Exposing Web Vulnerabilities

The State of Web Application Security

by Nick von Dadelszen

security-assessment.com

Security-Assessment.com – Who We Are

- NZ's only pure-play security firm
- Largest team of security professionals in NZ
- Offices in Auckland, Wellington and Sydney
- Specialisation in multiple security fields
  - Security assessment
  - Security management
  - Forensics / incident response
  - Research and development

Web Application Trends

- Still seeing old issues
  - XSS, SQL injection, parameter manipulation
- New ways to find and exploit existing issues
  - Input validation, Google
- Move to hacking the client
  - Phishing, man-in-the-middle, trojans

Examples Of New Attacks

Null Byte Upload

.Net XSS Filtering Bypass

HTTP Header Manipulation

Null Byte Upload 1

- ASP has trouble handling Null bytes when using FileScripting Object
- Take the following HTML code:

```
<form method=post enctype="multipart/form-data"
  action=upload.asp>
Your Picture: <input type=file name=YourFile>
<input type=submit name=submit value="Upload">
</form>
```

Null Byte Upload 2

- Form posts to the following ASP code:

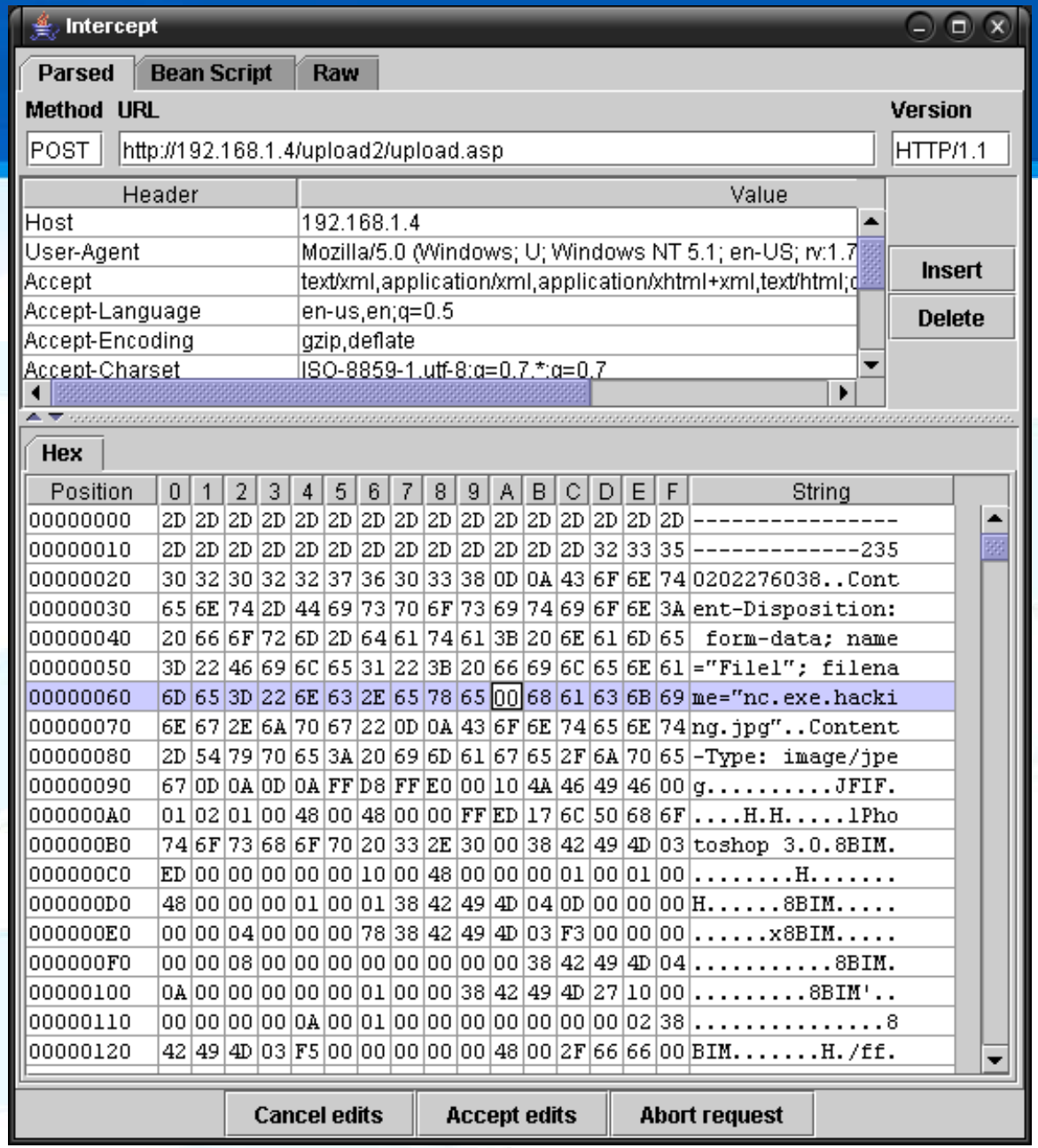
```
Public Sub Save(Path)
   Set objFSO =
   Server.CreateObject("Scripting.FileSystemObject")
   Set objFSOFile =
         objFSO.CreateTextFile(objFSO.BuildPath(Path,
   tFile + ".bmp"))
   ‘ Write the file contents
         objFSOFile.Close
End Sub
```

Null Byte Upload 3

- If the POSTED filename contains a NULL byte, the FileSystem object only uses the information up to the NULL byte to create the file

  nc.exe<0x00>test.bmp creates nc.exe in file system

- Must use Proxy to change filename
- WebScarab Handles Hex natively



.Net XSS Filtering Bypass 1

- ASP.Net 1.1 contains request Validation
- Built-in validators allow out-of-the-box protection for XSS and SQL injection
- Has a flaw allowing bypass of the filters
- Validator bans all strings in the form of <letter
- Close tags are allowed

.Net XSS Filtering Bypass 2

- Bypass performed by adding a NULL byte between the < and the letter

  foo.bar/test.asp?term=<%00SCRIPT>alert('Vulnerable')</SCRIPT>

- Validator no longer sees this as an invalid tag and allows it through

Browsers disregard NULL bytes when parsing so HTML code is still run

HTTP Header Manipulation 1

- HTTP Response headers are set by the server
- When user input is included in headers then an attacker can control those headers
- Examples of user input included in headers are:
  - Cookies
  - Redirections
  - Referer

HTTP Header Manipulation 2

- Standard redirect
  - Request:
    - www.example.com/redirect.asp?query=test
  - Response headers:
    - HTTP/1.1 302 Object moved
    - Location: /index.html?query=test

HTTP Header Manipulation 3

- Header Insertion
  - Request:
    - www.example.com/redirect.asp?query=test%0d%0aNew%20Header:%20blah
  - Response headers:
    - HTTP/1.1 302 Object moved
    - Location: /index.html?query=test
    - New Header: blah

HTTP Header Manipulation 4

- Malicious Redirect (Mozilla Only)
  - Request:
    - www.example.com/redirect.asp?query=test%0d%0aLocation:%20http://www.google.com

  Response headers:
    - HTTP/1.1 302 Object moved
    - Location: /index.html?query=test
    - Location: http://www.google.com

Examples Of Other Recent Issues

- .Net authentication bypass
- <script> tag escaping
- Use of TRACE to capture authentication credentials
- HTTP response splitting
- Session riding

GoogleMonster

Using The Google Search Engine For Underhand Purposes

security-assessment.com

Google

- Google is a great search tool
  - Trolls Internet searching for pages
  - Finds pages based on links
  - Finds even those pages you don't want people to know about
  - Caches pages

Simple Start

- We can use a standard Google search to find interesting pages such as indexes.
  - "index of /etc"
  - "index of /etc" passwd
  - "index of /etc" shadow
- Lots of irrelevant results

Advanced Operators

- Google allows us to do more than just simple searching using advanced operators
- E.g.
  - filetype:
  - inanchor:
  - intext:
  - intitle:
  - inurl:
  - site:

  - intitle:index.of./etc passwd
  - filetype:mdb users.mdb

Combining Operators

- We can combine multiple operators to create very specific searches
  - filetype:eml eml +intext:"Subject" +intext:"From" +intext:"To"
  - "# -FrontPage-" ext:pwd inurl:(service | authors | administrators | users) "# -FrontPage-" inurl:service.pwd

Searching For Vulnerabilities

- We can use Google to search for specific web vulnerabilities
  - +"Powered by phpBB 2.0.6..10" -phpbb.com -phpbb.pl
  - inurl:citrix/metaframexp/default/login.asp?ClientDetection=On

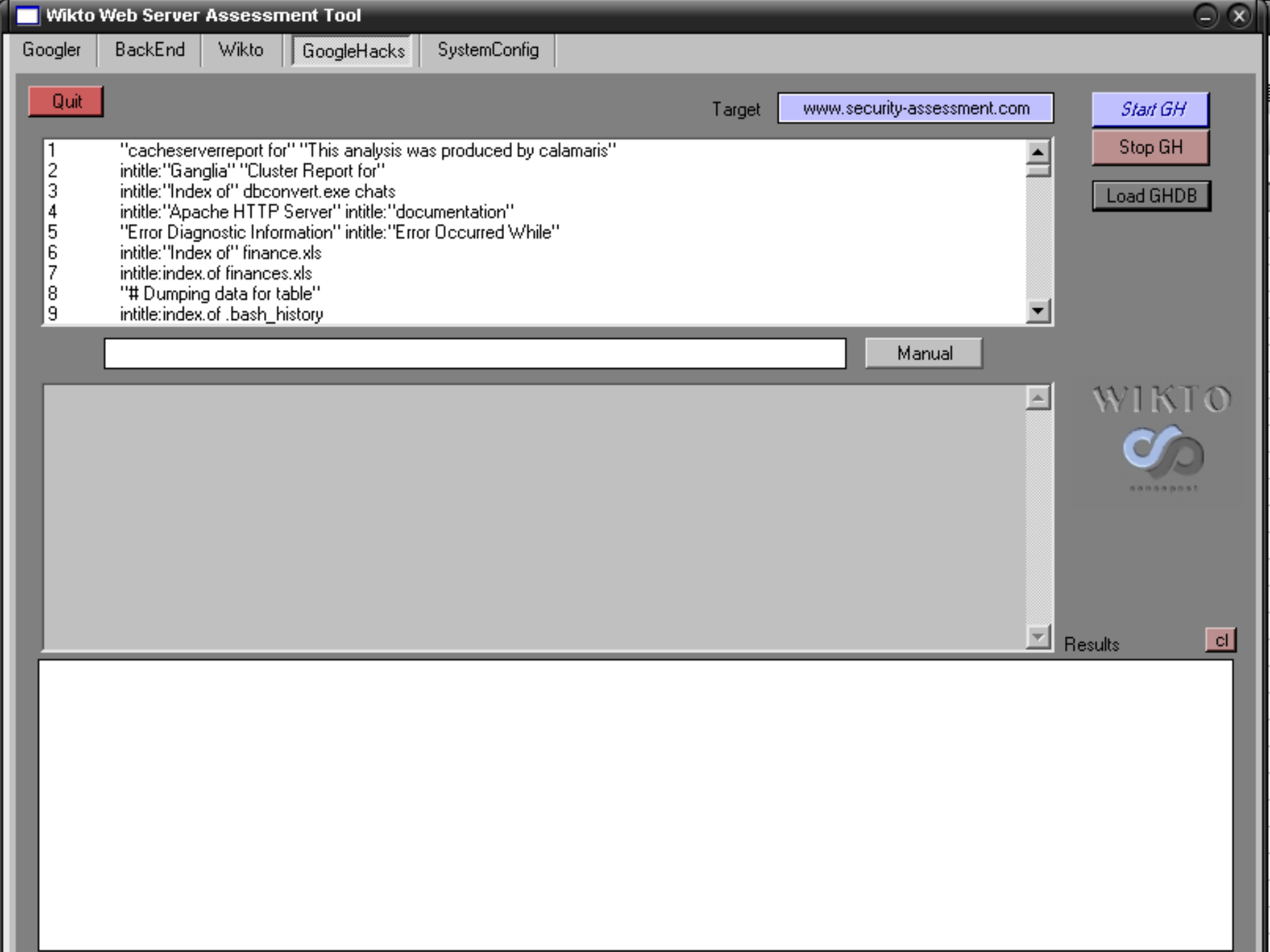
Enter the GHDB

- GHDB = Google Hacking Database
- Over 900 unique search criteria for finding information
- Created and maintained at johhny.ihackstuff.com

Targeting Websites

- We can use the site: operator to restrict queries to a particular domain
- This allows an attacker to use Google to test a site for vulnerabilities without actually touching that site.

- Enter Wikto – Web Server Assessment Tool


Wikto Web Server Assessment Tool
Googler
BackEnd
Wikto
GoogleHacks
SystemConfig
Quit
Target
www.security-assessment.com
Start GH
Stop GH
Load GHDB
1 "cacheserverreport for" "This analysis was produced by calamaris"
2 intitle:"Ganglia" "Cluster Report for"
3 intitle:"Index of" dbconvert.exe chats
4 intitle:"Apache HTTP Server" intitle:"documentation"
5 "Error Diagnostic Information" intitle:"Error Occurred While"
6 intitle:"Index of" finance.xls
7 intitle:index.of finances.xls
8 "# Dumping data for table"
9 intitle:index.of .bash_history
Manual
WIKTO
Results
cl

Protecting Against Client Attacks

Will Two-Factor Authentication Help?

What is Two-Factor Authentication

- Many different types of two-factor
  - One-time passwords
    - Password-generating token (SecureID, Vasco)
    - SMS tokens
    - Scratch pads
  - Client-side Certificates
    - Smart cards
    - USB keys
  - Biometrics

The Trouble With Two-Factor

Designed for small user base

- Has a usability cost
- No clear market leader
- Potentially large implementation costs
- Will not stop all attacks
  - Man-in-the-middle
  - Intelligent Trojans

The Weakness Of SSL

- Relies on trust
- Tells you that you have a secure session with A website, not THE website
- Certificates can be faked
- Root certificates can be installed – MarketScore
- Allows for Man-in-the-middle and IDN attacks

MITM vs Two-Factor

Customer	Man-in-the-middle		Website
-> Here is my username and password			
		-> Here is my username and password	
		<- What is your token password?	
<- What is your token password?			
-> Here is my token password			
		-> Here is my token password	
		<- Authenticated	
<- Authenticated			
-> Transfer $10 to Bill			
		-> Transfer $10 to Fred	
		<- Please re-authenticate	
<- Please re-authenticate			
-> Here is my token password			
		-> Here is my token password	
		<- Transaction accepted	
<- Transaction accepted			

Will Two-Factor Help?

- Does increase security
- Makes attacks harder
- Will require attacks to be more focused

- Must be a business decision
  - Amount of security required
  - Cost vs benefit

Defence Against Client Attacks

- Authentication is the key
  - Client authentication
  - Server authentication
- Users must protect themselves
  - Don’t use public terminals
  - Anti-virus
  - Firewall
  - Automatic updates

Questions?