Web Attacks For Fun and Profit

Sooel Son
KAIST

My Research Interests

Web & Mobile Security

Key Components of Mobile & Web Applications

Key Components of Mobile & Web Applications

Mobile Client

- Hybrid app
- Mobile ads

Web components

HTML5

Web Client

Client-side

HTTP/JSON

HTML/JS

Server-side

Inherent Architectural Threats

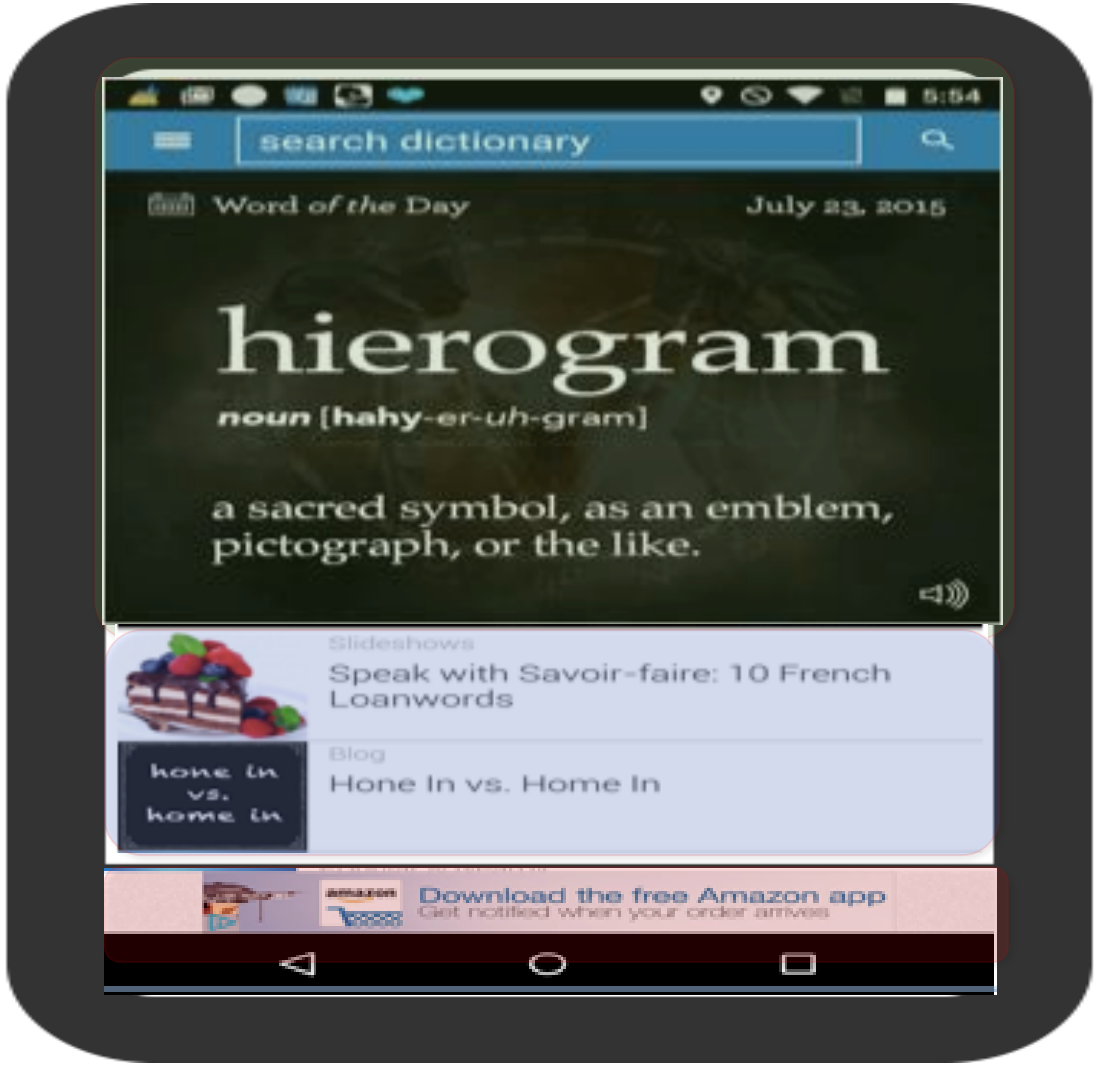
Malicious content

- Web pages
- Browser extensions
- Mobile apps
- Web & mobile ads

Malicious input

- Forged input parameters
- Incomplete authorization

Threat Model in Client-side Applications



Threat Model in Client-side Applications

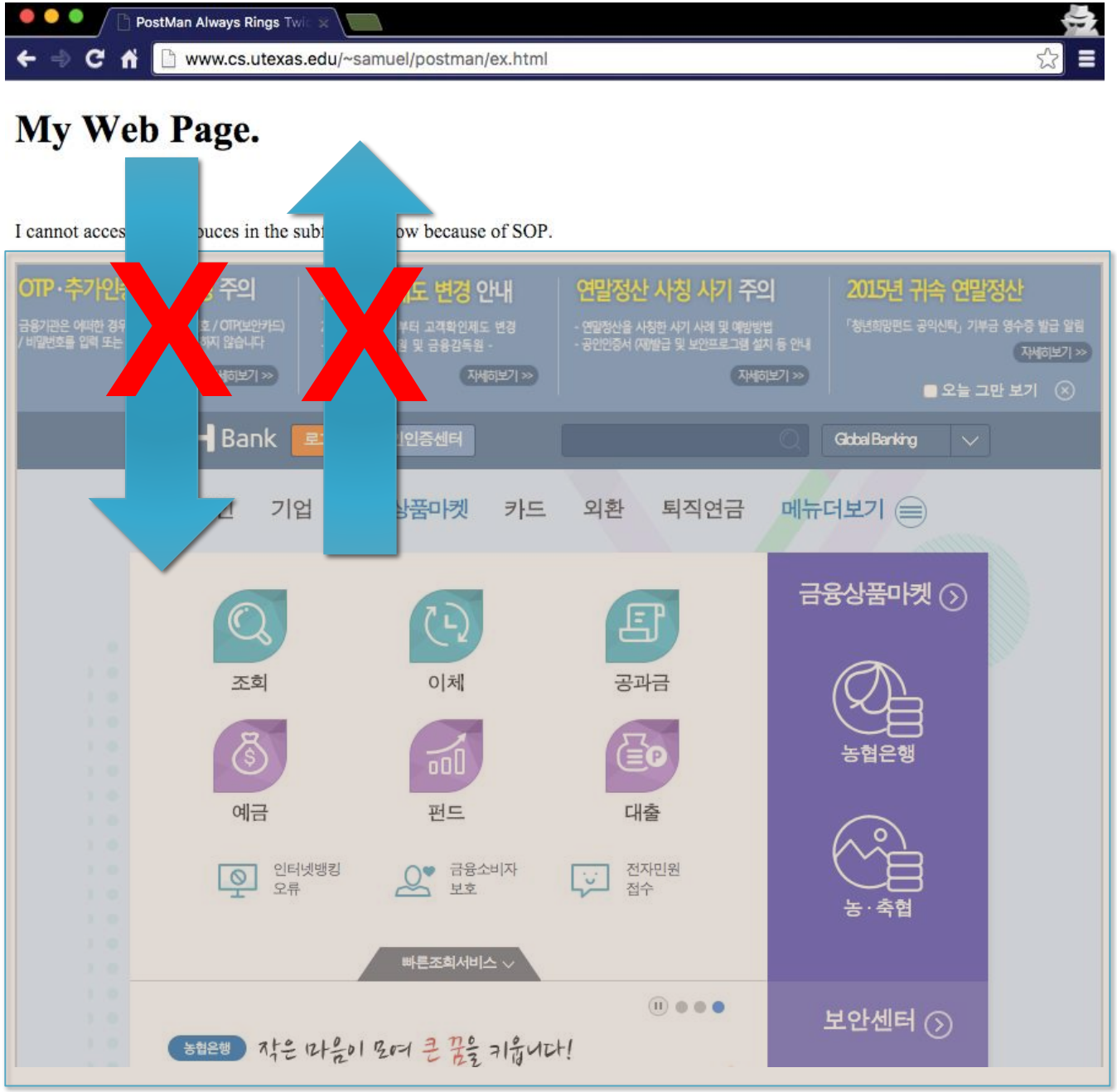
- Client-side mobile & Web apps serve trusted and untrusted content.
- How does a browser isolate trusted and untrusted?
  - Same Origin Policy (SOP)

Securing Frame Communication in Browsers
[Usenix Security 2008]

Same Origin Policy

- Script **S** that runs with origin A cannot access Web resources from any other origins.

- Origin: URI scheme, hostname and port.
  - https://www.kaist.ac.kr:80



Did SOP Solve Client-side Security?

- SOP is too strict.
  - Many apps need **cross-origin** communication.
  - Enforcing SOP is delegated to developers.

The Postman Always Rings Twice: Attacking and Defending postMessage in HTML5 Websites

Sooel Son and Vitaly Shmatikov [NDSS 2013]

- Found exploitable vulnerabilities in 84 popular domains.
- **Best Student Paper Award**

Did SOP Solve Client-side Security?

- SOP has loopholes.

Pride and Prejudice in Progressive Web Apps: Abusing Native App-like Features in Web Applications

Jiyeon Lee, Haeun Kim, Junghwan Park, Insik Shin, Sooel Son [CCS 2018]

- **Found new attacks** that abuse Progressive Web applications!

Today's Talk

Identify new vulnerabilities

The Postman Always Rings Twice: Attacking and Defending postMessage in HTML5 Websites ***[NDSS 2013]***

Pride and Prejudice in Progressive Web Apps: Abusing Native App-like Features in Web Applications ***[CCS 2018]***

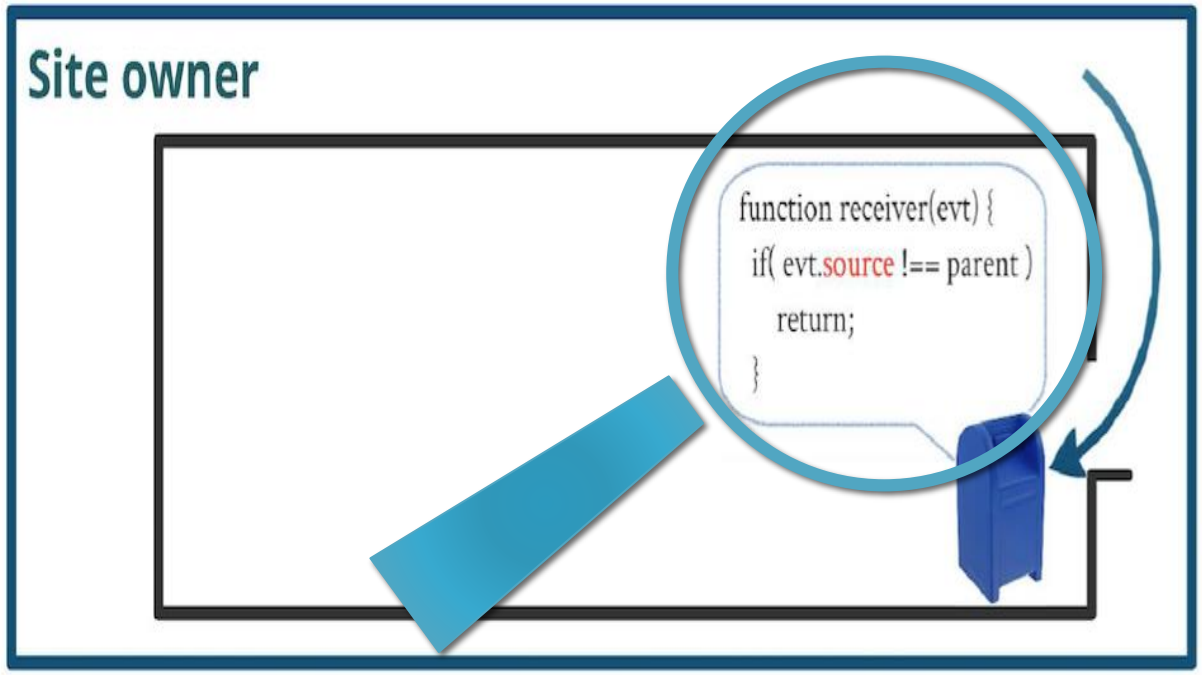
The Postman Always Rings Twice: Attacking and Defending postMessage in HTML5 Websites

Sooel Son and Vitaly Shmatikov

Network & Distributed System Security Symposium (NDSS) 2013

Best Student Paper Award

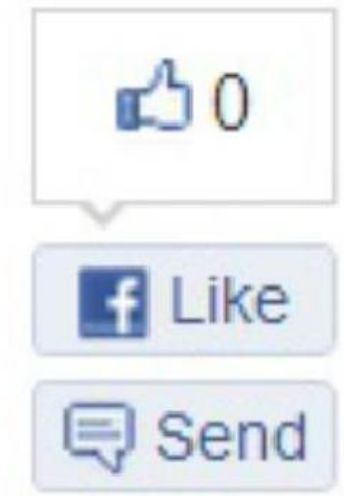
The Postman Always Rings Twice: Attacking and Defending postMessage in HTML5 Websites



- Identified cross-site scripting and other vulnerabilities due to delegated **same origin policy**.
- Found 84 vulnerable popular Web domains.

postMessage

- **Purpose: a "hole" in same origin**


0
Like
Send


Chartbeat


IEEE
Advancing Technology
for Humanity


WHOLE
FOODS
MARKET


amazon.com


Jump
Time™


CNN


COACH
EST. 1941
NEW YORK


UserReport

Web Browser

http://alice.edu

<script src="http://cntProvider/fancy.js"></script>

http://cntProvider/showFancy.html

Check the origin of the received message!

```
function msgReceiver(e) {
 if(e.origin !== "http://hostA")
```

HTML Living Standard (whatwg.org)

Authors should check the origin attribute to ensure that messages are only accepted from domains that they expect to receive messages from

What can go wrong with the absent checks

www.bogusjumptime.com/exploit/

Client-side XSS

People

Like 1.6m

Search

HOME | NEWS | PHOTOS | STYLE | RED CARPET | ROYALS | TV WATCH | BABIES | PETS | BEST OF 2012 | CELEBS | VIDEO | MAGAZINE

N D S S SYMPOSIUM

NDSS 2013 call for papers

TOP STORY 09:45AM EST

THE LATEST | MOST SHARED

The Postman Always Rings Twice: Attacking and Defending postMessage in HTML5 Websites
10:00AM EST

The camera-ready due for NDSS 2013 is coming up
TV WATCH ONLY ON PEOPLE.COM 09:10AM EST

Internet Society 20 years
09:05AM EST

19th Annual Network & Distributed System

```
var w = /jumptime\.com(:[0-9]?$/;
if( !v.origin.match(w) )
```

READ IT

Like 12k Tweet

People

Like 1.6m

Search

HOME | NEWS | PHOTOS | STYLE | RED CARPET | ROYALS | TV WATCH | BABIES | PETS | BEST OF 2012 | CELEBS | VIDEO | MAGAZINE

N D S S

NDSS 2013 call for papers

TOP STORY 09:45AM EST

THE LATEST | MOST SHARED

The Postman Always Rings Twice: Attacking and Defending postMessage in HTML5 Websites
10:00AM EST

The camera-ready due for NDSS 2013 is coming up
TV WATCH ONLY ON PEOPLE.COM 09:10AM EST

Internet Society 20 years
09:05AM EST

19th Annual Network & Distributed System Security Symposium

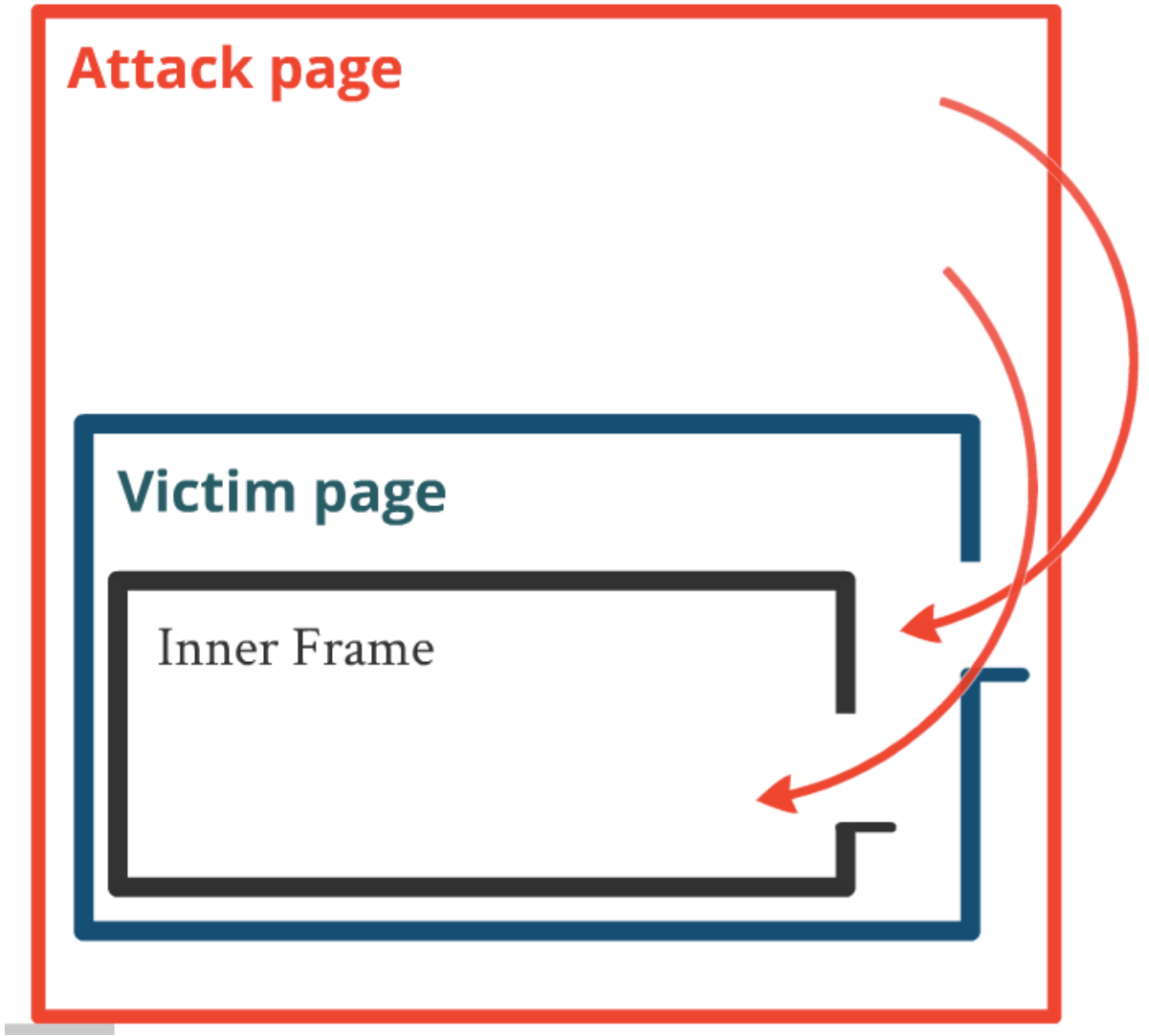
WHAT YOU ♥ RIGHT NOW

READ IT

Like 12k | Tweet | +1

postMessage Usage in the Wild

- **RvScope**: Tool to automatically collect receivers.
- Collected postMessage receivers from Alexa top 10,000 sites.
- 16,115 pages from 10,121 host names.



postMessage Vulnerabilities in the Wild

2,245 hosts (22%) have a postMessage receiver.

1,585 hosts have a receiver with **no origin check.**

262 hosts have **incorrect checks.**

postMessage Vulnerabilities in the Wild (2)

84 hosts have an **exploitable vulnerability.**

JavaScript check example:

```
if (m.origin.indexOf(“sharethis.com”) != -1)
```

Intended:
subdomain.sharethis.com

Possible:
sharethis.com.attacker.co.kr
evilsharethis.com

postMessage Vulnerabilities in the Wild (2)

- if(a.origin && a.origin.match(/\.kissmetrics\.com/))
  - www.kissmetrics.com.evil.com

- var w = /jumptime\.com(: [0 – 9])?$/;
- if (!v.origin.match(w))
  - evil**jumptime.com**

- if((/\api.weibo\.com$/).test(I.origin))
  - evil**api-weibo.com**

Consequences of postMessage attacks.

- Cross-site scripting attacks
- Reading cookies
- Reading or writing local storage values

Lessons

Put a correct origin check in every receiver!

Technical Challenges

Site owner: http://alice.edu

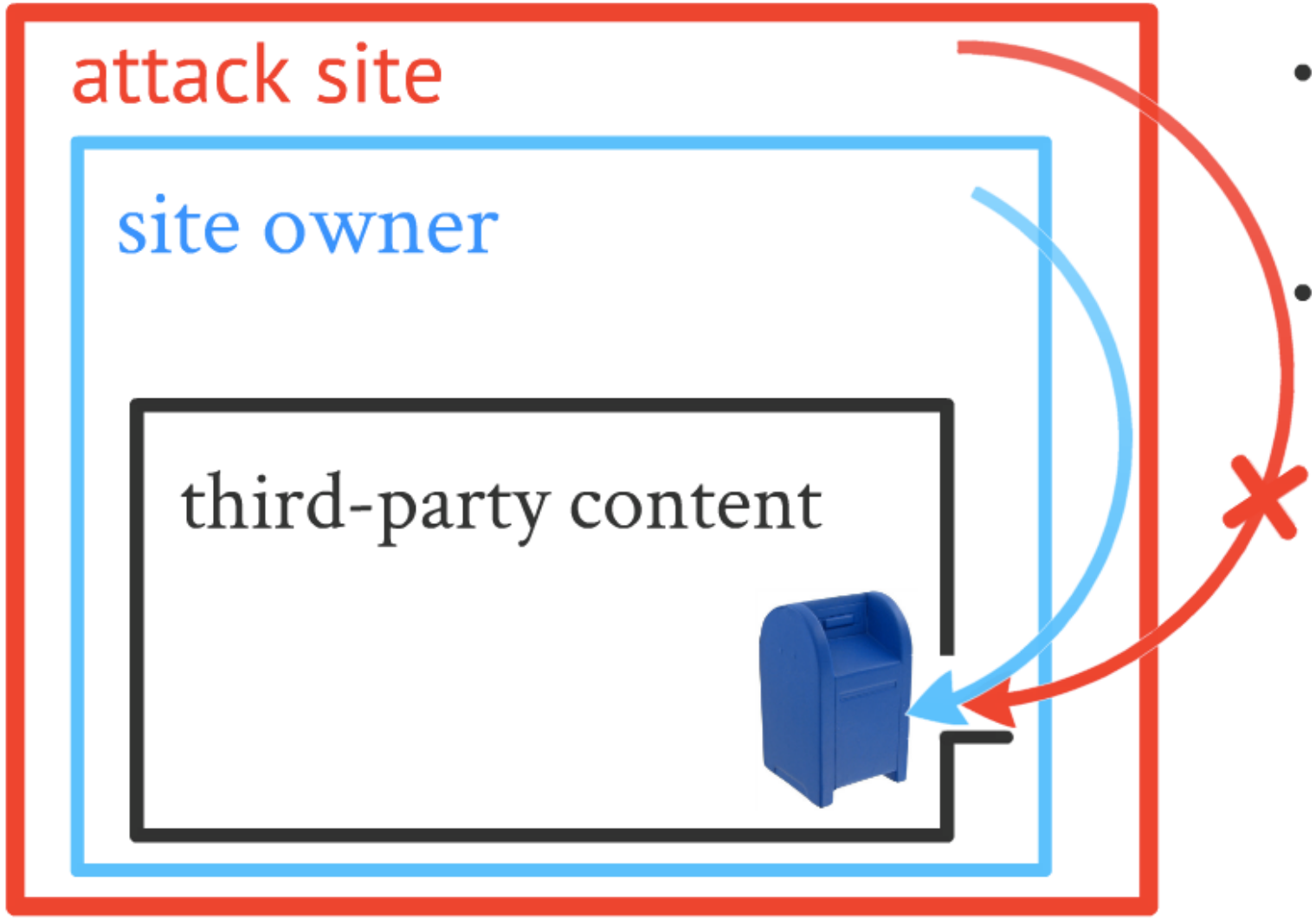
<script src=http://3rd-party.com/script.js></script>

No control over the included code.

3rd-party content provider: 3rd-party.com

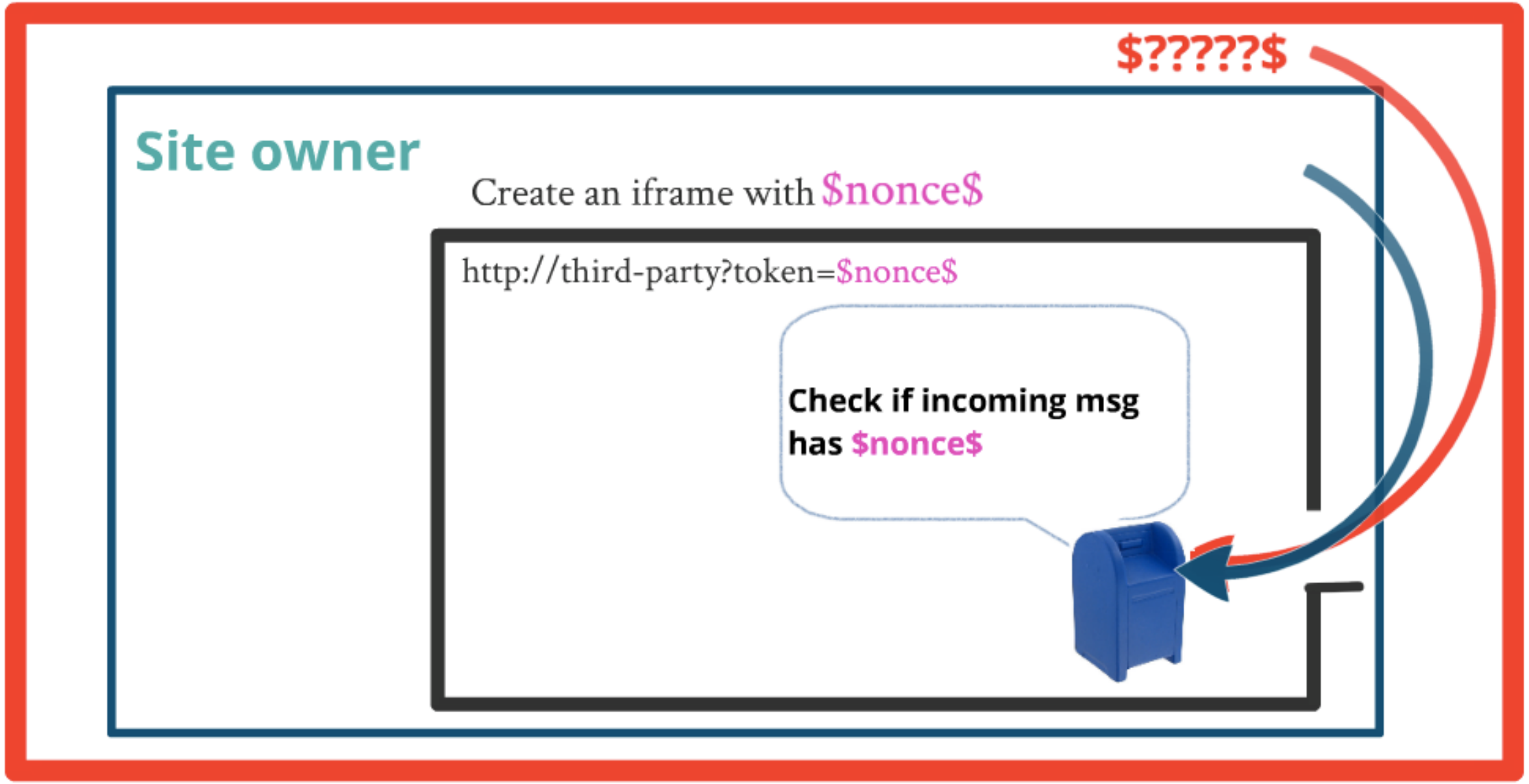
Correct origins depend where they are included.

Defense: Threat model



- Site owner is honest.
- The receiver should accept message only from site owner's origin.

Defense: Origin-based defense with a shared token.



Do Web vulnerabilities exist only in pure Web apps?

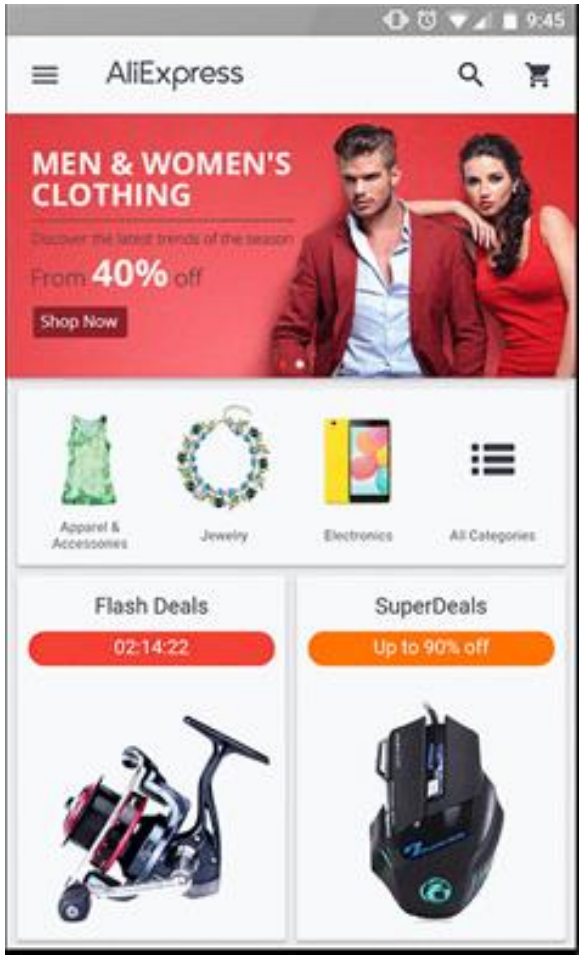
Let’s abuse new features in a new generation of Web applications: Progressive Web App

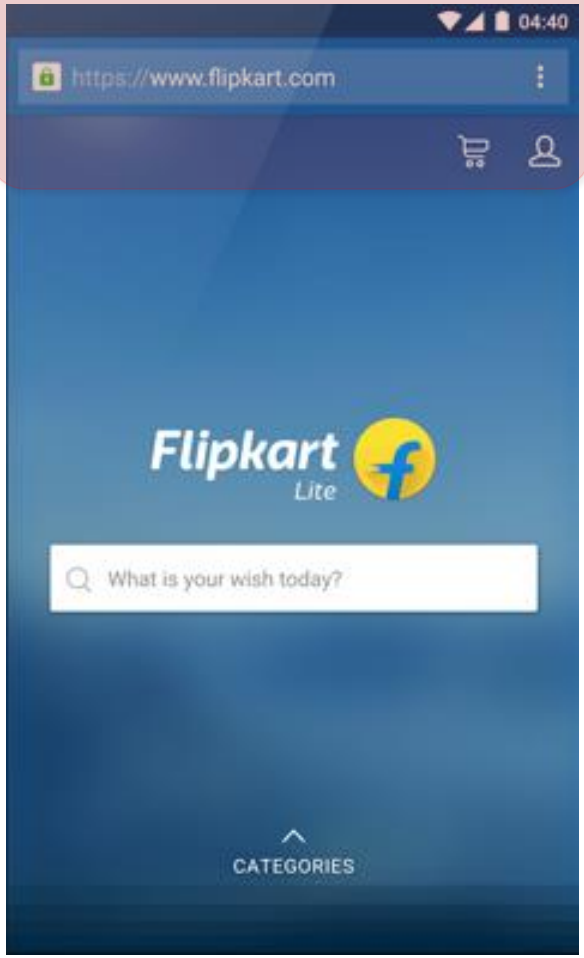
Pride and Prejudice in Progressive Web Apps: Abusing Native App-like Features in Web Applications

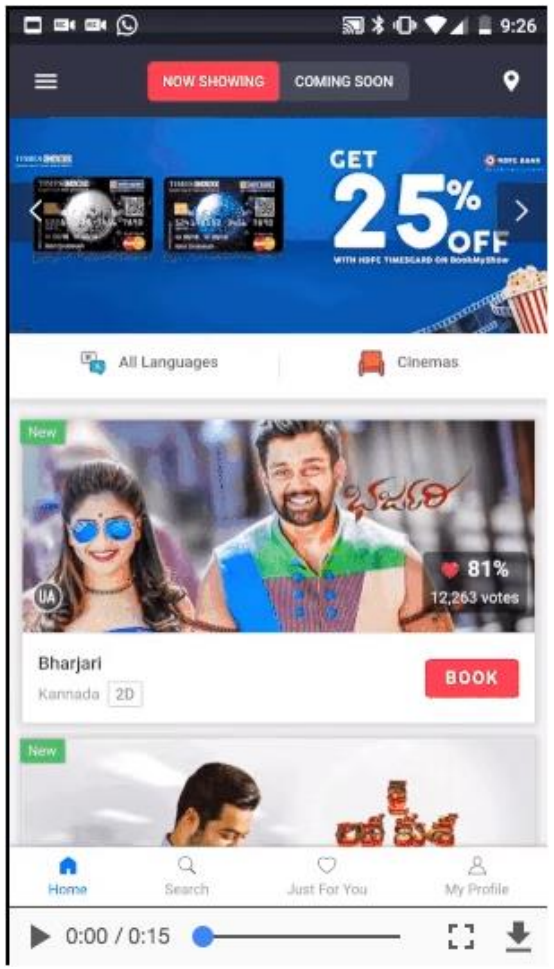
Jiyeon Lee, Hayeon Kim, Junghwan Park,
Insik Shin, and **Sooel Son**

ACM Conference on Computer and Communications Security (CCS) 2018

Progressive Web App







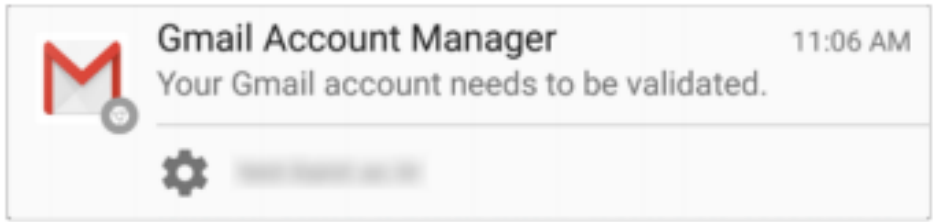
Why do they want a PWA?

Native app-like features

- They want to alarm you with notices whenever they want
- The website should be accessible even in offline

Progressive Web App

- Three key technical features
  - **Push notification**

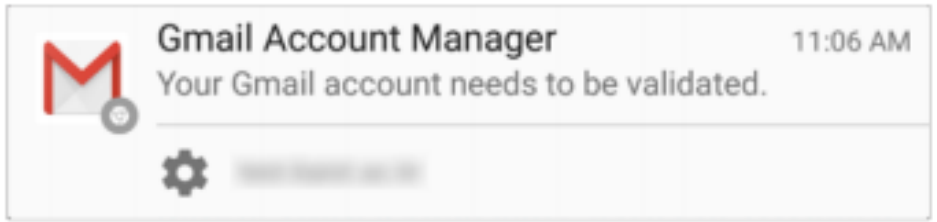


  - **Cache**

  - **Service worker**

Progressive Web App

- Three key technical features
  - **Push notification**



  - **Cache (History Sniffing Attack)**

  - **Service worker (Crypto mining Attack)**

How many do Progressive Web Apps (PWAs) exist?

- Among the Alexa top 100,000 sites

3,351 websites use push notifications

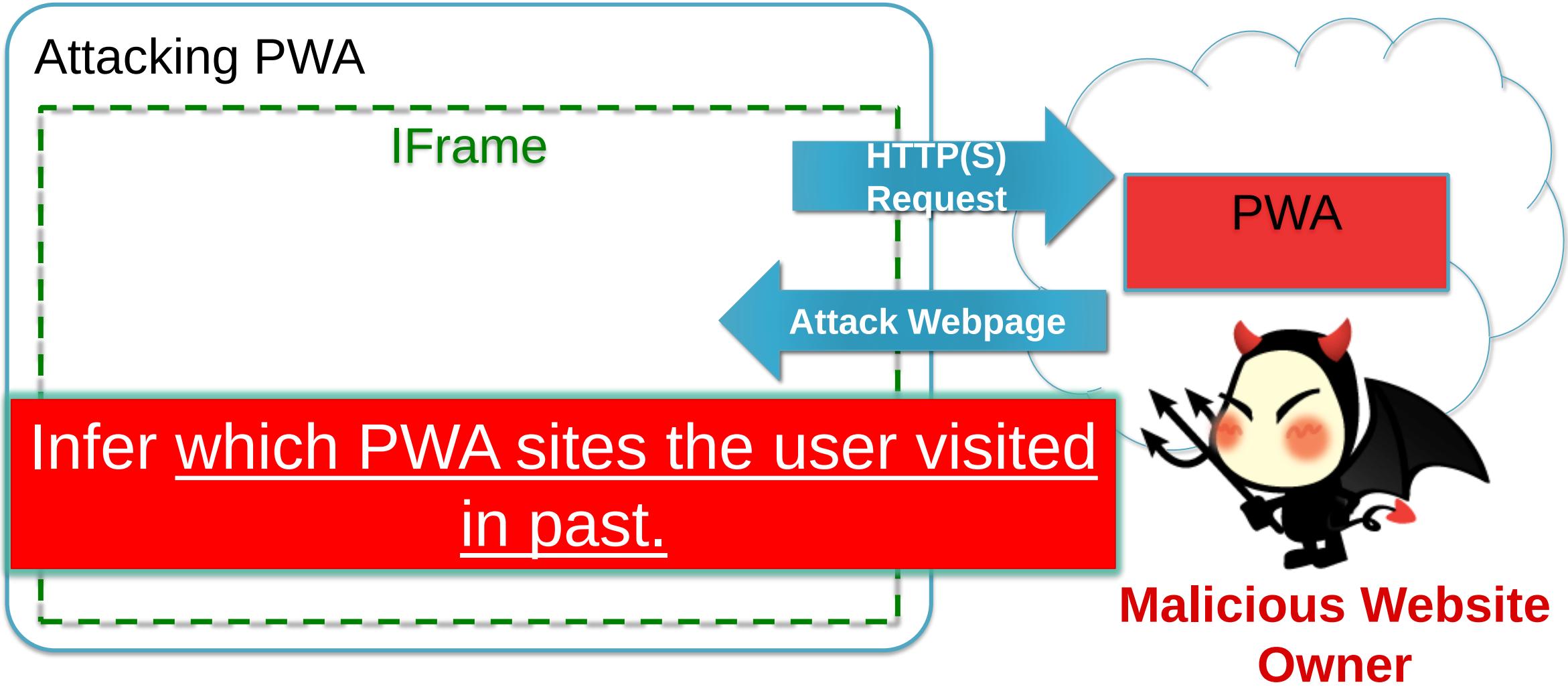
513 websites use caches

4,163 websites use service workers

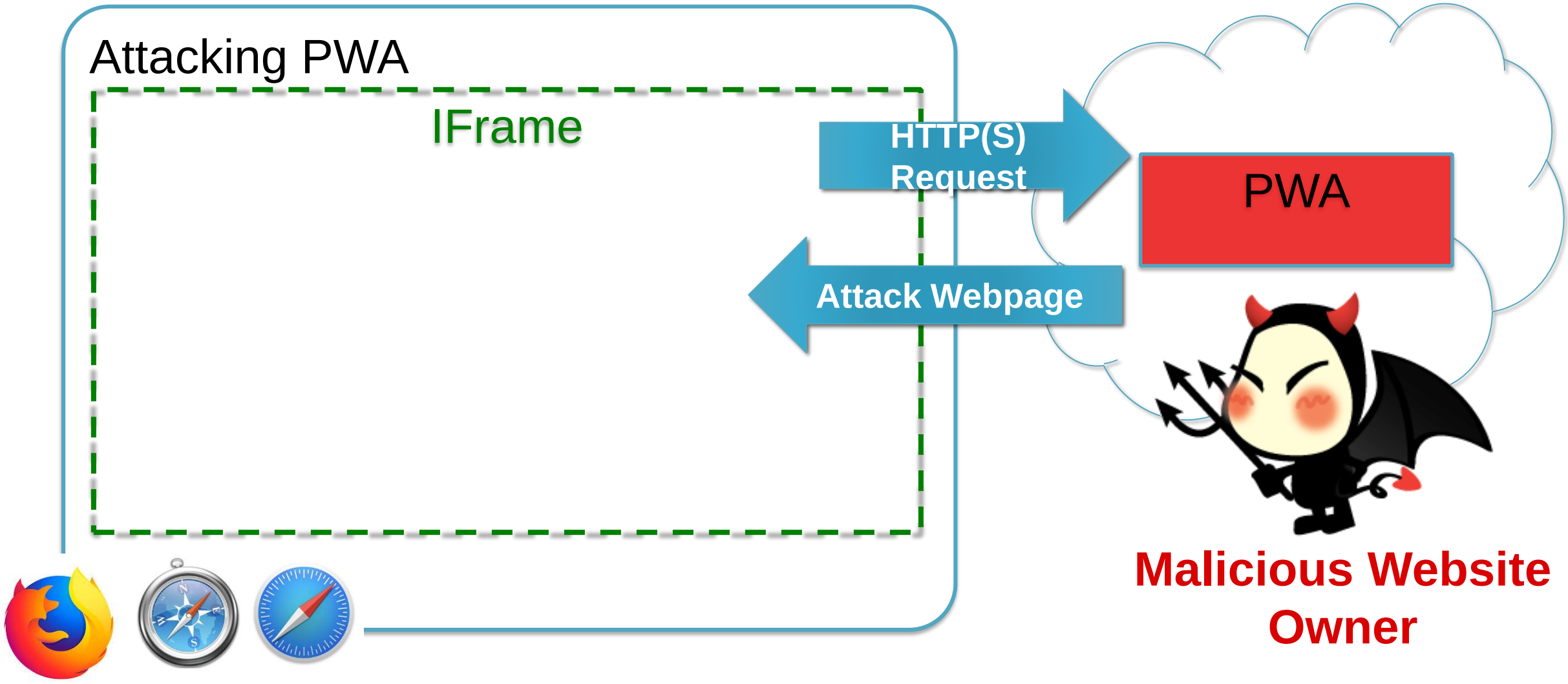
Features Used	# of Websites (% Percentage)
Push	440 (10.6%)
Push with library	2,911 (69.9%)
Cache	513 (12.3%)
Both	196 (4.7%)
Others	495 (11.9%)
Total	**4,163 (100%)**

Table 1: PWA statistics for the Alexa top 100,000 sites

History-Sniffing Attack Model



1st Step for The History-Sniffing Attack



2nd Step for The History-Sniffing Attack

Attacking PWA

iFram

HTML

```
<!DOCTYPE html>
<html>
<body>
  <iframe id='tgt'></iframe>
</body>
</html>
```

JavaScript

```
function loaded() {
  console.log('tested');
}

document.getElementById('tgt')
  .addEventListener('load', loaded, true);

document.getElementById('tgt').src
 = 'https://www.kaist.ac.kr'
```

Is loading successful?

https://www.pwaexample.com/

Malicious Website Owner

2nd Step for The History-Sniffing Attack

Attacking PWA

iFrame

HTTPS Request

PWA

Is loading successful?

https://www.pwaexample.com/

If Yes, the victim visited

https://www.pwaexample.com/

Malicious Website Owner

This website **cannot** read any content in iFrame

… but the website use the **presence of the cached webpage** to infer the user's browsing history.

Attacking PWA

iFrame

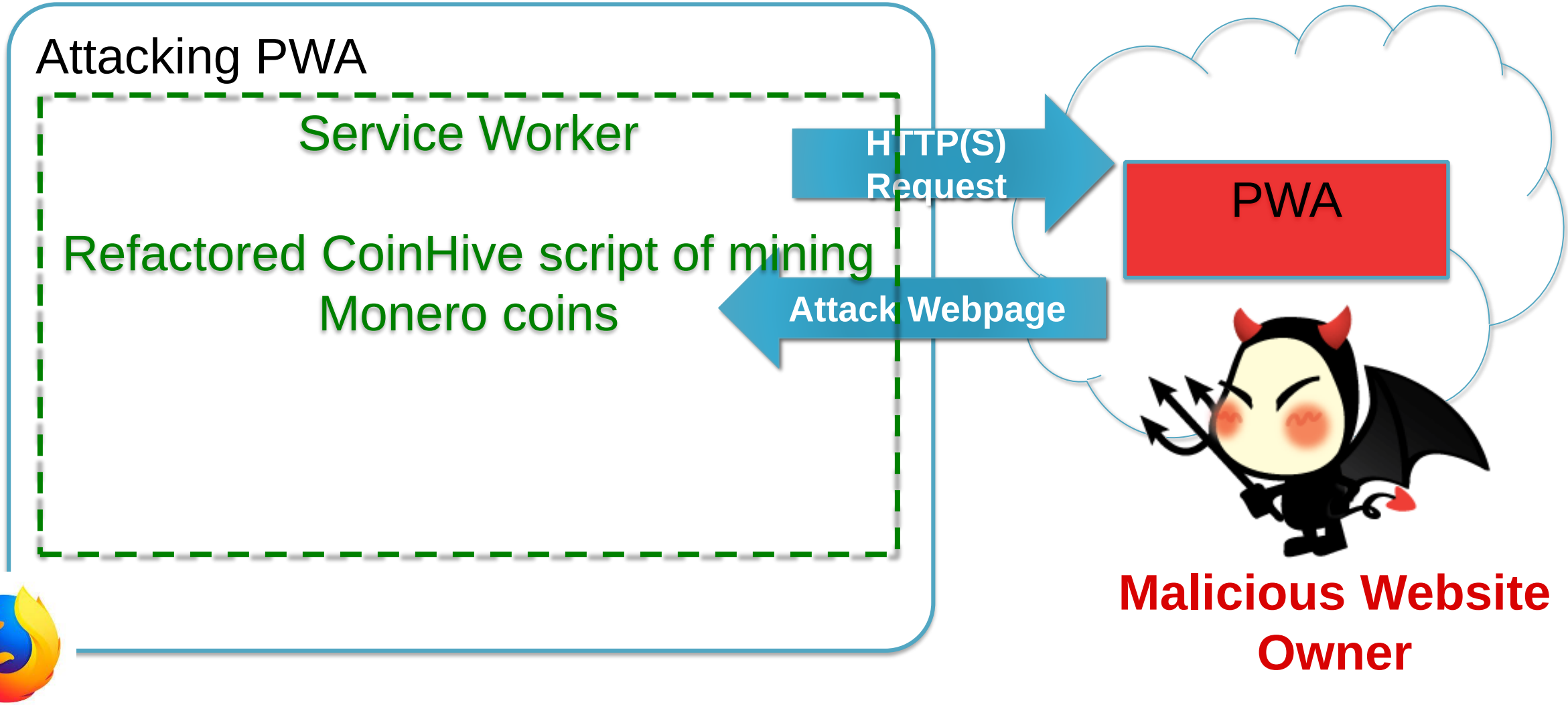
HTTPS Request

PWA

Does NOT violate same origin policy!

Malicious Website Owner

Cryptomining Attack Model



Cryptomining Attack Model

Service Worker

Refactored CoinHive script of mining Monero coins

- Technical Challenges
  - For the nature of mining, service worker should regularly update its mining transaction to verify
  - Service worker becomes idle when there is no event to process

Push Notifcation!

How Much Money Can You Make?

Monero price(Apr 23, 2018, close): $283.30

Browser	Execution Environment	Number of Solved Hashes		Amount of Monero	
		Total (24h)	Average (1h)	Total (24h)	Average (1h)
Chrome 65	Windows 10 Desktop	225,024	9,376	0.00001266 ($0.00358657)	0.00000053 ($0.00014944)
Firefox 59	Windows 10 Desktop	195,840	8,160	0.00001119 ($0.00317013)	0.00000047 ($0.00013209)
Chrome 65	Android 8.0 Google Pixel Phone	50,176	2,091	0.00000282 ($0.00079891)	0.00000012 ($0.00003329)
Chrome 65	macOS High Sierra 10.13.4	138,496	5,771	0.00000778 ($0.00220407)	0.00000032 ($0.00009184)

Table 3: Monero mining rewards for 24 hours by one service worker

From 1 visitors with 24 hours, 0.0035$

From 10 visitors with 24 hours, 0.035$

From 10,000 visitors with 24 hours, 35$

From 1M visitors with 24 hours, 3,500$

Contributions

- **First study** of how PWA can be abused by Web attackers.
- Standard Web same origin policy is no longer secure.
  - Mere existence of a cached websites can reveal the sensitive browsing history of a victim.
- Abusing the persistency of service worker is possible.
  - The attacker can abuse the computation powers of webpage visitors indefinitely even the victims leaves the webpage.

Questions?