



# 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.

# Day 1: Configuring an ELK Stack

# 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. This brief is often referred to as a README document.

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

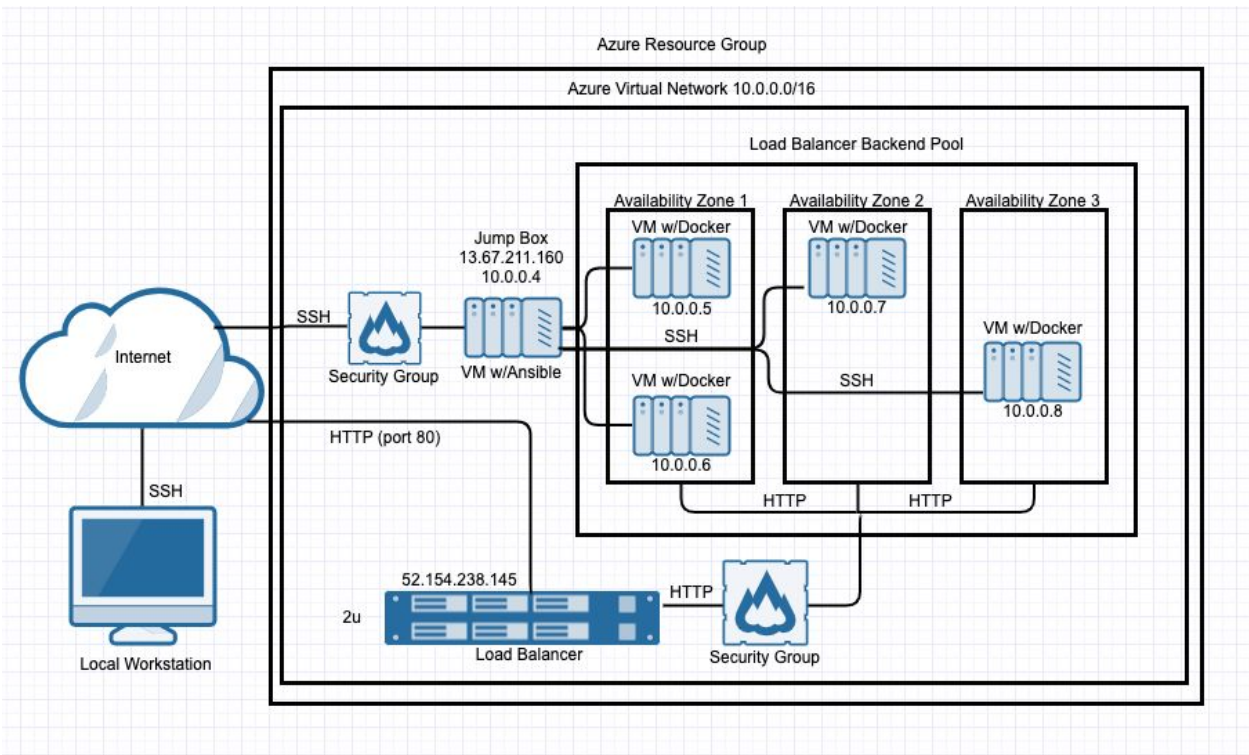


*Now it's time to begin deploying.*

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.

# 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



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:

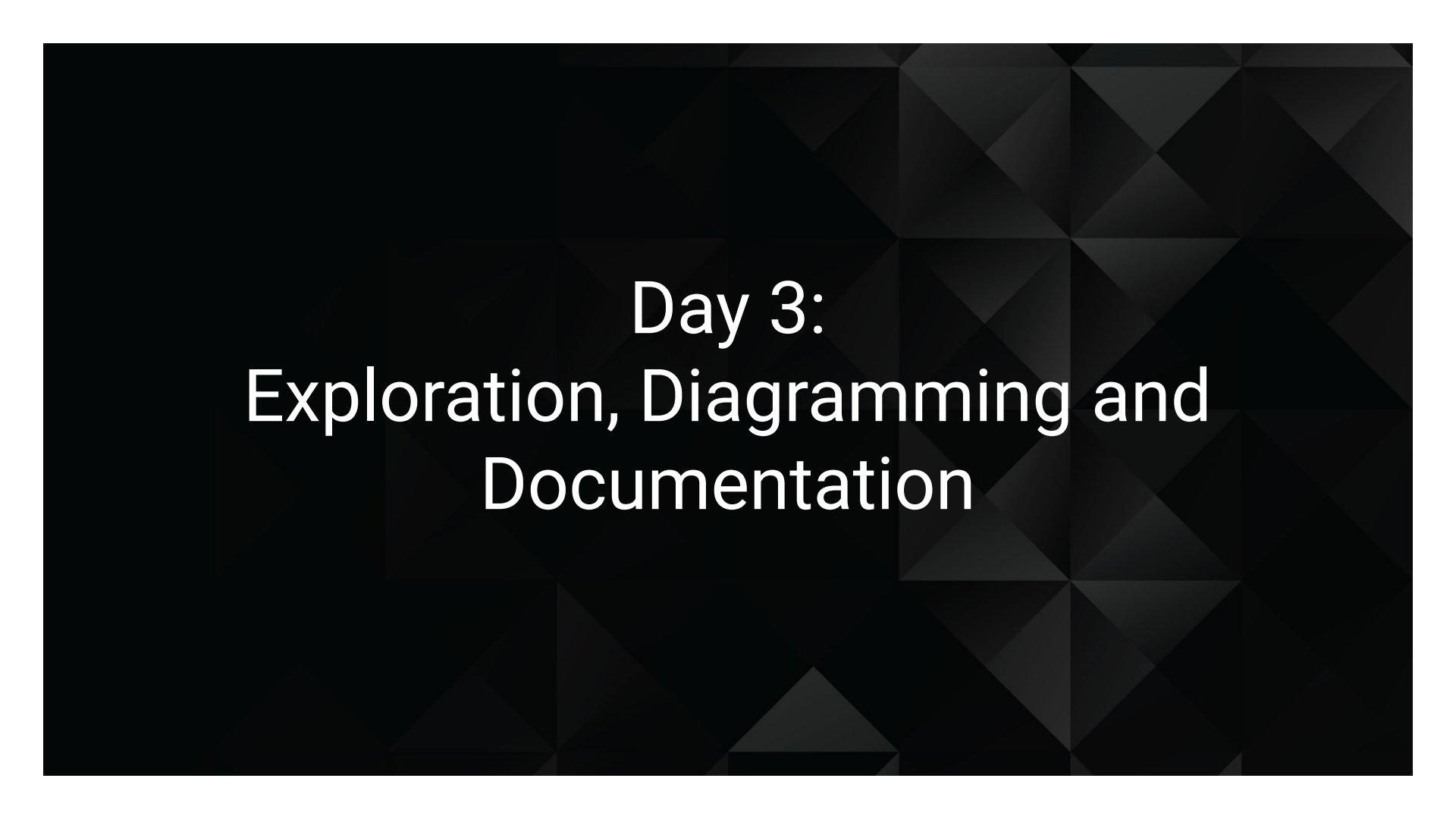


Installed and  
launched Docker  
containers to a  
host machine.

Configured and  
deployed an ELK  
server.

Installed  
Filebeat on a  
Linux server.

*(Completing the Metricbeat installation was a similar process.)*

The background is a dark, almost black, field filled with a complex, repeating pattern of triangles. These triangles are in various shades of dark gray and black, creating a subtle, textured effect. The triangles are arranged in a way that they interlock, forming a larger, more complex geometric structure.

# Day 3: Exploration, Diagramming and Documentation



In the final day of the project, you will use Kibana to navigate logs. Then, you will have the option to create a network diagram, complete a README to document the information of the project, answer interview-style questions about your project, or further explore Kibana's capabilities.

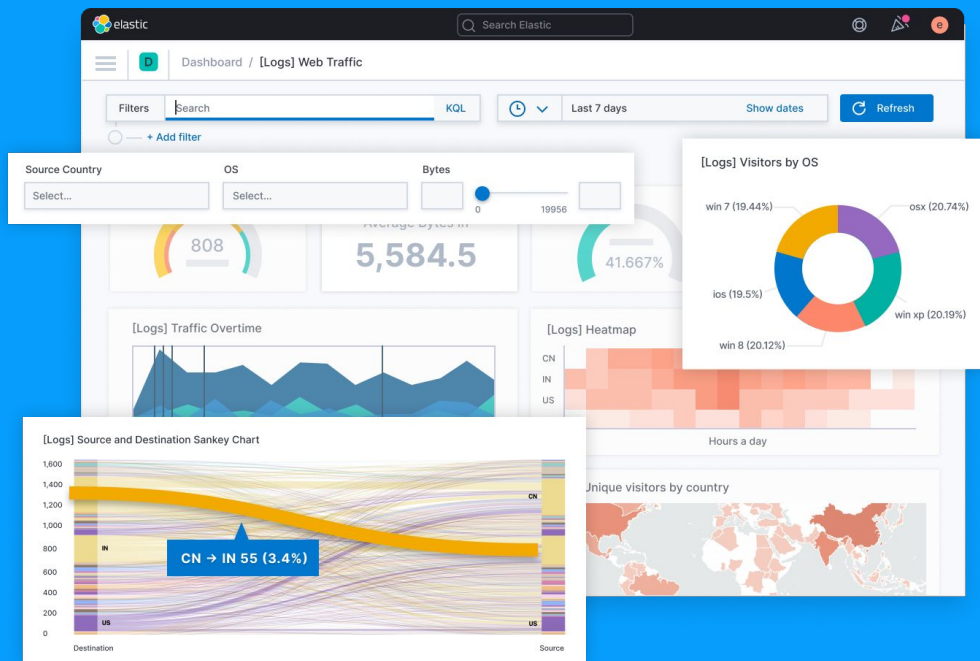
# Exploring Kibana



# Kibana

Companies use tools like Kibana to research events that have happened on their network

- Any attack leaves a trace that can be followed and investigated using logs.
- Kibana is an interface to view and gain insight from that type of data that would otherwise be unmanageable.





# Instructor Demonstration

## Kibana Walkthrough



## Exploring Kibana

In this activity, you will use Kibana to explore web servers logs and analyze packet data for potential issues


**Suggested Time:**

0:25



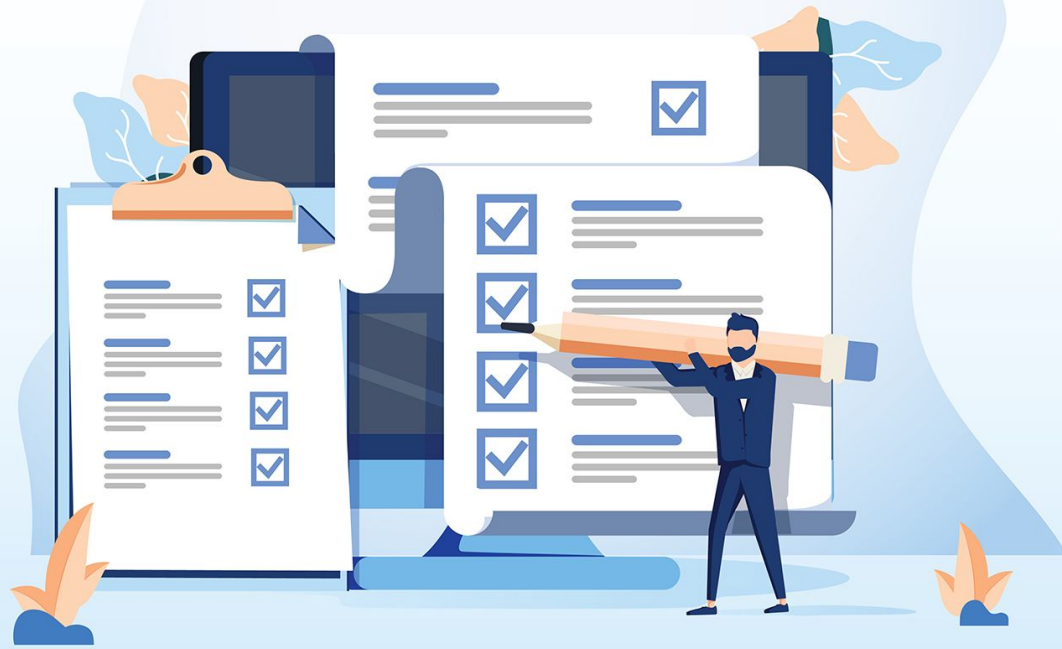


**Time's Up!** Let's Review.



# Project Communication: Documenting, Diagramming and Discussions

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.



# Mock Interview Questions

---

Along with providing the visual deliverables, you should also be able to verbally communicate the work you did on your project in order to address specific interview questions or to show your skills within a specific domain.

In this section, we will introduce an optional section of the project in which you will answer mock interview questions about your project and how they relate to various security domains.



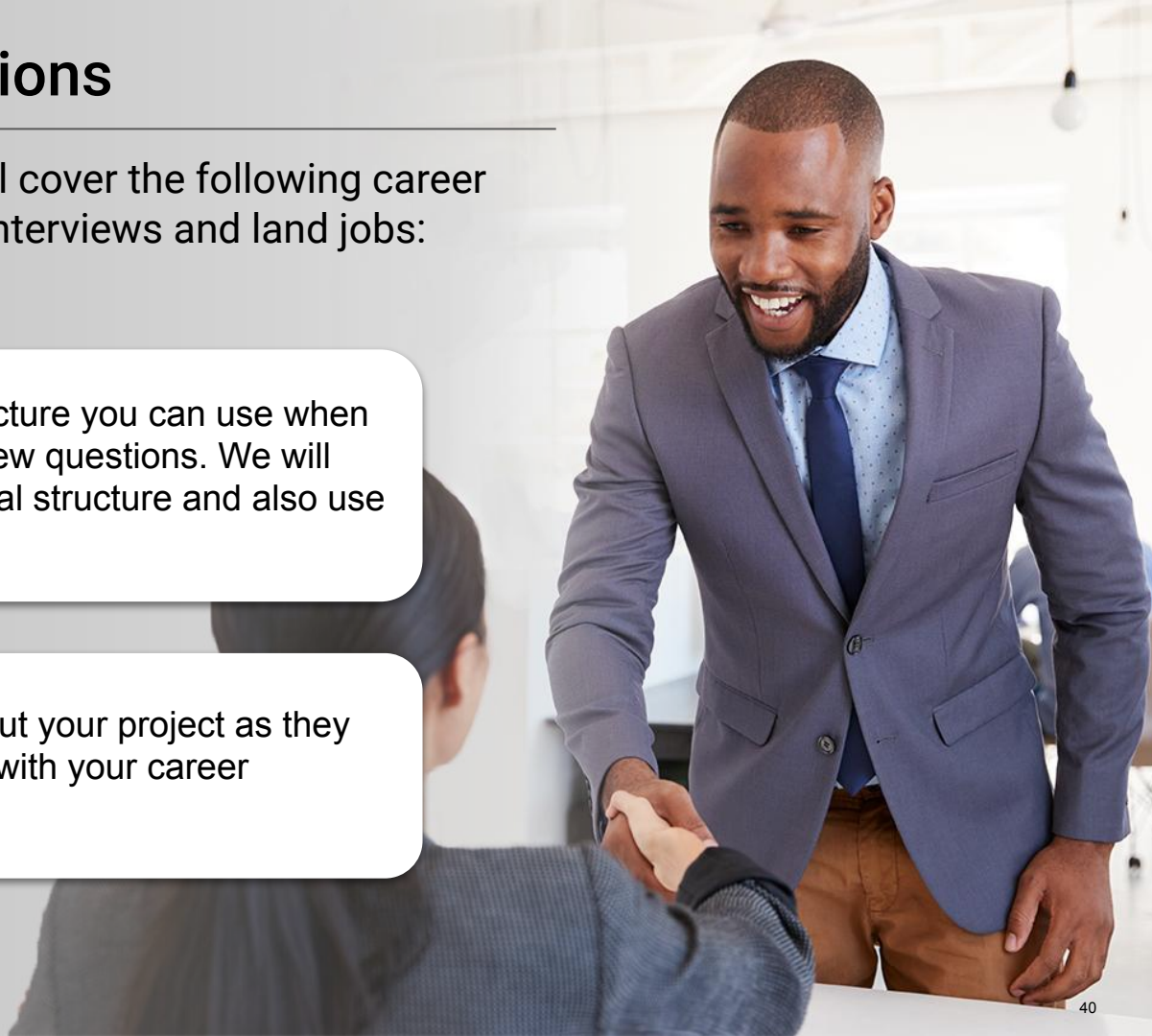
# Mock Interview Questions

---

Over the next two classes, we will cover the following career prep skills that will help you get interviews and land jobs:

First, we will look at the general structure you can use when answering common technical interview questions. We will look at examples that use this general structure and also use specific examples from Project 1.

Then, you will answer questions about your project as they relate to security domains that align with your career interests.





# Responding to Technical Questions

In this section, we will walk through the process of answering technical interview questions.

01

Restate the Problem

02

Provide a Concrete Example Scenario

03

Explain the Solution Requirements

04

Explain the Solution Details

05

Identify Advantages/Disadvantages of the Solution



# Sample Question

---

*“How would you control access to a cloud network?”*

01

## Restate the Problem

When restating the question in your own words, add additional details to demonstrate you understand what is being asked and why.

**Example:** *“It’s important that organizations control access to a cloud network, especially since it has resources that only the engineering team should be able to access. In the interest of the Principle of Least Privilege, you want to make sure engineers can access it easily, but no one else can.”*

# Sample Question

---

*“How would you control access to a cloud network?”*

02

## Provide a Concrete Example Scenario

Use the parameters of the question to create an example scenario of the problem you just restated. This makes the problem easier to talk through and further demonstrates your experience with the topic.

**Example:** *“In Project 1 of my Cybersecurity boot camp, we solved an almost identical problem. In that project, we deployed a virtual network containing several VMs to Azure, which only we and our instructional team were supposed to be able to access. Just as an organization would limit cloud network access to only engineers, we had to implement remote access controls limiting access to only a handful of authorized individuals.”*

# Sample Question

---

*“How would you control access to a cloud network?”*

03

## Explain the Solution Requirements

Before explaining the details of your solution, explain the high level actions taken at each step and explain what they accomplished.

**Example:** *“After deploying the network, I first had to configure a Network Security Group (NSG) around the whole subnet. This blocked traffic from all IP addresses, except for mine, my partners', and my instructors'. This NSG allowed inbound access to only one machine on the internal network, called the Jump Box. **Then**, I configured additional NSGs on the VMs within the subnet. This allowed connections only between the jump box and other local IP addresses. **Finally**, I forced the use of SSH keys to eliminate vulnerability to password-based brute-force.”*

# Sample Question

---

*“How would you control access to a cloud network?”*

04

## Identify Advantages and Disadvantages

Point out why your solution works in general. Then acknowledge any potential shortcomings and how you would address them..

**Example:** *“This solution worked well for my project because it ensured only the selected users have access. However, it is difficult to maintain and scale because it requires updating the NSG every time a new user requires access to the network. In addition, securely using SSH keys can be tricky in the long-run. An alternative solution that addresses these shortcomings would be implementing a VPN gateway to the private network. This would allow us to manage and monitor users more safely and scalably.”*

# Sample Question

---

*“How would you control access to a cloud network?”*

05

## Explain the Solution Details

Now that you laid out the high level steps and reflected on the pros and cons, use this time to explain the specifics of how you would implement the solution. The examples below are abbreviated for brevity, but real answers would typically include considerably more detail.

**Example:** *“To configure access controls around the entire subnet, I created an NSG with the following ruleset: [Omitted...] These rules allow access to the jump box from only the specified IP addresses specified. **Then**, to configure access controls within the subnet, I created NSGs with the following ruleset: [...] These rules allow the VMs within the network to communicate only with each other and with the jump box. To force the use of SSH keys, I modified the following configurations in the VMs on the network: [...] This ensures that password brute-force attacks will always fail.*

# Your Turn: Project Communication and Further Exploration

---

For the rest of the allotted project time, you will have the ability to focus on any of the following activities:

01

Finalize the network diagram you began during the cloud security week.

02

Draft a README explaining what you've built.

03

Answer mock interview questions.

04

Continue exploring Kibana's features.

---



## Day 3 Activity: Project Communication and Further Exploration

You will be provided four supplemental activity files.

- Additional Kibana Exploration
- Network Diagram (Part of homework)
- Project Readme (Part of Homework)
- Mock Interview Questions

**Suggested Time:**  
Remaining 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](https://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.