CS326 – Systems Security

Lecture 18

Attacking and Defending the Network

Elias Athanasopoulos
athanasopoulos.elias@ucy.ac.cy

Local vs Remote attacker

- Local attacker
  - `program \`printf "'\xc0\xbf…"\``
- Remote attacker
  - `wget http://victim/\xc0\xbf…`

Remote Inputs

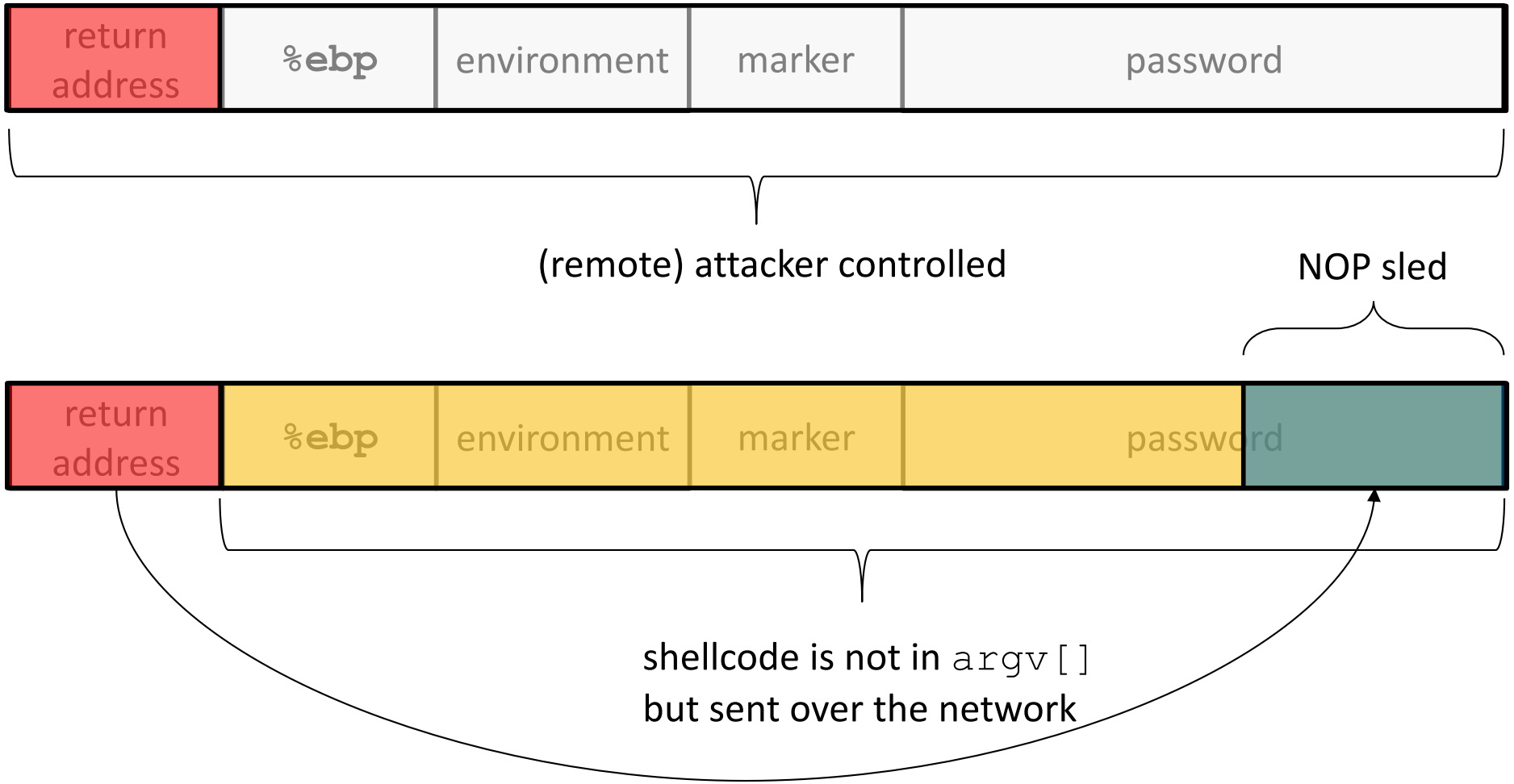
- Programs can take inputs from the network
- Inputs received using sockets
- Examples
  - A web server processes HTTP requests
  - A web browser processes HTML documents
  - A DNS server processes DNS requests
  - An e-mail server processes SMTP, IMAP, and POP3 commands

Remote Exploitation

- Shellcode should be embedded in a *network payload*
- Example
  - A web server includes a buggy function to parse URL parameters
  - `http://victim/fetch?par1=AA&par2=\xc0\xbf...`

shellcode

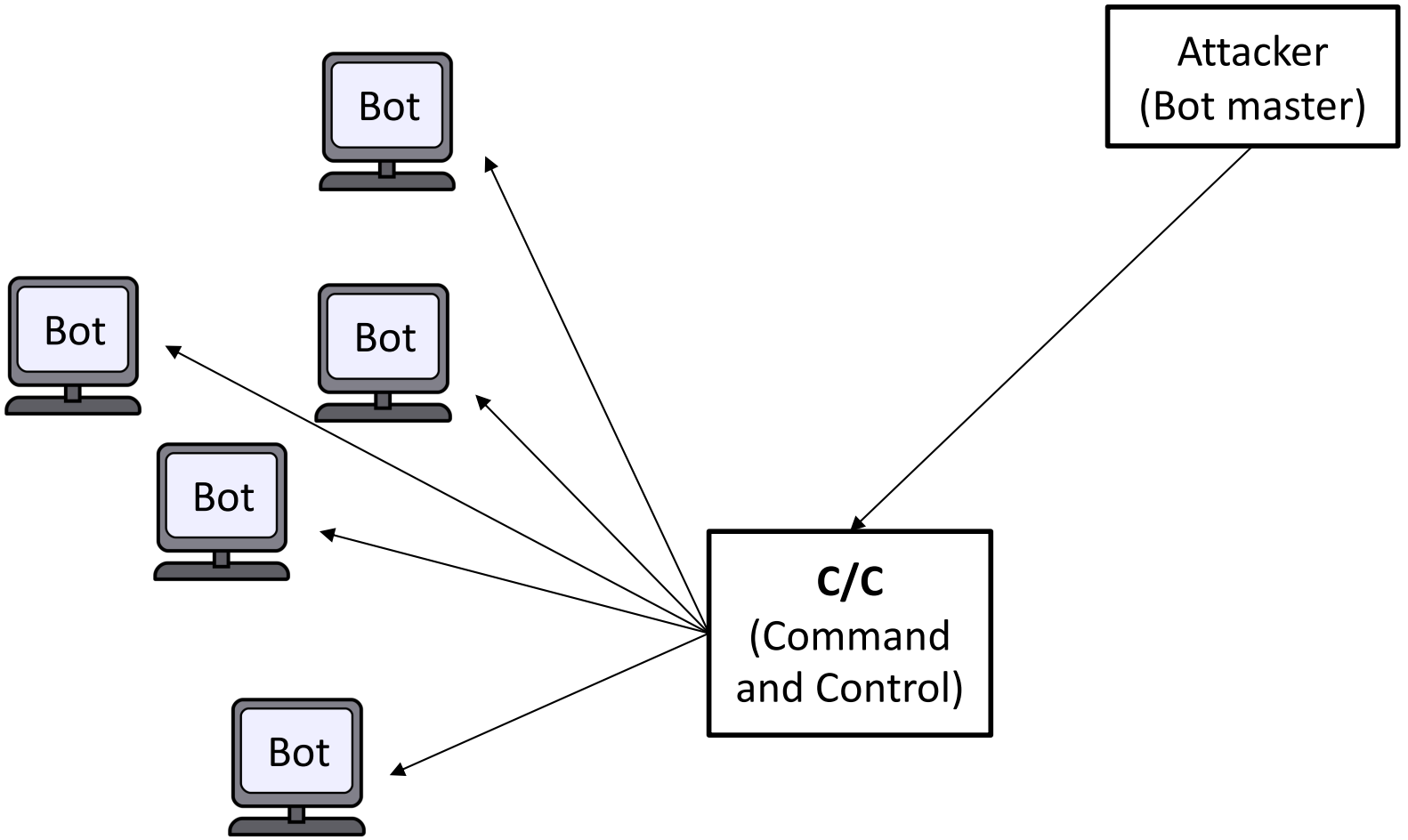
Remote Exploitation



Remote Attacker's Goals

- Servers
  - Usually contain valuable data
- Hosts
  - An attacker can control several ordinary hosts (bots)
  - These bots comprise a BotNet (army of compromised machines)
- Users
  - Compromise massively users (e.g., Ransomware)

BotNet

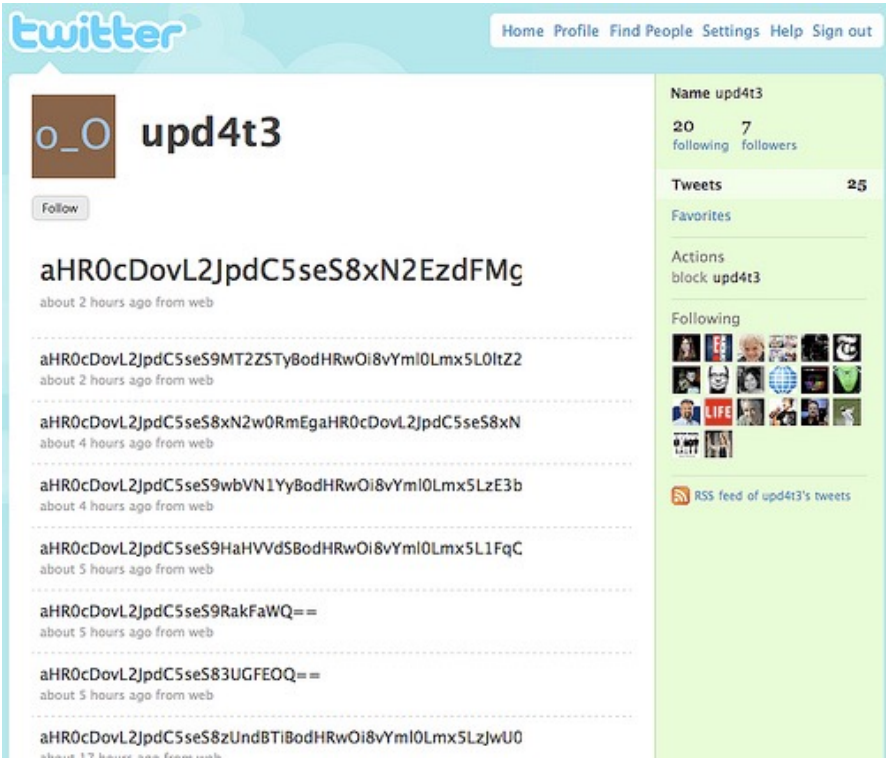

Attacker
(Bot master)
Bot
Bot
Bot
Bot
Bot
C/C
(Command
and Control)

BotNets

- A large collection of compromised hosts that can be controlled by an attacker (Bot master)
- Can be rent for all sorts of malicious activities
  - Click fraud
  - SPAM
  - Facebook/Twitter Likes or Retweets
  - Distributed Denial of Service (DDoS) attacks

BotNet C/C

- Bot master controls the BotNet through a hidden command and control channel
- Bots periodically check this channel to receive new commands
  - Check a twitter account for new tweets that embed commands
  - Command payload is encrypted


twitter
Home Profile Find People Settings Help Sign out
o_O
upd4t3
Follow
aHR0cDovL2JpdC5seS8xN2EzdFMg
about 2 hours ago from web
aHR0cDovL2JpdC5seS9MT2ZSTyBodHRwOi8vYml0Lmx5L0ltZ2
about 2 hours ago from web
aHR0cDovL2JpdC5seS8xN2w0RmEgaHR0cDovL2JpdC5seS8xN
about 4 hours ago from web
aHR0cDovL2JpdC5seS9wbVN1YyBodHRwOi8vYml0Lmx5LzE3b
about 4 hours ago from web
aHR0cDovL2JpdC5seS9HaHVVdSBodHRwOi8vYml0Lmx5L1FqC
about 5 hours ago from web
aHR0cDovL2JpdC5seS9RakFaWQ==
about 5 hours ago from web
aHR0cDovL2JpdC5seS83UGFEOQ==
about 5 hours ago from web
aHR0cDovL2JpdC5seS8zUndBTiBodHRwOi8vYml0Lmx5LzJwU0
Name upd4t3
20 following
7 followers
Tweets 25
Favorites
Actions
block upd4t3
Following
RSS feed of upd4t3's tweets

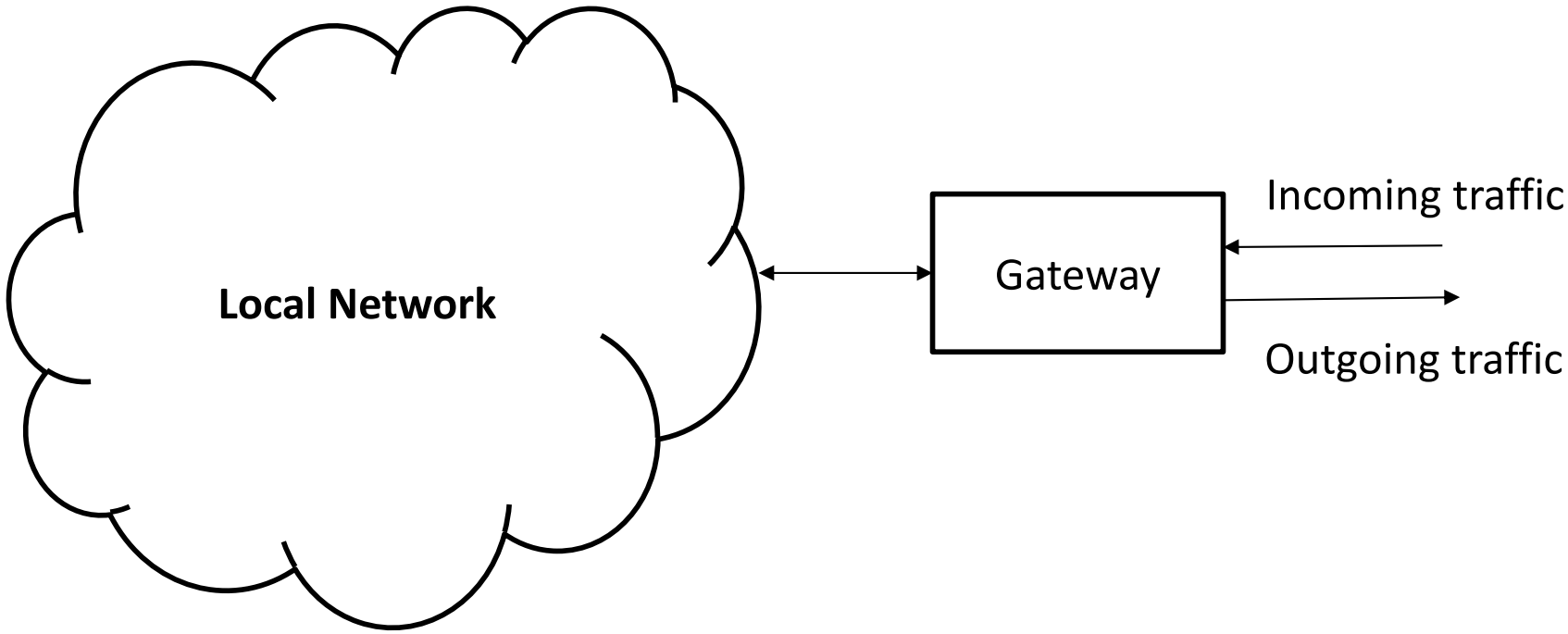
Network Scanning

- Interact with other hosts remotely to infer
  - **Operating System**, based on slightly different implementations of network protocols
  - **Running services**, based on different ports
  - **Versions of installed software**, based on application-layer replies
- `nmap`

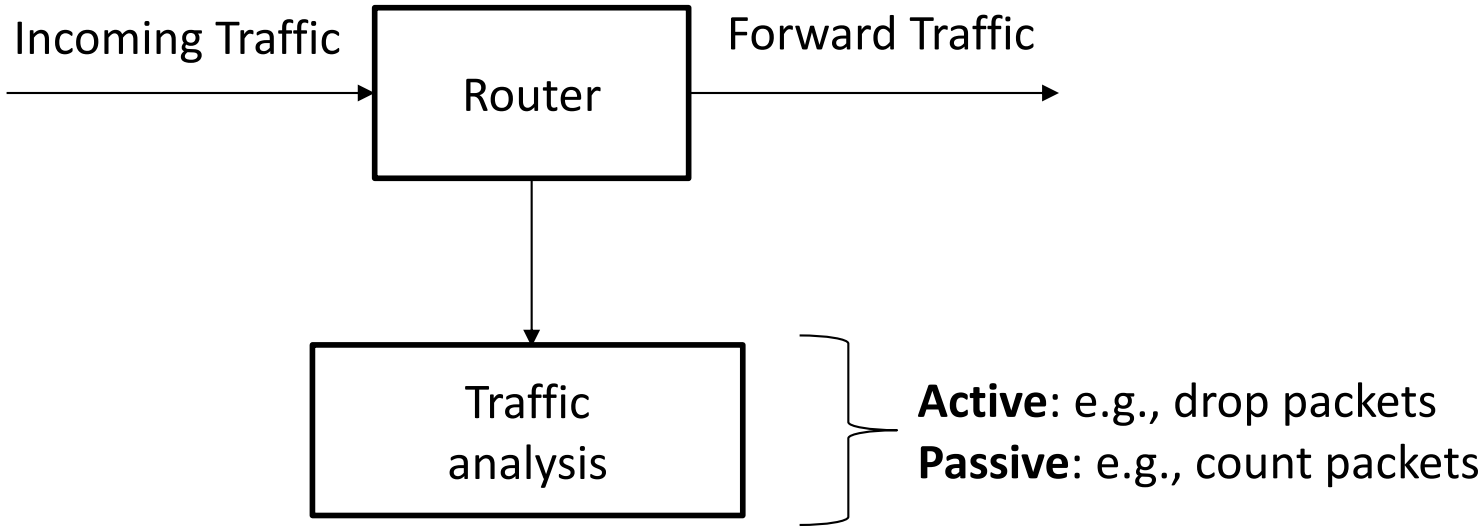
Network Monitoring

- Record and process network traffic
- Detect known attacks
- Detect anomalies
- Drop malicious traffic

Monitor Placement



Monitor


Incoming Traffic
Router
Forward Traffic
Traffic analysis
Active: e.g., drop packets
Passive: e.g., count packets

Firewalls

- Use a rule set with allowed services
- Inspect packet headers
  - Do not inspect the payload!
  - Relatively fast
- Enforce rules
  - E.g., drop all ICMP packets with ECHO_REQUEST
  - Drops pings

Intrusion Detection System (IDS)

- Inspect the payload of every packet
  - Deep Packet Inspection (DPI)
  - Slow, use of regular expressions
- Take decisions based on payloads (e.g., packets carrying shellcode)
- Complicated signatures

Monitor Framework

- `libpcap`
  - Packet CAPture library
  - `tcpdump, wireshark`
- Development of applications that can monitor and process network traffic

Berkley Packet Filter (BPF)

- Filter captured traffic
  - Sometimes only particular network traffic is interesting
- BPF expression anatomy
  - *Type:* qualifiers say what kind of thing the id name or number refers to. Possible types are **host**, **net, port** and **portrange**.
  - *Dir:* qualifiers specify a particular transfer direction to and/or from *id*. Possible directions are **src**, **dst**, **src or dst** and **src and dst**.
  - *Proto:* qualifiers restrict the match to a particular protocol. Possible protos are: **ether**, **fddi**, **tr**, **wlan**, **ip**, **ip6**, **arp**, **rarp**, **decnet**, **tcp** and **udp**.

BPF examples

- host foo
  - Capture all packets *from* or *to* foo
- ip host ace and not helios
  - Capture all IP packets between *ace* and any host except *helios*
- tcp port 80
  - Capture all tcp packets *from* or *to* port 80

BPF Expressions Language

- Mandatory Read
  - http://alumni.cs.ucr.edu/~marios/ethereal-tcpdump.pdf
- Suggested Read
  - http://www.tcpdump.org/papers/bpf-usenix93.pdf