<table>
<thead>
<tr>
<th>Addepar</th>
<th>Facebook</th>
<th>LinkedIn</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apple</td>
<td>Freshbooks</td>
<td>Microsoft</td>
</tr>
<tr>
<td>Buffer</td>
<td>Github</td>
<td>Netflix</td>
</tr>
<tr>
<td>Checkmate</td>
<td>Google</td>
<td>Pagerduty</td>
</tr>
<tr>
<td>Dollar Shave Club</td>
<td>Heroku</td>
<td>Pivotshare</td>
</tr>
<tr>
<td>Ericsson</td>
<td>Intercom</td>
<td>Practice Fusion</td>
</tr>
<tr>
<td></td>
<td>Iora Health</td>
<td>Thoughtbot</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ticketfly</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Travis-CI</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tumblr</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Twitch</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Yahoo</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Zenefits</td>
</tr>
</tbody>
</table>
We have a **BIG** problem

- Features & Deadlines vs. Security
- Web Developers have fallen behind
- Attacks are escalating in severity
- Barriers to staging an attack are lower than ever
Our Strawman

http://strawbank.com

Strawbank

mike@mike.works

ACCOUNTS TRANSFERS

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Checking</td>
<td>****7890</td>
<td>$10,000</td>
</tr>
<tr>
<td>Savings</td>
<td>****1234</td>
<td>$8,000</td>
</tr>
</tbody>
</table>

- Cookie-based "session" authentication
- List of accounts
- Ability to lookup other accounts & transfer funds
The Hacks

**NETWORK**
- Man in the middle
- HTTPS downgrading via SSLStrip

**CLIENT SIDE ATTACKS**
- XSS
- CSRF
- Clickjacking
Man in the middle

HTTP

Starbucks WiFi

HTTP

Web Security

MICHAEL L NORTH
Man in the middle
Public WiFi: Trusted forever by default

<table>
<thead>
<tr>
<th>Network Name</th>
<th>Security</th>
</tr>
</thead>
<tbody>
<tr>
<td>TWM WiFi</td>
<td>None</td>
</tr>
<tr>
<td>EliteWiFi</td>
<td>None</td>
</tr>
<tr>
<td>CCOosterdok</td>
<td>None</td>
</tr>
<tr>
<td>Internet</td>
<td>None</td>
</tr>
<tr>
<td>MSP Free WiFi</td>
<td>None</td>
</tr>
<tr>
<td>SouthPointWiFi</td>
<td>None</td>
</tr>
</tbody>
</table>

Drag networks into the order you prefer.

Remember networks this computer has joined
Man in the middle

WiFi Devices broadcast what they're looking for.
Man in the middle
Router as DNS
DNS Hijacking
Man in the middle

ATTACK

Airport Free WiFi

Starbucks WiFi
Man in the middle

WiFi Pineapple
- Linux
- 2x Wifi Cards
- High gain antennas
- "App store"

2000mw WiFi
9dB antenna

Web Security
MICHAEL L NORTH
Man in the middle
Man in the middle

Ominous Box for WiFi Pineapple

$5.00 $10.00
Let's say you've locked down WiFi
Femtocell

ATTACK

WEB SECURITY

MICHAE L L NORTH

MIKEWORKS
Man in the middle

Strawbank

- TLS not SSL
- Private key needed to read or alter request/response
- Getting a cert requires "Domain Validation"

https://strawbank.com

mike@mike.works

<table>
<thead>
<tr>
<th>ACCOUNTS</th>
<th>TRANSFERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Checking</td>
<td>****7890</td>
</tr>
<tr>
<td>Savings</td>
<td>****1234</td>
</tr>
</tbody>
</table>
~56% of the web uses HTTPS

% of page loads over HTTPS

Time
This is probably not the site you are looking for!

You attempted to reach **app.cupsapp.com**, but instead you actually reached a server identifying itself as **cupstelaviv.com**. This may be caused by a misconfiguration on the server or by something more serious. An attacker on your network could be trying to get you to visit a fake (and potentially harmful) version of **app.cupsapp.com**.

You should not proceed, especially if you have never seen this warning before for this site.

Proceed anyway  Back to safety

Help me understand
"...at least 44 percent of the top 382,860 SSL-enabled websites had certificates that would trigger warnings" [1]

After extensive data-driven improvement to Chrome warning messages, 42% of users ignore them instead of over 70% [2]

Over 50% of users don't understand eavesdropping vs. malware risk factors [3]

SSLStrip

Begins with downgrade to HTTP

Strawbank
SSLStrip

Client continues with HTTP, Server is unaware
HTTP Strict Transport Security

Do not allow plain HTTP

Strict-Transport-Security: max-age=31536000; includeSubDomains

• Failure to include subdomains permits a broad range of cookie-related attacks

• There's still the issue of the first request
HSTS Preload

Enter a domain for the HSTS preload list:

example.com

Check status and eligibility
HSTS WARNING

Your connection is not private

Attacks might be trying to steal your information from paypal.com (for example, passwords, messages, or credit cards).

Reload

paypal.com normally uses encryption to protect your information. When Chrome tried to connect to paypal.com this time, the website sent back unusual and incorrect credentials. Either an attacker is trying to pretend to be paypal.com, or a Wi-Fi sign-in screen has interrupted the connection. Your information is still secure because Chrome stopped the connection before any data was exchanged.

You cannot visit paypal.com right now because the website uses HSTS. Network errors and attacks are usually temporary, so this page will probably work later.
Treat Certificates With Care!

Webpack & Preact-CLI Vulnerability
Some users susceptible to undetectable man-in-the-middle attacks over HTTPS on public WiFi

🔍 How it was discovered
After live streaming Addy Osmani’s great Google I/O 2017 talk on advances in Progressive Web App tooling and technologies, I was very excited to take Preact-CLI for a spin.
Treat Certificates With Care!

**WARNING**

This certificate is marked as not trusted for this account.
Treat Certificates With Care!

Firefox and Chrome disable pin validation for pinned hosts whose validated certificate chain terminates at a user-defined trust anchor (rather than a built-in trust anchor). This means that for users who imported custom root certificates all pinning violations are ignored.
The Hacks

**NETWORK**
- Man in the middle
- HTTPS downgrading via SSLStrip

**CLIENT SIDE ATTACKS**
- XSS
- CSRF
- Clickjacking
Cross-Site Scripting (XSS)

<select>
  <option value="1">
    Mike's Checking
  </option>
  <option value="2">
    Lisa's Savings
  </option>
  <option value="3">
    Elliot's Checking
  </option>
</select>

<script src="https://...totally-fine.js"></script>
Cross Site Scripting (XSS)
Cross-Site Scripting (XSS)

- Escape all user input
- Use a view layer that has thorough built-in XSS protection
- Don't forget about styles too!
  `<img src="javascript:alert('XSS!')"/>`
Cross-Site Scripting (XSS)

- How confident are you in the XSS protection of your OSS libraries?
- How carefully do people scrutinize browser plugins?
- If XSS happens, what's your exposure?
Embedded Malware
Embedded Malware

Image file

Image file

Malware config file

DOWNLOADS NORTON ANTI-VIRUS

FILE CONTAINS A VIRUS

WEB SECURITY  MICHAEL L NORTH
Embedded Malware

*Never* trust user-generated content

- Optimize all images (nearly always drops non-visual content)
- Avoid spreading "raw" attachments
- Limit file types for uploads
- Don't permit arbitrary HTML input
- Whitelist content that can be embedded
Cross-Site Request Forgery

```
<form name="badform" method="post" action="https://strawbank.com/api/transfer">
  <input type="hidden" name="destination" value="2" />
  <input type="hidden" name="amount" value="8500" />
</form>
<script type="text/javascript">document.badform.submit();</script>
```
Cross-Site Request Forgery

• Exclusively targets state-changing requests

• Exploits browser "automatically" sending credentials

• A good reason to conform to RESTful HTTP verbs

\[\text{POST requests are also susceptible}\]

\[
<\text{img src="https://strawbank.com/api/transfer?amount=8500&destination=12345">}
\]
Cross-Site Request Forgery

• Only Basic or cookie authentication schemes are vulnerable
  • Exception: "Client side cookie"
• CSRF Token - non-predictable and per-request
• Ensure CORS headers are appropriately restrictive
Clickjacking

https://strawbank.com?amount=8500&dest=185...

Strawbank

mike@mike.works

ACCOUNTS

TRANSFERS

From Acct: Mike's Checking
To Acct: Lisa's Savings
Amount: $8500

Transfer Funds

https://strawbank.com-securebank.cc?tok=108...

Strawbank

You're approved!

StrawCard

You're approved!

Proceed

https://strawbank.com-securebank.cc?tok=108...
Clickjacking

X-Frame-Options: DENY
X-Frame-Options: SAMEORIGIN
X-Frame-Options: ALLOW-FROM https://strawbank.com/
The Hacks

**NETWORK**
- Man in the middle
- HTTPS downgrading via SSLStrip

**CLIENT SIDE ATTACKS**
- XSS
- CSRF
- Clickjacking
But Wait ... There's more.

- SQL Injection
- Timing attacks
- Resource depletion attacks
- Session hijacking
- Execution after redirect
- Log Injection attacks
- Content Security Policy (CSP)
- Cache Poisoning
- Subresource Integrity (SRI)
- Sandboxing untrusted content
- Preventing attack escalation
- Encryption at rest: best practices
Thanks!

Want to know more? Ask me about a security workshop.