Weird New Tricks for Browser Fingerprinting

yan (@bcrypt)
ToorCon 2015

real pic of me

HTTPS Everywhere

Privacy Badger

ALPHA

Install EFF's New Tool to Block
Spying Ads & Invisible Trackers

W3C

also work on these things

Now the Real Fight Against TPP Begins
ICANN'T EVEN
DOMAIN REGISTRARS MUST ASK ICANN'S PERMISSION TO PROTECT YOUR PRIVACY
JOIN THE TOR // CHALLENGE
NSA
TAKE ACTION NOW
Oppose NSA Mass Spying!
LAUNCHING IN 2015:
Let's Encrypt
A CERTIFICATE AUTHORITY TO ENCRYPT THE ENTIRE WEB
STOP CISA

EFF staff photo, 2015

Come to the dark side...
BETTER TRACKING METHODS
...We have ~~cookies~~

Tracking web users is all the rage

- Show ads!
- Inject QUANTUM malware
- Cybercatch cybercriminals
- Gather website analytics
- Detect fraud / droidnets
- Enforce paywalls
- etc.

A long time ago in a galaxy far, far away . . .

Obi-Wan tracked Luke using:

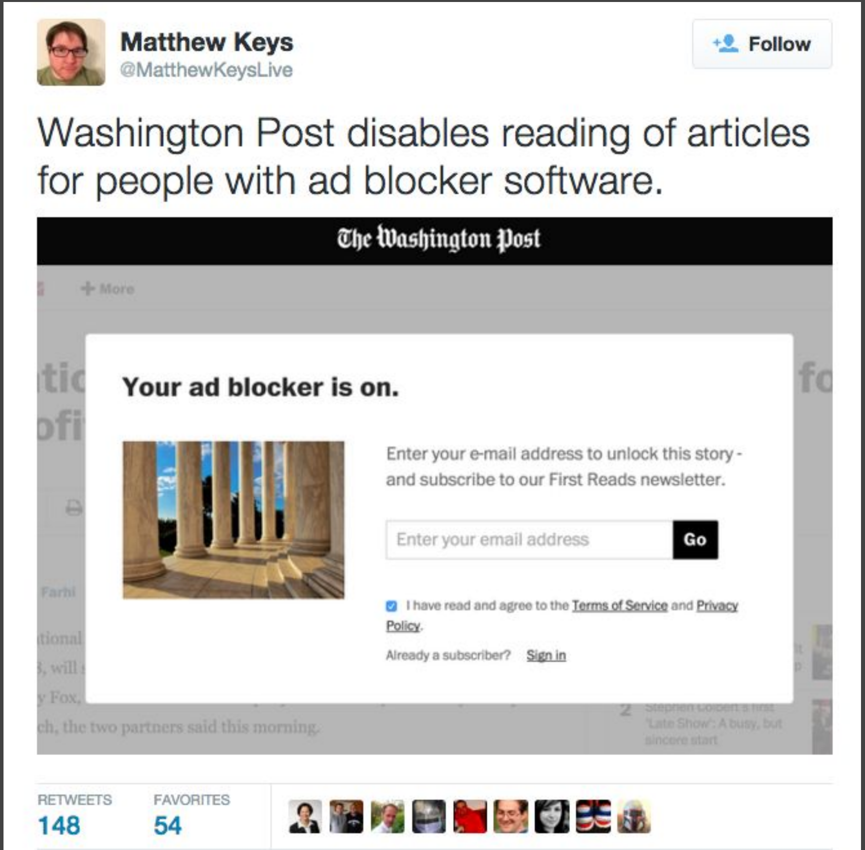
- cookies
- passive fingerprinting* (IP address, locales, user-agent, OS, etc.)
- sweet Jedi mind tricks

* In this presentation, fingerprinting == any non-cookie web tracking method.

THE ADBLOCKERS* STRIKE BACK

* In this presentation, adblocker == any tool that blocks web tracking (including non-advertising)

THE PHANTOM ADBLOCKER BLOCKERS


Matthew Keys
@MatthewKeysLive

Follow

Washington Post disables reading of articles for people with ad blocker software.

The Washington Post

Your ad blocker is on.

Enter your e-mail address to unlock this story - and subscribe to our First Reads newsletter.

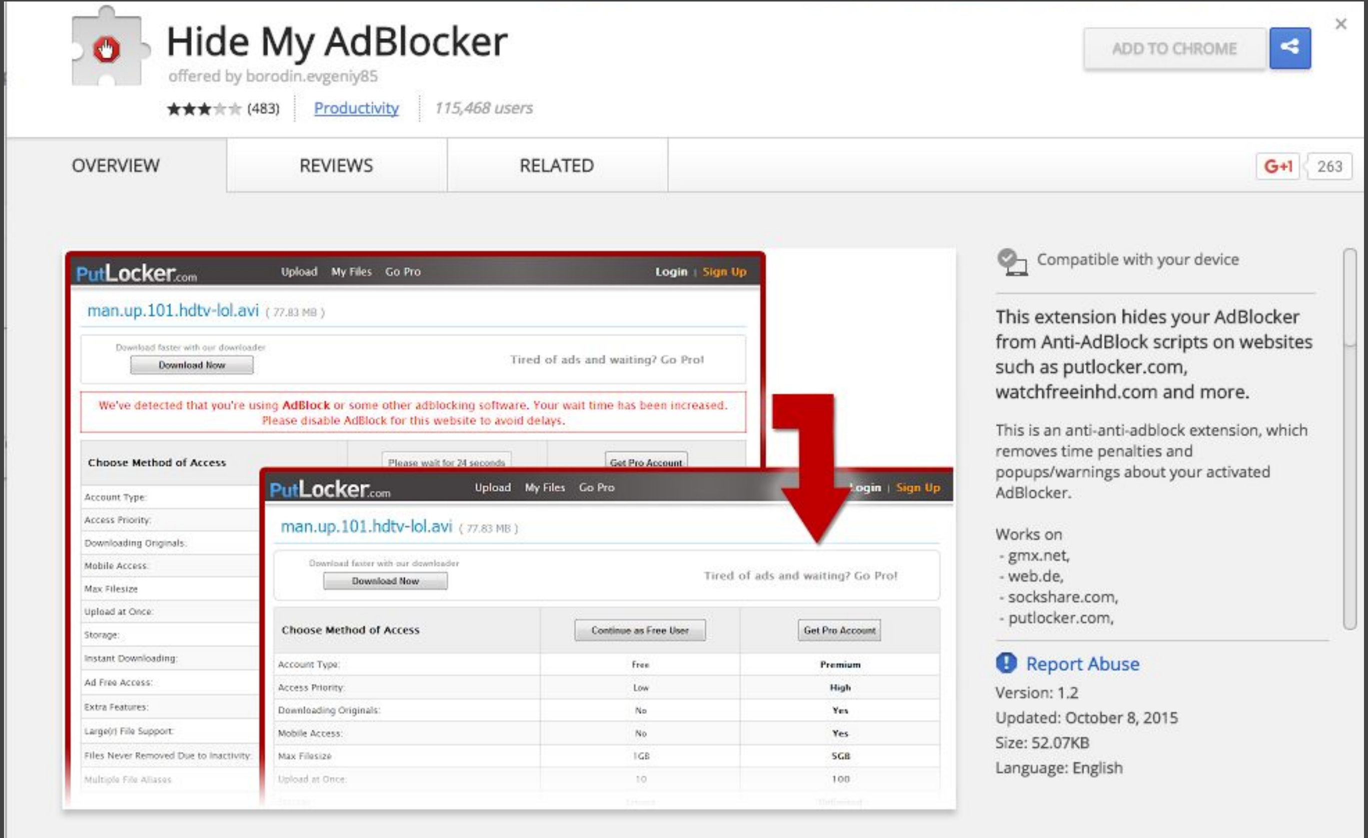
Enter your email address | Go

I have read and agree to the Terms of Service and Privacy Policy.

Already a subscriber? Sign in

RETWEETS 148 FAVORITES 54

REVENGE OF THE ADBLOCKER BLOCKER BLOCKERS!!!



Hide My AdBlocker

offered by borodin.evgeniy85

★★★☆☆ (483) | Productivity | 115,468 users

ADD TO CHROME

OVERVIEW | REVIEWS | RELATED | G+1 263

PutLocker.com Upload My Files Go Pro Login | Sign Up

man.up.101.hdtv-lol.avi (77.83 MB)

Download faster with our downloader

Download Now

Tired of ads and waiting? Go Pro!

We've detected that you're using **AdBlock** or some other adblocking software. Your wait time has been increased. Please disable AdBlock for this website to avoid delays.

Choose Method of Access	Please wait for 24 seconds	Get Pro Account
Account Type:		
Access Priority:		
Downloading Originals:		
Mobile Access:		
Max Filesize		
Upload at Once:		
Storage:		
Instant Downloading:		
Ad Free Access:		
Extra Features:		
Large(r) File Support:		
Files Never Removed Due to Inactivity:		
Multiple File Aliases		

PutLocker.com Upload My Files Go Pro Login | Sign Up

man.up.101.hdtv-lol.avi (77.83 MB)

Download faster with our downloader

Download Now

Tired of ads and waiting? Go Pro!

Choose Method of Access	Continue as Free User	Get Pro Account
Account Type:	Free	**Premium**
Access Priority:	Low	**High**
Downloading Originals:	No	**Yes**
Mobile Access:	No	**Yes**
Max Filesize	1GB	**5GB**
Upload at Once:	10	**100**

Compatible with your device

This extension hides your AdBlocker from Anti-AdBlock scripts on websites such as putlocker.com, watchfreeinhd.com and more.

This is an anti-anti-adblock extension, which removes time penalties and popups/warnings about your activated AdBlocker.

Works on
- gmx.net,
- web.de,
- sockshare.com,
- putlocker.com,

Report Abuse

Version: 1.2

Updated: October 8, 2015

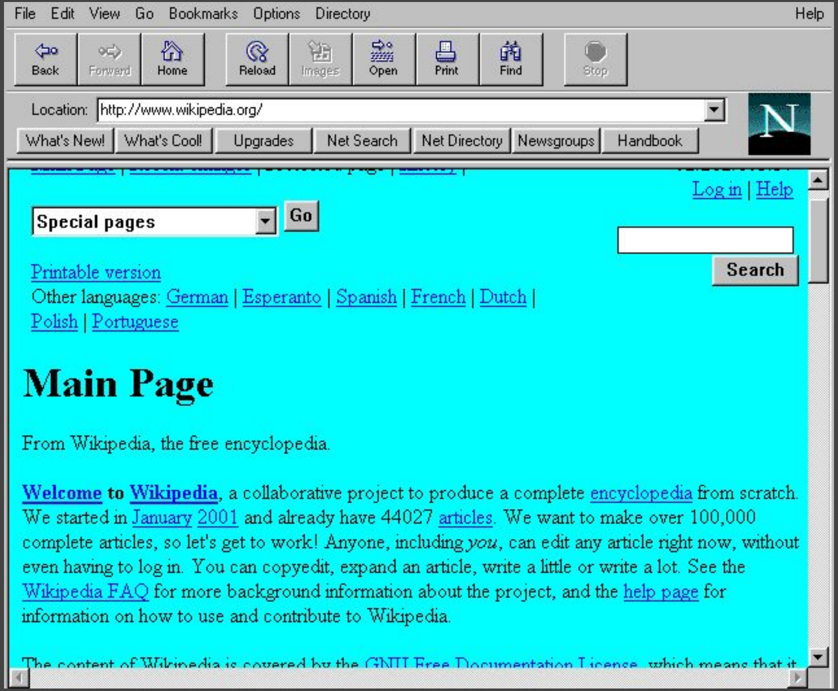
Size: 52.07KB

Language: English

A New Hope: Browser Fingerprinting

- Evade blocking algorithms that blacklist domains based on cookie frequency (ex: Privacy Badger).
- Track users who disable 3rd party cookies (ex: Safari).
- Harder to delete than cookies.
- Can reveal new information about a user.

new web features ==
new fingerprinting techniques



http://www.wikipedia.org/

Log in | Help

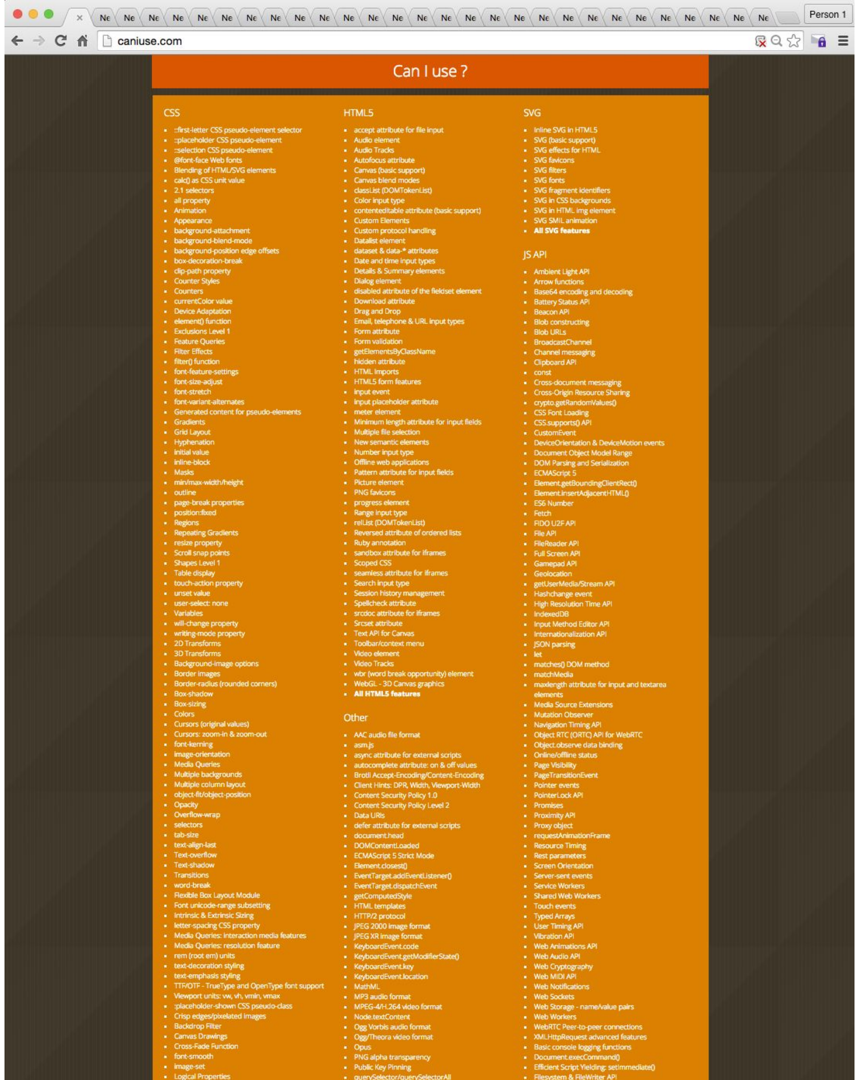
Printable version
Other languages: German | Esperanto | Spanish | French | Dutch | Polish | Portuguese

Main Page

From Wikipedia, the free encyclopedia.

Welcome to **Wikipedia**, a collaborative project to produce a complete encyclopedia from scratch. We started in January 2001 and already have 44027 articles. We want to make over 100,000 complete articles, so let's get to work! Anyone, including *you*, can edit any article right now, without even having to log in. You can copyedit, expand an article, write a little or write a lot. See the Wikipedia FAQ for more background information about the project, and the help page for information on how to use and contribute to Wikipedia.

The content of Wikipedia is covered by the GNU Free Documentation License, which means that it



caniuse.com

Can I use ?

CSS

- ::first-letter CSS pseudo-element selector
- ::placeholder CSS pseudo-element
- ::selection CSS pseudo-element
- @font-face Web fonts
- Blending of HTML/SVG elements
- calc() as CSS unit value
- 2.1 selectors
- all property
- Animation
- Appearance
- background-attachment
- background-blend-mode
- background-position edge offsets
- box-decoration-break
- clip-path property
- Counter Styles
- Counters
- currentColor value
- Device Adaptation
- element() function
- Exclusions Level 1
- Feature Queries
- Filter Effects
- filter() function
- font-feature-settings
- font-size-adjust
- font-stretch
- font-variant-alternates
- Generated content for pseudo-elements
- Gradients
- Grid Layout
- Hyphenation
- initial value
- inline-block
- Masks
- min/max-width/height
- outline
- page-break properties
- position:fixed
- Regions
- Repeating Gradients
- resize property
- Scroll snap points
- Shapes Level 1
- Table display
- touch-action property
- unset value
- user-select: none
- Variables
- will-change property
- writing-mode property
- 2D Transforms
- 3D Transforms
- Background-image options
- Border images
- Border-radius (rounded corners)
- Box-shadow
- Box-sizing
- Colors
- Cursors (original values)
- Cursors: zoom-in & zoom-out
- font-kerning
- image-orientation
- Media Queries
- Multiple backgrounds
- Multiple column layout
- object-fit/object-position
- Opacity
- Overflow-wrap
- selectors
- tab-size
- text-align-last
- Text-overflow
- Text-shadow
- Transitions
- word-break
- Flexible Box Layout Module
- Font unicode-range subsetting
- Intrinsic & Extrinsic Sizing
- letter-spacing CSS property
- Media Queries: interaction media features
- Media Queries: resolution feature
- rem (root em) units
- text-decoration styling
- text-emphasis styling
- TTF/OTF - TrueType and OpenType font support
- Viewport units: vw, vh, vmin, vmax
- :placeholder-shown CSS pseudo-class
- Crisp edges/pixelated images
- Backdrop Filter
- Canvas Drawings
- Cross-Fade Function
- font-smooth
- image-set
- Logical Properties

HTML5

- accept attribute for file input
- Audio element
- Audio Tracks
- Autofocus attribute
- Canvas (basic support)
- Canvas blend modes
- classList (DOMTokenList)
- Color input type
- contenteditable attribute (basic support)
- Custom Elements
- Custom protocol handling
- Datalist element
- dataset & data-* attributes
- Date and time input types
- Details & Summary elements
- Dialog element
- disabled attribute of the fieldset element
- Download attribute
- Drag and Drop
- Email, telephone & URL input types
- Form attribute
- Form validation
- getElementsByClassName
- hidden attribute
- HTML Imports
- HTML5 form features
- input event
- input placeholder attribute
- meter element
- Minimum length attribute for input fields
- Multiple file selection
- New semantic elements
- Number input type
- Offline web applications
- Pattern attribute for input fields
- Picture element
- PNG favicons
- progress element
- Range input type
- relList (DOMTokenList)
- Reversed attribute of ordered lists
- Ruby annotation
- sandbox attribute for iframes
- Scoped CSS
- seamless attribute for iframes
- Search input type
- Session history management
- Spellcheck attribute
- srcdoc attribute for iframes
- Srcset attribute
- Text API for Canvas
- Toolbar/context menu
- Video element
- Video Tracks
- wbr (word break opportunity) element
- WebGL - 3D Canvas graphics
- **All HTML5 features**

Other

- AAC audio file format
- asm.js
- async attribute for external scripts
- autocomplete attribute: on & off values
- Brotli Accept-Encoding/Content-Encoding
- Client Hints: DPR, Width, Viewport-Width
- Content Security Policy 1.0
- Content Security Policy Level 2
- Data URIs
- defer attribute for external scripts
- document.head
- DOMContentLoaded
- ECMAScript 5 Strict Mode
- Element.closest()
- EventTarget.addEventListener()
- EventTarget.dispatchEvent
- getComputedStyle
- HTML templates
- HTTP/2 protocol
- JPEG 2000 image format
- JPEG XR image format
- KeyboardEvent.code
- KeyboardEvent.getModifierState()
- KeyboardEvent.key
- KeyboardEvent.location
- MathML
- MP3 audio format
- MPEG-4/H.264 video format
- Node.textContent
- Ogg Vorbis audio format
- Ogg/Theora video format
- Opus
- PNG alpha transparency
- Public Key Pinning
- querySelector/querySelectorAll

SVG

- Inline SVG in HTML5
- SVG (basic support)
- SVG effects for HTML
- SVG favicons
- SVG filters
- SVG fonts
- SVG fragment identifiers
- SVG in CSS backgrounds
- SVG in HTML img element
- SVG SMIL animation
- **All SVG features**

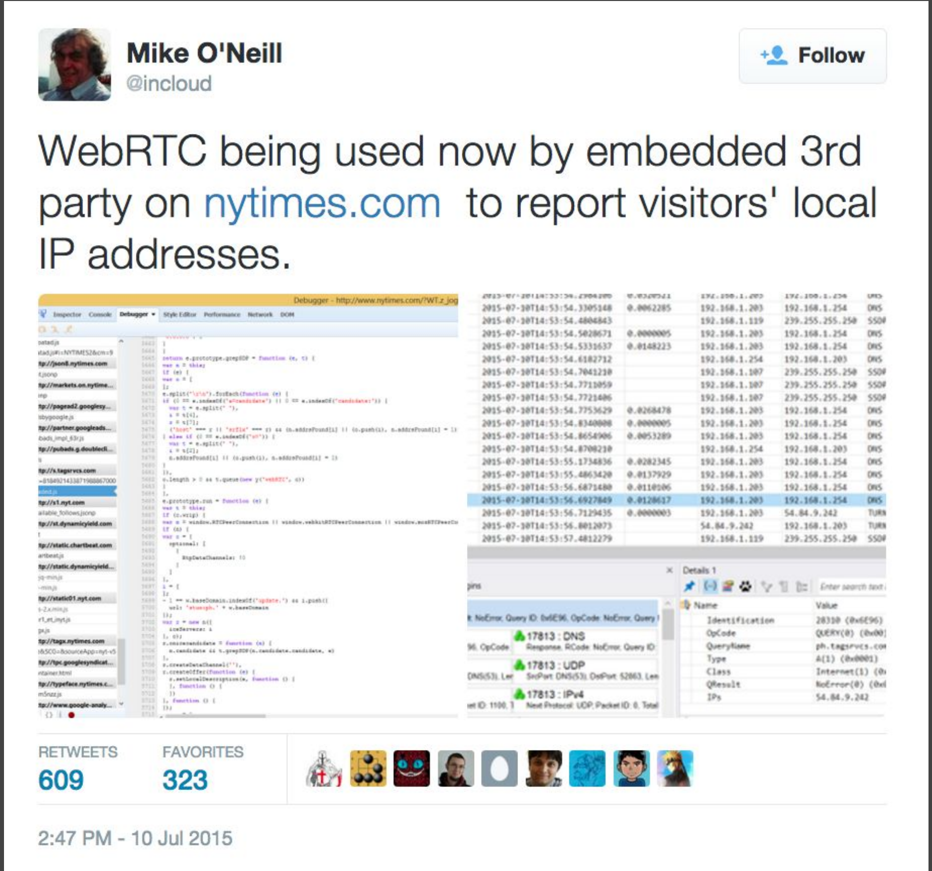
JS API

- Ambient Light API
- Arrow functions
- Base64 encoding and decoding
- Battery Status API
- Beacon API
- Blob constructing
- Blob URLs
- BroadcastChannel
- Channel messaging
- Clipboard API
- const
- Cross-document messaging
- Cross-Origin Resource Sharing
- crypto.getRandomValues()
- CSS Font Loading
- CSS.supports() API
- CustomEvent
- DeviceOrientation & DeviceMotion events
- Document Object Model Range
- DOM Parsing and Serialization
- ECMAScript 5
- Element.getBoundingClientRect()
- Element.insertAdjacentHTML()
- ES6 Number
- Fetch
- FIDO U2F API
- File API
- FileReader API
- Full Screen API
- Gamepad API
- Geolocation
- getUserMedia/Stream API
- Hashchange event
- High Resolution Time API
- IndexedDB
- Input Method Editor API
- Internationalization API
- JSON parsing
- let
- matches() DOM method
- matchMedia
- maxlength attribute for input and textarea elements
- Media Source Extensions
- Mutation Observer
- Navigation Timing API
- Object RTC (ORTC) API for WebRTC
- Object.observe data binding
- Online/offline status
- Page Visibility
- PageTransitionEvent
- Pointer events
- PointerLock API
- Promises
- Proximity API
- Proxy object
- requestAnimationFrame
- Resource Timing
- Rest parameters
- Screen Orientation
- Server-sent events
- Service Workers
- Shared Web Workers
- Touch events
- Typed Arrays
- User Timing API
- Vibration API
- Web Animations API
- Web Audio API
- Web Cryptography
- Web MIDI API
- Web Notifications
- Web Sockets
- Web Storage - name/value pairs
- Web Workers
- WebRTC Peer-to-peer connections
- XMLHttpRequest advanced features
- Basic console logging functions
- Document.execCommand()
- Efficient Script Yielding: setImmediate()
- Filesystem & FileWriter API

HOLY SHIT I HAVE 4 LIGHTSABERS ZOMG!!1

- active fingerprinting (HTML5 canvas, clock skew, installed fonts & plugins, WebRTC...)
- **supercookies** (Flash cookies, caches, HSTS, etags...)

Fingerprinting attacks in the wild



Mike O'Neill
@incloud

Follow

WebRTC being used now by embedded 3rd party on nytimes.com to report visitors' local IP addresses.

Debugger - http://www.nytimes.com/?WT.z_jog

```
return e.prototype.grepSDP = function (e, t) {
var n = this;
if (e) {
var o = [
];
e.split("\r\n").forEach(function (e) {
if (0 == e.indexOf("a=candidate") || 0 == e.indexOf("candidate:")) {
  var t = e.split(" "),
  i = t[4],
  r = t[7];
  ("host" === r || "srflx" === r) && (n.addrsFound[i] || (o.push(i), n.addrsFound[i] = 1))
} else if (0 == e.indexOf("c=")) {
  var t = e.split(" "),
  i = t[2];
  n.addrsFound[i] || (o.push(i), n.addrsFound[i] = 1)
}
}),
o.length > 0 && t.queue(new y("webRTC", o))
}
},
e.prototype.run = function (e) {
var t = this;
if (r.wrip) {
var n = window.RTCPeerConnection || window.webkitRTCPeerConnection || window.mozRTCPeerCo
if (n) {
var o = {
  optional: [
    {
      RtpDataChannels: !0
    }
  ]
},
i = [
];
- 1 == w.baseDomain.indexOf('update.') && i.push({
  url: 'stun:ph.' + w.baseDomain
});
var r = new n({
  iceServers: i
}, o);
r.onicecandidate = function (e) {
  e.candidate && t.grepSDP(e.candidate.candidate, e)
},
r.createDataChannel(''),
r.createOffer(function (e) {
  r.setLocalDescription(e, function () {
  }, function () {
  })
}, function () {
});
```

```
2015-07-10T14:53:54.3305148  0.0062285  192.168.1.203  192.168.1.254    DNS
2015-07-10T14:53:54.4804843             192.168.1.119  239.255.255.250  SSDP
2015-07-10T14:53:54.5028671  0.0000005  192.168.1.203  192.168.1.254    DNS
2015-07-10T14:53:54.5331637  0.0148223  192.168.1.203  192.168.1.254    DNS
2015-07-10T14:53:54.6182712             192.168.1.254  192.168.1.203    DNS
2015-07-10T14:53:54.7041210             192.168.1.107  239.255.255.250  SSDP
2015-07-10T14:53:54.7711059             192.168.1.107  239.255.255.250  SSDP
2015-07-10T14:53:54.7721406             192.168.1.107  239.255.255.250  SSDP
2015-07-10T14:53:54.7753629  0.0268478  192.168.1.203  192.168.1.254    DNS
2015-07-10T14:53:54.8340008  0.0000005  192.168.1.203  192.168.1.254    DNS
2015-07-10T14:53:54.8654906  0.0053289  192.168.1.203  192.168.1.254    DNS
2015-07-10T14:53:54.8708210             192.168.1.254  192.168.1.203    DNS
2015-07-10T14:53:55.1734836  0.0282345  192.168.1.203  192.168.1.254    DNS
2015-07-10T14:53:55.4863420  0.0137929  192.168.1.203  192.168.1.254    DNS
2015-07-10T14:53:56.6871480  0.0110106  192.168.1.203  192.168.1.254    DNS
2015-07-10T14:53:56.6927849  0.0120617  192.168.1.203  192.168.1.254    DNS
2015-07-10T14:53:56.7129435  0.0000003  192.168.1.203  54.84.9.242      TURN
2015-07-10T14:53:56.8012073             54.84.9.242    192.168.1.203    TURN
2015-07-10T14:53:57.4812279             192.168.1.119  239.255.255.250  SSDP
```

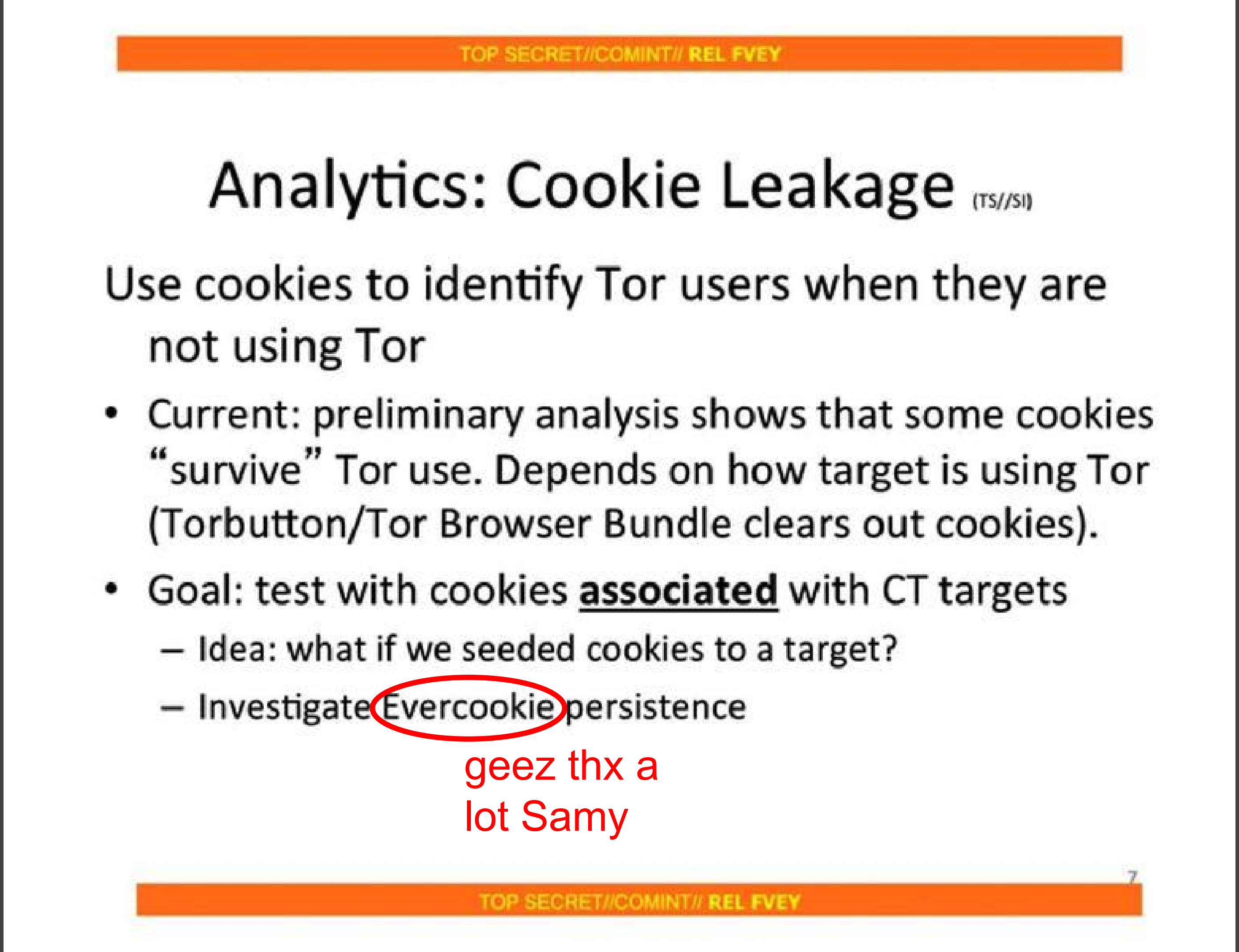
```
NoError, Query ID: 0x6E96, OpCode: NoError, Query
17813 : DNS
Response, RCode: NoError, Query ID:
17813 : UDP
SrcPort: DNS(53), DstPort: 52063, Len
17813 : IPv4
Next Protocol: UDP, Packet ID: 0, Total
```

Details 1

Name	Value
Identification	28310 (0x6E96)
OpCode	QUERY(0) (0x00)
QueryName	ph.tagsrvcs.com
Type	A(1) (0x0001)
Class	Internet(1) (0x
QResult	NoError(0) (0x
IPs	54.84.9.242

RETWEETS 609 FAVORITES 323

2:47 PM - 10 Jul 2015



Analytics: Cookie Leakage (TS//SI)

Use cookies to identify Tor users when they are not using Tor

- Current: preliminary analysis shows that some cookies "survive" Tor use. Depends on how target is using Tor (Torbutton/Tor Browser Bundle clears out cookies).
- Goal: test with cookies **associated** with CT targets
  - Idea: what if we seeded cookies to a target?
  - Investigate Evercookie persistence

geez thx a lot Samy

#realtalk

How would you track a paranoid user who clears cookies & uses an adblocker?

Could fingerprint them, but adblockers & browsers will get better at blocking you…

…unless blocking causes too much collateral damage.

Collateral:

Privacy-conscious users usually care about security.

Can we fingerprint them using security features that are too important for them to turn off?

Trick #1: Abuse HTTP Public Key Pinning

HPKP (RFC 7469)

Server: One of these hashes must be in the TLS cert chain you receive from me.

Browser: DOPE!! NEXT TIME I SEE YOU I WILL CHECK IT BEFORE I WRECK IT

```
Public-Key-Pins:

max-age=3000;

pin-sha256="
d6qzRu9zOECb90Uez27xWltNsj0e1Md
7GkYYkVoZWmM=";

pin-sha256="
E9CZ9INDbd+2eRQozYqqbQ2yXLVKB9+
xcprMF+44U1g=";

report-uri= “http://example.
com/report”;

includeSubdomains;
```

How long to cache this shit for

SHA-256 of a pub. key in the cert chain. Browser checks & caches this.

SHA-256 of a backup pub. key (required). Must NOT be in the cert chain. Browser caches this.

POST endpoint to report pin validation failures (optional).

Whether to pin for the host’s subdomains as well (optional).

Supercookie #1: fake backup pins

1. https://example.com sets a unique backup pin for each user + includeSubdomains + report-uri.
2. <img src=“https://bad.example.com”> serves a chain that deliberately fails pin validation.
3. **A validation failure report is sent which includes a unique cached backup pin!**

Trick #2: Abuse HTTP Strict Transport Security + Content Security Policy

HSTS (RFC 6797)

Server: Hey, I just met you, and this is crazy, but please only call me over HTTPS for the next 604800 seconds.

Browser: OK

```
Strict-Transport-Security:

max-age=3000;

includeSubdomains;
```

How long to remember to only connect to this host via HTTPS

Whether subdomains should also only be connected to over HTTPS (optional).

Supercookie #2: HSTS cache state

1. sneaky.com wants to fingerprint users.
2. example.com is known to support HSTS.
3. sneaky.com/index.html embeds <img src= ‘http://example.com’>.

What happens then?

Case 1: Browser has never visited example.com

-> makes a network round-trip, gets 301/302 to https://example.com

Case 2: Browser visited example.com before.

-> HSTS causes an “internal” redirect (307) to https://example.com/ ~immediately

If we can measure the HTTP to HTTPS redirect latency, we can distinguish Case 1 from Case 2!

Q: How do we measure that?

A: Abuse one more browser security feature.

Content Security Policy (W3C spec)

Server: For your safety, please only allow resources of type <X> from origins <A> & <B> while on this page.

Browser: I GOT U FAM

```
Content-Security-Policy:

img-src: https://*;

script-src: ‘self’ *.
scripts.com cdn.example.com
```

Allow images to load from HTTPS origins only

Allow scripts to load from the page’s origin, *.scripts.com, and cdn.example.com only.

The Missing Ingredient:
Set CSP to ‘**img-src http://***’

HTTPS image requests are blocked and fire an error event to JS listeners.

Why is this useful?

1. JS only lets us listen for img onerror and onload events. Turns out CSP violation triggers onerror consistently and early in the fetch pipeline.
2. If browser ever completes a request for https://example.com, it will get the HSTS pin and future results are polluted. CSP prevents this from happening!

After setting CSP:

```
Case 1: Browser has never visited example.com

 -> makes network request, gets 301/302 to
https://example.com, img onerror fires.

Case 2: Browser visited example.com before.

 -> HSTS rewrites src to https://example.com/
~immediately, img onerror fires.
```

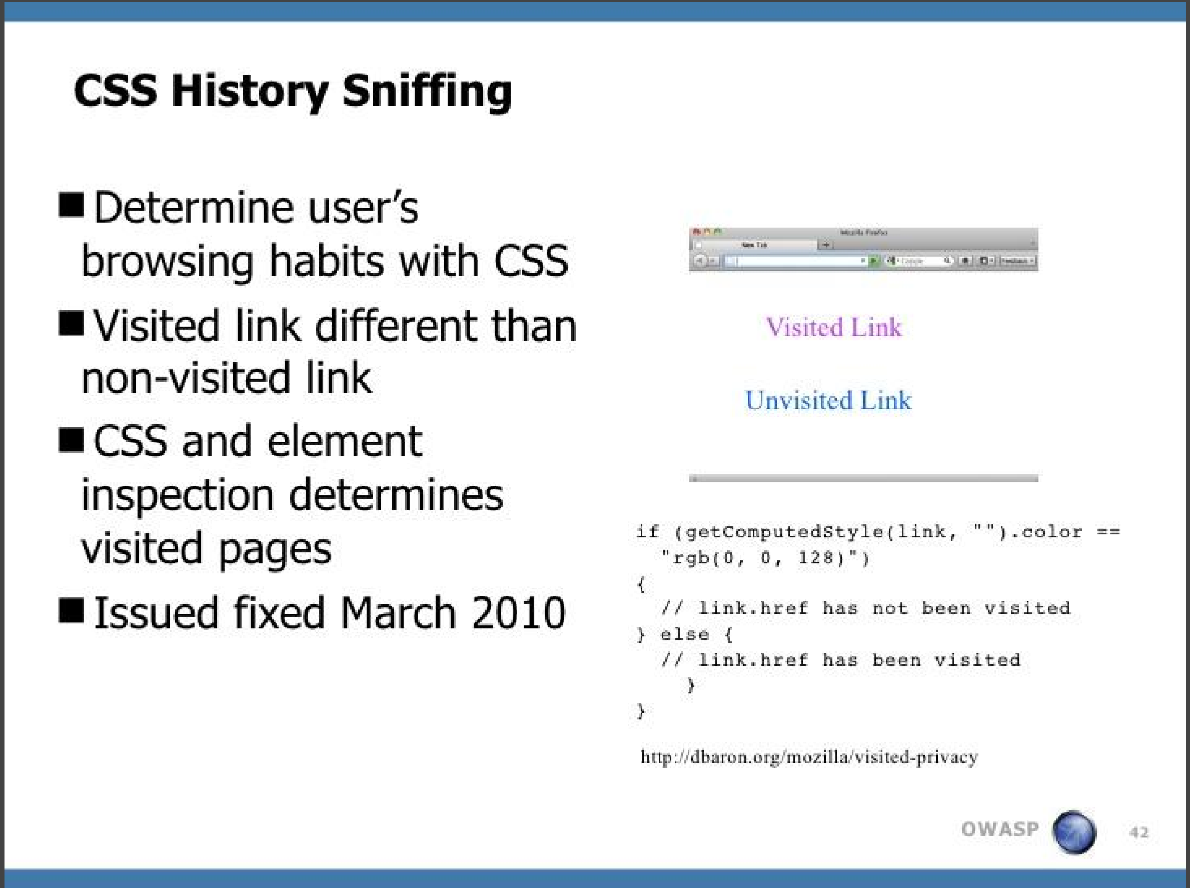
How long does the HTTP to HTTPS redirect take?

```
Case 1: Browser has never visited example.com

 -> Order of 100ms depending on network latency
and site response time.

Case 2: Browser visited example.com before.

 -> Order of 1ms, independent of the site and
network conditions.
```

Putting it all together

Remember the CSS visited-selector bug?

Slide from Michael Coates, 2011 ->



CSS History Sniffing

- Determine user's browsing habits with CSS
- Visited link different than non-visited link
- CSS and element inspection determines visited pages
- Issued fixed March 2010

Visited Link
Unvisited Link

```
if (getComputedStyle(link, "").color ==
  "rgb(0, 0, 128)")
{
  // link.href has not been visited
} else {
  // link.href has been visited
    }
}
```

http://dbaron.org/mozilla/visited-privacy

OWASP 42

That was soooooo 2010

New plan:

1. Scrape Alexa Top 1M for hosts that send HSTS and aren’t preloaded.
2. Load all the HSTS hosts asynchronously on one page.
3. Measure the onerror timing & separate hosts into visited and unvisited.

Turns out...

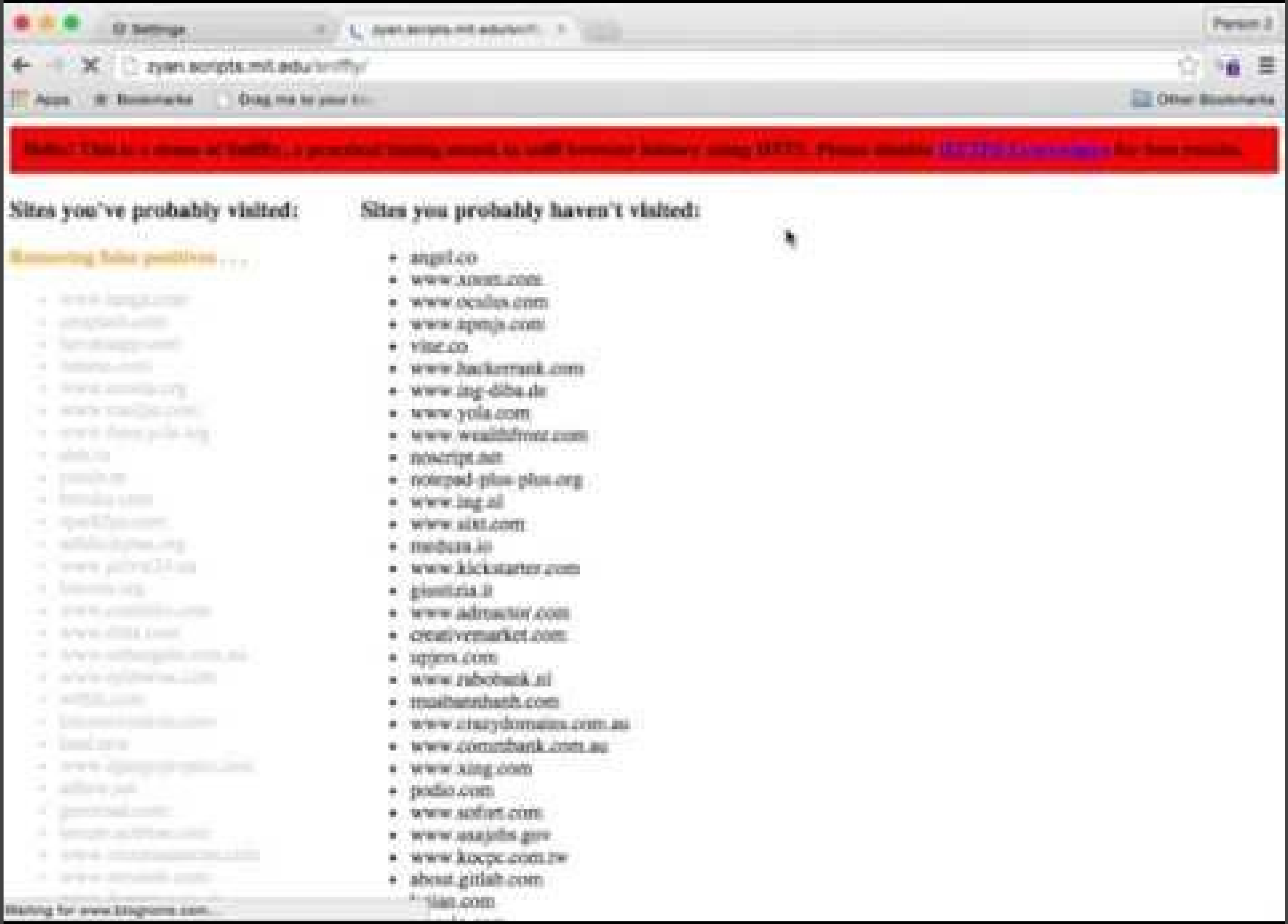
Redirect timing is hard to measure accurately for 300+ async image loads at once.

Improved by calibrating timing drift using a request to a preloaded HSTS host every other request.

Chrome still had many false positives; confirmed timings for positive results using synchronous loads.

demo:

http://zyan.scripts.mit.edu/sniffly



zyan.scripts.mit.edu/sniffly/

Note: This is a demo of Sniffly, a practical timing attack to sniff browser history using HSTS. Please disable HTTPS Everywhere for best results.

Sites you've probably visited:

Removing false positives . . .

- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]
- [illegible]

Sites you probably haven't visited:

- angel.co
- www.xoom.com
- www.oculus.com
- www.npmjs.com
- vine.co
- www.hackerrank.com
- www.ing-diba.de
- www.yola.com
- www.wealthfront.com
- noscript.net
- notepad-plus-plus.org
- www.ing.nl
- www.sixt.com
- mobura.io
- www.kickstarter.com
- giustizia.it
- www.adroactor.com
- creativemarket.com
- upjets.com
- www.rabobank.nl
- mutbarenbank.com
- www.crazydomains.com.au
- www.commbank.com.au
- www.xing.com
- podio.com
- www.sofort.com
- www.usajobs.gov
- www.kocpc.com.tw
- about.gitlab.com
- ·· ·tian.com

Waiting for www.blogrome.com...

scraper + tracker code:

https://github.com/diracdeltas/sniffly

Your mileage may vary

- Results depend on latest HSTS preload list.
- HTTPS Everywhere & other extensions cause false positives.
- Doesn’t work as-is in Tor Browser thanks to 100 ms timing buckets.

Your mileage may vary

- Only leaks origin, not full path . . . or does it?

Actually, looks feasible to adapt this attack to leak the 301 redirect cache instead of the HSTS cache. :)

TO BE CONTINUED...

The End

Call me maybe:

yan@mit.edu / @bcrypt

Special thanks to Scott Helme, Jan Schaumann, Chris Palmer, and Chris Rohlf for feedback and demo testing.

Many <3's to White Ops for sponsoring my trip to ToorCon!