Case Studies in Discovering Previously Unknown Web Application Vulnerabilities

Kenneth F. Belva, CISSP

Franklin Technologies United, Inc.

http://www.ftusecurity.com

Disclaimer

- The opinions in this presentation are my own and not my current employer’s

General Principals (pt. 1)

- What is the purpose of the application?
  - Banking app, blog, CMS?
  - Purpose of app hints at weak functionality in core code
  - Don't overlook non-core code

General Principals (pt. 2)

- How does the application work?
  - Critical features / functionality of the application
    - Can the user upload files?
    - Can the user send email?
  - Reoccurring security issues for functionality type

General Principals (pt. 3)

- How does it handle the following:
  - User Supplied Input
  - Authentication (and ACL permissions)
  - Session Management

User Supplied Input

- Not handling user input properly:
  - SQL Injection
  - XSS Issues
  - File Uploads
- Simple code changes may introduce critical security flaw

User Supplied Input Solution

- Lesson:
  - It's in the details - small changes could have large consequences
  - Run all parameters through a character filtering routine

Session Management

- Not handling a user session properly:
  - Finding Past or “Expired” Tokens
  - Cookie Tampering
  - Header Management

Session Management Solutions

- Create tokens that are long random strings
- Expire tokens after 30 minutes of non-use
- Keep as many needed fields in session variables (ideal = only session token available to app)

Authentication Issues

- Privilege Escalation
- Insecure ACL handling allow access to restricted content
- Test without authentication and with accounts at various privilege levels

Authentication Issues Solution

- Use “include” files to maintain consistency throughout application and across apps

Don't Doubt Google

- How widespread is the issue?
- Find sites that are running the same application/code base
- “allinurl:” - parameter to find URLs would uniquely match that application

Case Study 1: Online Banking Application (pt. 1)

- Config Error: internal, well known upload application internet-facing in well known server location
- Config Error: uploaded files could be executed by webserver from browser request

Case Study 1:
Online Banking Application (pt. 2)

- File types were restricted client-side: rewrote “front end” to upload my custom attack scripts
- Attacker could access internal servers from the DMZ via trusted ports/connections

Case Study 1:
Online Banking Application (pt. 3)

- Lessons:
  - Remove non-core applications
  - Restrict internet-facing apps
  - Uploaded file should not be under web root that could be executed by the browser

Case Study 2:
Banking Application Purchased from China (pt. 1)

- SQL Injection issue
  - Complex SQL statement could be simplified if one had prior knowledge of it, but not guessable
  - Execute *calc.exe* on DB Server

Case Study 2: Banking Application Purchased from China (pt. 2)

- Lessons:
  - 3rd Party/COTS/custom software independently audited
  - International ramifications
  - Deep access into internal network

Case Study 3: What's Up Gold Professional (pt. 1)

- Monitor Servers: Network Configuration
- Trusted Console based on Headers in HTTP Request
- Trust Console is given Administrator Access

Case Study 3:
What's Up Gold Professional (pt. 2)

- Trusted Console = Admin Access = Privilege Escalation
- Gives attacker network topology

Case Study 3: What's Up Gold Professional (pt. 3)

- Used Google “allinurl:” to determine who out there was running the application, internet-facing
  - National lab funded by US Gov't
  - Educational Institutions

Case Study 3: What's Up Gold Professional (pt. 4)

- Lessons:
  - Do not put software facing the internet if it is not necessary
  - Google will find your internet-facing application
  - Do not make HTTP headers trusted: They can be spoofed

Case Study 4:
Online Ordering System (pt. 1)

- HTTP header Cookie Tampering
- By changing the session number, older sessions could be retrieved

Case Study 4:
Online Ordering System (pt. 2)

- Previous Orders Info included:
  - Name, Addresses and Phone numbers
  - Selection of products
  - Could not retrieve financial information

Case Study 4:
Online Ordering System (pt. 3)

- Lesson:
  - Violation of privacy despite lack of information to commit fraud
  - The information was a marketer's dream: we know customer preferences

Case Study 5: SimplePHPBlog 0.4.0 (pt. 1)

- Unauthenticated user can access sensitive functions
- Allowed for complete remote compromise of application and possibly webserver

Case Study 5: SimplePHPBlog 0.4.0 (pt. 2)

- Lessons:
  - Make sure authentication is uniformly applied
  - Delete unnecessary files/functionality from application

Case Study 6: A social networking website (similar to myspace.com) pt. 1

- Anyone may create an account
- UserID stored in Cookie in field named “userid=”
- Replace the UserID in the cookie to become another user

Case Study 6:
A social networking website (similar to myspace.com) pt. 2

- Full access as other user
  - Read and write email
  - Change their profile
    - Change Picture
    - Details: Dating Preferences

Case Study 6: A social networking website (similar to myspace.com) pt. 3

- Lesson:
  - –Place fields, other than session token, in server session variables

Case Study 7:
Heath Benefits System (pt. 1)

- Weak ACLs: unauthenticated user requests web based report/ query engine with full DB access
- Query engine found without authenticating to app: link was not displayed to the end user

Case Study 7:
Heath Benefits System (pt. 2)

- Used Google to find other companies that were using the same system
- Other companies used same code base and were also vulnerable

Case Study 7:
Heath Benefits System (pt. 3)

- Lessons:
  - URLs may be discovered even if not displayed to end user
  - Ask providers for proof that their app was pen tested

Additional Topics to Consider

- Testing: automated vs manual
  - Manual may reveal unique situations
    - IM from user revealed session token
  - Automated Code Auditing
    - potential problems/issues

Additional Topics to Consider

- Lessons:
  - Both testing methods are part of the solution, but are not the total solution in themselves
  - Sometimes manual testing can find bugs - automated tools cannot

A Few Techniques for Discovering Unknown Web Application Vulnerabilities (pt. 1)

- Trace All User Supplied Input (including URLs)
  - Normally everywhere there is a "=" sign: GET and POST

A Few Techniques for Discovering Unknown Web Application Vulnerabilities (pt. 2)

- Test Existing ACLs before and after Authentication for uniformity across all pages
- Examine Parameters in HTTP Header Requests

Free Tools for Vulnerability Assessment and Discovery (pt. 1)

- Webscarab – Application Web Proxy
- Perl – Good for scripting most HTTP exploits

Free Tools for Vulnerability Assessment and Discovery (pt. 2)

- Grep – used to search for certain strings in code (normally used after reading code)
- Nessus – vulnerability scanner with some web application scripts for determining possible vulnerable URLs via parameters

Resources

- OWASP – http://www.owasp.org
- SecurityFocus – http://www.securityfocus.com
- Full Disclosure - http://lists.grok.org.uk/pipermail/full-disclosure/

Biography of Kenneth F. Belva, CISSP

- Currently employed at Credit Industriel et Commercial (New York)
  - Manages the Information Technology Risk Management Program
  - Reports directly to the Senior Vice President and Deputy General Manager
- On the Board of Directors for the New York Metro Chapter of the Information Systems Security Association
- Authored:
  - The contrarian paper: “How It’s Difficult to Ruin A Good Name: An Analysis of Reputation Risk”
  - Chapter “Encryption in XML” in *Hackproofing XML* published by Syngress
- Taught as an Adjunct Professor in the Business Computer Systems Department at the State University of New York at Farmingdale
- Credited by Microsoft and IBM for discovering vulnerabilities in their software
- Holds the Certified Information Systems Security Professional (CISSP), Certified Ethical Hacker (CEH) certifications and has passed the Certified Information Security Manager (CISM) exam
- Presented on topics such as patch management; Moderated a panel discussion on corporate governance

Security Website of Kenneth F. Belva, CISSP

- Main Website: http://www.ftusecurity.com
- Security Blog: http://www.ftusecurity.com/blog/