Miscellaneous: Malware cont'd & start on Bitcoin

CS 161: Computer Security

Prof. Raluca Ada Popa

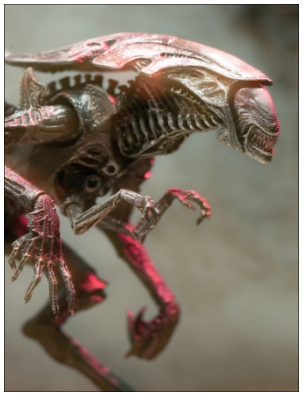
April 19, 2018

Credit: some slides are adapted from previous offerings of this course

Viruses vs. Worms

VIRUS

- Propagates by infecting other programs
- Usually inserted into host code (not a standalone program)



WORM

- Propagates automatically by copying itself to target systems
- A standalone program

Another type of virus: Rootkits

- Rootkit is a ”stealthy” program designed to give access to a machine to an attacker while actively hiding its presence
- Q: How can it hide itself?
  - Create a hidden directory
    - /dev/.lib, /usr/src/.poop and similar
    - Often use invisible characters in directory name
  - Install hacked binaries for system programs such as netstat, ps, ls, du, login

Q: Why does it become hard to detect attacker’s process?
A: Can’t detect attacker’s processes, files or network connections by running standard UNIX commands!

Sony BMG copy protection rootkit scandal (2005)

- Sony BMG published CDs that apparently had copy protection (for DRM).
- They essentially installed a rootkit which limited user's access to the CD.
- It hid processes that started with sys so a user cannot disable them.

A software engineer discovered the rootkit, it turned into a big scandal because it made computers more vulnerable to malware

Q: Why?

A: Malware would choose names starting with sys so it is hidden from antivirus programs

Sony BMG pushed a patch … but that one introduced yet another vulnerability

So they recalled the CDs in the end


IF YOU PLAY ONE ON A PC, IT INVISIBLY INSTALLS STUFF INTO YOUR SYSTEM THAT VIRUS WRITERS CAN USE TO HIDE ALL KINDS OF MALICIOUS CODE.
MAKES YOU FEEL SORRY FOR PEOPLE WHO BOUGHT THE NEW CELINE DION ALBUM.

Detecting Rootkit's Presence

How can we still find a rootkit?

- Sad way to find out
  - Run out of physical disk space because of sniffer logs
  - Logs are invisible because du and ls have been hacked
- Manual confirmation
  - Reinstall clean ps and see what processes are running
- Automatic detection
  - Rootkit does not alter the data structures normally used by netstat, ps, ls, du, ifconfig
  - Host-based intrusion detection can find rootkit files
    - …assuming an updated version of rootkit did not disable the intrusion detection system!

Worms

WORM

- Propagates automatically by copying itself to target systems
- A standalone program

1988 Morris Worm (Redux)

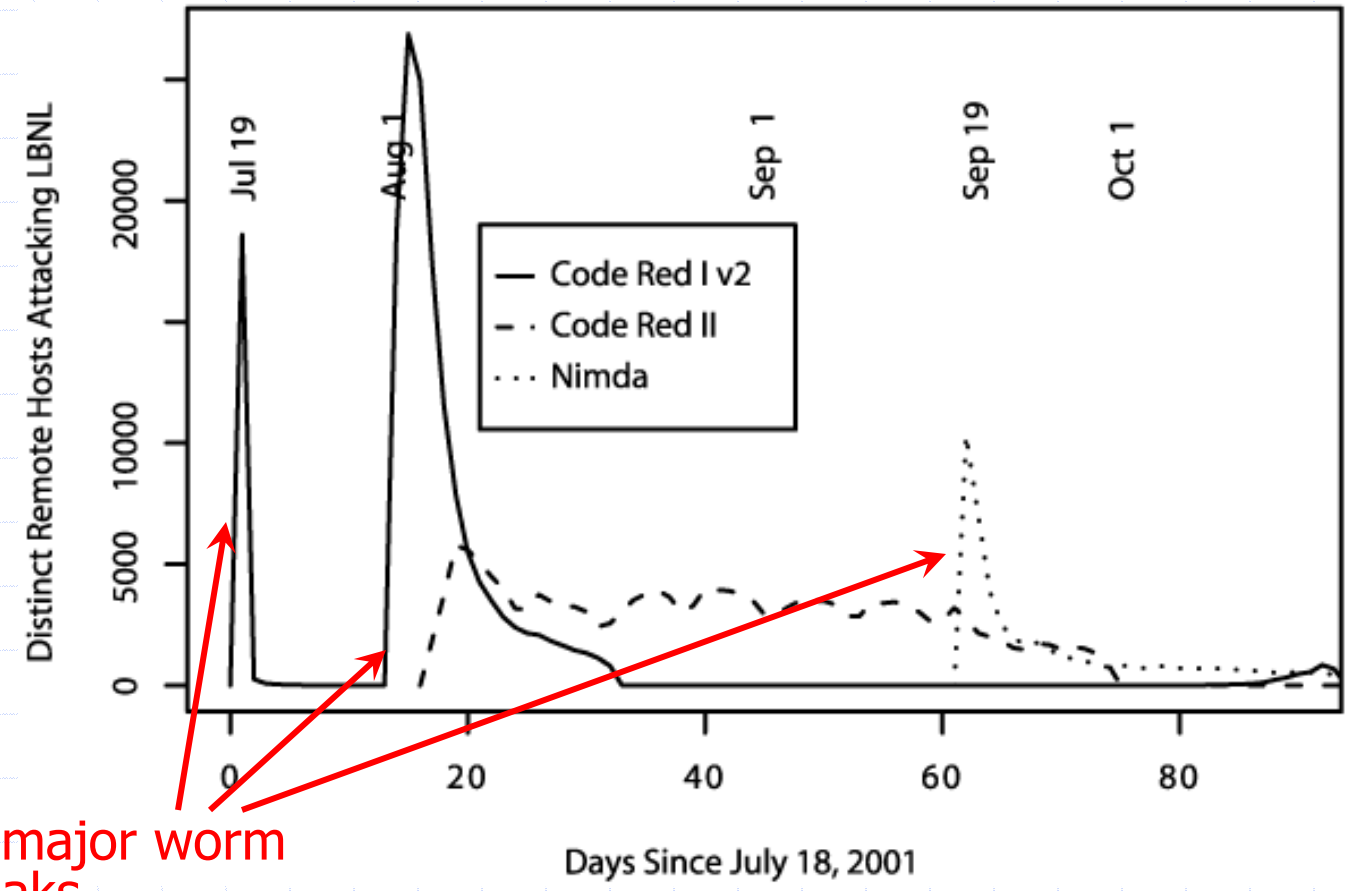
- Robert Morris, grad student, wanting to measure the internet
- No malicious payload, but what went wrong?
  - Bogged down infected machines by uncontrolled spawning
  - Infected 10% of all Internet hosts at the time
- Multiple propagation vectors
  - Remote execution using rsh and cracked passwords
    - Tried to crack passwords using a small dictionary and publicly readable password file; targeted hosts from /etc/hosts.equiv
  - Buffer overflow in fingerd on VAX
    - Standard stack smashing exploit

Dictionary attack

Memory corruption attack

Summer of 2001

["How to 0wn the Internet in Your Spare Time"]



Three major worm outbreaks

Code Red I

- July 13, 2001: First worm of the modern era
- Exploited buffer overflow in Microsoft's Internet Information Server (IIS)
- 1st through 20th of each month: spread
  - Finds new targets by random scan of IP address space
    - Spawns 99 threads to generate addresses and look for IIS
  - Creator forgot to seed the random number generator, and every copy scanned the same set of addresses ☺
- 21st through the end of each month: attack
  - Defaces websites with `"HELLO! Welcome to http://www.worm.com! "`

Code Red II

- August 4, 2001: Same IIS vulnerability, completely different code
  - Known as "Code Red II" because of comment in code
  - Worked only on Windows 2000, crashed NT
- Scanning algorithm prefers nearby addresses
  - Chooses addresses from same class A with probability ½, same class B with probability 3/8, and randomly from the entire Internet with probability 1/8
- Payload: installs root backdoor for unrestricted remote access
- Died by design on October 1, 2001

Nimda

- September 18, 2001: Multi-modal worm using several propagation vectors
  - Exploits same IIS buffer overflow as Code Red I and II
  - Bulk-emails itself as an attachment to email addresses harvested from infected machines
  - Copies itself across open network shares
  - Adds exploit code to Web pages on compromised sites to infect visiting browsers
  - Scans for backdoors left by Code Red II

Signature-Based Defenses Don't Help

Q: why are they not effective when a worm appears?

- Most antivirus filters simply scan attachments for signatures (code fragments) of known viruses
  - Nimda was a brand-new infection with a never-seen-before signature $\Rightarrow$ scanners could not detect it
- Big challenge: detection of zero-day attacks
  - When a worm first appears in the wild, its signature is often not extracted until hours or days later

Slammer Worm

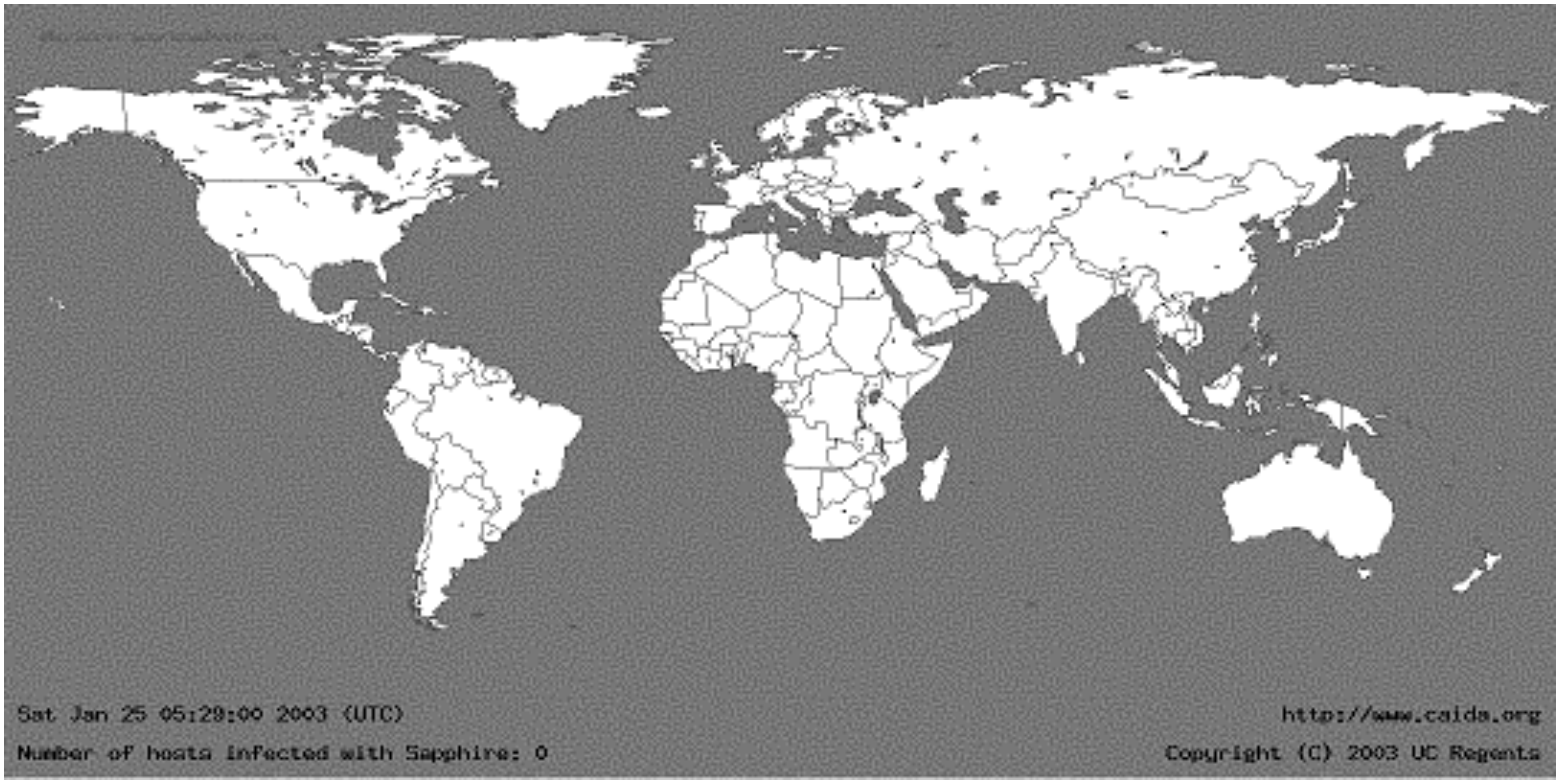
- January 24/25, 2003: UDP worm exploiting buffer overflow in Microsoft's SQL Server (port 1434)
  - Overflow was already known and patched by Microsoft… but not everybody installed the patch
- Entire code fits into a single 404-byte UDP packet
- Classic stack smash combined with random scanning: once control is passed to worm code, it randomly generates IP addresses and sends a copy of itself to port 1434

Slammer Propagation

- Scan rate of 55,000,000 addresses per second
  - Scan rate = the rate at which worm generates IP addresses of potential targets
  - Up to 30,000 single-packet worm copies per second
- Initial infection was doubling in 8.5 seconds (!!)
  - Doubling time of Code Red was 37 minutes
- Worm-generated packets saturated carrying capacity of the Internet in 10 minutes
  - 75,000 SQL servers compromised
  - … in spite of the broken pseudo-random number generator used for IP address generation

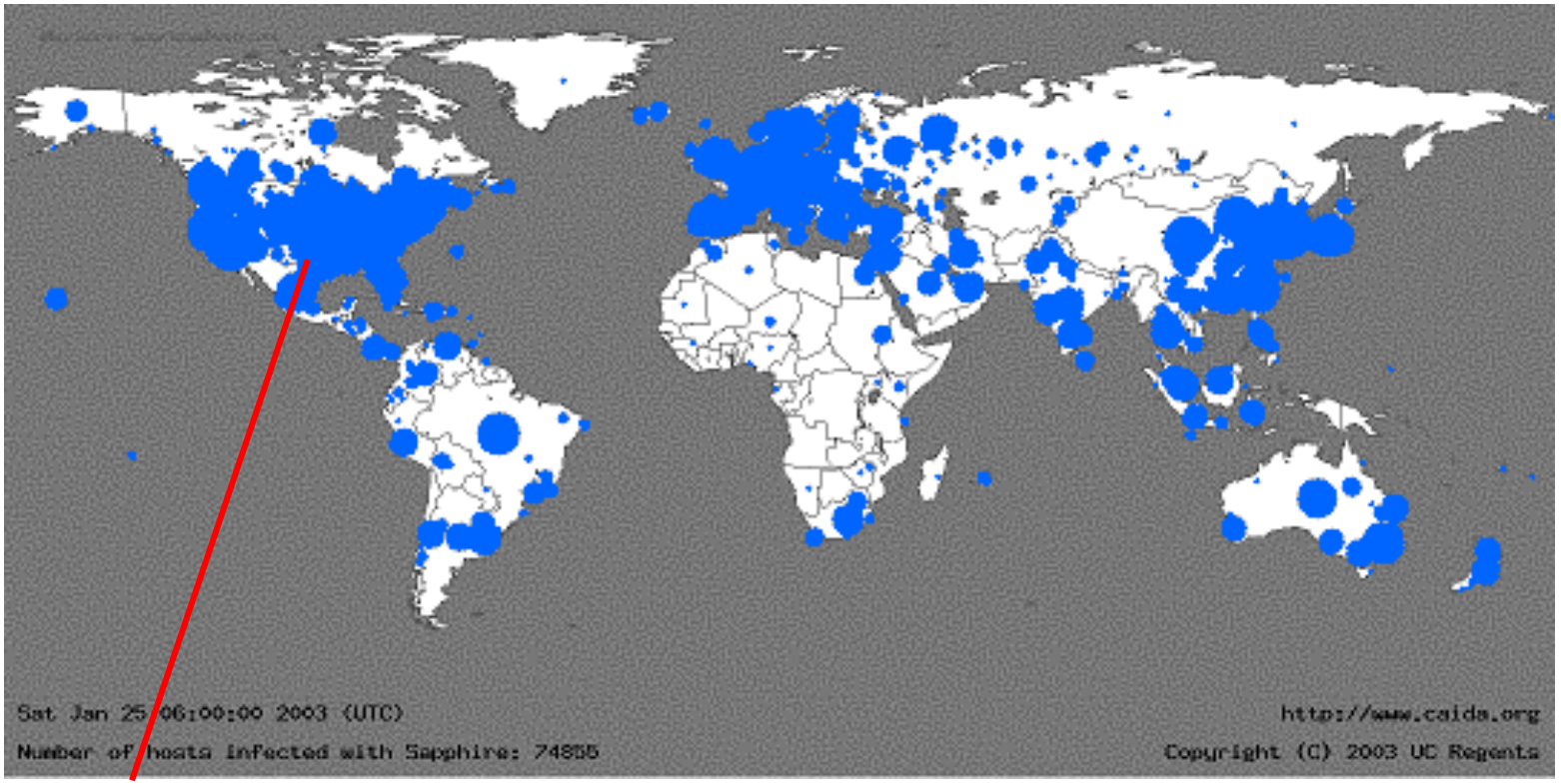
05:29:00 UTC, January 25, 2003

[from Moore et al. "The Spread of the Sapphire/Slammer Worm"]



30 Minutes Later

[from Moore et al. "The Spread of the Sapphire/Slammer Worm"]



Size of circles is **logarithmic** in the number of infected machines

Botnets

Botnets

- A botnet is a network of autonomous programs controlled by a remote attacker and acting on instructions from the attacker
  - Machine owners are not aware they have been compromised
- Used as a platform for various attacks
  - Distributed denial of service
  - Spam and click fraud
  - Launching pad for new exploits/worms

Bot History

- Eggdrop (1993): early IRC bot
- DDoS bots (late 90s): Trin00, TFN, Stacheldracht
- IRC bots (mid-2000s)
  - Active spreading, multiple propagation vectors
  - Include worm and trojan functionality
  - Many mutations and morphs of the same codebase
- Stormbot and Conficker (2007-09)

Life Cycle of an IRC Bot

- Exploit a vulnerability to execute a short program (shellcode) on victim's machine
  - Buffer overflows, email viruses, etc.
- Shellcode downloads and installs the actual bot
- Bot disables firewall and antivirus software
- Bot locates IRC server, connects, joins channel
  - Needs to make a DNS server lookup for the IP address of the IRC server
  - Joins channel of the attacker, attacker sends commands via the IRC channel

Command and Control via IRC

```
(12:59:27pm) -- A9-pcgbdv (A9-pcgbdv@140.134.36.124)
has joined (#owned) Users : 1646

(12:59:27pm) (@Attacker) .ddos.synflood 216.209.82.62

(12:59:27pm) -- A6-bpxufrd (A6-bpxufrd@wp95-
81.introweb.nl) has joined (#owned) Users : 1647

(12:59:27pm) -- A9-nzmpah (A9-nzmpah@140.122.200.221)
has left IRC (Connection reset by peer)

(12:59:28pm) (@Attacker) .scan.enable DCOM

(12:59:28pm) -- A9-tzrkeasv (A9-tzrkeas@220.89.66.93)
has joined (#owned) Users : 1650
```

Detecting Botnet Activity

How can you detect an IRC bot?

- Many bots are controlled via IRC and DNS
  - IRC used to issue commands to zombies
  - DNS used by zombies to find the master, and by the master to find if a zombie has been blacklisted
- IRC/DNS activity is very visible in the network
  - Look for hosts performing scans and for IRC channels with a high percentage of such hosts
  - Look for hosts who ask many DNS queries but receive few queries about themselves
- How can the bot evade such detection?
  - Easily evaded by using encryption and P2P ☹

Rise of Botnets

- 2003: 800-900,000 infected hosts, up to 100K nodes per botnet
- 2006: 5 million distinct bots, but smaller botnets
  - Thousands rather than 100s of thousands per botnet
  - Reasons: evasion, economics, ease of management
  - More bandwidth (1 Mbps and more per host)
- Other reasons than mischief:
  - Spread spam
  - Extort money by threatening/unleashing DoS attacks
  - Political strategy

Storm (2007)

- Spreads via cleverly designed campaigns of spam email messages with catchy subjects
  - First instance: "230 dead as storm batters Europe"
  - Other examples: "Condoleeza Rice has kicked German Chancellor", "Radical Muslim drinking enemies's blood", "Saddam Hussein alive!", "Fidel Castro dead", etc.
- Attachment or URL with malicious payload
  - FullVideo.exe, MoreHere.exe, ReadMore.exe, etc.
  - Also masquerades as flash postcards
- Once opened, installs a trojan (wincom32) and a rootkit, joins the victim to the botnet

Storm Characteristics

[Porras et al.]

- Between 1 and 5 million infected machines
- Obfuscated peer-to-peer control mechanism
  - Not a simple IRC channel
- Obfuscated code, anti-debugging defenses
  - Triggers an infinite loop if detects VMware or Virtual PC
  - Large number of spurious probes (evidence of external analysis) triggers a distributed DoS attack

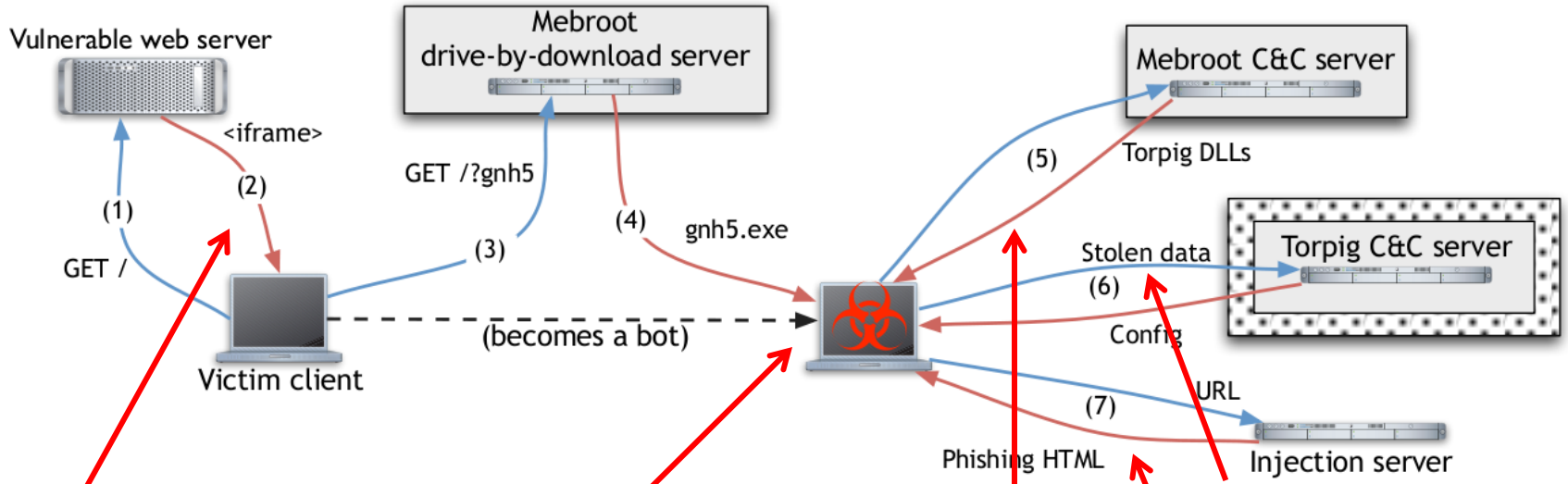
Torpig Study

["Your Botnet Is My Botnet"]

- Security research group at UCSB took over the Torpig botnet for 10 days in 2009
  - Objective: the inside view of a real botnet
- Takeover exploited domain flux
  - Bot copies generate domain names to find their command & control (C&C) server
  - Researchers registered the domain before attackers, impersonated botnet's C&C server

Torpig Architecture (also called Mebroot)

["Your Botnet Is My Botnet"]



Drive-by JavaScript tries to exploit multiple browser vulnerabilities to download Torpig installer

Installer writes Torpig into boot region on hard drive, reboots infected host

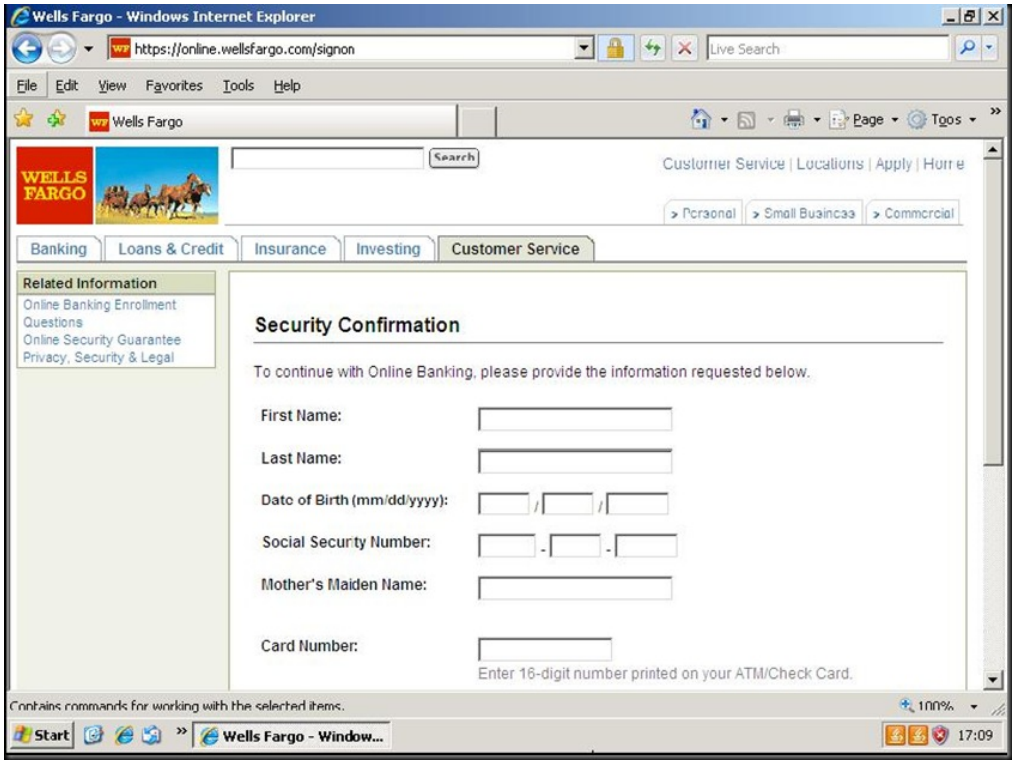
Torpig obtains malicious DLLs from its C&C server, injects them into applications, contacts C&C server every 2 hours over HTTP using custom encryption

DLLs upload stolen data to Torpig C&C server

C&C server acks or instructs bot to perform phishing attacks against specific sites using injected content

Man-in-the-Browser Attack

Victim user runs compromised browser (e.g., user installed malware by mistake) and this browser modifies user requests. E.g., instead of transferring a certain sum, it can change the sum, or instead of encrypting with a certain PK, it encrypts with the PK of the attacker



Target: Financial Institutions

["Your Botnet Is My Botnet"]

- Typical Torpig config file lists approximately 300 domains of financial institutions to be targeted for "man-in-the-browser" phishing attacks
- In 10 days, researchers' C&C server collected 8,310 accounts at 410 institutions
  - Top 5: PayPal (1770), Poste Italiane (765), Capital One (314), E*Trade (304), Chase (217)
- 1660 unique credit and debit card numbers

ZeroAccess Botnet

http://www.symantec.com/connect/blogs/grappling-zeroaccess-botnet

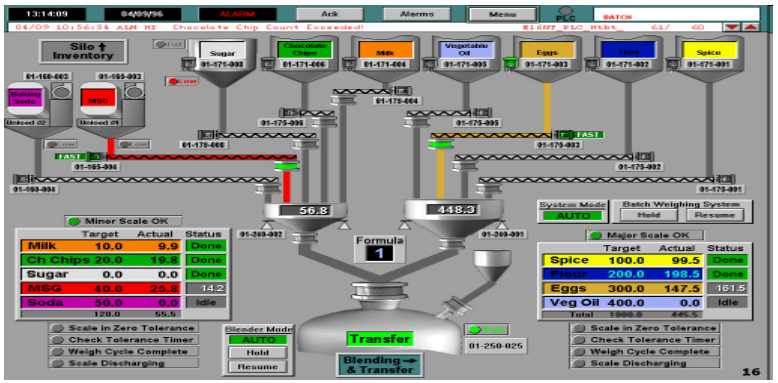
- Peer-to-peer structure, no central C&C server
- 1.9 million infected machines as of August 2013
- Used for click fraud
  - Trojan downloads ads and "clicks" on them to scam per-pay-click affiliate schemes
- Used for bitcoin mining
  - According to Symantec, one compromised machine yields 41 US cents a year…

Stuxnet (2010)

- Complex “Beast”
  - Computer Worm (Spreads on its own)
  - Trojan Horse (Does something it is not supposed to do)
  - Virus (Embeds itself with human interaction)
- Without finding its specific target, it would remain dormant

Its Target: Industrial Control Systems

- Run automated processes on factory floors, power and chemical plants, oil refineries, etc.



Stuxnet Firsts

- First to exploit multiple zero-day vulnerabilities
- First to use stolen signing keys and valid certificates of two companies
- First to target industrial control systems

  … and hide the code from the operator

  … and perform actual sabotage
- First example of true cyber-warfare?

Iranian Nuclear Program

- Sep 2010: "delays"
  - Warm weather blamed
- Oct 2010: "spies" arrested, allegedly attempted to sabotage Iran's nuclear program
- Nov 2010: Iran acknowledges that its nuclear enrichment centrifuges were affected by a worm
  - Foreign minister: "Nothing would cause a delay in Iran's nuclear activities"
  - Intelligence minister: "enemy spy services" responsible

Exploring the Attack Vector

- Two strikingly different attack vectors
- Overpressure Attack
  - Increase centrifuge rotor stress
  - Significantly stronger
  - More stealthy
  - Less documented in literature
- Rotor Speed Attack
  - Increase rotor velocity
  - Overpressure centrifuge is dormant in this attack
  - Independent from previous attack
  - Less concern about detection -> push the envelope

Who created Stuxnet?

- Not known for sure. Ideas?

- Edward Snowden claims that Israel and the United States created the Stuxnet to destroyed nuclear centrifuges in Iran

Who is Behind the Botnets?

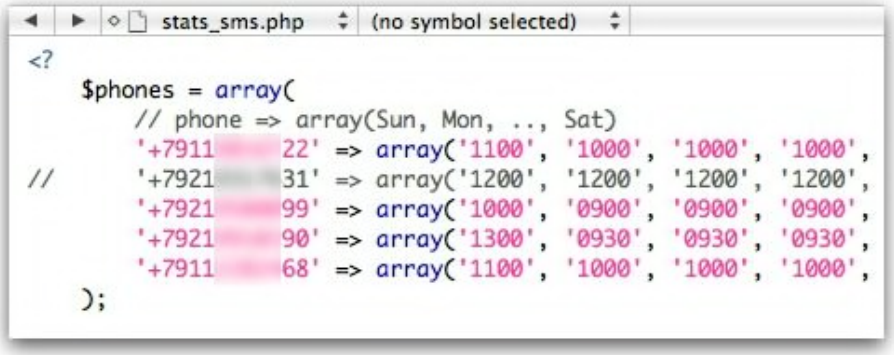
- Case study: Koobface gang

- Responsible for the 2008-09 Facebook worm
  - Messages Facebook friends of infected users, tricks them into visiting a site with a malicious "Flash update"
- Made at least $2 million a year from fake antivirus sales, spam ads, etc.
- De-anonymized by SophosLabs

KoobFace Deanonymization (1)

http://nakedsecurity.sophos.com/koobface/

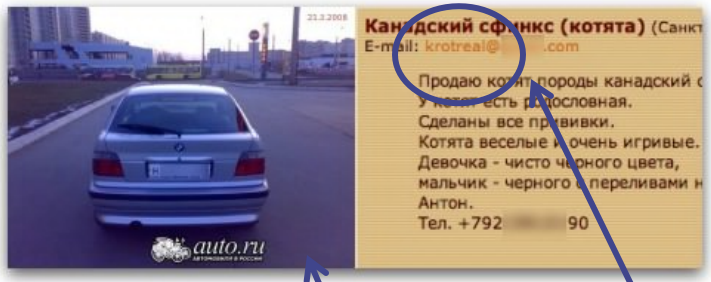
- One of the command-and-control servers had a configuration mistake, any visitor can view all requests, revealing file and directory names
- last.tar.bz2 file contained daily C&C software backup, including a PHP script for sending daily revenue statistics to five Russian mobile numbers



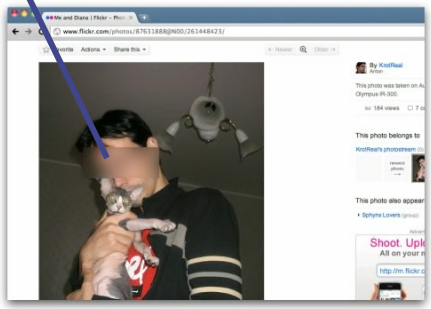
KoobFace Deanonymization (2)

http://nakedsecurity.sophos.com/koobface/

- Search for the phone numbers found Russian online ads for a BMW car and Sphynx kittens



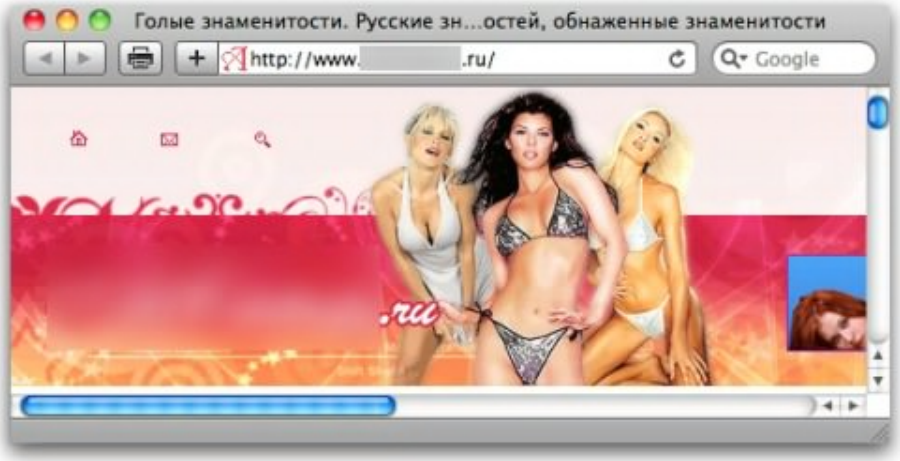
- Search for username "krotreal" found profiles in various social sites – with photos!



KoobFace Deanonymization (3)

http://nakedsecurity.sophos.com/koobface/

- One of the social-network profiles references an adult Russian website belonging to "Krotreal"



- "Whois" for the website lists full name of the owner, with a St. Petersburg phone number and another email (Krotreal@mobsoft.com)

KoobFace Deanonymization (4)

http://nakedsecurity.sophos.com/koobface/

- Krotreal profile on vkontakte.ru ("Russian Facebook") is restricted…
- … but he posted links to photos on Twitter, thus making photos publicly available

- Reveals social relations

KoobFace Deanonymization (5)

http://nakedsecurity.sophos.com/koobface/

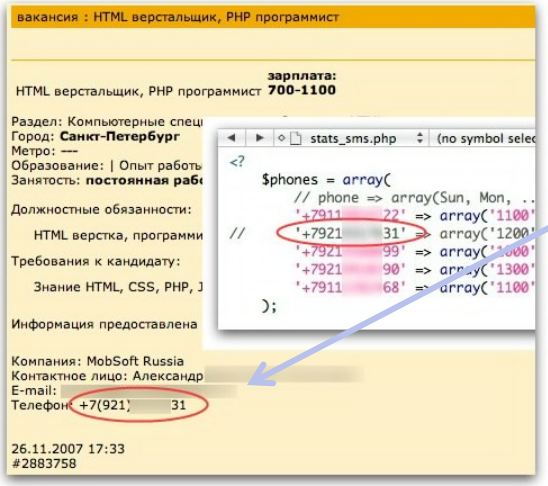
Hosted on the Koobface "mothership" server

- Czech government maintains an online portal providing easy access to company details
  - Includes registered address, shareholders, owners, their dates of birth and passport ID numbers

KoobFace Deanonymization (6)

http://nakedsecurity.sophos.com/koobface/

- Search for MobSoft on Russian Federal Tax Server reveals nothing, but search for МобСофт reveals owner's name and also a job ad:

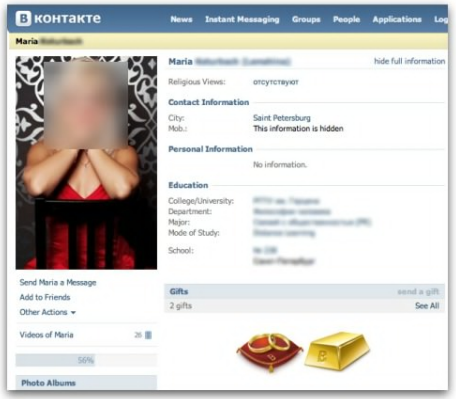


Same phone number as in the statistics script on the Koobface C&C server

- Contact person found on social sites

KoobFace Deanonymization (7)

http://nakedsecurity.sophos.com/koobface/



- The co-owner of one of the Mobsoft entities did not restrict her social profile
- Reveals faces, usernames, relationships between gang members
  - Hanging out, holidays in Monte Carlo, Bali, Turkey

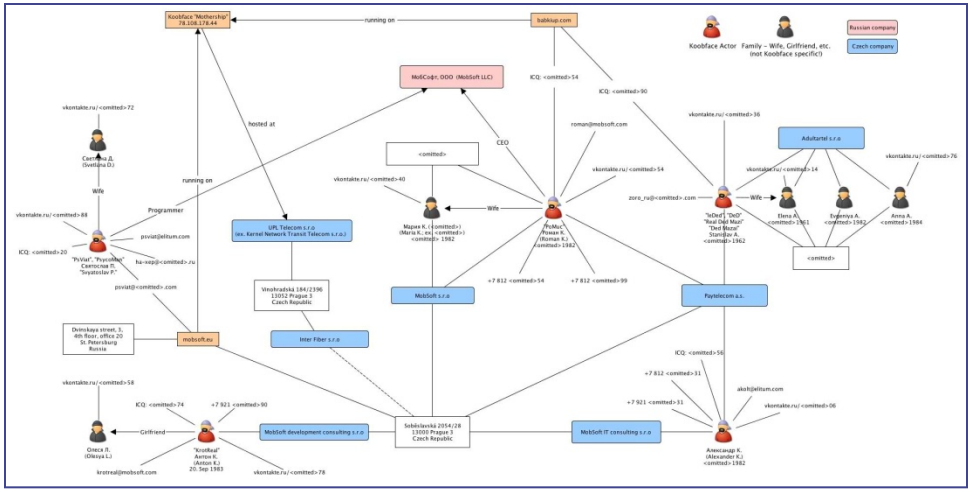
One photo shows Svyatoslav P. participating in a porn webmaster convention in Cyprus

"FUBAR webmaster" website has archive photo sets from various porn industry events

Username on the badge!

The Koobface Gang

- Антон Коротченко
  - “KrotReal”
- Станислав Авдейко
  - “LeDed”
- Святослав Полищук
  - “PsViat”, “PsycoMan”
- Роман Котурбач
  - “PoMuc”
- Александр Колтышев
  - “Floppy”



Conclusions

- Viruses infect other programs, worms spread alone
- Rootkits are stealthy and try to hide their existence
- Botnets infect many machines and listen for commands from a command and control server. Botnets can be very complex
- Motivation for malware creators can be financial, political, or personal

Let’s start thinking blockchain: Proof of work, Hash chaining

Math Puzzle – Proof of Work

- **Problem.** To prove to Bob I'm not a spammer, Bob wants me to do 10 seconds of computation before I can send him an email. How can I prove to Bob that I wasted 10 seconds of CPU time, in a way that he can verify in milliseconds?

Math Puzzle – Proof of Work

- **Problem.** To prove to Bob I'm not a spammer, Bob wants me to do 10 seconds of computation now before I can send him an email. How can I prove to Bob that I wasted 10 seconds of CPU time, in a way that he can verify in milliseconds?

- Hint: Computing 1 billion SHA256 hashes might take 10 seconds.

Solution 1

- I choose a random value r.
- I compute a billion hashes on r: h(h…(h(r))) and give the result to Bob

- What is the problem?
- Bob needs to do a lot of work to verify.

Solution 2

- I choose many random r-s until h(r) has the first 33 bits being 0
- That would take about 10 seconds
- Bob verifies with one hash

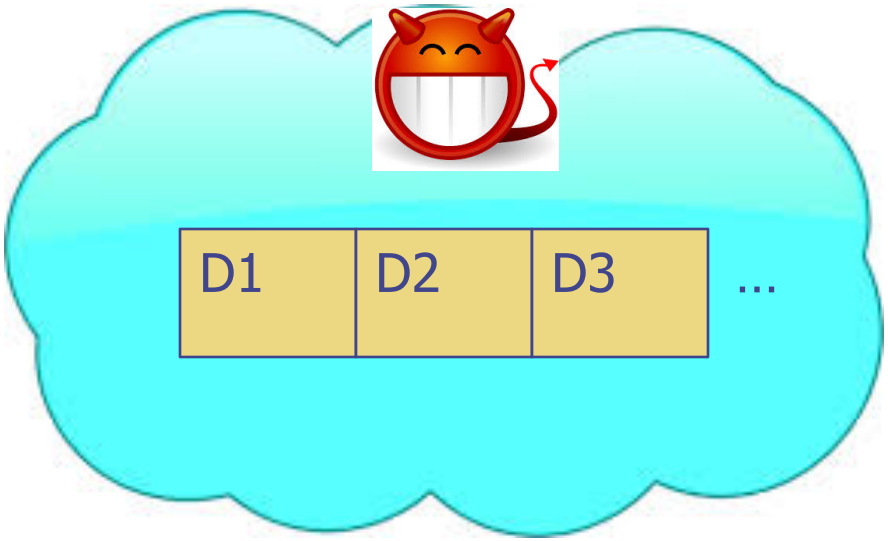
- What is the problem?
- Maybe I had this precomputed already. Maybe someone else found such a hash. How does Bob know **I** did this work **now**?

Solution 3

- Bob provides a random challenge r
- I compute: find x such that H(r,x) starts with 33 0 bits
  - This will take me 2^33 hash computations, on average
  - Geometric: coin flip, with 1 / 2^33 chance of heads
- Bob verifies by: checking that H(r,x) starts with 33 0 bits

This is the proof of work used in Bitcoin

Crypto puzzle: Tamper-evident logging

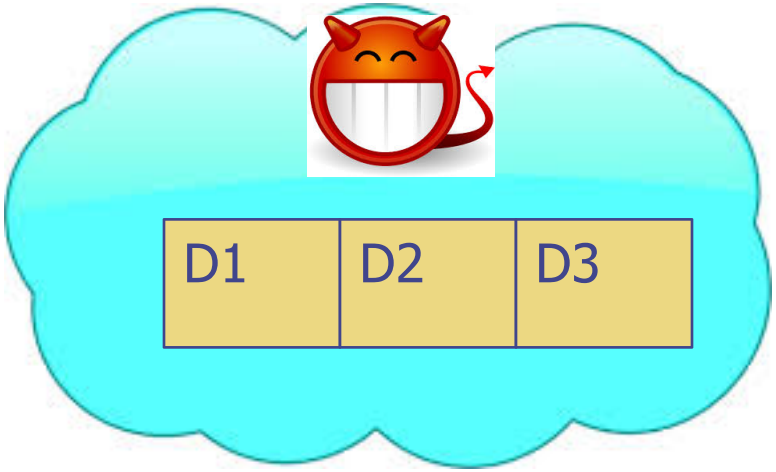


- Alice wants to store a log of data D1, D2, … Dn, … on a cloud service that could be compromised. Say each day a new data records gets added
- Later if she fetches some records, she should be able to verify they were not corrupted.
- She wants to store only one piece of data on her machine.

What can she do?

Solution 1: hash all files

Every day when Alice adds file Di, she recalculates hash(D1, D2, …, Di) and stores this hash.



Problems?

- She needs to calculate the hash over all files
- When she fetches some files and wants to check their integrity, she needs to download them all

Solution 2: **hash chain**

On day i, Alice needs to add data item Di, and she already has hash h_{i-1} from days 1…i-1. She computes h_i = hash(h_{i-1}, Di). This is a hash chain because is h_i calculated based on h_{i-1} which is calculated based h_{i-2}

Q: If Alice wants to fetch the last k data items, how does she check them?

A: Trust the server with h_{i-k} hash received data items from server and see if it matches h_i check them?

Q: The cloud cannot switch any item in the chain or truncate the chain. Why?

A: Hash is collision resistant