Botnets and Cybercrime

Corso di Sicurezza delle reti e dei sistemi software aa 2016/17

Ing. Antonio Pirozzi

Some definitions

- “A botnet is a collection of computers, connected to the internet, that interact to accomplish some distributed task.” 1
- Controlled by one person or a group of people (aka. the botmaster) under a command and control structure (C&C)
- The word botnet is a combination of the words robot and network.

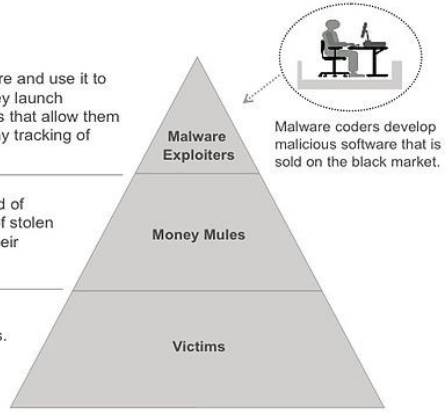
1 www.shadowserver.org

Cyber Theft Ring

Malware exploiters purchase malware and use it to steal victim banking credentials. They launch attacks from compromised machines that allow them to transfer stolen funds and deter any tracking of their activities.

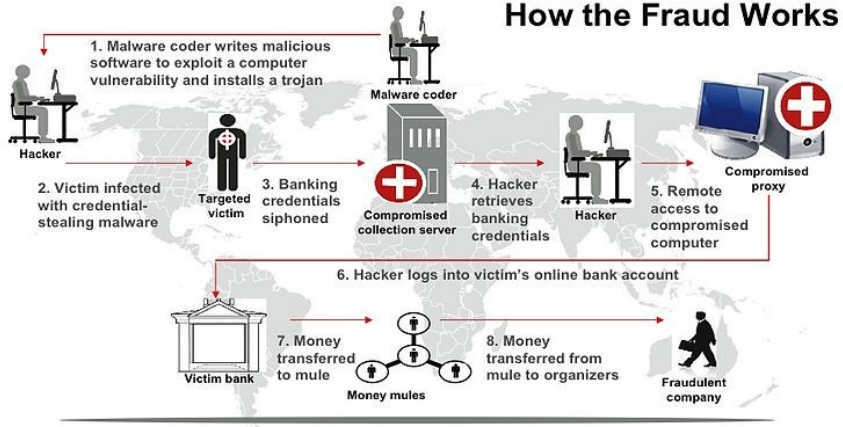
Money mule networks are comprised of individuals engaged in the transfer of stolen funds who retain a percentage for their services.

Victims include individuals, businesses, and financial institutions.

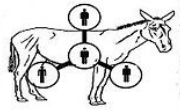


Malware coders develop malicious software that is sold on the black market.

How the Fraud Works



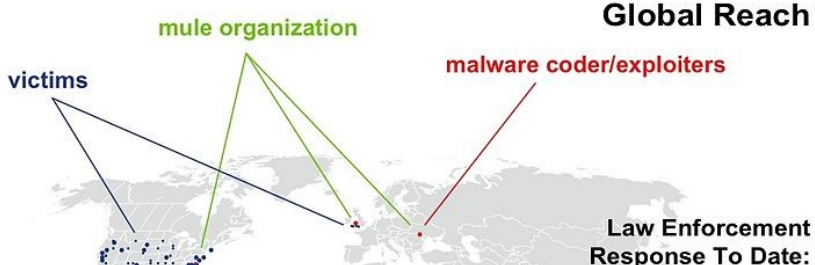
Victims are both financial institutions and owners of infected machines.



Money mules transfer stolen money for criminals, shaving a small percentage for themselves.

Criminals come in many forms:
- Malware coder
- Malware exploiters
- Mule organization

Global Reach



Law Enforcement Response To Date:

Total FBI cases: 390
Attempted loss: $220 million
Actual loss: $70 million

United States: 92 charged and 39 arrested

Botnet components

- **C&C** : The C&C manages a list of infected machines; it monitors their status and gives them operative instructions.
- **BOT**: A "bot" is a type of malware that allows an attacker to take control over an affected computer. Also known as “Web robots”, bots are usually part of a network of infected machines, known as a botnet 2
- **Zombie**: victim machines

2 https://us.norton.com/botnet

Botmaster

Police have arrested Algerian national Hamza Bendelladj, who is wanted by the FBI for cyber crimes. (Photo by Somchai Poomlard)

known as **Bx1** is an Algerian computer hacker.
Using log-in information obtained from a
Trojan horse called SpyEye
was sentenced to 15 years in prison

On May 9, 2006, Jeanson James Ancheta became the first person to be charged for controlling large numbers of hijacked computers or botnets

motivations

Cybercriminals cause harm with botnets in many ways:

Sending	**Stealing**	**DoS** (Denial of Service)	**Clickfraud**
They send - spam - viruses - spyware	They steal personal and private information and communicate it back to the malicious user: - credit card numbers - bank credentials - other sensitive personal information	Launching denial of service (DoS) attacks against a specified target. Cybercriminals extort money from Web site owners, in exchange for regaining control of the compromised sites. More commonly, however, the systems of everyday users are the targets of these attacks -- for the simple thrill of the botherder.	Fraudsters use bots to boost Web advertising billings by automatically clicking on Internet ads.

3

Additionally, botnets can also be used to mine bitcoins, intercept any data in transit, send logs that contain sensitive user information

3 https://us.norton.com/botnet

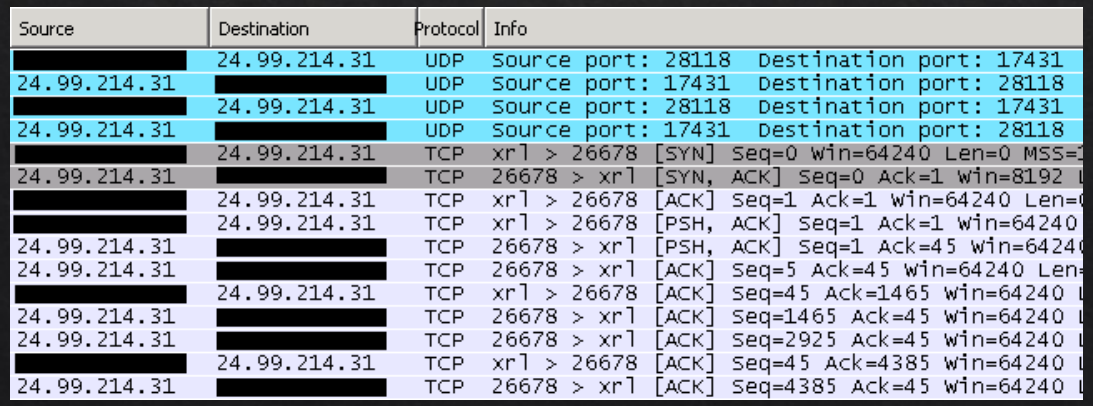
Architecture evolution

- Botnet Architecture evolved over time
- Advanced topology is more resilient to shutdown, enumeration or discovery
- **No need for C&C?** relying on a C&C server is a limitation

- IRC botnets were especially prevalent in the 1990s and early to mid 2000’s. Such botnets basically are comprised of a collection of infected systems that are controlled remotely via a preconfigured IRC server and channel.

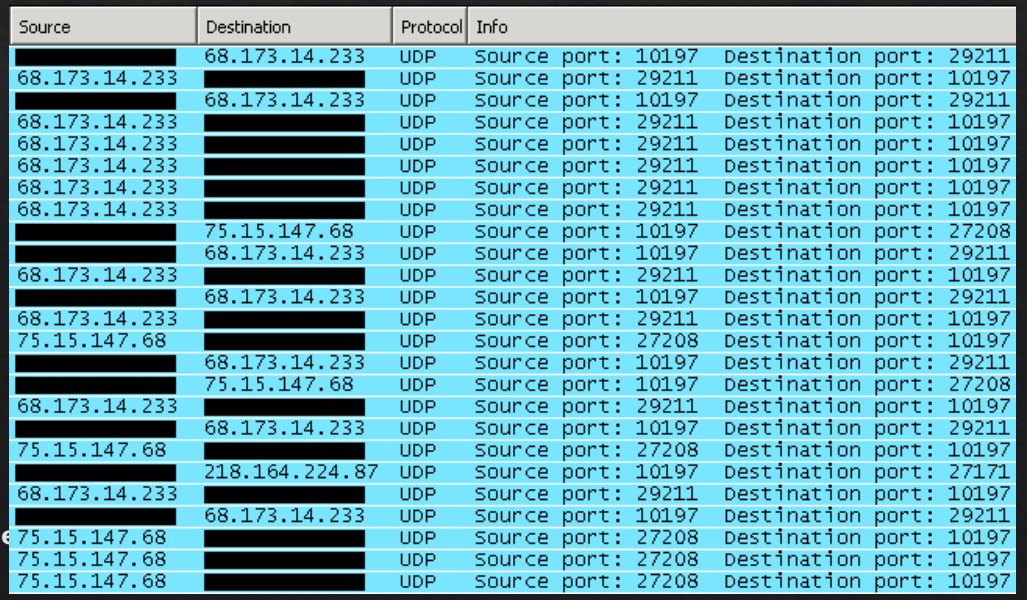
Architecture evolution

- **More UDP, less TCP**
  - The bots would contact each other by using a sort of homemade UDP handshake. If successful, this would cause the bots to exchange TCP data, such as configuration files, list of other peers, etc
  - TCP communications are easy to track and dump, and the bot does not perform any authentication on the packets exchanged, so anyone can impersonate a bot and successfully communicate with other bots, downloading stuff like configuration data.



Source	Destination	Protocol	Info
[redacted]	24.99.214.31	UDP	Source port: 28118 Destination port: 17431
24.99.214.31	[redacted]	UDP	Source port: 17431 Destination port: 28118
[redacted]	24.99.214.31	UDP	Source port: 28118 Destination port: 17431
24.99.214.31	[redacted]	UDP	Source port: 17431 Destination port: 28118
[redacted]	24.99.214.31	TCP	xrl > 26678 [SYN] Seq=0 Win=64240 Len=0 MSS=1
24.99.214.31	[redacted]	TCP	26678 > xrl [SYN, ACK] Seq=0 Ack=1 Win=8192 L
[redacted]	24.99.214.31	TCP	xrl > 26678 [ACK] Seq=1 Ack=1 Win=64240 Len=0
[redacted]	24.99.214.31	TCP	xrl > 26678 [PSH, ACK] Seq=1 Ack=1 Win=64240
24.99.214.31	[redacted]	TCP	26678 > xrl [PSH, ACK] Seq=1 Ack=45 Win=6424(
24.99.214.31	[redacted]	TCP	26678 > xrl [ACK] Seq=5 Ack=45 Win=64240 Len=
[redacted]	24.99.214.31	TCP	xrl > 26678 [ACK] Seq=45 Ack=1465 Win=64240 L
24.99.214.31	[redacted]	TCP	26678 > xrl [ACK] Seq=1465 Ack=45 Win=64240 L
24.99.214.31	[redacted]	TCP	26678 > xrl [ACK] Seq=2925 Ack=45 Win=64240 L
[redacted]	24.99.214.31	TCP	xrl > 26678 [ACK] Seq=45 Ack=4385 Win=64240 L
24.99.214.31	[redacted]	TCP	26678 > xrl [ACK] Seq=4385 Ack=45 Win=64240 L

https://www.symantec.com/connect/blogs/zeusbotspyeye-p2p-updated-fortifying-botne



Source	Destination	Protocol	Info
[redacted]	68.173.14.233	UDP	Source port: 10197 Destination port: 29211
68.173.14.233	[redacted]	UDP	Source port: 29211 Destination port: 10197
[redacted]	68.173.14.233	UDP	Source port: 10197 Destination port: 29211
68.173.14.233	[redacted]	UDP	Source port: 29211 Destination port: 10197
68.173.14.233	[redacted]	UDP	Source port: 29211 Destination port: 10197
68.173.14.233	[redacted]	UDP	Source port: 29211 Destination port: 10197
68.173.14.233	[redacted]	UDP	Source port: 29211 Destination port: 10197
68.173.14.233	[redacted]	UDP	Source port: 29211 Destination port: 10197
[redacted]	75.15.147.68	UDP	Source port: 10197 Destination port: 27208
[redacted]	68.173.14.233	UDP	Source port: 10197 Destination port: 29211
68.173.14.233	[redacted]	UDP	Source port: 29211 Destination port: 10197
[redacted]	68.173.14.233	UDP	Source port: 10197 Destination port: 29211
68.173.14.233	[redacted]	UDP	Source port: 29211 Destination port: 10197
75.15.147.68	[redacted]	UDP	Source port: 27208 Destination port: 10197
[redacted]	68.173.14.233	UDP	Source port: 10197 Destination port: 29211
[redacted]	75.15.147.68	UDP	Source port: 10197 Destination port: 27208
68.173.14.233	[redacted]	UDP	Source port: 29211 Destination port: 10197
[redacted]	68.173.14.233	UDP	Source port: 10197 Destination port: 29211
75.15.147.68	[redacted]	UDP	Source port: 27208 Destination port: 10197
[redacted]	218.164.224.87	UDP	Source port: 10197 Destination port: 27171
68.173.14.233	[redacted]	UDP	Source port: 29211 Destination port: 10197
[redacted]	68.173.14.233	UDP	Source port: 10197 Destination port: 29211
75.15.147.68	[redacted]	UDP	Source port: 27208 Destination port: 10197
75.15.147.68	[redacted]	UDP	Source port: 27208 Destination port: 10197
75.15.147.68	[redacted]	UDP	Source port: 27208 Destination port: 10197

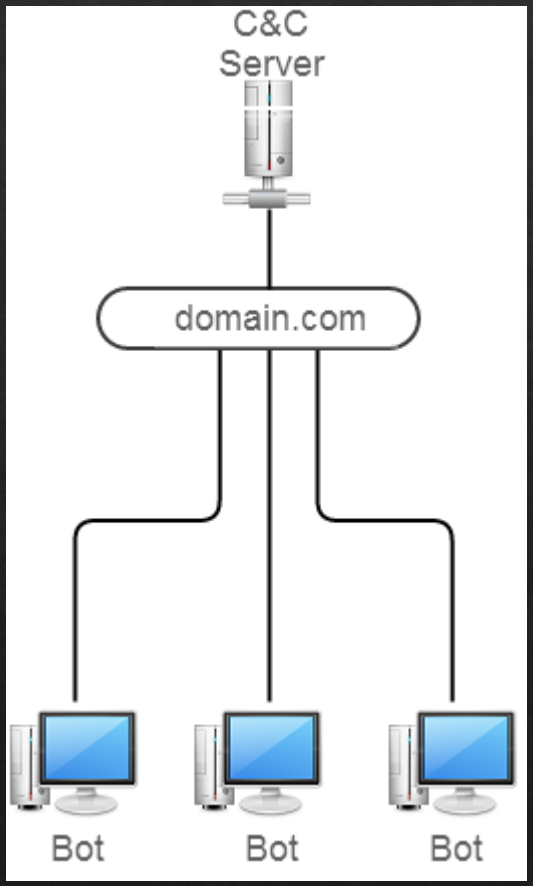
Architecture evolution

- **Changes in the compression and encryption**
  - data is still encrypted with RC4 and the XOR byte-with-preceding-byte + another added layer: a byte-per-byte XOR applied to each block of the configuration

https://www.symantec.com/connect/blogs/zeusbotspyeye-p2p-updated-fortifying-botnet

Architectures: client-server

- Centralized architecture
- built on Internet Relay Chat or by using Domains or Websites (HTTPS)
- The Bots connect to one or more server through one or more domains.
- C&C is the SPF. Easily to takedown
- The botmaster need to set up another C&C and redirect the domain to it.
- To take down this botnet, it would be required to suspend all domains associated or to seize the domain and point it to a functional server (sinkhole) in order to keep the bot away from the legit C&C.
- Rustock botnet and Srizbi botnet



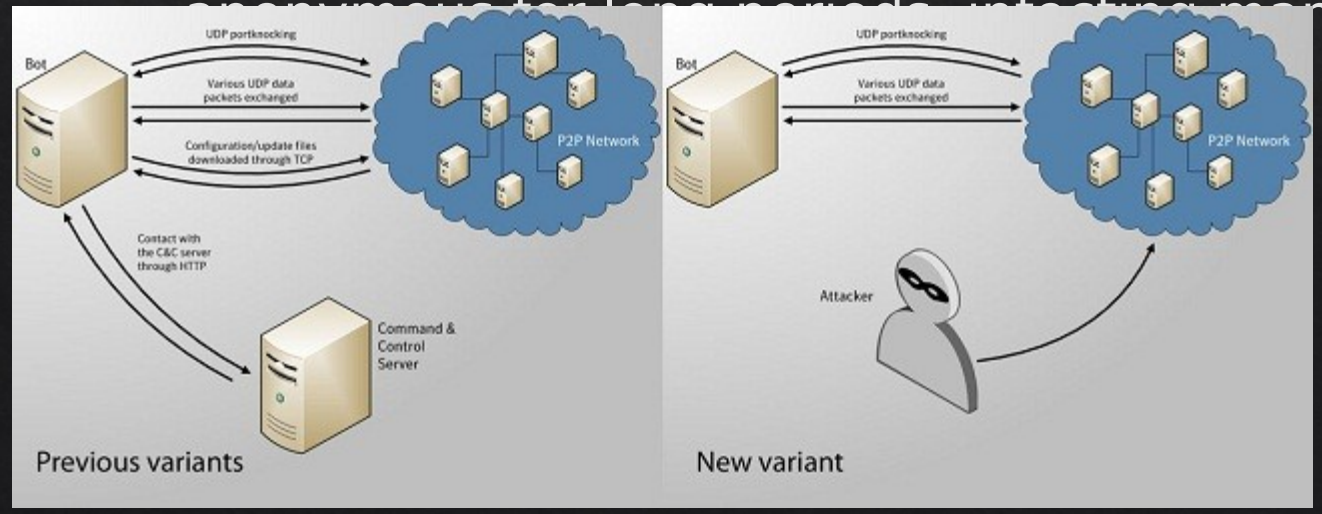
4 https://www.malwaretech.com/2013/12/peer-to-peer-botnets-for-beginners.htm

Architectures: P2P

- Decentralized Botnet Architecture
- Provides resiliency against network failures in the botnet
- Try to solve the problem of authorities targeting domains or server
- Properties:
  - The bots are not (necessarily) connected to C&C server
  - Bots are connected in a mesh network in which the commands are send from Zombie to Zombie
  - Each nodes mantains a list of IP addresses of other nodes “neighbor” with which they communicate and exchange commands
  - Commanders can be identified just through secure keys, and all data except the binary itself can be encrypted.

Architectures: P2P

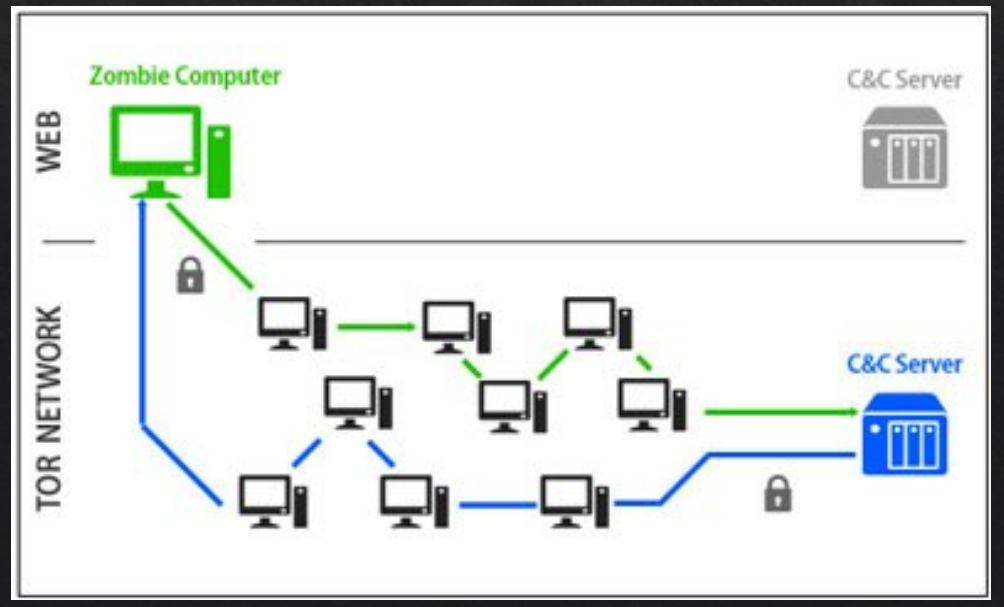
- Every peer in the botnet can act as a C&C server, while none of them really are one.
- Bots are now capable of downloading commands, configuration files, and executable from other bots — every compromised computer is capable of providing data to the other bots,
- The new trend in the development of botnet is to provide them the capability to be “independent” from control servers, surviving and becoming anonymous for long periods, infecting many machines.



Tracking systems such as ZeusTracker are not able to track this variant due the impossibility to add the complete list of components of a P2P network instead only the IP addresses of C&C servers.

TOR Botnet

- In September 2012 the German security firm **G Data Software** detected Skynet, a TOR-based botnet
- Dannis Brown at DefCon18 [4] has shown, for the first time, a possible implementation of a C&C channel over Tor to provide C&C server anonymity
- all critical communications of Skynet to its C&C servers are tunneled through a Tor SOCKS proxy running locally on compromised computers.



Even though Tor provides such an anonymity service, it also exposes the botnet activity due to recognizable patterns

•The server is anonymous and thus cannot point to the botnet owners' identity.
•The server cannot be taken down easily.
•The traffic is encrypted by Tor, so it can't be blocked by Intrusion Detection Systems.
•Tor traffic usually cannot be blocked altogether, because there are also legit use cases for Tor.
•The bot creator does not necessarily have to generate a custom protocol, but can use the known and reliable IRC protocol.

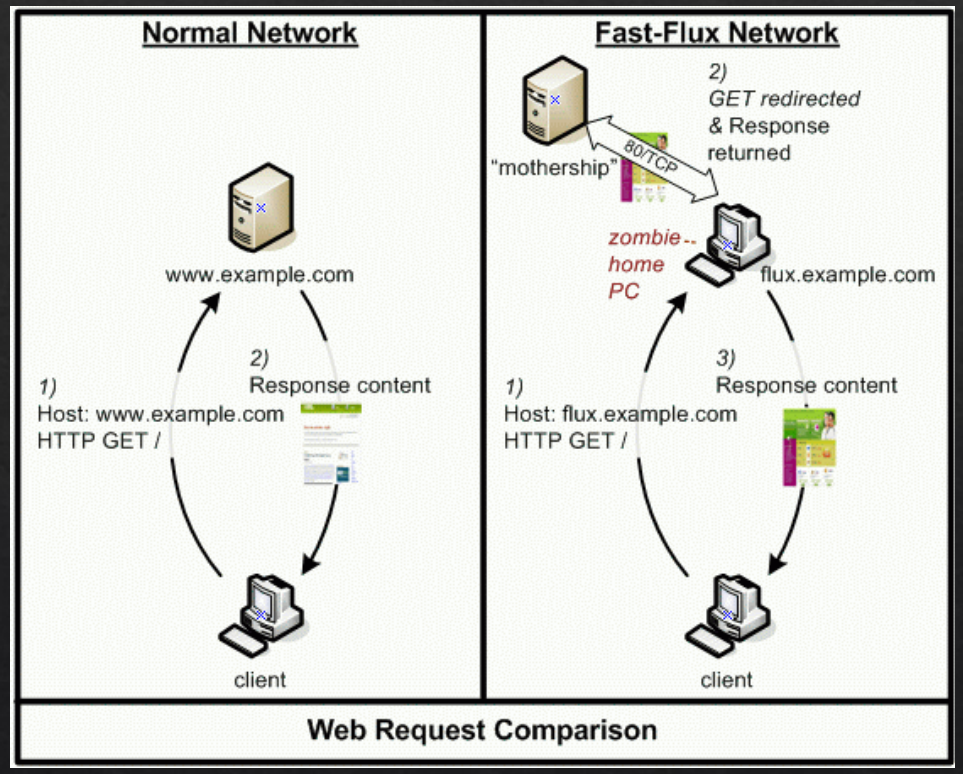
- Botnet Architecture evolved over time
- Advanced topology is more resilient to shutdown, enumeration or discovery

Approaches to detection & measurement of botnets

- Passive Techniques
  - Packet Inspection
  - Analysis of Flow Records
  - DNS-based Approaches
  - Analysis of Spam
  - Analysis of Log Files
  - Honeypots
  - Evaluation of AV Feedback
- Active Techniques
  - **Sinkholing**
  - **Infiltration**
  - DNS Cache Snooping
  - **Tracking of Fast-Flux Networks**
  - IRC-based detection & monitoring
  - Enumeration of Peer-to-Peer Networks

Tracking of Fast-Flux Networks

- **Fast flux** is a DNS technique used by botnets to hide phishing and malware delivery sites behind an ever-changing network of compromised hosts acting as proxies.
- The goal of fast-flux is for a fully qualified domain name (such as www.example.com) to have multiple (hundreds or even thousands) IP addresses assigned to it.
- These IP addresses are swapped in and out of flux with extreme frequency, using a combination of round-robin IP addresses and a very short Time-To-Live (TTL) for any given particular DNS Resource Record (RR)
- Fast-flux “motherships” are the controlling element behind fast-flux service networks, and are similar to the command and control (C&C) systems found in conventional botnet



Sinkholing

- Sinkholing is a technique that researchers use to redirect the identification of the malicious command-and-control (C&C) server to their own analysis server.
- Some of the larger botnets have been made unusable by TLD sinkholes that span the entire Internet

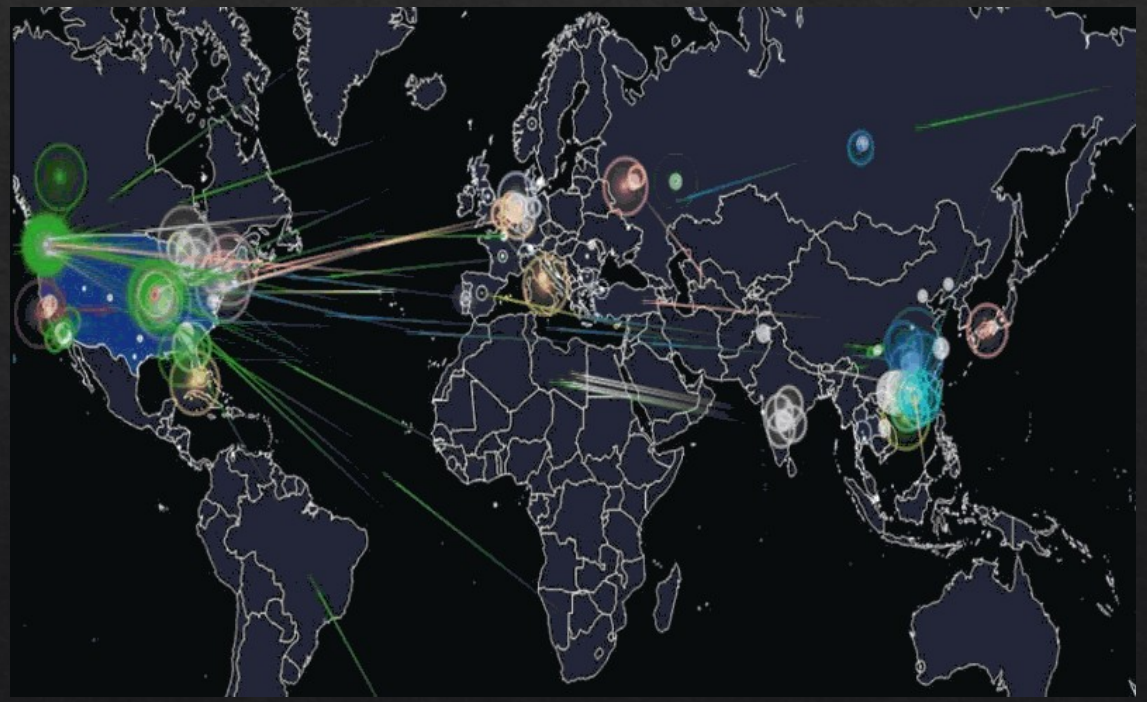
During 2012 Microsoft was running the sinkhole for the 70,000 malicious subdomains, it blocked more than 609 million connections from more than 7,650,000 unique IP addresses to the bad 3322.org subdomains. Legitimate subdomains were provided DNS service.

Mirai Botnet

- IoT (non conventional) Botnet, mostly CCTV camera
- Mirai propagates by bruteforcing telnet server with a list of 62 default password
- Mirai is capable of launching multiple types of DDoS attacks, including SYN-flooding, UDP flooding, Valve Source Engine (VSE) query-flooding, GRE-flooding, ACK-flooding (including a variant intended to defeat intelligent DDoS mitigation systems, or IDMSes), pseudo-random DNS label-prepending attacks (also known as DNS ‘Water Torture’ attacks), HTTP GET attacks, HTTP POST attacks, and HTTP HEAD attacks
- In early October, Mirai’s developer **released the malware’s source code**
- The researchers also note that the botnet’s command and control (C&C) code is coded in Go, while the bots are coded in C
- Mirai was found to include a list of IPs that bots should avoid scanning: the US Postal Service, the Department of Defense, the Internet Assigned Numbers Authority (IANA) and IP ranges belonging to Hewlett-Packard and General Electric

Mirai Botnet

- On October 21, Dyn received a global **distributed denial of service** (DDoS) attack on its DNS infrastructure on the east coast starting at around 7:10 a.m. ET (11:10 UTC).



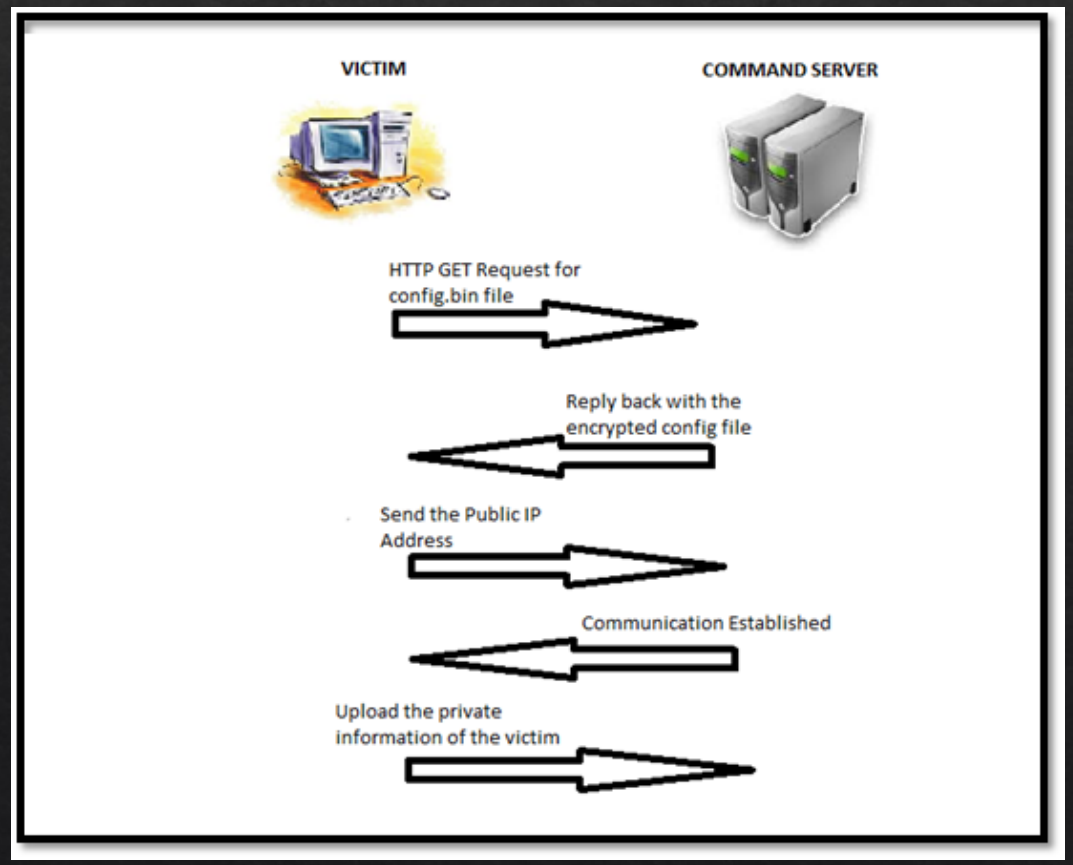
Zeus: A case study

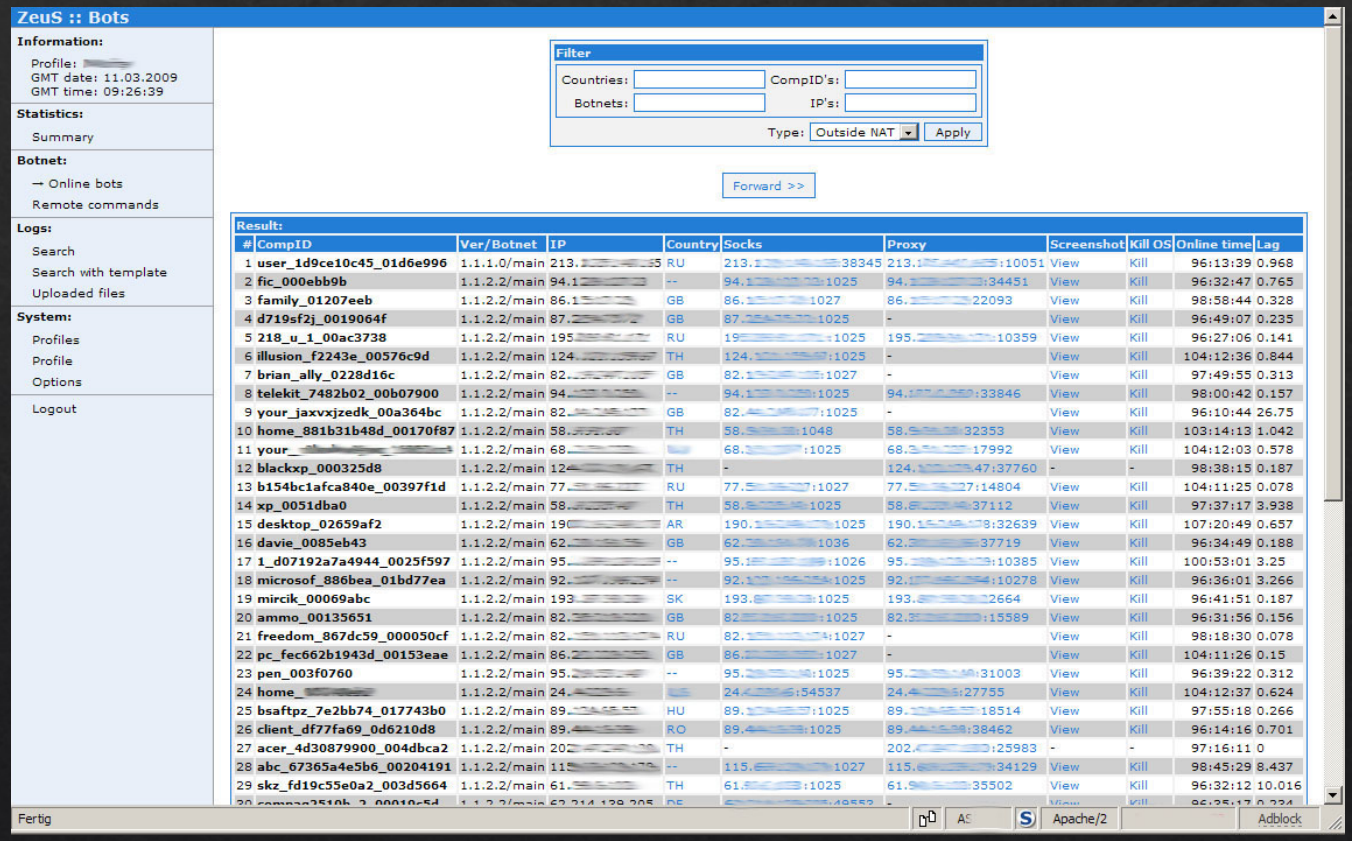
- In October 2010 the US FBI announced that hackers in Eastern Europe had managed to infect computers around the world using Zeus.[8] The virus was distributed in an e-mail, and when targeted individuals at businesses and municipalities opened the e-mail
- In 2013 Hamza Bendelladj, known as Bx1 online, was arrested in Thailand [11] and deported to Atlanta, Georgia, USA.

Zeus: A case study

- Zeus, also known as ZBot/WSNPoem, is famous for stealing banking information by using man in the browser keystroke logging and form grabbing
- The ZBot functions by downloading an encrypted configuration file and storing it in the location marked above. The Trojan opens up a backdoor connection for downloading/uploading from the command and control server, such as newer versions of configuration file, pushing the stolen data to a specific location as in the configuration file, etc
- any form of credentials or banking information is intercepted by it and uploaded to the secret location. It uses advanced mechanisms for form grabbing that sets up web pages and the user unknowingly enters his financial information,

Zeus: A case study





Zeus Botnet

Some of the features that this botnet displays are:

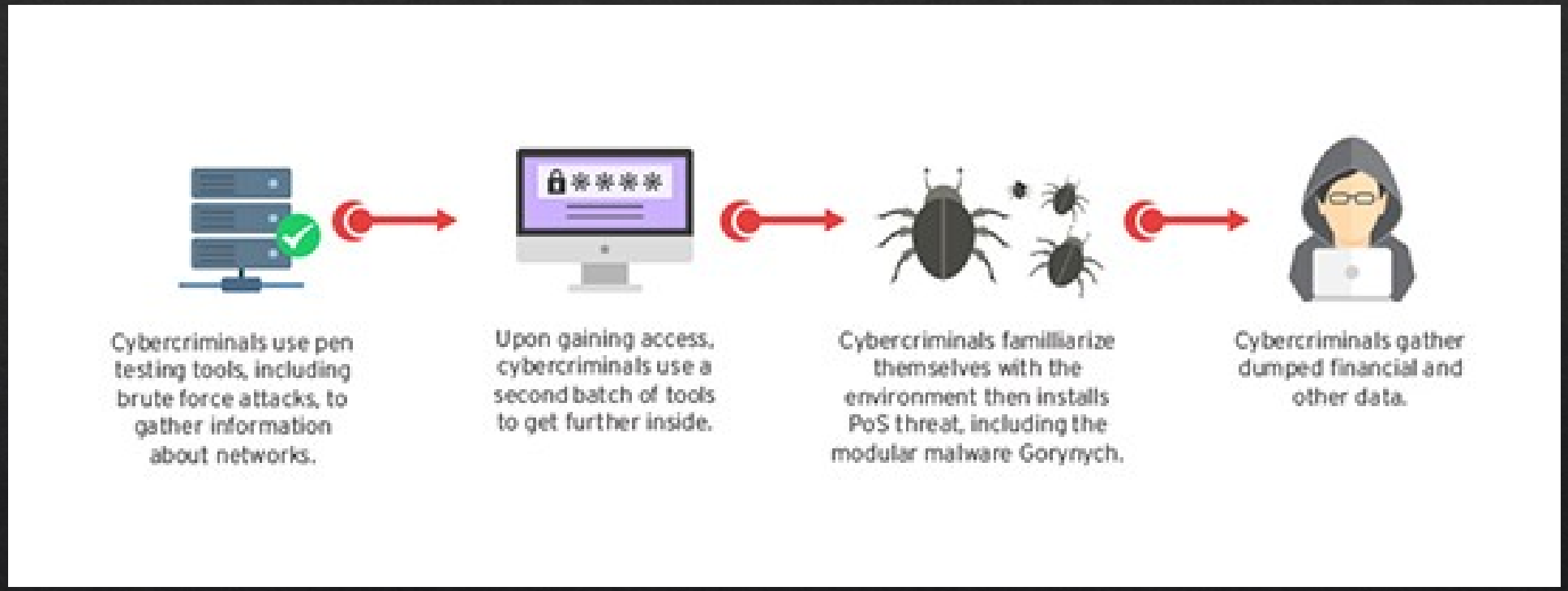
- Captures credentials over HTTP, HTTPS, FTP, POP3
- Steals client-side X.509 public key infrastructure certificates
- Has an integrated SOCKS proxy
- Steals/deletes HTTP and flash cookies
- Captures screenshots and scrapes HTML from target sites
- Modifies the local hosts file
- Groups the infected user systems into different botnets to distribute command and control
- Has search capabilities which may be used through a web form
- The configuration file is encrypted
- Has a major function to kill the operating system
- Contacts command and control server for additional tasks to perform
- Has a unique bot identification string
- Sends a lot of information to C&C server, such as the version of the bot, operating system, local time, geographic locations, etc.

Black Atlas

botnet to deliver PoS malware

- Black Atlas was targeted at small and medium-sized businesses with the goal of breaking into their networks and ultimately stealing data from point-of-sale terminals
- It includes brute-force tools for guessing passwords, SMTP (email) scanners and remote desktop scanners, among other tools.
- It can deliver advanced malware such as BlackPOS, which is best known for the pivotal role it played in the theft of approximately 40 million payment cards from retailer Target in late 2013.
- Finally, it can exfiltrate the data it captures, using HTTP POST.
- Black Atlas is a good example of how the most effective botnets can piece together different tools and techniques to take advantage of weak network defenses

Black Atlas: A case study



Black Atlas: A case study

- Black Atlas operators used the modular botnet **Gorynych** or **Diamond Fox** in some installations. Gorynich was used to download a repurposed BlackPOS malware with RAM scraping functionality and upload all the dumped credit card numbers in memory. As the original BlackPOS used a text file to store pilfered credit card data, Gorynych now grabs that text file and does an HTTP POST to complete the data exfiltration