**Dynnis Feliciano**

**Activity File: Interview Questions**

**Domain: Network Security**

To debug a firewall that is configured to block ssh connections, for project 1, I configured Jump Box Provisioner, Web-1 (10.1.0.5), Web-2 (10.1.0.6), Web-3 (10.1.0.7), and Web-4 (ELK) (10.2.0.5) to allow my firewall to connect via ssh. Using RedTeam-SG (Network Security Group) and under settings, I configured the Inbound & Outbound Security rules with priority 500 and 600 to allow any ssh request to these VMs. I can ssh to all VMs that I configured including Jump Box provisioner.

For example, when using the command ssh [sysadmin@10.1.0.5](mailto:sysadmin@10.1.0.5) (Web-1) and it fails, I would try to ping it first to see if the network is reachable. The initial problem I ran into was the Inbound/Outbound Security Rules that defaulted when I created my VMs. It was the DenyAllInbound / DenyAllOutbound rule configured to block any port, any protocol, and source and destination. I then check to make sure that I the rules I had configured is allowing traffic to port 22.

If one of my VMs accepts ssh connections, I would check my network security group named RedTeam-SG. I would check the Inbound and Outbound rules and make changes to port 22. To test the effectiveness of my configuration I will for example use the following command: ssh [sysadmin@10.1.0.5](mailto:sysadmin@10.1.0.5) to verify that this VM is reachable. I will perform the same command using other VMs private IP addresses.

The specific configurations and controls check I would perform is to check the Azure UI pane under support and troubleshooting and under Setting. By selecting the Reset Password under support > troubleshooting, this will ensure that the ssh-genkey is still up to date. This checks the credentials of an existing user and reset the ssh configuration; and ensure that I am using the right name of the VM and that I have the correct IP address.

For example, the name of my 3 VMs are called sysadmin (ssh [sysadmin@10.1.0.5](mailto:sysadmin@10.1.0.5)), and the Web-4 is named elk (ssh [elk@10.2.0.5](mailto:elk@10.2.0.5)). In order to test VM connections, I will use the command: ssh [sysadmin@10.1.0.5](mailto:sysadmin@10.1.0.5) (Web-1); ssh [sysadmin@10.1.0.6](mailto:sysadmin@10.1.0.6) (Web-2); ssh [sysadmin@10.1.0.7](mailto:sysadmin@10.1.0.7) (Web-3); and ssh [elk@10.2.0.7](mailto:elk@10.2.0.7) (Web-4). To follow the monitoring controls to mitigate risk, we need to identify critical data files within the VM that may need a higher degree of monitoring as well as log management

Furthermore, dormant VMs may have an outdated access control policy or may not be included from essential security monitoring functions. This may cause security loopholes in the virtualized world. Finally, packet-filtering which inspects the source and destination IP and port address in each TCP/IP packets; and IDS/IPS analyzes network traffic for known malicious threats.

The following monitoring controls will be considered for Project 1 to identify and protect any suspicious authentication attempts. For example, Snort is a software product that looks at network traffic in real time which logs packets in great details to facilitate security and authentication efforts. For best practice, all servers that’s for general web usage should not be firewalled off port 80, despite compliance guidelines requiring encryption.

We should offer both HTTP on port 80 and HTTPS on port 443. Keeping port 80 available allows users to get to the right HTTPS version (which often time a user receive a redirect to HTTPS). Allowing port 80 doesn’t reduce vulnerability on your server since request to this port are generally served by software which runs on port 443. The Web-4 (ELK) server was running HTTP on port 80. This is permissible because even for various services, the HTTP connection is received on port 80 by a system-wide drive which acts as a protocol analysis to figure out where to hand the request off to the correct service.

I think that server resources are allocated to handle connections, for this instance, using port 80. This port is a way to connect a request to a specific connection. A potential problem of running HTTP on port 80 is when an application server decides complete control of port 80. As a result, having multiple HTTP requests will not be possible especially if an Operating System (OS) has a HTTP dispatch mechanism which can handle multiple processes on port 80. Opening a port in firewall for specific zone in your gateway is critical when opening a port so that the designated traffic for it on a specified for it is accepted.

In Project 1, I successfully deployed Kibana to detect irregularities and its sources. Other deployed products include Filebeat, Metricbeat, Logstash which collects data, evaluates, and monitors the networks. Too many open ports can be attractive to attackers, bots, and bad actors as it is prone to an attack so it’s important to monitor and safeguard your network with great care.