

# HSTS for IE and others

～ セキュアな経路をIEにも ～

はせがわようすけ

@hasegawayosuke

<http://j.mp/yosuke>



# 最近の趣味

## ❖ 記号 + 顔文字プログラミング

```
%@"%"@, ~, %, !`_^ [^_^]-;>`_^ [^_^]%"! , ^, :`_^ [^_^]-@{-`{-  
?:`_^ [^_^]-`-`-`-@@`_^ [^_^]-`~`-`-@$`_^ [^_^]-`-`-`-  
@@`_^ [^_^]-`~`-`-@#`_^ [^_^]-+~`-/~`-?;`_^ [^_^]%!~`-;-  
, ;`_^ [^_^]-"$-@~`-@`_^ [^_^]-{[-);-@: `_^ [^_^]-  
/*, %`_^ [^_^]`_^ [^_^]`_^ [^_^]`_^ [^_^]%"@$-@;-?;`_^ [^_^]-  
/~`-&, #`_^ [^_^]-`~`-`-{, *`_^ [^_^]-@@-$!`_^ [^_^]-  
:$, [, <`_^ [^_^]-!|-.) , !`_^ [^_^]-@{-@`-  
/(`_^ [^_^]`_^ [^_^]`_^ [^_^]`_^ [^_^]-{!-{. , .`_^ [^_^]-~/`-  
/~`_^ [^_^]%"-}@$`_^ [^_^]%"@@-!/, !`_^ [^_^]-:*`-  
=%`_ [ [ [ [ [ [ [ `^ ^ ^ ^ ^ ^ -%+) @@ ^ ^ ^ !; @@_!, ((, . ((-$+) @*+@!!@-  
, !" (+@@, $-, !" ($%&, &, &_&, "@'" %_&'" , &$&-@*@$"
```

# IE9で動かない

- ❖ JavaScriptでバイナリを生成、ダウンロードさせたい
- ❖ IEはHTML埋め込みのdataスキームのみ対応

```
window.open( "data:application/octet-stream,..." );
```

- ❖ 将来的にはHTML5 FileAPI:Writer対応

がんばれIE9! 

# 今日の話題

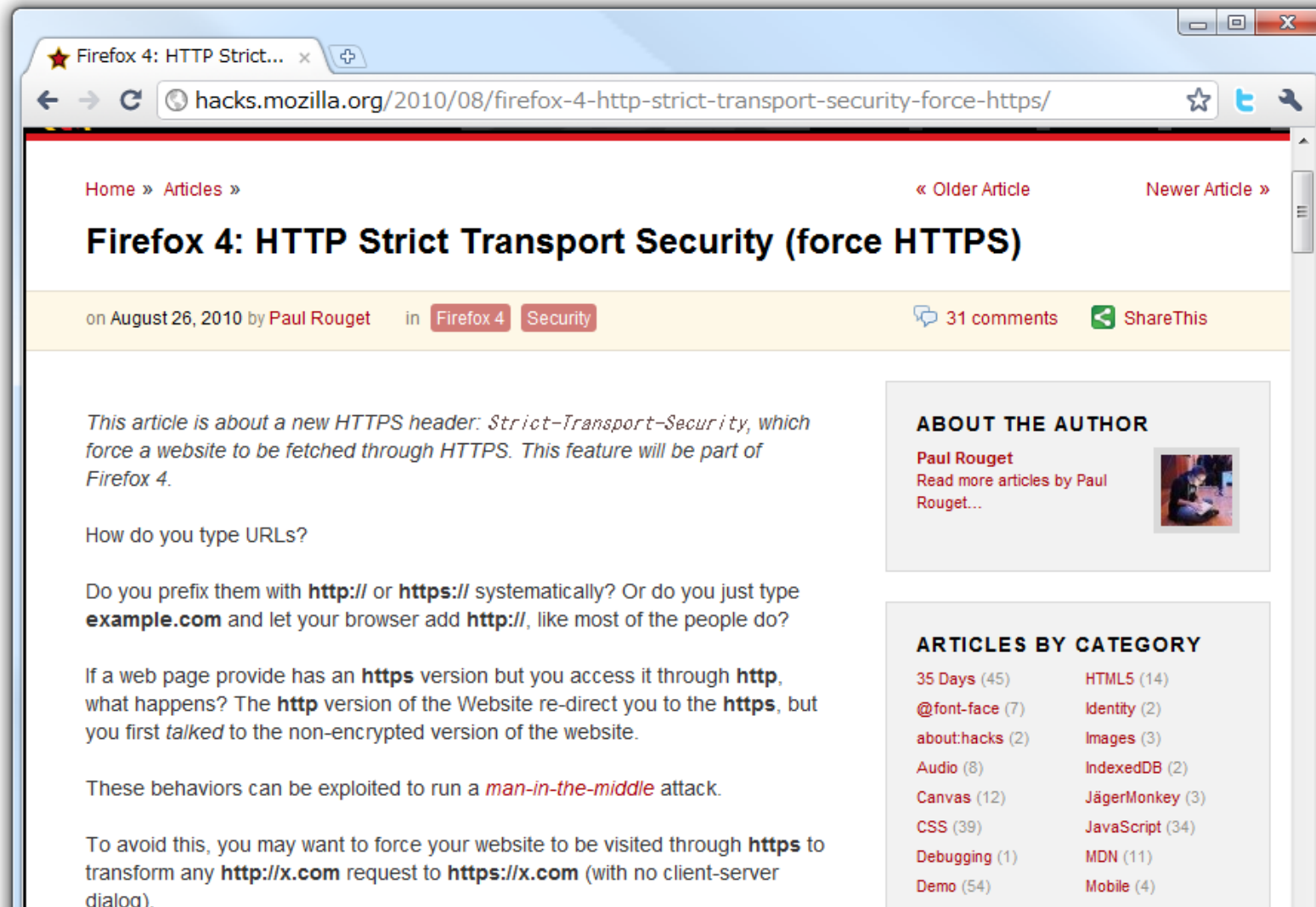
# HSTS

- **H**TT**S**T**S** **S**trict **T**ransport **S**ecurity -

# HSTS - HTTP Strict Transport Security

- ❖ HTTPとHTTPS、両方を提供しているサイトで、HTTPSの使用を強制する機能
- ❖ 中間者攻撃の低減に

# HSTS - HTTP Strict Transport Security



Home » Articles » « Older Article Newer Article »

## Firefox 4: HTTP Strict Transport Security (force HTTPS)

on August 26, 2010 by Paul Rouget in [Firefox 4](#) [Security](#) 31 comments [ShareThis](#)

*This article is about a new HTTPS header: `Strict-Transport-Security`, which force a website to be fetched through HTTPS. This feature will be part of Firefox 4.*

How do you type URLs?

Do you prefix them with **http://** or **https://** systematically? Or do you just type **example.com** and let your browser add **http://**, like most of the people do?


If a web page provide has an **https** version but you access it through **http**, what happens? The **http** version of the Website re-direct you to the **https**, but you first *talked* to the non-encrypted version of the website.

These behaviors can be exploited to run a *man-in-the-middle* attack.

To avoid this, you may want to force your website to be visited through **https** to transform any **http://x.com** request to **https://x.com** (with no client-server dialog).

### ABOUT THE AUTHOR

**Paul Rouget**  
Read more articles by Paul Rouget...



### ARTICLES BY CATEGORY

<a href="#">35 Days</a> (45)	<a href="#">HTML5</a> (14)
<a href="#">@font-face</a> (7)	<a href="#">Identity</a> (2)
<a href="#">about:hacks</a> (2)	<a href="#">Images</a> (3)
<a href="#">Audio</a> (8)	<a href="#">IndexedDB</a> (2)
<a href="#">Canvas</a> (12)	<a href="#">JägerMonkey</a> (3)
<a href="#">CSS</a> (39)	<a href="#">JavaScript</a> (34)
<a href="#">Debugging</a> (1)	<a href="#">MDN</a> (11)
<a href="#">Demo</a> (54)	<a href="#">Mobile</a> (4)

# HSTS - HTTP Strict Transport Security

HTTPSでのレスポンスヘッダで以下を返す

```
Strict-Transport-Security: max-age=15768000
```

```
Strict-Transport-Security: max-age=15768000 ; includeSubDomains
```

- ▶ これ以降のHTTPへのアクセスはHTTPSに置き換わる
- ▶ max-age は有効期間を秒数で指定
- ▶ includeSubDomainsが指定されるとサブドメインも対象

# HSTS - HTTP Strict Transport Security

- ❖ HTTPSサイトのみがStrict-Transport-Securityを返す

```
Strict-Transport-Security: max-age=15768000
```

- ❖ HTTPはすでに汚染されているかもしれないので
- ❖ Google Chrome, Firefox 4以降で対応

**IEでも  
HSTS対応したい!**

**Fiddlerの出番！**

# Fiddler

- ❖ MicrosoftのEric Lawrence氏によるProxy型のHTTPデバッガ
- ❖ JScriptによる柔軟なカスタマイズ
- ❖ HTTPSにも対応



<http://www.fiddler2.com/fiddler2/>

Web Sessions

#	Result	Protocol	Host	URL
40	200	HTTP	CONNECT	jottit.com:443
41	200	HTTPS	jottit.com	/?D
42	200	HTTPS	jottit.com	/?D
45	200	HTTPS	CONNECT	jottit.com:443
46	200	HTTPS	jottit.com	/?XXX
47	200	HTTPS	jottit.com	/?XXX
48	200	HTTPS	jottit.com	/?XX

ALT+Q > type HELP...

Request Builder FiddlerScript Log Filters Timeline  
 Statistics Inspectors AutoResponder  
 Headers TextView WebForms HexView Auth Raw

XML

```
GET https://jottit.com/?XX HTTP/1.1
Accept: image/gif, image/jpeg, image/pjpeg, application/
Accept-Language: ja-JP,ja;q=0.7,en-US;q=0.3
User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows N
Accept-Encoding: gzip, deflate
Proxy-Connection: Keep-Alive
Host: jottit.com
```

Find... View in Notepad

Transformer Headers TextView SyntaxView ImageView  
 HexView WebView Auth Caching Privacy Raw XML

```
HTTP/1.1 200 OK
Strict-Transport-Security: max-age=16070400; includes
Content-Type: text/html; charset=utf-8
Content-Length: 2393
Date: Wed, 15 Sep 2010 08:21:00 GMT
Server: lighttpd/1.4.26

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transiti
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.
<html xmlns="http://www.w3.org/1999/xhtml" lang="en" :
<head>
  <meta http-equiv="Content-Type" content="text/hm
  <link rel="shortcut icon" href="/favicon.ico" />
  <link rel="stylesheet" href="/static/css-generat
</head>
<title>Jottit</title>

<style type="text/css">
  .c3d8e9481b88748abff0a11a378b86435 { visibility: i
  .c411a2f6a435ae77edd63df6102bb1b3a { display: non
</style>
</head>
```

Find... View in Notepad

# Fiddler

- ❖ MicrosoftのEric Lawrence氏によるProxy型のHTTPデバッガ
- ❖ JScriptによる柔軟なカスタマイズ
- ❖ HTTPSにも対応

# Fiddler - Customize Rules

```
class Handlers
{
    private static var m_hsts = new Array();
    ....
    static function OnBeforeResponse(oSession: Session)
    {
        ....
        if( (oSession.isHTTPS) &&
            oSession.oResponse.headers.Exists( "Strict-Transport-Security" ) )
        {
            (function() {
                var s : String = oSession.oResponse[ "Strict-Transport-Security" ];
                var m = /^max-age=s*=s*(\d+)*s*(;s*(includeSubDomains)s*)*/.exec( s );
                if( m !== null ) {
                    m_hsts[ oSession.host ] = {
                        "timer" : +(new Date()) + ( m[ 1 ] ? m[ 1 ] : 0 ) * 1000,
                        "includeSubDomains" : m[ 3 ] ? true : false
                    };
                }
            })();
        }
        ....
    }
}
```

# Fiddler - Customize Rules

```
class Handlers
{
    private static var m_hsts = new Array();
    ....
    static function OnBeforeRequest(oSession: Session)
    {
        ....
        if( (undefined !== m_hsts[ oSession.host ]) &&
            (oSession.oRequest.headers.UriScheme=="http") )
        {
            (function() {
                var t1 = +(new Date());
                var t2 = m_hsts[ oSession.host ].timer;
                if( t2 >= t1 ) {
                    oSession.oRequest.headers.UriScheme = "https";
                }
            })();
        }
        ....
    }
}
```

# Fiddler

- ❖ RuleのカスタマイズでHSTS対応できた!  
(includeSubDomainsは非対応)
- ❖ IEの表示に問題が...  
実際にはHTTPSなのにアドレスバーが...



# まとめ

- ❖ Fiddler 超便利!!  
なんでもできるよ。
  - ❖ ドキュメントがほとんどないけど。
- ❖ IE9もHSTS対応してね。

❖ 質問などは → はせがわようすけ  
@hasegawayosuke  
<http://j.mp/yosuke>

