Bezpečnosť web aplikácii: Flash a HTML5

Bezpečnosť IT infraštruktúry

Michal Rjaško

LS 2012/2013

Flash vs. HTML
Kým neprišlo HTML5

- Flash sa rozširuje hlavne kvôli malej funkcionalite HTML prostredia.
  - umožňuje pridať na stránku interaktívny obsah (video, animácie, aplikácie)
- Nejednotné HTML prostredie – stránky je potrebné testovať v každom prehliadači
- Slabá podpora štandardov v prehliadačoch (hlavne v tom čase najrozšírenejší IE <= 8)

Flash

- Multiplatformová multimediálna platforma ☺
- Ideálny pre animácie, grafiku, video a jednoduché hry
- Flash súbor môže byť vložený do www stránky, kde sa prehráva pomocou pluginu do prehliadača - Flash playera
- Logika flashových aplikácií a hier je postavená na jazyku ActionScript (aktuálne vo verzii 3.0).

História Flashu

- 1993 - SmartSketch (Jonathan Gay, FutureWare software)
  - vektorový grafický editor pre Pen PC – počítače ovládané perom
- 1995 – FutureSplash Animator (využíva ho MSN a Disney)
- 1996 – Macromedia Flash 1.0
- 2005 – Adobe Flash 8.0

História Flashu

Flash 2	1997	Buttons, libraries, stereo audio, improved bitmap integration
Flash 3	1998	Alpha transparency, MP3
Flash 4	1999	Streaming MP3 (preinstalled in IE 5)
Flash 5	2000	ActionScript
Flash 6	2002	Video
Flash 7	2003	Charts & graphs, text effects, ActionScript 2.0
Flash 8	2005	Gif & Png, significant new video codec
Flash 9	2006-2007	ActionScript 3, AVM2, FullScreen, H.264, AAC, Flex Builder 2
Flash 10	2008	3D, advanced text features, AIR
	2010-2011	MultiTouch, dynamic streaming, hardware decoding H.264, 64-bit, Acoustic Echo Cancellation, Flash for Mobile Devices
Flash 11	2011-2013	3D hardware, secure random number generation, protected dynamic streaming, workers, ...

Flash je silná platforma

Dovoľuje:

- Posielať HTTP dotazy na server aj mimo vlastnej domény
- Vytvárať socketové pripojenia
- Ukladať dáta na disku klienta (tzv. SharedObjects)
- Využívať kameru, mikrofón
- Pristupovať k DOM www stránky do ktorej je vložený
- Spúšťať JavaScript
- Spúšťať iné Flash súbory

S príchodom HTML5 je

budúcnosť Flashu neistá

- Apple odmietol flash na svojich mobilných zariadeniach
- Od verzie 11.1 Adobe nevyvíja Flash player pre mobilné zariadenia
- Verzie od 11.2 - plugin pre linux bude k dispozícii iba cez „Pepper“ API, zatiaľ iba Google Chrome
- Orientácia na „Premium Video“ a hry
  - Adobe AIR, mobilné aplikácie

HTML5 - história

“It must be admitted that many aspects of HTML appear at first glance to be nonsensical and inconsistent.”

“HTML, its supporting DOM APIs, as well as many of its supporting technologies, have been developed over a period of several decades by a wide array of people with different priorities who, in many cases, did not know of each other's existence.”

[w3.org/TR/html5/introduction.html#introduction]

HTML5 - história

- **1990-1993** :: prvé verzie odvodené od SGML, využíva ho CERN
- **1995** :: W3C vydalo HTML 3.0
- **1997** :: HTML 3.2 – veľa nových rozšírení
- **1998** :: HTML 4 – používané doteraz. DOM level 1
  - W3C sa rozhodlo presedlať na XHTML
- **2000** :: DOM level 2 - getElementById(), events
- **2000** :: W3C vydalo XHTML 1.0,
  - začiatok vývoja XHTML2
- **2004** :: DOM level 3

HTML5 - história

- **2004** :: Idea HTML5, založené konzorcium WHATWG
  - W3C sa nepridáva ale pokračuje v XHTML2
- **2005** :: AJAX, XMLHttpRequest
- **2006 – 2007** :: W3C zmenilo rozhodnutie, podieľa sa na HTML5
- **2007 – doteraz** :: WHATWG a W3C spolupracujú na HTML5
- **2012** :: HTML5 W3C Candidate Recommendation

HTML5 – Aktuálny stav

- HTML5 nie je dokončený štandard!
  - Aktuálne je v stave „Candidate recommendation“
    - Zásadné zmeny by nemali nastať
  - Ďalšie fázy: Proposed Recommendation, Recommendation
  - Recommendation je naplánovaný na rok 2014
- W3C != WHATWG, HTML5 != HTML5
- Neúplná podpora prehliadačov
  - Stále otravujú staré prehliadače (IE8, IE7, ...)

HTML5 – Aktuálny stav

- W3C špecifikácia má okolo 4.4 MB textu
- WHATWG špecifikácia – 707 strán A4
- To je veľa práce na implementovanie
- Nezabúdajme ešte na
  - CSS3
  - JavaScript
  - SVG, MathML
  - Canvas, atď., atď.

HTML5 - Bezpečnosť

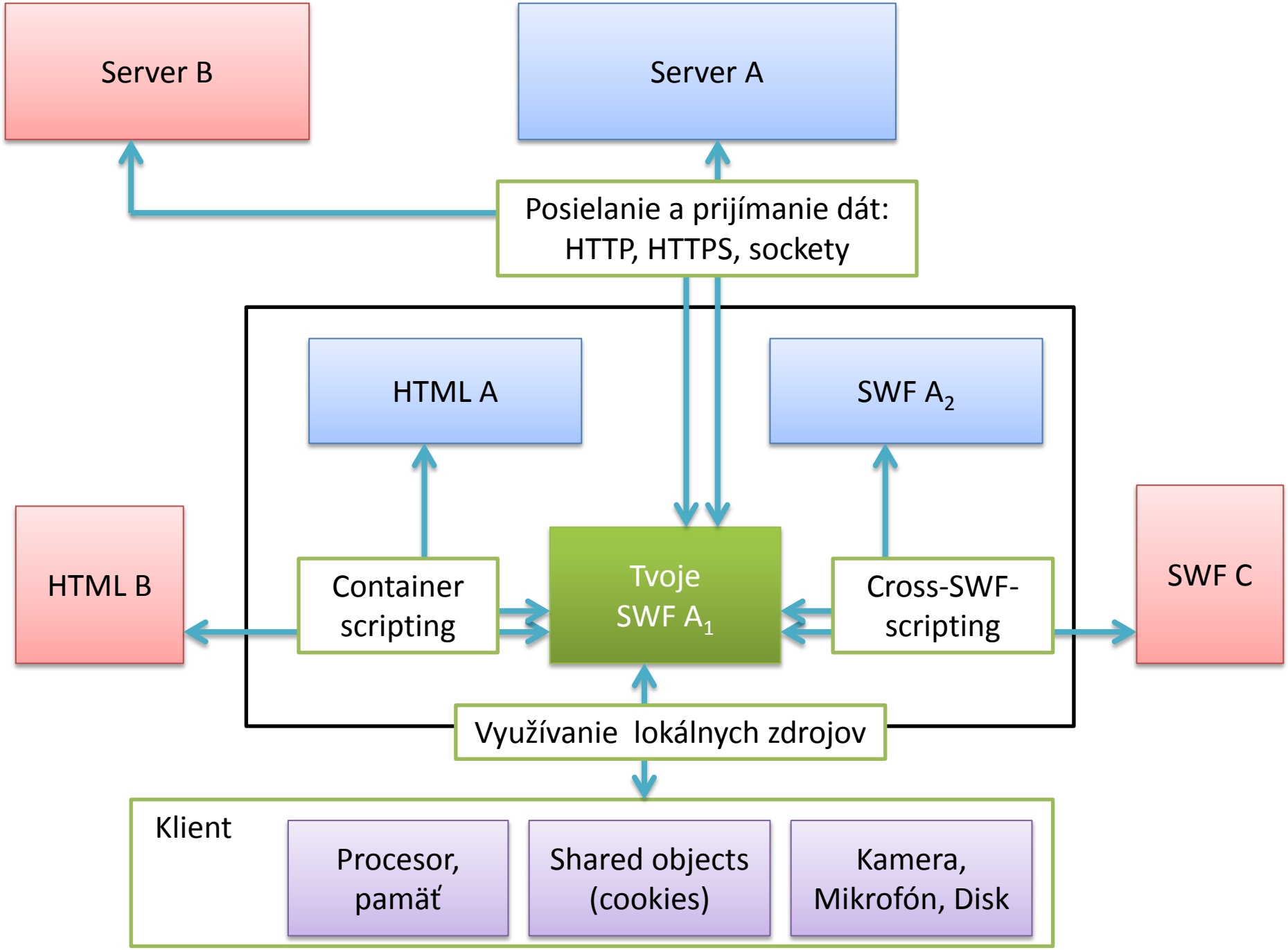
- Niektorí vravia, že HTML5 je samo o sebe zraniteľnosť
- Bezpečná implementácia si vyžaduje:
  - Jasnú špecifikáciu ✖
  - Rozumné množstvo práce ✖
  - Precízne testovanie ✖
  - Rýchle a presné odstraňovanie chýb ✖
  - Rýchle nasadzovanie záplat ✖

HTML5 – Bezpečnosť

- Nekonzistentná a stále sa vyvíjajúca špecifikácia
- Prehliadače súťažia, kto ako prvý implementuje funkcionality HTML5
- Programátori stále vyvíjajú stránky s chybami
- Nutnosť podporovať staré prehliadače
  - IE7, IE8, IE6 je niekde stále v hre.

HTML5 vs. Flash

FLASH


Server B
Server A
Posielanie a prijímanie dát:
HTTP, HTTPS, sockety
HTML A
SWF A2
HTML B
Container
scripting
Tvoje
SWF A1
Cross-SWF-
scripting
SWF C
Využívanie lokálnych zdrojov
Klient
Procesor,
pamäť
Shared objects
(cookies)
Kamera,
Mikrofón, Disk

Hrozby

- Chyby Flash playera – AVM
- Dekompilácia
- Cross-site scripting, Cross-site flashing
- Komunikácia so serverom, zle nastavená cross-domain policy, DDOS útoky
- Phishing
- Prístup k lokálnym zdrojom (Clipboard, Kamera, Mikrofón, Disk, FullScreen)
- Využitie procesora a pamäte

Chyby Flash playera – AVM

- Zdrojový kód je skompilovaný do byte-kódu
- Flash player na klientovi vykoná byte-kód pomocou AVM
  - V prípade AVM2 môže dôjsť k prekompilovaniu do strojového kódu (JIT compilation).
- 4 fázy: loading, linking, verification, execution
  - Fázy sa prelínajú, verifikácia beží v každej fáze
- Chyby a nedostatky vo verifikačnom procese umožňujú útočníkovi vykonať škodlivý (resp. akýkoľvek) kód.

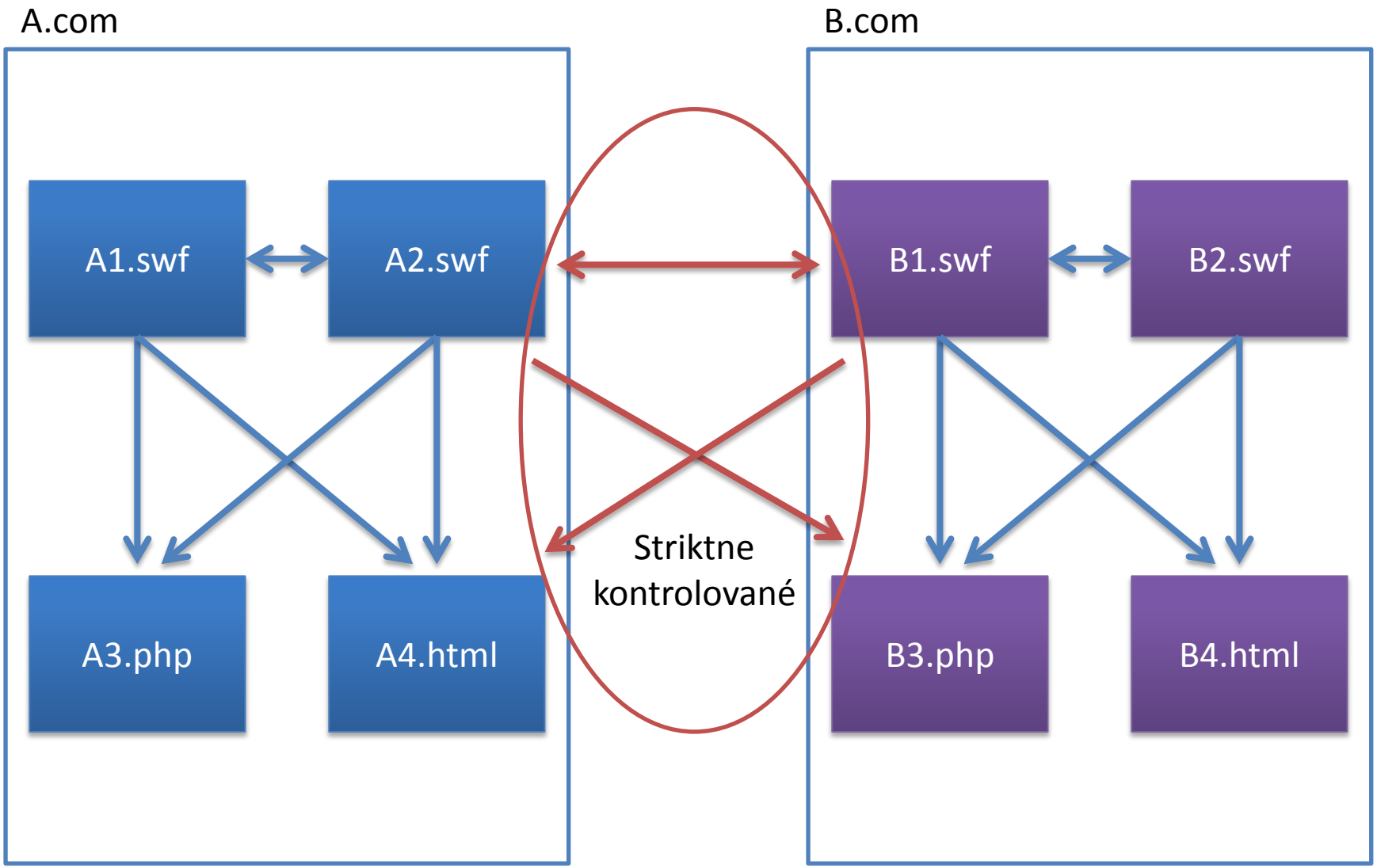
Dekompilácia

- Špecifikácia AVM2 je otvorená
- Bytecode pre AVM2 je ľahko dekompilovateľný
- Útočník si môže prezerať zdrojový kód
  - SWF súbor nesmie obsahovať citlivé údaje
  - Napr. šifrovanie/autentifikácia dát s kľúčom vloženým do SWF súboru
- Phishing útok dekompiláciou a úpravou funkcionality

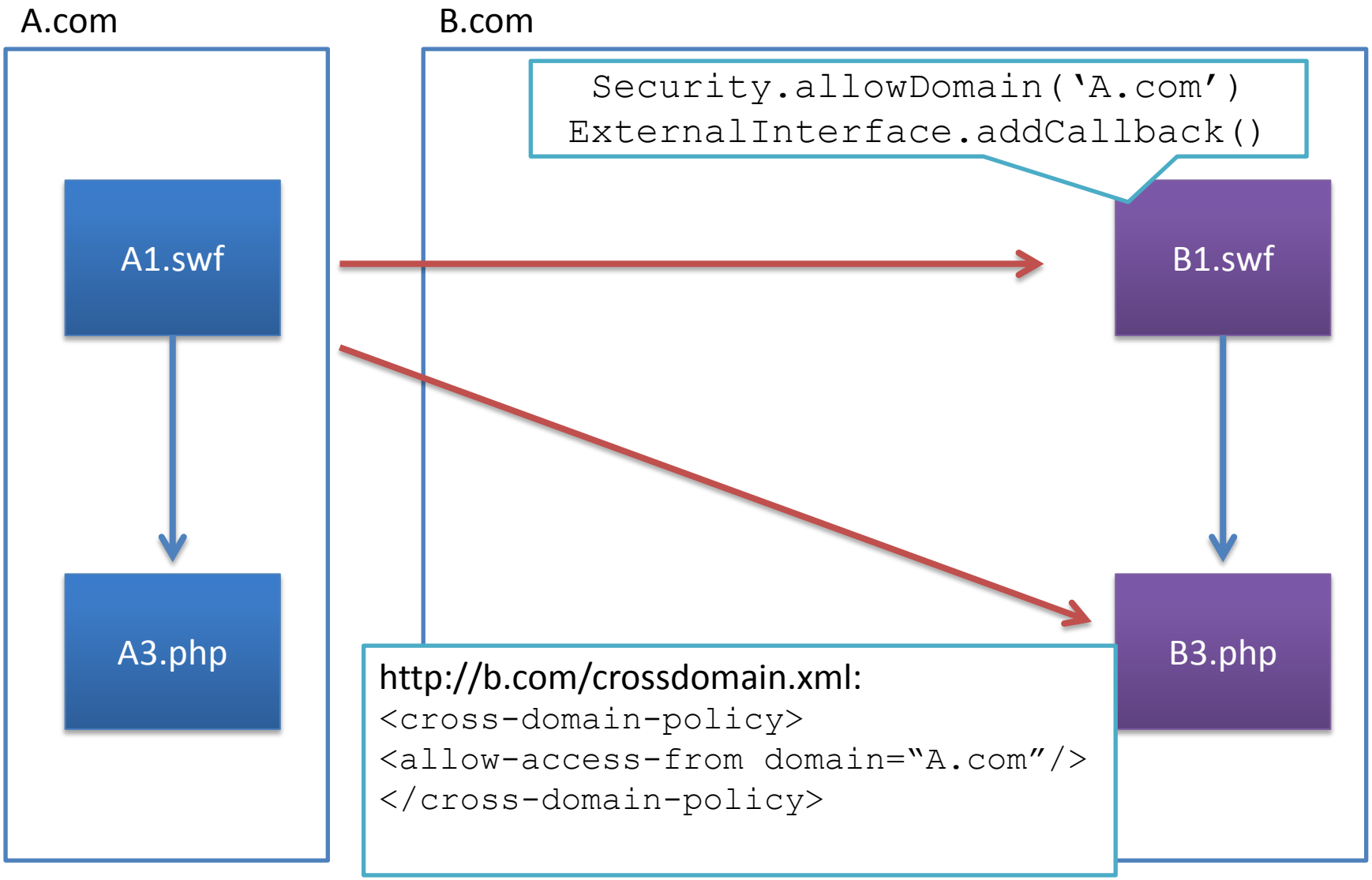
Flash Sandbox model

- Flash priradí SWF súboru tzv. „sandbox“ podľa toho, z akej domény je daný súbor.
- Komunikácia v rámci toho istého sandboxu nie je obmedzená
- Komunikácia medzi rôznymi sandboxami je možná, ale striktne kontrolovaná

Flash Sandbox model



Flash Sandbox model



Komunikácia so serverom cez HTTP
crossdomain.xml

- SWF súbor môže komunikovať so serverom v rámci tej istej domény (sandboxu) bez obmedzenia.
- Dotazy mimo sandboxu sú riadené pomocou tzv. „cross domain policy file“:

```
<cross-domain-policy>
<allow-access-from domain="A.com" secure="true"/>
<site-control permitted-cross-domain-policies="master-only">
</cross-domain-policy>
```

- SWF súbor môže načítať iný ako štandardný domain.com/crossdomain.xml súbor pomocou volania:

```
Security.loadPolicyFile("http://B.com/crossdomain.xml")
```

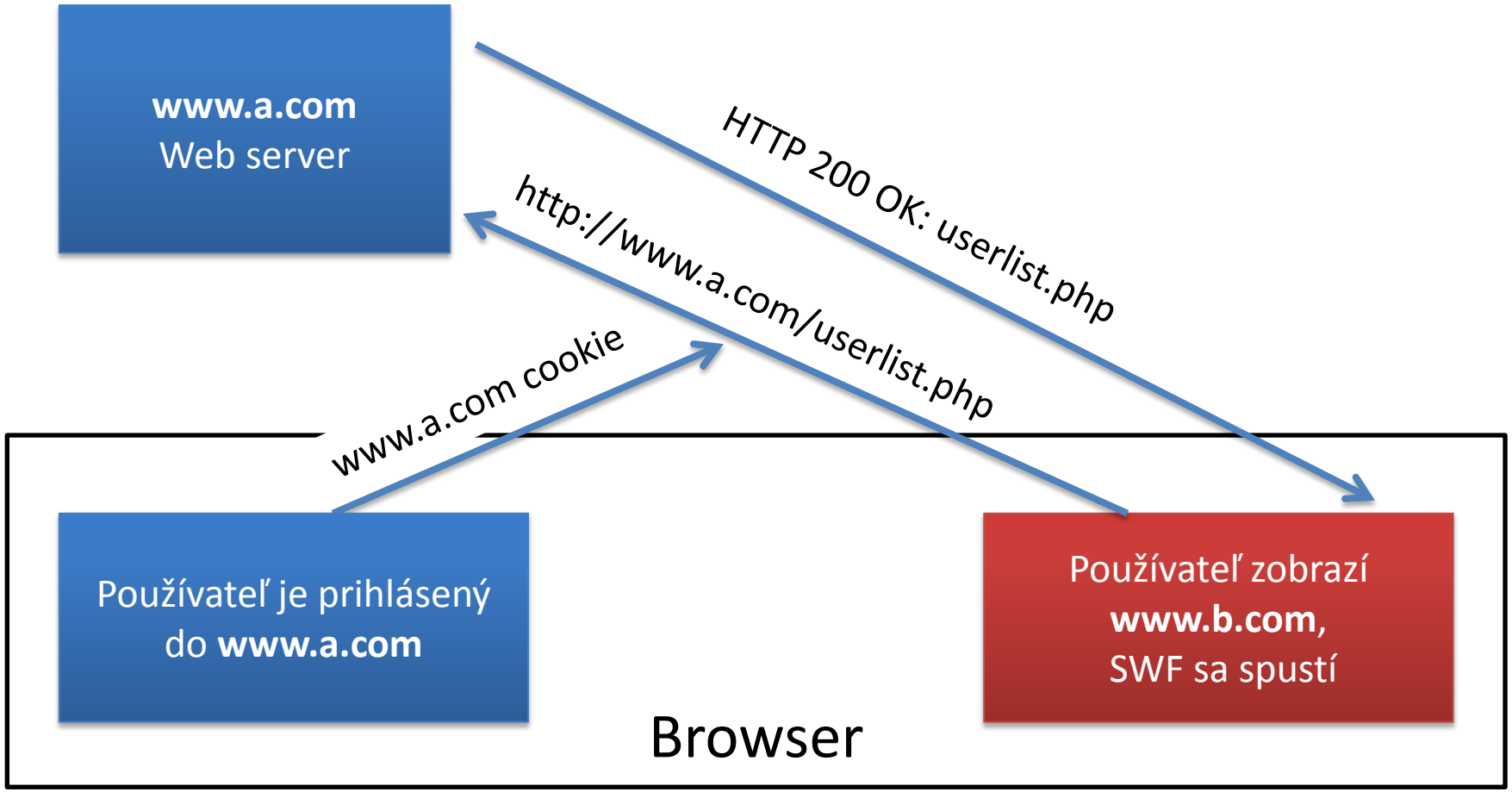
Komunikácia so serverom – sockety

- Flash umožňuje vytvárať socketové pripojenie na server, mimo HTTP protokolu
- Aplikujú sa tu rovnaké cross-domain pravidlá ako pri HTTP komunikácii
  - `allow-access-from` môže obsahovať atribút `ports="3045,4056"`

Komunikácia so serverom

Na čo si treba dať pozor

`<allow-access-from domain="*">`



Komunikácia so serverom

Na čo si treba dať pozor

```
<allow-access-from domain="*">
```

- DDOS útok ☺:
  - Agent MOSADu cez nastrčenú reklamnú agentúru objedná reklamu na sme.sk
  - Reklama bude swf súbor, ktorý v deň volieb v liráne začne posielať dotazy na http://www.elections.ir
  - Kým sme.sk reklamu stiahnu (ak si to vôbec všimnú) dôjde k narušeniu priebehu volieb

Komunikácia so serverom

Na čo si treba dať pozor

```
<site-control permitted-cross-domain-policies="all">
```

- Potenciálny útočník môže nahrať (napr. cez CMS) vlastný cross domain policy file.

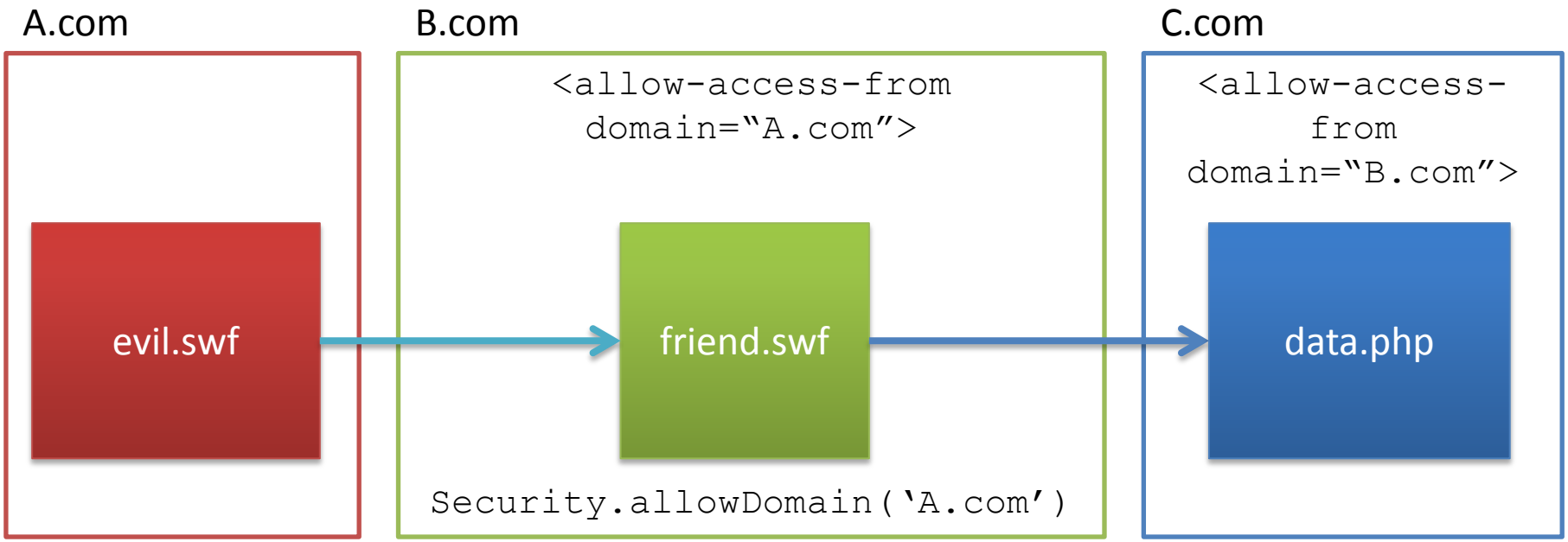
```
<allow-access-from domain="A.com" secure="false"/>
```

- Môže dôjsť (chybou alebo úmyselne) k prenosu údajov v otvorenom tvare, aj keď bol swf súbor načítaný cez HTTPS.

Komunikácia so serverom a XSF

Na čo si treba dať pozor

- Prístup k nedostupnej doméne cez ďalšie SWF



Cross-site scripting
atribút allowScriptAccess

- Vloženie SWF súboru do HTML:

```
<object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
   codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=6,0,0,0"
   width="100%" height="100%">
 <param name="src" value="mymovie.swf">
 <param name="allowScriptAccess" value="sameDomain">
 <embed src="mymovie.swf" width="100%" height="100%"
   allowScriptAccess="sameDomain"
   type="application/x-shockwave-flash"
   pluginspage="http://www.macromedia.com/go/getflashplayer">
</object>
```

Cross-site scripting
allowScriptAccess

- Atribútom allowScriptAccess môžeme obmedziť prístup SWF súboru k volaniu JavaScriptu
- Možné hodnoty:
  - **never** – bez prístupu k JavaScriptu / DOM
  - **sameDomain** – SWF súbor z rovnakej domény ako daná www stránka má prístup k JavaScriptu / DOM stránky
  - **always** – SWF aj z iných domén môžu volať JavaScript - **Nebezpečné!**

Cross-site scripting
allowScriptAccess="always"

```
<?xml version="1.0" encoding="utf-8"?>
<s:Application xmlns:fx="http://ns.adobe.com/mxml/2009"
               xmlns:s="library://ns.adobe.com/flex/spark"
               xmlns:mx="library://ns.adobe.com/flex/mx" minWidth="300" minHeight="300"
               applicationComplete="appComplete()">
<fx:Script>
<![CDATA[
import mx.controls.Alert;
import mx.controls.SWFLoader;

private function appComplete():void {
    javascriptEval();
}

public function javascriptEval():void {
    Alert.show(ExternalInterface.call('eval',"document.cookie;").toString());
}
```

Cross-site scripting
FlashVars

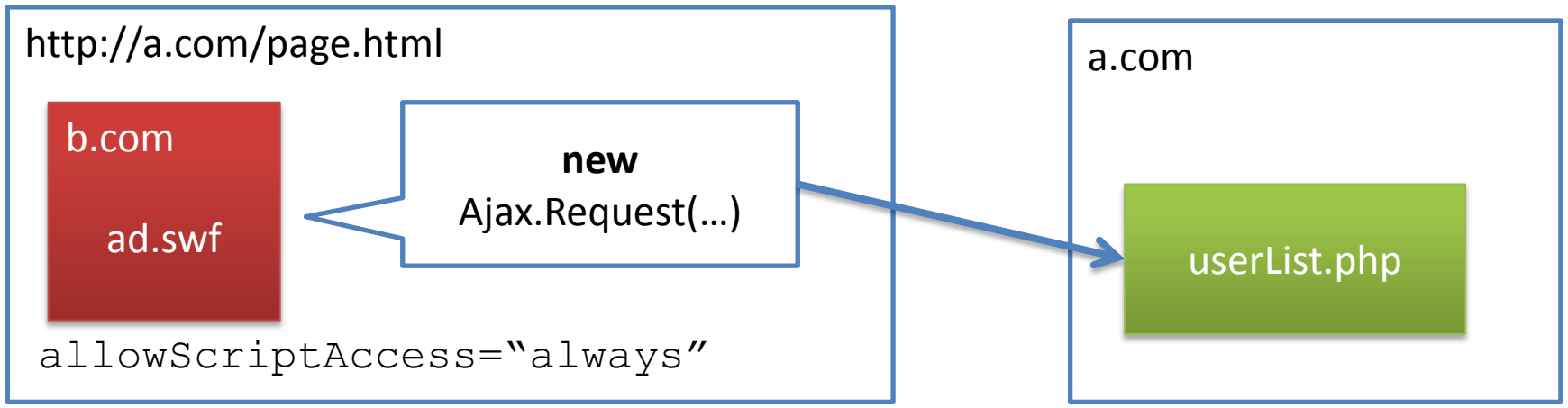
- FlashVars slúžia na odovzdanie parametrov z HTML do Flashu:

```
<object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
    codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=6,0,0,0"
    width="100%" height="100%">
 <param name="src" value="myad.swf?clickthru=http://www.sme.sk">
 <param name="allowScriptAccess" value="sameDomain">
 <param name="flashVars"
      value="clickthru=http://www.sme.sk">
```

- Úpravou FlashVars možno niekedy docieliť vykonanie JavaScriptu
  - mymovie.swf?clickthru=javascript:alert(document.cookie);

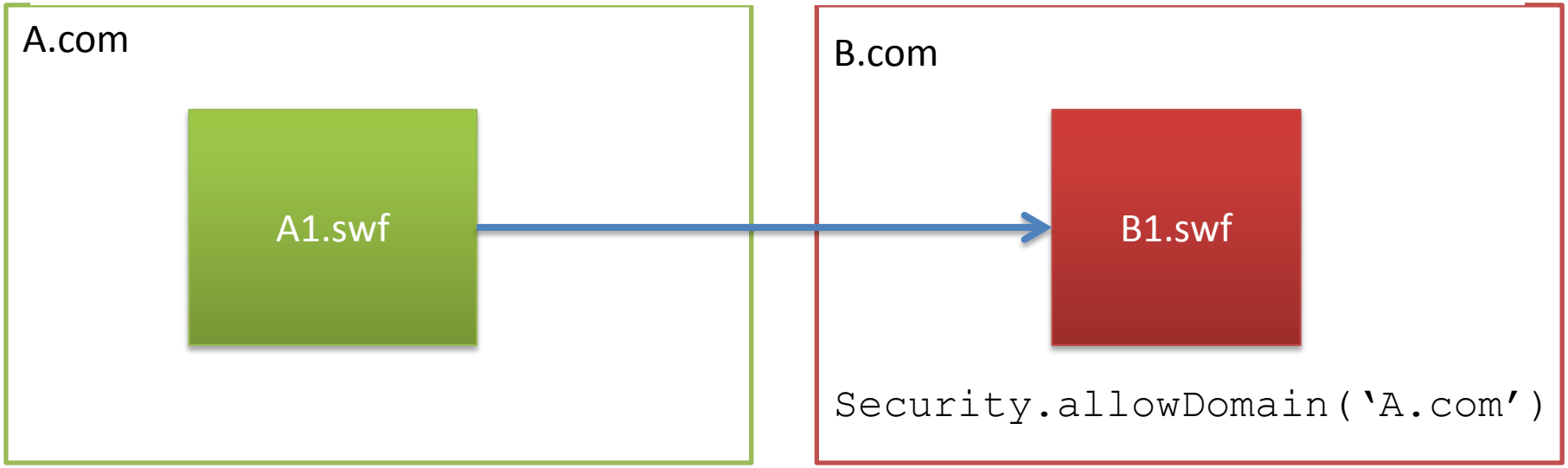
Cross-site scripting
XSS tunneling

- tunelovanie HTTP prenosu cez XSS kanály.
  - SWF súbor môže cez JavaScriptové volania získať inak nedostupné dáta
    - Napr. chránené IP verifikáciou, VPN, prípadne klasickou autentifikáciou



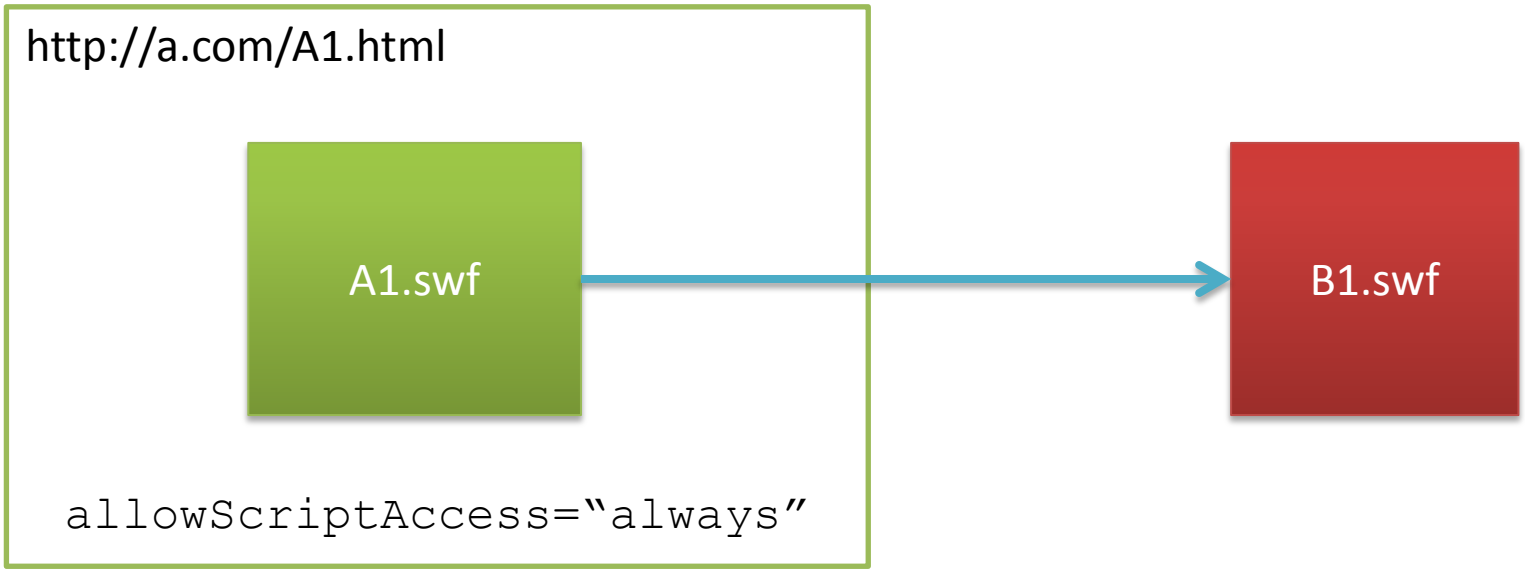
Cross-Site flashing

- Flash spúšťa iný flash
  - Volanie `Security.allowDomain('*')`
  - A1.swf môže mať prístup k metódam a vlastnostiam B1.swf a naopak
  - Hrozí únik dát, prípadne manipulácia s funkcionalitou SWF súboru



Cross-Site flashing
Na čo si treba dať pozor

- allowScriptAccess=“always”
- Ak A1.swf načíta B1.swf, B1.swf môže vykonať JavaScript v rámci A1.html
- XSS hrozba



Cross-Site flashing

Na čo si treba dať pozor

Dva spôsoby načítania SWF súboru:

```
var swfloader:SWFLoader = new SWFLoader();
swfloader.load('http://b.com/b1.swf');
```

vs.

```
var r:URLRequest = new URLRequest('http://b.com/b1.swf')
var loader:URLLoader = new URLLoader(request);
loader.dataFormat = URLLoaderDataFormat.BINARY;
loader.load(request);
...
swfloader.load(loader.data);
```

V druhom prípade dôjde k strate kontextu – načítaný SWF súbor bude mať rovnaký sandbox.

Prístup k lokálnym zdrojom

- Procesor:
  - Flash player kontroluje či vykonanie niektorej operácie netrvá dlhšie ako 15s, potom umožní beh SWF zastaviť
  - Do verzie 11.4 flash aplikácia beží v jednom vlákne
  - Od verzie 11.4 podpora viacerých vlákien

- Pamäť:
  - Flash player neobmedzuje množstvo využitej pamäte vo Flash aplikácii

Prístup k lokálnym zdrojom

- Disk:
  - Možnosť ukladať na klienta tzv. SharedObjects – cookies pre Flash.
    - Veľkosť pre jednu doménu je prednastavená na 100 KB, používateľ ju môže zmeniť
  - Možnosť uploadovania/čítania súborov (aj viacerých naraz)
    - Iba pomocou triedy FileReference
    - FileReference.browse zavolá systémový dialóg na vybratie súborov
    - Metóda môže byť zavolaná iba v „user-event handler”, t.j. pri spracovaní udalosti kliknutie myšou alebo stlačenia klávesy.
    - FileReference.download otvorí systémový dialóg na uloženie súboru

Prístup k lokálnym zdrojom

- Schránka (Clipboard)
  - Ukladanie do schránky je možné len v “user-event handler”
    - Clipboard.generalClipboard.setData()
  - Čítanie je možné len v spracovaní udalosti “paste” (čiže ak používateľ stlačí CTRL+V)
    - Clipboard.generalClipboard.getData()
- Kamera, Mikrofón
  - Flash player si vypýta povolenie
  - Prístup sa dá povoliť/zakázať v nastaveniach

Prístup k lokálnym zdrojom

- Celoobrazovkový režim
  - K dispozícii od verzie 9
  - Klávesa ESC ho ukončuje
  - Spustenie celoobrazovkového režimu je môžné len v “user-event” handler
- V celoobrazovkovom režime je limitované použitie klávesnice (DEMO)
  - Do verzie 10 boli zakázané akékoľvek klávesy
  - Od verzie 10 sú povolené tzv. “non-printing” klávesy – šípky, medzera, tabulátor, enter, …
  - Od verzie 11.3 sú podporované všetky klávesy – špeciálny atribút allowFullScreenInteractive
    - Flash player si vyžiada povolenie od používateľa

Flash: zhrnutie

Je potrebné:

- Limitovať prístup Flashu k JavaScriptu pri spúšťaní externých SWF súborov
- Dať pozor na JavaScript vo FlashVars
- Dať pozor na skriptovanie SWF-SWF medzi súbormi z rôznych domén
- Správne nastaviť cross-domain policy file

Flash vs. HTML5

HTML5

HTML5

- Zámer:
  - Vytvoriť jednoduchú platformu na vytváranie interaktívnych web aplikácií
  - Menej XML striktnosti (oproti XHTML), viac voľnosti
  - Dôraz na bezpečnosť

- HTML5 = HTML + JS + CSS (+ SVG + SQL + …)

HTML5 – Nové funkcie

- Nové typy formulárových prvkov
  - date, tel, color, number, email, url,...
  - Autofocus
  - Formulárový element môže byť mimo formulára
  - Validácia formulára na strane klienta
- Nové vlastnosti Iframe: sandbox, seemless
- History API
- Local storage
- SQLLite – databáza na klientovi

HTML5 – Nové funkcie

- Geolocation
- Notifications
- SVG, Canvas
- MathML
- Animácie a tranformácie
- WebGL – 3D akcelerácia v prehliadači
- Audio / Video
- Webfonts
- Offline application cache
- Viac sématiky:
  - nav, figure, section, …
- CORS – Cross Origin Resource Sharing
- WebSockets
- WebWorkers
- HSTS: HTTP Strict-Transport-Security
- CSP: Content Security Policy
- …

HTML5 – Nové funkcie

- HTML5 obsahuje aj niekoľko bezpečnostných vylepšení, avšak
- Rozšírená funkcionalita HTML5 vo všeobecnosti zjednodušuje prácu útočníkom
  - Veľké množstvo nových funkcií – scenáre útokov budú postupne pribúdať
- Určite je jednoduchšie
  - Sledovať používateľov (Geolocation, localStorage, history API)
  - Cross-site scripting – XSS (CORS, autofocus, nové formuláre, …)

JavaScript a jeho bezpečnosť

- Bezpečnosť z roku 1995
- Dva základné bezpečnostné požiadavky
  - Zabrániť škodlivej stránke útočiť na Váš počítač
  - Zabrániť škodlivej stránke pristupovať k inej stránke
- Avšak, ak máte v súčasnosti všetky dokumenty v cloude, koho zaujíma že sa útočník nevie dostať k adresáru „My documents“?

JavaScript

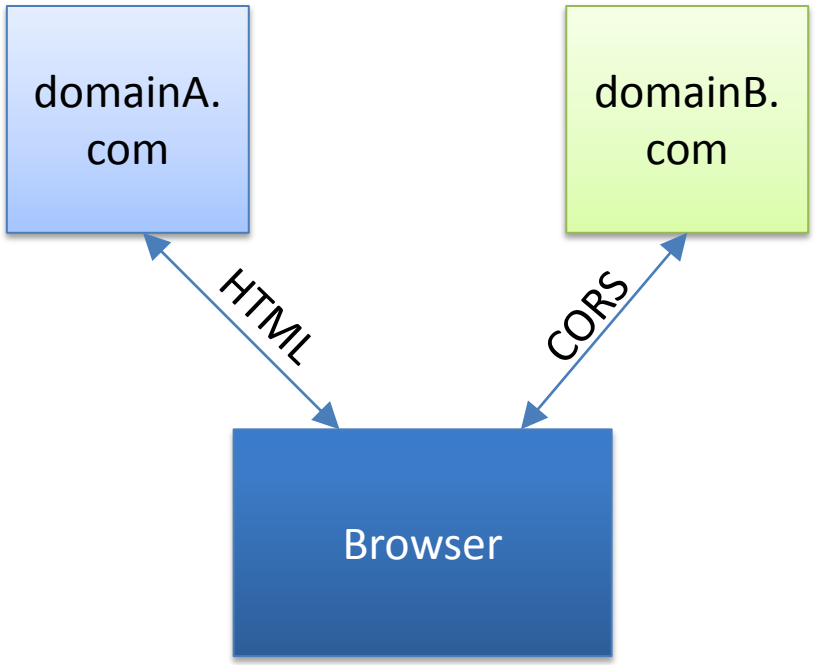
- Same Origin Policy: Skripty na stránke nemôžu interagovať so stránkami v inej doméne
  - Napr. skript načítaný stránkou www.fmph.uniba.sk nemôže interagovať so stránkou www.virus.com
- Avšak, skripty načítané v rámci tej istej stránky (hoci aj z iných domén) medzi sebou interagujú
  - JavaScript je inherentne globálny
  - Skripty si môžu navzájom prepisovať globálne premenné, funkcie, definície objektov a pod.

Cross Origin Resource Sharing

- Umožňuje JavaScriptu sťahovať dáta aj z inej domény
- Podobný základný princíp ako crossdomain.xml vo Flashi
- Prehliadač na základe hlavičky v HTTP odpovedi určí, či daný JavaScript dotaz na inú doménu povolí
  - nevýhoda oproti crossdomain.xml, kde sa najprv načíta politika a dotaz sa zakáže ešte pred jeho odoslaním na server

Cross Origin Resource Sharing

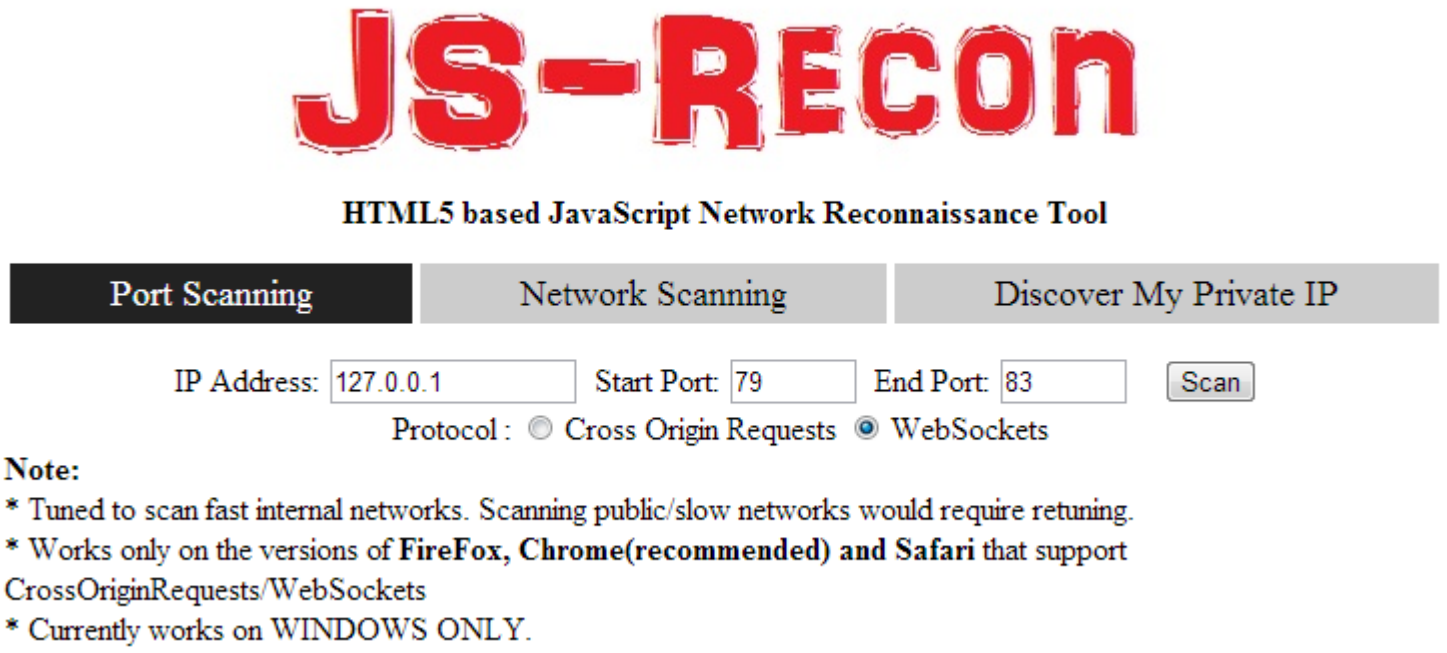
```
GET / HTTP/1.1
Host: domainB.com
Origin: http://domainA.com
…

HTTP/1.1. 200 OK
Content-type: text/html
Access-Control-Allow-Origin: http://domainA.com
…
[data]
```



Cross Origin Resource Sharing

Port / network scanner DEMO:

http://www.andlabs.org/tools/jsrecon.html



Iframe sandbox

- Možnosť umiestniť Iframe do „sandboxu“ cez atribút sandbox
- Dá sa špecifikovať, čo je povolené a čo nie
- `<iframe src=“infected.html” sandbox=“...”>`
- Lepšia správa prístupu oproti allowScriptAccess vo Flashi

Bez atribútu sandbox	Javascript beží normálne
`sandbox` atribút	Bez akéhokoľvek JavaScriptu
`sandbox=“allow-scripts”`	JavaScript sa vykonáva ~~document.cookie~~ ~~localStorage()~~ ~~sessionStorage()~~

Content Security Policy

- Obrana proti XSS útokom
- V HTTP hlavičke server špecifikuje, z ktorých domén môže stránka sťahovať údaje

Použitie:

- `Content-Security-Policy: script-src 'self' https://apis.google.com`
- Možnosť špecifikovať CSP pre viacero rôznych zdrojov:
  - `connect-src, font-src, frame-src, img-src, media-src, object-src, style-src`

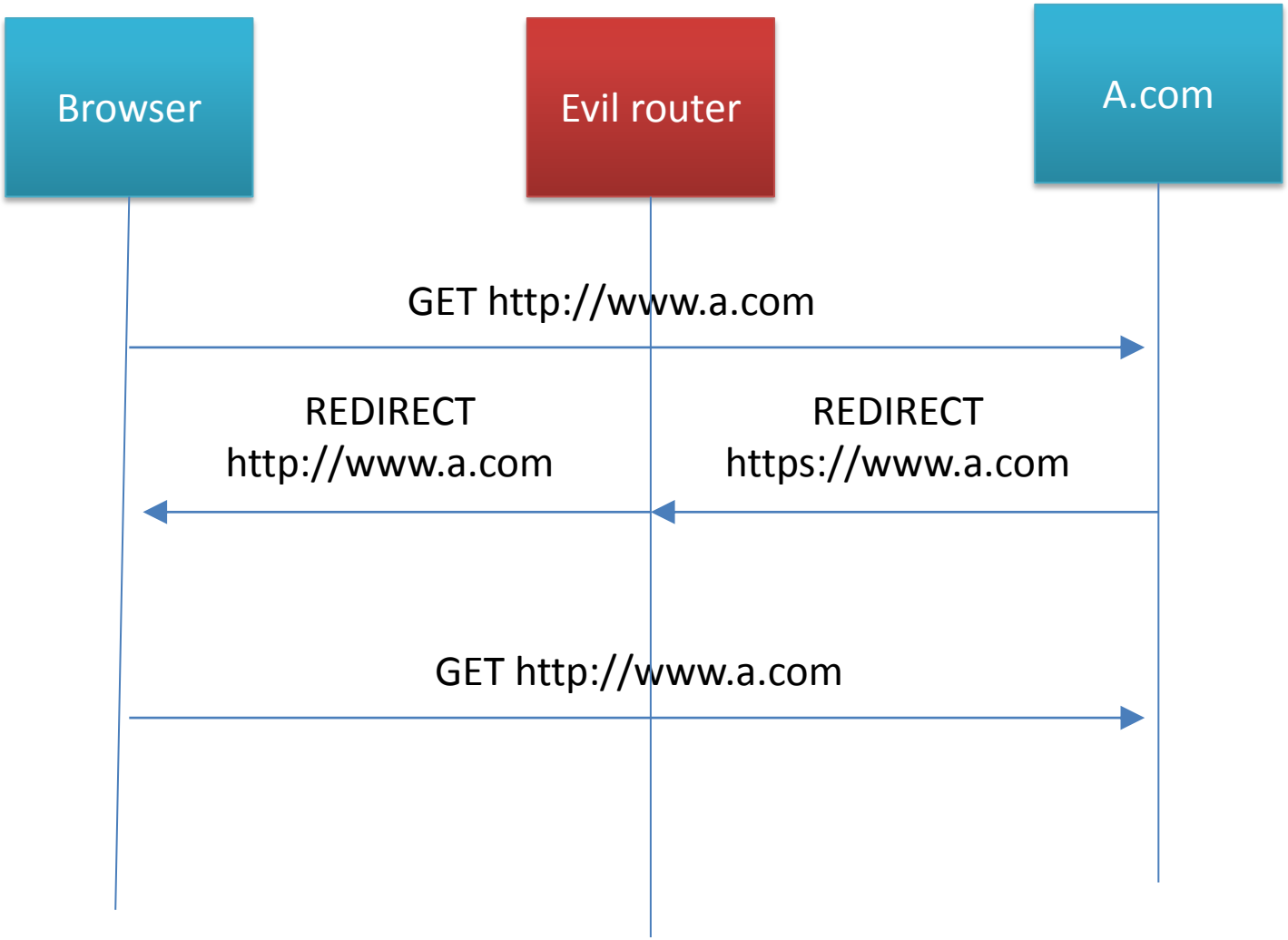
Content Security Policy

- Nutnosť rozdeliť zdroje do samostatných súborov
- “inline” <script> je blokovaný
  - ak v hlavičke nie je explicitne povolený cez “unsafe-inline” – neodporúča sa (znižuje ochranu)
- Umožňuje report chýb
  - report-uri /my_amazing_csp_report_parser;
  - Na danú adresu sa bude posielať zoznam zdrojov, ktoré boli zablokované
  - Podpora iba „reporting“ módu, bez blokovania

HTTP Strict Transport Security

- Používatelia sa na HTTPS stránky väčšinou dostanú presmerovaním z HTTP
  - Útočník však môže za istých okolností presmerovanie eliminovať
    - SSL stripping útok

HTTP Strict Transport Security



HTTP Strict Transport Security

- HTTP hlavička, ktorá minimalizuje SSL stripping útoky
- `Strict-Transport-Security: max-age=2592000; includeSubDomains`
  - Prehliadač si na dobu špecifikovanú v max-age zapamätá, že k stránke má pristupovať len cez HTTPS
  - Všekty odkazy a zdroje v stránke sú transformované na HTTPS
- Prvý dotaz na server (a prvý dotaz po expirácii max-age) nie je odolný voči SSL stripping útoku

X-Frame-Options

- Ochrana voči „Clickjacking“ útokom
  - Útočník zobrazí cieľovú stránku v Iframe, nad ktorým zobrazí svoj skrytý obsah.
  - Umožňuje „odchytávanie“ klikov a stlačení kláves
- V HTTP hlavičke je možné zakázať zobrazenie stránky v Iframe:
  - `X-Frame-Options: DENY | SAMEORIGIN | ALLOW-FROM origin`
- Navrhnuté Microsoftom v IE8

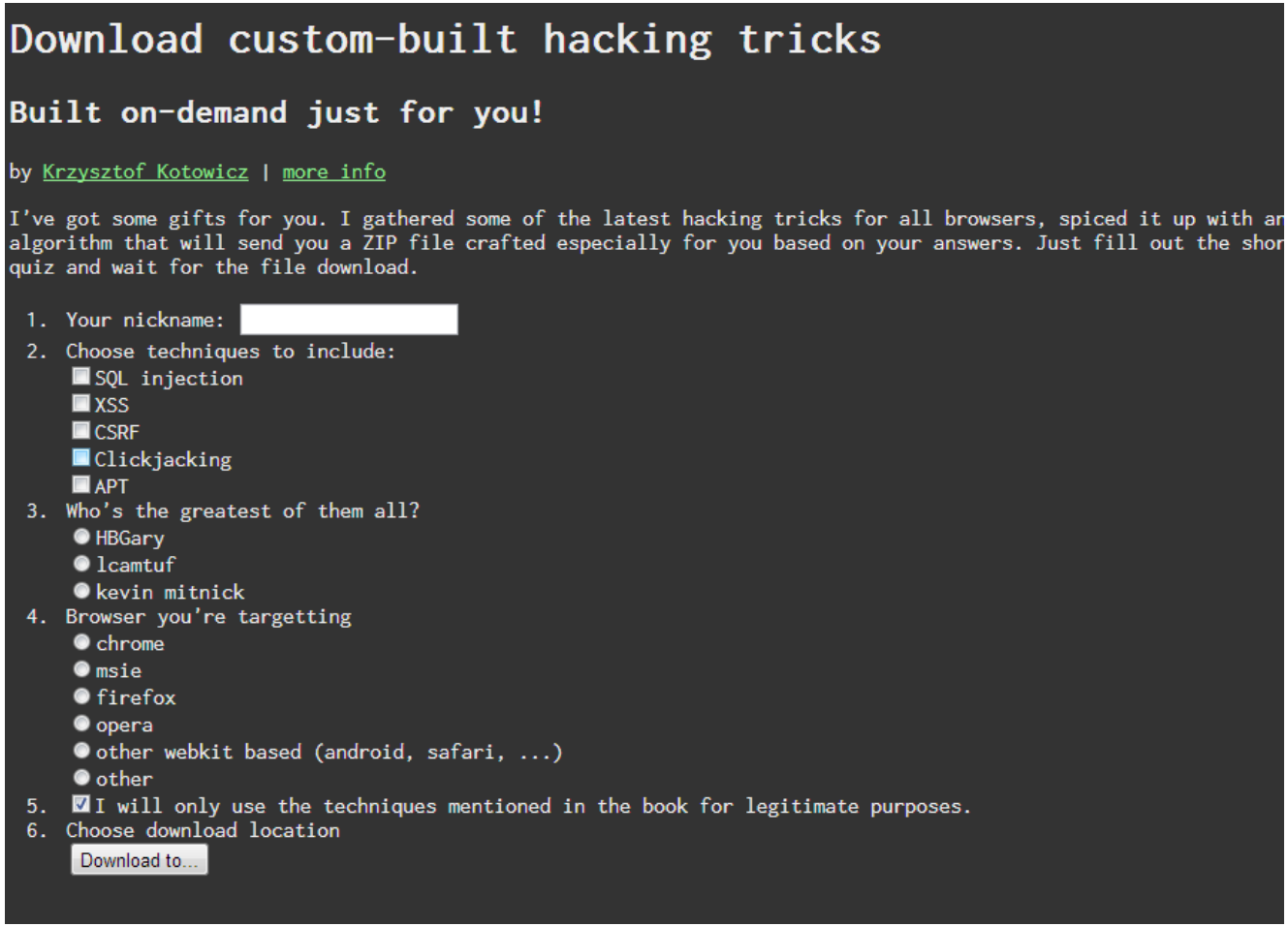
Ukážka útoku: file jacking

- HTML5 umožňuje upload celého adresára
  - <input type="file" directory>
- http://kotowicz.net/wu/
- V chrome može dialóg na vybratie adresára používateľa zmiasť (viď demo)

Ukážka útoku: file jacking

- Krystof Kotowicz vytvoril stránku simulujúcu útok na svojom blogu
  - Väčšinou návštevíci zaujímajúci sa o bezpečnosť
  - Za 13 mesiacov 298 klientov (217 IP) nahralo nejaký adresár (väčšinou Downloads)
  - Veľa zaujímavých súborov:
    - Downloads/#BeNaughtyLive.com/
    - Downloads/#GoLiveTrannies.com/
    - ..
  - Aj súkromné dáta
    - onlinePasswords.txt
    - Faktura_numer_26_2011_<company>.pdf
    - …

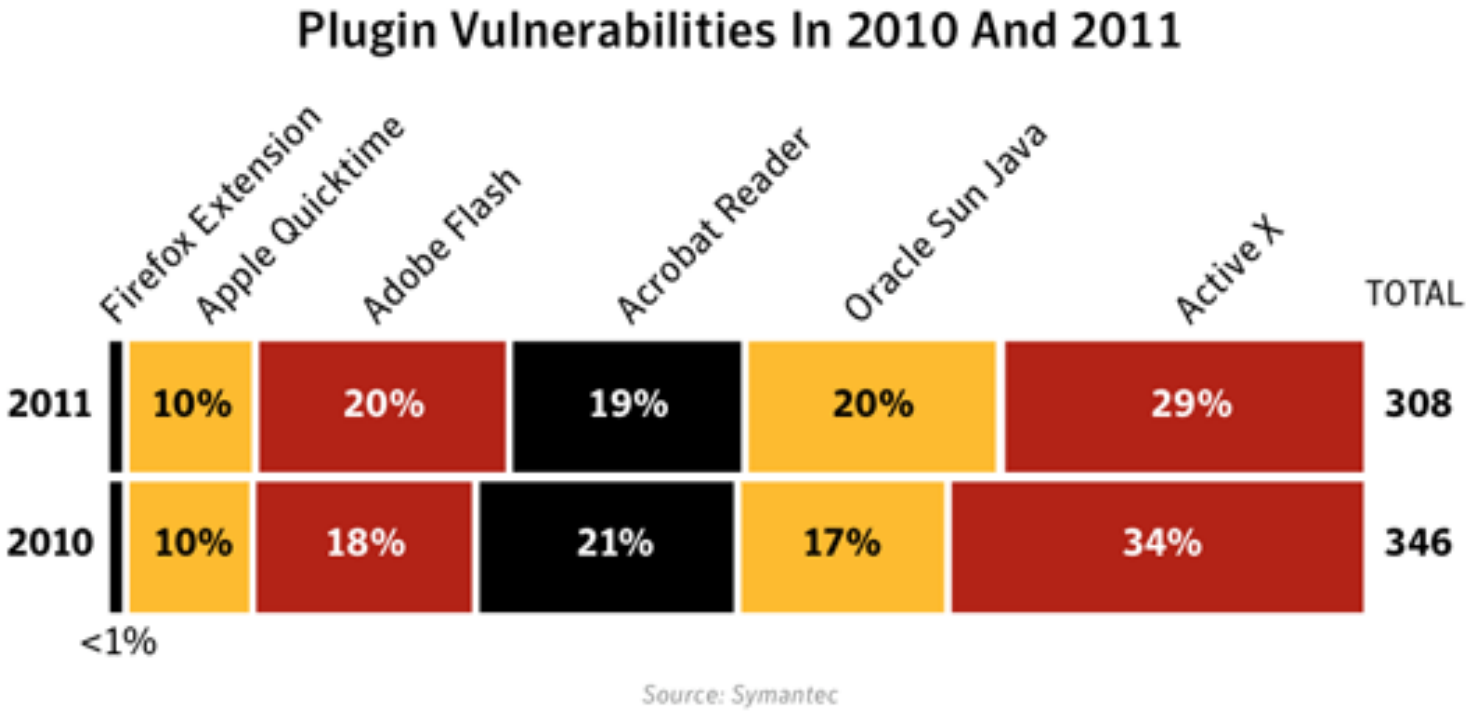
Ukážka útoku: file jacking - demo



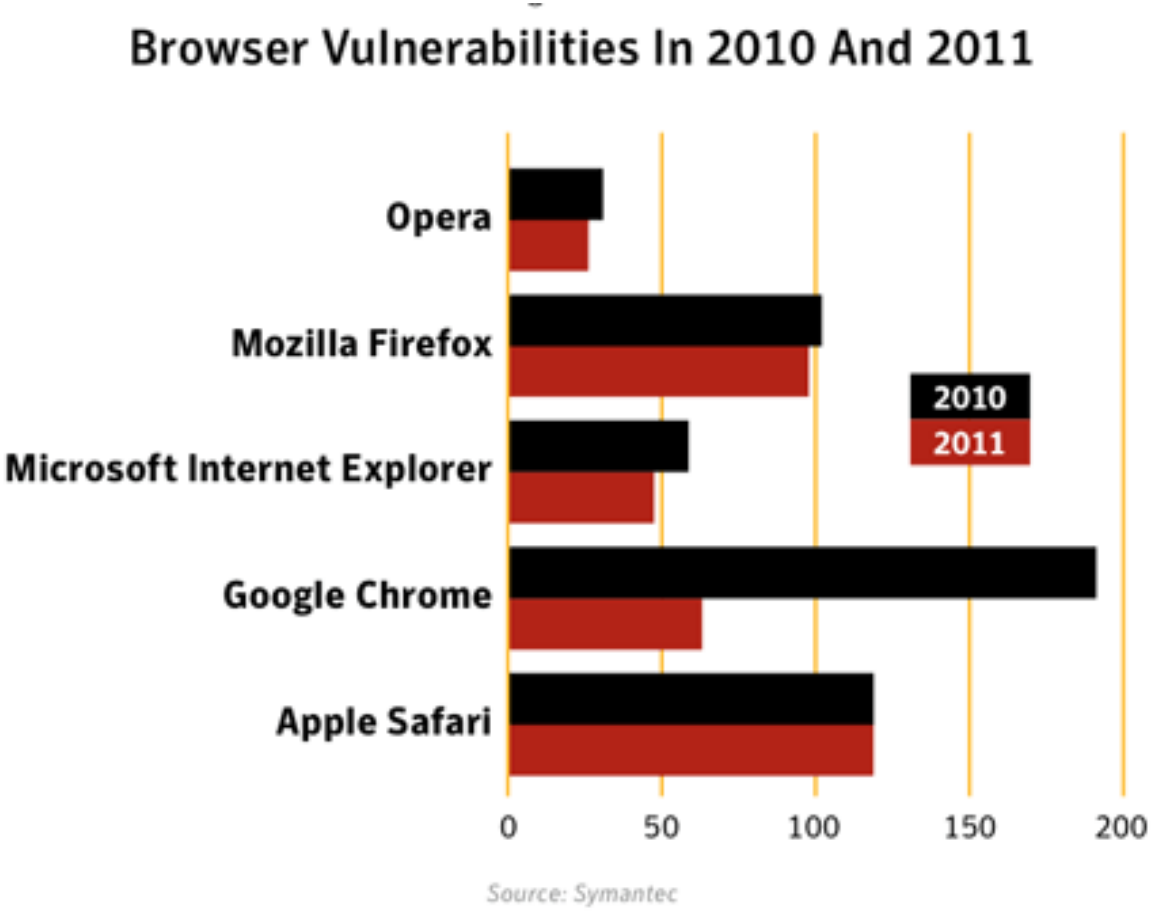
Záver

- Zdá sa, že HTML5 postupne nahradí Flash
- HTML5 poskytuje veľa užitočných rozšírení a funkcií
  - Veľa funkcií → veľa zraniteľností
- Treba vyzývať používateľov pravidelne aktualizovať prehliadač
  - Podpora starých prehliadačov je aj tak náročná
- Pri vytváraní web aplikácie využívať osvedčenú JavaScriptovú knižnicu (jQuery)
- Pokiaľ možno čo najviac využívať bezpečnostné funkcie HTML5

Symantec Internet Security Threat Report 2011



Symantec Internet Security Threat Report 2011



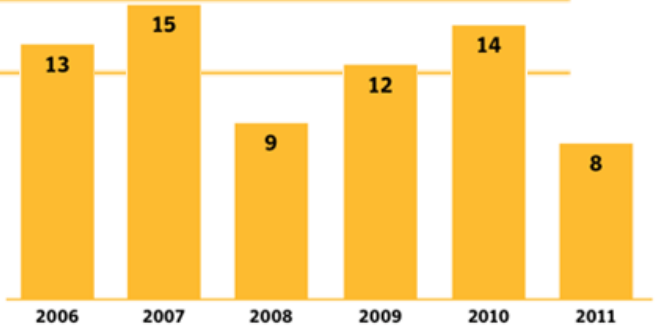
Symantec Internet Security Threat Report 2011

Zero-Day Vulnerabilities Identified In 2011

CVE Identifier	Description
CVE-2011-0609	Adobe Flash Player 'SWF' File Remote Memory Corruption Vulnerability
CVE-2011-0611	Adobe Flash Player 'SWF' File Remote Memory Corruption Vulnerability
CVE-2011-1255	Microsoft Internet Explorer Time Element Remote Code Execution Vulnerability
CVE-2011-1331	JustSystems Ichitaro Memory Management Program Remote Heap Buffer Overflow Vulnerability
CVE-2011-2107	Adobe Flash Player Cross-Site Scripting Vulnerability Alert
CVE-2011-2462	Adobe Reader/Acrobat U3D Memory Corruption Vulnerability
CVE-2011-3402	Win32k True Type Font Parsing Vulnerability
CVE-2011-3544	Oracle Java Rhino Script Engine

Source: Symantec

Volume Of Zero-Day Vulnerabilities 2006 – 2011



Source: Symantec

Zdroje, Referencie

- Flashsec Wiki
- OWASP – Finding vulnerabilities in flash applications
- Adobe Flash Player 10 security white paper
  - http://www.adobe.com/devnet/flashplayer/articles/flash_player10_security_wp.html
- Symantec Internet Security Threat Report
  - http://www.symantec.com/threatreport/
- HTML 5 rocks
  - http://www.html5rocks.com/
- http://blog.kotowicz.net
- http://www.andlabs.org