Melting Pot of Origins

Compromising the Intermediary Web Services that Rehost Websites

Takuya Watanabe[†‡], Eitaro Shioji[†], Mitsuaki Akiyama[†], Tatsuya Mori[‡§⊥]

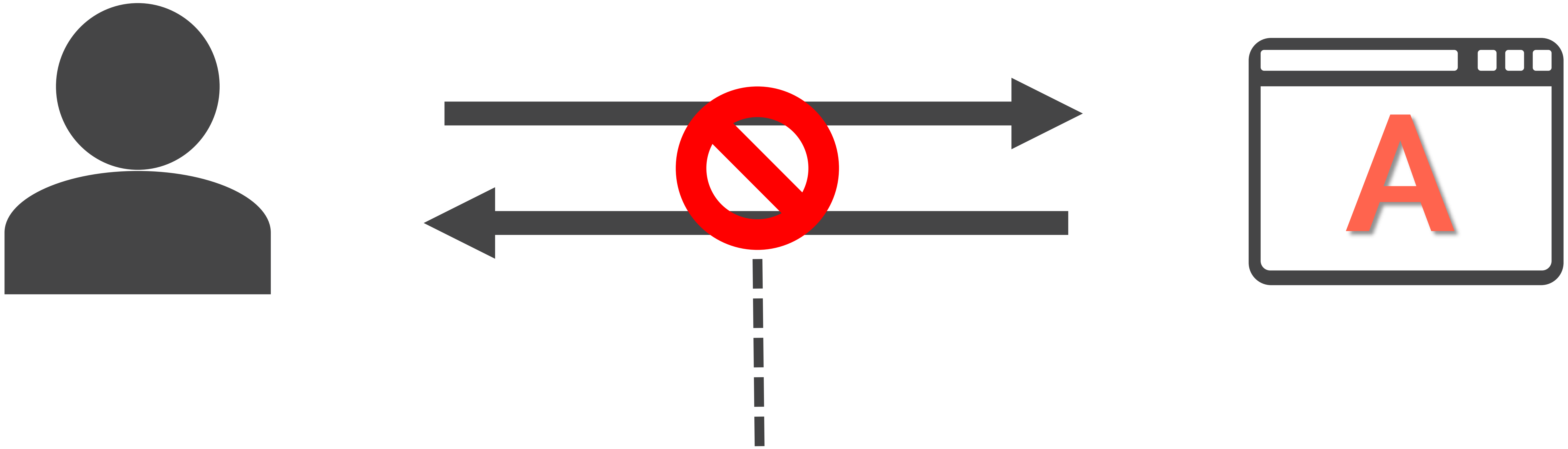
[†]NTT Secure Platform Laboratories,
[‡]Waseda University, [§]NICT, [⊥]RIKEN AIP

NDSS Symposium
24th February, 2020

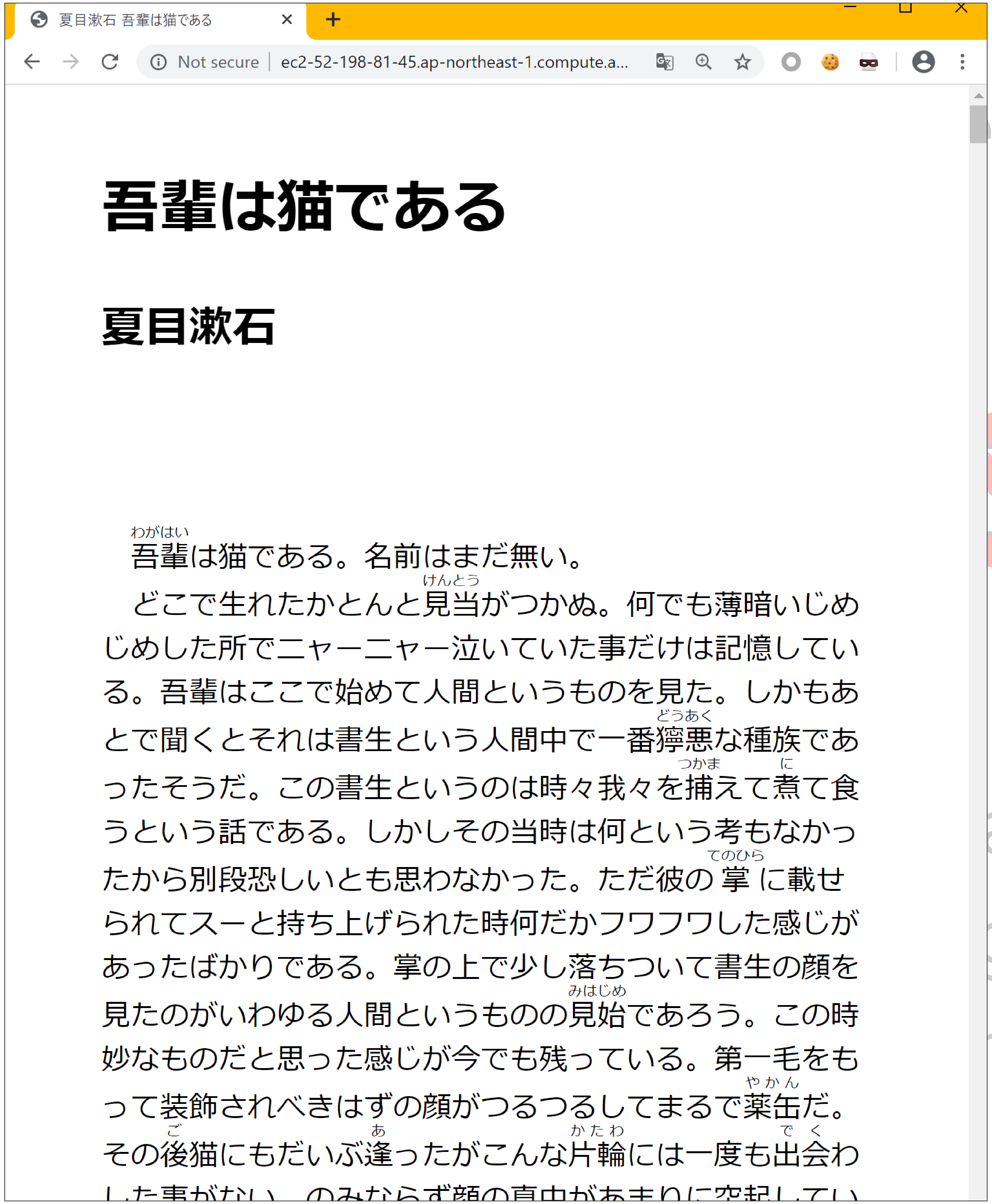
This work...

- Study security flaws of *web rehosting services*
- Present five attacks (e.g., persistent MITM)
- Demonstrate feasibility on real services
- Provide countermeasures

Obstacles to Web Access



- Language barrier
- Missing web page
- Blocking

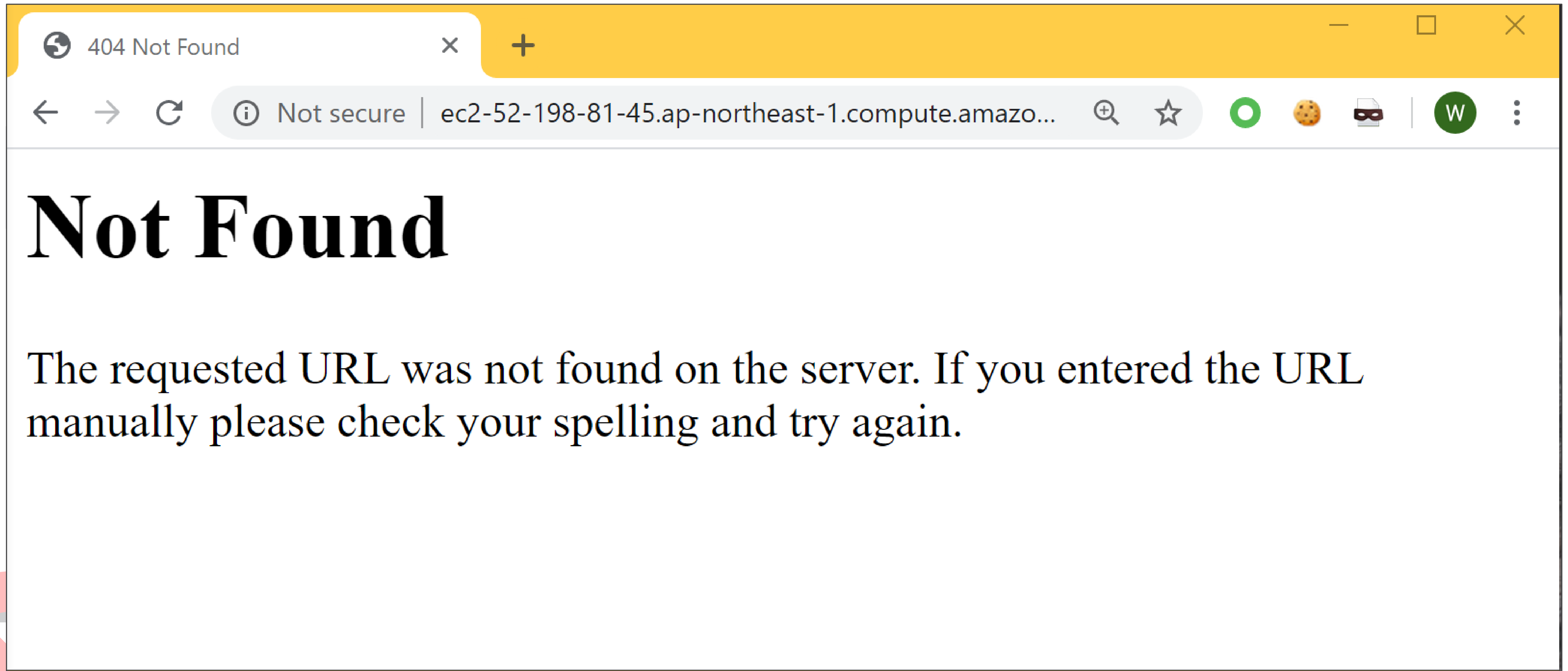


吾輩は猫である

夏目漱石

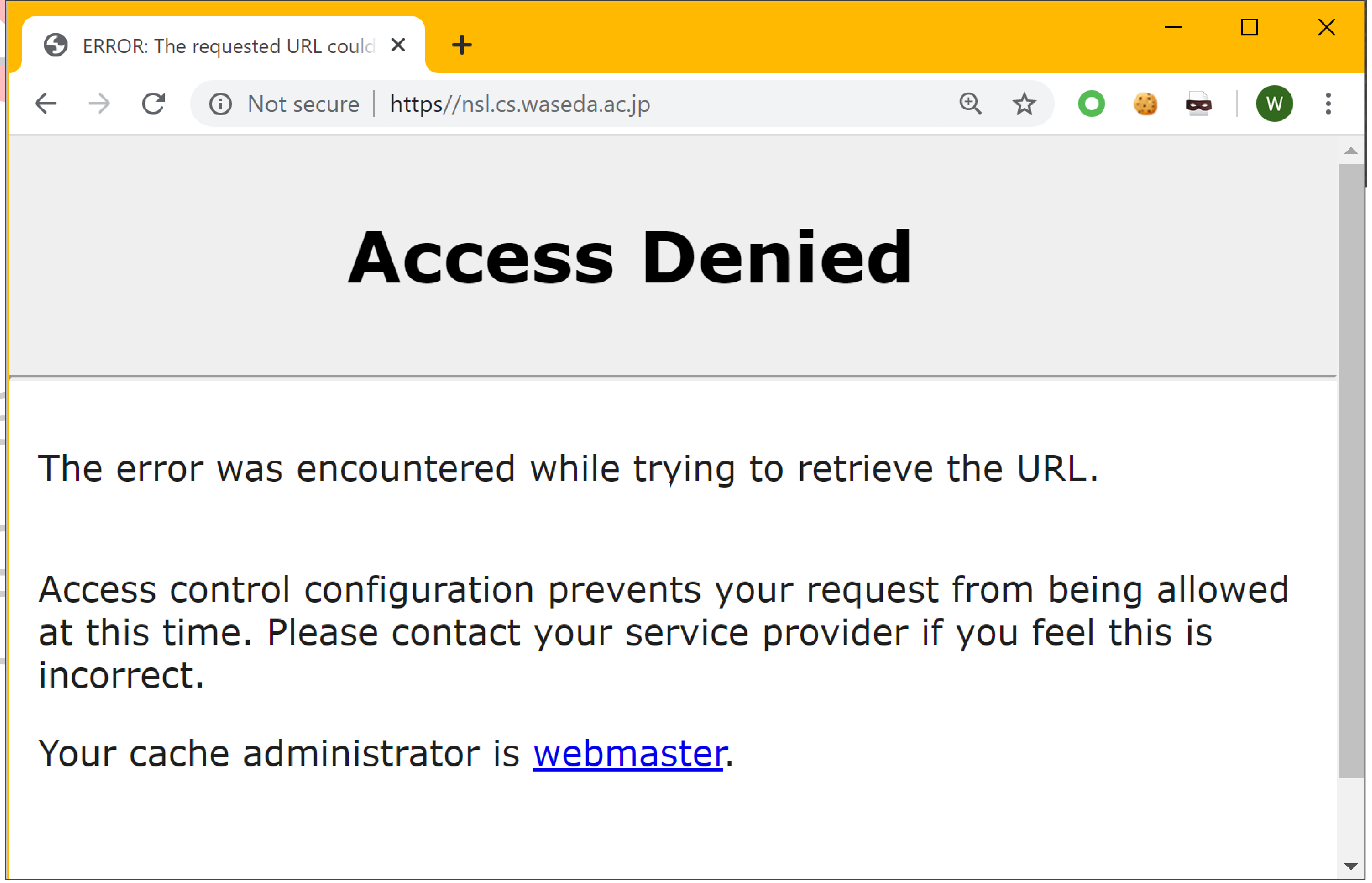
吾輩(わがはい)は猫である。名前はまだ無い。

どこで生れたかとんと見当(けんとう)がつかぬ。何でも薄暗いじめじめした所でニャーニャー泣いていた事だけは記憶している。吾輩はここで始めて人間というものを見た。しかもあとで聞くとそれは書生という人間中で一番獰悪(どうあく)な種族であったそうだ。この書生というのは時々我々を捕(つかま)えて煮(に)て食うという話である。しかしその当時は何という考もなかったから別段恐しいとも思わなかった。ただ彼の掌(てのひら)に載せられてスーと持ち上げられた時何だかフワフワした感じがあったばかりである。掌の上で少し落ちついて書生の顔を見たのがいわゆる人間というものの見始(みはじめ)であろう。この時妙なものだと思った感じが今でも残っている。第一毛をもって装飾されべきはずの顔がつるつるしてまるで薬缶(やかん)だ。その後(ご)猫にもだいぶ逢(あ)ったがこんな片輪(かたわ)には一度も出会(でく)わした事がない。のみならず顔の真中があまりに突起してい



Not Found

The requested URL was not found on the server. If you entered the URL manually please check your spelling and try again.



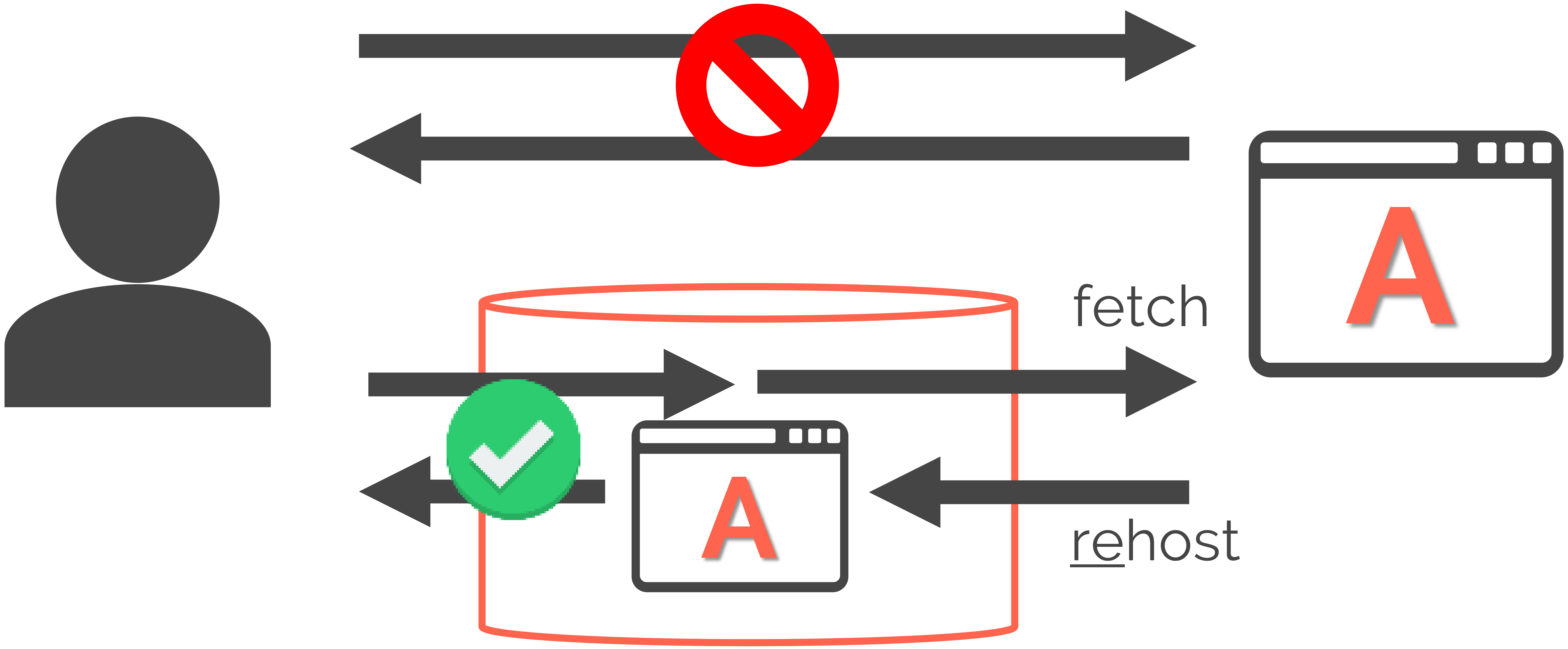
Access Denied

The error was encountered while trying to retrieve the URL.

Access control configuration prevents your request from being allowed at this time. Please contact your service provider if you feel this is incorrect.

Your cache administrator is webmaster.

Solution: Web Rehosting



Intermediary service

Web Rehosting Services Enhance Openness of Web

Website translator

Web archive

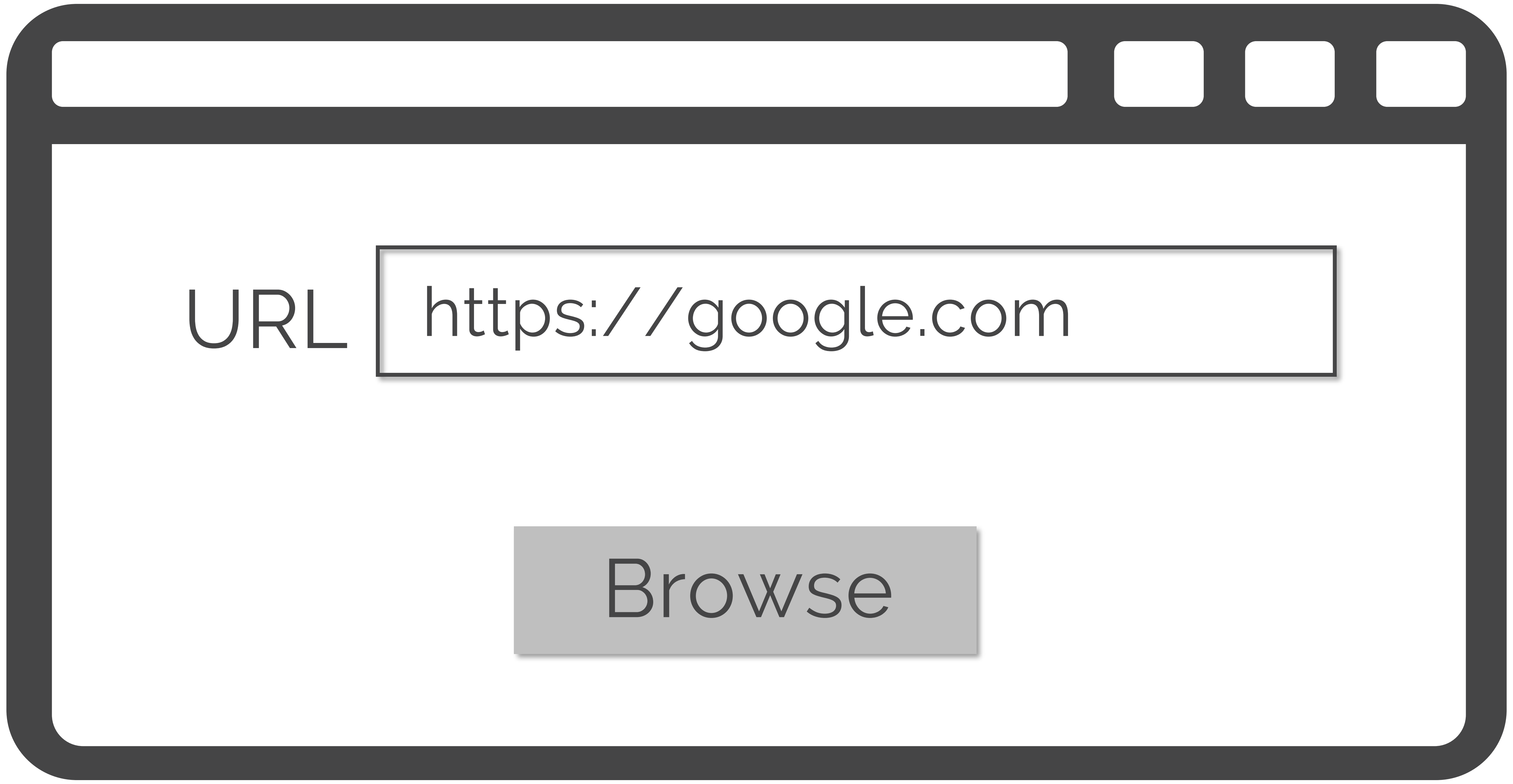
Web-based proxy

21 Web Rehosting Services We Examined

Web Proxy	ProxySite, Hide My Ass!, Hide me, Sitenable Proxy, FilterBypass, ProxFree, toolur, hidester, GenMirror, UnblockVideos, Service-α
Web Translator	Google Translate, Bing Translator, Weblio, PROMT, Yandex.Translate, Baidu Translate, Service-β
Web Archive	Wayback Machine, Google Cache, FreezePage

> 200M sessions/day

Typical Web Rehosting Usage

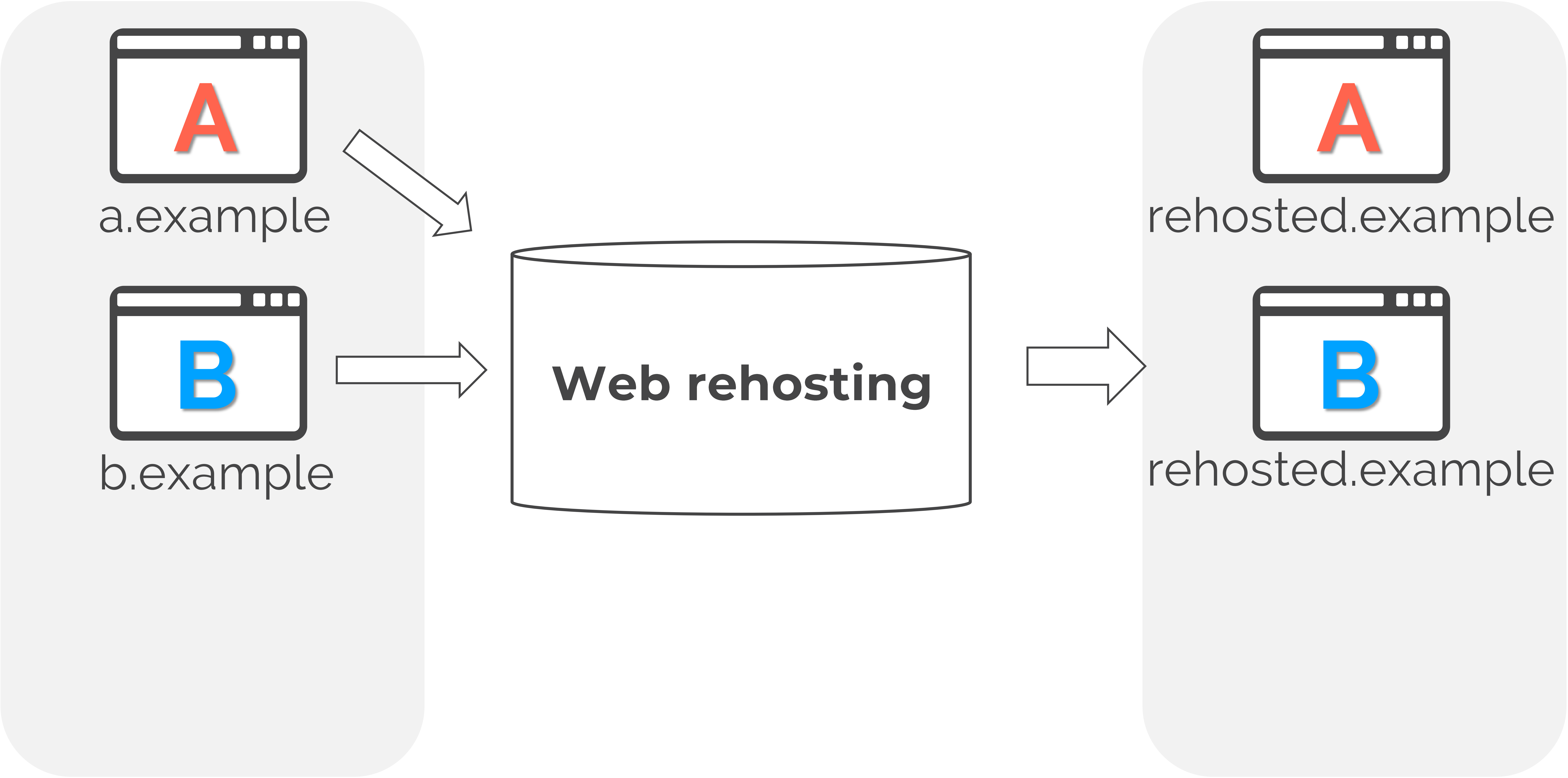


URL https://google.com

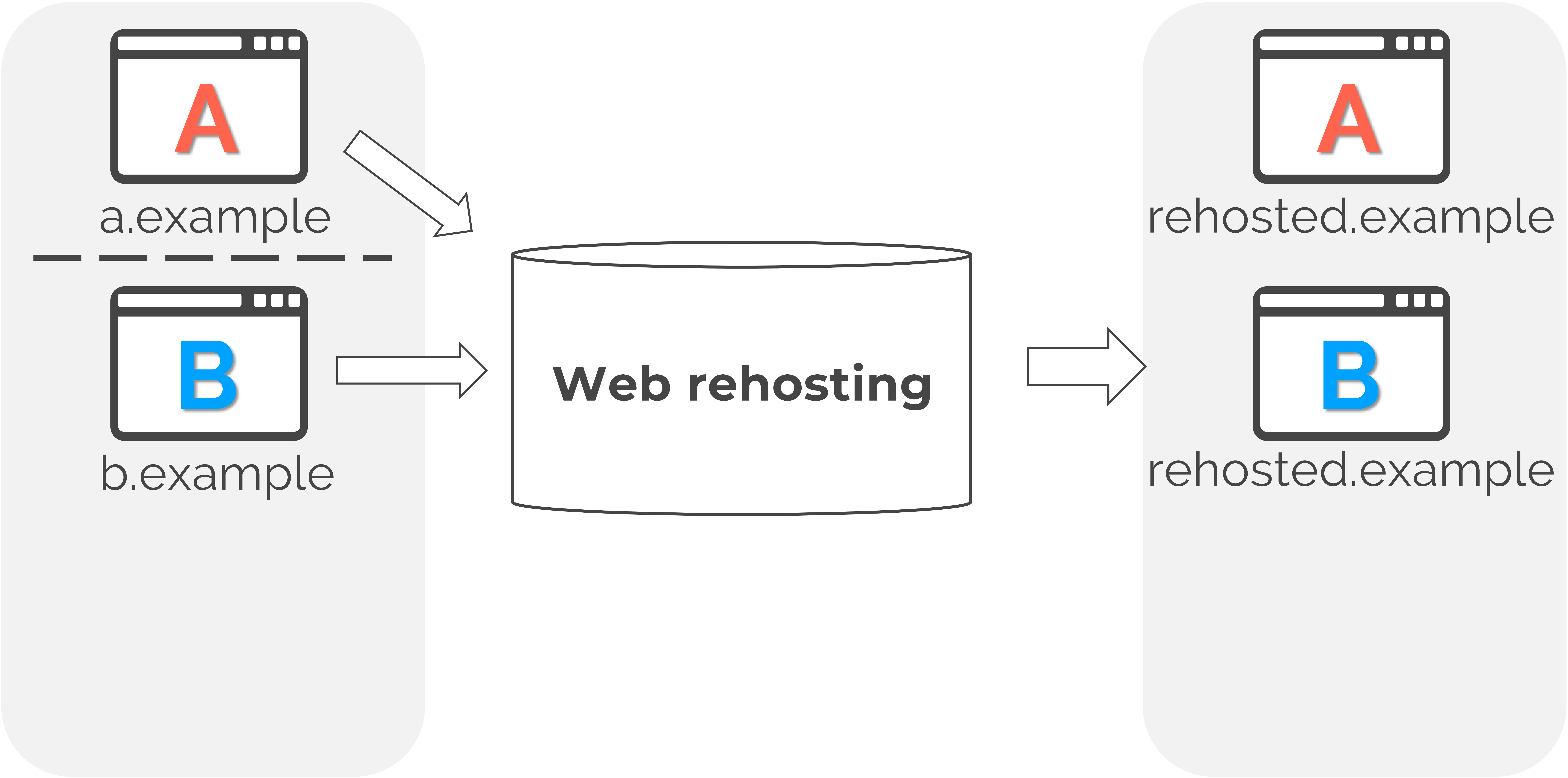
or

Direct link:
https://rehosted.example/?url=https://google.com

Web Rehosting Architecture



Web Rehosting Architecture



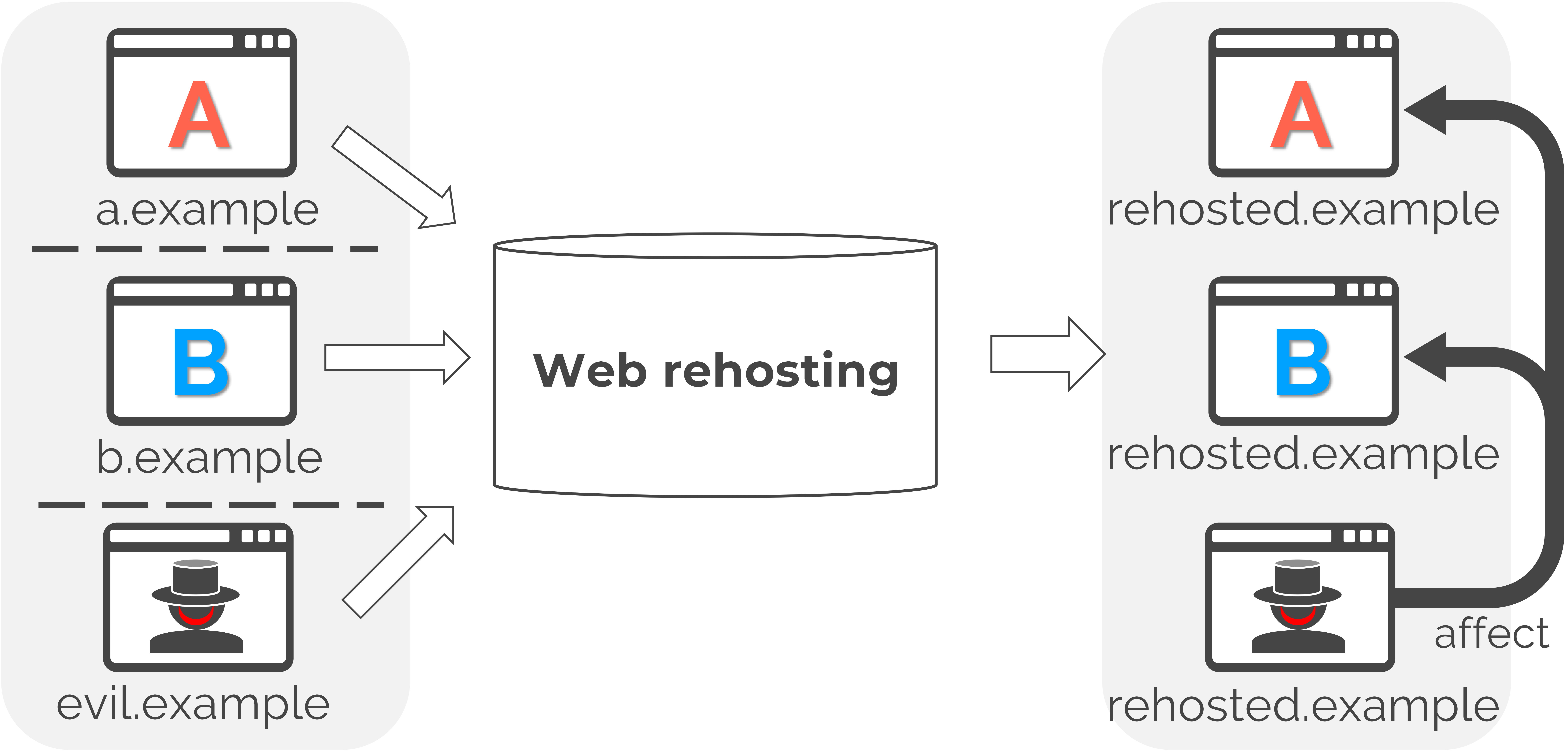
(— — — Boundary of origins)

Rehosting Rules

- URL Rewriting

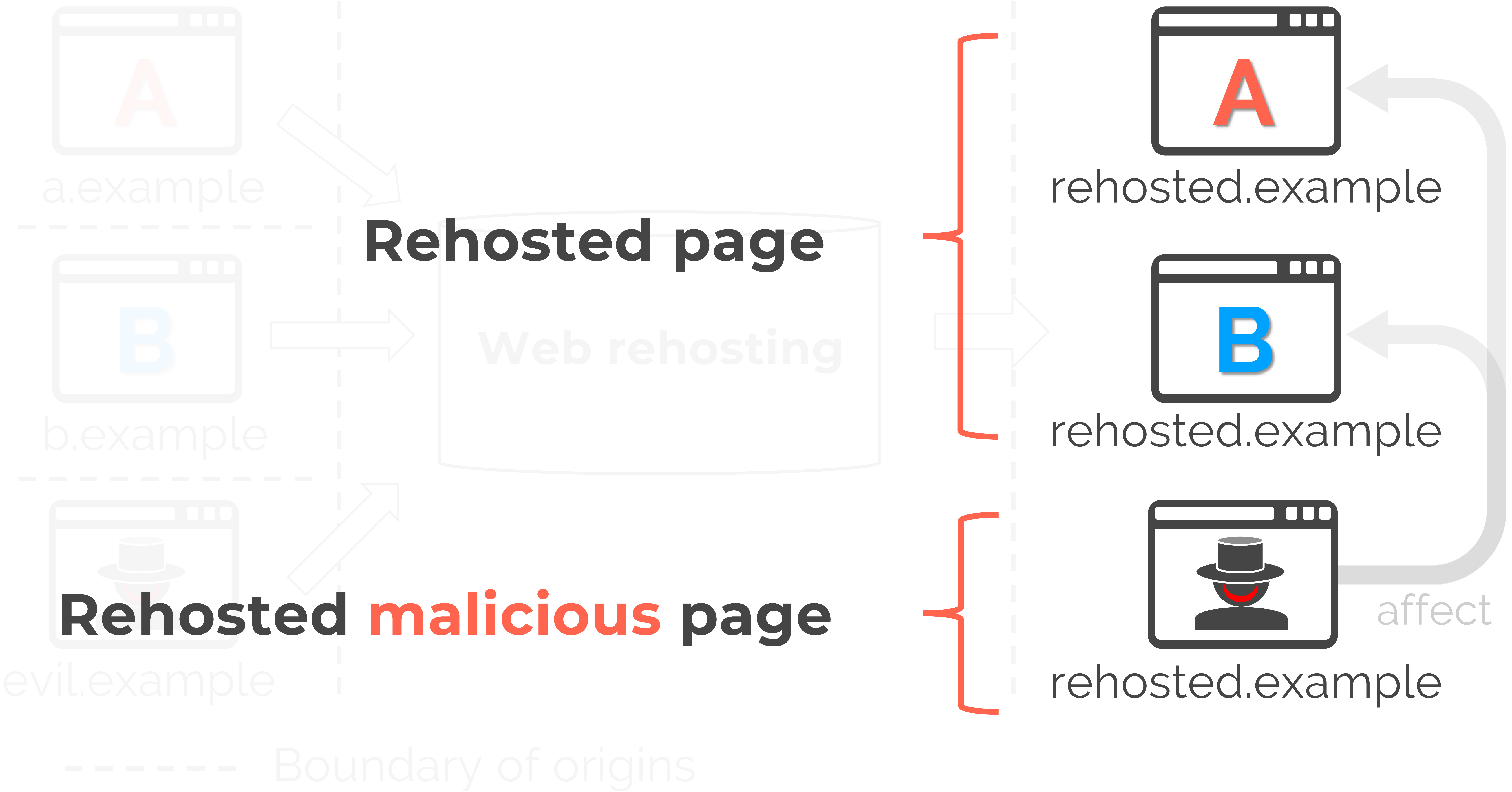
  https://a.example
  → https://rehosted.example/?url=https://a.example
- Rehostable File Type
  - HTML, plaintext
  - JavaScript (except some translators)
- Handling Browser Resources
  - remain resource accesses via JavaScript
  - relay HTTP cookie (web proxy)

Attack Surface



(— — — Boundary of origins)

Attack Surface



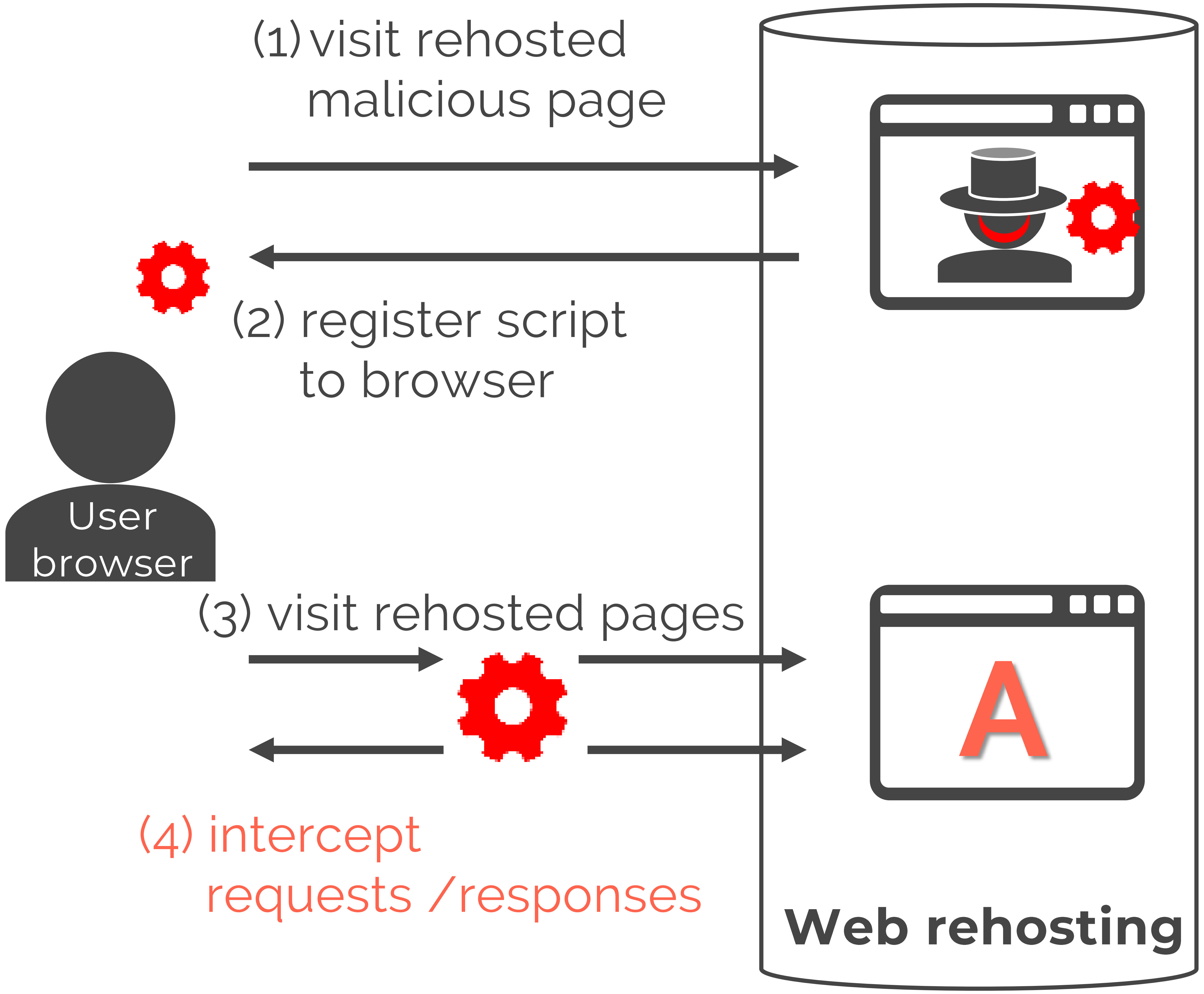
Rehosted page

Rehosted malicious page

Attacks against Web Rehosting

#	Attacks	Exploited Resources
I	Persistent MITM	Service Worker, AppCache
II	Privilege Abuse	Camera, Microphone, Location, Notification, etc.
III	Credential Theft	Password Manager
IV	History Theft	Cookie (written by JavaScript), localStorage
V	Session Hijacking and Injection	Cookie (written by HTTP header)

Threat Model



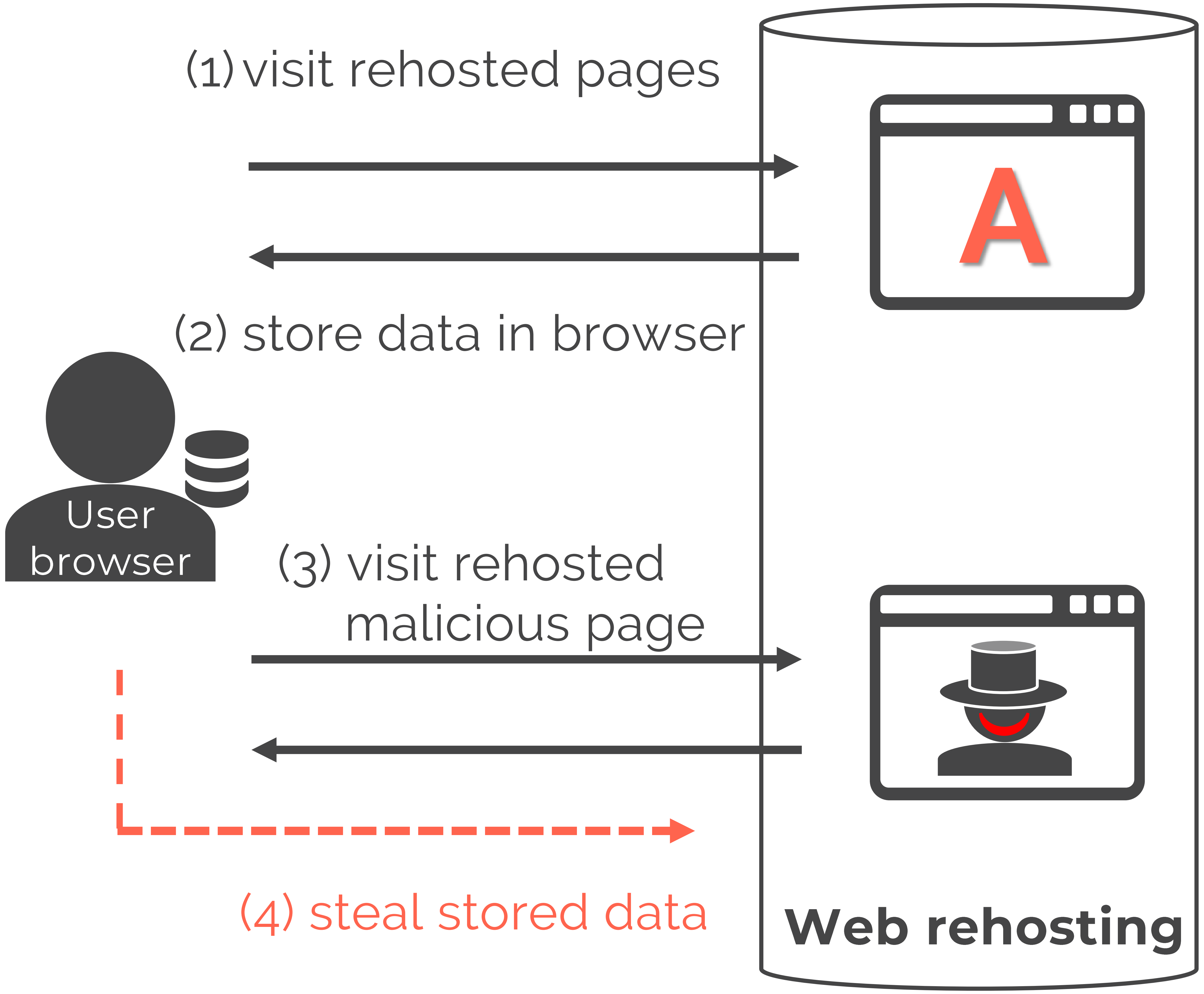
(1) visit rehosted malicious page

(2) register script to browser

(3) visit rehosted pages

(4) intercept requests /responses

Attack I



(1) visit rehosted pages

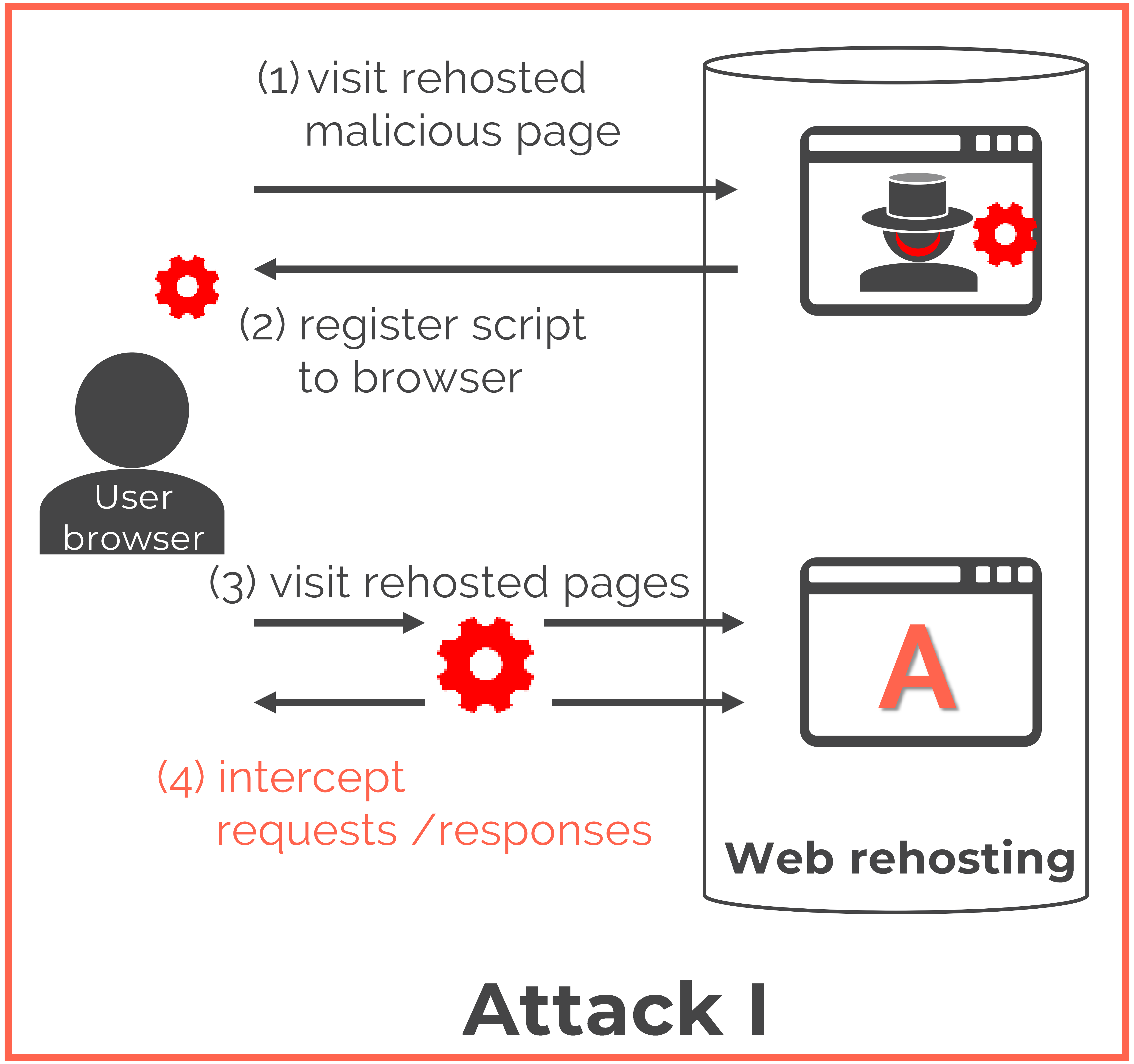
(2) store data in browser

(3) visit rehosted malicious page

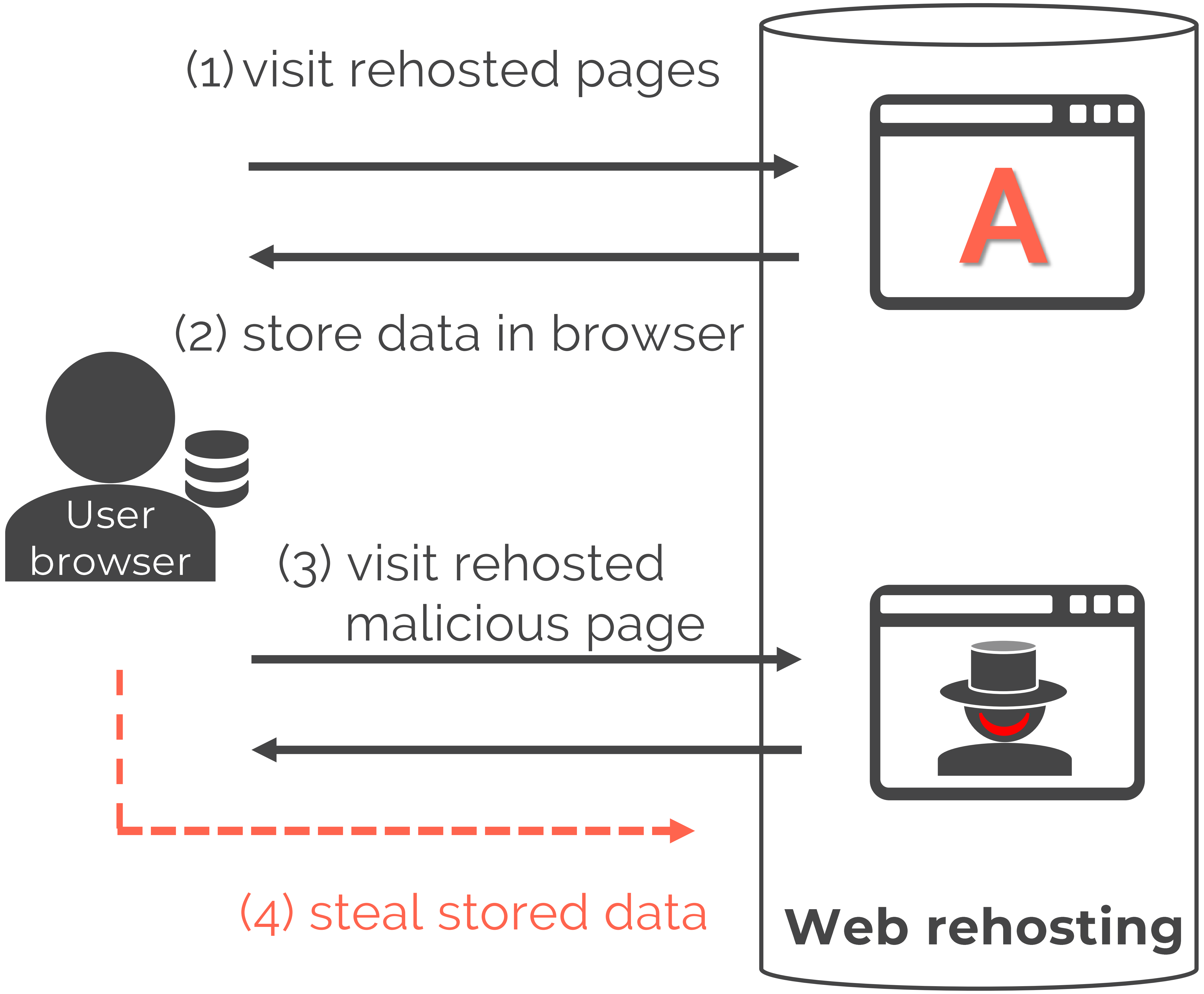
(4) steal stored data

Attack II - V

Threat Model

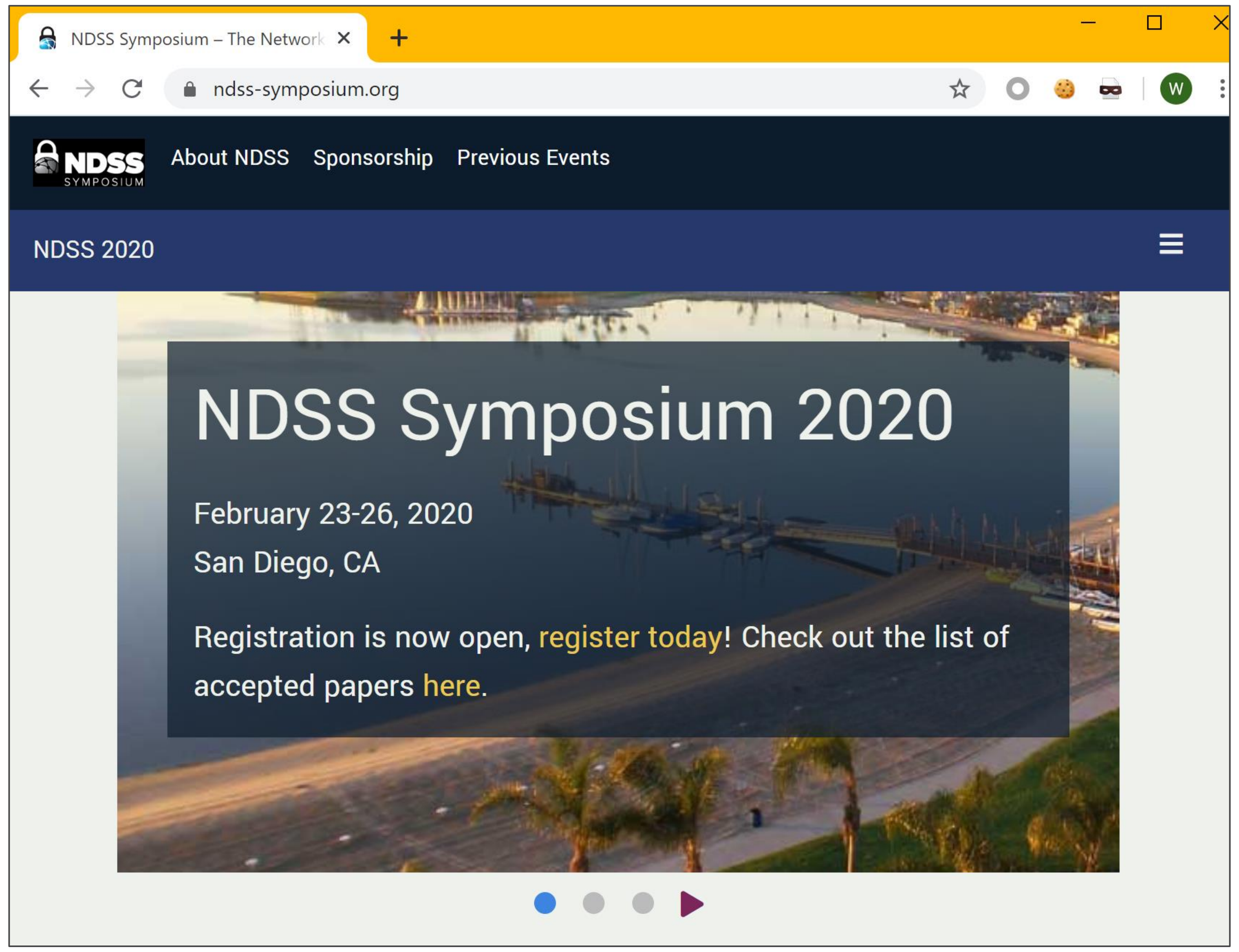

(1) visit rehosted malicious page
(2) register script to browser
User browser
(3) visit rehosted pages
A
(4) intercept requests /responses
Web rehosting

Attack I



Attack II - V

Attack I: Persistent MITM

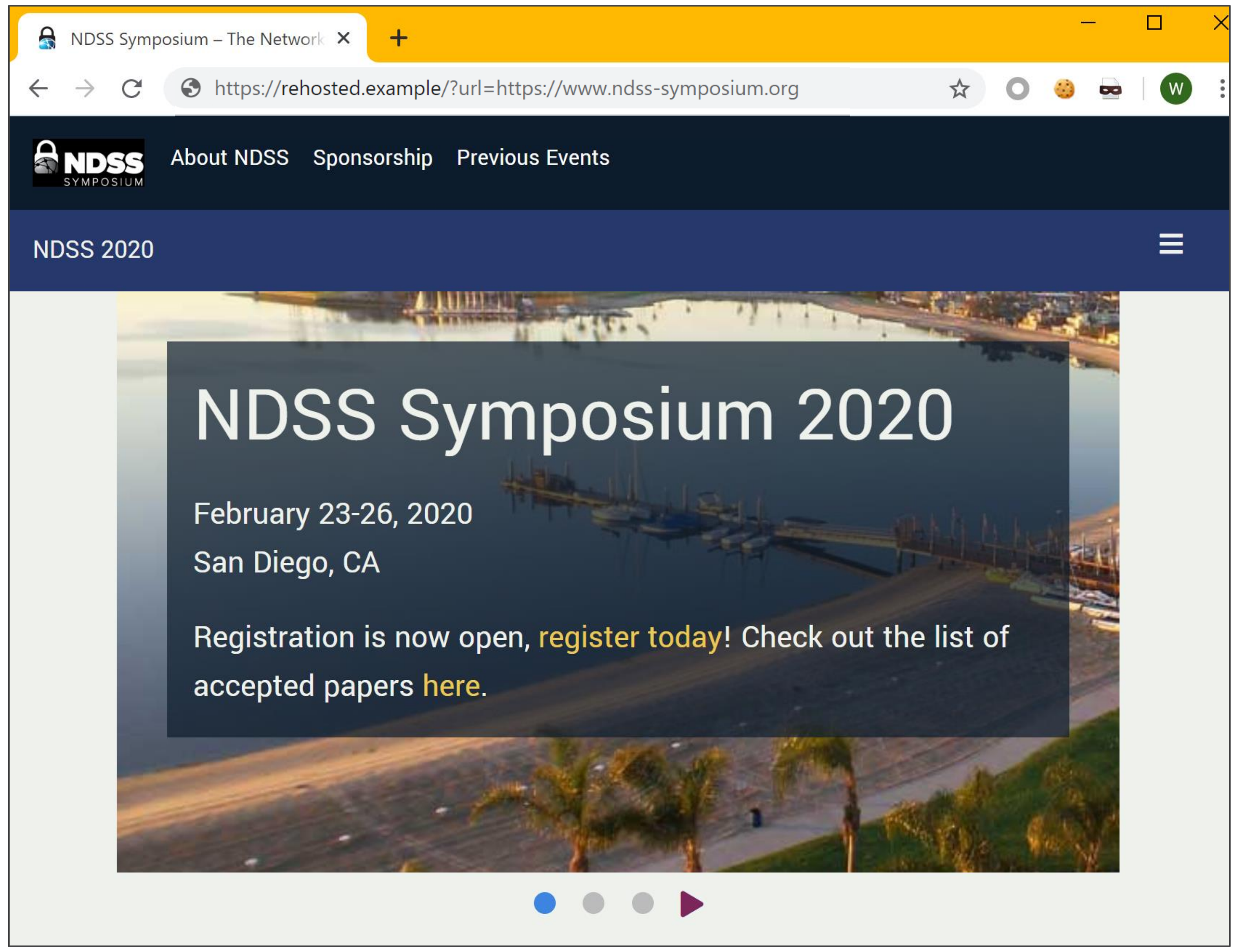


NDSS Symposium 2020

February 23-26, 2020
San Diego, CA

Registration is now open, register today! Check out the list of accepted papers here.

Direct

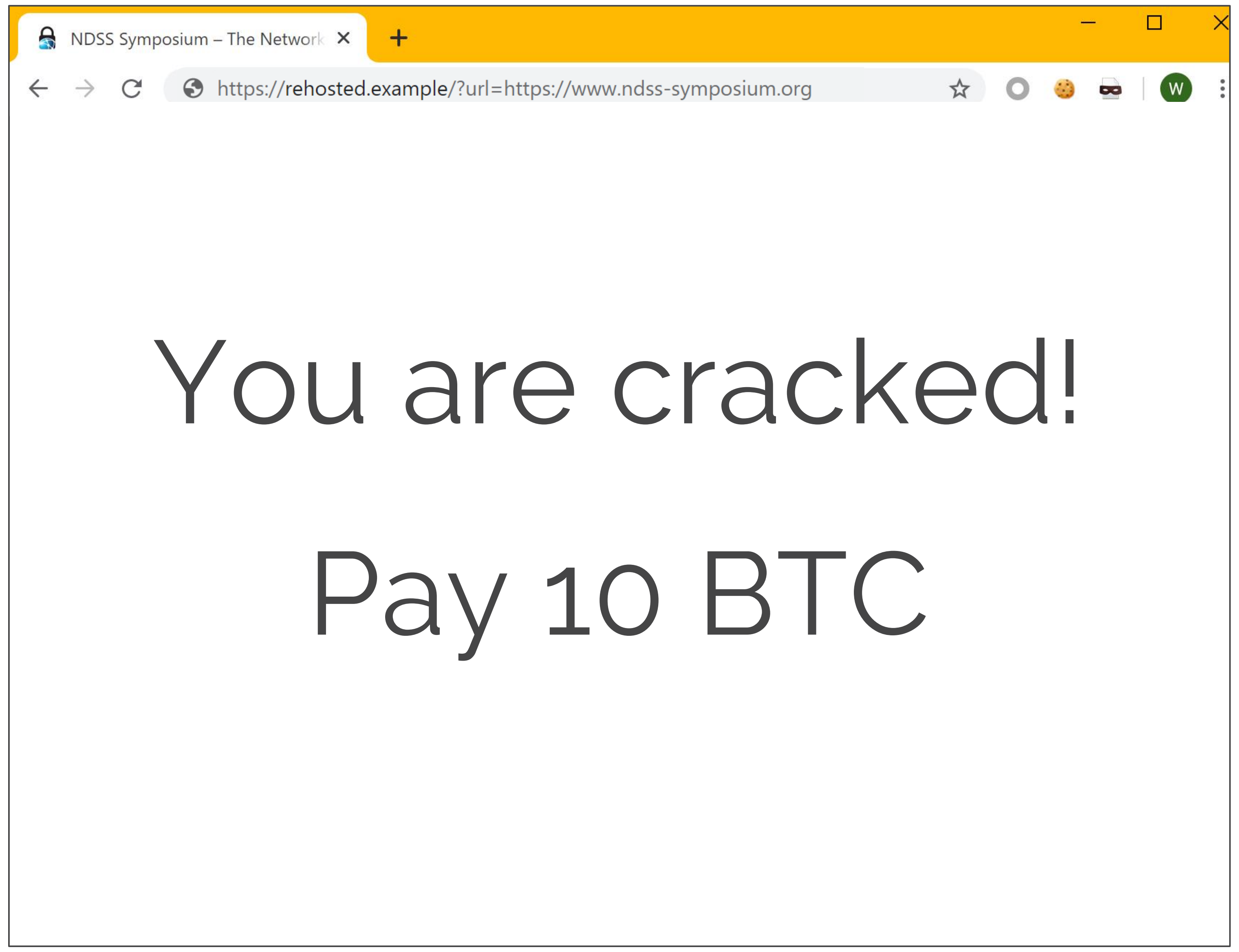

NDSS Symposium – The Network
https://rehosted.example/?url=https://www.ndss-symposium.org
NDSS SYMPOSIUM
About NDSS
Sponsorship
Previous Events
NDSS 2020

NDSS Symposium 2020

February 23-26, 2020
San Diego, CA

Registration is now open, register today! Check out the list of accepted papers here.

Through web rehosting


NDSS Symposium – The Network
https://rehosted.example/?url=https://www.ndss-symposium.org

You are cracked!

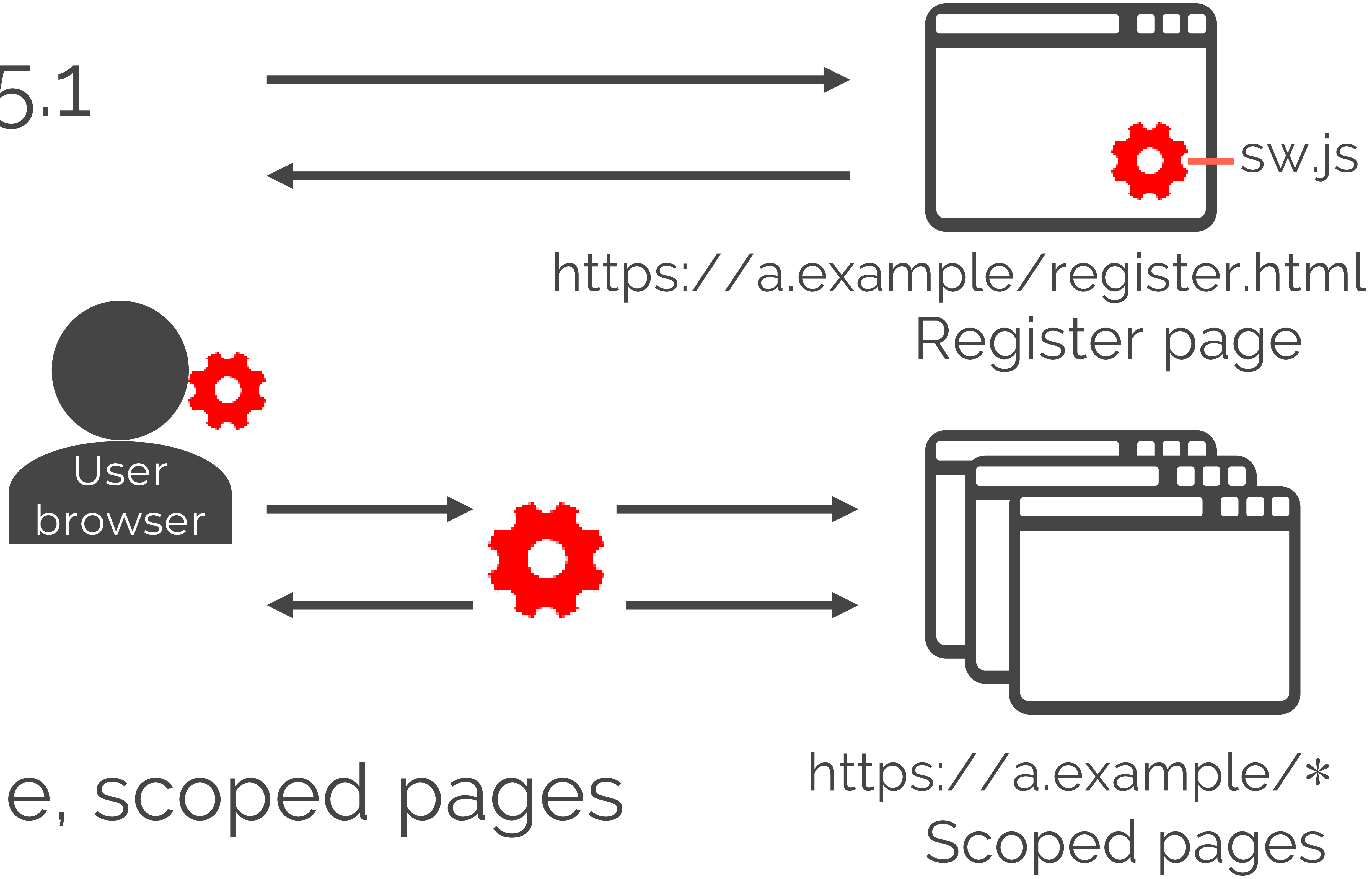
Pay 10 BTC

Through web rehosting

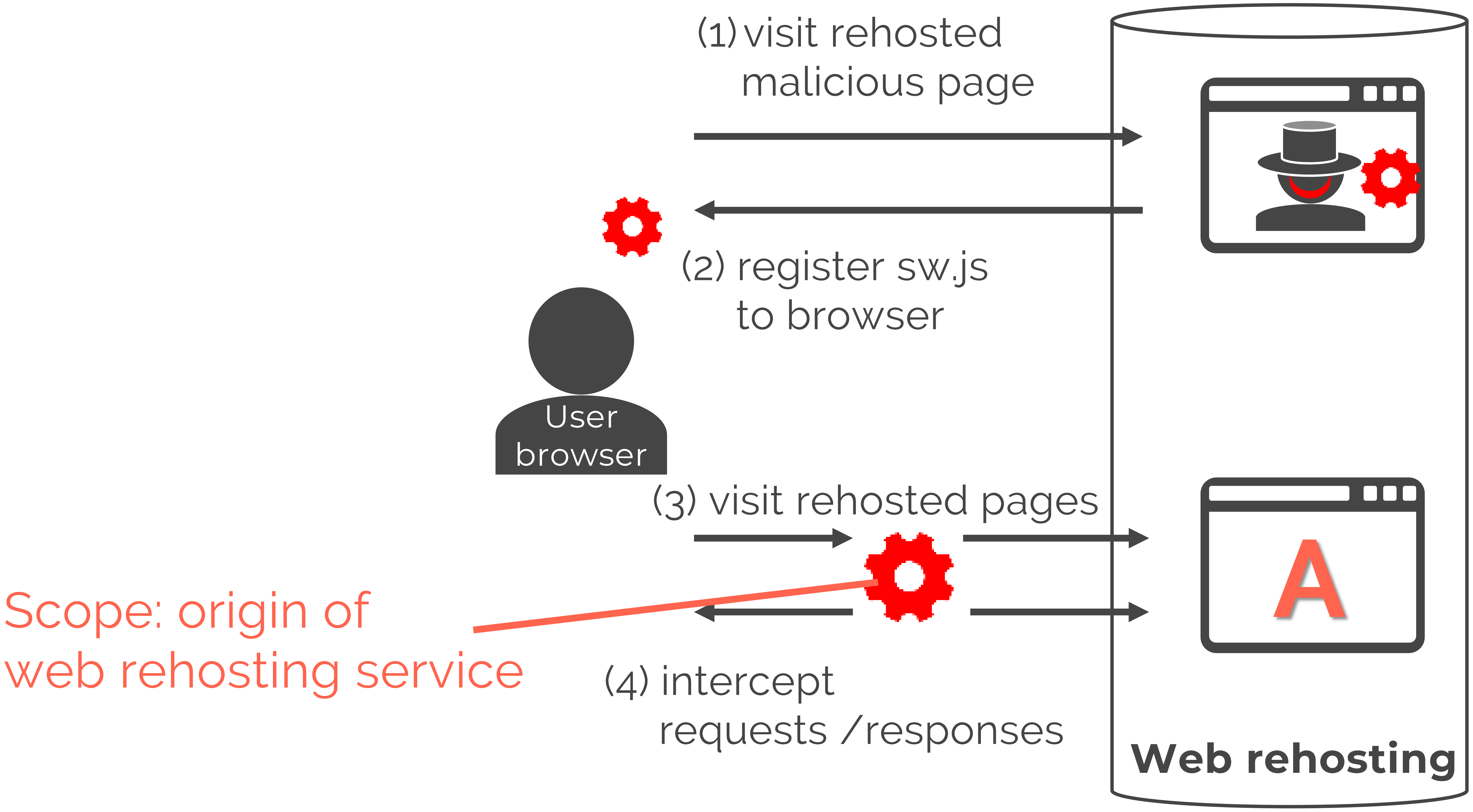
(after attack)

Service Worker (SW)

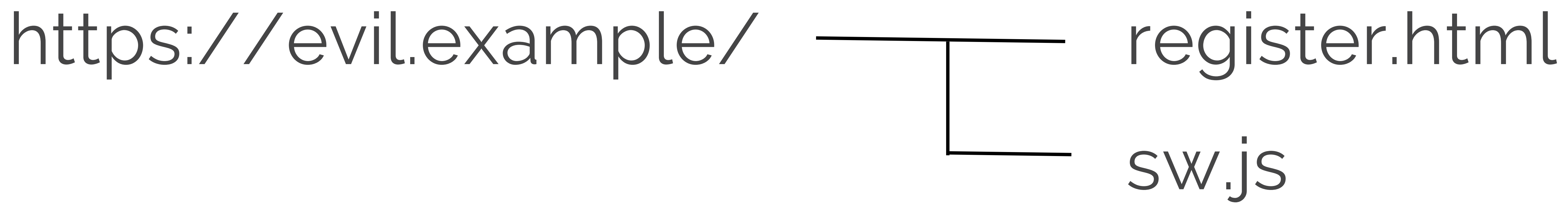
- Powerful feature in HTML 5.1
  - intercept all req./res.
- Restrictions
  - HTTPS
  - Same Origin
    - SW script, register page, scoped pages
  - MIME Type (JavaScript)



Attack I: Persistent MITM (with SW)


(1) visit rehosted malicious page
(2) register sw.js to browser
User browser
(3) visit rehosted pages
A
Scope: origin of web rehosting service
(4) intercept requests /responses
Web rehosting

SW Script and Register Page



https://evil.example/

register.html

sw.js

```
self.addEventListener('fetch', function(event) {
    customizeResponse(fetch(event.request));
    return;
});
```

sw.js

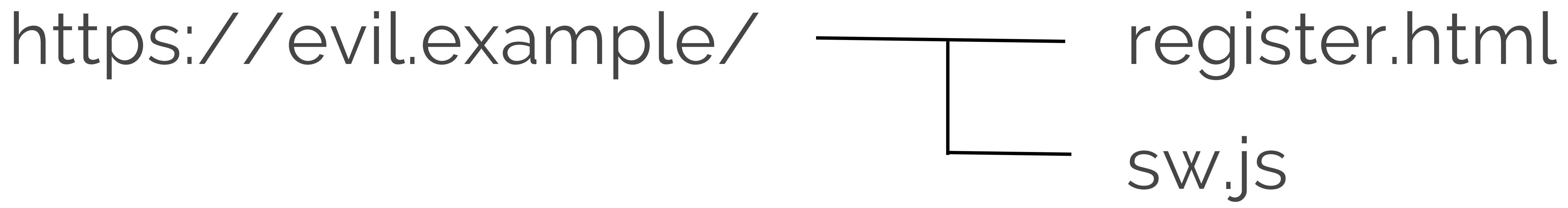
```
<script>
navigator.serviceWorker.register('sw.js');
</script>
```

register.html

generate rehosted malicious page:

“*https://rehosted.example/?url=https://evil.example/register.html*”

SW Script and Register Page



https://evil.example/

register.html

sw.js

```
self.addEventListener('fetch', function(event) {
     customizeResponse(fetch(event.request));
     return;
});
```

sw.js

```
<script>
navigator.serviceWorker.register('sw.js');
</script>
```

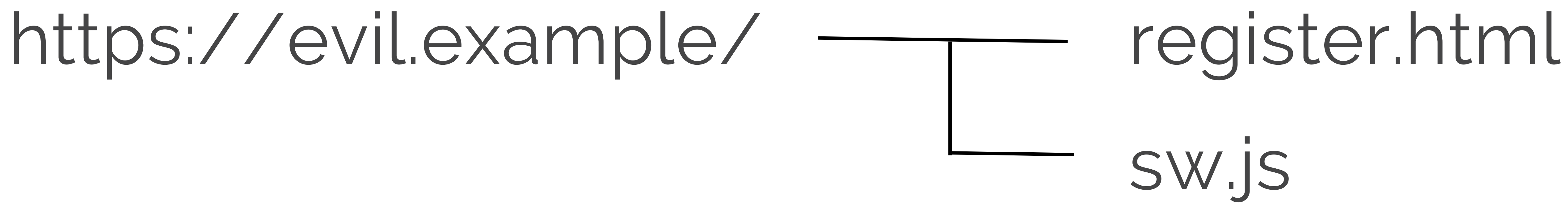
register.html

https://rehosted.example/sw.js (404)

Does not work

“https://rehosted.example/?url=https://evil.example/register.html”

SW Script and Register Page



https://evil.example/

register.html

sw.js

```
self.addEventListener('fetch', function(event) {
    customizeResponse(fetch(event.request));
    return;
});
```

sw.js

```
<script>
navigator.serviceWorker.register('https://rehosted.example/
?url=https://evil.example/sw.js');
</script>
```

register.html

Works

"https://rehosted.example/?url=https://evil.example/register.html"

Interesting Case Study for Google Translate

URL for website translation (type of web rehosting):

```
https:// translate.googleusercontent.com/translate_c?u=https://a.example&...
```

⬆ SW attack works

Interesting Case Study for Google Translate

URL for website translation (type of web rehosting):

```
https:// translate.googleusercontent.com/translate_c?u=https://a.example&...
```

URL for uploaded document translation:

```
https://translate.googleusercontent.com/translate_f
```

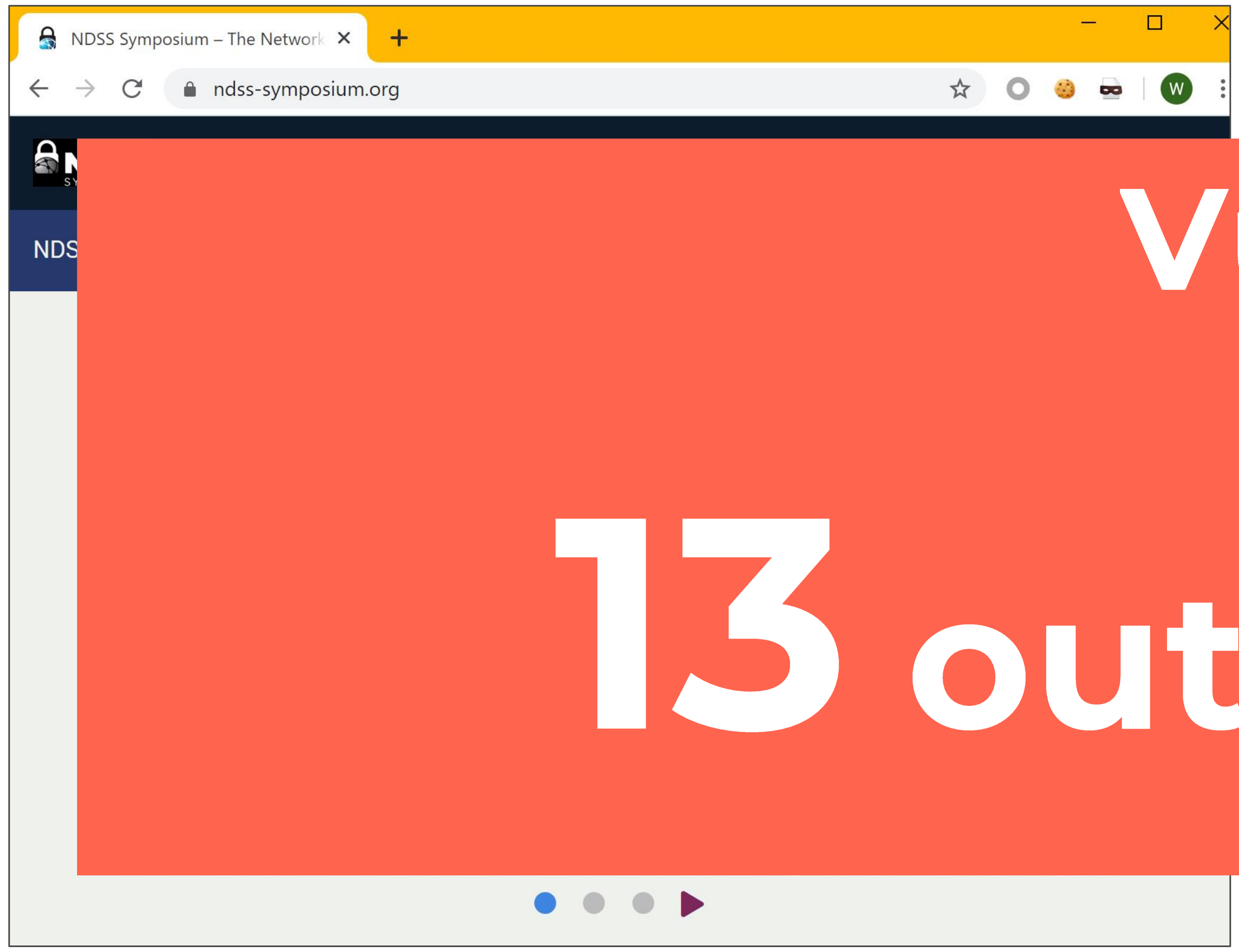
More details in our paper

- Techniques to rehost SW scripts on web translator
- Discussion of path scope
- Attack using AppCache instead of SW
    - Rewriting fallback pages + cookie bomb

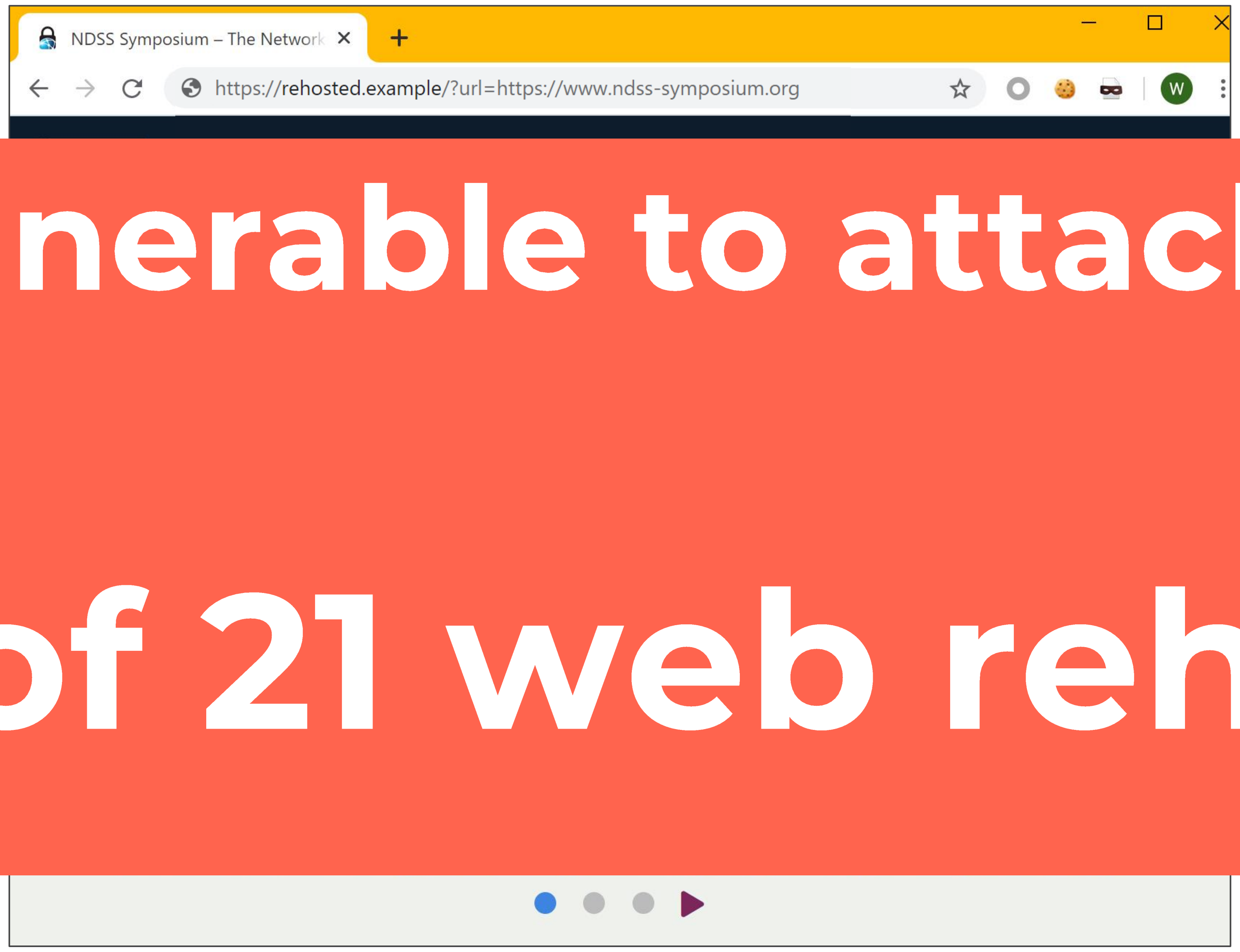
Attack I: Persistent MITM

Vulnerable to attack I:

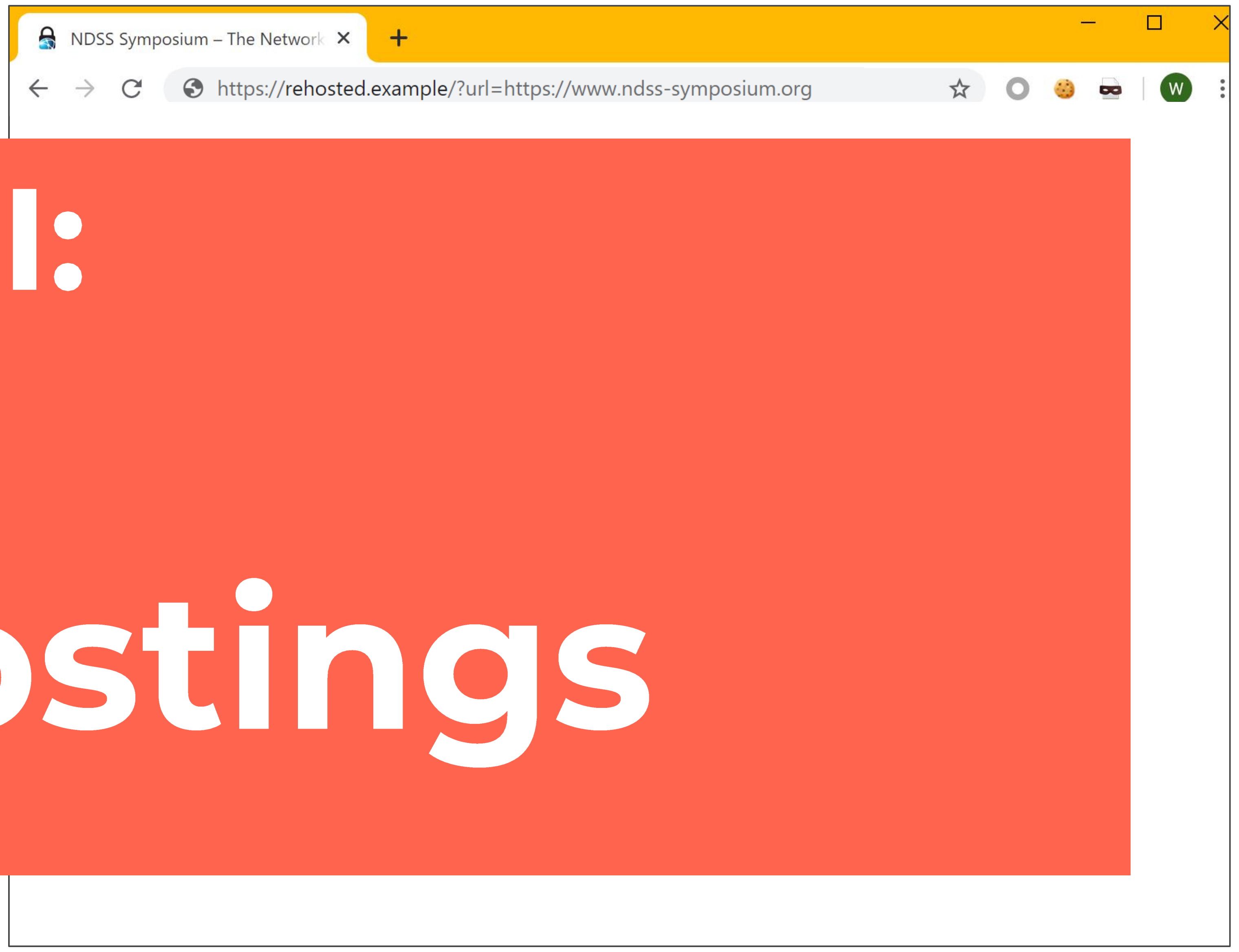
13 out of 21 web rehostings



Direct

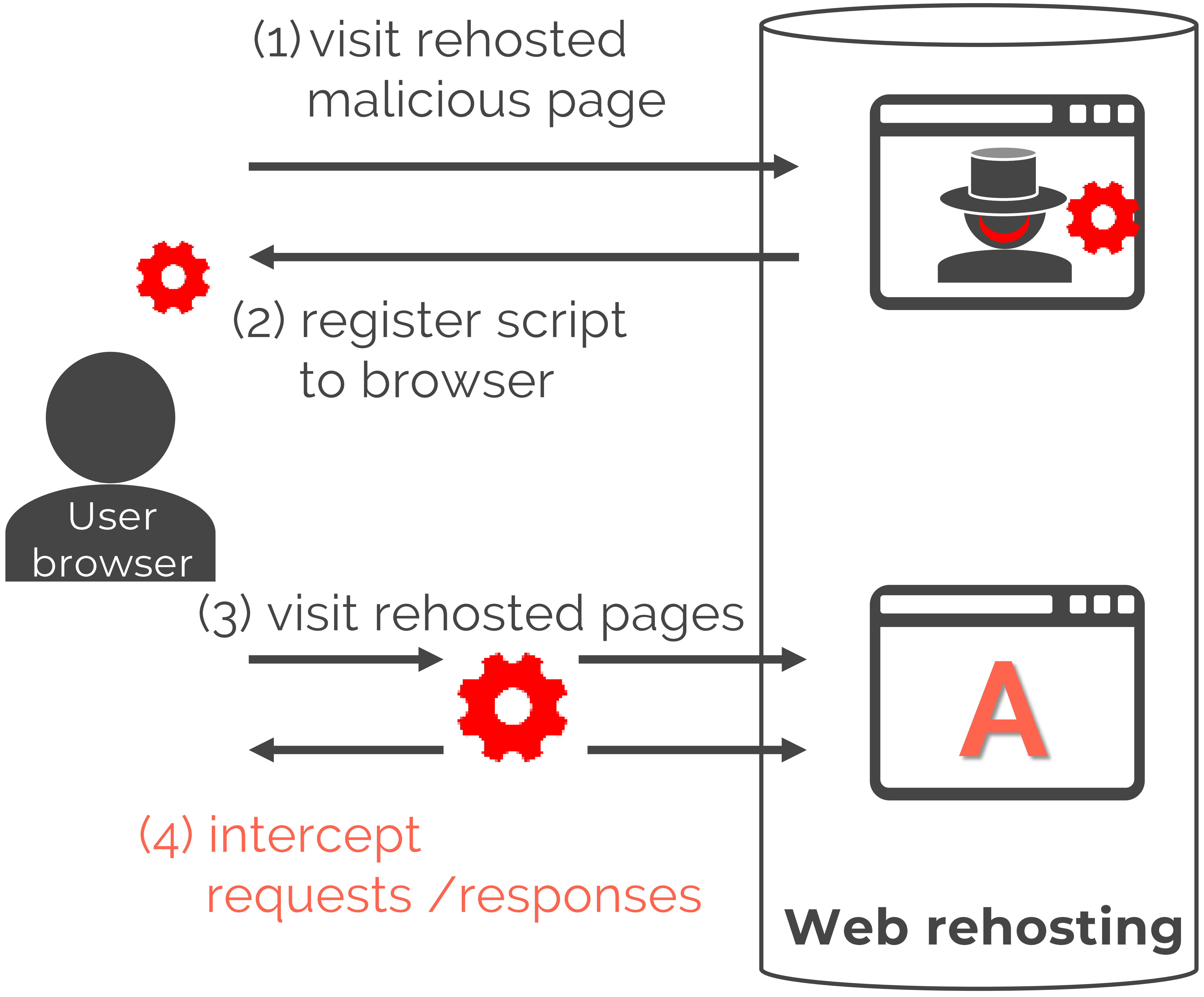

NDSS Symposium – The Network
https://rehosted.example/?url=https://www.ndss-symposium.org

Through web rehosting


NDSS Symposium – The Network
https://rehosted.example/?url=https://www.ndss-symposium.org

Through web rehosting (after attack)

Threat Model



(1) visit rehosted malicious page

(2) register script to browser

User browser

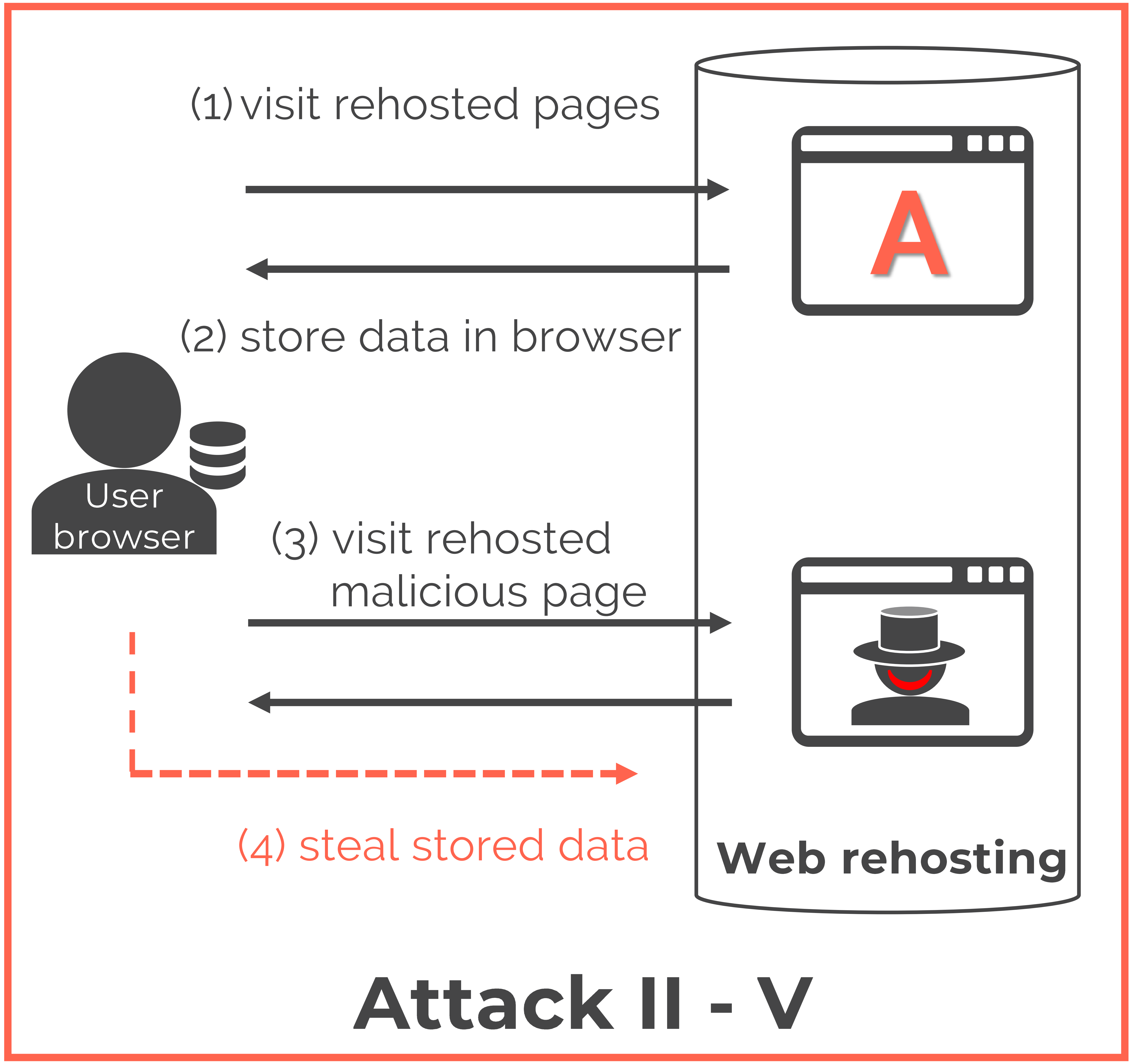
(3) visit rehosted pages

(4) intercept requests / responses

Web rehosting

A

Attack I



(1) visit rehosted pages

(2) store data in browser

User browser

(3) visit rehosted malicious page

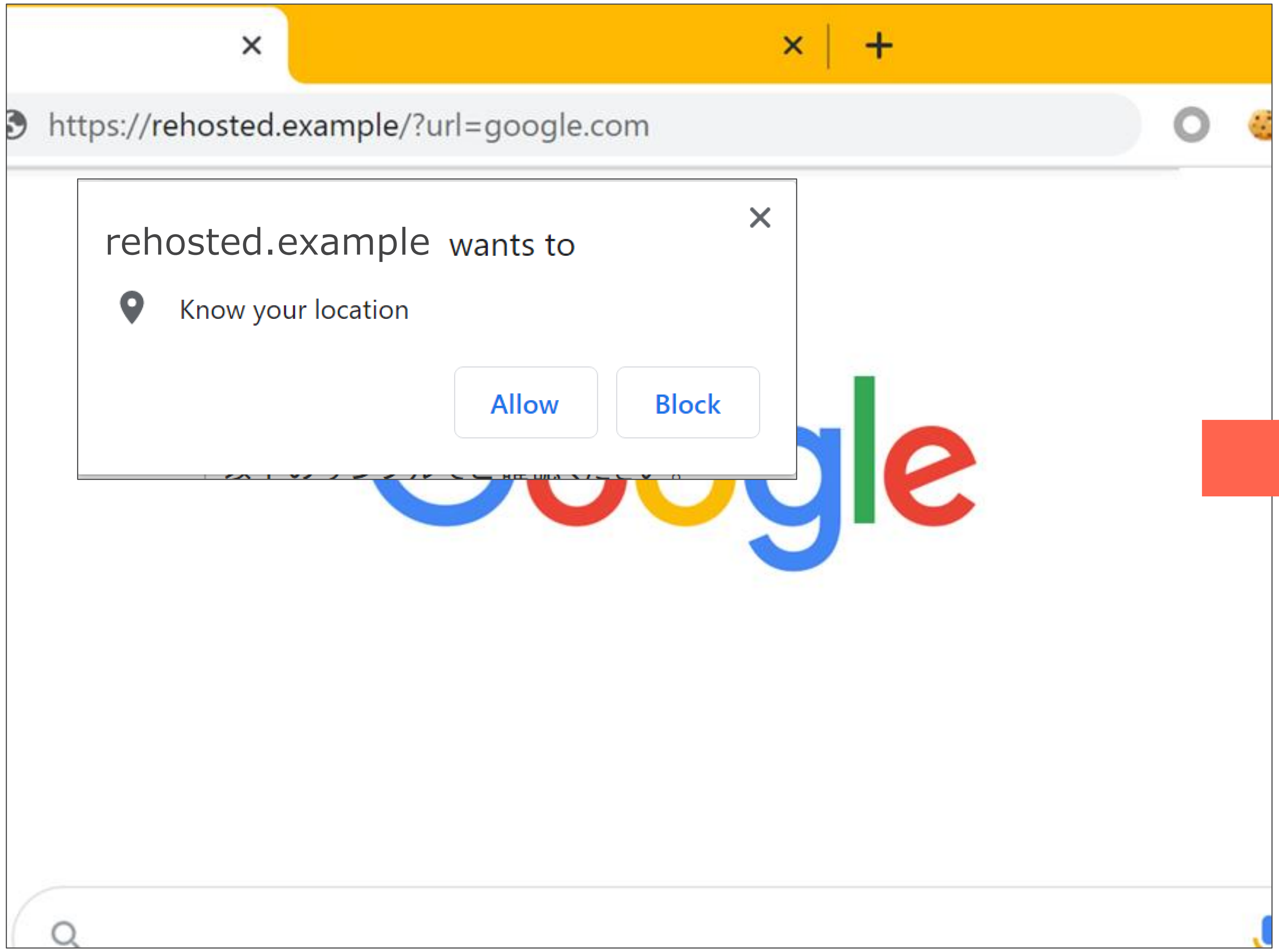
(4) steal stored data

Web rehosting

A

Attack II - V

Attack II: Privilege Abuse

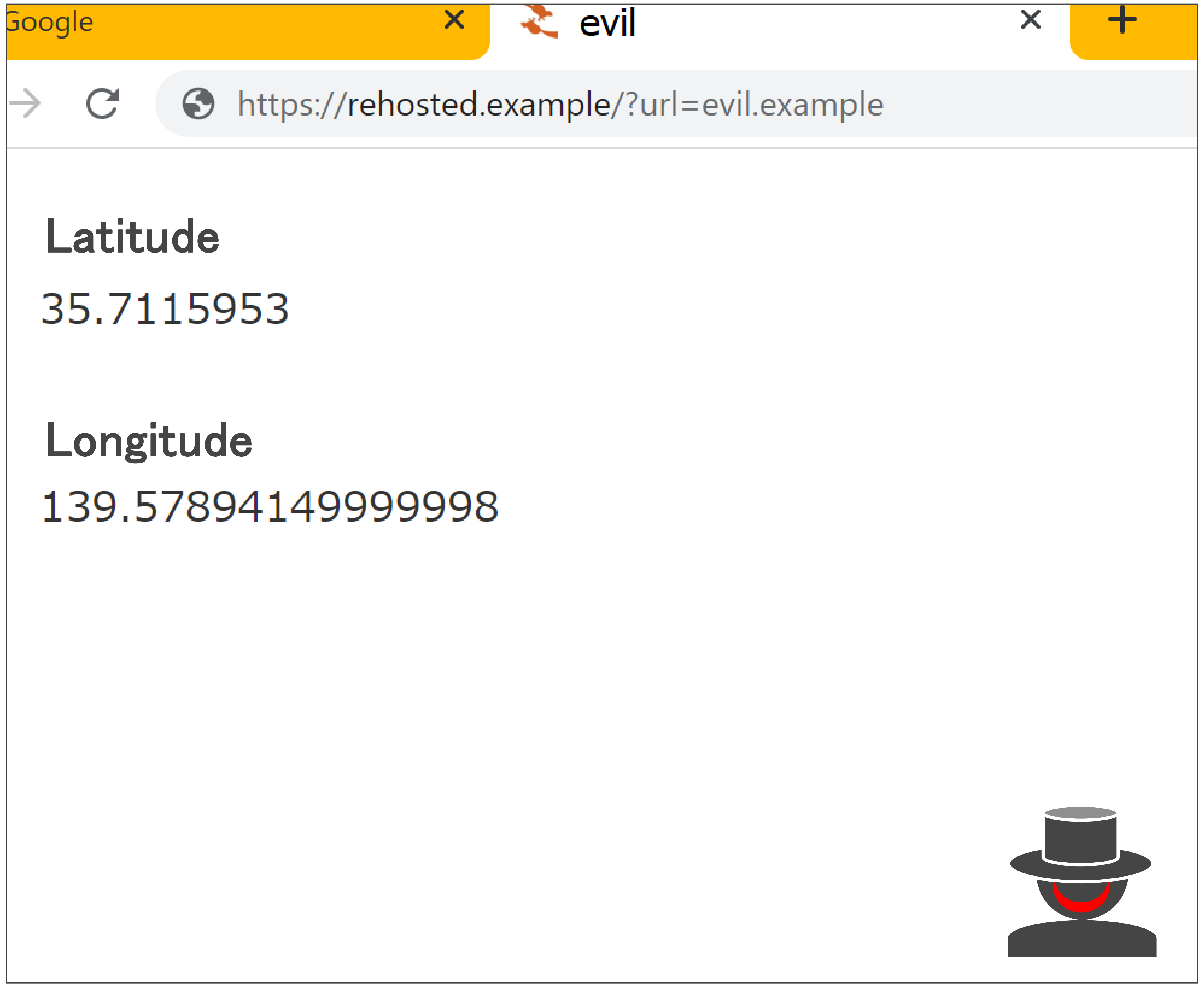


https://rehosted.example/?url=google.com

rehosted.example wants to

Know your location

User grant permission
at rehosted benign pages



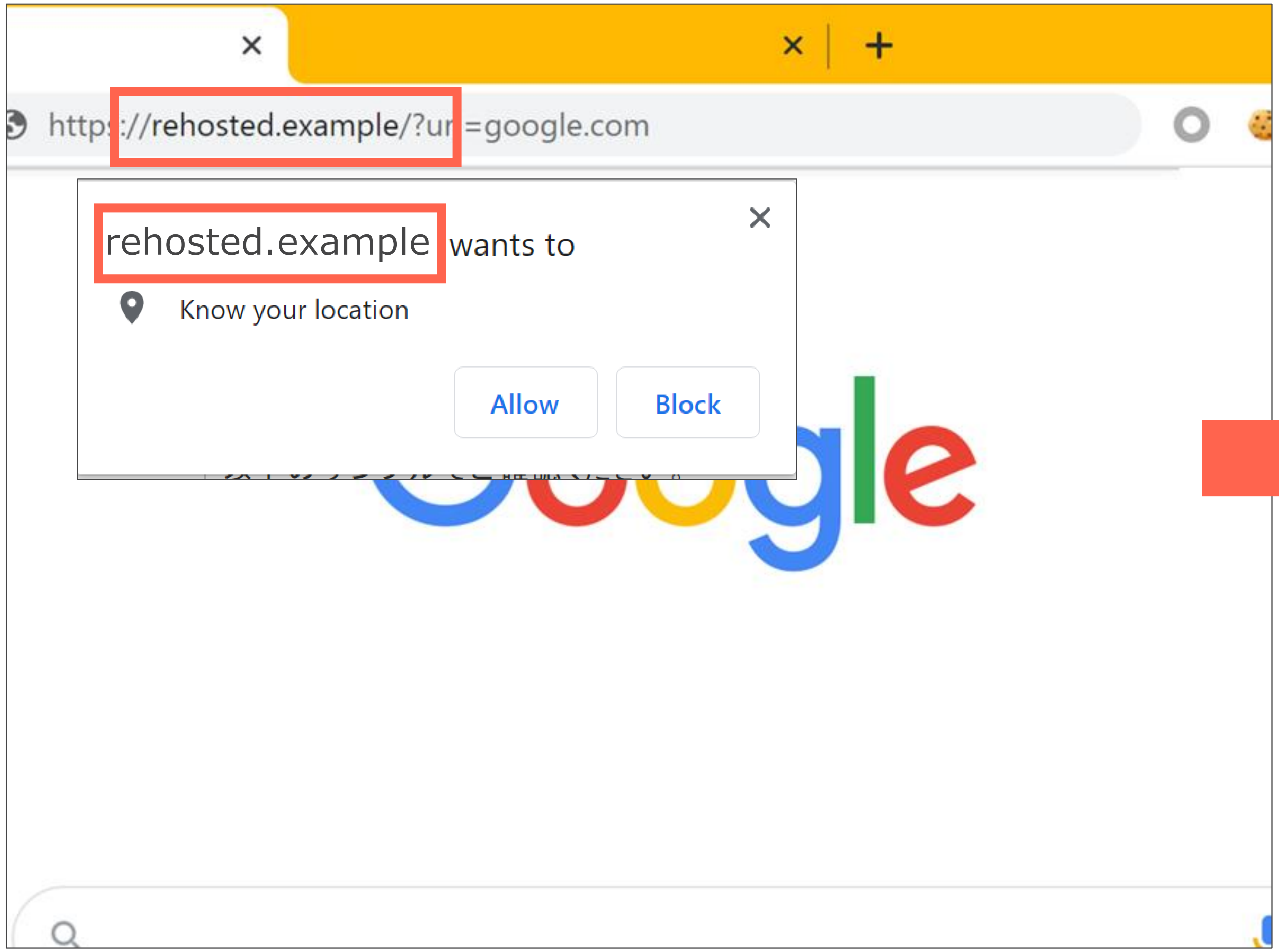
https://rehosted.example/?url=evil.example

Latitude
35.7115953

Longitude
139.57894149999998

Permission is reused by
rehosted **malicious** page

Attack II: Privilege Abuse

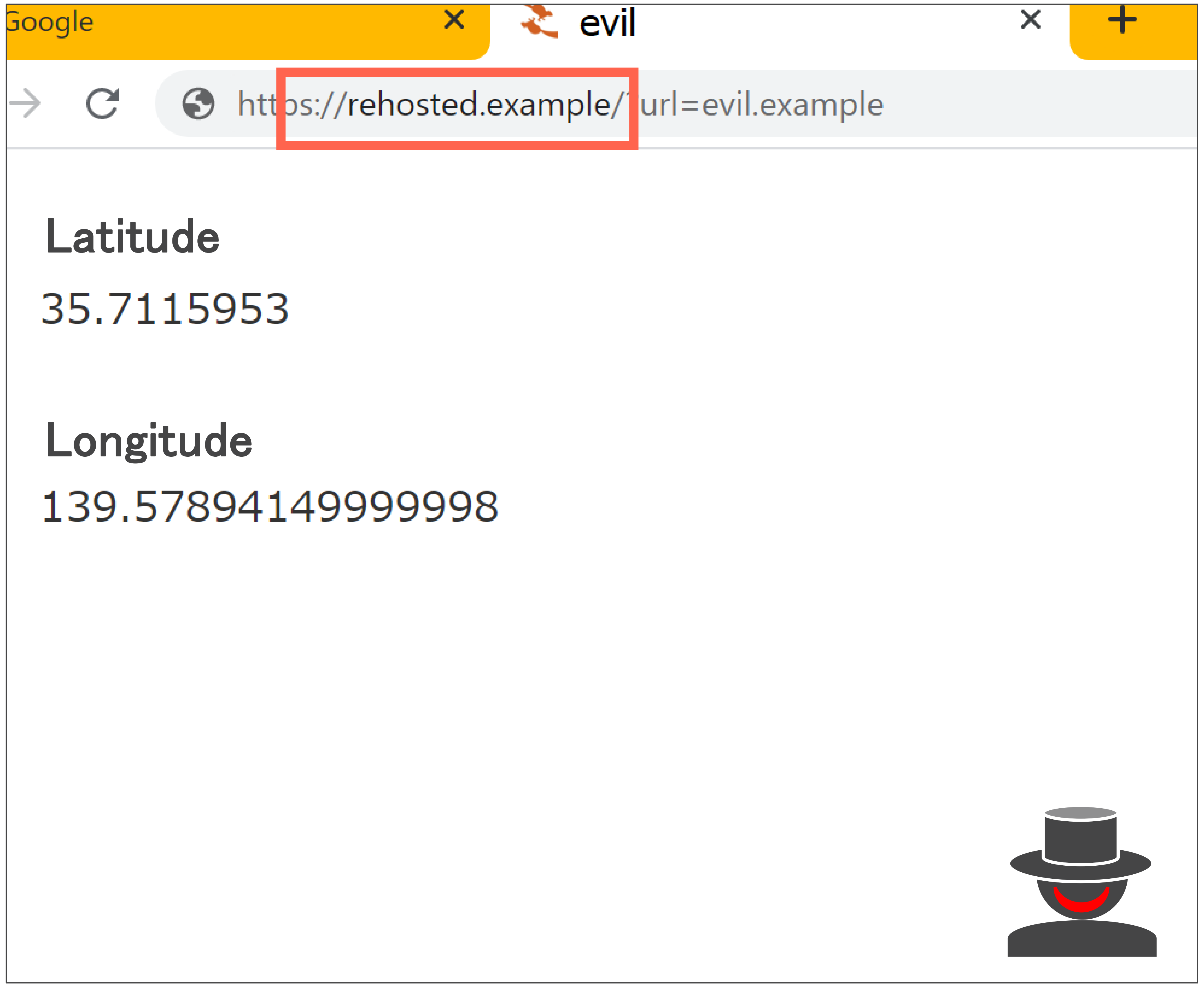


https://rehosted.example/?url=google.com

rehosted.example wants to

Know your location

User grant permission
at rehosted benign pages



https://rehosted.example/?url=evil.example

Latitude

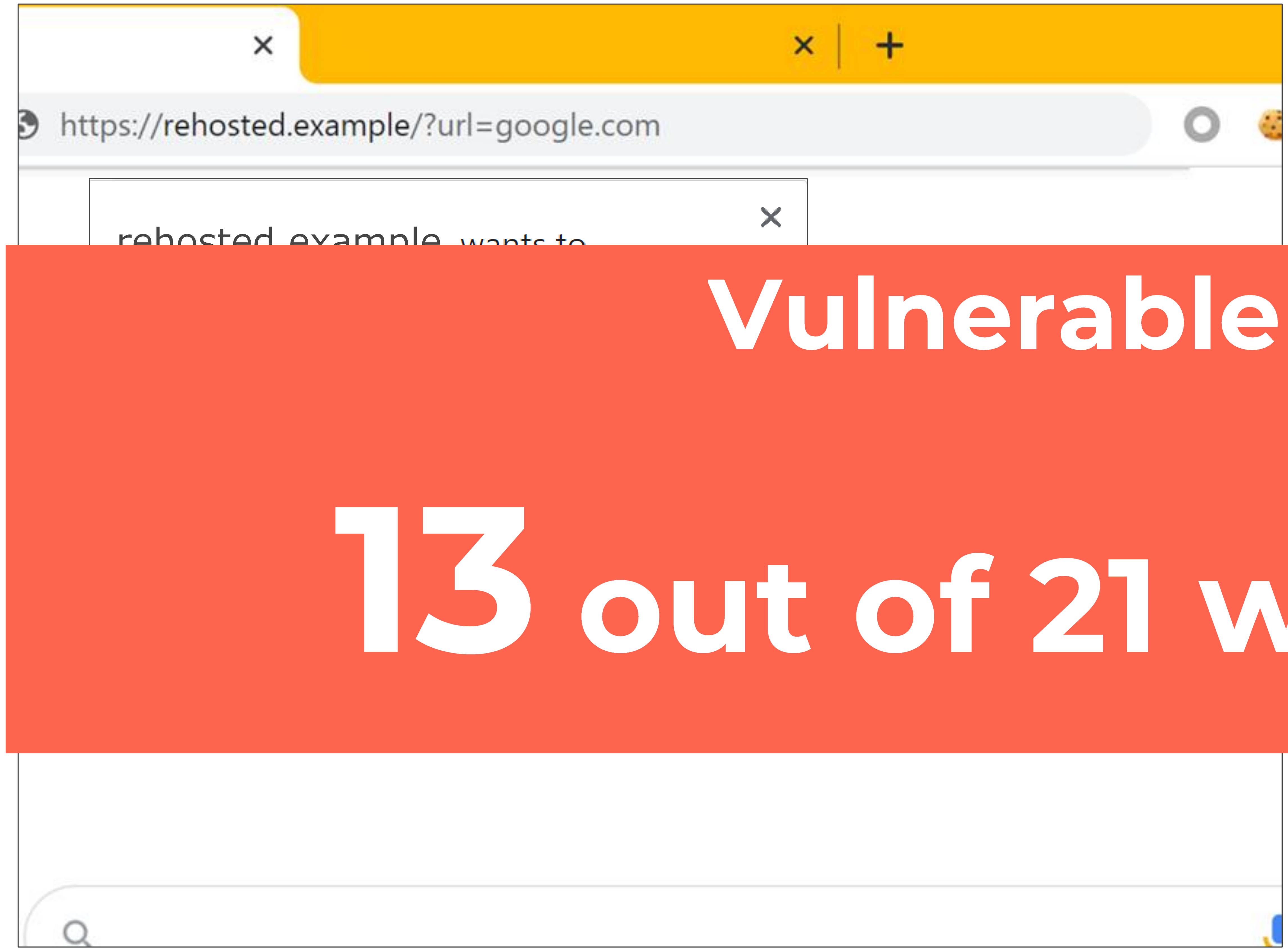
35.7115953

Longitude

139.57894149999998

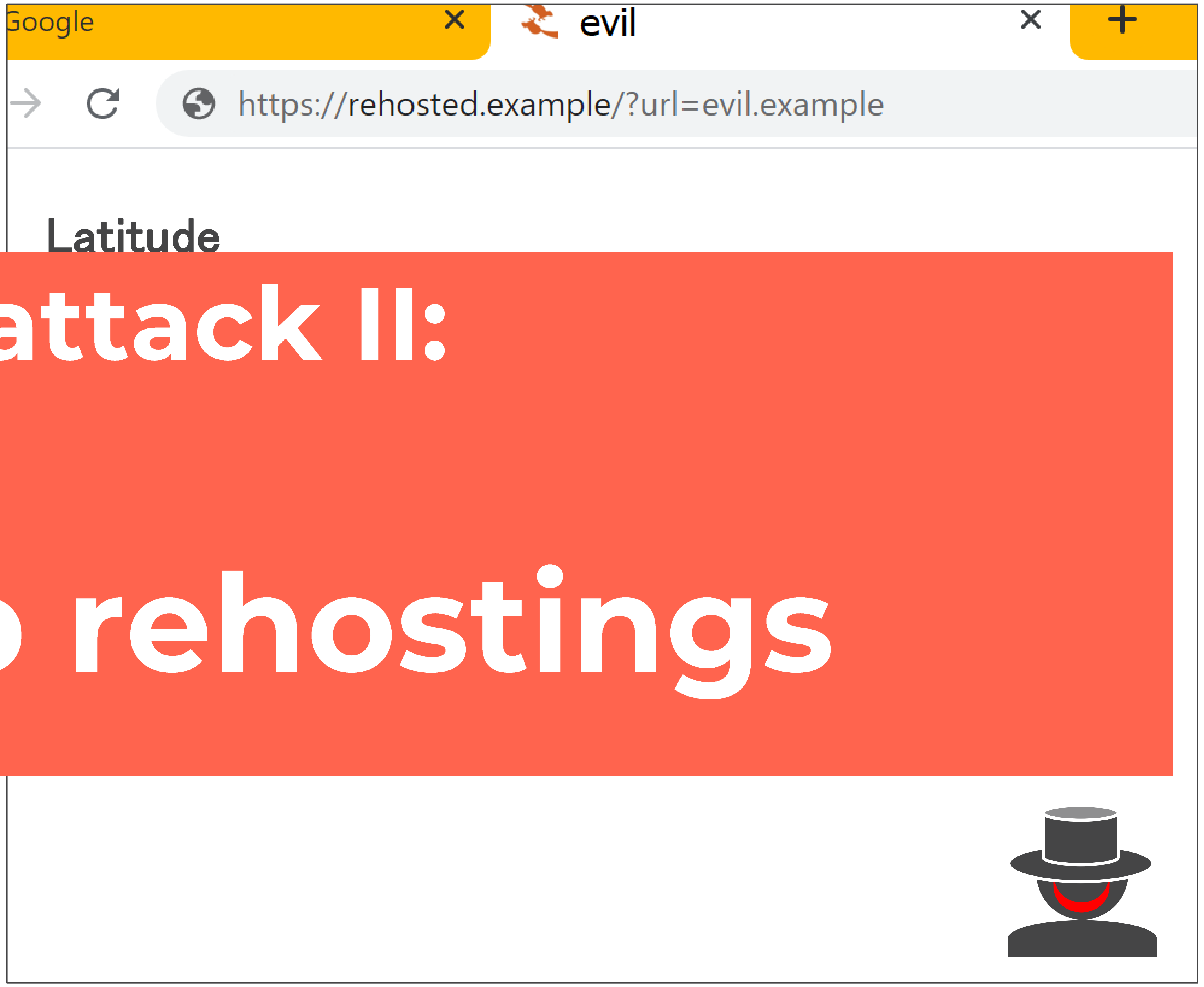
Permission is reused by
rehosted **malicious** page

Attack II: Privilege Abuse



https://rehosted.example/?url=google.com

rehosted.example wants to



https://rehosted.example/?url=evil.example

Latitude

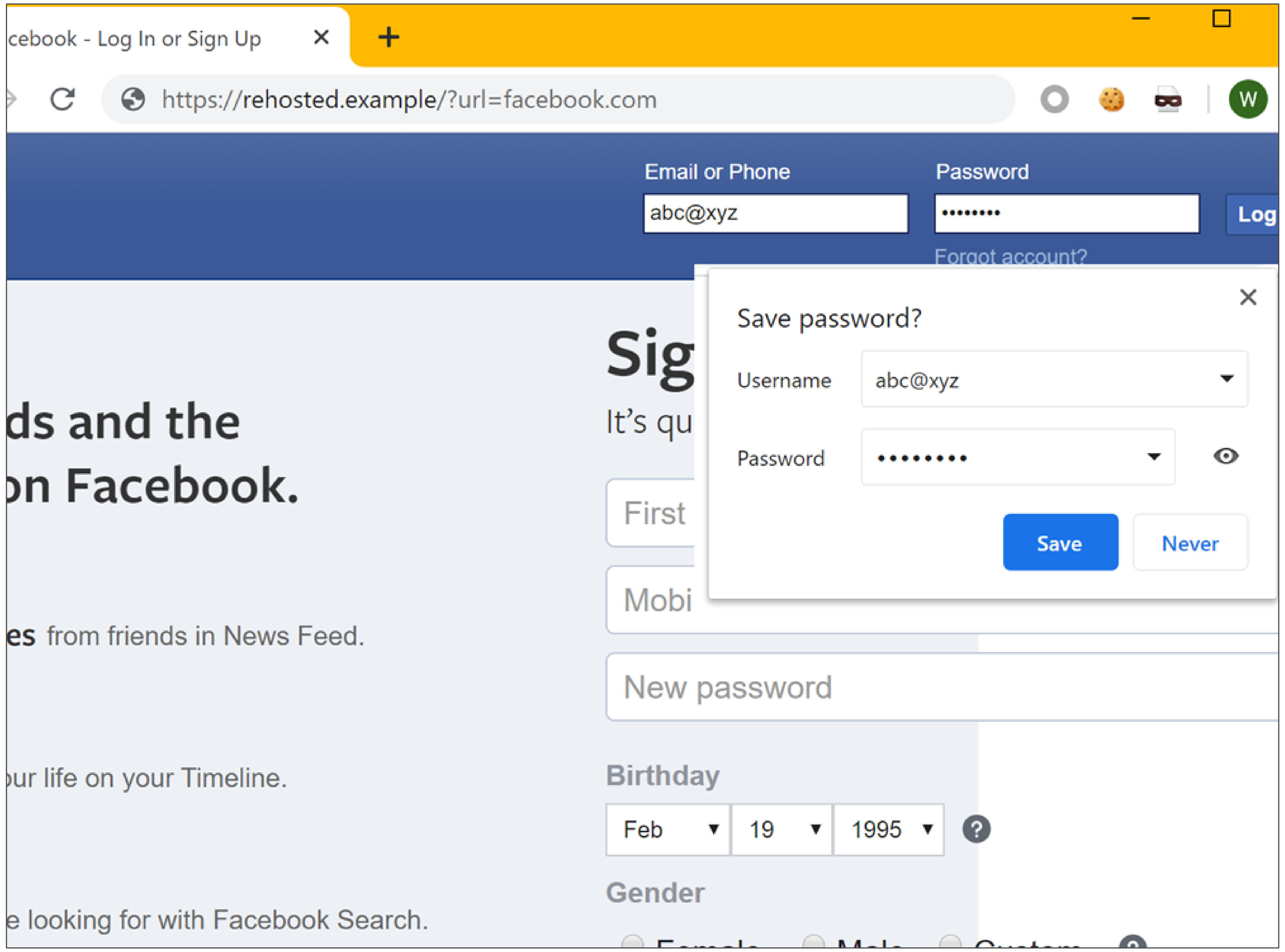
Vulnerable to attack II:

13 out of 21 web rehostings

User grant permission
at rehosted benign pages

Permission is reused by
rehosted **malicious** page

Attack III: Credential Theft (for Web-based Proxy)



cebook - Log In or Sign Up

https://rehosted.example/?url=facebook.com

Email or Phone: abc@xyz

Password: ••••••••

Forgot account?

Save password?

Username: abc@xyz

Password: ••••••••

ds and the
on Facebook.

es from friends in News Feed.

our life on your Timeline.

e looking for with Facebook Search.

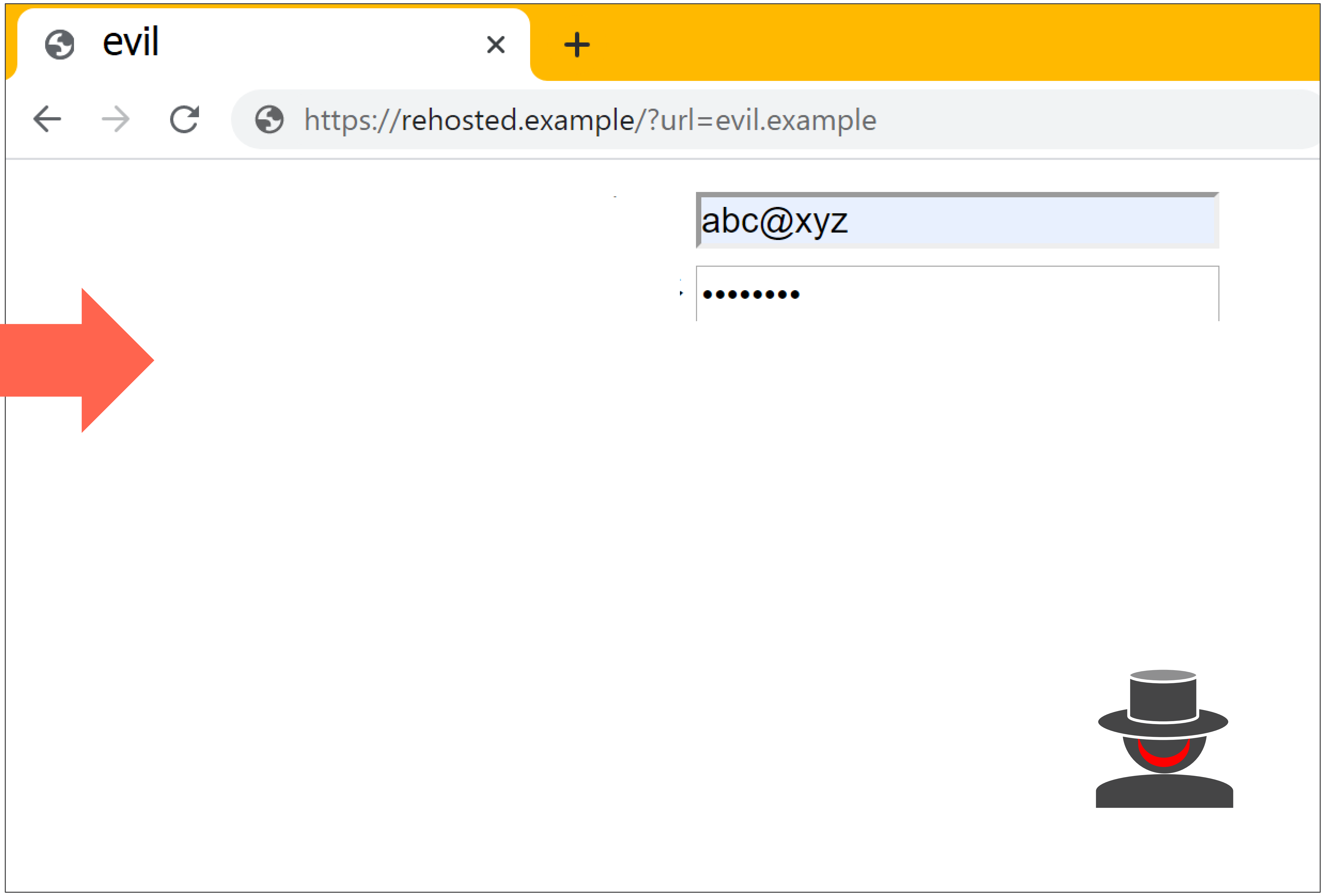
Sig

It's qu

Birthday: Feb 19 1995

Gender

User logs in to rehosted benign page and save credential in password manager



evil

https://rehosted.example/?url=evil.example

abc@xyz

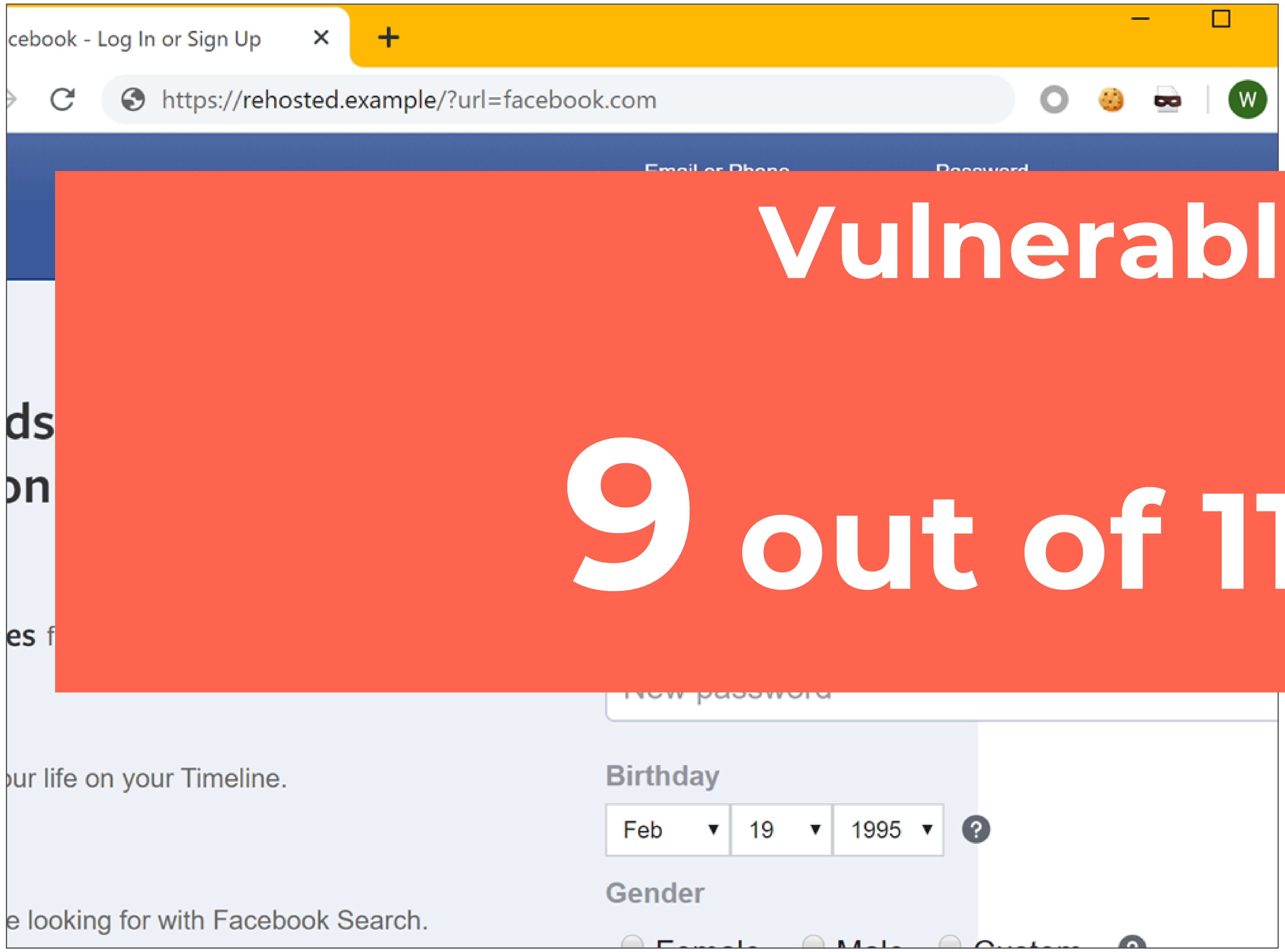
••••••••

Password manager auto-fills credential on fake form of rehosted **malicious** page

Attack III: Credential Theft (for Web-based Proxy)

Vulnerable to attack III:

9 out of 11 web proxies



cebook - Log In or Sign Up

https://rehosted.example/?url=facebook.com

Email or Phone

Password

ds

on

es f

New password

our life on your Timeline.

Birthday

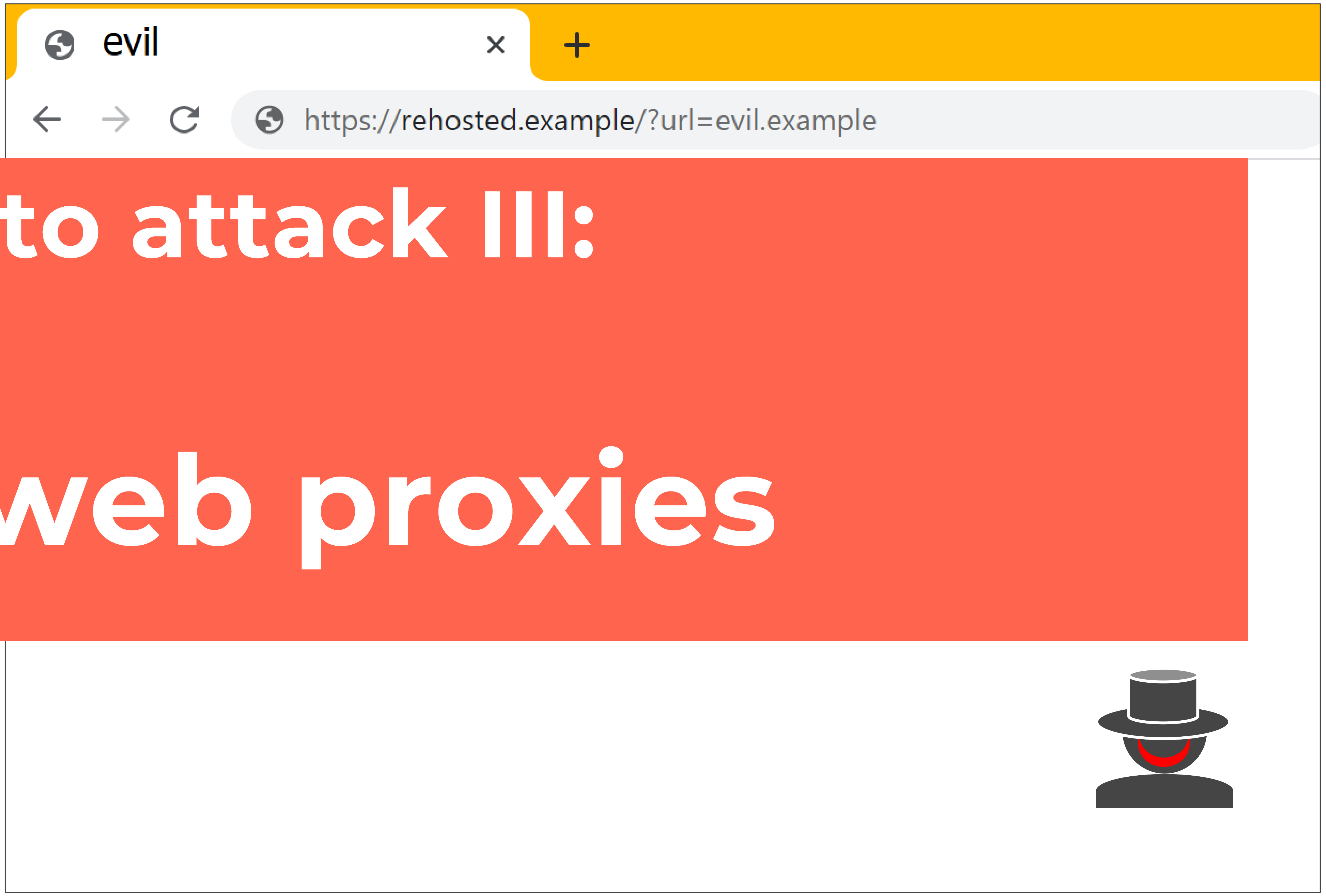
Feb ▾ 19 ▾ 1995 ▾ ?

Gender

e looking for with Facebook Search.

○ Female ○ Male ○ Custom ?

User logs in to rehosted benign page and save credential in password manager



evil

https://rehosted.example/?url=evil.example

Password manager auto-fills credential on fake form of rehosted **malicious** page

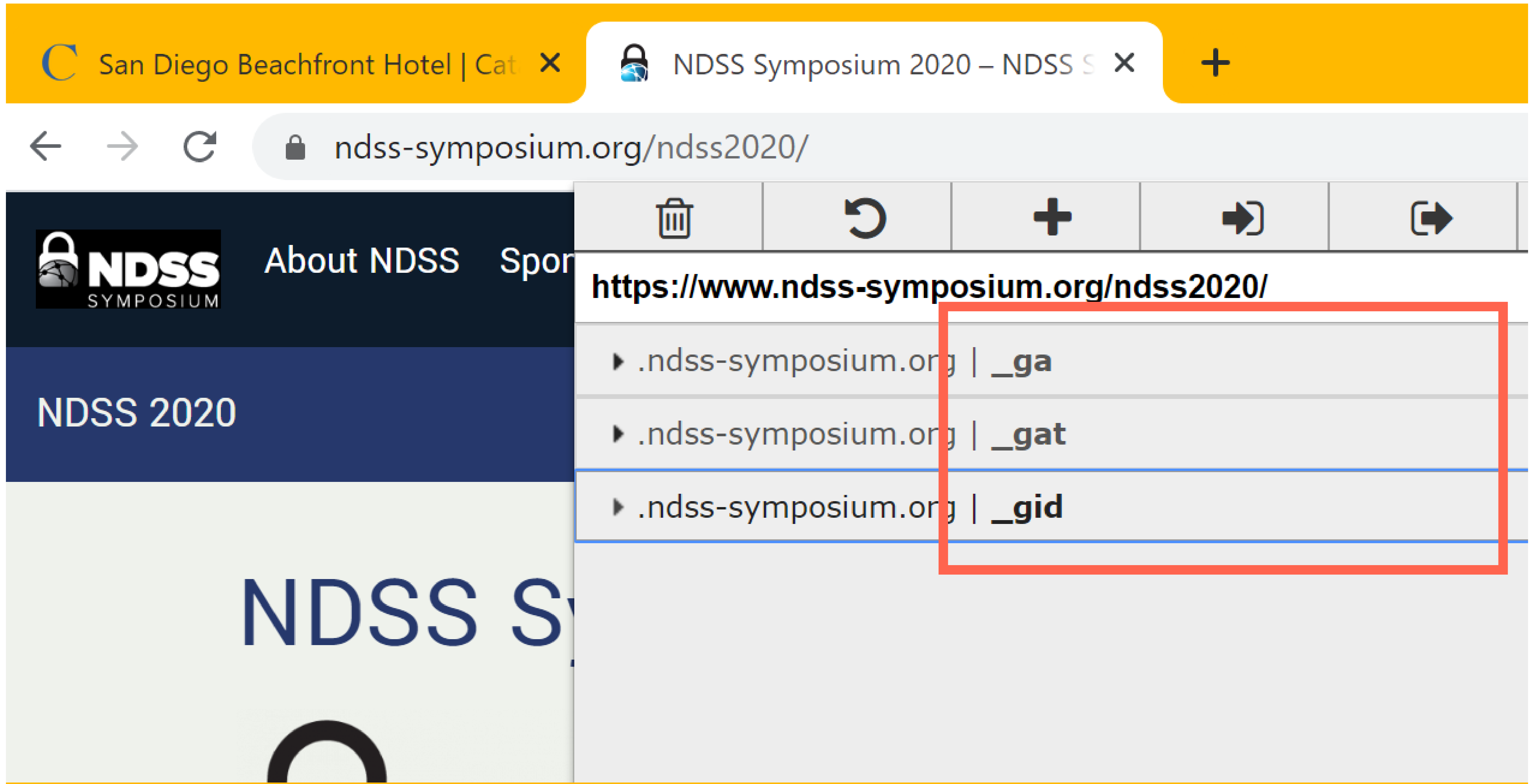
Attack IV: History Theft

1. User visits rehosted page.
2. Page writes cookie or localStorage by using JavaScript.

```
document.cookie = “name=value”;
```

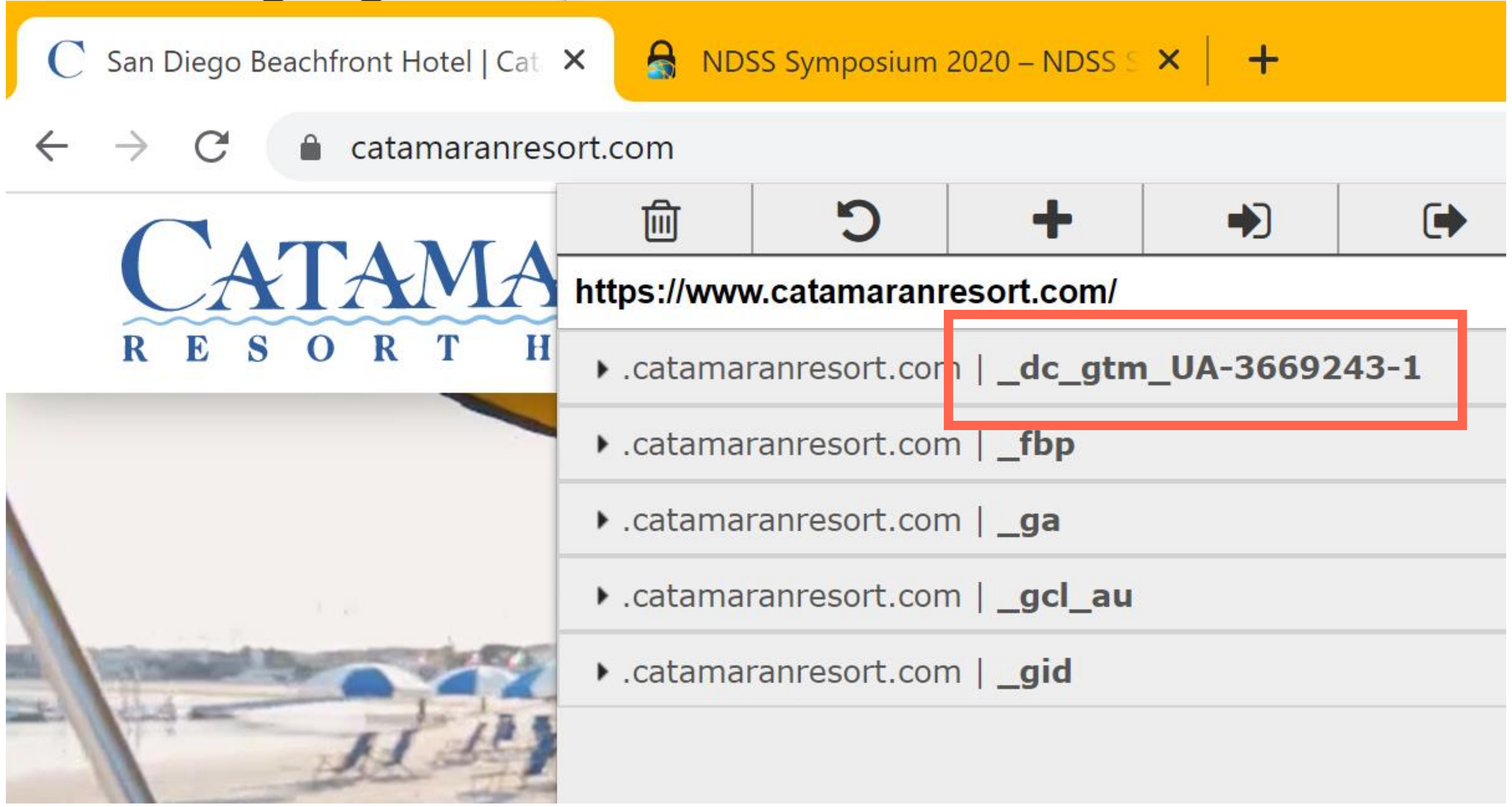
```
localStorage.setItem(‘name', value');
```

3. Rehosted malicious page retrieves cookie/localStorage.
4. Attacker estimates browsing history by using retrieved data.


San Diego Beachfront Hotel | Cat
NDSS Symposium 2020 – NDSS S
ndss-symposium.org/ndss2020/
About NDSS
Spon
NDSS 2020
NDSS Sy

https://www.ndss-symposium.org/ndss2020/

- .ndss-symposium.org | **_ga**
- .ndss-symposium.org | **_gat**
- .ndss-symposium.org | **_gid**


San Diego Beachfront Hotel | Cat
NDSS Symposium 2020 – NDSS S
catamaranresort.com
CATAMA
RESORT H

https://www.catamaranresort.com/

- .catamaranresort.com | **_dc_gtm_UA-3669243-1**
- .catamaranresort.com | **_fbp**
- .catamaranresort.com | **_ga**
- .catamaranresort.com | **_gcl_au**
- .catamaranresort.com | **_gid**

Non-identifiable website
(has only general cookie names
/localStorage keys)

Identifiable website
(has unique cookie name
/localStorage keys)

39.1% of alexa top 10k

Attack IV: History Theft

1. User visits rehosted page.
2. Page writes cookie or localStorage by using JavaScript

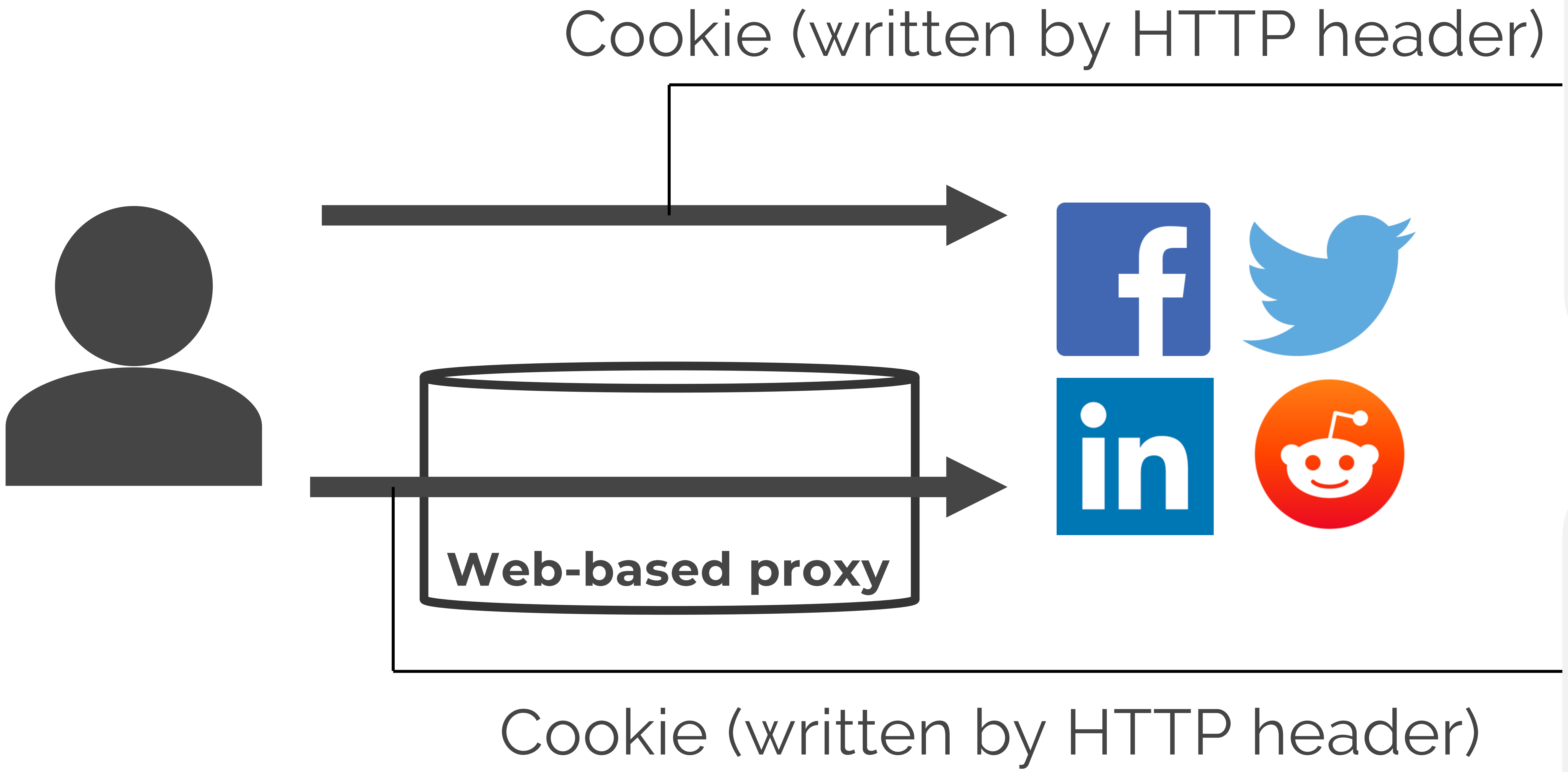
Vulnerable to attack IV:

18 out of 21 web rehostings

3.

4. Attacker estimates browsing history.

Attack V: Session Hijacking & Injection (for Web-based Proxy)



Cookie (written by HTTP header)

Domain: .facebook.com

Name: xs

Value: XXXXXXXXXXXXXXXX

Option: HttpOnly

Cookie (written by HTTP header)

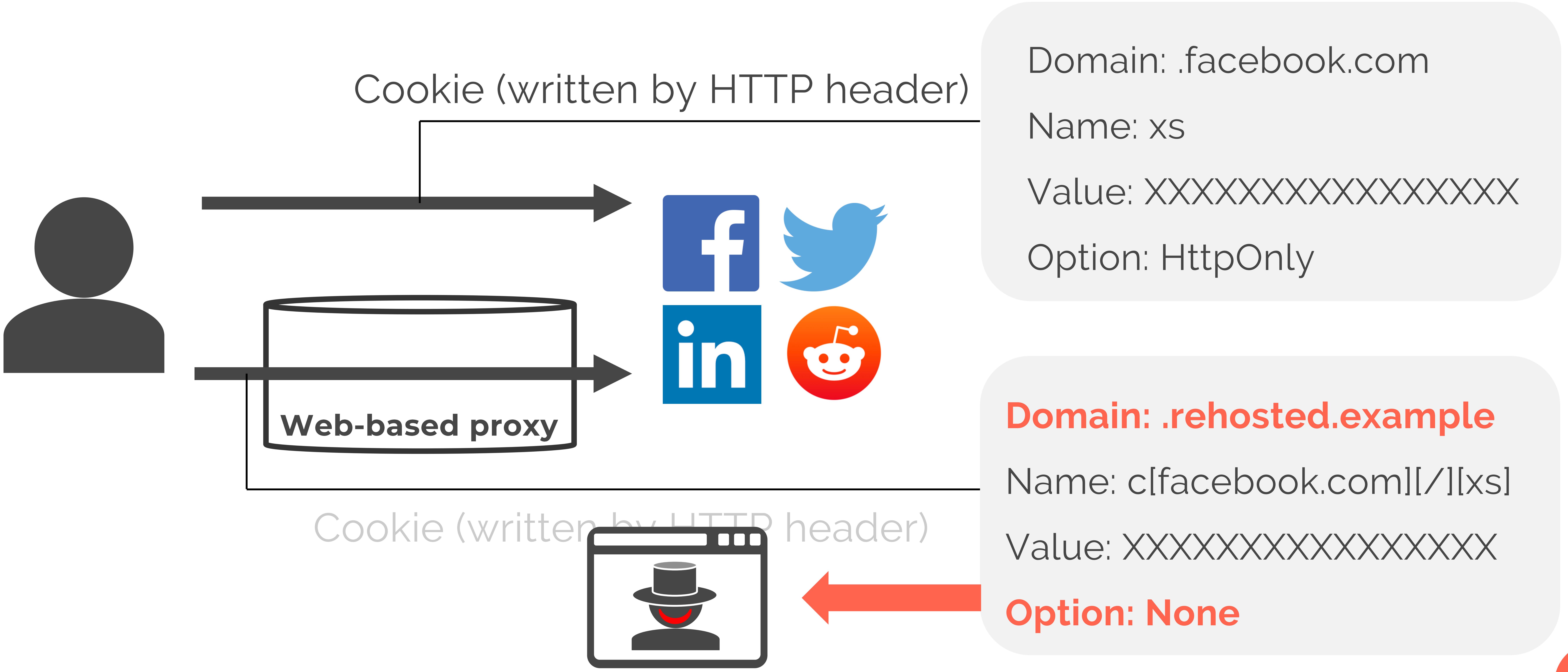
Domain: .rehosted.example

Name: c[facebook.com][/][xs]

Value: XXXXXXXXXXXXXXXX

Option: None

Attack V: Session Hijacking & Injection (for Web-based Proxy)



Cookie (written by HTTP header)

Domain: .facebook.com

Name: xs

Value: XXXXXXXXXXXXXXXX

Option: HttpOnly

Cookie (written by HTTP header)

Domain: .rehosted.example

Name: c[facebook.com][/][xs]

Value: XXXXXXXXXXXXXXXX

Option: None

Attack V: Session Hijacking & Injection (for Web-based Proxy)

Domain: .facebook.com

Vulnerable to attack V:

8 out of 11 web proxies

Cookie (written by HTTP header)
Web-based proxy
Cookie (written by HTTP header)

Domain: .rehosted.example

Name: c[facebook.com][/][xs]

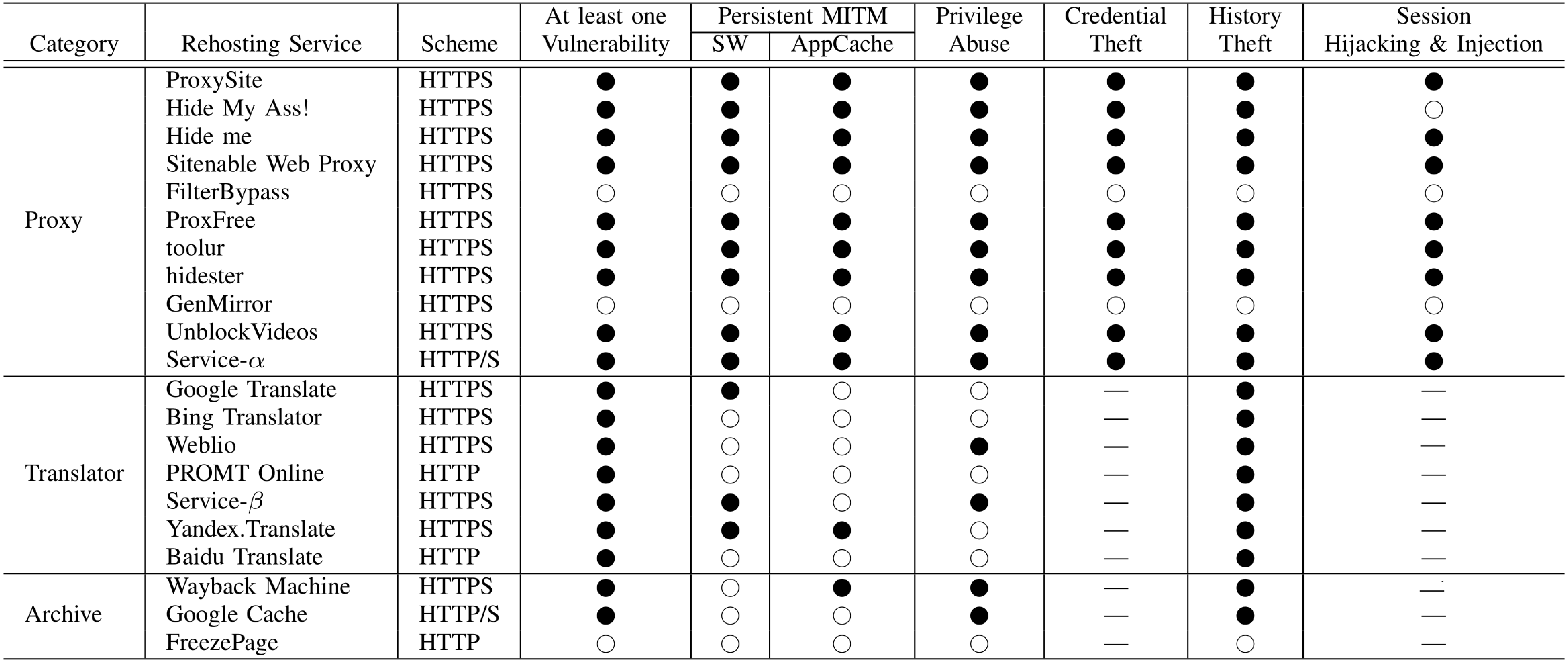
Value: XXXXXXXXXXXXXXXX

Option: None

Summary of Results

● Vulnerable

○ Secure

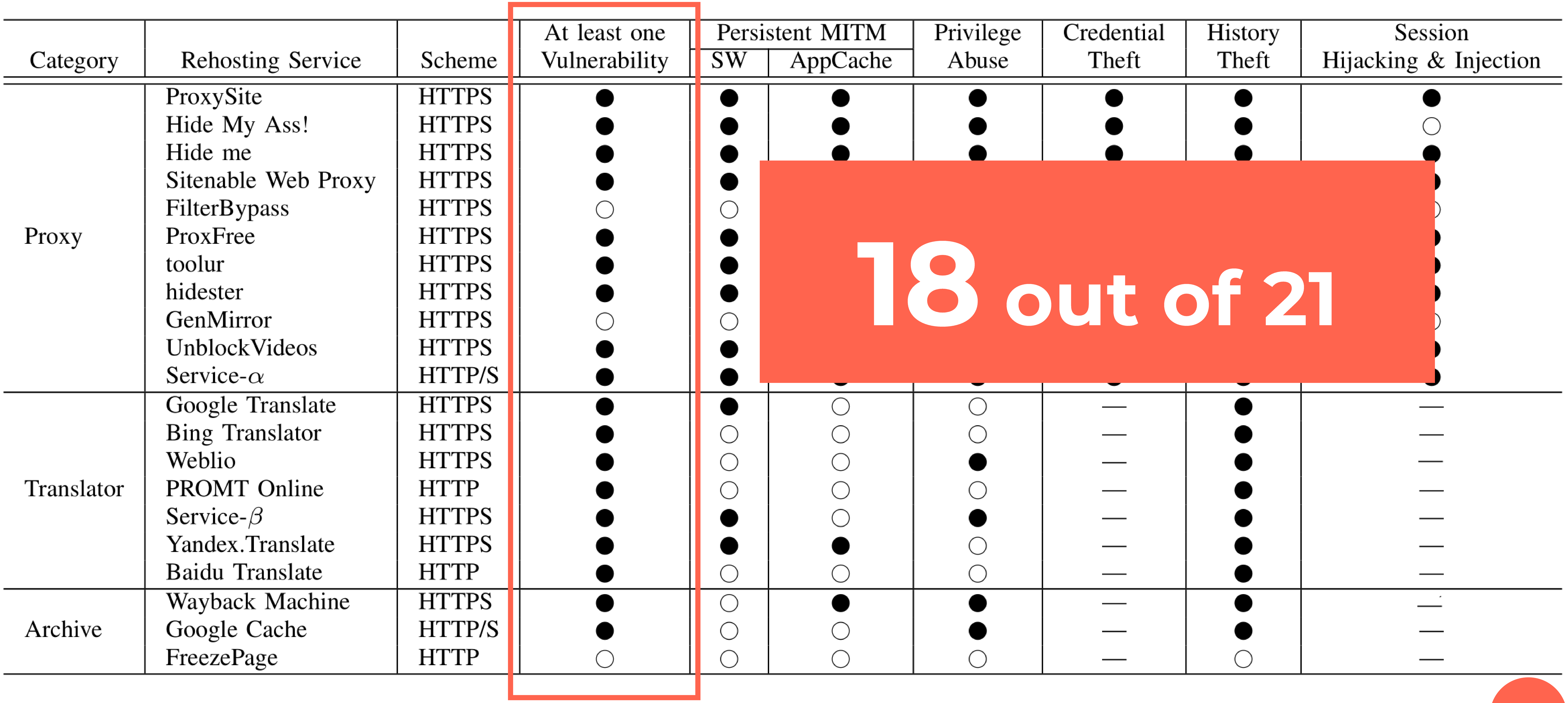


Category	Rehosting Service	Scheme	At least one Vulnerability	Persistent MITM		Privilege Abuse	Credential Theft	History Theft	Session Hijacking & Injection
				SW	AppCache				
Proxy	ProxySite	HTTPS	●	●	●	●	●	●	●
	Hide My Ass!	HTTPS	●	●	●	●	●	●	○
	Hide me	HTTPS	●	●	●	●	●	●	●
	Sitenable Web Proxy	HTTPS	●	●	●	●	●	●	●
	FilterBypass	HTTPS	○	○	○	○	○	○	○
	ProxFree	HTTPS	●	●	●	●	●	●	●
	toolur	HTTPS	●	●	●	●	●	●	●
	hidester	HTTPS	●	●	●	●	●	●	●
	GenMirror	HTTPS	○	○	○	○	○	○	○
	UnblockVideos	HTTPS	●	●	●	●	●	●	●
	Service-α	HTTP/S	●	●	●	●	●	●	●
Translator	Google Translate	HTTPS	●	●	○	○	—	●	—
	Bing Translator	HTTPS	●	○	○	○	—	●	—
	Weblio	HTTPS	●	○	○	●	—	●	—
	PROMT Online	HTTP	●	○	○	○	—	●	—
	Service-β	HTTPS	●	●	○	●	—	●	—
	Yandex.Translate	HTTPS	●	●	●	○	—	●	—
	Baidu Translate	HTTP	●	○	○	○	—	●	—
Archive	Wayback Machine	HTTPS	●	○	●	●	—	●	—
	Google Cache	HTTP/S	●	○	○	●	—	●	—
	FreezePage	HTTP	○	○	○	○	—	○	—

Summary of Results

● Vulnerable

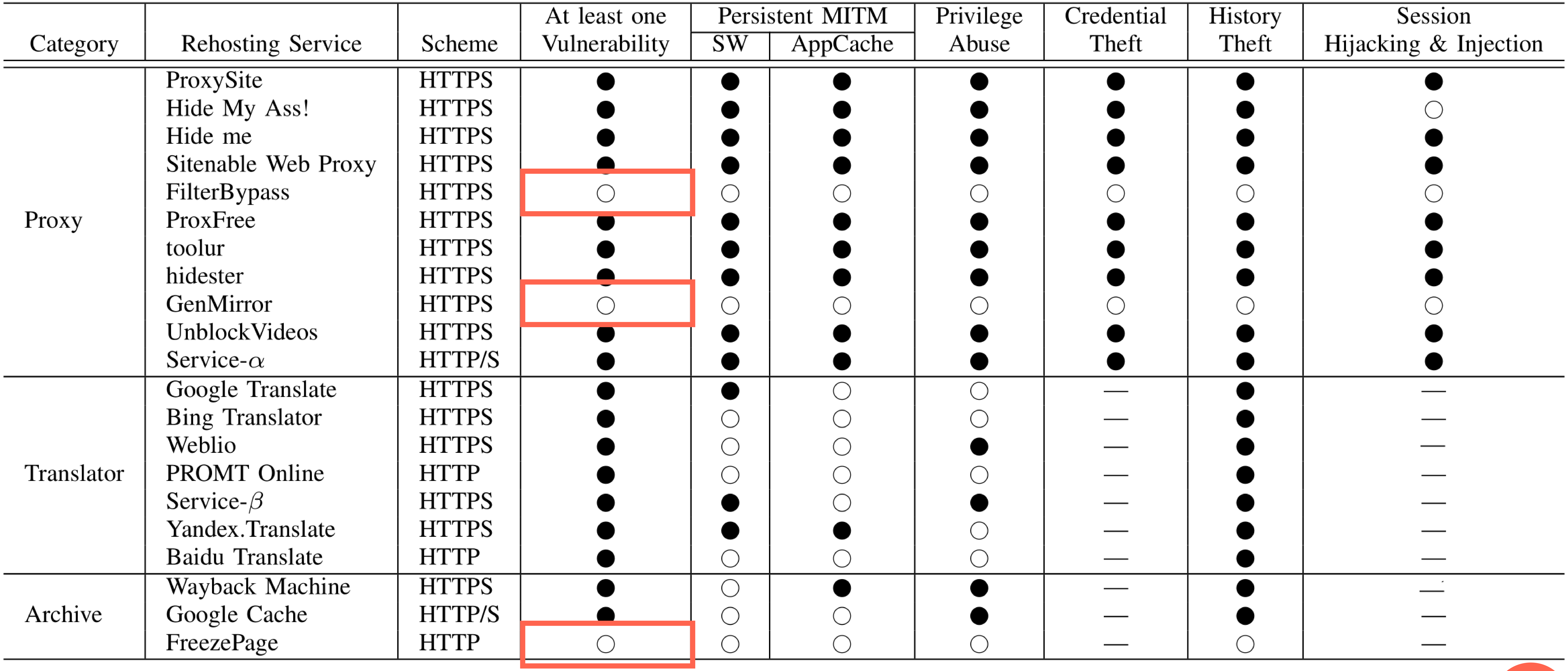
○ Secure



Category	Rehosting Service	Scheme	At least one Vulnerability	Persistent MITM		Privilege Abuse	Credential Theft	History Theft	Session Hijacking & Injection
				SW	AppCache				
Proxy	ProxySite	HTTPS	●	●	●	●	●	●	●
	Hide My Ass!	HTTPS	●	●	●	●	●	●	○
	Hide me	HTTPS	●	●	●	●	●	●	●
	Sitenable Web Proxy	HTTPS	●	●	[illegible]	[illegible]	[illegible]	[illegible]	●
	FilterBypass	HTTPS	○	○	[illegible]	[illegible]	[illegible]	[illegible]	○
	ProxFree	HTTPS	●	●	[illegible]	[illegible]	[illegible]	[illegible]	●
	toolur	HTTPS	●	●	[illegible]	[illegible]	[illegible]	[illegible]	●
	hidester	HTTPS	●	●	[illegible]	[illegible]	[illegible]	[illegible]	●
	GenMirror	HTTPS	○	○	[illegible]	[illegible]	[illegible]	[illegible]	○
	UnblockVideos	HTTPS	●	●	[illegible]	[illegible]	[illegible]	[illegible]	●
	Service-α	HTTP/S	●	●	●	●	●	●	●
Translator	Google Translate	HTTPS	●	●	○	○	—	●	—
	Bing Translator	HTTPS	●	○	○	○	—	●	—
	Weblio	HTTPS	●	○	○	●	—	●	—
	PROMT Online	HTTP	●	○	○	○	—	●	—
	Service-β	HTTPS	●	●	○	●	—	●	—
	Yandex.Translate	HTTPS	●	●	●	○	—	●	—
	Baidu Translate	HTTP	●	○	○	○	—	●	—
Archive	Wayback Machine	HTTPS	●	○	●	●	—	●	—
	Google Cache	HTTP/S	●	○	○	●	—	●	—
	FreezePage	HTTP	○	○	○	○	—	○	—

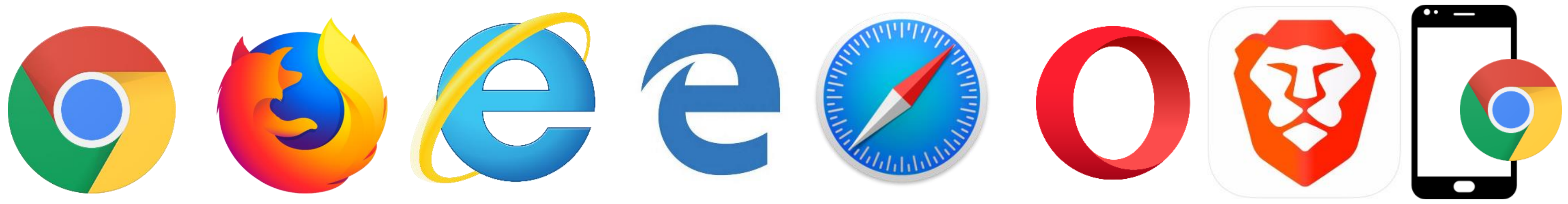
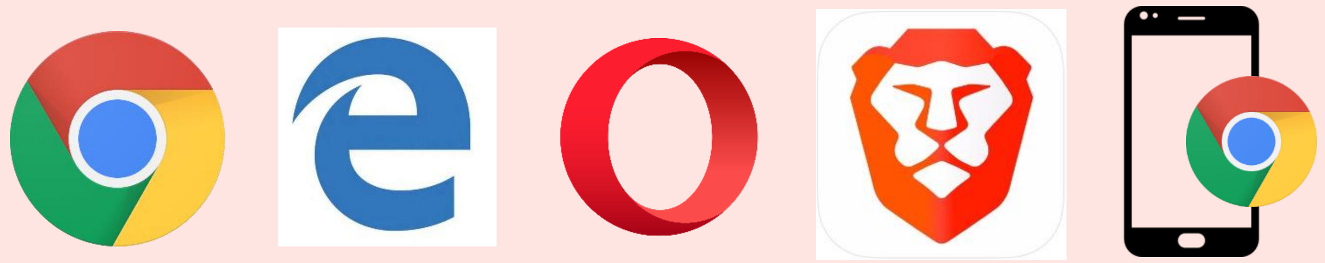
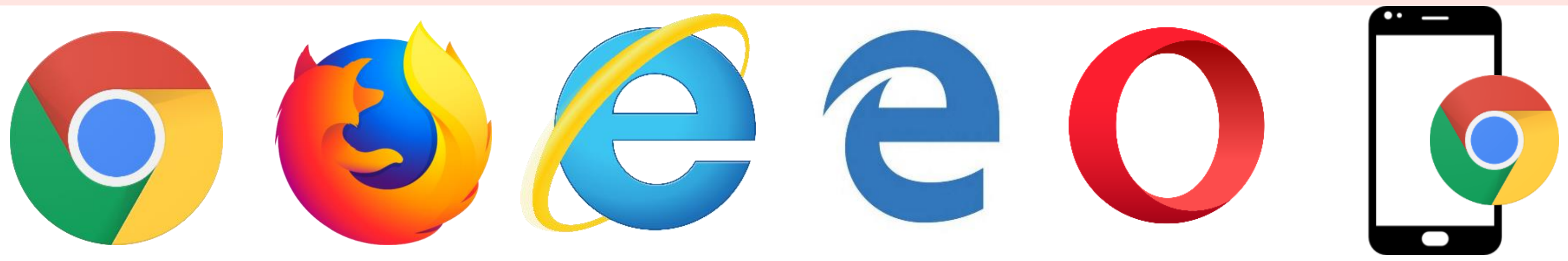
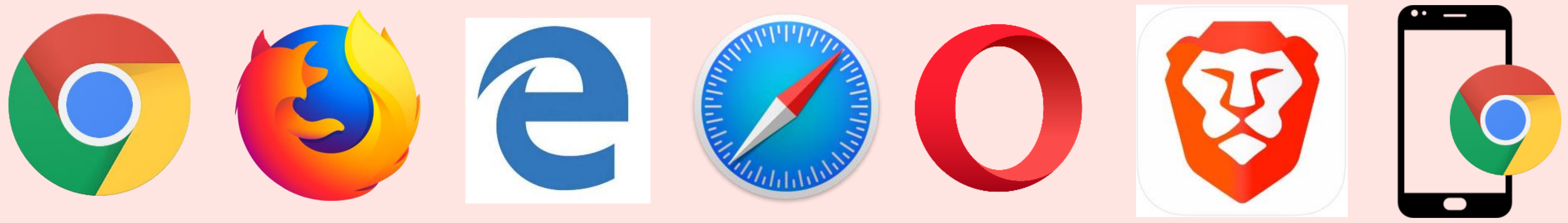
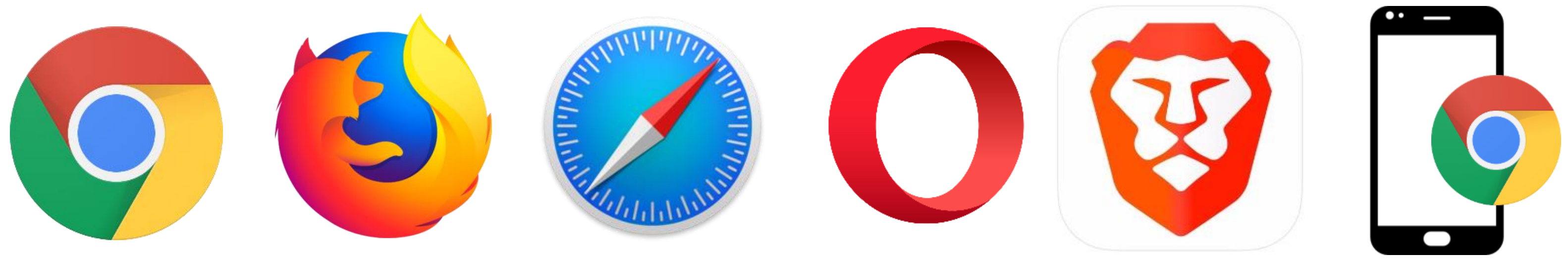
18 out of 21

Summary of Results



Category	Rehosting Service	Scheme	At least one Vulnerability	Persistent MITM		Privilege Abuse	Credential Theft	History Theft	Session Hijacking & Injection
				SW	AppCache				
Proxy	ProxySite	HTTPS	●	●	●	●	●	●	●
	Hide My Ass!	HTTPS	●	●	●	●	●	●	○
	Hide me	HTTPS	●	●	●	●	●	●	●
	Sitenable Web Proxy	HTTPS	●	●	●	●	●	●	●
	FilterBypass	HTTPS	○	○	○	○	○	○	○
	ProxFree	HTTPS	●	●	●	●	●	●	●
	toolur	HTTPS	●	●	●	●	●	●	●
	hidester	HTTPS	●	●	●	●	●	●	●
	GenMirror	HTTPS	○	○	○	○	○	○	○
	UnblockVideos	HTTPS	●	●	●	●	●	●	●
	Service-α	HTTP/S	●	●	●	●	●	●	●
Translator	Google Translate	HTTPS	●	●	○	○	—	●	—
	Bing Translator	HTTPS	●	○	○	○	—	●	—
	Weblio	HTTPS	●	○	○	●	—	●	—
	PROMT Online	HTTP	●	○	○	○	—	●	—
	Service-β	HTTPS	●	●	○	●	—	●	—
	Yandex.Translate	HTTPS	●	●	●	○	—	●	—
	Baidu Translate	HTTP	●	○	○	○	—	●	—
Archive	Wayback Machine	HTTPS	●	○	●	●	—	●	—
	Google Cache	HTTP/S	●	○	○	●	—	●	—
	FreezePage	HTTP	○	○	○	○	—	○	—

Browsers Comparison

#	Attacks	Browser
I	Persistent MITM	
II	Privilege Abuse	
III	Credential Theft	
IV	History Theft	
V	Session Hijacking and Injection	

Defenses for Web Rehosting

- Separate domain names for each rehosted page

https://rehosted.example/?url=a.example

➡ *https://a-example.rehosted.example/*

- Generate tentative URL inaccessible by 3rd party

⬌ Inhibit direct links

- Disable SW and AppCache (attack I)
- Use HTTPOnly (attack V)

Ethics

- We reported to affected service providers we examined.
  - 9 providers responded
  - 4 providers certified as vulnerability
  - 2 providers asked us not to be named

- We plan to make risks more widely known in cooperation with JPCERT/CC.

Future Directions

Other web rehosting services?

Other attacks?

- iframe [Lerner_CCS'17]
- Persistent XSS [Steffens_NDSS'19]

Human behaviors while using web rehosting?

- Private browsing
- Login
- Permission

Conclusion

- Explored security flaws of web rehosting services
- Presented 5 attacks exploiting various web features
- Found that 18 out of 21 services are vulnerable
- Reported risk to service providers with feasible defenses

✉ watanabe@nsl.cs.waseda.ac.jp

🐦 @twatanabe1203