  <-- handlers file
        templates/        #  <-- files for use with the template resource
            ntp.conf.j2   #  <------- templates end in .j2
        files/            #
            bar.txt       #  <-- files for use with the copy resource
            foo.sh        #  <-- script files for use with the script resource
        vars/             #
            main.yml      #  <-- variables associated with this role
        defaults/         #
            main.yml      #  <-- default lower priority variables for this role
        meta/             #
            main.yml      #  <-- role dependencies
        library/          # roles can also include custom modules
        module_utils/     # roles can also include custom module_utils
        lookup_plugins/   # or other types of plugins, like lookup in this case

    webtier/              # same kind of structure as "common" was above, done for
        the webtier role
    monitoring/           # ""
    fooapp/               # ""
```

Conclusion

In this rapidly evolving world of Cyber security, the use of CTF games has become a important way for individuals and organizations to test their skills and knowledge. However, a well-managed infrastructure to support such games is needed. That's when Ansible comes to, as it can help automate the deployment and management of various components in order to save time, effort and errors.
Benefits of Ansible are many, one of them is its declarative approach, which is regarding the desired state of the infrastructure that is defined and making it easier to be managed and maintained. For such a fast-paced and constantly changing environment as a CTF game, these features of Ansible is ideal for the management of it.
It is necessary to keep the best practices in mind to avoid common mistakes though. For that reason keeping Ansible playbooks simple and modular, in order to avoid confusion and errors. Notwithstanding, keeping the documentation clear and comments for each task is the key. In order to prevent unauthorized access to resources and protect sensitive data, it is important to ensure that Ansible is used securely.

In conclusion, Ansible is a powerful automation tool, and can be very effective for managing the infrastructure of a CTF game. The benefits of Ansible can make it a well-fit for managing such a fast-moving environment. The best practices and keeping the security considerations always in mind, can improve the management of the infrastructure management, increase efficiency and reduce errors. Ansible is a valuable tool for everyone who want to adopt DevOps practices and improve the management processes in the context of a CTF game.

Bibliography

[1] *Ansible (Software)*. URL: https://en.wikipedia.org/wiki/Ansible_(software)/.

[2] *Ansible Community Documentation*. URL: https://docs.ansible.com/.

[3] *Ansible Playbooks*. URL: https://docs.ansible.com/ansible/latest/playbook_guide/playbooks_intro.html.

[4] *Best Practises*. URL: https://docs.ansible.com/ansible/2.8/user_guide/playbooks_best_practices.html#version-control.

[5] *Capture The Flag (Cyber Security)*. URL: https://en.wikipedia.org/wiki/Capture_the_flag_(cybersecurity).

[6] *Capture the Flag(CTF): The game for developers to learn information security*. URL: https://nulab.com/learn/software-development/capture-the-flag-ctf-game-developers-learn-information-security/.

[7] *Group By roles*. URL: https://docs.ansible.com/ansible/2.8/user_guide/playbooks_best_practices.html#group-by-roles.

[8] *How Ansible got started and grew*. URL: https://opensource.com/article/21/2/ansible-origin-story.

[9] *How Ansible works*. URL: https://www.ansible.com/overview/how-ansible-works.

[10] *Idempotent nature of Ansible*. URL: https://www.linkedin.com/pulse/idempotent-nature-ansible-naresh-kumar/.

[11] *Manege Jenkins jobs*. URL: https://docs.ansible.com/ansible/2.8/modules/jenkins_job_module.html.

[12] *Run Ansible Restore Playbook Remotely*. URL: https://docs.starlingx.io/backup/kubernetes/system-backup-running-ansible-restore-playbook-remotely.html.

[13] *Security Compliance*. URL: https://www.ansible.com/use-cases/security-and-compliance.

[14] *Testing Strategies*. URL: https://docs.ansible.com/ansible/2.8/reference_appendices/test_strategies.html.

[15] *The Benefits of Agentless Architecture*. URL: https://www.ansible.com/hubfs/pdfs/Benefits-of-Agentless-WhitePaper.pdf.

[16] *What is Infrastructure as Code (IaC)?* URL: https://www.redhat.com/en/topics/automation/what-is-infrastructure-as-code-iac.