When it comes to a cloud network, access control is vital. Implementing security is key to ensuring that data and information stay protected from unwanted parties. There are many ways that one can implement these measures in this virtual environment, including SSH keys and firewalls.

In this project, we deployed a virtual cloud network using Microsoft Azure. This cloud-based system is ideal for deploying and testing new cloud-based networks and monitoring their activity from a central interface. In creating this network, security measures were implemented in many ways. Upon sign up for Azure, multifactor authentication was implemented to ensure only I could be the one to access the Portal. Anyone with the ability to access the Portal has complete control over all other forms of security when it comes to the network. Multifactor authentication and a strong password are your first line of defense. Next, we generated SSH keys for our machines that give what is almost like a fingerprint to that specific machine that we then copied into the virtual machines within the Portal. Doing this allows only the specified machine(s) that generated the key to access the virtual network through the open (but restricted) port 22, providing another barrier of protection. Finally, the network security group(s) dictate which incoming and outgoing connections are allowed to the devices connected to that specific group. With this layer of protection, we were able to control the types of connections established between each virtual machine and specify which machines can communicate with one another. Security measures like these exponentially increase the protection of sensitive data that can all be implemented and managed from Azure.

This network contains security groups put in place to filter traffic in and out of the network. Within these groups, you can configure the protocols and IPs allowed to communicate. NSGs take place on the outer layer of the network. We then implemented a list of allowed and denied protocols. Doing so allowed the SSH connections through an opened port 22 and HTTP connections through port 80 going in and out of the network while blocking all other incoming and outgoing activity. With this, we could connect and manage our machines through SSH protocols while also having a web interface for monitoring.

For each of the NSGs we set up through Azure, protocols that specifically allowed connections from one machine to another while also allowing our webservers to communicate out to the web. The first two rulesets were the SSH rules into our Jump-Box VM and from the Jump-Box container into the webserver VMs, as well as the newly created Elk-server VM. This NSG rule blocked all other connections into these machines except for the specified machine that held the private key that corresponded with the set public key. From there, another rule allowing for HTTP connections to the load balancer was created, allowing for incoming IP connections to be scanned and filtered. With Jump-Box, you are using a virtual machine that is accessed through the web. More specifically, you are using the opened port 22 to use the SSH protocol, allowing access to the machine(s). This port can be very insecure to leave open unless protected by NSGs, blocking all unspecified traffic. Like we gained access to the Jump-Box, we similarly gain access to our webservers. However, the difference here is that we are first launching and attaching a container that holds the proper credentials to get into the webservers but can be detached from the network. After the container is booted, the public ID is listed in the two web servers, granting control to the Jump-Box container.

In this example, our solution can scale as large or small as one would like. To achieve a larger scale of this instance, a more extensive network of virtual machines that are all within the same resource groups would have to be curated, sharing rules that allow communication in the NSGs. While using Jump-Box is ideal in this situation for the ease of access, there are other options out there. Not all these options offer the security the Jump-Box can if configures appropriately. The disadvantages of a VPN are that they cost money to deploy and run and requires more support. We did not use them this time around because they would have been added work and money for little benefit. However, if one were traveling away from their specified public network address or had someone else trying to access the network, using a VPN could be helpful to change that remote person’s IP to the public IP allowed in the network security groups. When scaling the network and having approved remote workers access the network, it would be acceptable to implement a VPN.