Team 17 Infrastructure Building

<https://youtu.be/hf4Eml6J2ms>

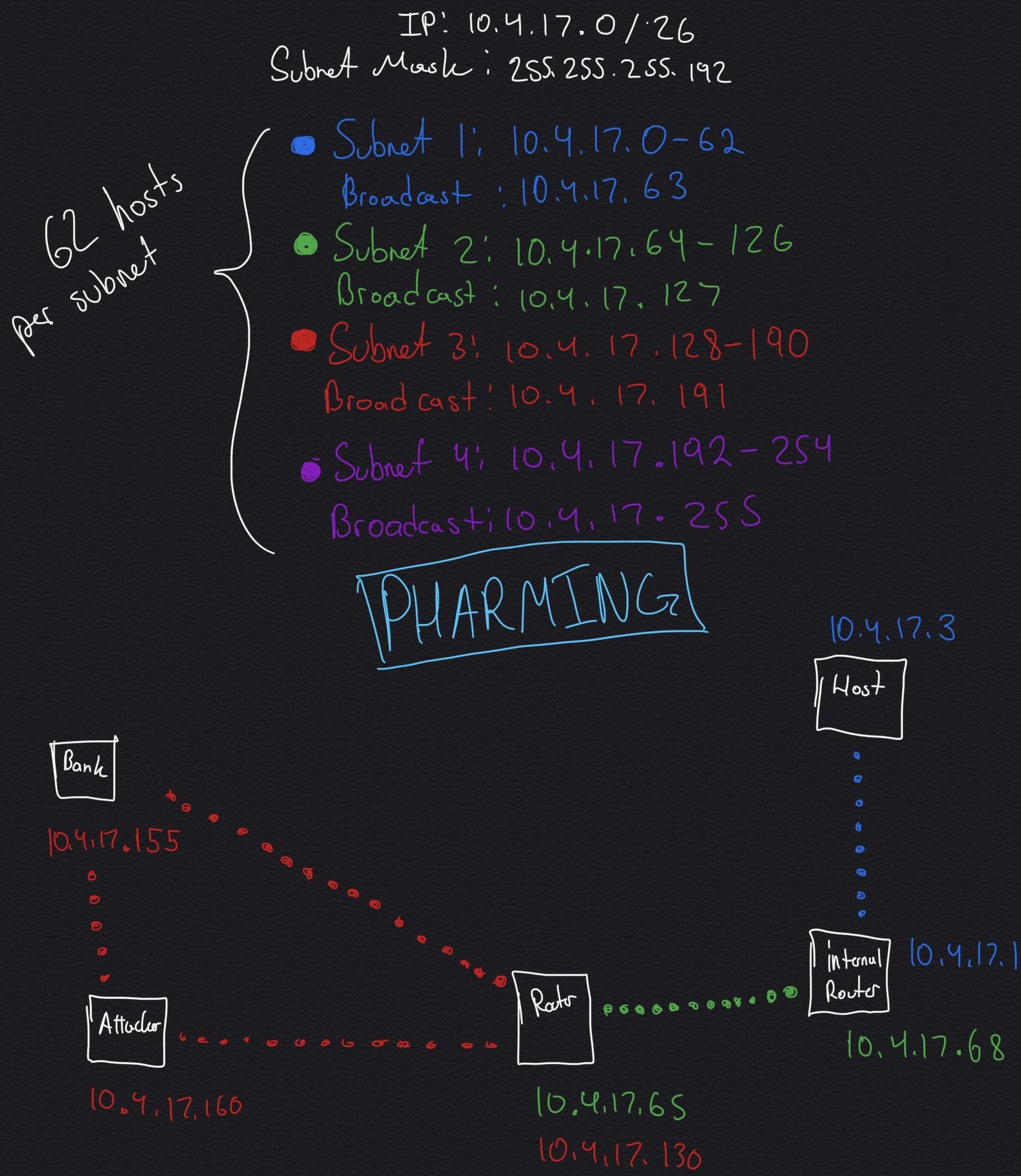
**Initial Setup**

Use Virtualbox and set up a Xubuntu (Ubuntu 16.04) machine. Clone it 5 times. (During our implementation we used hard cloning. We are unsure of the implications of linked I dont think it matters I did hard though clones.

* We will need to put all of the virtual machine on the same network
* In each instance of the machine, Go to Settings -> Network -> Adapter 2
  + Enable the network adapter
  + Select Internal Network
  + Name the network “intnet1” on all 5 machines

**Infrastructure Overview**

Below is a figure giving a brief overview of the network setup and what each of the machines does.



Our Network information:

Address: 10.4.17.0

Mask: 255.255.255.192

|  |  |  |  |
| --- | --- | --- | --- |
| **Client # + Desc** | **IP Address** | **IP Alias** | **Subnet Numbers** |
| 5 (Client) | 10.4.17.3 |  | 1 |
| 4 (Insecure Internal Router) | 10.4.17.1 | 10.4.17.68 | 1 and 2 |
| 3 (Main Router) | 10.4.17.65 | 10.4.17.130 | 2 and 3 |
| 2 Attacker (Mirror bank server) | 10.4.17.160 |  | 3 |
| 1 Bank Server | 10.4.17.155 |  | 3 |

**Machine Network Setup**

* In each of the machines, we will want to update apt-get
  + In terminal type Sudo apt-get update
* We will also need to download node dependencies for each machine
* We will need to configure each of the machines’ network settings.
  + Now in each machine terminal type “sudo nano /etc/network/interfaces”
  + Machine 1 (Banking Server)
    - auto enp0s8
    - iface enp0s8 inet static
    - address 10.4.17.155
    - netmask 255.255.255.192
    - network 10.4.17.0
    - broadcast 10.4.17.191
  + Machine 2 (Attack site and banking server)
    - auto enp0s8
    - iface enp0s8 inet static
    - address 10.4.17.160
    - netmask 255.255.255.192
    - network 10.4.17.0
    - broadcast 10.4.17.191
  + Machine 3 (Main Router for Client)
    - auto enp0s8
    - iface enp0s8 inet static
    - address 10.4.17.65
    - netmask 255.255.255.192
    - network 10.4.17.0
    - broadcast 10.4.17.127
    - auto enp0s8:0
    - iface enp0s8:0 inet static
    - address 10.4.17.130
    - netmask 255.255.255.192
    - network 10.4.17.0
    - broadcast 10.4.17.191
  + Machine 4 (Insecure internal Router for Client)
    - auto enp0s8
    - iface enp0s8 inet static
    - address 10.4.17.68
    - netmask 255.255.255.192
    - network 10.4.17.0
    - broadcast 10.4.17.127
    - auto enp0s8:0
    - iface enp0s8:0 inet static
    - address 10.4.17.1
    - netmask 255.255.255.192
    - network 10.4.17.0
    - broadcast 10.4.17.63
  + Machine 5 (client)
    - auto enp0s8
    - iface enp0s8 inet static
    - address 10.4.17.3
    - netmask 255.255.255.192
    - network 10.4.17.0
    - broadcast 10.4.17.63
* To save these changes we must reload the machine’s network configuration
  + In terminal, type
    - /etc/init.d/networking restart
* Machine 3 and 4s will act as routers linking between the different subnets on our network. We must enable packet forwarding on machine 4 and 3.
  + In terminal type
    - Sudo nano /etc/sysctl.conf
  + We must scroll down to find this line:
    - #net.ipv4.ip\_forward=1
  + Take out the pound sign so the line simply reads
    - net.ipv4.ip\_forward=1
  + **IMPORTANT: THE CHANGE WON’T TAKE EFFECT TILL REBOOT**
    - The workaround to rebooting is to type the following command in terminal
      * sudo sysctl net.ipv4.ip\_forward=1
* Finally we need to add ip routes so that our machines can communicate through the routers.
  + Below each machine’s description, execute the commands in terminal in the correcsponding machine’s terminal.
  + Machine 5:
    - Will need to communicate with the attacker and bank server’s websites on Subnet #3
      * sudo ip route add 10.4.17.64/26 via 10.4.17.1 dev enp0s8
    - Will need to communicate with the DNS server running on the main router on Subnet #2
      * sudo ip route add 10.4.17.128/26 via 10.4.17.1 dev enp0s8
  + Machine 4:
    - Will need to communicate with Machine 1, 2, and 3 to get DNS records from Machine 2 and 3 and websites from 1 and 2 on Subnet #3
      * sudo ip route add 10.4.17.128/26 via 10.4.17.65 dev enp0s8
  + Machine 3:
    - Will need to forward web pages from Machine 2 and 1 to Client 5 on Subnet #1
      * sudo ip route add 10.4.17.0/26 via 10.4.17.68 dev enp0s8
  + Machine 2:
    - Will need to send DNS queries to Client 5 and it’s attack and bank mirror webpages on Subnet #1
      * sudo ip route add 10.4.17.0/26 via 10.4.17.130 dev enp0s8
    - Will need to forward DNS records to Machine 4 that acts as a DNS forwarder on Subnet #2
      * sudo ip route add 10.4.17.64/26 via 10.4.17.130 dev enp0s8
  + Machine 1:
    - Will need to send bank webpage to Client 5 on Subnet #1
      * sudo ip route add 10.4.17.0/26 via 10.4.17.130 dev enp0s8
    - Will need to send DNS records to DNS forwarder to Machine 4 on Subnet #2
      * sudo ip route add 10.4.17.64/26 via 10.4.17.130 dev enp0s8

**DNS Setup**

Three of our machines will be running BIND9 DNS servers. Machine 2, the attacker, and Machine 3, the main router will act as nameservers responding to DNS queries for the website mysecurebank.com. Machine 4 will also be running a DNS server but it will strictly act as a forwarder. When Client 5 sends a DNS query with nslookup or dig, it will be forwarded to Machine 4. Machine 4 will be configured to forward the request to some other DNS Server (in our case it’s either going to be Machine 1 or if the attacker has their way, it will be Machine 2).

* Machine 4 DNS Setup
  + First of all, remove password prompts
    - In terminal type
      * sudo visudo
    - Then go to the bottom of the file and add
      * myusername ALL = (ALL) ALL
    - This will be important for the attack later on
  + Machine 4 (internal router) will act as a forwarding dns server.
  + We need to download BIND9. Execute this command in terminal
    - sudo apt-get install bind9
  + Modify the configuration files for the DNS server to set who it will accept queries for and which DNS server it will forward packets to. <https://www.digitalocean.com/community/tutorials/how-to-configure-bind-as-a-caching-or-forwarding-dns-server-on-ubuntu-14-04>
  + In terminal type
    - sudo nano/etc/bind/named.conf.options
      * acl goodclients{
      * 10.4.17.0/26;
      * localhost;
      * localnets;
      * };
      * options {
      * directory "/var/cache/bind";
      * recursion yes;
      * allow-query { goodclients; };
      * forwarders { 10.4.17.130; };
      * forward only;
      * dnssec-enable yes;
      * dnssec-validation yes;
    - We need to restart the server for this to take place
      * sudo service bind9 restart
  + In machine 5 we need to configure the nameserver to which it will send its DNS queries.
    - sudo nano /etc/resolv.conf
      * Nameserver 10.4.17.68
  + Now, all nslookup queries will be forwarded to machine 4 from machine 5
* Machine 3 DNS Setup:
  + sudo apt-get install bind9
  + sudo nano /etc/bind/named.conf.local
    - zone "mysecurebank.com" {
    - type master;
    - file "/etc/bind/zones/mysecurebank.com.db";
    - };
    - zone "3.2.1.in-addr.arpa" {
    - type master;
    - file "/etc/bind/zones/rev.3.2.1.in-addr.arpa";
    - };
  + cd /etc/bind/
  + sudo mkdir zones
  + cd zones
  + sudo nano mysecurebank.com.db
    - ; BIND data file for mysecurebank.com
    - ;
    - $TTL 14400
    - @ IN SOA ns1.mysecurebank.com. host.mysecurebank.com. (
    - 201006601 ; Serial
    - 7200 ; Refresh
    - 120 ; Retry
    - 2419200 ; Expire
    - 604800) ; Default TTL
    - ;
    - mysecurebank.com. IN NS ns1.mysecurebank.com.
    - mysecurebank.com. IN NS ns2.mysecurebank.com.
    - mysecurebank.com. IN MX 10 mail.mysecurebank.com.
    - mysecurebank.com. IN A 10.4.17.155
    - ns1 IN A 10.4.17.155
    - ns2 IN A 10.4.17.155
    - www IN CNAME mysecurebank.com.
    - mail IN A 10.4.17.155
    - ftp IN CNAME mysecurebank.com.
    - mysecurebank.com. IN TXT "v=spf1 ip4:10.4.17.155 a mx ~all"
    - mail IN TXT "v=spf1 a -all"
  + sudo nano /etc/bind/zones/rev.3.2.1.in-addr.arpa
    - @ IN SOA mysecurebank.com. host.mysecurebank.com. (
    - 2010081401;
    - 28800;
    - 604800;
    - 604800;
    - 86400 );
    - IN NS ns1.mysecurebank.com.
    - 4 IN PTR mysecurebank.com.
  + sudo nano /etc/resolv.conf
    - search mysecurebank.com
  + /etc/init.d/bind9 restart
  + service bind9 status
  + **TEST THIS FROM MACHINE WITH “NSLOOKUP MYSECUREBANK.COM 10.4.17.130**
* Machine 2 DNS Setup:
  + sudo apt-get install bind9
  + sudo nano /etc/bind/named.conf.local
    - zone "mysecurebank.com" {
    - type master;
    - file "/etc/bind/zones/mysecurebank.com.db";
    - };
    - zone "3.2.1.in-addr.arpa" {
    - type master;
    - file "/etc/bind/zones/rev.3.2.1.in-addr.arpa";
    - };
  + cd /etc/bind/
  + sudo mkdir zones
  + cd zones
  + sudo nano mysecurebank.com.db
    - ; BIND data file for mysecurebank.com
    - ;
    - $TTL 14400
    - @ IN SOA ns1.mysecurebank.com. host.mysecurebank.com. (
    - 201006601 ; Serial
    - 7200 ; Refresh
    - 120 ; Retry
    - 2419200 ; Expire
    - 604800) ; Default TTL
    - ;
    - mysecurebank.com. IN NS ns1.mysecurebank.com.
    - mysecurebank.com. IN NS ns2.mysecurebank.com.
    - mysecurebank.com. IN MX 10 mail.mysecurebank.com.
    - mysecurebank.com. IN A 10.4.17.160
    - ns1 IN A 10.4.17.160
    - ns2 IN A 10.4.17.160
    - www IN CNAME mysecurebank.com.
    - mail IN A 10.4.17.160
    - ftp IN CNAME mysecurebank.com.
    - mysecurebank.com. IN TXT "v=spf1 ip4:10.4.17.160 a mx ~all"
    - mail IN TXT "v=spf1 a -all"
  + sudo nano /etc/bind/zones/rev.3.2.1.in-addr.arpa
    - @ IN SOA mysecurebank.com. host.mysecurebank.com. (
    - 2010081401;
    - 28800;
    - 604800;
    - 604800;
    - 86400 );
    - IN NS ns1.mysecurebank.com.
    - 4 IN PTR mysecurebank.com.
  + sudo nano /etc/resolv.conf
    - search mysecurebank.com
  + /etc/init.d/bind9 restart
  + service bind9 status
* If the machine is patching the connections over the aliases, run these commands
  + Now disable ICMP on machine 4 so it doesn’t patch the connection between Machine 5+3 (for some reason pinging between machine still works despite this being disabled...)
    - In terminal type
      * sudo iptables -A INPUT -p icmp -j DROP
      * sudo iptables -A OUTPUT -p icmp -j DROP
* Machine 4 needs to run node
  + sudo apt-get install nodejs
  + Copy over mission1.zip for router app
  + To run the application it will need to run with elevated privileges because it will be modifying the DNS settings
* Machine 4 Nodejs Code (Internal Router)
  + Install Node.js
  + Grab and uncompress the “Machine 4.tar.gz”
  + Use Webrtc to detect a user’s internal IP address
    - <https://github.com/diafygi/webrtc-ips>
* Machine 2 (Attacker)
  + (You will need to open 2 terminals in order to run both the script and the malicious website)
  + Install Node.js
  + Grab and uncompress the “Machine 2.tar.gz”
  + Designing Attack sources
    - <https://jsfiddle.net/GSSCD/203/>
    - <https://github.com/diafygi/webrtc-ips>
  + sudo apt-get install nodejs
  + We will be running two sites
    - Run the evil site script to trip up the user on not port 80
    - Run the mysecurebank.com on port 80
* Machine 1 (Bank Server)
  + Install Node.js
  + Grab and uncompress the “Machine 1.tar.gz”
  + Download nodejs for legitimate banking website