



Cross-Site Request Forgery: Danger, Detection, and Defenses

Eric Sheridan
Aspect Security, Inc.
eric.sheridan@aspectsecurity.com

OWASP

11-14-2007

Copyright © The OWASP Foundation
Permission is granted to copy, distribute and/or modify this document
under the terms of the OWASP License.

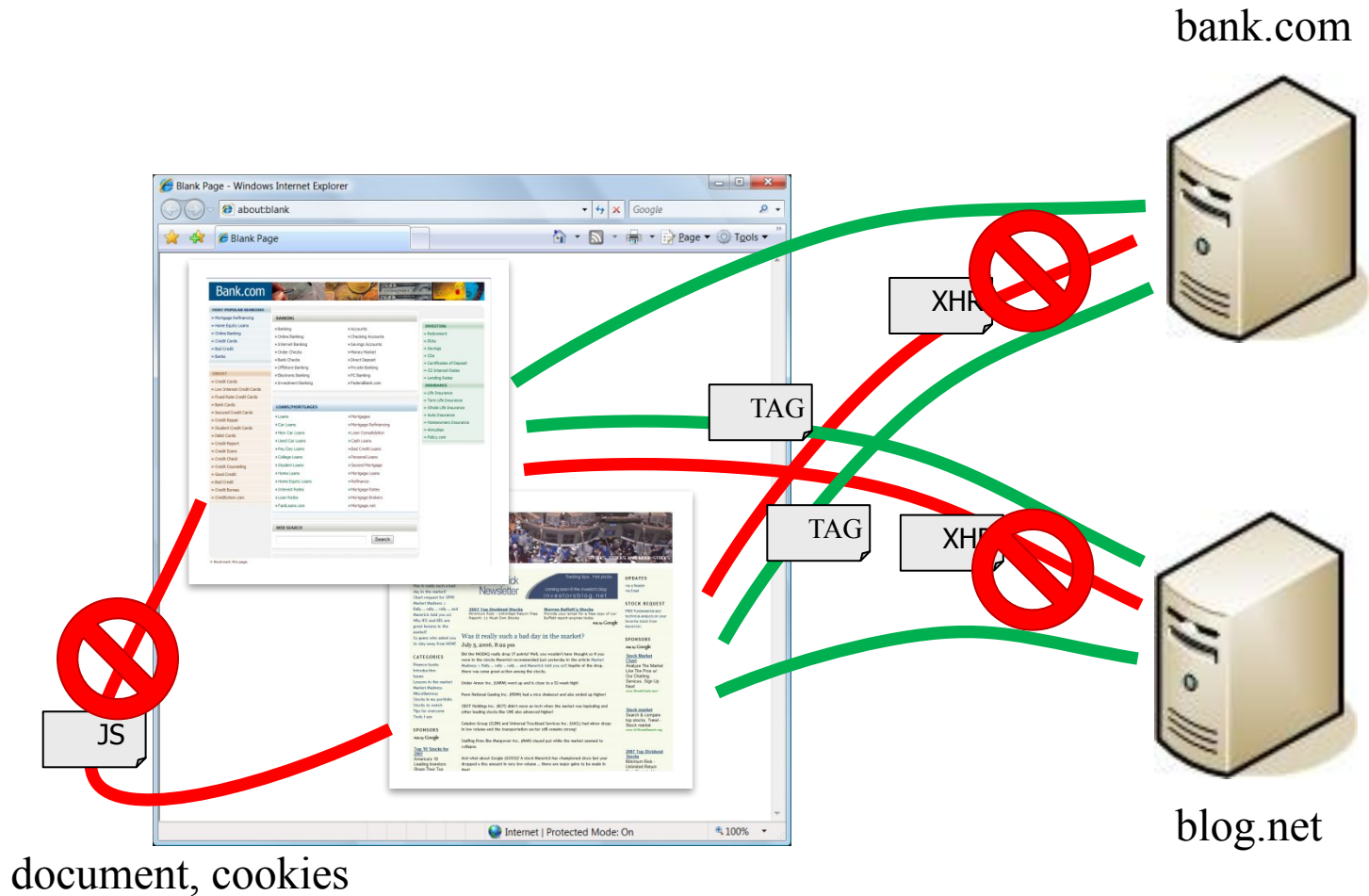
The OWASP Foundation

<http://www.owasp.org>

Overview

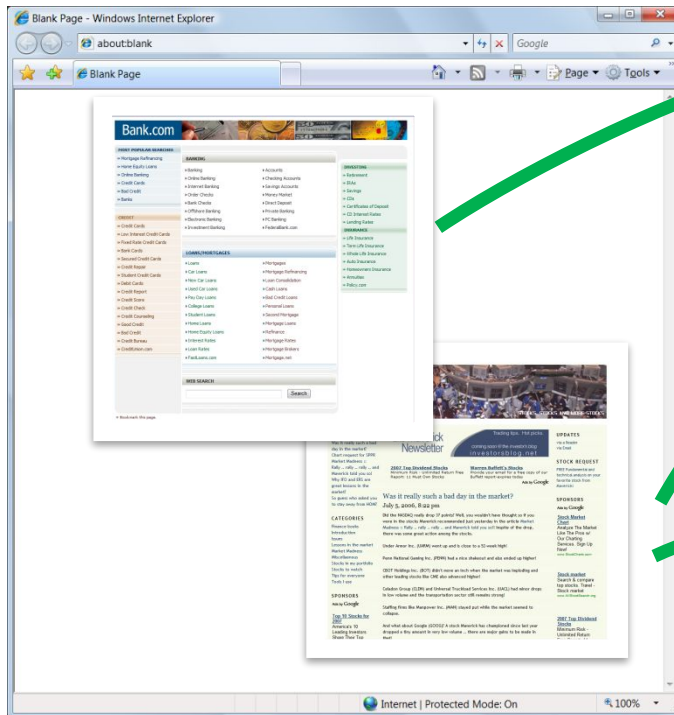
- < Discussion of the “Same Origin Policy”
- < Overview of the “Sleeping Giant”
- < The Introduction of
 - 4 2 New OWASP Tools
 - 4 A Series of New WebGoat Labs
- < Enterprise CSRF Mitigation Strategy

The Browser "Same Origin" Policy



Cross-Site Request Forgery

bank.com



- Go to Transfer Assets
 - Select FROM Fund
 - Select TO Fund
 - Select Dollar Amount
 - Submit Transaction
 - Confirm Transaction
- <https://bank.com/fn?param=1>



attacker's post at blog.net

How Does CSRF Work?

< Tags

```
  
<iframe src="https://bank.com/fn?param=1">  
<script src="https://bank.com/fn?param=1">
```

< Autoposting Forms

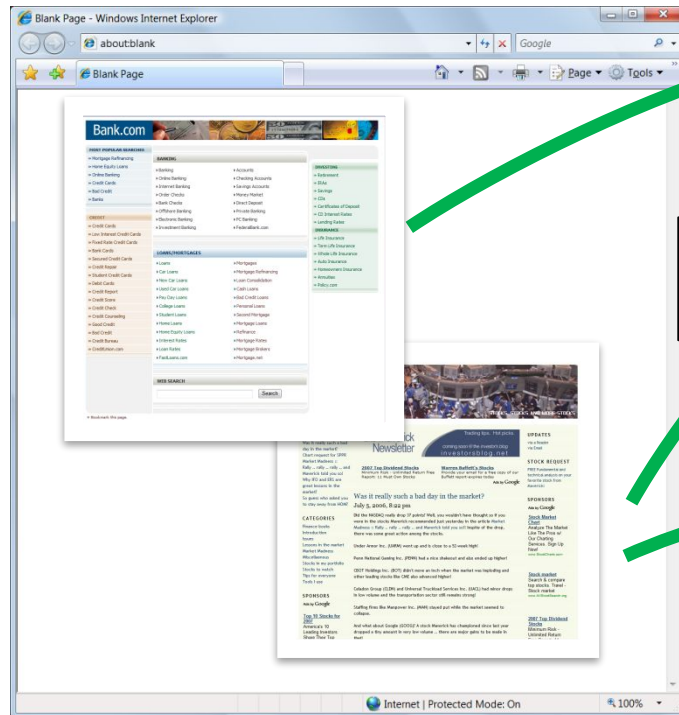
```
<body onload="document.forms[0].submit()">  
<form method="POST" action="https://bank.com/fn">  
  <input type="hidden" name="sp" value="8109"/>  
</form>
```

< XmlHttpRequest

4 Subject to same origin policy

Credentials Included

bank.com



<https://bank.com/fn?param=1>
JSESSIONID=AC934234...



blog.net

New Tool: OWASP CSRFTester

< Test your applications for CSRF

- 4 Record and replay transactions
- 4 Tune the recorded test case
- 4 Run test case with exported HTML document

< Test case alternatives

- 4 Auto-Posting Forms
- 4 Evil iFrame
- 4 IMG Tag
- 4 XMLHttpRequest
- 4 Link

DEMO: OWASP CSRFTester

The screenshot shows the OWASP CSRFTester application window. The title bar reads "OWASP CSRFTester". Below the title bar is a menu bar with "File" and "Options". The main area contains the text "OWASP CSRFTester" and two buttons: "Clear All" and "Stop Recording".

Step	Method	URL	Parameters	Pause
Request 18	GET	http://www.google-anal...		63
Request 19	GET	http://www.owasp.org:...		15
Request 25	GET	http://www.owasp.org:...		125
Request 28	GET	http://www.owasp.org:...		312
Request 29	GET	http://www.owasp.org:...		31
Request 32	GET	http://www.owasp.org:...		62
Request 33	GET	http://www.owasp.org:...		109
Request 34	GET	http://www.owasp.org:...		109
Request 36	GET	http://www.google-anal...		78
Request 37	GET	http://www.google.com...		94
Request 39	GET	http://www.google.com...		109

Below the table, the details for Request 36 are shown. The method is "GET" and the URL is "http://www.google-analytics.com:80/___utm.gif".

Query Parameters:

- utmw=1
- utmn=524956485
- utmcs=UTF-8
- utmsr=1280x1024

Form Parameters:

Include Regex: *

Exclude Regex: /\.(gif|jpg|png|css|ico|js|axdl|?\.ico)\$

Report Type: Forms iFrame IMG XHR Link

Display in Browser

Buttons: Reset, Reset, Generate HTML

Footer: Moving to row 21

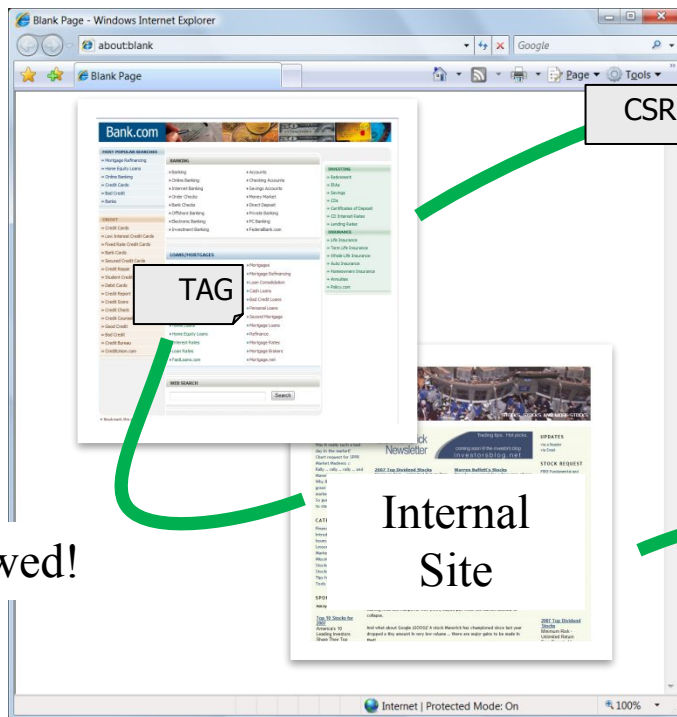
What Can Attackers Do with CSRF?

- < Anything an authenticated user can do
 - 4 Click links
 - 4 Fill out and submit forms
 - 4 Follow all the steps of a wizard interface
- < No restriction from same origin policy, except...
 - 4 Attackers cannot read responses from other origins
 - 4 Limited on what can be done with data
- < Severe impact on accountability
 - 4 Log entries reflect the actions a victim was tricked into executing

Using CSRF to Attack Internal Pages

internal browser

attacker.com



Allowed!

Internal Site



internal.mybank.com

Misconceptions – Defenses That Don't Work

< Only accept POST

- 4 Stops simple link-based attacks (IMG, frames, etc.)
- 4 But hidden POST requests can be created with frames, scripts, etc...

< Referer checking

- 4 Some users prohibit referers, so you can't just require referer headers
- 4 Techniques to selectively create HTTP request without referers exist

< Requiring multi-step transactions

- 4 CSRF attack can perform each step in order

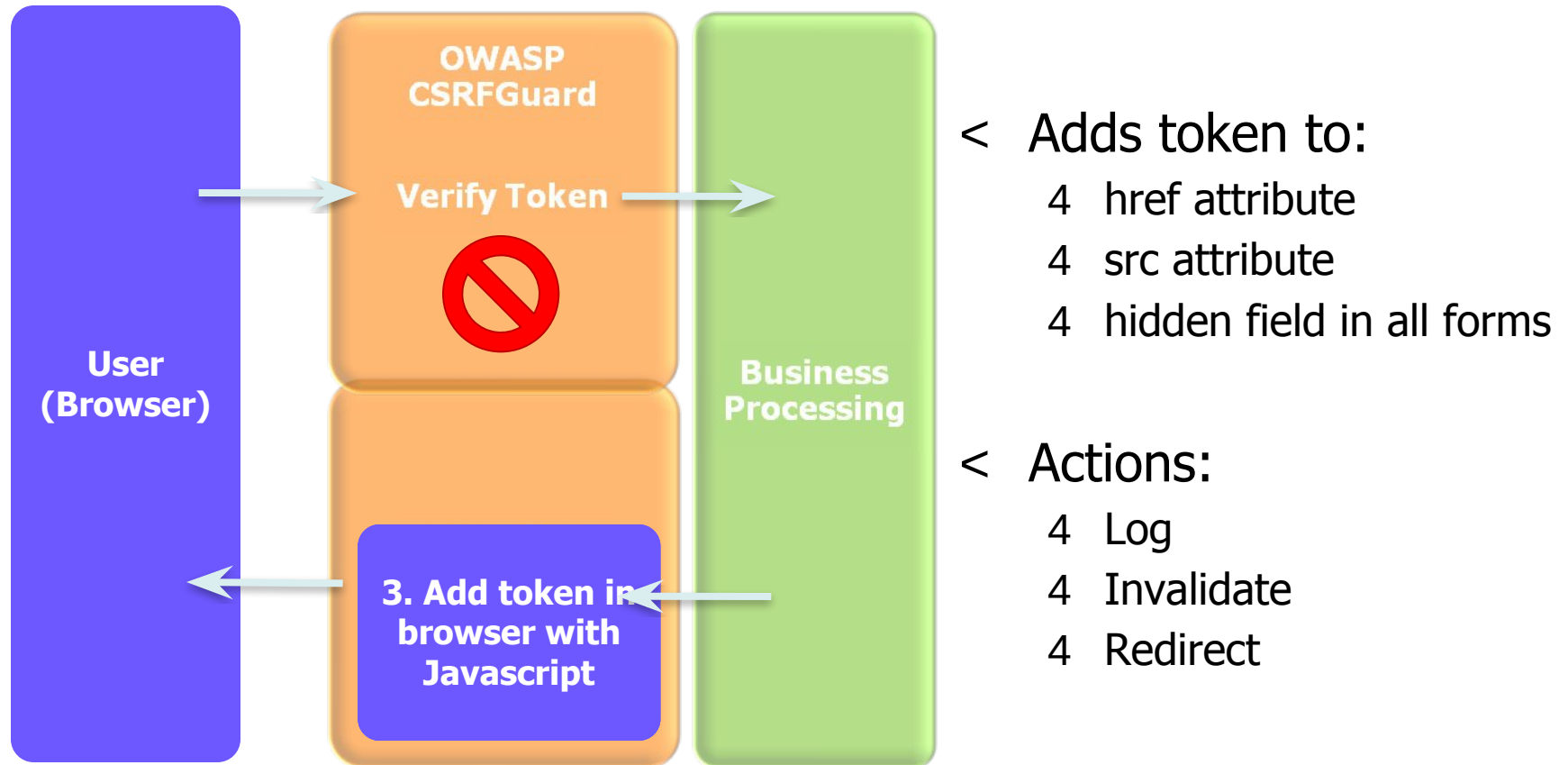
< URL Rewriting

- 4 General session id exposure in logs, cache, etc.

None of these approaches will sufficiently protect against CSRF!

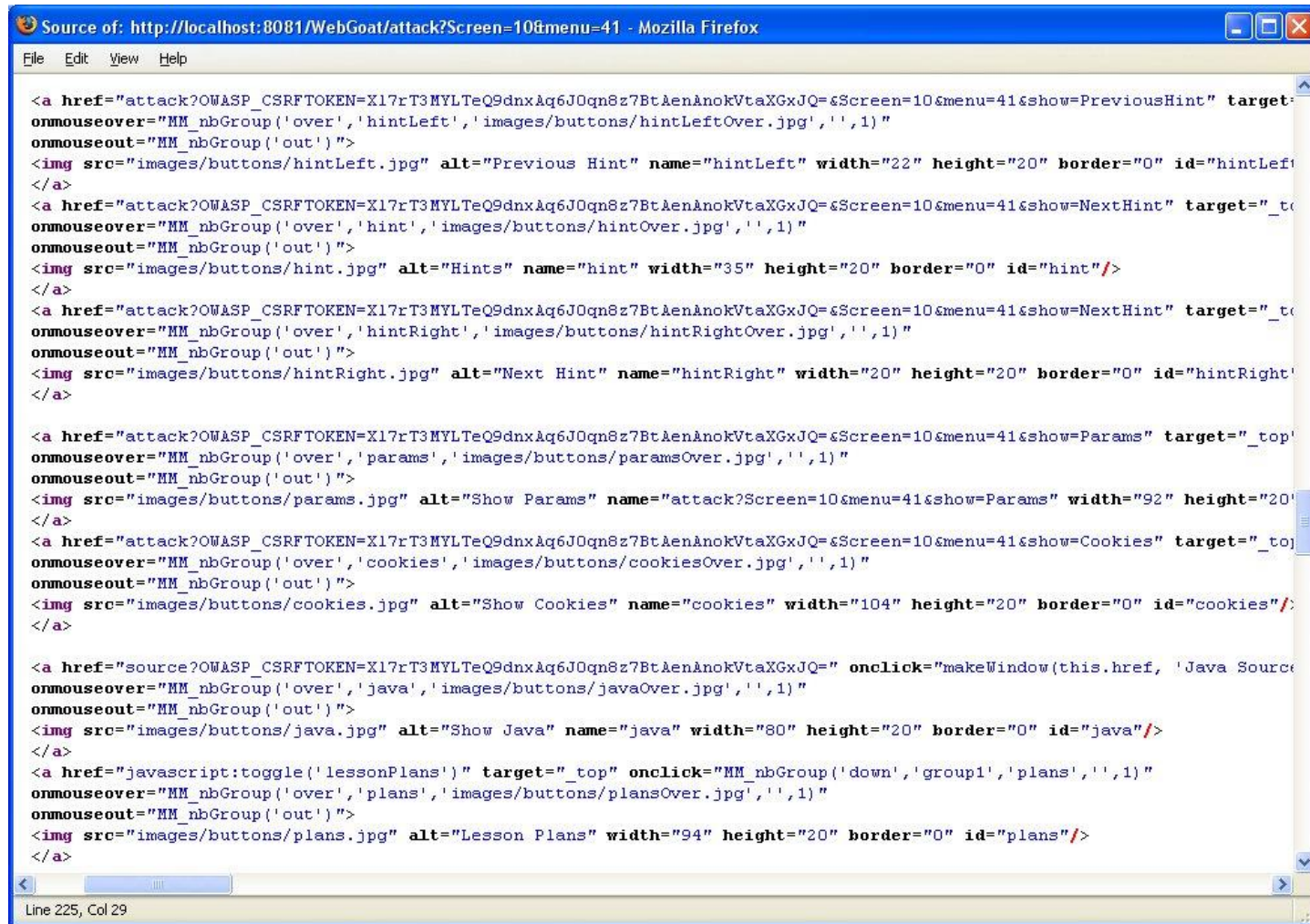


New Tool: OWASP CSRFGuard 2.0



<http://www.owasp.org/index.php/CSRFGuard>

DEMO: OWASP CSRFGuard 2.0



```
Source of: http://localhost:8081/WebGoat/attack?Screen=10&menu=41 - Mozilla Firefox
File Edit View Help

<a href="attack?OWASP_CSRFTOKEN=X17rT3MYLTeQ9dnxAq6J0qn8z7BtAenAnokVtaXGxJQ=&Screen=10&menu=41&show=PreviousHint" target="_top"
onmouseover="MM_nbGroup('over','hintLeft','images/buttons/hintLeftOver.jpg',' ',1)"
onmouseout="MM_nbGroup('out')">

</a>
<a href="attack?OWASP_CSRFTOKEN=X17rT3MYLTeQ9dnxAq6J0qn8z7BtAenAnokVtaXGxJQ=&Screen=10&menu=41&show=NextHint" target="_top"
onmouseover="MM_nbGroup('over','hint','images/buttons/hintOver.jpg',' ',1)"
onmouseout="MM_nbGroup('out')">

</a>
<a href="attack?OWASP_CSRFTOKEN=X17rT3MYLTeQ9dnxAq6J0qn8z7BtAenAnokVtaXGxJQ=&Screen=10&menu=41&show=NextHint" target="_top"
onmouseover="MM_nbGroup('over','hintRight','images/buttons/hintRightOver.jpg',' ',1)"
onmouseout="MM_nbGroup('out')">

</a>

<a href="attack?OWASP_CSRFTOKEN=X17rT3MYLTeQ9dnxAq6J0qn8z7BtAenAnokVtaXGxJQ=&Screen=10&menu=41&show=Params" target="_top"
onmouseover="MM_nbGroup('over','params','images/buttons/paramsOver.jpg',' ',1)"
onmouseout="MM_nbGroup('out')">

</a>
<a href="attack?OWASP_CSRFTOKEN=X17rT3MYLTeQ9dnxAq6J0qn8z7BtAenAnokVtaXGxJQ=&Screen=10&menu=41&show=Cookies" target="_top"
onmouseover="MM_nbGroup('over','cookies','images/buttons/cookiesOver.jpg',' ',1)"
onmouseout="MM_nbGroup('out')">

</a>

<a href="source?OWASP_CSRFTOKEN=X17rT3MYLTeQ9dnxAq6J0qn8z7BtAenAnokVtaXGxJQ=" onclick="makeWindow(this.href, 'Java Source Code', 'source', ' ', 1)"
onmouseover="MM_nbGroup('over','java','images/buttons/javaOver.jpg',' ',1)"
onmouseout="MM_nbGroup('out')">

</a>
<a href="javascript:toggle('lessonPlans')" target="_top" onclick="MM_nbGroup('down','group1','plans',' ',1)"
onmouseover="MM_nbGroup('over','plans','images/buttons/plansOver.jpg',' ',1)"
onmouseout="MM_nbGroup('out')">

</a>

Line 225, Col 29
```

Similar Implementations

< PHP CSRFGuard

- 4 PHP Implementation of CSRFGuard

- 4 http://www.owasp.org/index.php/PHP_CSRF_Guard

< JSCK

- 4 PHP & JavaScript implementation

- 4 <http://www.thespanner.co.uk/2007/10/19/jsck/>

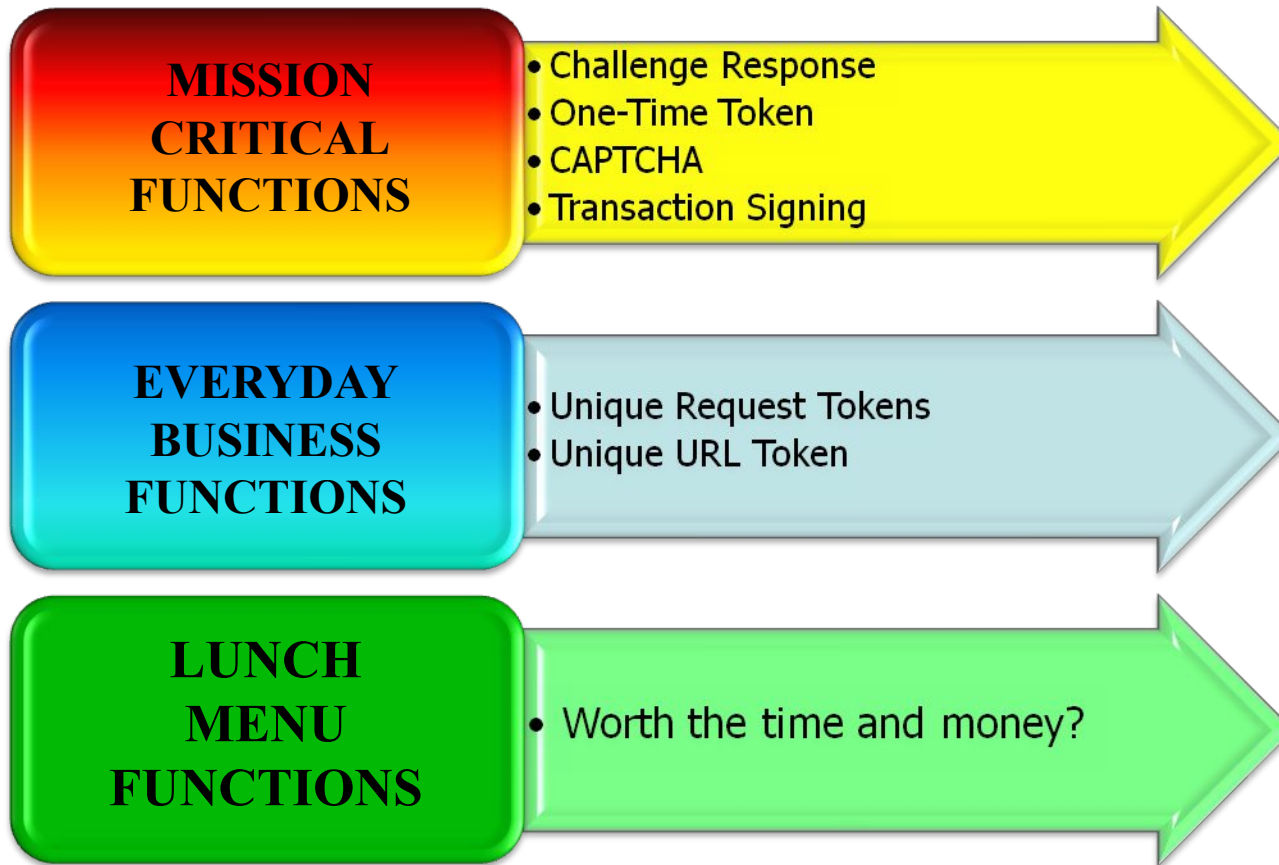
DEMO: Cross-Site Scripting vs. CSRFGuard



Enterprise CSRF Mitigation Strategy



< Balance Between Security, Usability, and Cost



QUESTIONS & ANSWERS

http://www.owasp.org/index.php/Cross-Site_Request_Forgery

<http://www.cgisecurity.com/articles/csrf-faq.shtml>

http://www.darkreading.com/document.asp?doc_id=107651&WT.svl=news1_2

Extra: How Widespread Are CSRF Holes?

- < Very likely in most web applications
 - 4 Including both intranet and external apps
 - 4 Including Web 1.0 and Web 2.0 applications
 - 4 Any function without specific CSRF defenses is vulnerable

- < How do victims get attacked?
 - 4 Victim simply opens an infected webpage, HTML file, or email
 - 4 Single Sign On (SSO) extends “authenticated user”

- < CSRF recently found in 8 security appliances
 - 4 Including CheckPoint

Extra: Real World CSRF Examples



```
<iframe style="display:none"  
src="http://www.google.com/setpre  
fs?hl=xx-klingon&submit2=Save  
%20Preferences%20&prev=http:/  
/www.google.com/&q=&submit=  
Save%20Preferences%20"></iframe>
```



```
<img  
src=http://www.netflix.com/AddTo  
Queue?  
movieid=70011204 width="1"  
height="1" border="0">
```

Extra: CSRF Defenses

< CAPTCHA

- 4 Attacker must know CAPTCHA answer
- 4 Assuming a secure implementation

< Re-Authentication

- 4 Password Based
 - Attacker must know victims password
 - If password is known, then game over already!
- 4 One-Time Token
 - Attacker must know current token
 - Very strong defense!

< Unique Request Tokens

- 4 Attacker must know unique request token for particular victim for particular session
- 4 Assumes token is cryptographically secure and not disclosed.
 - `/accounts?auth=687965fdfaew87agrde ...`

