

# Are You Feeling Secure?

*Shlomy Gantz*

*President, Bluebrick Inc.*



# About me

- President, BlueBrick Inc.
- 15 Years Software Development / Project Management
  - ✓ 12 Years of Web Experience
  - ✓ 10 Years of CF Development
  - ✓ Adobe Certified Instructor/Developer
  - ✓ Member, Adobe Community Expert
  - ✓ Manager, NYFLEX user group
- Author/Speaker
  - ✓ CFUNITED, MAX...
  - ✓ Co-Author - ColdFusion Developer's Handbook
  - ✓ ColdFusion Developer's Journal

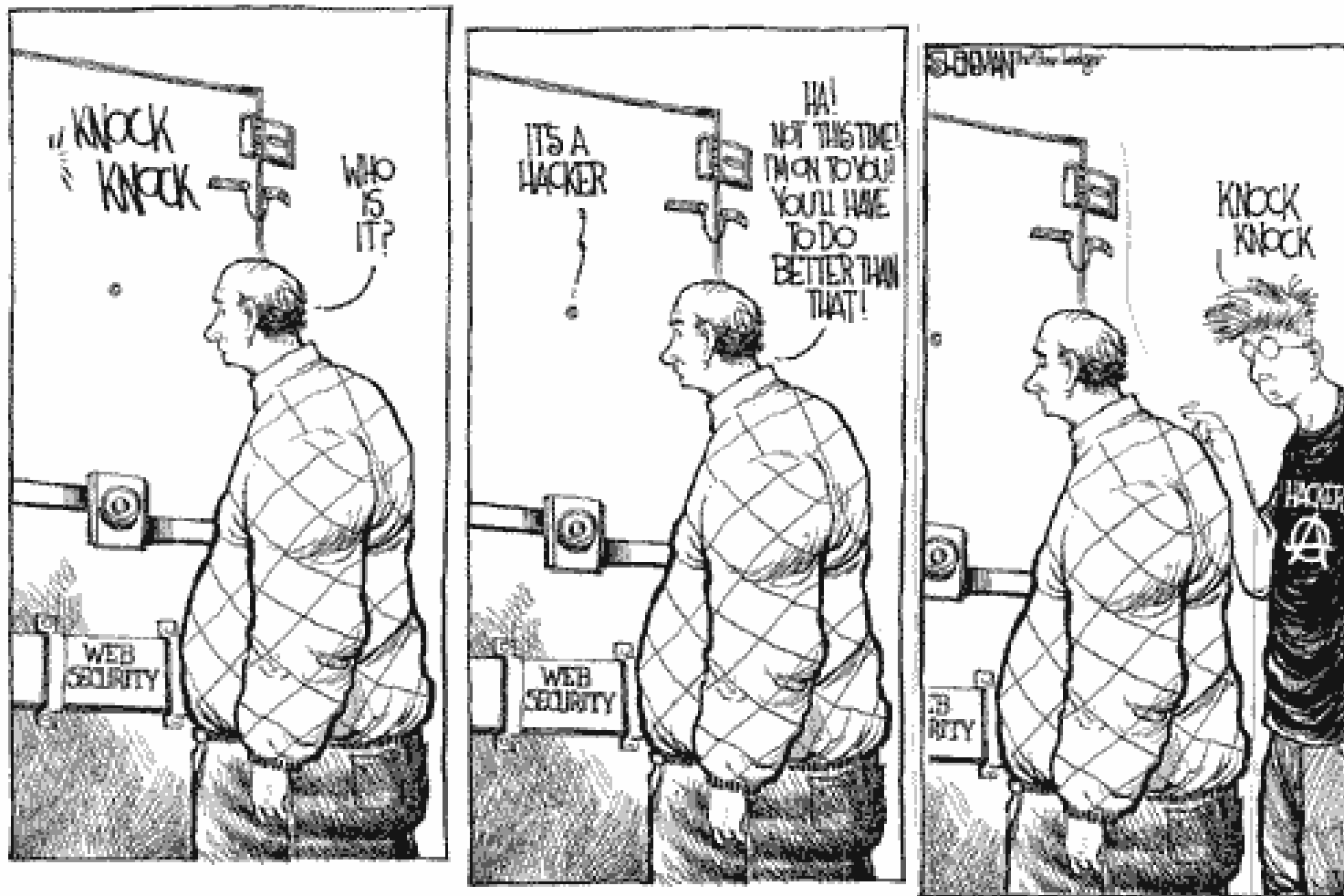


# Agenda

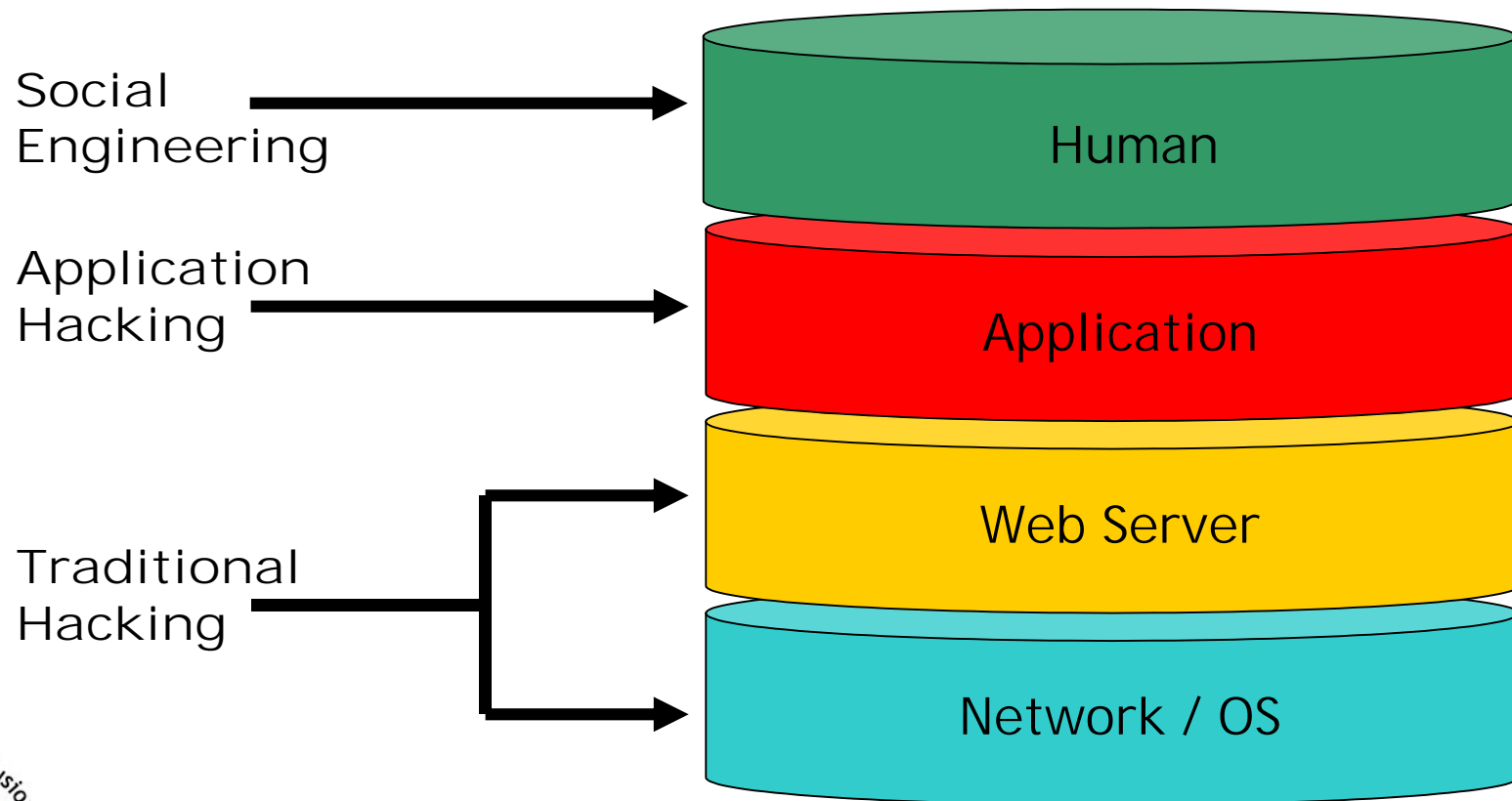
- Progression Of Attack
- Application Attacks
- Vulnerabilities - Information Gathering
- Vulnerabilities - Application Top 10
- Additional Information
- Q&A



# Thinking Like A Hacker



# Layers Of Vulnerability



# Potential Risks

- Information Disclosure
- Data loss / Data manipulation
- Application Downtime
- Damage to reputation



CFUnited Express March 2007

# Progression Of An Attack

- Assessment
  - ✓ Footprinting
  - ✓ Scanning
  - ✓ Enumeration
- Exploitation
  - ✓ Gaining Access
  - ✓ Escalating Privileges
  - ✓ Data Theft/Pilfering
  - ✓ Covering Tracks
  - Creating Backdoors



# Assessment

- **Footprinting**

- ✓ Whois
- ✓ DNS Lookup
- ✓ HTTP Headers

- **Scanning -**

- ✓ IP Addresses - Ping Sweep
- ✓ Port Scan (TCP/UDP)

- **Enumeration**



# Exploitation

- **Gaining Access**
  - ✓ Password/ Session Hijacking
  - ✓ Buffer Overflows
  - ✓ Application Vulnerabilities
- **Escalating Privilege**
  - ✓ Password Cracking
  - ✓ Application Vulnerabilities
- **Data Theft/Pilfering**
- **Covering Tracks**
  - Loading A “Root Kit”
  - ✓ Clear Log Files / Hide Footprints
  - ✓ Re-secure The System
  - ✓ Creating Back Doors For Re-entry



# But...

- I'm Using A Firewall...
- I'm Using SSL ...
- I've Installed The Latest Patches...



# Application Attacks



CFUnited Express March 2007

# Application Attacks

- Relate To The Meaning Of Application Messages:
  - ✓ Interpretation Of The HTTP Requests
  - ✓ Handling Of SQL Queries
  - ✓ Interpretation Of Application Specific Messages



# Application Vs. Infrastructure Attacks

- Harder To Identify Or Replicate
  - ✓ Requires Understanding Of Both Technology and Application Domain
  - ✓ Vulnerabilities Differ Between Applications
- Easier To Exploit...
  - ✓ Coding Is Simple
  - ✓ GUI Assisted



# Application Vs. Infrastructure

- Application Attacks Are Often More Dangerous
  - ✓ Involve Organization's Core Operation
  - ✓ (Infrastructure Attacks Usually Target The Servers Themselves Only)
- And Harder To Repair...
  - ✓ May Require Code and Design Changes
  - ✓ Most Security Staff Has IT Background Rather Than Development Background



# Application Vulnerabilities

- Applications Assume Certain Client Behaviors
- Developers Anticipate Only “Real” User Will Input Data
- All Input Can Be Modified By The User
- Hackers Use A Variety Of Tools To Modify Data (Paros, ITR, Netcat...)



# Vulnerabilities - Information Gathering



CFUnited Express March 2007

# Information Gathering

- The Attacker Should Identify:
  - ✓ Infrastructure (OS & Web Server)
  - ✓ Dynamic Content Technology
  - ✓ Application/DB Servers Used
- Two Main Sources Of Information:
  - ✓ Normal Responses Analysis
  - ✓ Detailed Error Messages



# Response Analysis

- Responses May Contain Valuable Information Left Behind By Programmers Or System Administrators
- Information Can Be Found In:
  - ✓ Response Headers
  - ✓ Programmers Comments
  - ✓ Commented-out HTML/Javascript
  - ✓ Hidden Fields & Values
  - ✓ Client-side Scripts



# Information Gathering - Mitigation

- Do Not Include Any Redundant Code
  - ✓ Change All Comments To CFML
  - ✓ **Remove** Old Files
  - ✓ **Remove** Documentation
- Apply Global and Application Error Handler
  - ✓ <Cferror>
  - ✓ Onerror()

## Log, Alert and Review



# Source Code Disclosure

- Techniques For Source Disclosure:
  - ✓ Web Server Vulnerabilities
    - Most Are Known and Fixed
  - ✓ Application Vulnerabilities
    - Directory Traversal
    - File Downloads
    - Includes
    - Detailed Error Messages
  - ✓ Open Source Code



# Source Code Disclosure - Mitigation

- Apply Latest Patches To Webserver/CF
- Turn Off Debugging
- Remove All CFDOCS and Sample Apps
- Disable RDS
- Prevent Direct Execution Of Included Files and Modules By Placing An Application.cfm With <CFABORT> in Subdirectory



# Vulnerabilities - Application Top 10



CFUnited Express March 2007

# OWASP Top 10 List

- Unvalidated Input
- Broken Access Control
- Broken Account and Session Management
- Cross-site Scripting(xss) Flaws
- Buffer Overflows
- Command Injection Flaws
- Error Handling Problems
- Insecure Storage
- Application DoS

Insecure Configuration



CFUnited Express March 2007

# 1. Unvalidated Input

- The Most Simple Form Of Application Attack
- Targets The Business Logic Of The Application
- Does Not Require Any Special Tools
- Can Be Done On Both Get and Post Variables



# 1. Unvalidated Input

- Forms

- ✓ Hidden Fields
- ✓ User Selection
- ✓ Type and Range Validation

- URL

- ✓ Master/Detail Pages
- ✓ Attribute Parameters



# 1. Unvalidated Input

- Reduce Dependency On Hidden Fields By Using The Session Scope
- Do Not Rely On Client Side Validation Alone
- Check Validity Of User Selection and Input Type/Range



## 2. Broken Access Control

- Forceful Browsing
  - ✓ Static File Links
  - ✓ Hidden Files (Security By Obscurity)
  - ✓ File Name Predictions
    - Known System Files
    - .Log / .Old Files
- Path Traversal
- Client Side Caching
- File Permissions



## 2. Broken Access Control Mitigation

- Use Hash() To Perform Checksum Of URL
- Check Data Access Permissions On Every Request
- Store Files Outside Of Webroot and Use <CFCONTENT> To Serve Files To The User



# 3. Broken Account and Session Management

- Account Authentication Bypassing
  - ✓ Login Tampering
  - ✓ Brute Force
  
- Session Hijacking
  - ✓ Brute Force
  - ✓ ID Predicting
  - ✓ Sniffing and Eavesdropping
  - ✓ Using HTTP\_REFERER When SessionID Is Passed On URL



## 3. Broken Account and Session Management - Mitigation

- Enforce At Least 8 Characters Password
- Require Numbers and Special Characters
- Do Not Sent Permanent Passwords Via Email
- Do Not Disclose Reason For Login Failure
- Expire Passwords
  
- Log, Alert and Restrict Access After Failed Login Attempts



## 3. Broken Account and Session Management - Mitigation

- Require Re-authentication On Email Change
- Consider Using SSL To Encrypt Transmissions
- Disable Browser Caching
- Use UUID For CFTOKEN
- Use J2EE Sessions
- Control Session Timeout



# 3. Broken Account and Session Management - Mitigation

- Check CGI Variables
  - ✓ **CGI. HTTP\_REFERER**
  - ✓ **CGI. CF\_TEMPLATE\_PATH**
  - ✓ Note: They Can Be Spoofed!
- Check Cookies



# 4. XSS

- Stored

- ✓ Script Is Stored in Trusted Source

- Forums
- User Comments
- Contact Forms
- Online Web Mail System

- Reflected

- ✓ Script Reflected Off The Web Server In

- Error Messages
- Search Results
- ...



## 4. XSS - Mitigation

- Built in CF Protection
  - ✓ Coldfusion Admin Setting
  - ✓ `this.scriptprotect` in `Application.cfc`
- `HtmlTrans()`
  - ✓ <http://www.cflib.org/udf.cfm?id=945>
- `CF_XSSBLOCK`
- Log, Alert and Review Violations



## 5. Buffer Overflow

- Application Fails To Allocate Sufficient Memory For Its Input
- May Allow The Attacker To Achieve:
  - ✓ Denial Of Service
  - ✓ Remote Command Execution
  - ✓ Data Alteration/ Leakage
- May Appear in Sub Components Executed By Non Vulnerable Code



## 6. Command Injection Flaws

- SQL Injection
  - ✓ Alter The Syntax Of The SQL Statements Sent By The Application To The Server
- Not Just SQL



## 6. Command Injection Flaws - Mitigation

- <CFQUERYPARAM>
- Consider Stored Procedures
- Limit DB Permissions On CF Admin and in Database
- Disable XP\_cmdshell and Equivalents
- Consider Server Sandboxing



# 7. Error Handling

- Error Messages Might Disclose Sensitive Information
  - ✓ Directory Structure
  - ✓ Code Snippets
  - ✓ Query Structure



# 7. Error Handling - Mitigation

- Disable Debugging on Production
- Define Site Wide Error and 404 Handler
- `<CFERROR>` / `OnError()`



# 8. Insecure Storage

- Storing Sensitive Information Using Inadequate Encryption Schemas
  - ✓ Failure to encrypt critical data
  - ✓ Insecure storage of keys, certificates, and passwords
  - ✓ Improper storage of secrets in memory
  - ✓ Poor sources of randomness
  - ✓ Poor choice of algorithm
  - ✓ Attempting to invent a new encryption algorithm
  - ✓ Failure to include support for encryption key changes and other required maintenance procedures



# 8. Insecure Storage - Mitigation

- Encrypt Sensitive Data

- ✓ Encrypt()/Decrypt() – Two Way

- Uses Symmetric Key

- Cfm7

- Additional Algorithms (AES,BLOWFISH,DES...)

- Generatesecretkey

- ✓ Hash() – One Way

- Impossible To Revert

- Does Not Require Key

- Best For Passwords



# 9. Application DoS

Rendering A Service Offered By A Workstation Or Server Unavailable To Others

- Reasons:
  - ✓ To Get A System Reboot
  - ✓ Hacker Covering His/Her Tracks
  - ✓ Malicious Intent
  
- How It's Done:
  - ✓ Ping Of Death - ICMP Techniques
  - ✓ Syn (Network) Vulnerabilities
  - ✓ Data DoS



# 9. Application DoS

- Update With Latest Patches
  - ✓ ColdFusion
    - [http://www.adobe.com/support/coldfusion/downloads\\_updates.html](http://www.adobe.com/support/coldfusion/downloads_updates.html)
  - ✓ Web Server
  - ✓ OS
- Optimize Use Of Session Resources
  - ✓ Use Caching
  - ✓ Avoid Large Data Sets in Session



# 10. Insecure Configuration

- Many Developers Do Not Configure Their Server Beyond The Initial Install
  - ✓ Missing Patches
  - ✓ Sample Files
  - ✓ Default Accounts



# 10. Insecure Configuration

- Configure CF
  - ✓ Secure Admin Directory With NT Authentication Or Completely Remove
  - ✓ Do Not Deploy Docs, Sample Apps and RDS To Production
  - ✓ Do Not Store DB Password in code
  - ✓ Disable Unused Services



# Additional Information



CFUnited Express March 2007

# Social Engineering

- Simply Asking For :
  - ✓ Information
  - ✓ Passwords
  - ✓ Assistance
- Requires No Technical Skills



# Security and Project Management

- Integrate Security Into Your Process
  - ✓ Hack Test During/After Development
  - ✓ Create Anti-requirements
  - ✓ Review Code Regularly



# Resources

- Open Web Application Security Project (OWASP)
  - ✓ [Http://Www.Owasp.Org](http://www.owasp.org)
- Cgi Security,
  - ✓ [Http://Www.Cgisecurity.Net](http://www.cgisecurity.net)
- Web Application Security Mailing List,
  - ✓ [Http://Online.Securityfocus.Com/Archive/107](http://online.securityfocus.com/archive/107)
- Hacktics
  - ✓ [Http://Www.Hacktics.Com/Presentations.Html](http://www.hacktics.com/presentations.html)
- MIT Publications
  - ✓ [Http://Pdos.Lcs.Mit.Edu/Cookies/Pubs.Html](http://pdos.lcs.mit.edu/cookies/pubs.html)  
“Dos and Don'ts Of Client Authentication On The Web”



# Questions

Shlomy Gantz

[Shlomy@bluebrick.Com](mailto:Shlomy@bluebrick.Com)

[Http://Www.Shlomygantz.Com/Blog](http://Www.Shlomygantz.Com/Blog)



CFUnited Express March 2007