REST FOR THE WICKED

Bryan Sullivan

Security Program Manager, Microsoft

Cross-Site Request Forgery (XSRF)

- Silent but deadly!
- Can be a page you’ve visited hundreds of times before and trust
- Not the same as an XSS attack

Cross-Site Request Forgery in action

DEMO

SOAP on the ropes?

RESTful services or products available from all of these companies:

- Amazon
- Google
- MySpace
- Flickr
- Microsoft
- Yahoo

But why?

SOAP vs REST

COMPARISON

RESTful web services

GET /movies HTTP/1.1

GET /movies/Wanted HTTP/1.1

GET /movies/year(2008) HTTP/1.1

POST /movies/Wanted/review HTTP/1.1

PUT /movies/Wanted/review HTTP/1.1

DELETE /movies/Indiana_Jones_4 HTTP/1.1

“Pseudo”-REST

GET /movies/Wanted&**action=read** HTTP/1.1

GET /movies/Wanted/review&**action=update** HTTP/1.1

GET /movies/Wanted/review&**action=insert** HTTP/1.1

GET /movies/Indiana_Jones_4&**action=delete** HTTP/1.1

- As is, trivially exploitable by XSRF attacks

POST-based XSRF

```
<body
   onload=javascript:document.evil.submit()>

<form name="evil" method="POST"
   action="http://www.bank.com/transfer" >
   <input name="transfer_to" value="bryan"/>
   <input name="amount" value="10000"/>
</form>
```

PUT and DELETE

Verb	Exploitable?
POST	Yes
GET	?
PUT	?
DELETE	?

Access Control for Cross-Site Requests

http://www.w3.org/TR/access-control/

Access-control: allow <www.good.com>

GET

Verb	Exploitable?
POST	Yes
PUT	Yes
DELETE	Yes
GET	?

Crossdomain.xml strikes again

```
<cross-domain-policy>
   <allow-access-from domain="*"/>
</cross-domain-policy>
```

- XSRF through Flash via URLRequest
- XSRF through Silverlight via WebClient

The <script src> hole

```
<script src="http://www.evil.com/script.js"></script>
```

Attacking RESTful GETs

JavaScript function redefinition

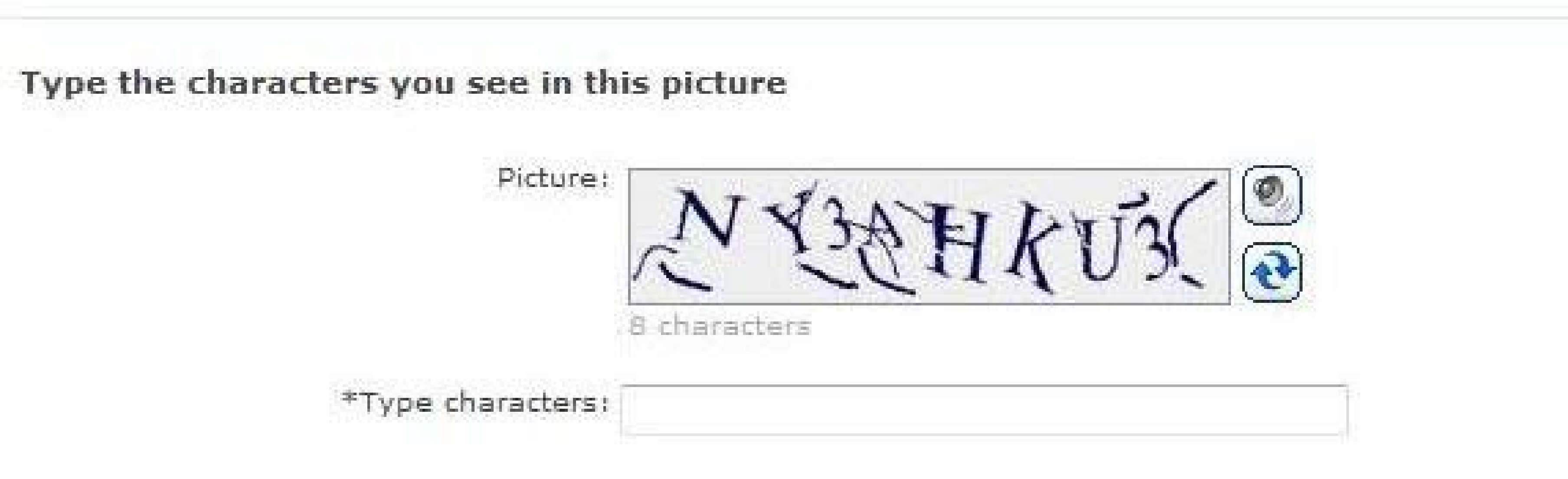
+

Cross-Site Request Forgery

=

JSON hijacking

CAPTCHA defense



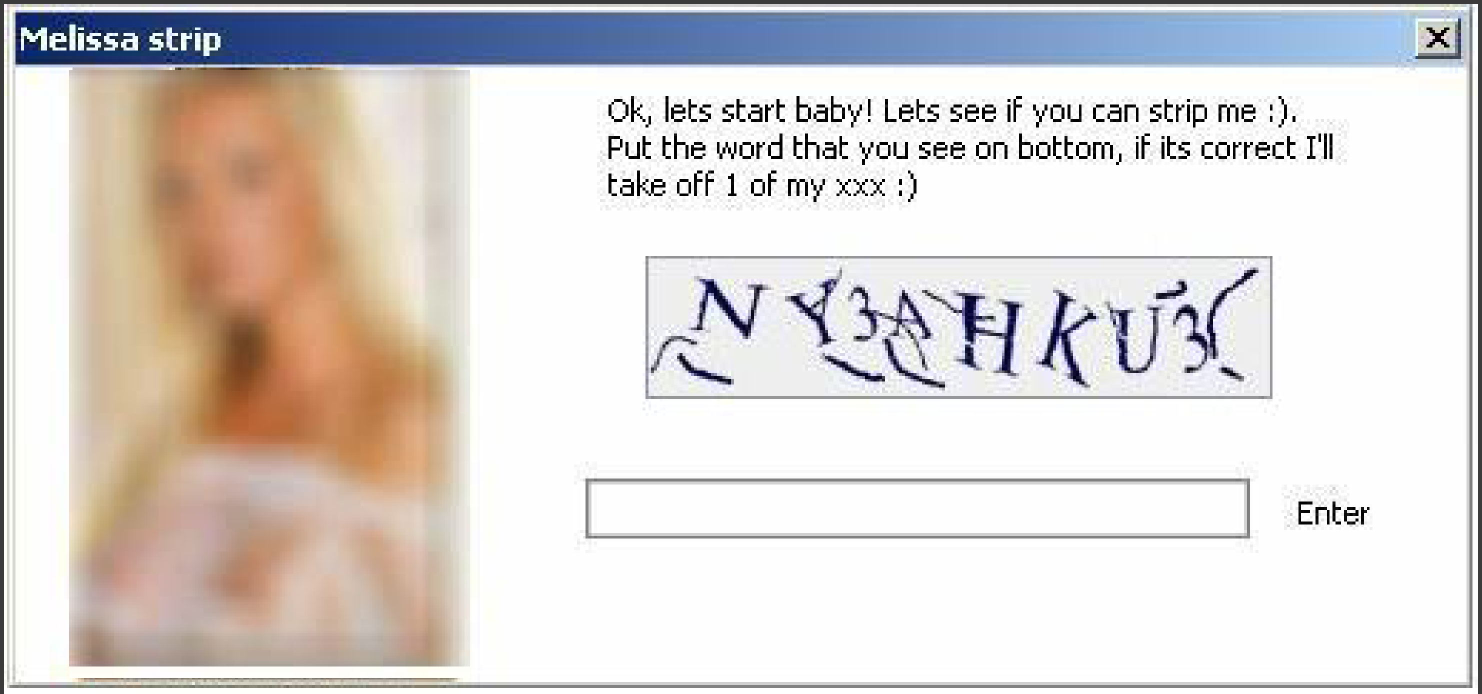
Type the characters you see in this picture

Picture:

8 characters

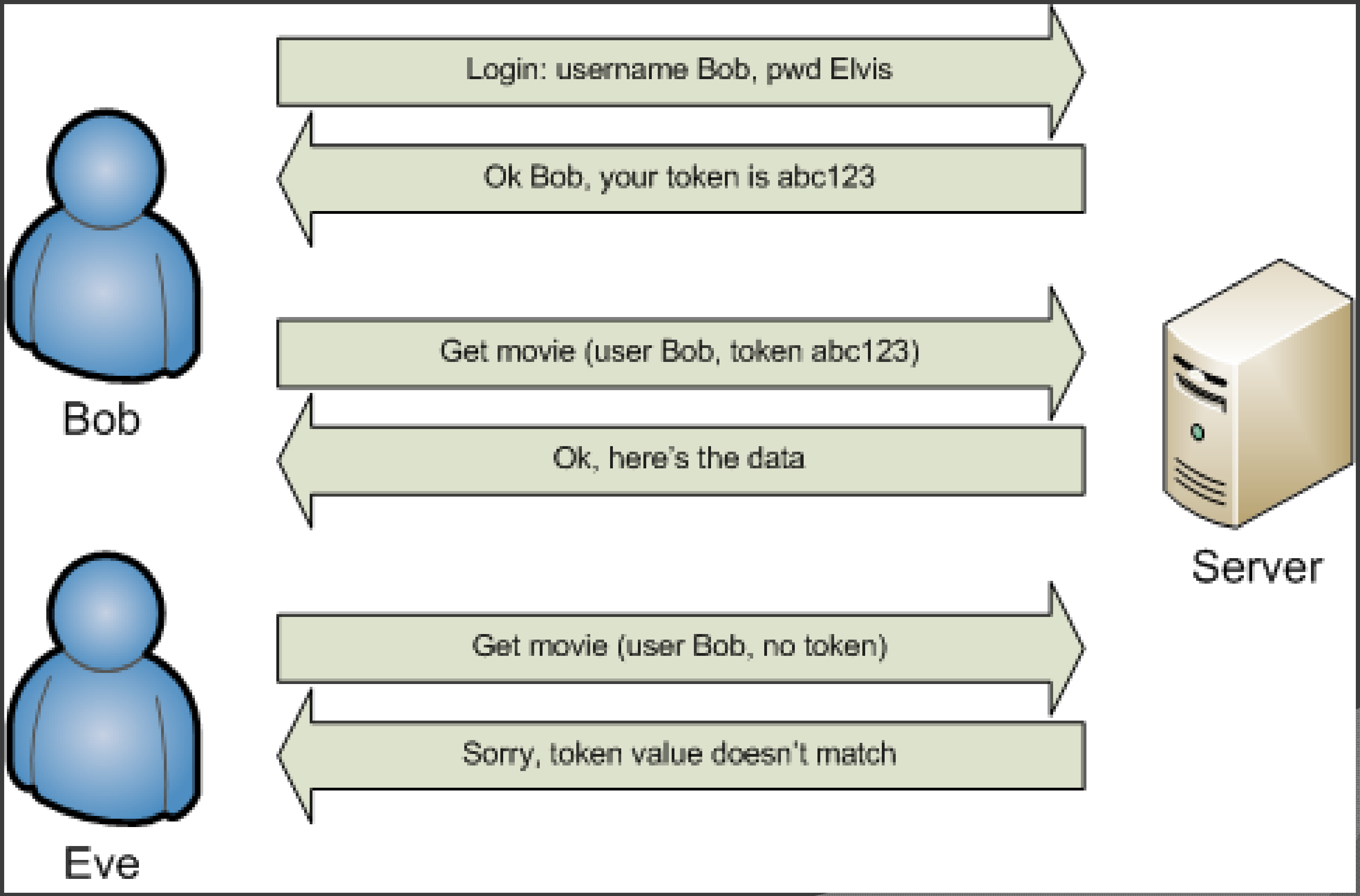
*Type characters:

Breaking CAPTCHAs



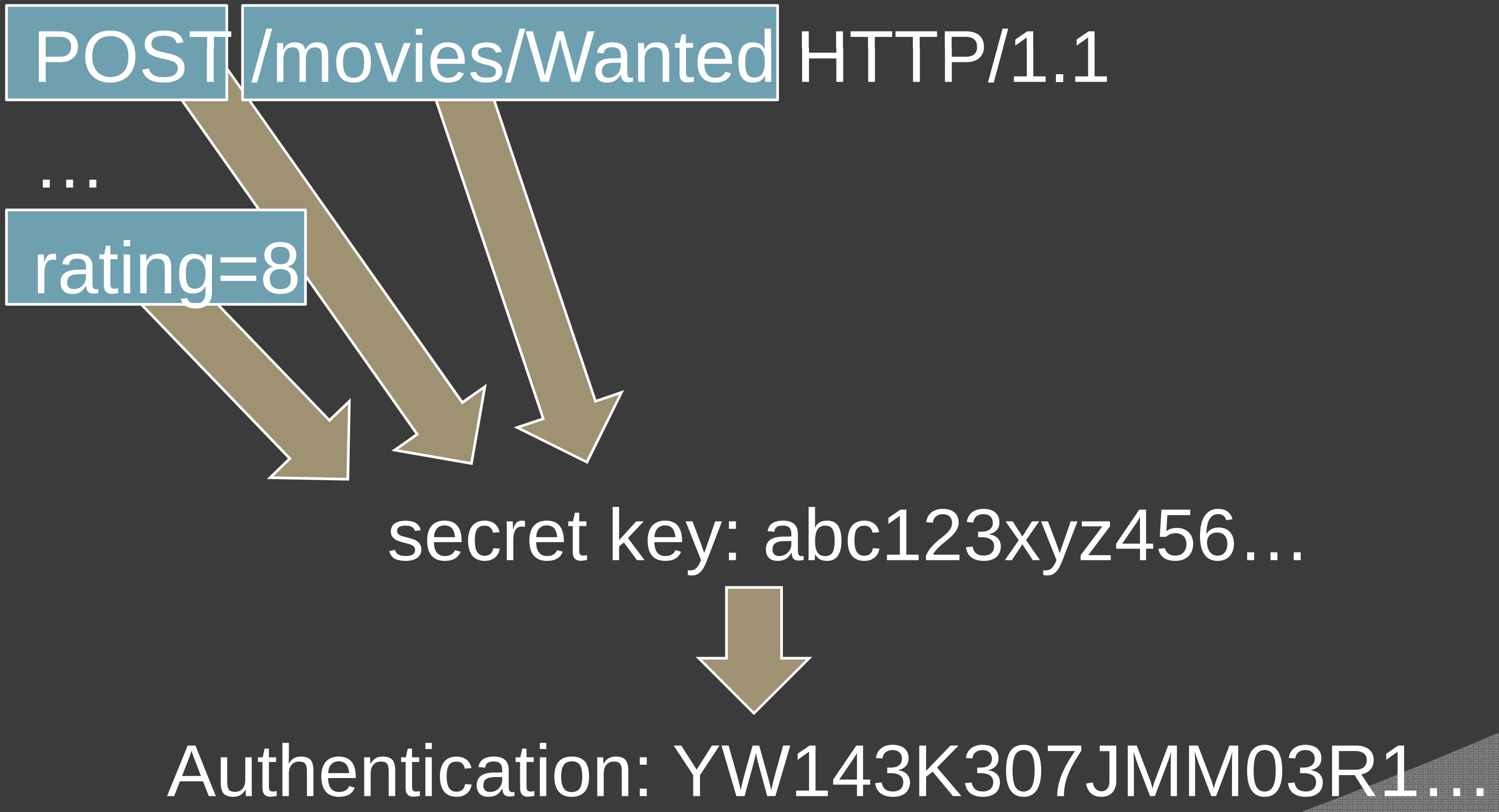
Ok, lets start baby! Lets see if you can strip me :).
Put the word that you see on bottom, if its correct I'll
take off 1 of my xxx :)

Canary defense



Login: username Bob, pwd Elvis
Ok Bob, your token is abc123
Bob
Get movie (user Bob, token abc123)
Ok, here's the data
Server
Get movie (user Bob, no token)
Sorry, token value doesn't match
Eve

Message Authentication Code defense



POST /movies/Wanted HTTP/1.1

…

rating=8

secret key: abc123xyz456…

Authentication: YW143K307JMM03R1…

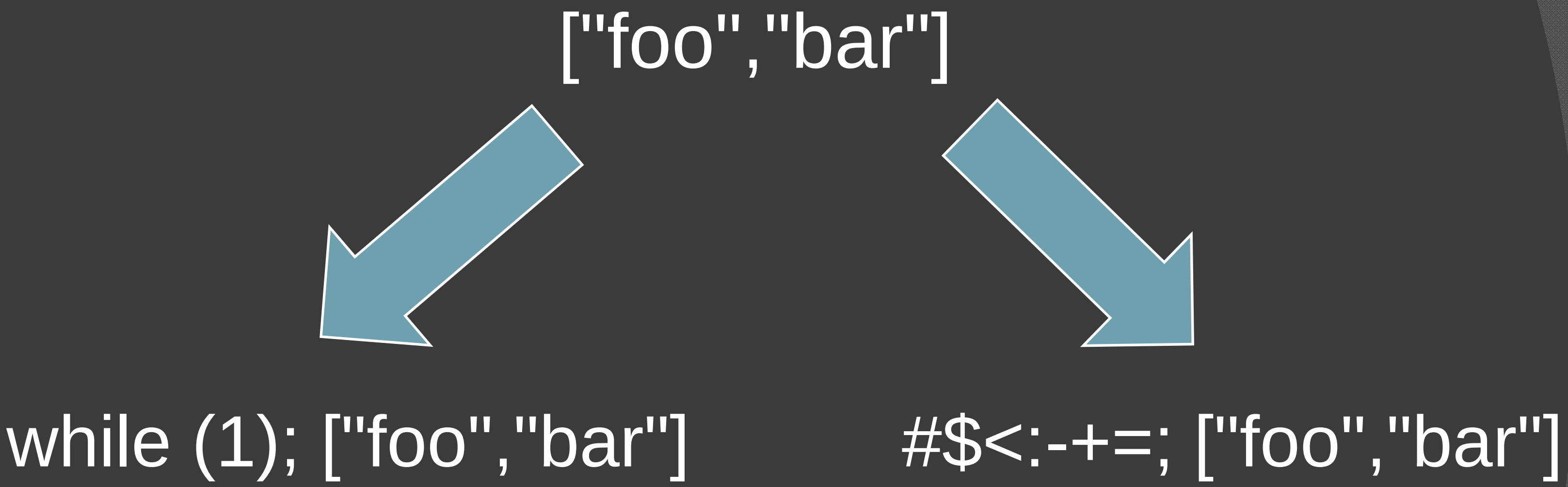
Message Authentication Code defense

DEMO

XSRF Defenses – the good, bad, and ugly

- Bad:
  - POST method
  - Referer header checking
- Ugly:
  - CAPTCHA
- Good:
  - Canary
  - Double-submitting cookie
  - Message Authentication Code

JSON tainting



["foo","bar"]

while (1); ["foo","bar"]

#$<:-+=; ["foo","bar"]

Contact Information

- bryansul at microsoft.com

- http://blogs.msdn.com/sdl
- http://blogs.msdn.com/bryansul