Module 4: Network Attacks I

Nitesh Saxena

Adopted from previous lectures by Keith Ross

Overview of the Module

L1 Network Reconnaissance with Whois
L2 Network Reconnaissance with DNS
L3 Network Mapping
L4 Network Scanning Background
L5 Nmap Tool

Module 4, Lecture 1

Network Reconnaissance
With Whois

Reconnaissance

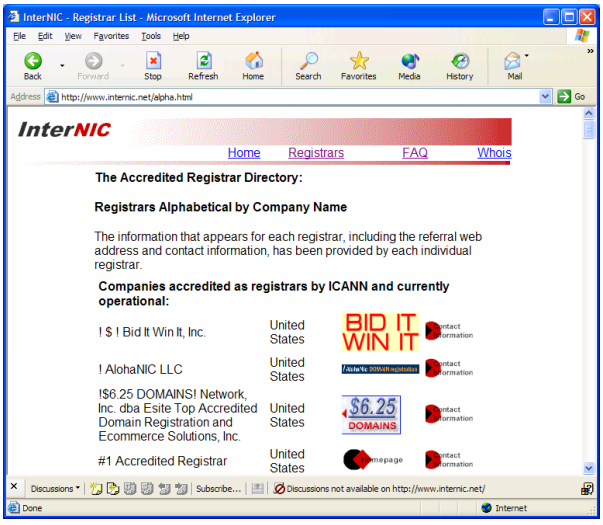
- "casing the joint"

Let's take a close look at:

- Reconnaissance with whois (this lecture)
- Reconnaissance with DNS (next lecture)

- A few words about a Registrar:
  - Organization where you register a domain name
  - Verifies uniqueness of name
  - Enters domain name into various databases: whois & DNS

List of registrars from internic.net:



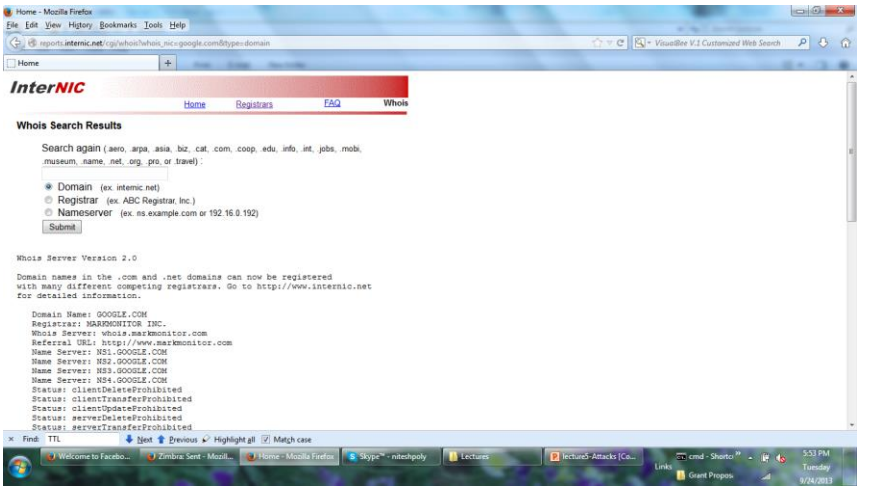
Whois databases

- Input: domain name or company name
- Output: registrar, whois server, dns server

Some useful whois sites:

- www.internic.net
  - For com, net and org top-level domains
- www.who.is
  - For country-code top-level domains, e.g., jp, fr

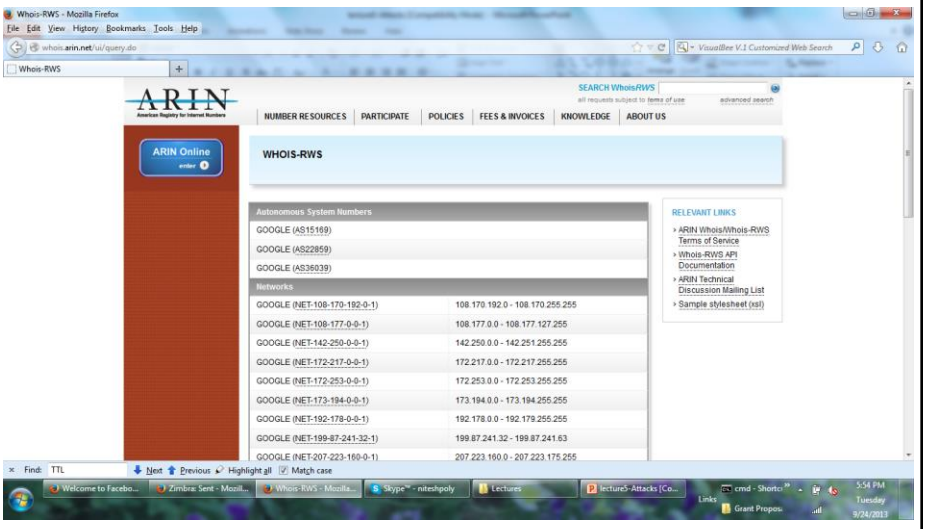
Internic Whois: Target "google"



Reconnaissance: IP Ranges

- ARIN: American Registry for Internet Numbers
  - Maintains whois database that includes IP address ranges in US
- RIPE: Europe
- APNIC: Asia

Query at ARIN



Why whois databases needs to be publicly available

- If you're under attack, can analyze source address of packets.
- Can use whois database to obtain info about the domain from where the attack is coming.
- Can inform admin that their systems are source of an attack

Module 4, Lecture 2

Network Reconnaissance
With DNS

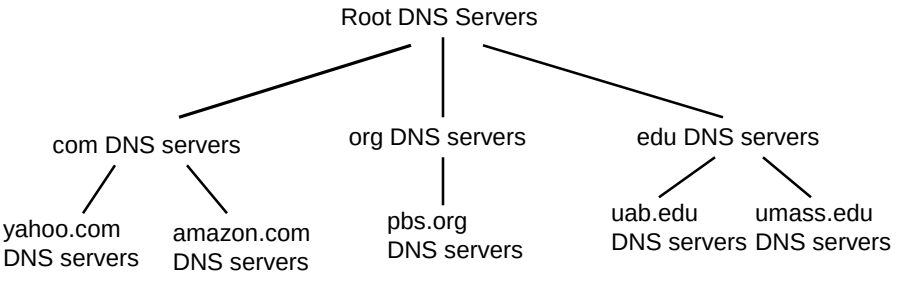
Reconnaissance: DNS database

Let's quickly review DNS:

- *distributed database* implemented in hierarchy of many *DNS servers*

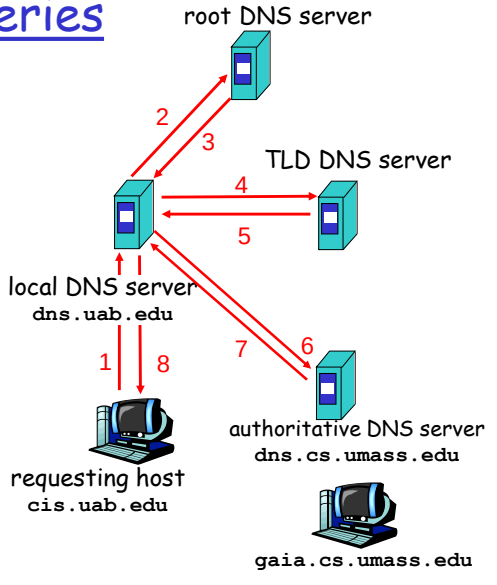
Authoritative name server:

- for a given domain (e.g., uab.edu), provides server name to IP address mappings for servers (Web, email, ftp, etc) in domain
- Primary and secondary name server for reliability



Portion of the hierarchy of DNS servers

DNS: queries



DNS records

DNS: distributed db storing resource records (RR)

RR format: `(name, value, type, ttl)`

- Type=A (address)
  - `name` is hostname
  - `value` is IP address
- Type=NS
  - `name` is domain (e.g. foo.com)
  - `value` is IP address of authoritative name server for this domain
- Type=MX
  - `value` is name of mailserver associated with `name`

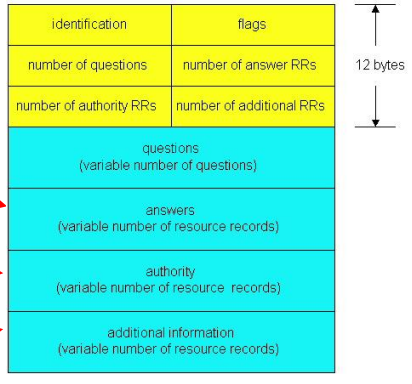
DNS protocol, messages

Name, type fields for a query

RRs in reponse to query

records for authoritative servers

additional "helpful" info that may be used



Query and reply messages sent Over UDP on port 53

DNS: caching and updating records

- once (any) DNS server learns mapping, it *caches* mapping
  - cache entries timeout (disappear) after some time
  - Improves efficiency of lookups of name/address mapping

Interrogating DNS servers

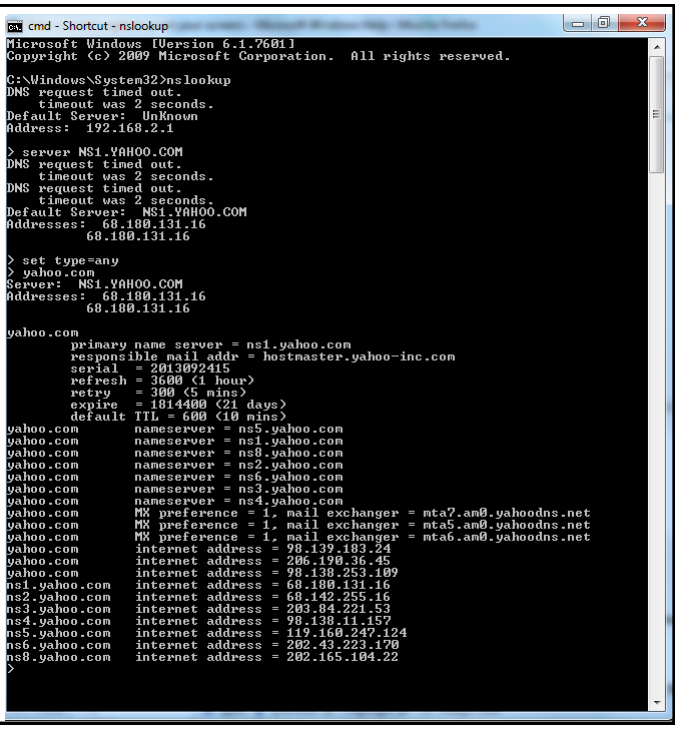
- Attacker first gets primary or secondary authoritative server for target organization using whois.
- Attacker can then query the DNS by sending DNS query messages.
- Tools (often available in Unix and Windows machines; also available at web sites):
  - nslookup
  - host
  - dig

nslookup

Avaiable in most unix & Windows machines

Get yahoo DNS server name using whois

set type=any
"get all"



Reconnaissance summary

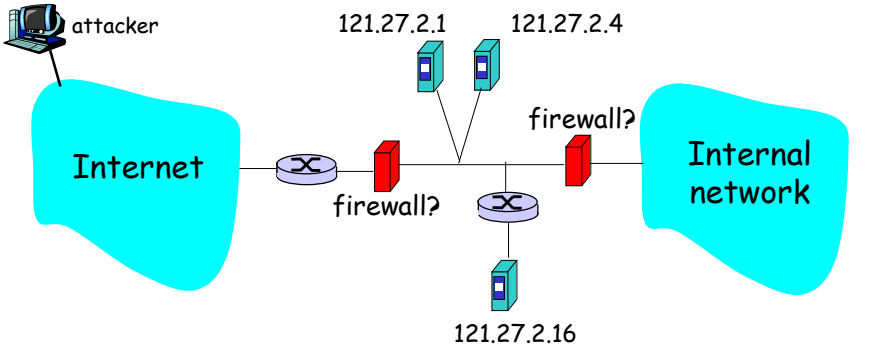
- Obtaining information from public databases:
  - whois databases
    - Tool: web sites
  - DNS database
    - Tool: nslookup
- Defense
  - Keep to a minimum what you put in the public database: only what is necessary

Module 4, Lecture 3

Network Mapping

Network mapping

- Goal: Learn about a remote network



Network mapping

- Attacker often uses traceroute to determine path to each host discovered during ping sweep.
  - Overlay results from traceroute to create an approximate network diagram

Traceroute

traceroute: gaia.cs.umass.edu to www.eurecom.fr

Three delay measements from gaia.cs.umass.edu to cs-gw.cs.umass.edu

```
1  cs-gw (128.119.240.254)  1 ms  1 ms  2 ms
2  border1-rt-fa5-1-0.gw.umass.edu (128.119.3.145)  1 ms  1 ms  2 ms
3  cht-vbns.gw.umass.edu (128.119.3.130)  6 ms 5 ms 5 ms
4  jn1-at1-0-0-19.wor.vbns.net (204.147.132.129)  16 ms 11 ms 13 ms
5  jn1-so7-0-0-0.wae.vbns.net (204.147.136.136)  21 ms 18 ms 18 ms
6  abilene-vbns.abilene.ucaid.edu (198.32.11.9)  22 ms  18 ms  22 ms
7  nycm-wash.abilene.ucaid.edu (198.32.8.46)  22 ms  22 ms  22 ms
8  62.40.103.253 (62.40.103.253)  104 ms 109 ms 106 ms
9  de2-1.de1.de.geant.net (62.40.96.129)  109 ms 102 ms 104 ms
10  de.fr1.fr.geant.net (62.40.96.50)  113 ms 121 ms 114 ms
11  renater-gw.fr1.fr.geant.net (62.40.103.54)  112 ms  114 ms  112 ms
12  nio-n2.cssi.renater.fr (193.51.206.13)  111 ms  114 ms  116 ms
13  nice.cssi.renater.fr (195.220.98.102)  123 ms  125 ms  124 ms
14  r3t2-nice.cssi.renater.fr (195.220.98.110)  126 ms  126 ms  124 ms
15  eurecom-valbonne.r3t2.ft.net (193.48.50.54)  135 ms  128 ms  133 ms
16  194.214.211.25 (194.214.211.25)  126 ms  128 ms  126 ms
17  * * *
18  * * *
19  fantasia.eurecom.fr (193.55.113.142)  132 ms  128 ms  136 ms
```

trans-oceanic link

* means no reponse (probe lost, router not replying)

Traceroute: How it works

- Source sends UDP packets to target
  - Each to an unlikely port
  - 3 packets with the same TTL, then increments TTL
- When router decrements TTL to 0, sends back to source ICMP packet
  - type 11, code 0, TTL expired
- When target receives packet, sends back to source ICMP packet
  - type 3, code 0, destination port unreachable

Module 4, Lecture 4

Network Scanning
Background

Ping Sweep

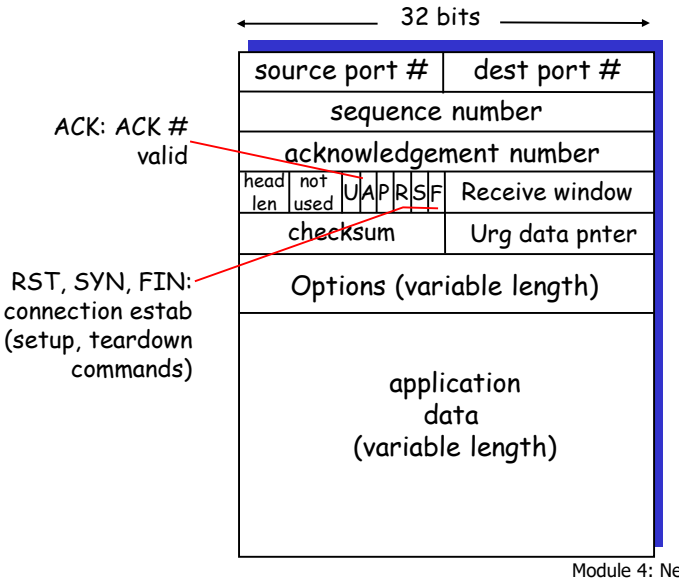
Ping

- Recall ICMP messages are directly encapsulated in IP datagrams (protocol 1)
- To ping a host:
  - send ICMP Echo Request (ICMP type 8)
  - Host responds with ICMP Echo Reply (type 0)
- So let's ping the entire IP address range
  - Use automated tool for this ping sweep
- If firewall blocks ping packets:
  - Try sweeping with TCP SYN packets to port 80
  - Or try sending UDP packets to possible ports

Port scanning

- Now that we have a map with some hosts, let's find out what ports are open on a target host
- 65,535 TCP ports; 65,535 UDP ports
  - Web server: TCP port 80
  - DNS server: UDP port 53
  - Mail server: TCP port 25
- Port scanning tools can scan:
  - List of ports
  - Range of ports
  - All possible TCP and UDP ports
- Attacker may scan a limited set of ports, to avoid detection

Interlude TCP segment structure



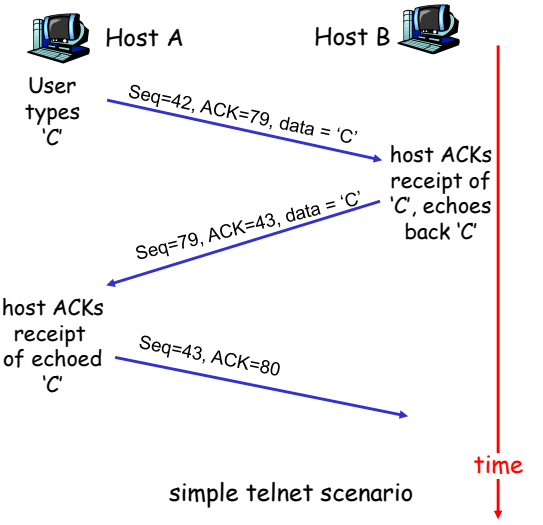
Interlude: TCP seq. #'s and ACKs

Seq. #'s:

- "number" of first data packet

ACKs:

- seq # of next packet expected from other side



simple telnet scenario

Interlude: TCP Connection Establishment

Three way handshake:

Step 1: client host sends TCP SYN segment to server

- SYN=1, ACK=0
- specifies initial seq #
- no data

Step 2: server host receives SYN, replies with SYN-ACK segment

- SYN=1, ACK=1
- server host allocates buffers; ack # is client seq # + 1
- specifies server initial seq. #

Step 3: client receives SYN-ACK, replies with ACK segment, which may contain data

- SYN=0, ACK=1
- ack # is server seq# + 1

TCP: Reset packet

- If machine receives a TCP packet it is not expecting, it responds with TCP packet with RST bit set.
  - For example when no process is listening on destination port
- For UDP, machine returns ICMP "port unreachable" instead

Module 4, Lecture 5

The Nmap Tool

Nmap

- Extremely popular
  - usually run over linux
  - rich feature set, exploiting raw sockets
  - need root to use all features
- Ping sweeping
  - over any range of IP addresses
  - with ICMP, SYN, ACK
  - OS determination
- Port scanning
  - Over any range of ports
  - Almost any type of TCP, UDP packet
- Source IP address spoofing
  - Decoy scanning

Excellent reference: Nmap man page

Nmap

Input:

- `nmap [Scan Type] [Options] <target hosts>`
- Default for port scanning: ports 1-1024 plus ports listed in nmap service file

Output:

- open ports: syn/ack returned; port is open
- unfiltered (closed) ports: RST returned: port is closed but not blocked by firewall
- filtered ports: nothing returned; port is blocked by firewall

Nmap: ping sweep

```
Nmap -sP -v 116.27.38/24
```

- Sends ICMP echo request (ping) to 256 addresses
- Can change options so that pings with SYNs, ACKs…
- -sP = ping
- -v = verbose

Nmap: polite port scan

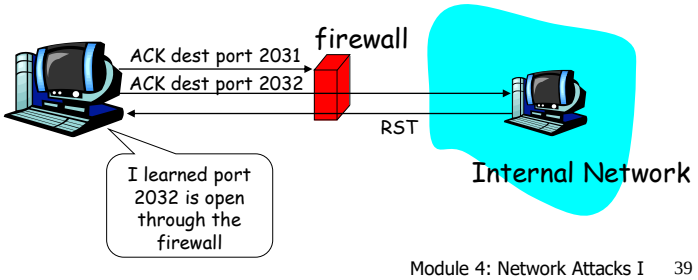
- `nmap -sT -v target.com`
- Attempts to complete 3-way handshake with each target port
- Sends SYN, waits for SYNACK, sends ACK, then sends FIN to close connection
- If target port is closed, no SYNACK returned
  - Instead RST packet is typically returned
- TCP connect scans are easy to detect
  - Target (e.g. Web server) may log completed connections
  - Gives away attacker's IP address

Nmap: TCP SYN port scan

- `nmap -sS -v target.com`
- Stealthier than polite scan
- Send SYN, receive SYNACK, send RST
- Stealthier: hosts do not record connection
  - But routers with logging enabled will record the SYN packet
- Faster: don't need to send FIN packet

Nmap: TCP ACK scans

- Example: `nmap -PA -v target`
- Many filters (in firewalls and routers) only let internal systems hosts initiate TCP connections
  - Drop packets for which ACK=0 (ie SYN packet): no sessions initiated externally
- To learn what ports are open through firewall, try an ACK scan (segments with ACK=1)



Nmap: UDP port scans

- UDP doesn't have SYN, ACK, RST packets
- nmap simply sends UDP packet to target port (example: `nmap -PU target`; may require root access)
  - ICMP Port Unreachable: interpret port closed
  - Nothing comes back: interpret port open
    - False positives common

Nmap: Obscure source

- Attacker can enter list of decoy source IP addresses into Nmap
- For each packet it sends, Nmap also sends packets from decoy source IP addresses
  - For 4 decoy sources, send five packets
- Attacker's actual address must appear in at least one packet, to get a result
- If there are 30 decoys, victim network will have to investigate 31 different sources!
- Example: `nmap -n -D IP1,IP2,…`

Nmap: TCP stack fingerprinting

- In addition to determining open ports, attacker wants to know OS on targeted machine:
  - exploit machine's known vulnerabilities
  - sophisticated hacker may set up lab environment similar to target network
- TCP implementations in different OSes respond differently to illegal combinations of TCP flag bits.
- Example: `nmap -O target`

Nmap: Fingerprinting

- Nmap sends
  - SYN to open port
  - NULL to open port (no flag bits set)
  - SYN/FIN/URG/PSH to open port
  - SYN to closed port
  - ACK to closed port
  - FIN/PSH/URG to closed port
  - UDP to closed port
- Nmap includes a database of OS fingerprints for hundreds of platforms

Nmap: more examples

- `nmap -v target.com`
  - Scans all TCP default ports on target.com; verbose mode
- `nmap -sS -O target.com/24`
  - First pings addresses in target network to find hosts that are up. Then scans default ports at these hosts; stealth mode (doesn't complete the connections); tries to determine OS running on each scanned host
- `nmap -sX -p 22,53,110,143 198.116.*.1-127`
  - Sends an Xmas tree scan to the first half of each of the 255 possible subnets in the 198.116/16. Testing whether the systems run ssh, DNS, pop3, or imap
- `nmap -v -p 80 *.*.2.3-5`
  - finds all web servers on machines with IP addresses ending in .2.3, .2.4, or .2.5

Notes and Warnings when using nmap

- GUI versions available: zenmap:
  - http://nmap.org/zenmap/
- USE CAREFULLY
  - Do not scan entire network
  - Scanning a host for testing/learning purposes is fine
  - Please keep in mind the ethics of security education
    - Lab will be the safest platform to try it

Defenses against network scanning

- Filter using firewalls and packet-filtering capabilities of routers
  - Block incoming ICMP packets, except to the hosts that you want to be pingable
  - Filter Time Exceeded ICMP messages *leaving* your network
- Close all unused ports
- Scan your own systems to verify that unneeded ports are closed
- Intrusion Detection Systems
  - e.g., Snort