

# HTML5 saugumas

Tomas Lažauninkas  
tomas@critical.lt


# HTML5. Kas tai ?

- ▶ 5 HTML versija
- ▶ Vis dar kūrimo stadijoje
- ▶ ... bet dalinai palaikoma daugelio naršyklių
- ▶ Funkcijos leidžiančios pakeisti Flash, Silverlight ir kitus trečiųjų šalių priedus naršyklėms


# Kas naujo HTML5 ?

- ▶ Nauji tagai:
  - `<button>`, `<video>`, `<audio>`, `<article>`, `<footer>`
- ▶ Nauji atributai:
  - Autofocus, autocomplete, form, pattern
- ▶ `<canvas>` tagas dinamiškai kurti vaizdus
- ▶ Geolokacijos funkcija
- ▶ Webworkers
- ▶ WEB SQL
- ▶ CORS

# HTML5 ir saugumas

- ▶ Nauji tagai ir XSS
  - ▶ Cross Origin Requests
  - ▶ Application cache
  - ▶ WebWorkers
- 

# HTML5 ir XSS

- ▶ Filtrai priimant turinį iš vartotojų.
  - ▶ Dažnai kuriami remiantis blacklist
  - ▶ Nauji tagai ir atributai leidžia apeiti egzistuojančius filtrus
  - ▶ Visada naudoti whitelist filtrus!
- 

# HTML5 ir XSS

- ▶ Blokuojami “<script”, “<iframe” ir pan tagai ?
- ▶ Apeinam su  
*<video onerror=javascript:alert(1)>*
- ▶ Blokuojami “<“, “>” simboliai?
- ▶ ... tačiau galim įterpt turinį tago viduje, tačiau blokuojami atributai kaip onload, onerror?
- ▶ Apeinam su naujais atributais:  
*<form onforminput=“javascript:alert(XSS)” >*

# HTML5 ir XSS

- ▶ Išnaudojant XSS su kai kuriais event handleriais dažniausiai reikia vartotojo įsikišimo
- ▶ Pvz. XSS `<input>` tago viduje turim naudoti `onmouseover`
- ▶ HTML5 eliminuoja tai su nauju paramtru `onfocus`

*`<input type="text" onfocus=alert(/XSS/) autofocus>`*

# HTML5 ir XSS

- ▶ Daugiau XSS atakos vektorių naudojant HTML5 naujoves (ir ne tik) galima rasti:

<http://heideri.ch/jso/#html5>

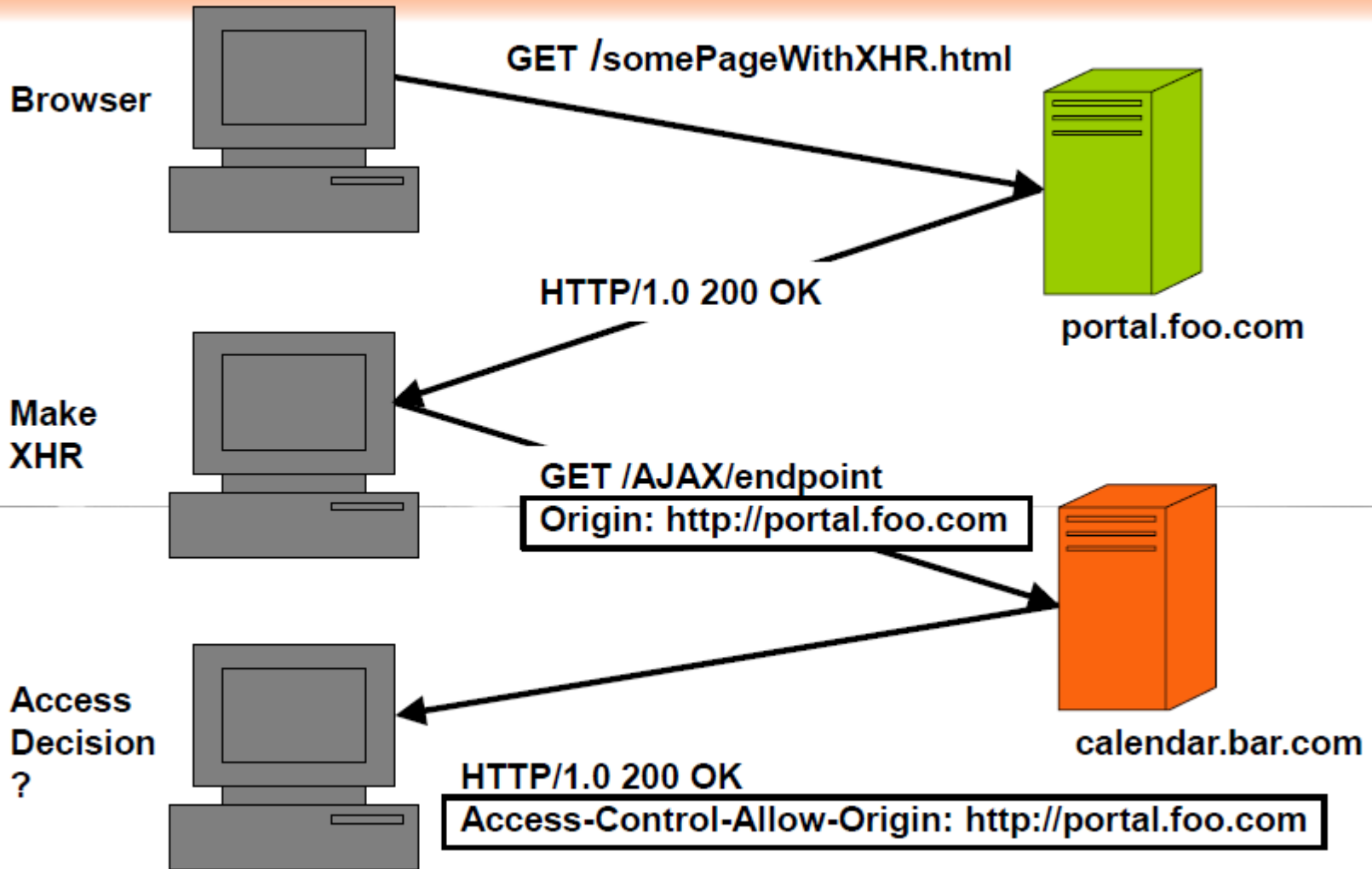
# XSS filtro apėjimo pavyzdys



# Cross Origin Requests

- ▶ Pagrindinis interneto naršyklės saugumo modelis Same Origin Policy neleidžiantis vienam tinklapiui atlikti užklausos į kitus tinklapius ir matyti gautus duomenis
- ▶ HTML5 turi naują funkciją Cross-origin Resources Sharing, kuri leidžia atlikti užklausas tarp skirtingų domenų

# CORS veikimo pavyzdys



# CORS veikimo pavyzdys

- ▶ CORS veikimas remiasi papildoma antrašte HTTP užklausoje atsakyme:

*Access-Control-Allow-Origin:*

<http://portal.foo.com>

- ▶ Teoriškai viskas saugu, tačiau užklausoje galima pateikti “\*” simbolį, kuris leidžia bet kokiam tinklapiui gauti duomenis iš tinklapio.



Access-Control-Allow-Origin

Results 1 - 10 of about 255 for Access-Control-Allow-Origin

» Top countries matching your search

<a href="#">United States</a>	112
<a href="#">Japan</a>	24
<a href="#">Germany</a>	16
<a href="#">Korea, Republic of</a>	15
<a href="#">United Kingdom</a>	12

**Exploit Search**

Search for exploits and vulnerabilities in all the major archives using Shodan Exploits. Supports Exploit DB, Metasploit, CVE, OSVDB and PacketStorm. [Try it now for free!](#)

**211.128.99.44**

Added on 26.01.2011



```
HTTP/1.0 200 OK
Date: Wed, 26 Jan 2011 05:44:09 GMT
Server: Apache
Last-Modified: Sat, 20 Nov 2004 20:16:24 GMT
ETag: "3881a4-2c-3e9564c23b600"
Accept-Ranges: bytes
Content-Length: 44
Access-Control-Allow-Origin: *
Connection: close
Content-Type: text/html
```

**184.73.225.228**


Added on 25.01.2011



ec2-184-73-225-228.compute-1.amazonaws.com

```
HTTP/1.0 200 OK
Access-Control-Allow-Origin: *
Cache-Control: private, no-cache, no-store, must-revalidate
Content-Type: application/json; charset=utf-8
Date: Tue, 25 Jan 2011 21:48:32 GMT
Expires: 0
Pragma: no-cache
Server:
Vary: Accept-Encoding
Content-Length: 164
```

# Application cache

- ▶ HTML5 naujovė papildomas duomenų saugojimo cache'as nepriklausomas nuo naršyklės
  - ▶ Kiekvienam tinklapiui skiriami 5MB vietos
  - ▶ Bet kurie tinklapyje esantys failai gali būti cache'uojami
  - ▶ Perkrovus tinklapį aplikacijos cache'as nėra atnaujinamas
- 

# Phishing ataka naudojant application cache

1. Vartotojas prisijungia prie nesaugaus Wifi tinklo kontroliuojamo įsilaužėlio
2. Vartotojui naršant tinklapius yra įterpiamas nematomas iframe į tinklapį kurį norima suklastot (tarkim gmail.com)
3. Įsilaužėlis perima užklausą gmail.com tinklapiui ir gražina savo rezultata su netikru gmail.com prisijungimo tinklapiu bei nurodymu išsaugoti jį aplikacijos cach'e
4. Vartotojui kitą kartą kreipiantis į gmail.com bus rodomas įsilaužėlio suklastotas tinklapis
5. Ataka įgyvendinta naudojantis Imposter įrankiu prieš Chrome ir Safari naršyklės  
<http://blog.andlabs.org/2010/06/chrome-and-safari-users-open-to-stealth.html>

# WEB SQL

- ▶ Funkcija leidžianti tinklapiui sukurti SQL duomenų bazę vartotojo kompiuteryje
- ▶ Nauja atakos kryptis, ateityje galbūt pamatysime SQL injekcijų išnaudojimo naršyklių duomenų bazėse pavyzdžių

# HTML5 WebWorkers

- ▶ WebWorkers tai fone veikiančios JavaScripto thread'ai
- ▶ Dabar bet kuris tinklapis gali sukurti fone veikiančius thread'us, kurie veiks tol kol tinklapis bus aktyvus
- ▶ Tai įgalina naršyklę išnaudoti sudetingesniems procesams/skaičiavimams atlikti:
  - DDoS'inimui
  - Slaptažodžių crackinimui
  - Efektyvesniam prievadų skenavimui su JavaScript'u

# HTML WebWorkers

Parazitiniai skaičiavimai: užvaldyti internetą - Opera

File Edit View Bookmarks Widgets Mail Tools Help

Parazitiniai skaičia... x +

Web www.critical.lt/blogs/show/parazitiniai-skaiciavimai-uzvaldy

Search with Google

Lietuvių | English

security security today

Paslaugos Sprendimai Laboratorija Tinklaraštis Partneriams Apie mus

## PARAZITINIAI SKAIČIAVIMAI: UŽVALDYTI INTERNETĄ

Mirosław Lučinskij, 2008-10-10 20:55:24

Šiais metais tik ir girdime apie per daug išpūstus „labai baisius“ pažeidžiamumus: pradžioje buvo Dan Kaminsky su DNS dizaino klaidomis, dabar internetą užlenkti žada keli TCP/IP steko tyrinėtojai. Tačiau DNS serverių eksploatacija bei priešiška nusiteikusių serverių gesinimas vis tiek nedaro tavęs interneto valdovu, galbūt yra kitas būdas?

Iš tiesų parašyti šia tema mane paskatino ne tik perdėtas dėmesys ir panika dėl šiomet aptiktų pažeidžiamumų, kurių pradžioje nenorima atskleisti „nes ateis interneto galas“, o kai jie atskleidžiami, pasirodo, jog internetas dar kurį laiką pagyvens (o reklaminė kampanija pasisekė gerai). Užsiėmęs svetainės atnaujinimo darbais ir paskendęs praeities apmąstymuose prisiminiau dar Critical.lt ištakose užduotą klausimą kolegai: Povilai, o kaip tu užvaldytum internetą? - paklausiau aš. Šis pamąstė, išpūtė cigaretės dūmą ir tuomet prasidėjo ilga diskusija, kurios esmę pamėginsiu ir išdėstyti savo tekste.


View (100%)

# HTML WebWorkers

<http://www.andlabs.org/tools/ravan.html>

The screenshot shows the RAVAN web interface in an Opera browser window. The browser title is "Ravan - JavaScript Distributed Computing System - Opera". The address bar shows the URL "www.andlabs.org/tools/ravan.html". The page content includes a logo of a stylized head with circuitry, the text "RAVAN JavaScript Distributed Computing System (BETA)", and a form for submitting and cracking hashes. The form has two buttons: "Submit Hashes" and "Crack Hashes". Below the buttons are input fields for "Hash:" and "Salt:". The "Charset:" field is set to "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789`~!@". The "Algorithm:" dropdown is set to "MD5". The "Add Salt" options are "Before Hash" (unselected) and "After Hash" (selected). A "Submit Hash" button is present. At the bottom of the form, there are links for "Show Advanced Options" and "Hide Submit Form". The browser's status bar at the bottom right shows "View (100%)".

More Details Feedback/Comments/Questions: [@lavakumark](#)



**RAVAN**  
JavaScript Distributed Computing System (BETA)

Hash:  Salt:

Charset:  Algorithm:  Add Salt :  Before Hash  After Hash

[Show Advanced Options](#) [Hide Submit Form](#)

# Klausimai ?

