1. Question 1: Faulty Firewall.

Ensuring SSH traffic is limited to the people who need to have access is critical to the security of not just information of the servers, but the continued operation of them itself. A way I limited this was only allowing SSH from the web into a specific jump box limited by the firewall to my personal IP address and from there, then gave the ability to SSH into the 4 different servers via that jump box server. That way you cannot gain access to the system unless your logging in directly to that jump box. This is important because it limits avenues of attack and ensures only individuals on my IP that are trying to log into the jump box specifically have access to the system. Any other attempted logins would be denied with a "timed out" error message. This is because those other servers do not have direct ssh access to port 22 from the internet and can only be accessed via the jump box. If there was an issue where one or all of the other machines was accepting ssh connections it would most likely be due to a flat configuration of the firewalls rules. It would be best to then double-check the rules that allow ssh connections into the network and the deny all rule, making sure that only the exceptions I made in the firewall would be let through. The best way to test this would be through trying to both ssh into the jumpbox via a different IP address as well as attempting to ssh-ing into any of the other VM's. The first step would be to look into the Network Security Group and see which rules regulate ssh traffic into the network. Once in the NSG rules I would then look into the configuration of the rules that allow SSH connections and check the source and destination IP's as well as the ports that allow ssh connections to the network.
The best way to then test the rules have taken effect is to simply try to connect to the various VM's, this be done by attempting SSH's both from my personal IP and a different IP. the only valid connection should be the SSH from my personal machine into the jumpbox. This of course does not leave the system immune to attack and unauthorized access. It simply limits the avenues of breaking into the system that still indeed can be brute-forced through. A way we can monitor

this however is through the configuration of services running to Kibana this will be able to be monitored, hence giving us the ability to respond before the attack is successful.

2. Question 2: Cloud Access Control

Cloud Deployment has several benefits as well as several negative outcomes for its use. It can help protect privacy, shift the physical infrastructure needs and be a safe way of separating local and virtual data. In my project, I used the cloud entirely for the deployment of my server. The way I connected the system was all through the initial Jumpbox. Once ssh'd into that you could then access the various different components in the virtual structure. In order to keep this secure, I build my firewall very specifically so that connections to the network is blocked unless from connection to all IP addresses except my personal IP I used to build the VN. This hence kept the other servers safe from most attacks.

The way I configured this was through a combination of load balancer rules and network security groups. Through the use of NSGs I was able to apply the same security rules to multiple VMs, therefore, saving time without losing security for building individual rule sets for each VM. I did this by allowing the NSG for jumpbox, Web 1, and Web 2, to allow SSH only from the IP of my personal network to the jumpbox's IP with TCP protocols. Following that I allowed the load balancer to be able to allow my personal IP to connect to its IP address with TCP protocols once again. While then denying all other traffic to the network. The other NSG I \ created was for my Elk VM. This game me access to port 5601 so that I was able to use Kibana to monitor my system. I didn't want to unlock this port for all of my VMs however, being this would leave more ports open than absolutely needed leaving a security concern. I still needed to be able to connect all the VMs together tho so they can SSH over the virtual network

The solution I created however scaleable can be done an easier way for a large corporate structure through a VPN being you would only have to connect to the network to gain access to these different machines. VPNs are far easier to use for scaling however are more complex to

initially set up and for the size of my project I didn't see the need. The benefits to using them however they are typically much more secure even if they are broken into. VPNs are best used to protect data from multiple locations where you can't always control everything in a network such as BIOB policies.

3. Question 3: Challenges of Collecting Large Amounts of Log Data

For my project, I worked primarily with log data with I inspected after having everything up and going to investigate the traffic on my server. I wanted to investigate any suspicious traffic on my server. In this case, it was primarily RPM downloads being they were the most suspicious. The overall data I collected for the 7-day investigation was just over 69.4 kb.

I was primarily filtering through data spikes, to search for anything out of the ordinary. Once I limited those encounters I then filtered through on Kibana to find the source of the anomalies, in this instance, it was RPM files. So therefore that's what I investigated. Whats nice about my use of the service Kibana is that I wasn't stuck reading large amounts of data but instead was able to easily filter through and find what I was looking for. Once I found the activity in question via Metricbeat, I was then able to use the system logs collected by filebeat to review the incident. The charts I used where heat lamps, maps, bph, and the filter options as well as the syslogs ability once I defined my search.