



Project Week: ELK Stack

Cybersecurity
Project Week 1





Welcome to Project Week!

This week, you will set up an ELK stack server to monitor your cloud network.

Project Week 1: ELK Stack

Placing an ELK monitoring stack within your virtual network will allow you to:

01

Easily collect logs from multiple machines into a single database.

02

Quickly execute complex searches, such as:

Find the 12 internal IP addresses that sent the most HTTP traffic to my gateway between 4 a.m. and 8 a.m. in April 2019.

03

Build graphs, charts, and other visualizations from network data.

ELK Stack

- Deploying and configuring an ELK stack is a common task for network engineers.
- SOC analysts and other security professionals use it often.

Completing this project will provide convincing proof of your skills, which you can present to hiring managers.



ELK Stack

- The ELK stack is commonly used in network production.
- You'll likely work for organizations that use either ELK or Splunk, covered later in the course.
- Experience with both tools is a valuable addition to any job application.



ELK Stack

You can expand this network with additional machines on your own time to generate a lot of interesting log information.

This sort of independent research is useful for learning, and hiring managers love to see it.



Project Week 1: ELK Stack

You'll develop the following deliverables, which you can present in job interviews:

01

Network Diagram

An architecture diagram describing the topology of your network.

02

Technical Brief

Answers to a series of questions explaining the important features of the suite, completed after deploying the stack.

03

GitHub Repository

When complete, you will save your work to a Git repository, along with an in-depth description. This makes it easy to redeploy your work in the future, and share it with others.

The ELK Stack

ELK

ELK is an acronym. Each letter stands for an open-source technology:



Elastic Stack

These tools are collectively known as **ELK stack**.



Search and analytics engine.

Server-side data processing pipeline that sends data Elasticsearch.

Tool for visualizing Elasticsearch data with charts and graphs.



- ELK started with Elasticsearch.
- It was initially designed to handle *any* kind of information. This means that logs and arbitrary file formats, such as PCAPs, can be easily stored and saved.



- After Elasticsearch became popular for logging, Logstash was added to make it easier to save logs from different machines into the Elasticsearch database.
- Logstash also processes logs before saving them, to ensure data from multiple sources has the same format before it is added to the database.



- Since Elasticsearch can store so much data, analysts often use visualizations to better understand the data at a glance.
- Kibana is designed to make it easy to visualize massive amounts of data in Elasticsearch.
- Kibana is known for its complex dashboards.

The Beats Family

Beats

The ELK stack works by storing log data in Elasticsearch with the help of Logstash.

- While functional, this approach is not ideal because it requires administrators to collect all data reported by tools like `syslog`, even if they only need a small portion of it.

For example: Administrators often need to monitor changes to specific files, such as `/etc/passwd`, or track specific information, such as a machine's uptime.

In cases like this, it is wasteful to collect all of the machine's log data in order to only inspect a fraction of it.

Beats

Recently, ELK addressed this issue by adding an additional tool to its data collection suite, called **Beats**.

- Beats are special-purpose data collection modules. Rather than collecting all a machine's log data, Beats allow you to collect only the very specific pieces you're interested in.
- ELK officially supports eight Beats. We will use two of them in this project:
 - **Filebeat** collects data about the file system.
 - **Metricbeat** collects machine metrics, such as uptime.



beats

Project Overview

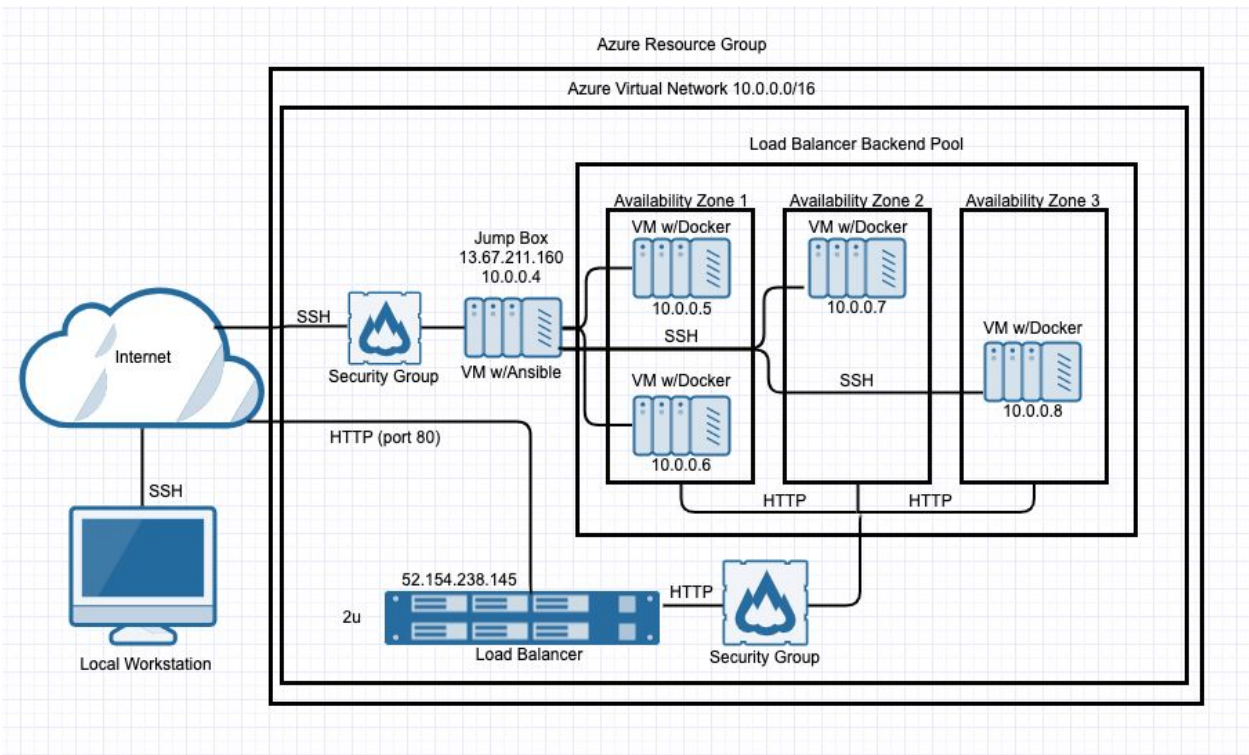


Now it's time to begin deploying.

Make sure you're logged into
your **personal Azure account**, *not*
your cyberxsecurity account.

Project Setup

We'll continue to build off the cloud week architecture.



This network has:

A gateway: the jump box configured during the cloud week.

Three additional VMs: one configuring the others, and two functioning as load-balanced web servers.



Project Milestones



Day 1: Configure the ELK server.



Day 2: Install Filebeat and Metricbeat.



Day 3: Finish leftover work, and create a network diagram and documentation.

Day 1: Configure the ELK Server

The rest of today will consist of the following:

01

Create a VM. Deploy a new VM onto the network to host the ELK server.

02

Download and configure the container. Download and configure the elk-docker container on the new VM.

03

Launch and expose the container. Launch the elk-docker container to start the ELK server.

04

Implement identity and access management. Configure your preexisting security group so you can connect to ELK via HTTP and view it through the browser.



Day 1 Activity: ELK Installation


For the remainder of class, you will work on the ELK installation, configuration, launch.

Suggested Time:
Full Class Time



Time's Up

By the end of this class, you should have completed the following:



Deployed a new
VM on your
virtual network.

Created an
Ansible play to
install and
configure an
ELK instance.

Restricted
access to the
new server.

Completing these steps required you to leverage your systems administration, virtualization, cloud, and automation skills. This is an impressive set of tools to have in your toolkit.

Day 2: Filebeat and Metricbeat



You completed installing the ELK server and will now install data collection tools called Beats.

If you have not completed all Day 1 activities, you can continue working on those tasks.

Filebeat

Filebeat helps generate and organize log files to send to Logstash and Elasticsearch. Specifically, it logs information about the file system, including which files have changed and when.



FILEBEAT

- Filebeat is often used to collect log files from very specific files, such as those generated by Apache, Microsoft Azure tools, the Nginx web server, and MySQL databases.
- Since Filebeat is built to collect data about specific files on remote machines, it must be installed on the VMs you want to monitor.



Day 2 Activity: Filebeat and Metricbeat

Today, you will install Filebeat on the DVWA container you created during the cloud week.

This will provide a rich source of logs when you complete your deployment.


If you have time, you can also install Metricbeat.

Suggested Time:
Full Class Time



Time's Up

By the end of this class, your ELK server should be receiving logs. You have:

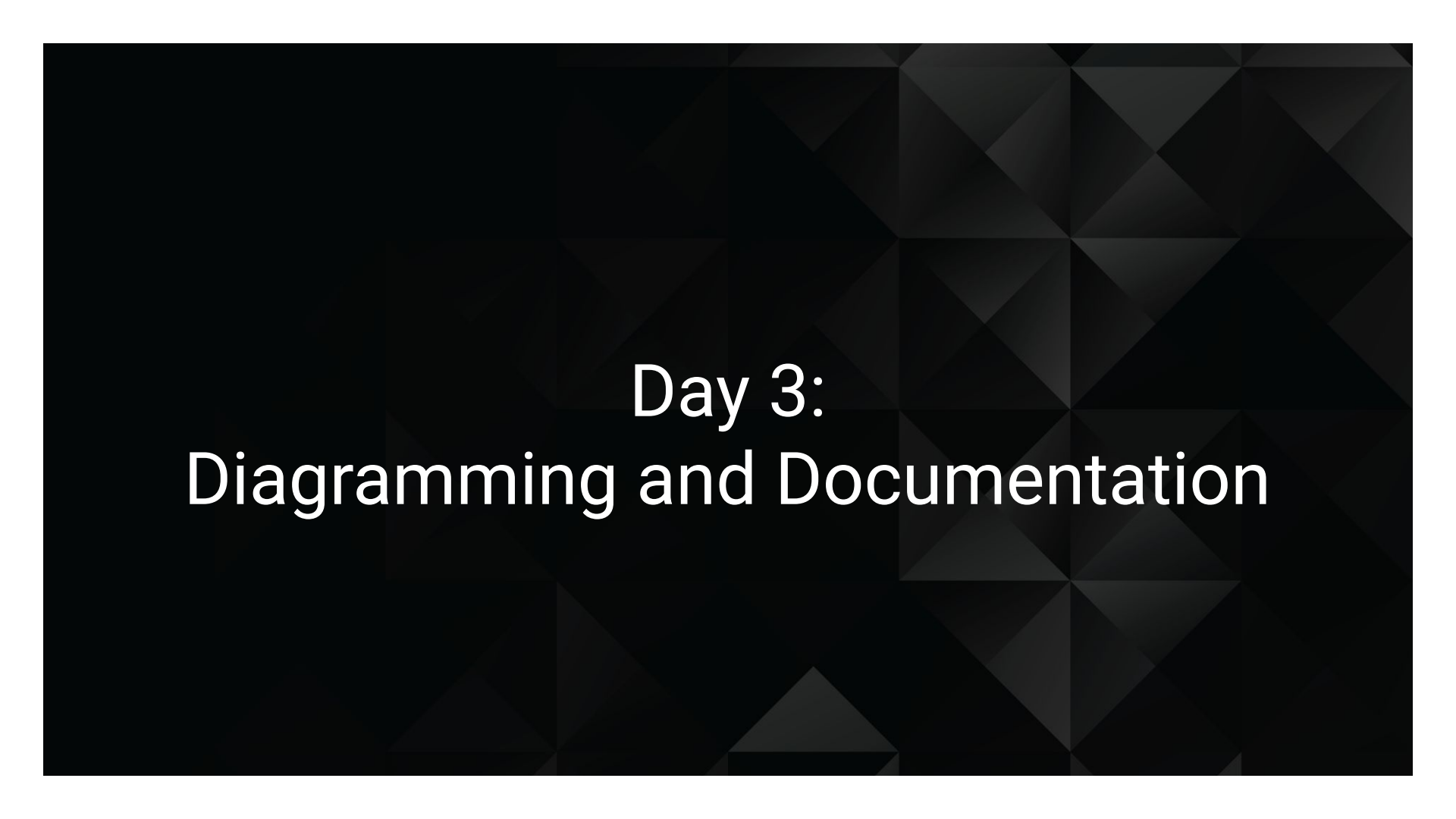


Installed and
launched Docker
containers to a
host machine.

Configured and
deployed an ELK
server.

Installed
Filebeat on a
Linux server.

(Completing the Metricbeat installation was a similar process.)

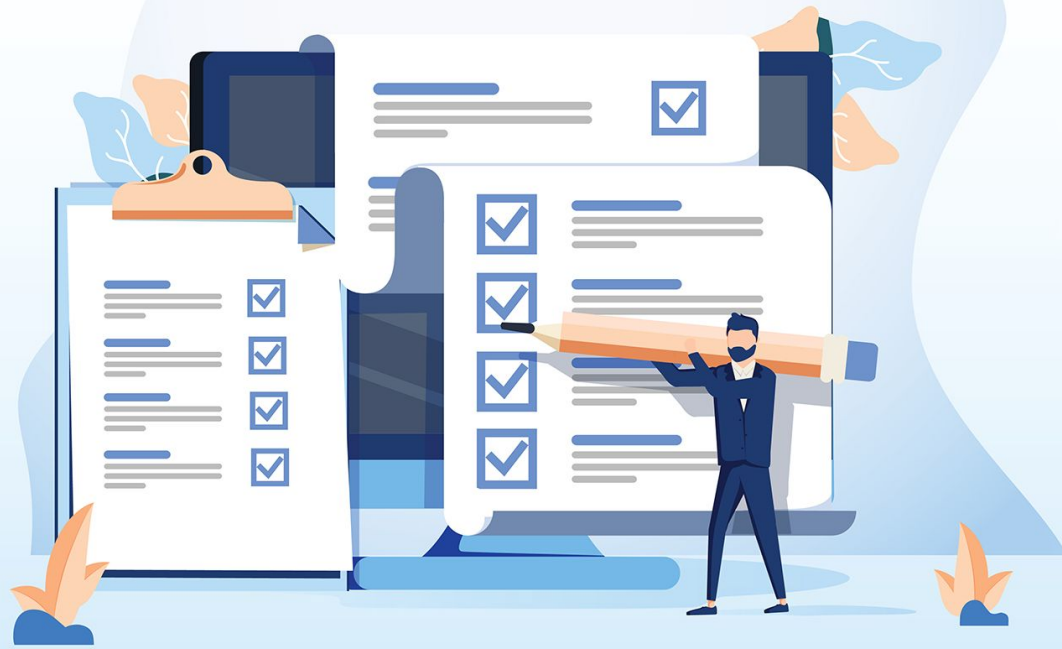


Day 3: Diagramming and Documentation



In the final day of the project, you will create a network diagram of your new setup and complete a README to document the basic information of the project.

Along with the GitHub repository you will complete for homework, the project documentation and diagram will be valuable **deliverables** to show **employers**, proving knowledge and experience.





Day 3 Activity: Network Diagramming and README

Today, you will finalize the network diagram you began in the cloud week to include the ELK stack.

You will also draft a README file documenting the network you built.

Suggested Time:
Full Class Time



Homework

Create a GitHub repository where you will save your project files and the README. You can use this repo to easily share the following with colleagues and employers:

01

Network diagram

02

Description of the deployment

03

Tables specifying access policies and network addresses

04

Usage instructions



Don't forget to power off your machine!

- Navigate to portal.azure.com.
- Search for and select **Virtual Machines**.
- Select every VM in the list.
- Click **Stop**. This will ensure you're not charged for any of the machines used in the project.