Lab 8 - Defense in Depth

Start Assignment

Due Apr 24 by 11:59pm **Points** 100 **Submitting** a file upload

As we saw in the last lab, it is very important to restrict inbound traffic to very specific services on a Unix/linux machine. However, accidents happen, and a user or administrator can inadvertently start a misconfigured or vulnerable daemon. Exposing such a daemon to the public Internet invites a system compromise. It is typically considered best practice to utilize the concept of Defense in Depth, meaning deploying several security mechanisms to guard against malicious activity.

We will explore one mechanism for Defense in Depth by setting up Machine A to mirror the chains of the other machines in our production network. You shouldn't need to modify the existing chains on these machines, we're simply adding additional rules to the FORWARD chain on Machine A:

Machine A/Router

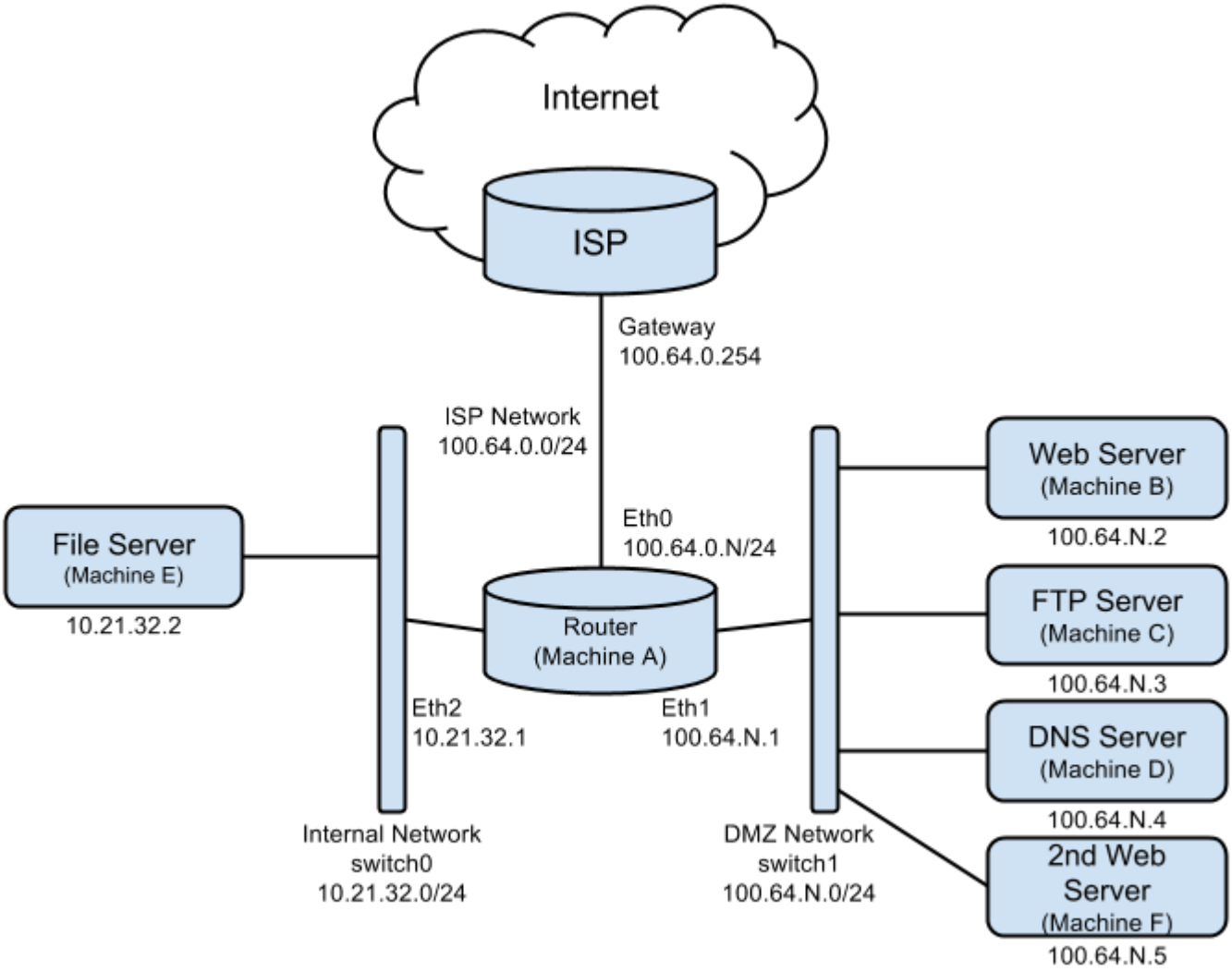
1. Allow the appropriate DHCP traffic to/from 100.64.N.0/24 & 10.21.32.0/24 (So your other machines can get their configs).
2. Deny your users access to Facebook from any machine on your network. You need not block all Facebook IP addresses, just the one you receive from a one-time resolve of facebook.com. To go above and beyond the requirements by blocking all Facebook IP addresses, get a list as follows:
   root@machineA# yum install jwhois
   root@machineA# whois -h whois.radb.net '!gAS32934'
3. Deny your users access to icanhas.cheezburger.com and cheezburger.com. Again, you need not block all such IP addresses, just the ones you receive from a one-time resolve.

4. Only forward packets to/from machines behind the router, based on the intended purpose of that specific machine. In other words, there should be rules on Machine A that mimic the rules for Machines B-F. This is an added layer of security in case the firewall on one of the other machines is inadvertently dropped.

Dunder Mifflin Network Topology & Configuration

Dunder Mifflin currently has the following CentOS Linux machines:

1. The router does dhcp, ip_forwarding, and network/port address translation (nat/pat).
2. The http server, carriage, hosts the corporate websites.
3. The ftp server, platen, updates prices and accepts batch orders.
4. The dns server, chase, maps between domain names and IP addresses.
5. The file server, roller, stores company files centrally.
6. The secondary http server, saddle, makes Michael Scott happy.



Submission Requirements

1. Please submit your lab notes in as either a word, PDF or text document. Minimally they should contain the commands and procedures to configure iptables on all D-M production machines. You may use the script program to record commands in a text file.
2. The active iptables rules for each machine must match the overview above. They must be configured to be applied at boot time.
3. All services on the machines should remain accessible from the specified locations when the lab is finished.

Hints & Troubleshooting

- iptables -vL shows the number of packets each rule matches. This can be used for troubleshooting.
- To log messages, use the iptables logging module. It can be enabled with the -m flag. For example:
  root@machineX# iptables -A INPUT -j LOG --log-prefix "IPTABLES-DROP: "
- For services you cannot test directly, try using the wget, ping, nmap, tcpdump and netcat (nc) programs to verify that your firewall rules behave as expected.