Overtaking Google Desktop
Leveraging XSS to Raise Havoc

Yair Amit
Senior Security Researcher, Watchfire
yaira@watchfire.com
+972-9-9586077 ext 4039

Copyright © 2007 - The OWASP Foundation
Permission is granted to copy, distribute and/or modify this document under the terms of the Creative Commons Attribution-ShareAlike 2.5 License. To view this license, visit http://creativecommons.org/licenses/by-sa/2.5/

The OWASP Foundation
http://www.owasp.org/
Presentation Outline

■ Background
■ Google Desktop Overview
■ Overtaking Google Desktop – Step by Step
■ Impact
  ▸ What harm can a malicious attacker do?
  ▸ Attack characteristics
■ Lessons learned
■ Q&A
Background

■ XSS
  ▶ The most widespread web-application vulnerability
    ▪ WASC Web Application Security Statistics Project
      (http://www.webappsec.org/projects/statistics/)
  ▶ Used to be perceived as an identity theft attack
  ▶ XSS has so much more to offer. It has teeth!
    ▪ Change settings and steal data from attacked victim account
    ▪ Web worms (Samy)

■ What we are about to see...
  ▶ Stealth attack
  ▶ Sensitive information theft from the local computer
  ▶ Command execution
Google Desktop - Overview

■ Purpose: provide an easily to use and powerful search capability on local and other personal content

■ Some traits:
  ▸ Runs a local web-server for interaction (port 4664)
    ▷ Google.com like interface
  ▸ Uses a service to run the indexing
  ▸ User interface is almost purely web
  ▸ Preferences control what to index, and indexing can be broad
    ▷ Office documents, media files, web history cache, chat sessions, etc.
    ▷ Easily extendible
  ▸ Special integration with Google.com
Google Desktop Security Mechanisms

- Web server only accessible from localhost
  - Not available from network
Google Desktop Protection Mechanism (cont.)

- The main threats are XSS and XSRF attacks.
- Every request (except some images) has a unique signature:
  - Signature is generated using a strong key stored in the registry.
  - If signature doesn’t match query, request is denied.
  - Key is different per installation:
    - Signatures cannot be deduced from one installation to another.
- A powerful protection against XSS and XSRF.
Signatures Protection Strength Example

2
“Great... What do you plan to do with it?”

4
“Do you have the unique signature for Jacky's preferences page?”

1
“I found XSS in the preferences page of Google Desktop!”

3
“Classical XSS attack. I plan to lure Jacky to click on a malicious link.”

5
“No, I don’t! Damn it! Can this protection be bypassed?”

Attacker’s friend

Malicious attacker

Jacky

Jacky’s computer

* Jacky is a Google Desktop user

Google Desktop Vulnerability – Sticky XSS

- Available through the “under” keyword
  - For searching under specific folders in the hard-drive or a network drive.
- XSS is Sticky
  - Saved in the history of the “under” option
- Stickiness applies to all search results
  - “Under” history shown on all search results (added for usability)
- Stickiness requires 3 “overwrites” to be cleared
- How can this vulnerability be exploited, given the protection mechanisms?
  - http://127.0.0.1:4664/search?q=under:XSS_PAYLOAD&flags=68&num=10&s=9pKHqow9s-J4YfGgBjGF75g-ZwM
Google Desktop & Google.com integration

- Google Desktop interjects between browser and website, and adds content
  - Google Desktop search results are displayed in Google.com’s results
  - ‘Desktop’ link – our way in...
Google Desktop & Google.com integration: Our way in

- JavaScript on site has access to modified content
- Signature can be harvested
  - Interesting point: Google.com-originating searches all use the same signature
- This cannot be turned off...
  - Possible in newer versions
- Attacker needs control over victim’s browser in Google.com context...
Google.com XSS Vulnerability

- Standard XSS
- For the purpose of demonstration, a UTF-7 XSS vulnerability on search page is used.
- Can apply to any XSS on Google.com and some of its subdomains
  - And there are plenty of those...
Complete overtaking process

- Perform Google.com XSS exploit
  - Through SPAM mail, talkback links, social networks worms, etc. – the usual way

- Injected JavaScript will do the rest...
  - Harvest the signature from the search results
  - Infect the local machine by issuing XXSed Google Desktop search query (using the acquired signature)
  - Hide all traces of that occurring...

- The system is now fully compromised!
What harm can a malicious attacker do?

- Take advantage of Google Desktop’s powerful search and indexing capabilities
  - Search for sensitive information
- Change user preferences to index more local information
- “Search Across Computers”
  - Hijacking information with style. ;)
- Execute commands through Google Desktop
  - Change preferences to index network drives
  - Complete takeover...
Web User Interface...

- Attacker controls what the victim sees!
- Hide changed preferences options
- Hide version
  - Make the user think he’s using a more current version
- Auto-correction if “under” parameter is used with other values
  - Makes sure the JavaScript malware remains active
Attack Characteristics

- Low footprint
  - No need for malicious binary code to be injected
  - The code is automatically executed by the browser when visiting legitimate Google Desktop Web pages

- Easy data leakage
  - Hijacked information can be covertly leaked back to the attacker via seemingly innocent encoded requests to an external Web site

- Almost undetectable
  - No mangled URL in the address bar
  - The attack continues to persist across sessions and across browsers
Lessons Learned

■ XSS is a big issue
  ‣ Very common
  ‣ Very dangerous
    ▪ Sticky XSS is even worse
  ‣ Should be taken more seriously in the development process

■ Applications like Google Desktop are risky
  ‣ Access to sensitive information means greater risk for the user
  ‣ RIA trend

■ Integration between web applications and desktop applications is risky
  ‣ The attack took advantage of this integration in order to overcome powerful protection mechanisms
  ‣ Classical functionality/security tradeoff

■ Antivirus vendors should find creative ways to fight JavaScript Malware
More Information

- Short Overview:

- White paper:

- Video Demo (11 Minutes):
  http://download.watchfire.com/googledesktopdemo/index.htm
Questions?

Q & A
Thank you! 😊

GOODBYE!!!