Cybozu Labs

Outbound Port 80 blocking のご提案

サイボウズ・ラボ株式会社

竹迫 良範

<takesako@shibuya.pm.org>

Outbound Port 80 blockingのご提案

概要

2012年11月にIETFで策定されたRFC 6797 - HTTP Strict Transport Security のプロトコルが誕生した背景と現在のブラウザの実装状況について解説します。

2010年10月にEric Butler氏によって公開されたFiresheepのデモでは、無線LAN上に流れているパケットを解析して、そこに流れているFacebookやTwitterの平文のHTTP Cookieの情報を盗み取り、マウスのクリックだけで簡単にセッションハイジャックできる様子が示されました。

安心してWebブラウジングを行えるインターネット接続環境としてユーザ様への「OP80B」メニューの提供をご提案いたします。

発表者

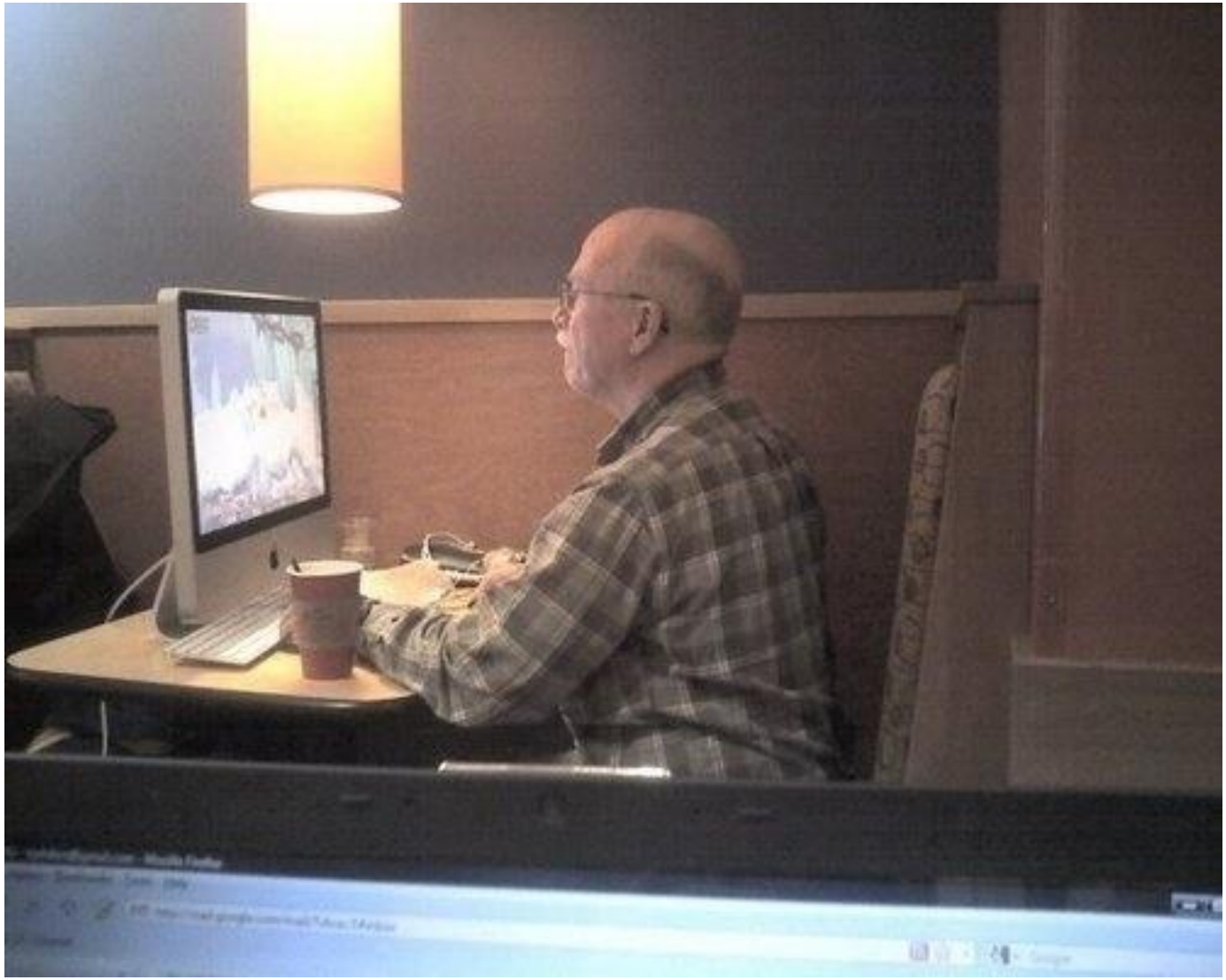
竹迫 良範 (サイボウズ・ラボ株式会社)

http://www.janog.gr.jp/meeting/janog31/program/OP80B.html

[固有名詞]ドヤリング とは？

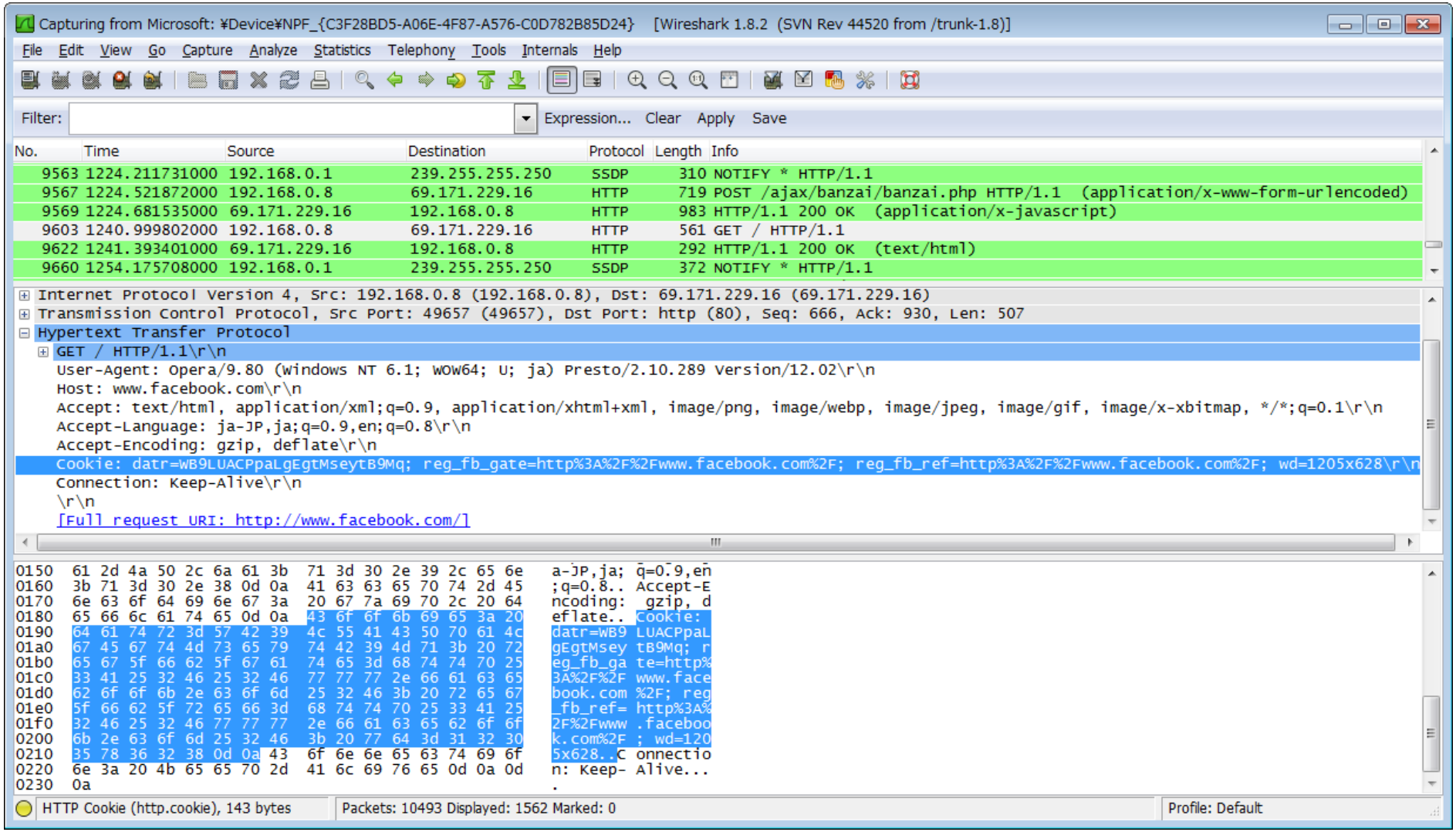
- スタバでMacBook Airをドヤ顔でいじる行為
- ドヤラー[名]
  - ドヤリングを行っている人
- ドヤリズム[名]
  - ドヤリングを取り入れたライフスタイル、思想
- ドヤリスト[名]
  - ドヤリズムを信奉し、また実践する人
- ドヤる[動ラ五]
  - ドヤリングをする（俗語）

本場のドヤリング（海外の上級者）



引用元：Google画像検索キーワード「ドヤリング」

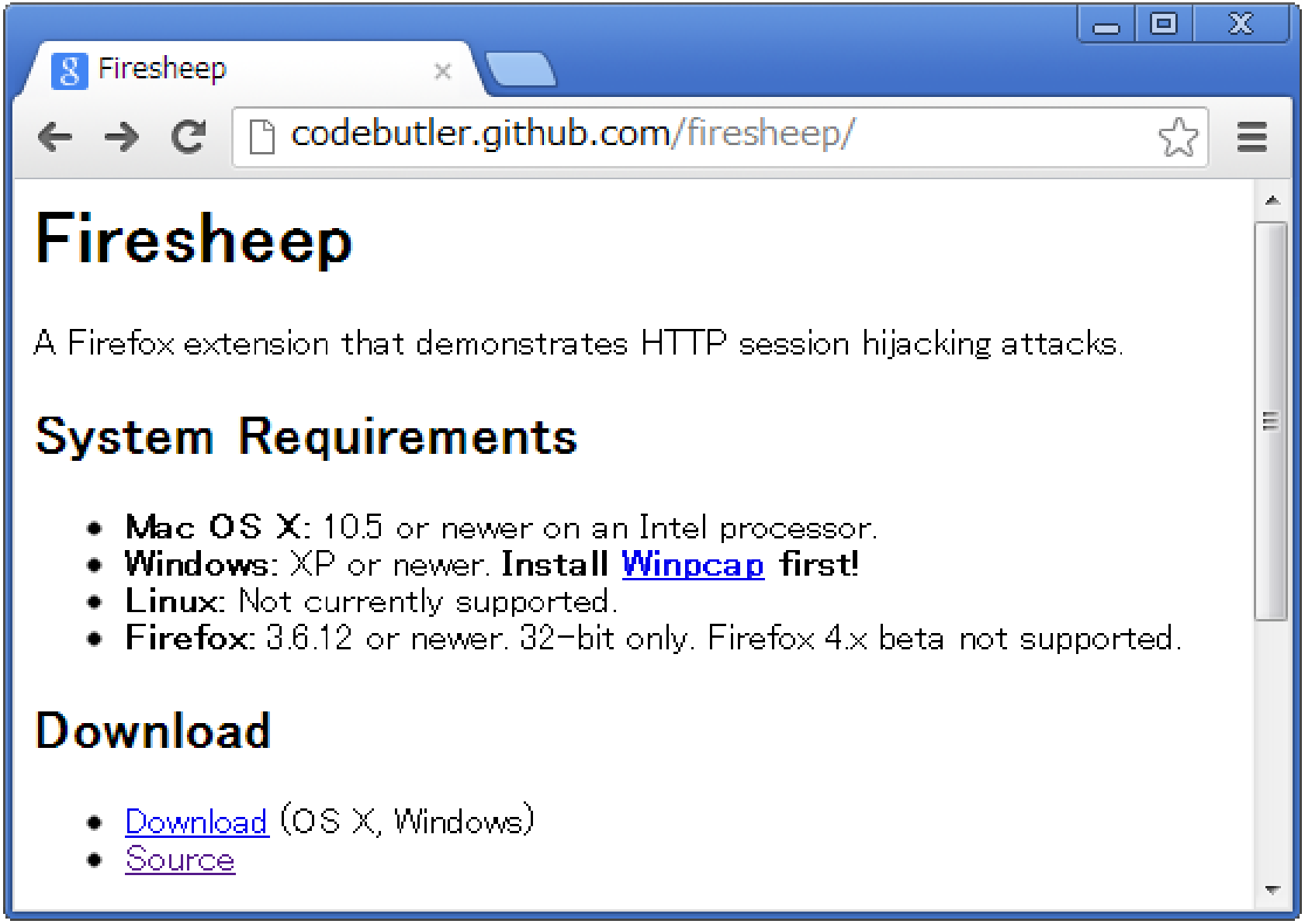
ドヤ顔で Wireshark の画面を眺める人？？？



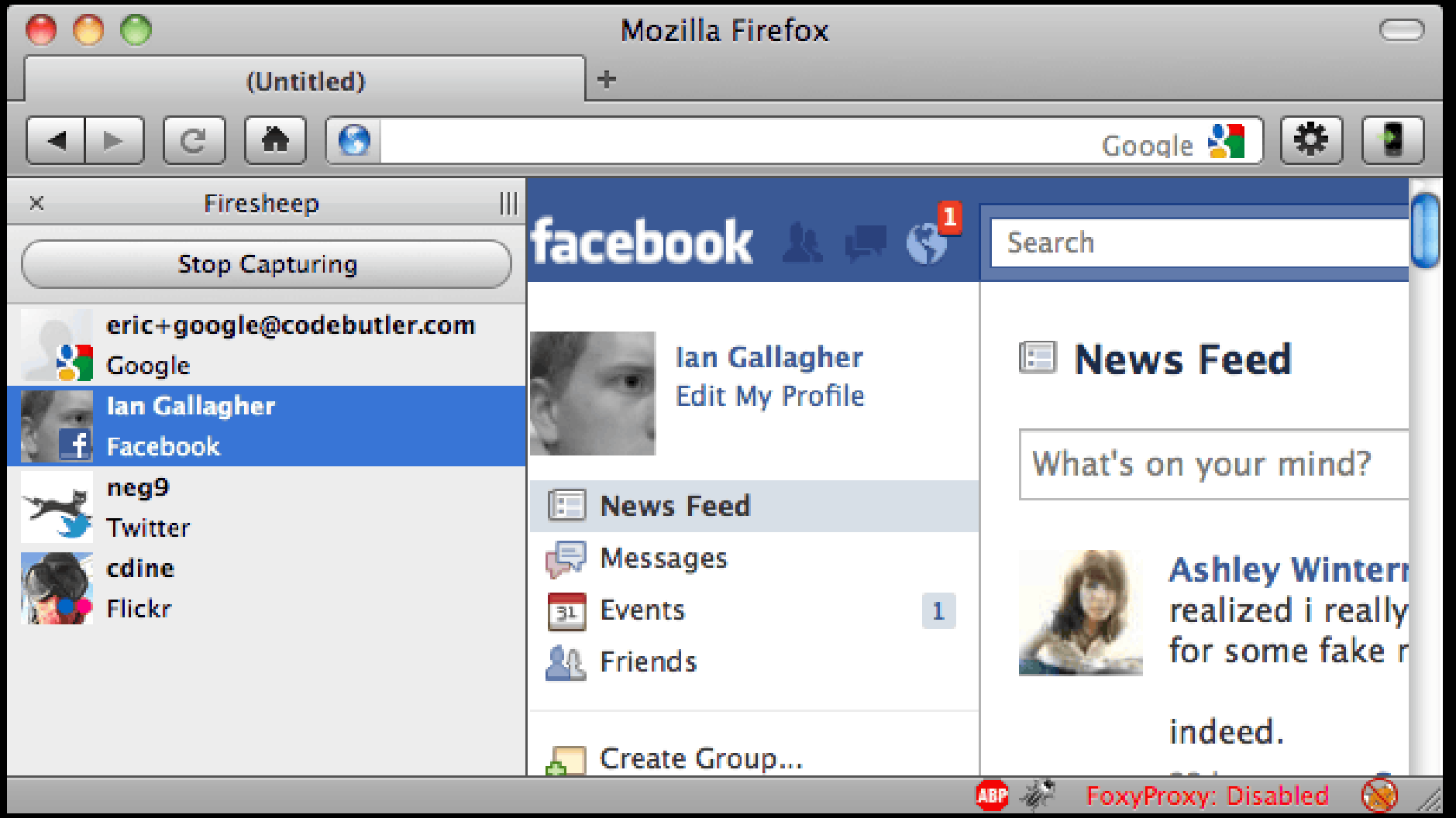
＿人人人人人人人人人＿
＞　突然のFiresheep　＜
￣Y^Y^Y^Y^Y^Y^Y^Y^Y￣

Firesheepとは

- 2010年10月
  - Firefoxの拡張として公開
- 開発者
  - Eric Butler氏
- 無線LAN上に流れているパケットを解析して、そこに流れているfacebookやTwitterの平文のHTTP Cookieの情報を盗み取り、マウスのクリックだけでセッションハイジャックする
- インストールがとても簡単
  - PoCとしてツールを公開し現実的な脅威を示した


Firesheep
codebutler.github.com/firesheep/
Firesheep
A Firefox extension that demonstrates HTTP session hijacking attacks.
System Requirements
Mac OS X: 10.5 or newer on an Intel processor.
Windows: XP or newer. Install Winpcap first!
Linux: Not currently supported.
Firefox: 3.6.12 or newer. 32-bit only. Firefox 4.x beta not supported.
Download
Download (OS X, Windows)
Source

【参考】Firesheep の動作画面

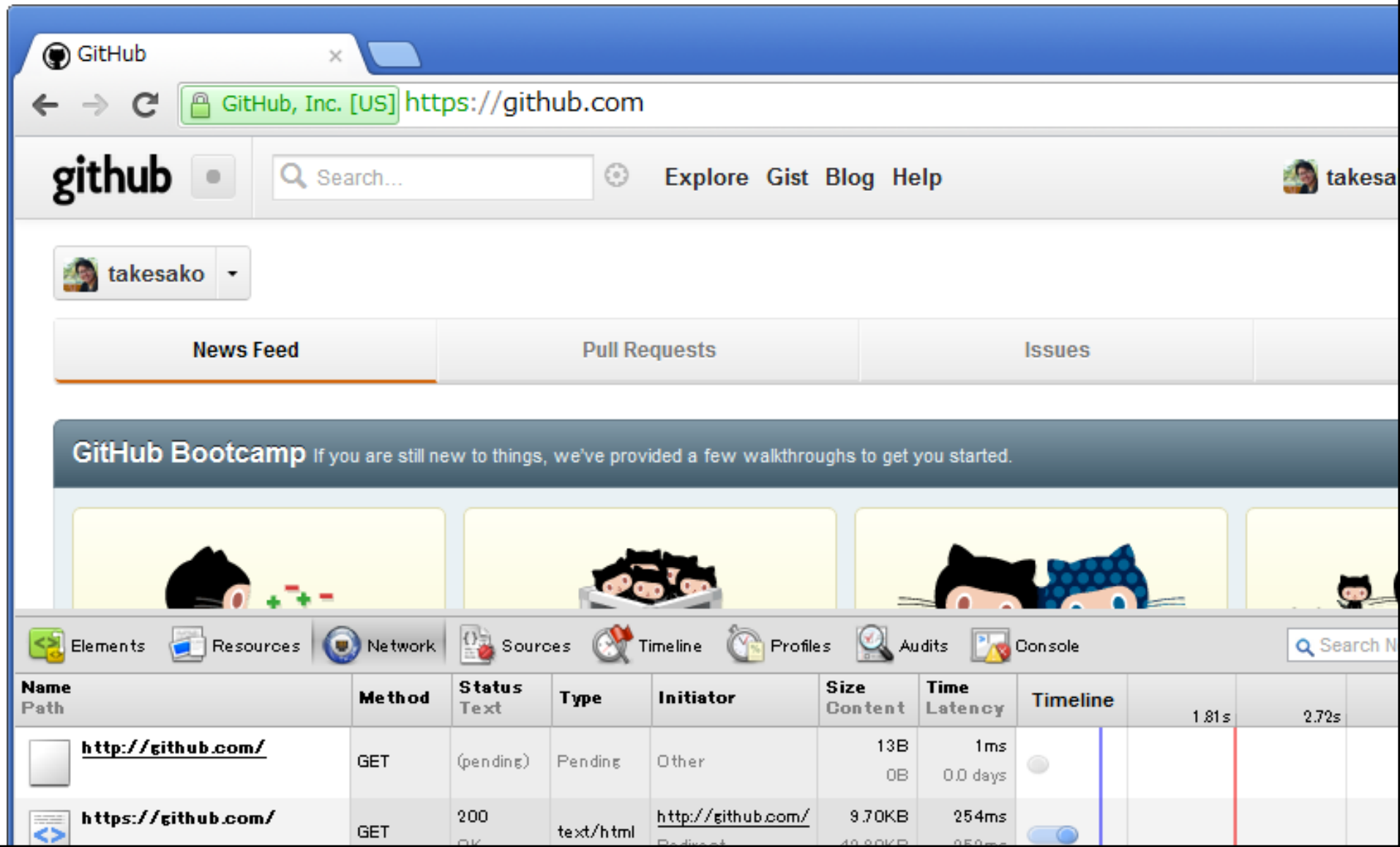


Eric Butler氏の記事より引用

Firesheepがデフォルトで対象とするサービス

- アカウントを乗っ取る対象のWebサービス
  - Amazon.com、CNET、dropbox、Evernote、Facebook、Flickr、Github、Google、WindowsLive、NY Times、tumblr、Twitter、WordPress、Yahoo . . .
- マウスのクリック操作だけで
  - 簡単になりすましができる
- 大手Webサービス運営各社に衝撃
  - → ＼突然のSSL化／

github.com ＼突然のSSL化／



Set-Cookieにsecureフラグが付くようになった

- (例) https://github.com/ からのレスポンス

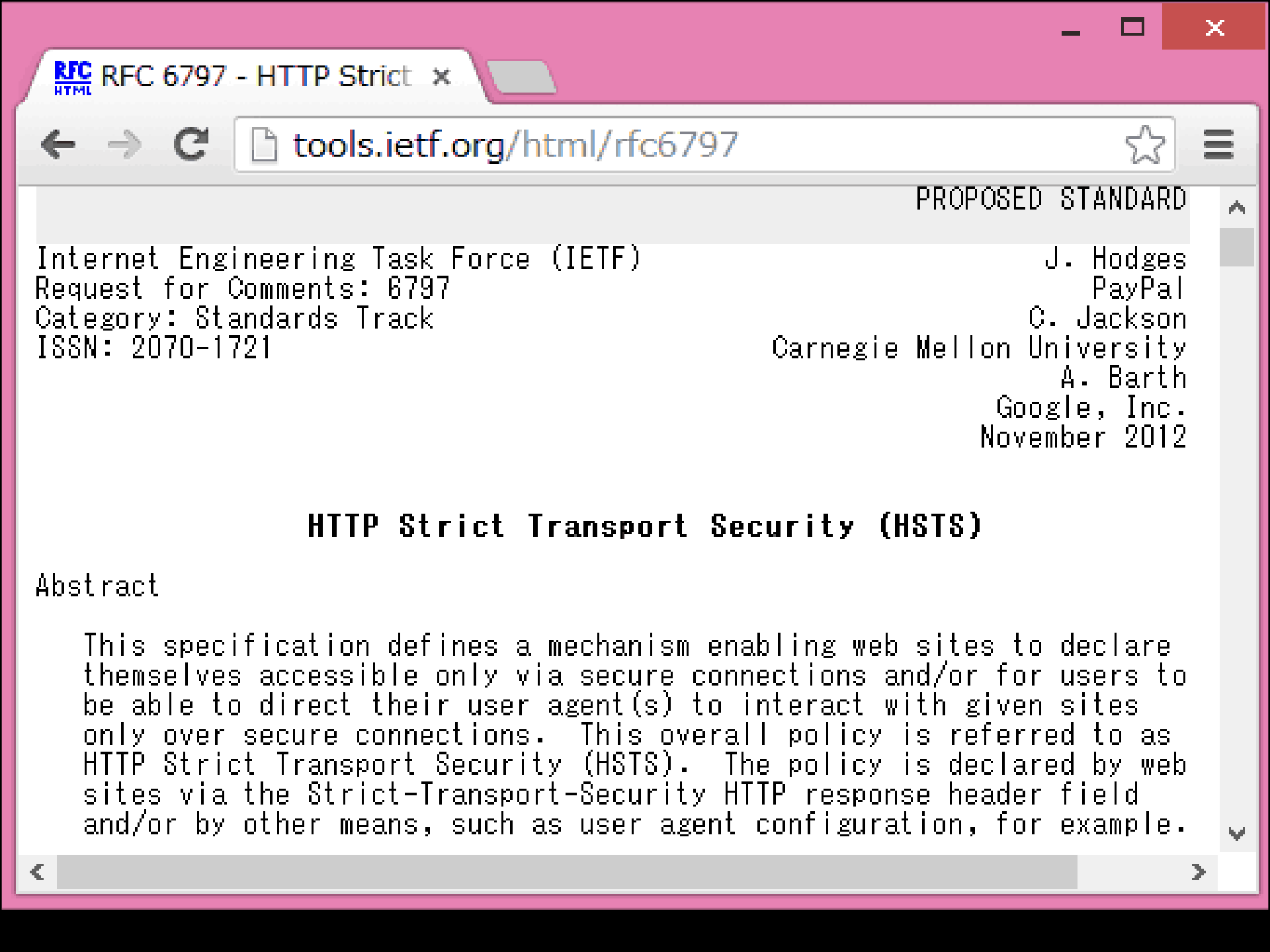
```
HTTP/1.1 200 OK
Server: nginx
Date: Fri, 14 Sep 2012 09:39:57 GMT
Content-Type: text/html; charset=utf-8
Transfer-Encoding: chunked
Connection: keep-alive
Status: 200 OK
Cache-Control: private, max-age=0, must-revalidate
Strict-Transport-Security: max-age=2592000
X-Frame-Options: deny
Set-Cookie: _gh_sess=XYZ; path=/; secure; HttpOnly
```

Strict-Transport-Security ヘッダも出力！！

- (例) https://github.com/ からのレスポンス

```
HTTP/1.1 200 OK
Server: nginx
Date: Fri, 14 Sep 2012 09:39:57 GMT
Content-Type: text/html; charset=utf-8
Transfer-Encoding: chunked
Connection: keep-alive
Status: 200 OK
Cache-Control: private, max-age=0, must-revalidate
Strict-Transport-Security: max-age=2592000
X-Frame-Options: deny
Set-Cookie: _gh_sess=XYZ; path=/; secure; HttpOnly
```

HTTP Strict Transport Security (HSTS)



PROPOSED STANDARD

Internet Engineering Task Force (IETF)
Request for Comments: 6797
Category: Standards Track
ISSN: 2070-1721

J. Hodges
PayPal
C. Jackson
Carnegie Mellon University
A. Barth
Google, Inc.

November 2012

HTTP Strict Transport Security (HSTS)

Abstract

This specification defines a mechanism enabling web sites to declare themselves accessible only via secure connections and/or for users to be able to direct their user agent(s) to interact with given sites only over secure connections. This overall policy is referred to as HTTP Strict Transport Security (HSTS). The policy is declared by web sites via the Strict-Transport-Security HTTP response header field and/or by other means, such as user agent configuration, for example.

Strict-Transport-Securityヘッダで指定できる値

- max-age=有効期限（秒）
  - ブラウザがそのサイトに対してHTTPS通信のみでアクセスする有効期限を秒単位で指定
  - 今後1年間のhttp://へのアクセスをhttps://に切り替えるには365×24×3600＝31536000を指定

```
HTTP/1.1 200 OK
Server: nginx
Date: Fri, 14 Sep 2012 09:39:57 GMT
Content-Type: text/html; charset=utf-8
Cache-Control: private, max-age=0, must-revalidate
Strict-Transport-Security: max-age=31536000; includeSubdomains
X-Frame-Options: deny
Set-Cookie: _gh_sess=XYZ; path=/; secure; HttpOnly
```

【例】Apache の httpd.conf の設定

```
<VirtualHost *:80>
 # サイト上のすべてのHTTPアクセスをhttps://にリダイレクトする
 ServerAlias *
 RewriteEngine On
 RewriteRule ^(.*)$ https://%{HTTP_HOST}$1 [redirect=301]
</VirtualHost>

<VirtualHost _default_:443>
 # HTTPSサイト上でStrict-Transport-Securityヘッダを出力する
 Header set Strict-Transport-Security ¥
            "max-age=31536000; includeSubDomains"

 # SSL関連の設定
 SSLEngine on
 # ...
</VirtualHost>
```

【例】Apache の httpd.conf の設定

```
<VirtualHost *:80>
 # サイト上のすべてのHTTPアクセスをhttps://にリダイレクトする
 ServerAlias *
 RewriteEngine On
 RewriteRule ^(.*)$ https://%{HTTP_HOST}$1 [redirect=301]
</VirtualHost>

<VirtualHost _default_:443>
 # HTTPSサイト上でStrict-Transport-Securityヘッダを出力する
 Header set Strict-Transport-Security ¥
            "max-age=31536000; includeSubDomains"

 # SSL関連の設定
 SSLEngine on
 # ...
</VirtualHost>
```

Webサイトをすべて SSL化するときは・・・

1. Webサイト上の通信をすべてHTTPSに変更
2. Set-Cookie時にsecure属性をつける
   - HTTPでの盗聴を防止（SSL通信時のみ送信）
3. Set-Cookie時にHttpOnly属性をつける
   - JavaScriptから読み取りを禁止（XSS被害を低減）
4. Cookie の ID を SSL 専用に変えておく
   - ses_id => sess_id（同じIDを使わない）
5. Strict-Transport-Securityヘッダを出力する
   - RFC 6797 (HSTS) を参考に

欠点

HSTS 対応ブラウザ

- Google Chrome 4.0.211.0以降
- Firefox 4以降
- Opera 12以降

- HSTS未対応のブラウザ
  - Internet Explorer
  - Internet Explorer
  - Internet Explorer
    - 大事なことなので３回・・・

ISP様へ
ご提案

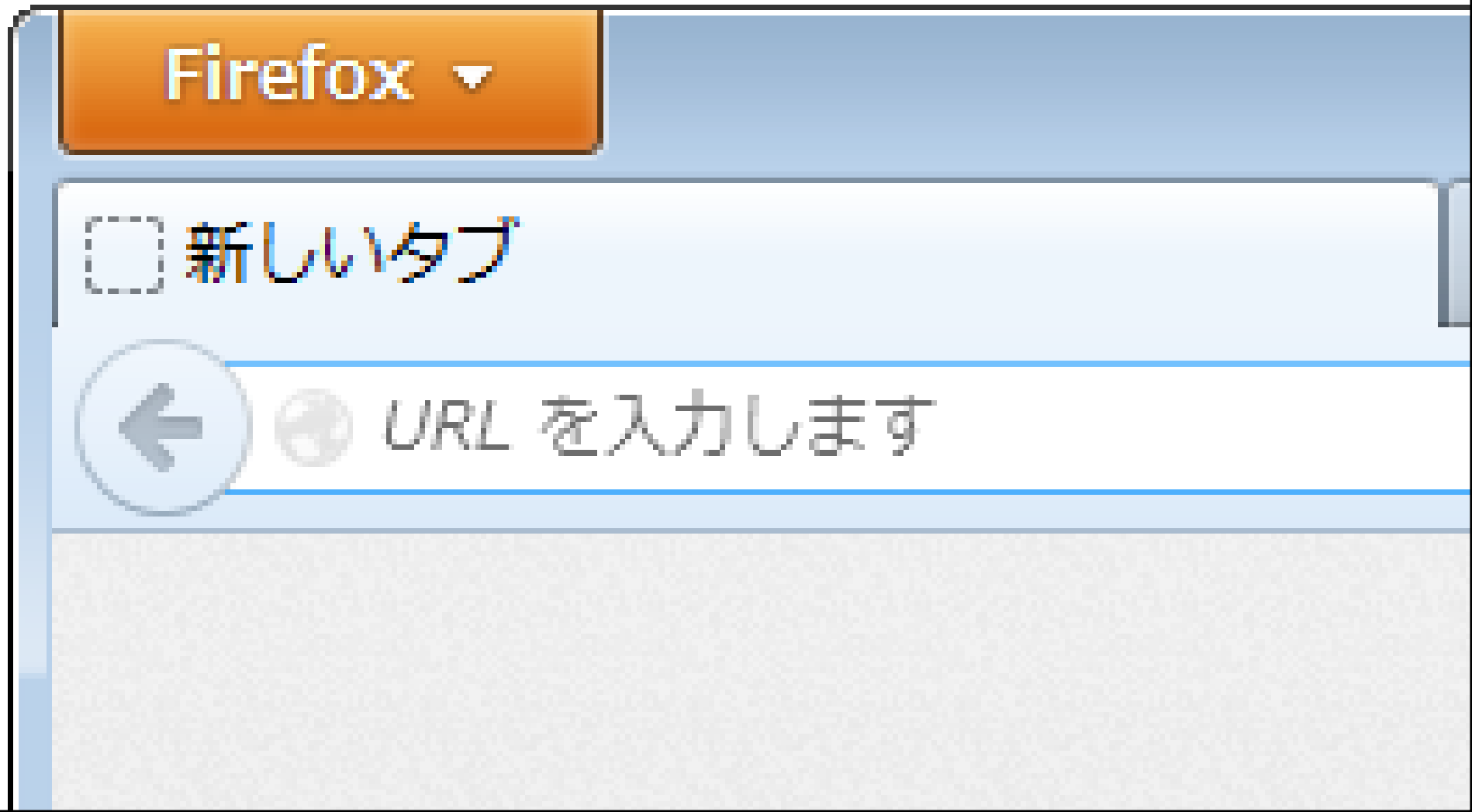
Outbound Port80 Blocking (HTTP)

Outbound Port 80 blocking (OP80B) のご提案

- HTTP (80) 通信をすべてブロック
  - HTTP は信頼できない通信経路を通ることがある
    - 盗聴、DNS汚染、中間者攻撃、ARP spoofing・・・
  - 児童ポルノも完全遮断（リストの管理必要なし）
- 信頼できる SSL/TLS (443) 通信のみを許可
  - Gmail、Google検索、Twitter、facebook、
    - 大手Webサイトは既に https:// に対応している
- WebサイトのSSL化を促す
  - SSLワイルドカード証明書 (*.example.com)
  - SPDY、HTTP/2.0の登場でTLS通信が重要に

利用者として

ブラウザのアドレスバー



何を入力していますか？

何を入力していますか？

JANOGメンバーならhttps://を入力しましょう