



# Load Balancing

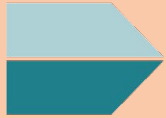
Cybersecurity  
Cloud Security Day 3



# Class Objectives

---

By the end of today's class, you will be able to:



Create an Ansible Playbook to configure VMs.



Create a load balancer on the Azure platform.



Create firewall and load balancer rules to allow traffic to the correct VMs.

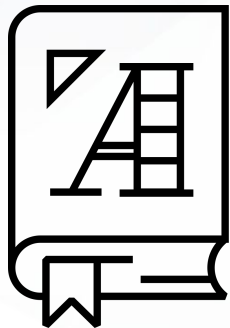
# Ansible Playbooks



We have implemented a jump box  
that is running an Ansible container.

---

The Ansible container has full access to our  
VNet and can connect with our new VM. Now  
we will write code that will be “infrastructure as  
code” for this vulnerable web server.



Ansible reads **YAML** code.

YAML stands for *YAML ain't markup language* and is designed to be very readable and easy to use.



Today, we'll get started  
with a guided tour of YAML.

# YAML

Today, we'll get started with a guided tour of YAML.

```
---  
  
- name: My first playbook  
  hosts: webservers  
  become: true  
  tasks:  
  
- name: Install apache httpd (state=present is optional)  
  apt:  
    name: apache2  
    state: present
```

Ansible reads  
YAML code.



# Instructor Demonstration

## YAML Guided Tour





## **Activity:** Ansible Playbooks

In this activity, you will create an Ansible playbook that installs Docker and configures a VM with the DVWA web app.

**Suggested Time:**  
25 Minutes





**Time's Up!** Let's Review.

# Load Balancing



So far, we created a virtual network, deployed a jump box running an Ansible Docker container, and used that container to configure another VM running a DVWA container.

# Load Balancing

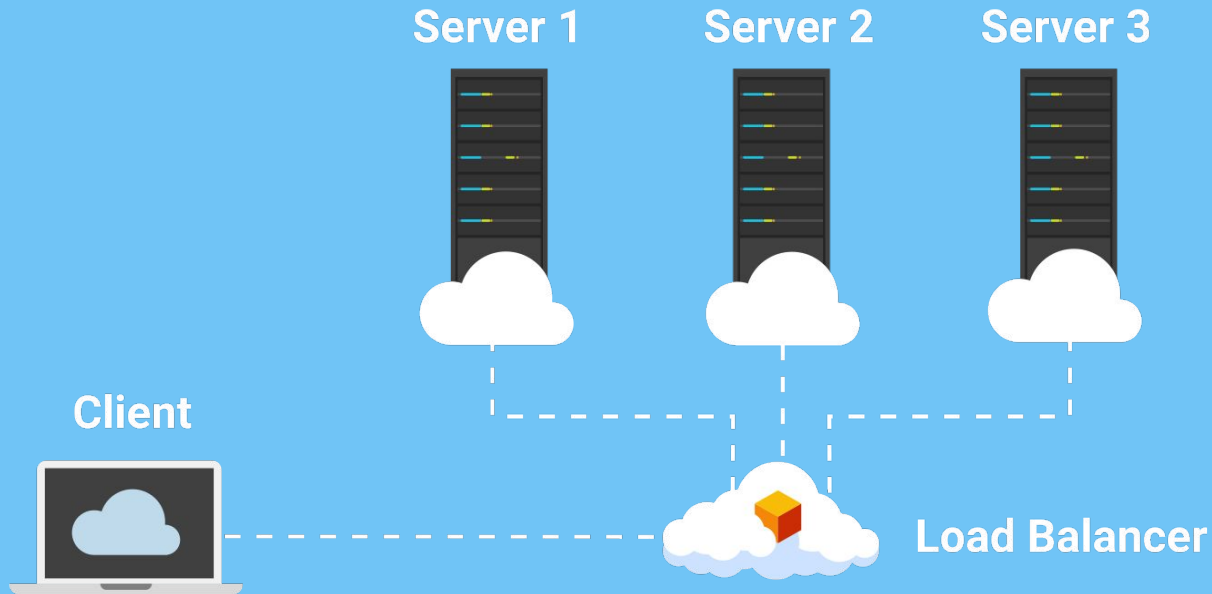
---

If the Red Team attacks this DVWA container with enough traffic, they may be able to trigger a Denial of Service on the machine.



# Load Balancers

A load balancer provides the external IP address that the rest of the internet can access. Then, it receives traffic that comes into the website and distributes it across multiple servers.

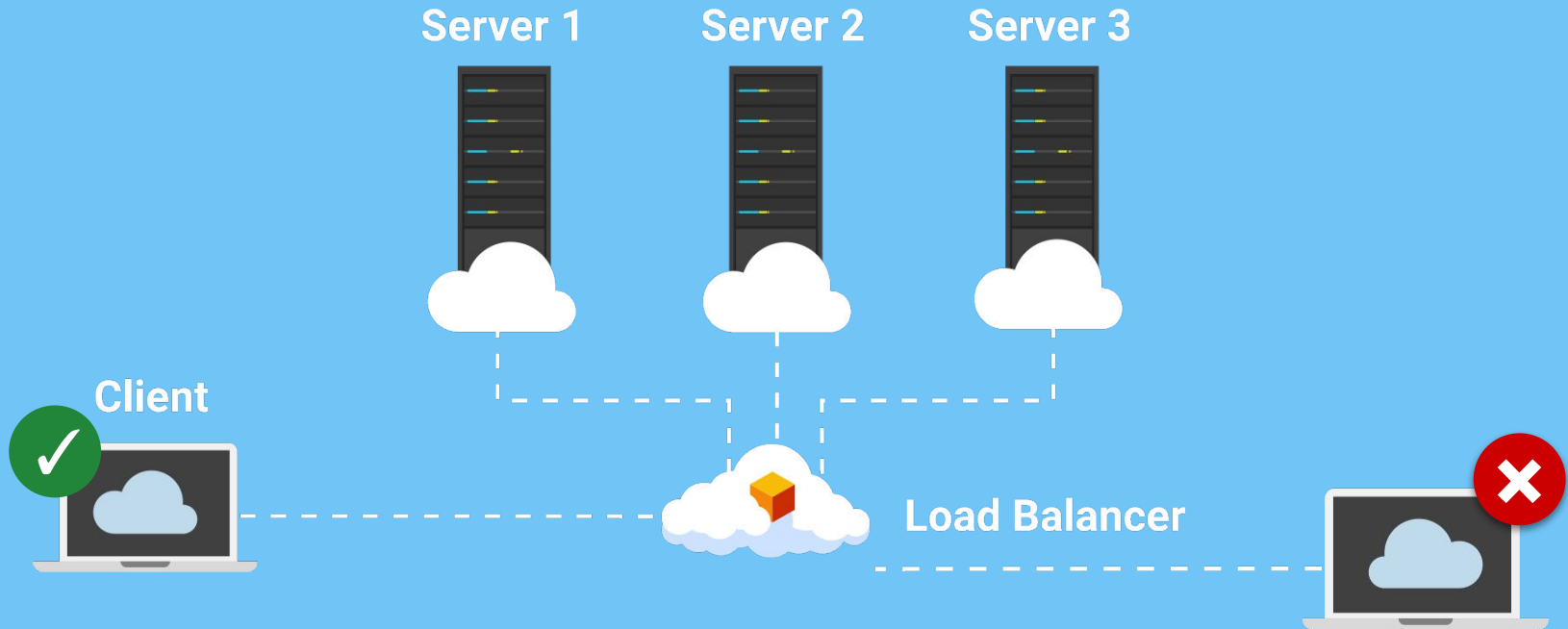


As websites receive more traffic, more servers can be added to the group ("pool") of servers that the load balancer has access to.

This helps distribute traffic evenly across the servers and mitigates DoS attacks.

# Load Balancers

Load balancers offer a **health probe** function to regularly check all the machines behind the load balancer. Machines with issues are reported, and the load balancers stop sending traffic to those machines.





The DVWA VM we set up is not accessible from the internet at this time. This is intentional.

The next step is to set up a load balancer that has an external IP, and point it to the VM.





# Instructor Demonstration

## Setting Up a Load Balancer



## **Activity:** Load Balancing

In this activity, you will install a load balancer in front of the VM to distribute the traffic across more than one VM.

**Suggested Time:**  
20 Minutes





**Time's Up!** Let's Review.

A close-up, high-angle shot of a computer keyboard. The central focus is a large, white, rectangular key with rounded corners. On this key, there is a dark blue icon of a coffee cup with three wavy lines above it representing steam. Below the icon, the word "Break" is printed in a dark blue, serif font. The key is set against a light-colored, textured keyboard surface. Surrounding the main key are other keys, including one with a double quote symbol to the left and one with a dash/slash symbol to the right, all slightly out of focus.

Break

# Firewall Configuration



Now that we have a load balancer running, we want to make sure it is configured properly to allow traffic to the VM backend pool.

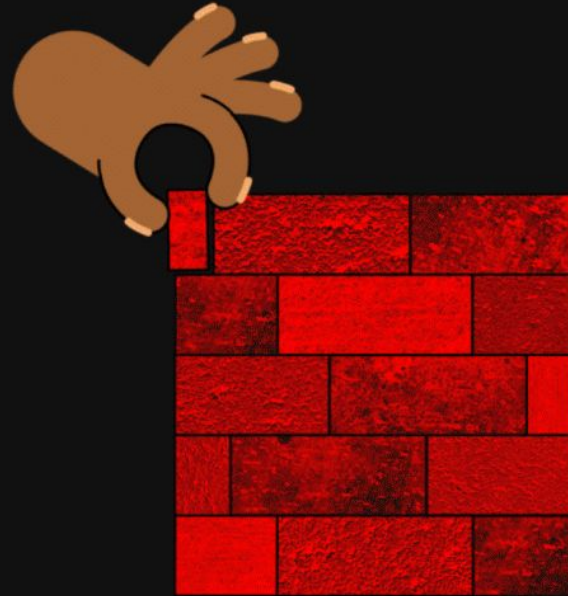
By the end of the next activity, we will be able to reach the DVWA website from the internet.

# Firewall Configuration

---

We need to configure a security group to allow web traffic into the VNet from the load balancer.

In the following guided tour, we'll create a load balancing rule.





# Instructor Demonstration

## Load Balancing Rule





## **Activity:** Security Configuration

In this activity, you will configure the load balancer and security group to work together to expose port **80** of the VM to the internet.

**Suggested Time:**  
20 Minutes





**Time's Up!** Let's Review.

# Daily Checklist

---

By the end of today, you should have completed the following critical tasks:



An Ansible playbook has been created that configures Docker and downloads a container.



The Ansible playbook is able to be run on the Web VMs.



The Web VMs are running a DVWA Docker container.

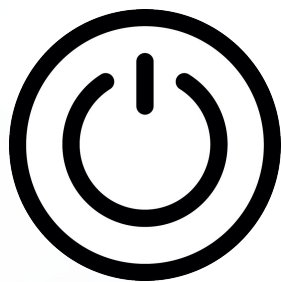


A load balancer has been created and at least two Web VMs placed behind it.



The DVWA site is able to be accessed through the load balancer from the internet.

---



**Don't forget to power off your machine!**

*The  
End*