OXFORD BROOKES UNIVERSITY

COMP6016 – Command and Control

Dr. MUHAMMAD HILMI KAMARUDIN

OXFORD BROOKES UNIVERSITY

Lecture Learning Outcomes

- By the end of today's lecture and practical you should –

- Understand the principles of command and control
- Knowledge on top existing network analysis tools
- Understand how to capture and analyse network traffic

Command & Control (C2)

OXFORD BROOKES UNIVERSITY

- Called C&C or C2
- Maintain communication with compromise system
- Like maintaining a timed beacon or "heartbeat" so that the operators running the attack can keep an inventory of the systems they have compromised
- Malware enters the enterprise through a number of channels
- Typical method is through email phishing attempts
- Infection Stages:

Many pieces of malware just ping back to the C&C to get further instructions/functionality so as to stay small and manageable. This also let's us establish a foothold in more systems with simpler pieces of malware and then implement functionality itself within the server so that it's downloaded.

1) malware or agent is dropped on the target computer as an email attachment or drive-by link, and executed, it will load a malicious program to begin its operations

2) this program is unpacked within the operating kernel as a device driver to maintain persistence on the system and evade detection.

3) malware code sends a beacon out to the command-and-control servers to look for its next instruction set.

4) If the target's network has liberal outbound or egress firewall rules, the malware will establish a communication channel with the command-and-control server.

5) Once the communications channel is established, the command-and-control server will instruct the malware to download additional rootkits and remote access tools on the compromised host.

Detection

OXFORD BROOKES UNIVERSITY

- Malware today is defined by the communications of the compromised host with the command-and-control network
- The network signatures are able to identify known communications of the compromised host
- Out-of-the-box antivirus and malware signatures often fail to identify current indicators of compromise (IOCs)
- Several indicators can be identified through analysis of potential malware in a sandbox or live environment Cuckoo Sandbox
- A lot of command-and-control programs communicate using direct-to-IP-address HTTP requests

No need for DNS queries, simpler malware implementation

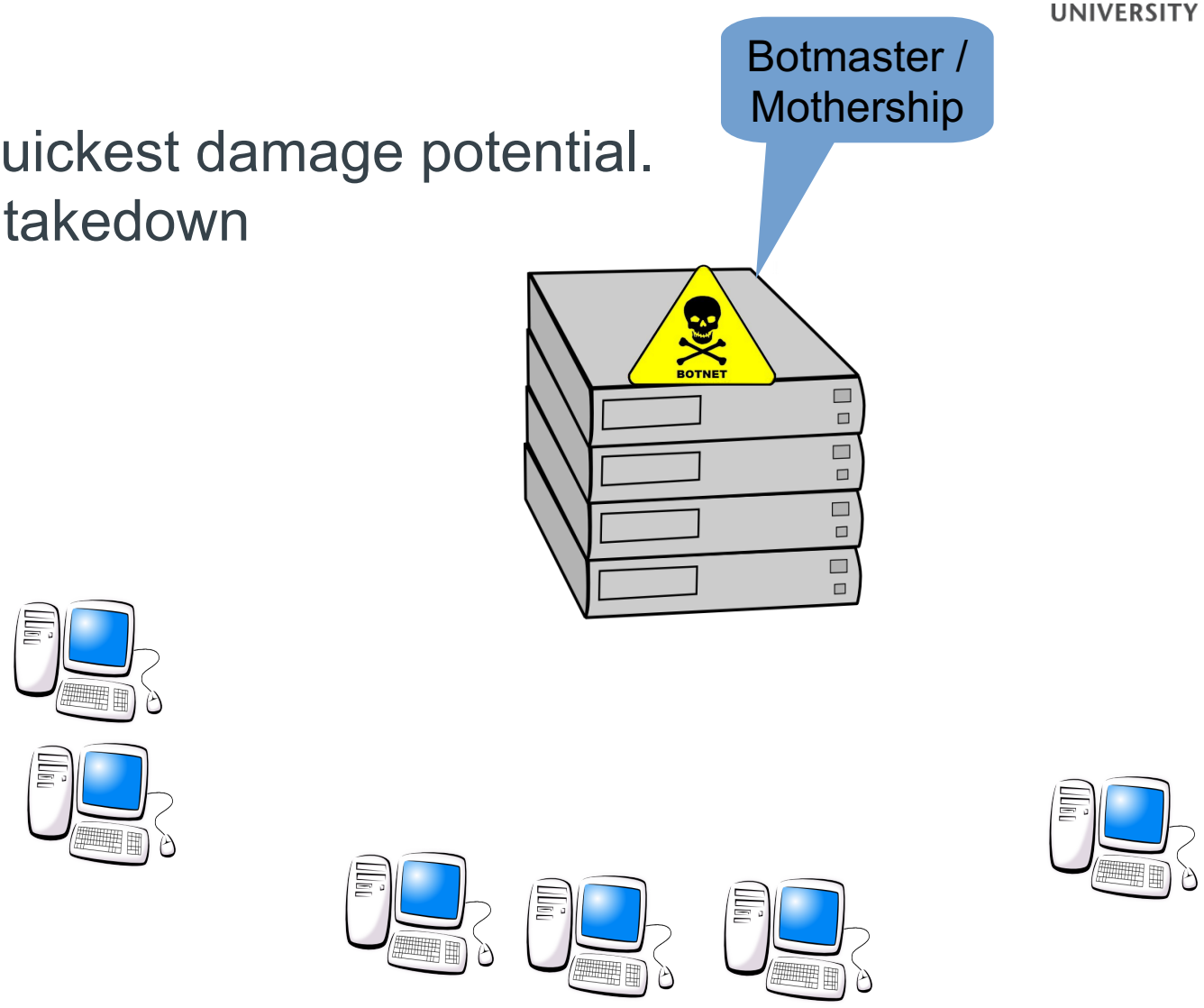
We can dump traffic to then statically analyze it with tcpdump and wireshark

- Popular command-and-control communications technique is to use publicly available DNS servers
- Another technique to avoid detection is to leverage dynamic DNS hosting sites.

Command & Control architecture – Centralised

OXFORD BROOKES UNIVERSITY

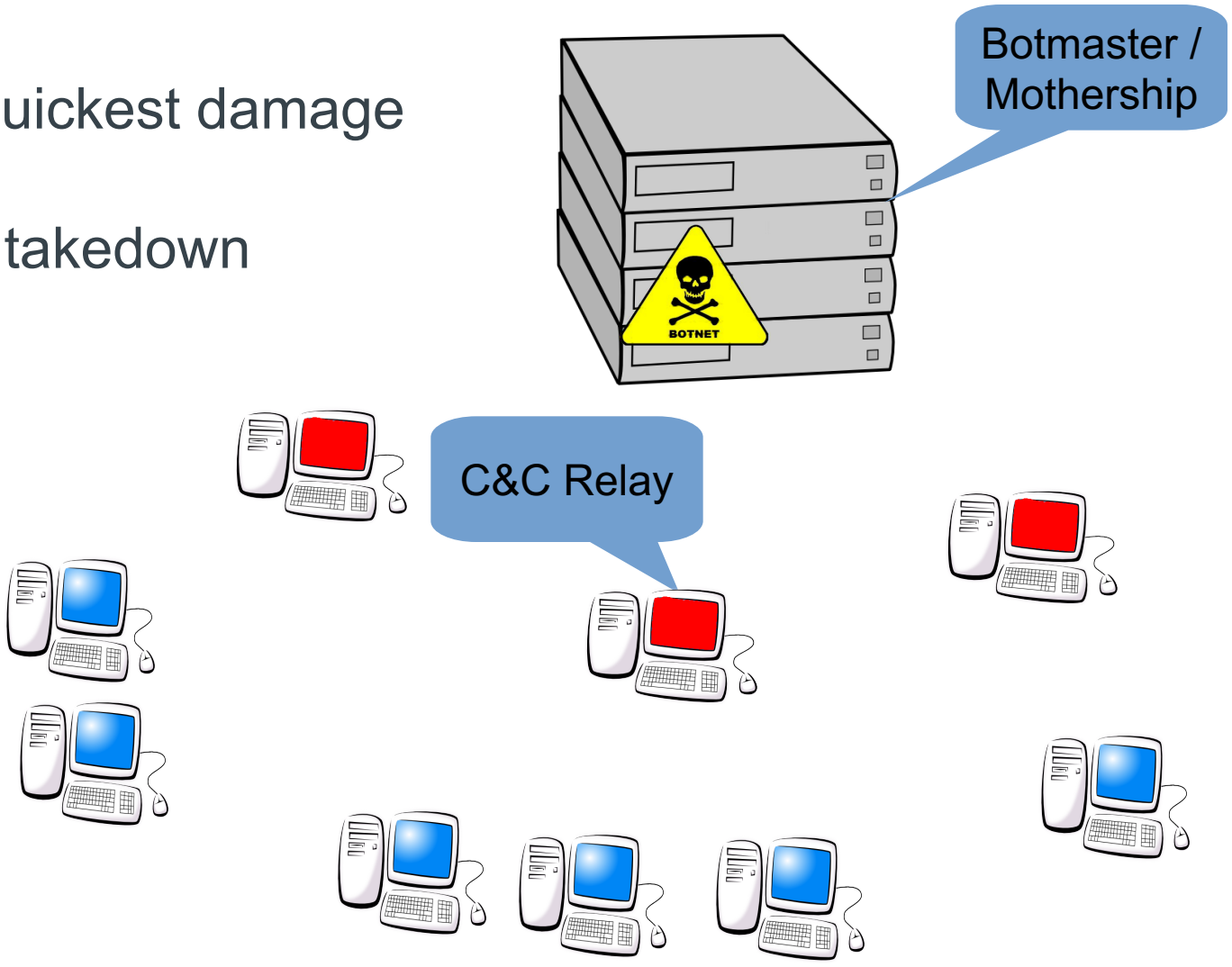
- Simplest and quickest damage potential.
- Not resilient to takedown



C&C architecture – Hierarchical

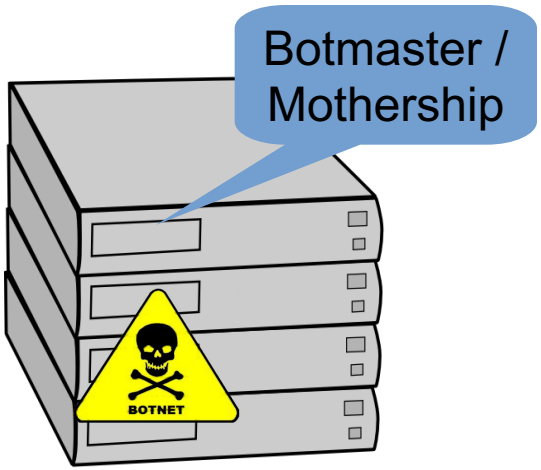
OXFORD BROOKES UNIVERSITY

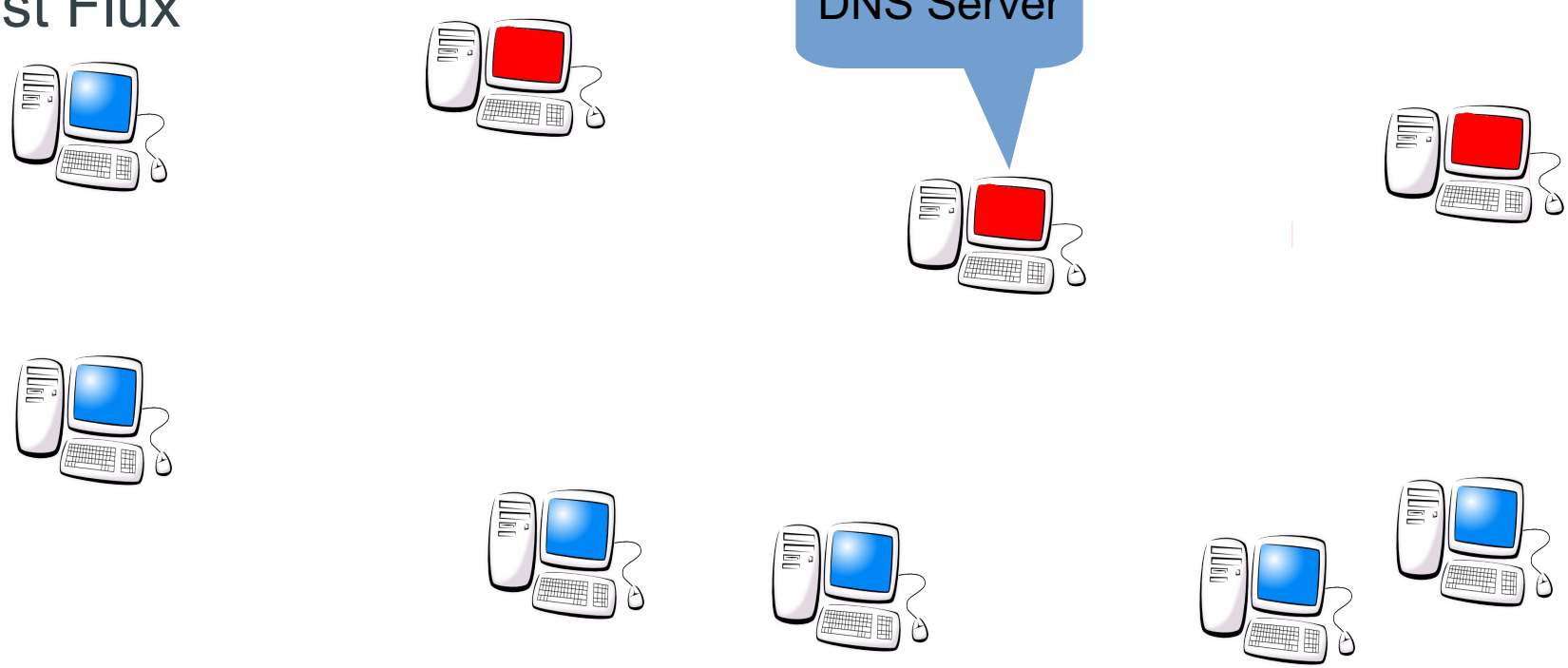
- Simplest and quickest damage potential.
- Not resilient to takedown



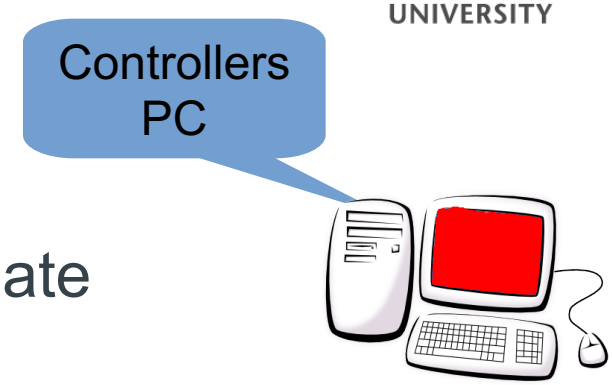
C&C architecture – Distributed DNS

- Hardest to detect and destroy
- Uses DNS and own DNS servers to provide resiliance
- Fast Flux

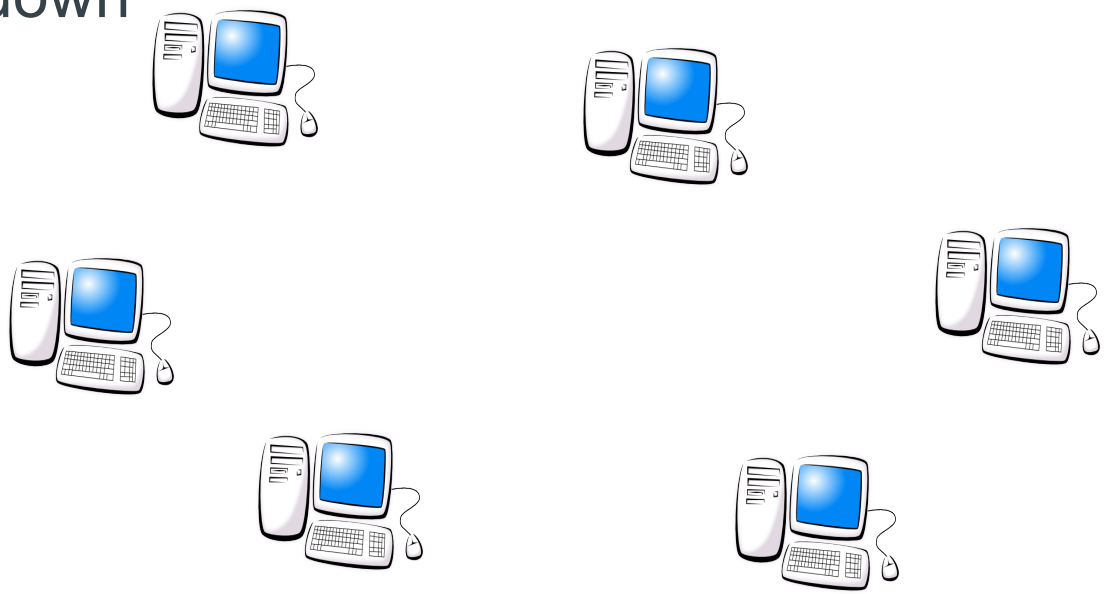




C&C architecture – Peer to peer



- More modern approach with limited immediate attack potential
- Not resilient to takedown



C&C Capture

OXFORD BROOKES UNIVERSITY

- Actively trying to disrupt the C&C of malware is not recommended unless you have substantial resources at your disposal (i.e. the authorities, government agencies, Google, MS, etc.)
- Often difficult to compromise the mothership without breaking the law
- Compromising the command relays by redirecting them to a fake mothership

Revision - ARP

OXFORD BROOKES UNIVERSITY

- **Scenario**: Computers A,B,C,D are in the same local network (e.g. your house). Computer A wishes to send information to computer B. Lets see how this is done.

- Computer A checks its ARP cache to determine whether it can map the target IP address to a MAC address (aka computer B). *(Note: check the ARP cache of your machine by running arp -an)*

- If the address is not listed in the ARP cache then Computer A sends a broadcast message across the entire network asking for the owner of that IP address. (*Note: The broadcast message is created with a MAC address of FF-FF-FF-FF-FF-FF*)

- The owner of that IP address then sends back an ARP packet containing the MAC address associated with that IP address back to computer A.

- Once the request is received, computer A stores the MAC address in its ARP cache and can now communicate to computer B until the cache expires.

Network sniffing – the basics

OXFORD BROOKES UNIVERSITY

- Passive sniffing -
- Use a **tap** which sits inline between two devices
- Use a **hub** which broadcasts all traffic sent between devices
- Sniffer sets NIC in promiscuous **mode**
- Hosts are not aware of the eavesdropping
- Sniffer is silent (though may be discoverable)

OXFORD BROOKES UNIVERSITY

Network sniffing – the basics

- Active sniffing -
- Using layer 2 switches
- Attacker poisons protocols to redirect traffic
- Attacker compromises switch to put it into **span/mirror** mode
- Detectable

OXFORD BROOKES UNIVERSITY

Protocol analysers

- Hardware
- Purpose built hardware device to capture (and sometimes analyse) network traffic in real time
- Necessary for large bandwidth networks
- Software
- Software (e.g. wireshark) runs on local machine (normally laptop)
- NIC is set into **promiscuous mode** and reads all traffic
- Good for quick analysis, but not so good for large bandwidth situations

The driver tells the NIC to pass ALL the traffic it receives to the network stack, not only the one addressed at it. This only makes sense in a wireless scenario!

Online and offline analysis

OXFORD BROOKES UNIVERSITY

Online

- Capture and analysis are done in realtime
- Traffic is often filtered to make analysis manageable
- Issues with large bandwidth

Offline

- Packets are captured in real time, analysis is done offline
- Allows more interactive investigation of all packets
- Packet capture files may be big
- Storage media must be ***very*** fast (SSDs normally)

Network analysis tools

OXFORD BROOKES UNIVERSITY

- In order to analyse the network traffic in malware we need a range of tools to help us a) capture the traffic, b) analyse the traffic, and c) spoof the traffic.
- Common tools are –

- Wireshark – common network capture and analysis tool
- tcpdump – capture tool useful for capturing traffic for later analysis
- ApateDNS – DNS spoofer
- INetSim – fake network services system
- Ettercap - network traffic capture and redirection system
- Netcat - “TCP/IP Swiss Army knife”, used to easy read and write to TCP & UDP connections

Wireshark

- Video available at –

- https://www.youtube.com/watch?v=TkCSr30UojM

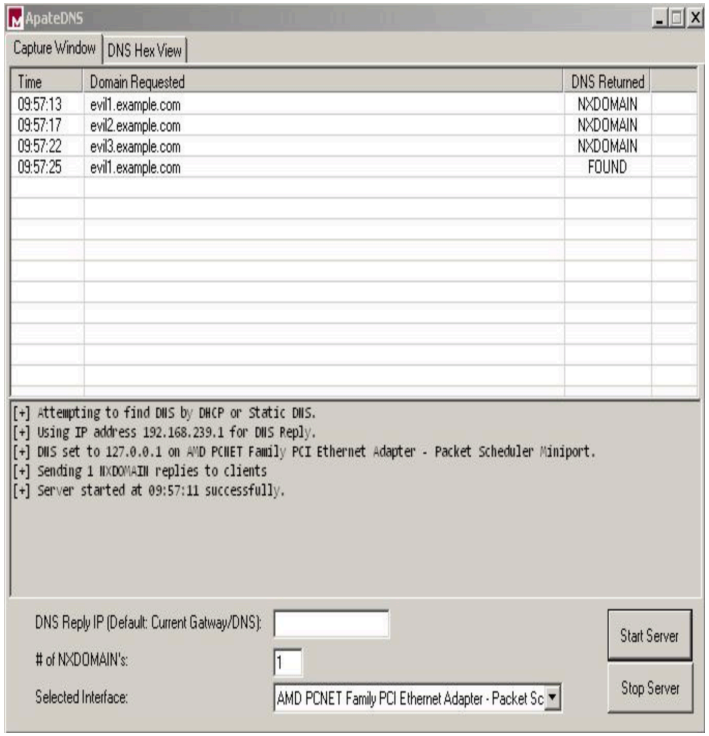
TCPDump

- Data-network packet analyzer run under CLI
- Packet capture format is standard used by many other tools
- Default choice if packet capture is needed for offline analysis
- WinDump is windows port (unix like OSes virtually all have TCPDump)

ApateDNS

OXFORD BROOKES UNIVERSITY

- A windows based DNS spoofing tool by FireEye - https://www.fireeye.com/services/freeware/apatedns.html
- Listens in on UDP port 53 and sets up fake DNS server
- The IP address or hostname is often retrieved from the malware by performing static malware analysis
- Spoofs DNS responses to a user-specified IP address
- Will not work if malware has hardcoded DNS server, but in most situations will provide suitable DNS capture



OXFORD BROOKES UNIVERSITY

INetSim

- Software for simulating common internet services in a lab setting - http://www.inetsim.org/
- Designed to provide easy to use and setup internet services that behave almost like the real thing but are much more configurable (i.e. you can pretend to be a Windows IIS server on a Linux box)
- Primary servers are HTTP(S), SMTP(S), POP3(S), DNS, FTP(S), TFTP, IRC, NTP, Ident, Finger, Syslog, and Dummy
- INetSim fakes the common internet services which malware might use and answers the requests made accordingly
- When malware scans a webserver, INetSim will return a Microsoft IIS webserver banner in order to keep the malware running.

OXFORD BROOKES UNIVERSITY

Ettercap (Available in Kali Linux)

- Man in the middle attack tool for *nix OSes used to compromise switched networks using ARP poisoning link here: https://www.youtube.com/watch?v=3UD738uE7Tg
- Main functionality -
- Sniff on a switched LAN between 2 hosts (full duplex)
- Sniff traffic on a switched LAN between target and all other hosts (half duplex)
- SSH1 sniffing
- HTTPS sniffing
- Built in password collectors for plaintext protocols
- DNS hijacks
- Connection killing
- Detection of other poisoners

OXFORD BROOKES UNIVERSITY

Netcat & Ncat (Available in Kali Linux)

- Multiplatform tool used to easily create bespoke connections and servers
- Designed to be easily scriptable
- Used to transmit crafted network packets to target destinations
- Used to easily create simple servers to respond to packets
- Netcat can be used to make inbound and outbound connections on any port and can be used in client mode for connecting and in server mode for listening.

```
ncat --exec "/bin/bash" --max-conns 3 --allow
192.168.0.0/24 -l 8081 -keep-open
```

- Ncat is more modern version of netcat developed by the Nmap developers
- https://www.youtube.com/watch?v=iGGB4dMCq28

OXFORD BROOKES UNIVERSITY

Aside - OPSEC

- **Operational SECurity** – the process of ensuring that we don't leak sensitive information when doing an analysis
- Sensitive information can be information about yourself, or even the simple fact that you are analysing the malware
- Information leaks can result in you being attacked, or the attacker just hiding
- Care needs to be taken to ensure that any investigation is done ***SAFELY***

Connection hiding

OXFORD BROOKES UNIVERSITY

- It is often inadvisable to connect from your own network – care must be taken to hide your online identity.
- Possible solutions are -
- Mobile phone hotspot
- AWS (or similar) instance
- Use of a VPS (Virtual Private Server)
- Online research tool
- Domain and IP registry (for domain checks – suggest RobTex (http://www.robtex.com/) or mxtoolbox (mxtoolbox.com/dnscheck.aspx))

OXFORD BROOKES UNIVERSITY

Network analysis – the process – part 1

- Identify ***all*** traffic endpoints and categorise into -
- Green – known protocol and endpoints
- Orange – known protocol, unknown endpoints
- Red – unknown protocol
- Start with red first and work your way up to green. DO NOT presume that green is safe (it is just more likely to be)
- **NB** we are just considering the network traffic here. If possible, you should also examine the code and look for networking code

OXFORD BROOKES UNIVERSITY

Network analysis – the process – part 2

- Filter off unknown protocols and analyse
- Can start with the protocol or the endpoints
- If starting with the protocol consider the following -
- Is there a TCP connection with no handshake? (used to bypass firewalls)
- Is the protocol actually a known one on a non standard port? (not always an issue, but still an alarm bell)
- Is the protocol encrypted? (may require code investigation to break)
- What plaintext can you see in the protocol?

OXFORD BROOKES UNIVERSITY

Network analysis – the process – part 2

- Plaintext extraction is a very powerful tool for unknown protocol analysis
- Simplest way is to run strings command on packet trace taken by something like TCPDump
- TCPReplay + Diftnet can also be used to look for images which can also be useful

OXFORD BROOKES UNIVERSITY

Network analysis – the process – part 3

- If starting with the endpoints consider the following -
- Should that endpoint be receiving information on that port? (for local machine)
- What is the destination endpoints FQDN? (look for obvious consumer FQDNs – not always an issue (esp with ISPs which are home & business), but still a warning flag)
- Where is the destination endpoint in terms of network topology? (again, IPs used by consumer ISPs are a warning)
- Where is the destination endpoint in terms of geographical location? (Is it likely that you would be talking to a server there?)

OXFORD BROOKES UNIVERSITY

Network analysis – the process – part 4

- Don't presume that known endpoint and protocol are safe – malware will often use known protocols to hide itself
- Things to watch out for -
- Connections to Amazon, Google, MS, etc virtualised infrastructure
- Multiple repeated requests with the same information to different IPs (possible C&C infrastructure)
- Multiple unusual requests (e.g. WHOIS requests, lots of DNS requests to machines on the same domain)

OXFORD BROOKES UNIVERSITY

Network analysis – the process – part 5

- Determine the command protocol and establish what information is sent and received and how. This is best done by –

- Examining the code
- Matching byte patterns to malware action (including possible IP addresess, etc.)
- Plaintext extraction
- Replay attacks

Network analysis – the process – part 6

OXFORD BROOKES UNIVERSITY

- Determine the control infrastructure and where the mothership (if possible) or command relays are. This is best done by

- Monitoring changeover of command endpoints (indicates a hidden mothership)
- Blocking endpoints and seeing where and how changes occur
- Investigating the suspected C&C endpoints (be careful not to do anything illegal or spook the attacker)
- Reporting to the authorities