IERG4841 Course Report

Name: Doria Tang

SID: 1155126139

Lab 1

In Lab 1, I configured Cisco switch and ASA so that windows PC can connect to the internet. The windows PC IP is dynamically assigned by DHCP server in ASA and the range is from 192.168.205.10 to 192.168.205.41. ESXi is configured with a given static IP.

|  |  |  |
| --- | --- | --- |
| Hostname | IP | Gateway |
| Windows PC  (DESKTOP-S50KLBN) | 192.168.205.10 | 192.168.205.1 |
| ESXi server | 192.168.203.102 | 192.168.203.254 |

Switch:

| Interface | Vlan assigned | Switchport mode |
| --- | --- | --- |
| G0/1 | 15 | access |
| G0/2 | 11 | access |
| G0/7 | / | trunk |
| G0/8 | / | trunk |

ASA:

| Interface | Vlan assigned | Switchport mode |
| --- | --- | --- |
| E0/0 | 15 | access |
| E0/1 | 11 | access |
| E0/7 | 10 | access |

IP address of vlans:

|  |  |  |
| --- | --- | --- |
| Vlan | description | IP address |
| 10 | outside | 10.2.46.101/24 |
| 11 | DMZ | 192.168.203.254 |
| 15 | inside | 192.168.205.1 |

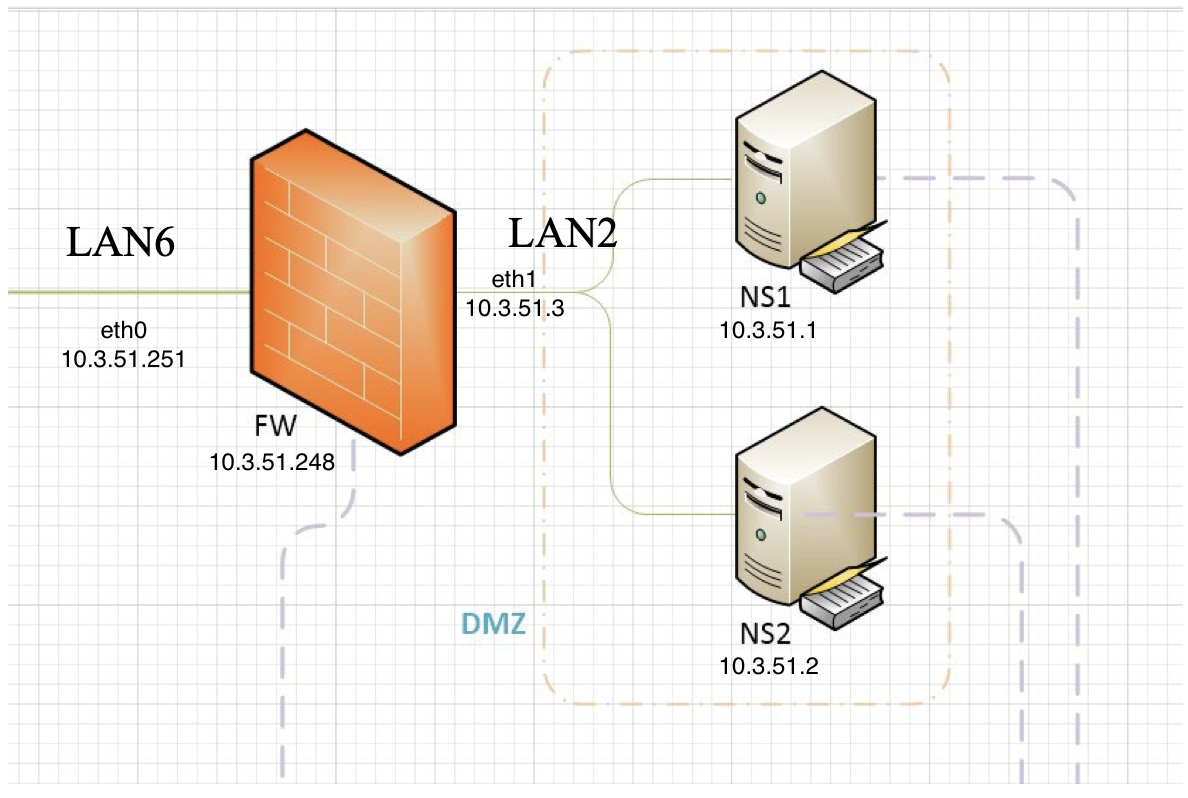
In order to allow the inside host to access the Internet, I configured the static route to outside and applied NAT configuration with dynamic interface. In order to test connectivity, I allowed ICMP traffic inspection in ASA, so that ICMP packets can pass through the inside host to outside. By assigning IP and gateway in ESXi, ESXi can be accessed through windows PC.

Lab 2

As no traffic should be routed to DMZ, this can be done by setting vlan 11 security level to highest so that low security level vlan traffic will not be able to access high security level vlan.

Then I set up the vpn tunnel by configuring the vpn pool, IPSec encryption, ikev1 and group policies. When a user connects to vpn, it will reassign a new IP address from 192.168.199.1 to 192.168.199.254. Firstly, I configured the group-policy by setting the DNS server and domain. Secondly, I added the tunnel-group to which I set MSCHAPv2 authentication as an encryption method. Thirdly, I configured VPN phase 1 and phase 2. In phase 2, 3DES encryption and SHA hashing is used for the VPN tunnel. The tunnel is set to transport mode. Then I map the configuration to both the inside and outside interface, so that both the inside host and outside host can connect to the VPN. In phase 1, pre-shared key is used for authentication, SHA is used for hashing and Diffel Hellman group 2 is used for secure key exchange. The configuration is applied both inside and outside. I also set up the split tunnel to allow traffic to flow through the encrypted vpn. 192.168.205.1 is set as the server address for the inside host to connect to the vpn. Finally, I set up the username and password so that the user can access the vpn.

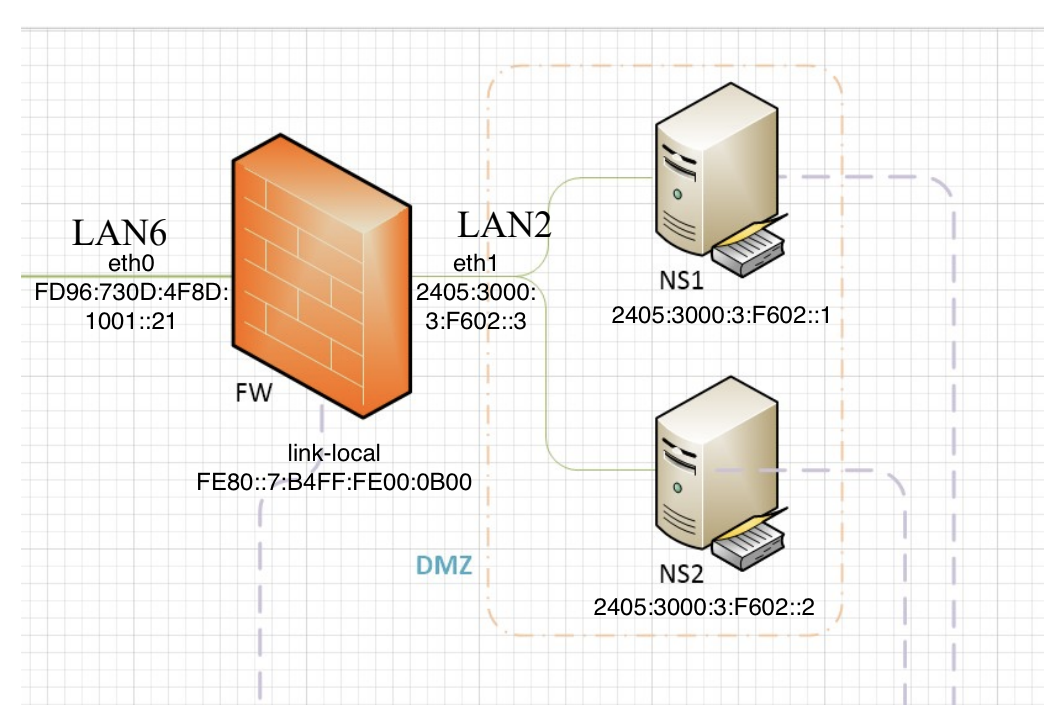
In order to access the remote hypervisor, G0/8 is configured as trunk mode. For the switch, I allowed vlan access on vlan 11 to 16. For the server management network, I assigned port groups and virtual switches to the local hypervisor. For the remote hypervisor, I assigned LAN to the switch with virtual NIC to provide Internet access. Then I deploy the management server to the remote hypervisor. After deploying the server, I update the yum package manager. Since CentOS has reached EOL, I look up the mirror site to configure the repo manually. After updating the package, I set up the ip subnetting according to the figure below.



I modified the files in /etc/sysconfig/network-scripts/ for the interface with their corresponding IP and gateway. I also enabled port forwarding by setting up ARP proxy in the firewall. By configuring the DNS server in the /etc/named folders in ns1 and ns2, ns1 will be set as primary DNS and ns2 will be set as secondary DNS. I set up the web server in ns1 so that when a user visits 10.3.51.1, it will display the web page in the browser. In order to filter the traffic, I set the default policy in iptables to drop and add the corresponding filtering rules, such as NAT rules and forward rules across interfaces, so that other unnecessary traffic will be blocked.

Lab 3

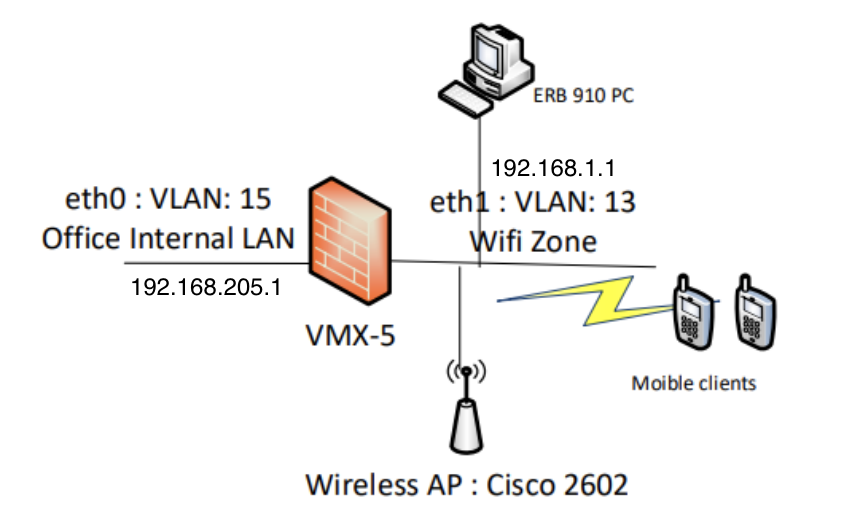
As I am given one set of IPv6 addresses, I assigned the IPv6 addresses according to the picture shown below.



I enabled IPv6 forwarding by setting net.ipv6.conf.all.forwarding = 1. Since IPv6 does not come with NAT, I set up an NDP proxy and manually added the neighbours in the firewall, so that the firewall can discover other nodes and maintain reachability information on vmx1 and vmx2. I also configured the DNS server by setting up a new zone area in /etc/named.conf with the AAAA records, and configured the zone file and reverse lookup file, so that users can lookup the IPv6 address of domains. I also configured the ip6tables by dropping the input and forwarding packets, and enabling only ICMP, HTTP to ns1, and DNS queries.

Lab 4

Since wireless AP, windows PC and other mobile clients are in vlan 13, interface g0/3 is configured to access mode and allow vlan 13 access. Then I deployed the firewall to the local ESXi and setup Internet facing eth0 and internal host facing eth1. Eth0 will get IP address from ASA DHCP server, while configuring DHCP service in vmx5, internal clients will get their ip address. I set up IP forwarding and NAT service so that the internal hosts can be routed to the internet.



After allowing internal hosts to access the internet, I access the wireless AP through the web console. I set up a broadcasting SSID named wave1 and enabled the 2.4GHz network interface, so that mobile clients can search for the SSID in the hidden networks. I enabled the cipher mode by setting it as AES-CCMP and TKIP for encryption. Afterwards, I set up the ASCII WPA Pre-Shared Key so that users can type the password and get access to the network.

In order to let users access the coovachilli webpage, manual password configuration is disabled. I use wget to download the coovachilli and haserl. I removed the -Werror option in src/Makefile.am and src/Makefile.in, then I installed the coovachilli with -enable-miniportal -with-openssl option. I also installed haserl and set up symbolic links between the files. In /etc/init.d/chilli, I enabled the port 3799 to allow portal login. I configured the /usr/local/etc/chilli/config and set the correct network of eth0 and eth1, and set radius username and password. After that, I edited the /etc/raddb/users and /etc/raddb/clients.conf and added the username and password, so that when user access to the network, they will first direct to the coovachilli portal and prompt to enter username and password.

Since I need to configure the RADIUS server, I stopped the coovachilli portal and reset it back to WPA-PSK encryption. First I installed the freeRadius server into vmx-5 and enabled EAP by editing /etc/raddb/sites-enabled/default and /etc/raddb/sites-enabled/inner-tunnel files. As RADIUS server requires server and root certificates, I first edited /etc/raddb/ca.cnf by changing the certificate\_authority details. For the Common Name, it must be a unique string. Then I generate the certificate by make ca.pem and make ca.der commands. To generate a server certificate, I edited the /etc/raddb/server.cnf file by making similar changes as the root certificate. However, the Common Name will be different from the one generated in the root certificate. Again, I used the make command to generate server.pem and server.csr.

Then, I edited the /etc/raddb/eap.conf by changing the attribute default\_eap\_type to peap. As I have configured the input\_password and output\_password attributes in the ca.cnf, I also changed the private\_key\_password the same as the two attributes. I edited the path of CA\_file, dh\_file and random\_file under certs/, so that eap.conf can locate the files under the certs/. I set the default\_eap\_type to mschapv2. Then, I edited the clients.conf users by adding username and password.

To enable the port 1812 and 1813 for RADIUS, I edited the iptables rules by adding the UDP ports of 1812 for authentication and 1813 for accounting. As packets will be encrypted, I also set two rules that allow ESP and AH traffic to flow through vmx-5.

To enable EAP authentication and accounting, I configured the cisco AP by setting the open authentication with EAP, and removed the PSK that was configured at the beginning. I also created the corporate servers and set up the share secret, which is the private\_key\_password in the eap.conf. To start the WPA EAP-PEAP connection, the AP also needs to enable WPA and use the configuration at the beginning. Users can now access the internet by inputting their username and password.

Questions to answer

Compare the pros and cons of the web portal encryption + authentication and the WPA + EAP-PEAP encryption + authentication. If you are the IT consultant of your company, which method would you recommend to your senior management??

Pros of web portal encryption and authentication:

* Coovachilli web portal provides a secure login page for users to enter their credentials. This can prevent unauthorised access to the network compared to WPA encryption.
* Coovachilli web portal offers more flexibility and enhances the user experience because the login page can be further customised.

Cons of web portal encryption and authentication:

* It requires the user to login with their credentials every time as it will not remember user credentials after the expiration time.

Pros of WPA + EAP-PEAP encryption + authentication:

* Allow automatic authentication if users have logged in to the network and authenticated before.
* The encryption is stronger than using the coovachilli web portal.

Cons of WPA + EAP-PEAP encryption + authentication:

* The encryption is weak because PEAP+MSCHAPV2 encryption can be cracked by an evil-twin attack, in which a hacker can set up a fake wifi-hotspot and get the user credentials.
* WPA encryption can be easily cracked by brute-force attack

As an IT consultant, I would recommend the CoovaChilli web portal. It is because most companies will value user experience first and thus can reinforce a company's brand. Moreover, network administrators can manage users easily as they can see from the web management portal. For security concerns, it can be fine-tuned in the web portal configuration, which is more convenient than manually configuring EAP-PEAP encryption protocols in the wireless controller.

Conclusion

Pros & Cons of the current infrastructure:

Pros:

* The Management zone is well protected because even users are in the closed network, it still requires VPN to access the server management portal. This can ensure that only authenticated users within the closed network can access to the server.
* For the DNS server, it has restricted access to traffic protocols as iptables are setup to allow specific network traffic to pass through. This can lower the chance of DNS server being attacked by hackers.
* The management zone is deployed in a private cloud server using vmware and ESXi. This can allow easy server deployment and management by the network administrators.
* Use of IPv6 can help to reduce IPv4 usage.

Cons:

* VPN encryption uses pre-shared key for authentication and authorisation, which is not safe because hackers can retrieve the key by man-in-the-middle attack.
* ICMP pings are still accepted by the firewall, which can potentially suffer from ICMP flood.
* The operating system in the deployed VM is too old. It uses CentOS 6, which is deprecated and no longer maintained. There may exist some potential vulnerabilities that cannot be resolved as some packages cannot be upgraded.

How to improve

Based on the cons:

* Consider using a more secure method for authentication and authorization, such as digital certificates or two-factor authentication to authenticate users. For example, EAP-TLS encryption can be deployed to increase the security of the data transmission. If for some reasons that the encryption protocols are fixed, then the pre-shared key should be changed from time to time, and it is long and complex enough to guess it.
* Configure the firewall to block ICMP pings if they are not required for network troubleshooting or monitoring.
* Upgrade the operating system to a more recent version that is still supported by the vendor. If the operating system cannot be upgraded, then extra security measures should be taken, such as removing unnecessary packages, or disabling services if needed.