Business Logic Vulnerabilities

Trey Ford
Director of Solution Architecture
WhiteHat Security
trey.ford@whitehatsec.com

OWASP
July 2008

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

Target #1: Layer 7

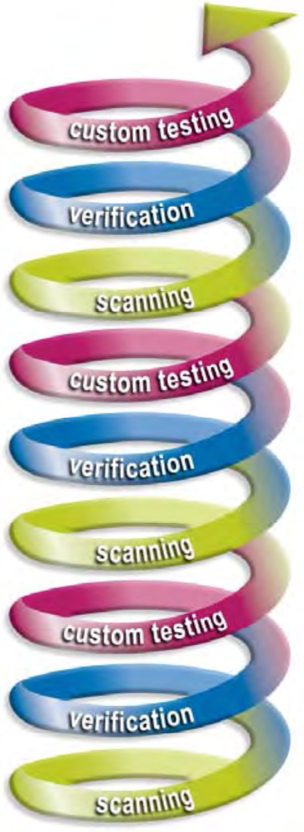
- 168+ million websites
- Many are mission-critical and gateways to highly sensitive customer and corporate information
- These websites are accessible by over 1 billion people

Oklahoma DOC
RIAA
orkut
Sprint ahead
BEST BUY
OBAMA
TIFFANY & CO.
myspace.com a place for friends
verizon
PayPal
NOKIA Connecting People
Georgia Tech
Macworld Conference & Expo
DUKE UNIVERSITY School of Medicine
facebook
Passport Network
YAHOO!
Redmond
comcast.net
Google
STANFORD UNIVERSITY
CNBC
Mizzou University of Missouri-Columbia
COMPUTER GEEKS.COM
Scottrade
T-Mobile
travelocity
THE UNIVERSITY OF CHICAGO
USDA
SUNTRUST
PAPA JOHNS Better Ingredients. Better Pizza.
ebay
USC
UNIVERSITY of VIRGINIA
Life is good
indiatimes
University of Idaho
VICTORIA'S SECRET
E*TRADE FINANCIAL

hacked

WhiteHat Sentinel

- Unlimited Assessments – customer controlled and expert managed – the ability to scan websites no matter how big or how often they change
- Coverage – authenticated scans to identify technical vulnerabilities and custom testing to uncover business logical flaws
- Virtually Eliminate False Positives – Operations Team verifies results and assigns the appropriate severity and threat rating
- Development and QA – WhiteHat Satellite Appliance allows us to service intranet accessible systems remotely
- Improvement & Refinement – real-world scans enable fast and efficient updates



9 out of 10 websites have vulnerabilities

allowing hackers unauthorized access

WASC 24 Classes of Vulnerabilities

Found by experts

Business Logic:

Authentication

- Brute Force
- Insufficient Authentication
- Weak Password Recovery Validation

Authorization

- Credential/Session Prediction
- Insufficient Authorization
- Insufficient Session Expiration
- Session Fixation

Logical Attacks

- Abuse of Functionality
- Denial of Service
- Insufficient Anti-automation
- Insufficient Process Validation

Found by scanners

Technical:

Command Execution

- Buffer Overflow
- Format String Attack
- LDAP Injection
- OS Commanding
- SQL Injection
- SSI Injection
- XPath Injection

Information Disclosure

- Directory Indexing
- Information Leakage
- Path Traversal
- Predictable Resource Location

Client-Side

- Content Spoofing
- Cross-site Scripting

Business Logic Flaws vs. QA

- Examples of Web-enabled business logic flaws: Session handling, credit card transactions, password recovery, etc.
- These vulnerabilities are routinely overlooked during QA because the process is intended to test what a piece of code is supposed to do and not what it can be made to do.
- The other problem(s) with business logic flaws is:
  - Scanners can’t identify them
  - IDS can’t detect them
  - Web application firewalls can’t defend them
  - Plus, the more sophisticated and Web 2.0 feature rich a website, the more prone it is to have flaws in business logic due to the complexities involved

Winning an Online Auction

Abuse of Functionality

Situation

- An on-line auction website prevents attackers from guessing the passwords of users by temporarily locking accounts that receive too many failed attempts (5 tries) in a given amount of time.
    - Once an account is locked, the attacker (or the user) must wait for a timeout to expire (1 hr) before attempting to login again. Account locking is one of several techniques used to slow down brute force attacks.
- Logged-in users are able to browse items and view bids.
    - To place a bid, a user is asked for their password to verify their intent, (prevents unintended bids/stops Cross-Site Request Forgery attacks).
- The bidding process is tied into the login security system to deny password guessing.

Business Logic Flaw

- If a malicious user wanted to place competing bidders at a disadvantage and improve their odds of winning an auction…
  1) Bid early and bid low on an item
  2) When/if any competitive bids are placed, bid slightly higher, then…
  3) Execute a sustained login brute force attack against that user's account.
- The result: Competitive bidders would be unable to bid on the item because their account would be purposely locked by the attacker, since the bidding system is tied to the login security system.

Solutions

- **Do not display user names on the website.**
  - This not only increases user privacy, but also prevents an attacker from knowing which bidder they need/want to lockout.
- **As an alternative to an account lockout, a CAPTCHA system may be employed** if an account has received too many failed login attempts.
  - This method has the benefit of preventing brute force attacks, without the potential side effect of locking out legitimate users who are making bids.
- **Allow sellers to specify a minimum bid price before they must sell the item**.
  - Prevents an attacker from getting an unreasonabley priced item.

Making Millions by Trading on Semi-public Information

Predictable Resource Location and Insufficient Authorization

Situation

- Business Wire provides a service where registered website users receive a stream of up-to-date press releases. Press releases are funneled to Business Wire by various organizations, which are sometimes embargoed temporarily because the information may affect the value of a stock.
  - Press release files are uploaded to the Web server (Business Wire), but not linked, until the embargo is lifted. At such time, the press release Web pages are linked into the main website and users are notified with URLs similar to the following:
  - http://website/press_release/08/29/2007/00001.html
  - http://website/press_release/08/29/2007/00002.html
  - http://website/press_release/08/29/2007/00003.html
- Before granting read access to the press release Web page, the system ensures the user is properly logged-in.

Business Logic Flaw

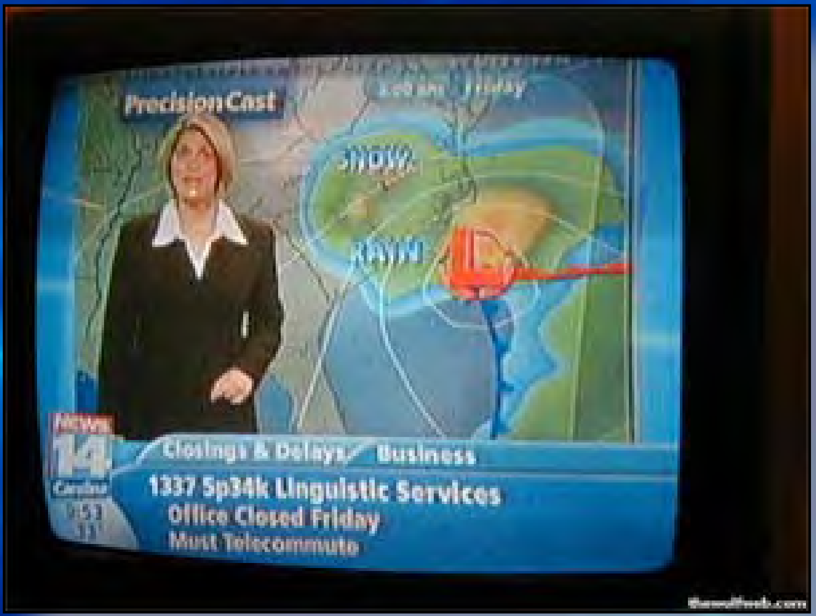
- An Estonian financial firm, Lohmus Haavel & Viisemann, discovered that the press release Web page URLs were named in a predictable fashion.
  - And, while links might not yet exist because the embargo was in place, it didn't mean a user couldn't guess at the filename and gain access to the file. This method worked because **the only security check Business Wire conducted was to ensure the user was properly logged-in, nothing more**.
- Result: According to the SEC, which began an investigation, Lohmus Haavel & Viisemann profited over $8 million by trading on the information they obtained.

SEC Vs. The Estonian Spidershttp://www.webpronews.com/topnews/2005/11/02/sec-vs-the-estonian-spiders

Solution

- The system should ensure that press releases are only served to authorized users after the embargo date has been passed.

“Interactive” T.V.



1337 Sp34k Linguistic Services
Office Closed Friday
Must Telecommute

Insufficient Process Validation

OWASP

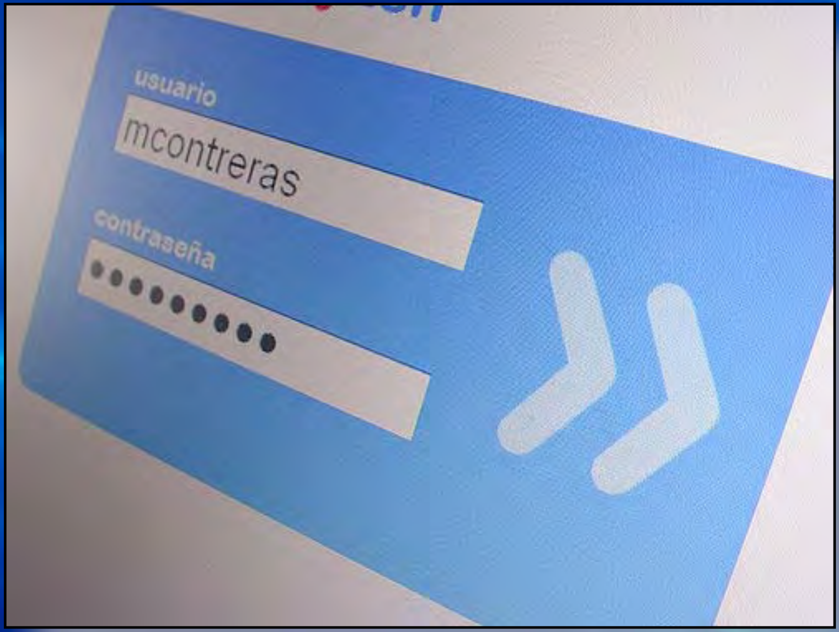
Situation

- The **website for Cable News 14 in North Carolina allowed registered users to submit weather related announcements for T.V.**
- The **submissions are posted to the onscreen crawl** during the newscast as a public service and periodically rotated. Think Web 2.0 for television.
- To prevent abuses **station personnel reviews** the submission's content before it's allowed to air.
- **Afterward, users are free to edit the content** to reflect any changes in status.
  - For example, if a business or a school reopened or is to remain closed for an extended period, local residents can stay informed by monitoring the crawl.

Solution

- The same process to sanity check posts prior to submission should have taken place upon editing.

Recovering Other People Passwords



Weak Password Recovery Validation

Situation

- The business owners of a website **plan to reduce support costs by supplementing expensive customer support representatives with a Web-based customer self-service tool**.
- One feature includes the **ability to recover forgotten passwords**.
  - If a user wants to reset their password, they enter their email address and answer a previously defined secret question. In this case it happens to be their favorite color.
  - When the user correctly answers the question, they're presented with an HTML form to enter a brand new password.

Business Logic Flaw

- Another secret question was introduced – one that would be harder to guess. **The date of birth (DOB)** is decided upon, which includes the month, day and year, because it provides a significantly larger amount of possible answers.
- When the **user correctly answer both secret questions** (color and DOB) **they would be allowed to reset their password**.

Business Logic Flaw

- While the **DOB is harder to guess, the data isn't exactly confidential** (besides the fact that it only has roughly 16,200 possible answers.
  - Attackers attempting to brute force the answer may easily do so at an average speed of 1 guess per second, taking only 4.5 hours
  - There is **no limit on the number of guesses an attacker may try** before the account is locked for a period of time or protected with a CAPTCHA.
- Business owners decide to add yet another secret question – the user's **city of birth (COB)**.
- Also added was an image-based CAPTCHA system to prevent brute force attacks.

Business Logic Flaw

- **The COB often doesn't scale internationally**.
  - For example in Mexico, home to 106 million people, 30% of the population is from one of five urban areas (Mexico City, Guadalajara, Monterey, Puebla and Toluca). Suddenly, what was a hard to guess secret question for a U.S. citizen has been greatly reduced to 1 in 5 for roughly 1/3 of Mexican users.
- The business owner decides to **utilize the user's email address deciding email sniffing is considered an acceptable risk**.
  - When a user requests a password reset, the back-end system sends them an email containing the following link for them to click on: http://website/password_reset?account=user@email.tld
  - When clicked, the user is presented with a password reset form.

Business Logic Flaw

- Result: The URL format is predictable. **Attackers can easily brute force email addresses to reset user account passwords**; that is, if they can't find valid addresses ahead of time.
- To improve the security of the system, the user's email address is removed and replaced with a session ID to track which account the request is tied to. To ensure uniqueness, the session ID uses a 12-digit number that increments each time a user requests a password reset. For example:
  - http://website/password_reset?id=000000001000
  - http://website/password_reset?id=000000001001
  - http://website/password_reset?id=000000001002

Business Logic Flaw

- To reset another account password, **a malicious user would first attempt to reset their own password a few times** in order to analyze the new URL format.
  - They would notice that the format uses a predictable incrementing number.
- Result: In one attack they could **decrement their session ID number** manually to see if they can beat any users to resetting their passwords.
- Result: Or, they could **initiate an account password reset for a user** and start incrementing the session ID in the URLs until they find the right number.

Solution

- Password recovery systems are especially difficult to secure against abuse.
  - The best way is to **keep them as simple as possible** and utilizing a user’s email address provides a well-accepted form of authentication.
  - **Make sure the session identifiers are not predictable** by an attacker.

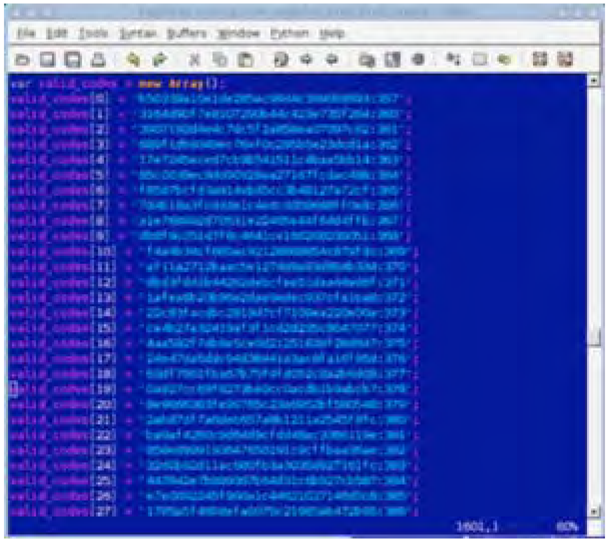
See Steve Jobs Up Close

AP / Paul Sakuma

Information Leakage

Situation

- During the MacWorld 2007 Expo, special Priority Codes could be used by VIPs to obtain free Platinum Passes with on-line registration.
  - Platinum Passes were a $1,695 and came with a chance to see Apple CEO Steve Jobs up close.
- Hidden in the source code of the sign-up Web page was a list of available PCs encrypted with a one-way algorithm (MD5) used to ease Web server load.
- Before users submitted their order, any submitted PCs would be MD5'ed using JavaScript and then compared client-side against the hidden list.
  - If the PC matched one on the list, the order would be sent to the server. If not, the user would receive an error message and the server would not be contacted.



```
var valid_codes = new Array();
valid_codes[0] = '[illegible]';
valid_codes[1] = '[illegible]';
valid_codes[2] = '[illegible]';
valid_codes[3] = '[illegible]';
valid_codes[4] = '[illegible]';
valid_codes[5] = '[illegible]';
valid_codes[6] = '[illegible]';
valid_codes[7] = '[illegible]';
valid_codes[8] = '[illegible]';
valid_codes[9] = '[illegible]';
valid_codes[10] = '[illegible]';
valid_codes[11] = '[illegible]';
valid_codes[12] = '[illegible]';
valid_codes[13] = '[illegible]';
valid_codes[14] = '[illegible]';
valid_codes[15] = '[illegible]';
valid_codes[16] = '[illegible]';
valid_codes[17] = '[illegible]';
valid_codes[18] = '[illegible]';
valid_codes[19] = '[illegible]';
valid_codes[20] = '[illegible]';
valid_codes[21] = '[illegible]';
valid_codes[22] = '[illegible]';
valid_codes[23] = '[illegible]';
valid_codes[24] = '[illegible]';
valid_codes[25] = '[illegible]';
valid_codes[26] = '[illegible]';
valid_codes[27] = '[illegible]';
```

Business Logic Flaw

- Several people noticed the hidden list of MD5 PCs in the Web page source code and also that the key space was small - so small in fact that they could be easily brute-forced.
- Result: Hackers quickly created programs for doing so; and, a few minutes later were cracking the PCs (usable during conference registration) to obtain free Platinum Passes.

Macworld crack offers VIP passes, hacker says
http://www.news.com/Macworld+crack+offers+VIP+passes,+hacker+says/2100-1002_3-6149994.html

Your Free MacWorld Expo Platinum Pass (valued at $1,695)http://grutztopia.jingojango.net/2007/01/your-free-macworld-expo-platinum-pass_11.html

Solution

- There is a strong desire to have the web browser perform data input validation to ease Web server load, and often this can be done safely.
- In this case the developer chose to place sensitive data on the client, even encrypted, in such a way that cryptanalysis could be performed. It would have been **better to let the server solely perform this process and preserve the security of the system**.

The House Almost Always Wins

Abuse of Functionality and Information Leakage

OWASP

Situation

- Blackjack is a card game where the player plays against the dealer and attempts to get closer than the dealer to 21 without going over (busting).
- When reduced to code, even **simple games of chance tend to have complex trees of logic flows**, which may take different amounts of time to execute.
- These logic flaws also dictate when cards are dealt, opportunities to bet, and when the dealer must hit or stand.

Business Logic Flaw

- Blackjack rules say the dealer should o**ffer a player an opportunity to buy insurance if the dealer's up card is an Ace**, in case the hole card is a 10-value.
- Result: In one published case, a Paradise Poker player noticed that when the dealer was showing an Ace and DID have a pocket 10-value card, **there was a noticeable timing delay before the game offered insurance**. Sort of a digital version of a poker player's "tell". This tell provided the player a slight edge over the house, providing them the advantage to make money.

Online Games Are Written By Humanshttp://haacked.com/archive/2005/08/29/9748.aspx

Solutions

- **Pad certain area of decision logic with extra time** to smooth out timing nuances that can be fingerprinted.
- Or, **optimize the code in areas to allow it to run with the same execution time** as other areas of the system.

Day trading contest for $1,000,000 (US)

Insufficient Process Validation

Situation

- CNBC’s Million Dollar Portfolio Challenge provided amateur traders a chance to match their skills against the portfolios of the Internet’s best.
  - 375,000 contestants competed in ten one-week challenges for a $10,000 prize and a chance to go to the finals and compete for the million dollar prize.
  - To win, all they had to do was make the most “funny” money.
- Placing stock trades is essentially a two-step process:
  - Step 1. Select the stocks and number of shares. The system calculates the order using the current share price and waits for user confirmation before executing the trade.
  - Step 2. The user can either drop out of the transaction or confirm the order, which then executes the mock stock transaction to update their portfolio.

Business Logic Flaw

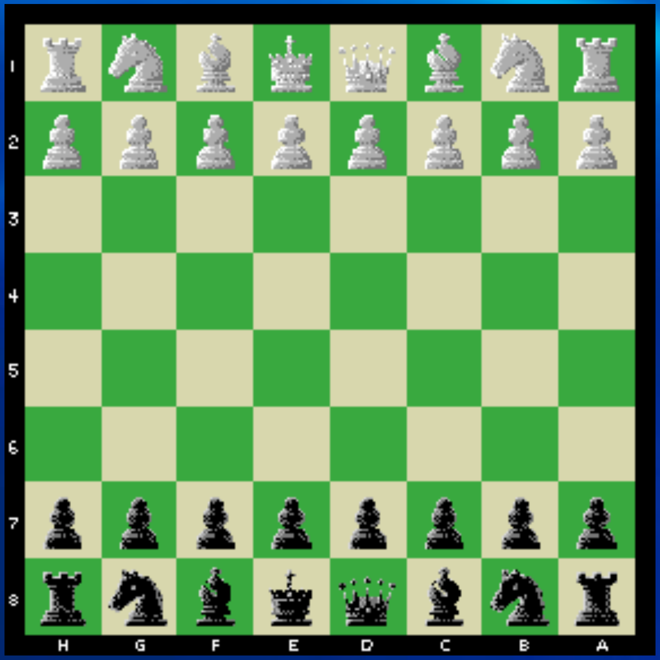
- To make impossibly accurate picks, a malicious trader would select several stocks to buy (but NOT execute the order in step 2) with companies scheduled to post earnings after trading closes that day.
- After setting up the order, they’d leave their browser window open until after the closing bell. If the stock price rose by a significant percentage during after hours trading, the trader would only then execute the transaction.
- Result: Since their session contained the original stock price and did not recalculate using the current share price, the trader would be guaranteed huge portfolio gains and be well on their way to winning the million.

$1,000,000 CNBC stock trading contest hackedhttp://jeremiahgrossman.blogspot.com/2007/06/1000000-cnbc-stock-trading-contest.html
CNBC's Easy Money
http://www.businessweek.com/bwdaily/dnflash/content/jun2007/db20070607_007145.htm
Finalists allege hacking in $1 million stock contesthttp://www.securityfocus.com/brief/521

Solutions

- When executing the trade, the system should always calculate based upon the true current share price.
- The session for a pending trade should have an expiration time set; 20 minutes would be sufficient.
- Reject any incoming trades when the market is closed.

On top of the Yahoo Games Chess Ladder



Abuse of Functionality

OWASP

Situation

- Yahoo Games has an online chess ladder. A ladder system essentially ranks all the players from top to bottom, and you increase or decrease rank on the ladder by winning or losing (or not playing).

Business Logic Flaw

- There are literally thousands of people (or more) with an amazing about of free time to do the most mundane tasks for the most inane rewards. “Cheating” players would code purpose built programs to bot 100’s of chess games 24x7 simultaneously. They’d sit up late into the evening because every so often the ladder ranks would be reset, and when they did, they’d snatch the top spots. And once they owned a block of the top spots they’d only play within their controlled accounts to rise slowly in ranks. The way the ladder logic worked, “legit” ranked players must play against other equally or higher rank players, and since cheaters wouldn’t play against them, legit players would drop in rank.

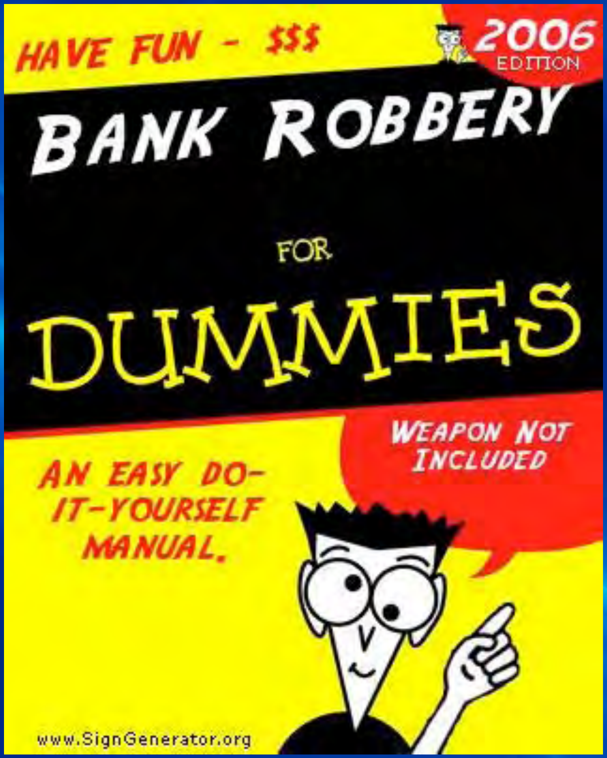
A Nation of Cheaters?
http://www.csoonline.com/read/110106/edit_cheaters.html

Business Logic Flaws and Yahoo Games
http://jeremiahgrossman.blogspot.com/2006/12/business-logic-flaws.html

Solution

- Implement a CAPTCHA system to prevent automated bots from playing the gamePlace limits on the number of games an account can play during a 24 hour period, which should still be humanly feasible.

Robbing a Bank Blindfolded



Information Leakage and Insufficient Authorization

Situation

- An ASP provides hosting for banks, credit unions, and other financial services companies. ASPs are attractive targets because **instead of focusing one back at a time**, an attack could **compromise dozens/hundreds/thousands at a time** with the same vulnerability.
- The banking application had three important URL parameters: **client_id, bank_id, and acct_id**. To the ASP, each of their clients has an unique ID, each potentially with several different banking websites, and each bank having any number of customer bank accounts.

  - http://website/app.cgi?client_id=10&bank_id=100&acct_id=1000

Business Logic Flaw

- By changing the acct_id to an arbitrary yet valid account #, the system would error and say “Account #X belongs to Bank #Y”
- If you changed the bank_id to #Y, the system would error again and say “Bank #Y belong to Client #Z”
- If you changed the client_id to #Z, you could drop into anyone else’s bank account, on any bank, on any client.
  - *http://website/app.cgi?client_id=10&bank_id=100&acct_id=1000*

Solution

- The system should not be providing that level of useful error message detail to the screen. This was likely a remnant of development debugging. Best to use nondescript error messages.
- The system should have ensured the user had the appropriate permissions to access to the client bank account.

Business Logic Flaw

- **Reverse Wire Transfer**

Normal: $10,000 from Account A to Account B

- A = A - ($10,000)
- B = B + ($10,000)

Negative: -$10,000 from Account A to Account B

- A = A - (-$10,000)
- B = B + (-$10,000)

More Business Logic Flaws...

SEARCH INSIDE!™
The Web Applicati
Hacker's
Handbook
Discovering and Exploiting
Security Flaws
Dafydd "PortSwigger" Stuttard Marcus Pinto

by Dafydd Stuttard and Marcus Pintohttp://www.amazon.com/Web-Application-Hackers-Handbook-Discovering/dp/0470170778

- Found in Chapter 11

Identifying Business Logic Flaws

- Business logic flaws are pervasive and extremely diverse.
- It’s easy to see why even the best QA processes overlook these issues.
  - They don’t check for what the system can be manipulated to do. And, vulnerability scanners, intrusion detection systems, and Web application firewalls would have an equally hard time.
- Data value requires knowledge of context and does not know what the website is supposed to do, or the path through the logic, so it can’t tell if it did something wrong.
- To find business logic issues, the **pairing of experienced security experts with automated scanning is a best practice** for achieving complete website vulnerability coverage.

Best Practices

- Asset Tracking – Find your websites, assign a responsible party, and rate their importance to the business. Because you can't secure what you don't know you own.
- Measure Security – Perform rigorous and on-going vulnerability assessments, preferably every week. Because you can't secure what you can't measure.
- Development Frameworks – Provide programmers with software development tools enabling them to write code rapidly that also happens to be secure. Because, you can't mandate secure code, only help it.
- Defense-in-Depth – Throw up as many roadblocks to attackers as possible. This includes custom error messages, Web application firewalls, security with obscurity, and so on. Because 8 in 10 websites are already insecure, no need to make it any easier.

Thank You!

For more information, please visit www.whitehatsec.com/

Trey Ford, Director of Solutions Architecture
blog: http://treyford.wordpress.com
email: trey@whitehatsec.com

Additional References

What scanners can and can't find. Who cares and why does it matter?
http://jeremiahgrossman.blogspot.com/2006/11/what-scanners-can-and-cant-find-who.html

Challenges of Automated Web Application Scanning
http://www.whitehatsec.com/home/resources/presentations/files/challengesofscanning.pdf

5 challenges of web application scanning
http://eremiahgrossman.blogspot.com/2006/07/5-challenges-of-web-application.html