# Advanced Client-Side Security: What many users do not know

SET09103 Advanced Web Technologies

School of Computing
Napier University, Edinburgh, UK
Module Leader: Uta Priss

2010

# Outline

# The biggest security risk on the web is ...

# The biggest security risk on the web is ...

... Javascript!

# The biggest security risk on the web is ...

... Javascript!

Many current web-based exploits (trojans, viruses, etc) rely on Javascript in one form or other.

# Browser history stealing

Browsers display visited links in a different colour than unvisited links.

Javascript has access to this information. (It is a DOM-accessible page state).

The owner of a website that you visit can determine which other websites are in your browser's history.

# Browser cache timer attacks

It takes longer to load a remote site than a site from the local cache.

The owner of a website that you visit can determine which other websites are in your browser's cache (at least if Javascript is turned on).

# Exploit example: tabnabbing

- Replace a tab which is in the background with a Gmail (or banking) login page.
- The user thinks they logged out and logs in again.
- Attacker now has the username and password of the user.
- The user is then directed to the proper login page and does not notice the hack.

Using history sniffing or timer attacks an attacker can find out which email (or bank account) a user is using.

# Exploit example: de-anonymisation of social network users

Information about group memberships (which groups within a social network a user belongs to) may be enough to identify the login name of a social network user.

Group memberships can be learned via history sniffing.

An attacker can match the login name of a social network user to the current visitor of the attacker's page.

# Is SVG secure?

Because browsers have to do more work to display SVG, there might be more risks.

There were security exploits with older versions of Adobe Viewer and WebKit (used by Safari and Chrome).

Some exploits make use of Javascript embedded in SVG.

# Adobe Flash Cookies

Many users do not know that Flash stores cookies in some other location than the cookies folder.

Officially, users can only view/delete the cookies on Adobe's website.

Flash does not work if cookies are disabled (by removing write permission from the Flash Cookies directory).

Some Flash cookies restore deleted ordinary cookies.

# Java Web Start

Used these days instead of applets.

Downloads and installs an application (jar files) into a client side directory.

Reloads the application each time the user visits the corresponding website, but can also run off-line.

Security risks?

# Private browsing

Modern browsers often have a "privacy mode" which presumably makes sure that information about the session (e.g. which websites were visited) is not stored on the browser.

How well does this work?

Threat from local attacker versus threat from web attacker.

# What needs to be saved/deleted by the browser?

Should be deleted (maybe):

- history
- cache
- cookies
- form autocompletion

Should be kept (maybe):

- downloaded files
- passwords
- SSL certificates
- bookmarks
- installed browser patch/plugin

# Extensions used while browser in “private mode”

- Extensions are not necessarily disabled while in “private mode”.
- Extensions can have behaviour that is inconsistent with “private mode”.
- Flash cookies are probably not affected by “private mode”.

# "Fingerprinting a browser"

Presumably a remote website only knows the IP address of a client.

Because web browsers transmit other information (browser type, installed plug-ins and fonts, screen resolution, timezone) which are available via environment variables or Javascript, many browsers can be uniquely "fingerprinted".

# What users can do:

- Install extensions that protect privacy/improve security.
- Hide IP address, reduce information transmitted by browser.
- Turn private browsing on.
- Use several browsers (one browser with Javascript; one without Javascript; one for social networking, etc).