

Day 1 Solution Guide: ELK Installation

1. Creating a New vNet

Make sure that you are logged into your personal Azure account, where your cloud security unit VMs are located.

- Create a new vNet located in the same resource group you have been using.
 - Make sure this vNet is located in a *new* region and not the same region as your other VM's.

Create virtual network

Basics IP Addresses Security Tags Review + create

Azure Virtual Network (VNet) is the fundamental building block for your private network in Azure. VNet enables many types of Azure resources, such as Azure Virtual Machines (VM), to securely communicate with each other, the internet, and on-premises networks. VNet is similar to a traditional network that you'd operate in your own data center, but brings with it additional benefits of Azure's infrastructure such as scale, availability, and isolation. [Learn more about virtual network](#)

Project details

Subscription * ⓘ Azure subscription 1 ▼

Resource group * ⓘ RedTeam ▼
[Create new](#)

Instance details

Name * ELK-NET ✓

Region * (US) West US ▼

Here we are adding it to the (US) West US region because all the other resources are in the (US) East US region.

- Note that *which* region you select is not important as long as it's a different US region than your other resources.

Create virtual network

Basics **IP Addresses** Security Tags Review + create

The virtual network's address space, specified as one or more address prefixes in CIDR notation (e.g. 192.168.1.0/24).

IPv4 address space

10.2.0.0/16 10.2.0.0 - 10.2.255.255 (65536 addresses)



☐ Add IPv6 address space ⓘ

The subnet's address range in CIDR notation (e.g. 192.168.1.0/24). It must be contained by the address space of the virtual network.

[+ Add subnet](#) [🗑 Remove subnet](#)

☐ Subnet name

Subnet address range

☐ default

10.2.0.0/24

- Leave the rest of the settings at default.
 - Notice, in this example, that the IP Addressing has automatically created a new network space of `10.2.0.0/16`. If your network is different (10.1.0.0 or 10.3.0.0) it is ok as long as you accept the default settings. Azure automatically creates a network that will work.

Create virtual network

✓ Validation passed

Basics

IP Addresses

Security

Tags

Review + create

Basics

Subscription	Azure subscription 1
Resource group	RedTeam
Name	ELK-NET
Region	West US

IP addresses

Address space	10.2.0.0/16
Subnet	default (10.2.0.0/24)

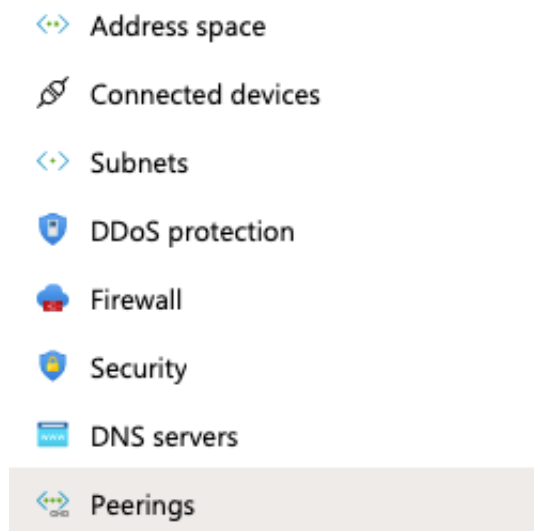
Tags

None

Security

BastionHost	Disabled
DDoS protection plan	Basic
Firewall	Disabled

- Create a Peer connection between your vNets. This will allow traffic to pass between your vNets and regions. This peer connection will make both a connection from your first vNet to your Second vNet *And* a reverse connection from your second vNet back to your first vNet. This will allow traffic to pass in both directions.
- Navigate to 'Virtual Network' in the Azure Portal.
- Select your new vNet to view it's details.
- Under 'Settings' on the left side, select 'Peerings'.
- Click the + Add button to create a new Peering.



- Make sure your new Peering has the following settings:
 - A unique name of the connection from your new vNet to your old vNet.
 - Elk-to-Red would make sense
 - Choose your original RedTeam vNet in the dropdown labeled 'Virtual Network'. This is the network you are connecting to your new vNet and you should only have one option.
 - Name the resulting connection from your RedTeam Vnet to your Elk vNet.
 - Red-to-Elk would make sense
- Leave all other settings at their defaults.

i For peering to work, two peering links must be created. By selecting remote virtual network, Azure will create both peering links.

This virtual network

Peering link name *

ELK-to-Red



Traffic to remote virtual network ⓘ

☒ Allow (default)

☐ Block all traffic to the remote virtual network

Traffic forwarded from remote virtual network ⓘ

☒ Allow (default)

☐ Block traffic that originates from outside this virtual network

Virtual network gateway or Route Server ⓘ

☐ Use this virtual network's gateway or Route Server

☐ Use the remote virtual network's gateway or Route Server

☒ None (default)

Remote virtual network

Peering link name *

Red-to-ELK



Virtual network deployment model ⓘ

☒ Resource manager

☐ Classic

☐ I know my resource ID ⓘ

Subscription * ⓘ

Azure subscription 1



Virtual network *

elk-net



Traffic to remote virtual network ⓘ

☒ Allow (default)

☐ Block all traffic to the remote virtual network

Traffic forwarded from remote virtual network ⓘ

☒ Allow (default)

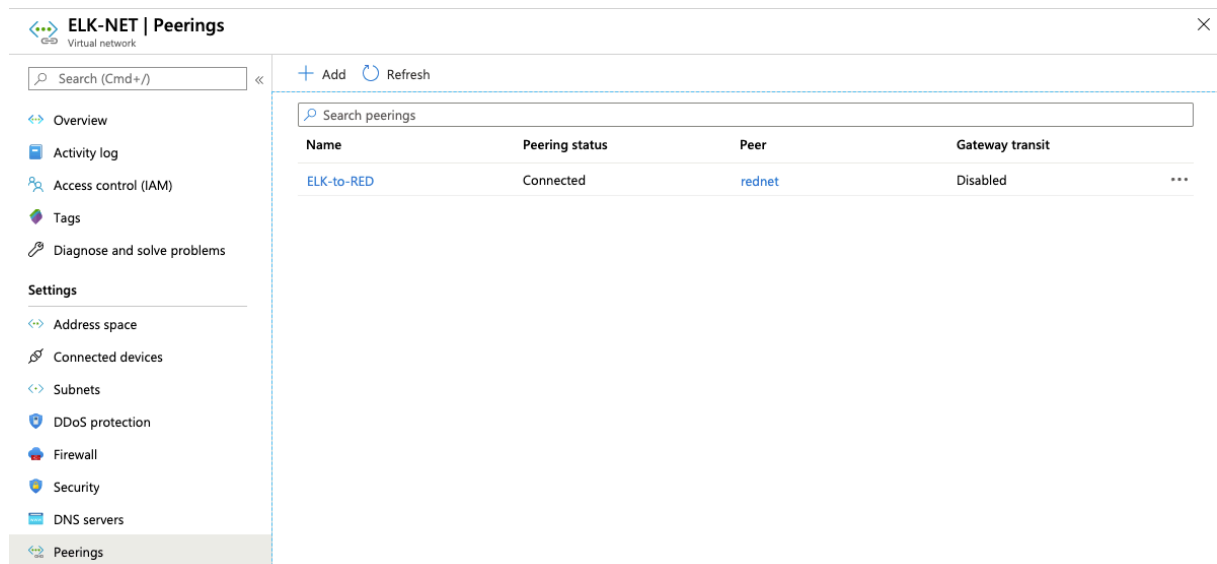
☐ Block traffic that originates from outside this virtual network

Virtual network gateway or Route Server ⓘ

☐ Use this virtual network's gateway or Route Server

☐ Use the remote virtual network's gateway or Route Server

☒ None (default)



2. Creating a New VM

Set up a new virtual machine to run ELK.

- SSH into your Jump-Box using `ssh username@jump.box.ip`
- Check for your Ansible container:

```
sysadmin@Jump-Box-Provisioner:~$ sudo docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED
STATUS	PORTS	NAMES	

- Locate the container name:

```
sysadmin@Jump-Box-Provisioner:~$ sudo docker container list -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
4d16db8c80d6	cyberxsecurity/ubuntu:bionic	"bash"	3 days ago	Exited (0) 3 days ago		

- Start the container:

```
sysadmin@Jump-Box-Provisioner:~$ sudo docker container start romantic_noyce
romantic_noyce
sysadmin@Jump-Box-Provisioner:~$
```

- Connect to the Ansible container:

```
sysadmin@Jump-Box-Provisioner:~$ sudo docker container attach romanti
c_noyce
root@6160a9be360e:~#
```

- Copy the SSH key from the Ansible container on your jump box:

```
# cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDUfoIGFxTFyZXWV0QuCCmPKxsvGhnW/
sKwGrOZ/K7nozKxsaRSCSG/oLgbugTyi9+fRY9wYWCmK/HLpjOaTEi8iU+ydvGM8nTloD
/dI1je9PClUCxFQjql2XyQz32FqDjHV8rCZA+Pz+9ozc7BogQwLLg/0c4beQYbVQPKs1Q
GHf31YuXs6hAraJMXCx7VsDJHQwfv1kScE2s+yGeUJMt0ny3xaED8y2Pn+mBF2Tw7HLT+
HPkmvXcuCkLxo6gY3ad+EH9Ko0r2AEFvtZTcFyGfIDLcS6jo+GUlKuCLGRAzeKNhq+D78
fHf8Vt4qvUSIywP9HHnvnqfUCVKXsKxZGGl root@6160a9be360e
```

- Configure a new VM using that SSH key.
 - Make sure this VM has at least 4 GB of RAM.
 - Make sure it has a public IP address.
 - Make sure it is added to your new vNet and create a new Security Group for it.
- Solutions:

Create a virtual machine

Subscription * ⓘ

Azure subscription 1

Resource group * ⓘ

RedTeam

[Create new](#)

Instance details

Virtual machine name * ⓘ

ELK-SERVER

Region * ⓘ

(US) West US

Availability options ⓘ

No infrastructure redundancy required

Image * ⓘ

Ubuntu Server 18.04 LTS

[Browse all public and private images](#)

Size * ⓘ

Standard D2s v3

2 vcpus, 8 GiB memory (\$85.41/month)

[Change size](#)

Administrator account

Authentication type ⓘ

☒ SSH public key ☐ Password

Username * ⓘ

sysadmin

SSH public key * ⓘ

ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQDUfolGFxTFyZXWV0QuCCmPKxsv
GhnW/sKwGrOZ/K7nozKxsaRSCSG/oLgbugTyi9+fRY9wYWCmK/HLpjOaTEi8i

[Learn more about creating and using SSH keys in Azure](#)

Inbound port rules

Select which virtual machine network ports are accessible from the public internet. You can specify more limited or granular network access on the Networking tab.

Public inbound ports * ⓘ

☐ None ☒ Allow selected ports

Select inbound ports *

SSH (22)



This will allow all IP addresses to access your virtual machine. This is only recommended for testing. Use the Advanced controls in the Networking tab to create rules to limit inbound traffic to known IP addresses.


Basics Disks **Networking** Management Advanced Tags Review + create

Define network connectivity for your virtual machine by configuring network interface card (NIC) settings. You can control ports, inbound and outbound connectivity with security group rules, or place behind an existing load balancing solution. [Learn more](#)

Network interface

When creating a virtual machine, a network interface will be created for you.

Virtual network *	<div>ELK-NET</div> <div>Create new</div>
Subnet *	<div>default (10.2.0.0/24)</div> <div>Manage subnet configuration</div>
Public IP	<div>(new) ELK-SERVER-ip</div> <div>Create new</div>
NIC network security group	<div><input type="radio"/> None <input checked="" type="radio"/> Basic <input type="radio"/> Advanced</div>
Public inbound ports *	<div><input type="radio"/> None <input checked="" type="radio"/> Allow selected ports</div>
Select inbound ports *	<div>SSH (22)</div>

 This will allow all IP addresses to access your virtual machine. This is only recommended for testing. Use the Advanced controls in the Networking tab to create rules to limit inbound traffic to known IP addresses.

Accelerated networking ☐ On ☒ Off

The selected VM size does not support accelerated networking.

Load balancing

You can place this virtual machine in the backend pool of an existing Azure load balancing solution. [Learn more](#)

Place this virtual machine behind an existing load balancing solution? ☐ Yes ☒ No

3. Downloading and Configuring the Container

In this step, you had to: - Add your new VM to the Ansible `hosts` file. - Create a new Ansible playbook to use for your new ELK virtual machine. - The header of the Ansible playbook can specify a different group of machines as well as a different remote user (in case you did not use the same admin name):

```

```bash
- name: Config elk VM with Docker
 hosts: elk
 remote_user: azadmin
 become: true
 tasks:
 ...

- Before you can run the elk container, we need to increase the memory:

```yaml
- name: Use more memory
  sysctl:
    name: vm.max_map_count
    value: '262144'
    state: present
    reload: yes
  ...

- This is a system requirement for the ELK container. More info [at the `
elk-docker` documentation](https://elk-docker.readthedocs.io/#prerequisites).
- The playbook should then install the following services:
  - `docker.io`
  - `python3-pip`
  - `docker`, which is the Docker Python pip module.

```

4. Launching and Exposing the Container

After Docker is installed, download and run the `sebp/elk:761` container. - The container should be started with these published ports: - `5601:5601` - `9200:9200` - `5044:5044`

Your Ansible output should resemble the output below and not contain any errors:

```

root@6160a9be360e:/etc/ansible# ansible-playbook elk.yml

PLAY [Configure Elk VM with Docker] *****

TASK [Gathering Facts] *****
ok: [10.1.0.4]

TASK [Install docker.io] *****
changed: [10.1.0.4]

TASK [Install python3-pip] *****
changed: [10.1.0.4]

TASK [Install Docker module] *****
changed: [10.1.0.4]

TASK [Increase virtual memory] *****
changed: [10.1.0.4]

TASK [Increase virtual memory on restart] *****
changed: [10.1.0.4]

TASK [download and launch a docker elk container] *****
changed: [10.1.0.4]

TASK [Enable service docker on boot] *****
**
changed: [10.1.0.4]

PLAY RECAP *****
10.1.0.4          : ok=1    changed=7    unreachable=0    failed
=0      skipped=0    rescued=0    ignored=0

```

- SSH from your Ansible container to your ELK machine to verify the connection before you run your playbook.

- After the ELK container is installed, SSH to your container and double check that your `elk-docker` container is running.

Run `sudo docker ps`

```
sysadmin@elk:~$ sudo docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED
STATUS            PORTS
842caa422ed8       sebp/elk           "/usr/local/bin/star... 3 hours
ago               Up 3 hours         0.0.0.0:5044->5044/tcp, 0.0.0.0:5601->560
1/tcp, 0.0.0.0:9200->9200/tcp, 9300/tcp   elk
sysadmin@elk:~$
```

Solutions: - [Ansible Configuration File](#) - [Ansible Hosts File](#) - [ELK Playbook](#)

5. Identity and Access Management

This ELK web server runs on port `5601` . Create an incoming rule for your security group that allows TCP traffic over port `5601` from your IP address.

Verify that you can load the ELK stack server from your browser at

`http://[your.VM.IP]:5601/app/kibana` .

Solutions: Sending traffic to the entire ELK-NET is fine here because there are no other resources besides the ELK server.

 Save  Discard  Delete

Source ⓘ

IP Addresses ▼

Source IP addresses/CIDR ranges * ⓘ

207.5.43.36

Source port ranges * ⓘ

*

Destination ⓘ

Any ▼

Service ⓘ

Custom ▼

Destination port ranges * ⓘ

5601

Protocol

☒ Any

☐ TCP

☐ UDP

☐ ICMP

Action

☒ Allow

☐ Deny

Priority * ⓘ

100

Name

ALL

Description

Access to port 5601 ✓

You can also choose to send traffic *only* to the ELK server by changing "Virtual Network" to the IP of your ELK Server.

If everything is working correctly, you should see this webpage:

