Faulty Firewall

A firewall was created to block SSH, however SSH is not being blocked. In project 1, we allowed public SSH ports to be visible for our jumpbox virtual machine. We then constructed our internal virtual machines to not have a public IP address so that SSH would not be able to connect from a public IP. Since someone is able to connect directly to our internal web servers, it means that the target machines would have to have a public IP address assigned which is not what was intended. If the project was completed correctly and someone attempted to SSH into an internal VM, they would have to use the private IP which has no meaning on the outside of the virtual network.

Since the jump box virtual machine does accept SSH connections from the public, the error could be assumed to be the firewall not restricting specific IP addresses from accessing the public SSH port. The admin would need to double check the firewall settings for SSH to make sure that only specific IP addresses are able to connect. Once the correction is made, the admin can use a VPN to change their IP and attempt to SSH into the jumpbox system. If set up correctly, this attempt should fail and there would be no response from the server.

To investigate this issue within the Azure UI, the admin should navigate to the network security group page associated with the jumpbox virtual machine. They would then view the firewall rule that allows SSH and view the configuration to see if the rule is restricting specific IP addresses, or allowing all connections. After opening the SSH firewall rule, the source field should say IP Addresses and the Source IP addresses range should include a list of IP addresses allowed to connect. Once corrected, the admin would connect to the jumpbox VM via SSH using a machine that has an IP address listed in the firewall whitelist.

The solution does not guarantee against unauthorized access, because if the whitelisted machine is compromised, someone could connect via SSH to the jumpbox VM using the whitelisted IP. In order to identify suspicious authentication attempts, the admin could set up logging or auditing for the jumpbox machine that records all SSH login attempts. If there is a login attempt that the admin did not do themselves, then this is an indicator of attack.

Unsecured Web Server

Compliance requires web servers to run on an encrypted port, however there is a webserver running on port 80 which is not secure. In project 1, we had multiple web servers running on port 80: web1, web2 and one running on port 5601 which was also not secure. For the sake of the project, the webservers running DVWA are using unencrypted HTTP. Also, the Kibana port 5601 was unencrypted however for the sake of the project, data viewed here did not need to be protected. However in a production environment it would be important to encrypt Kibana data as that would reveal a lot of information if compromised. If this was a real deployment, I would modify the Kibana webserver to implement some sort of encrypted communication.

Running on HTTP port 80 is an issue because data sent over the wire is not encrypted and therefore able to be captured by anyone. In order to reconfigure the kibana server to serve HTTPS, you would have to configure an SSL certificate by following a series of steps. 1) run elasticsearch-certutil to create a private key and certificate. 2) add appropriate configuration to kibana.yml to instruct kibana to use sll encryption. Adding SSL to the kibana webserver will solve the issue of having unencrypted traffic traveling over the wire from your browser to the kibana webserver.

The solution of using HTTPS and SSL will not break clients ability to connect with the webserver on port 80. The clients will send requests from a random port and the server will respond with a response that came from an encrypted HTTPS port. This solution is long-term friendly, however major advancements in SSL encryption or future vulnerabilities with SSL might require websites to implement a newer version of SSL or a better alternative if one exists in the future.