

INTO THE new
BREW 2007 CONFERENCE

BREW[®] Security: A Carriers' Perspective

Philip Varughese, Manager
Verizon Wireless

Vijay Akasapu, Director
Mandiant





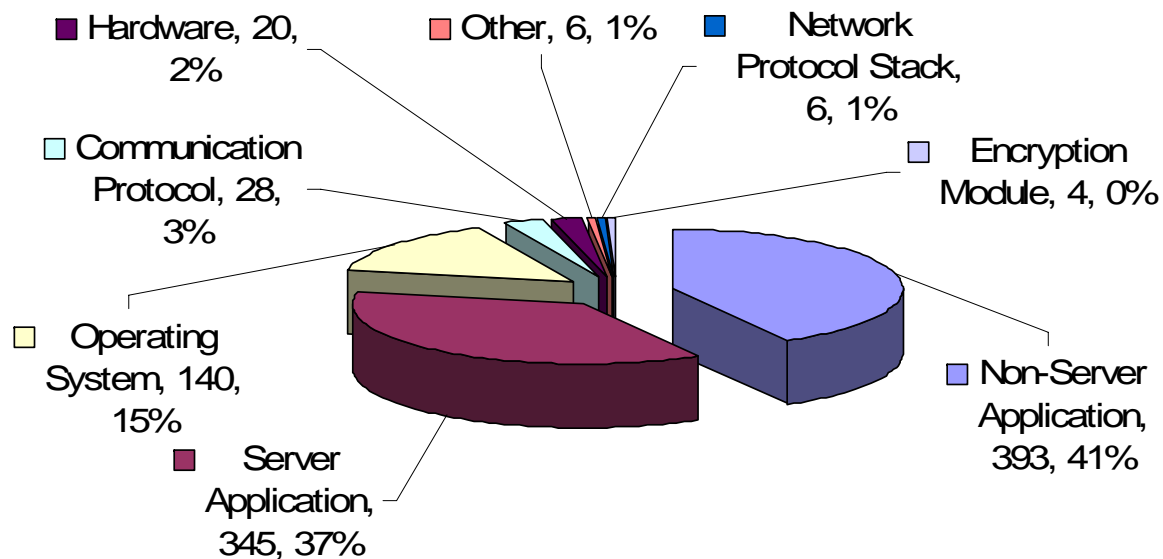
Agenda

- Emerging threats: DRM issues, location disclosure, malware (viruses, worms, etc), Spam, theft of content
- Impact: Impact on businesses, end users, and carriers
- How does the BREW platform mitigate against threats?
- Security Assurance

Vulnerability Trends

- 2001, 0% reported web application vulnerabilities
- Weak network services constituted the majority of targets

Vulnerability Targets in 2004



Targeted attacks

- Rise of application-directed attacks on the Internet
 - 66% web application targeted (according to Symantec)

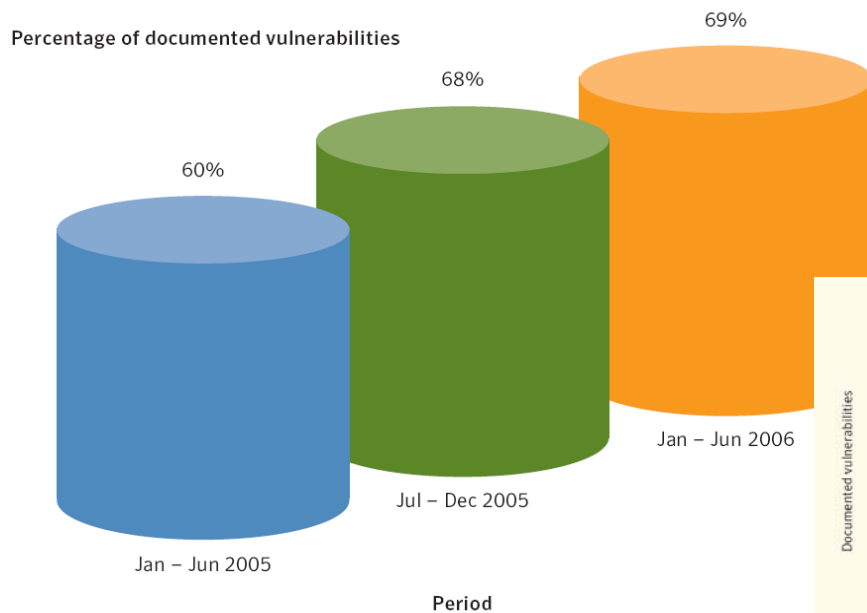


Figure 17. Web application vulnerabilities
Source: Symantec Corporation

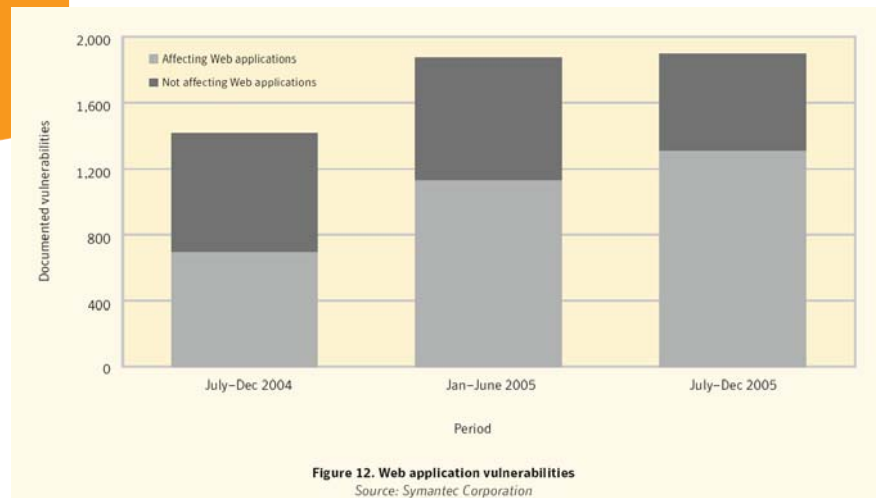
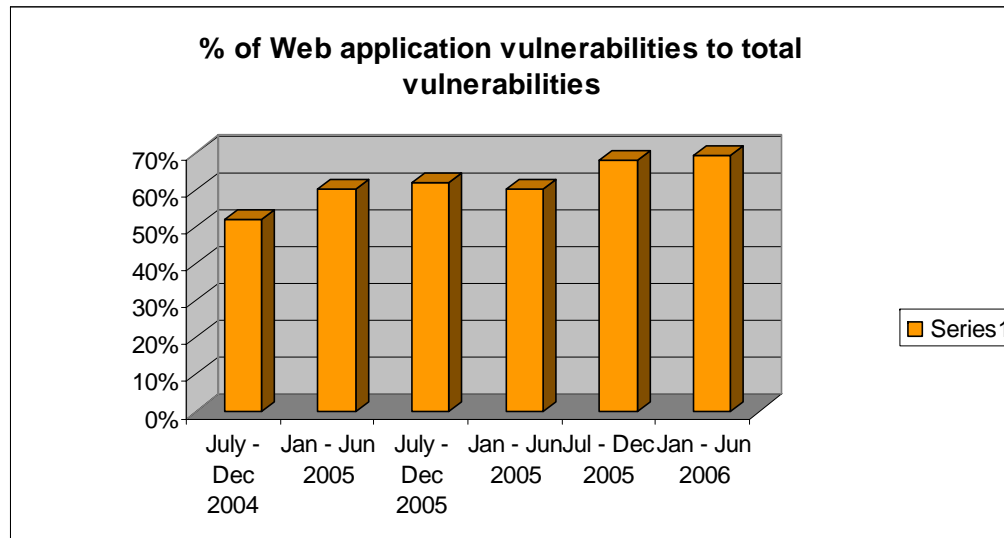


Figure 12. Web application vulnerabilities
Source: Symantec Corporation

Continuing rise in web application vulnerabilities

- Continuing increase in targeted application vulnerabilities





SANS reports

"Every week hundreds of vulnerabilities are being reported in ... web applications, and are being actively exploited. The number of attempted attacks every day for some of the large web hosting farms range from hundreds of thousands to even millions.

All web frameworks (PHP, .NET, J2EE, Ruby on Rails, ColdFusion, Perl, etc) and all types of web applications are at risk from web application security defects, ranging from insufficient validation through to application logic errors."
(emphasis theirs)

-- SANS Top-20 Internet Security Attack Targets
(2006
Annual Update) - <http://www.sans.org/top20/>



Application Attack Tools

- One reason that application attacks are increasing is due to the increased availability of application specific attack tools:
- Google Hacking
- Manual Tools
 - WebSleuth
 - Paros
 - WebScarab
 - SPIKE Proxy
 - ...
- Automated Tools
 - SSLDigger
 - Nikto
 - Wikto
 - SQLInjector
 - Absinthe
 - XSS-Shell
 - ...

Automated SQL Injection - Absinthe

The screenshot shows the Absinthe application window with the following configuration:

- Host Information: DB Schema, Download Records
- Exploit Type: Select the type of injection: Blind Injection Error Based
- Select The Target Database: PostgreSQL
- Connection: Target URL: http://internal.0x90.org/~nummish/sql.php; Connection Method: Get Post Use SSL
- Authentication: Use Authentication; Basic, Digest, NTLM options; Name, Password, Domain fields.
- Form Parameters: Name, Default Value, Injectable Parameter, Treat Value as String, Add Parameter, Add Cookie buttons.
- Parameters table:

Name	Value	Injectable
id	2	True
- Buttons: Edit, Remove, Initialize Injection

The screenshot shows the Absinthe application window displaying the output of a SQL injection attack:

- Output: Filename: /home/nummish/code/Absinthe_1_3/bir/datapull.txt
- Available Fields table:

Table	Field	Field I
users	id	0
- Selected Fields table:

Table	Field	Field I
users	password	2
users	username	1
- Buttons: Add, Remove, Download Fields to XML



Mobile Threats

- Network-based threats (BlueTooth, etc)
- Malware: Viruses, Trojans, Worms
- Theft of phones
- Next-generation Phones (VoIP, WiFi)
- SMS Spam (July 2006, introduced as new vector)
- Insecure Applications

Parallel to Internet threats?

- Liberty
 - Trojan deletes applications on PalmSource
 - Requires manual install
- Cabir
 - First instance of mobile malware
 - Spreads using bluetooth
- Trojans, Viruses, Worms spreading through network weaknesses or install themselves via "Sync" features
- Are application-targeted threats next?





Impact of abuse of insecure applications

- Denial of Service
- Loss of revenue
- Privacy
- Brand Name/Reputation
- Contractual obligations
- Cost of fixing
- CIA
- Reliability



Threats faced by mobile applications

- Spoofing
- Tampering
- Repudiation
- Information Disclosure
- Denial of Service
- Elevation of Privilege



Spoofting

- Ability to log in as a legitimate user by stealing session data
- Ability to log in as a legitimate user by replaying authentication or session data
- Ability to log in as a legitimate user by guessing logon credentials
- Ability to view information associated with a different user once authenticated
- Server impersonation
- Application directed SMS message to impersonate a server



Tampering

- Application directed SMS message that changes settings on a client device
- Custom client and tampers with requests sent to the server
- Modification of files on the device file system to enable a Man-in-the-Middle or other attack.
- Launch of BREW applications with malicious arguments



Repudiation

- Ability to refute that they accessed a service or information
- Insufficient audit information allows an attacker to misuse a system without being held accountable



Information Disclosure

- Obtain sensitive information without authenticating to the server
- Obtain information that is related to another user, or outside the scope of their assigned role
- Strip DRM controls from paid content stored on a device
- Obtain paid content without incurring a billing event
- Capture unencrypted network traffic that contains sensitive information
- Obtain sensitive information by examining and/or disassembling files stored on a device
- Obtain sensitive information by capturing BREW debugging output
- Obtain sensitive information by causing the application to fail and return detailed error information



Denial of Service

- Crash application running on device by sending malformed traffic
- Crash application running on device by sending application directed SMS
- Crash server application by sending malformed data



Elevation of Privilege

- Ability to gain access to information or performs actions associated with a more privileged role



How does BREW mitigate against these threats?

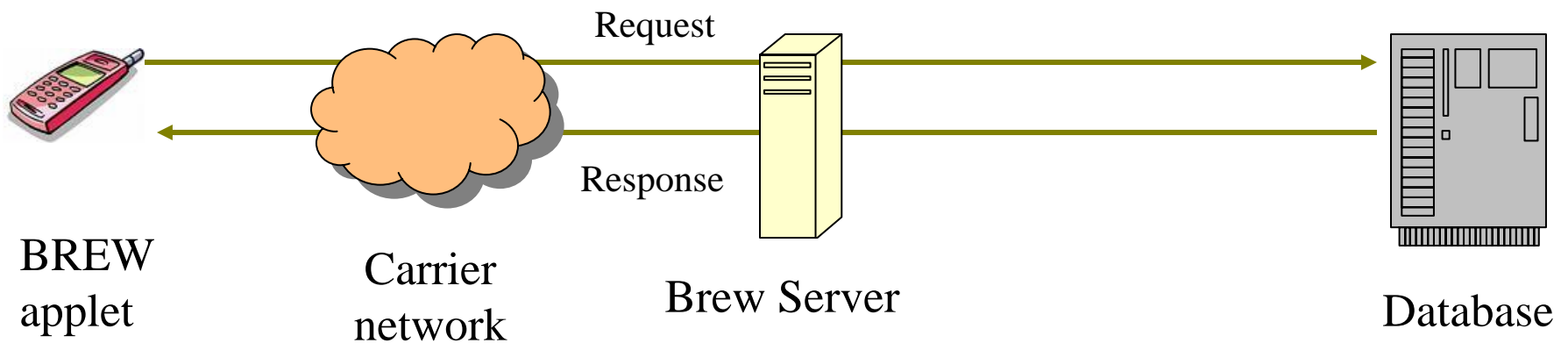
- Digital Signatures
 - Protects application integrity
 - Provides Attribution
 - Verification of privileges
 - > Access the network
 - > Read from shared directory
 - > Access address book, etc
- Testing by NSTL
- File System permissions
- Support for Encryption and Hashing
- Digital Rights Management



Encryption and Hashing support

- Symmetric Ciphers
 - ICipher class which implements RC4, SEED, AES
- Random Number Generator
 - GETRAND (not cryptographically strong)
 - RANDOM
- Hashing Functions
 - IHash
 - IHashCtx
- Public Key Ciphers
 - IRSA class
- Digital Certificates
 - IX509Chain
- SSL
 - ISSL
- HTTPS
 - IWeb

Does BREW security suffice



- Insecure applications
 - Insecure communication protocols
 - Insecure BREW server
 - Lack of SMS authentication



Triage

- Operators:
 - Need to collaborate on best practices
 - Require broad industry development effort
 - International scope of regulatory efforts
- Developers:
 - Need to practice best practices



Security Assurance

- Assurance that product and service is acceptably secure
- Identifying, characterizing and managing risk to an acceptable level
- Consistent means of showing compliance
- Testing and validating security controls



Why

- Mobile vulnerabilities and malicious attacks are increasing
- Security is not integrated into development cycle
- Strategically addressing security



Applicability

- Commercial Applications & Products
 - Threat Modeling
 - Application Penetration Test
 - Source Code Review
 - Ongoing Assessment
- ASPs
 - Security of ASP environment
 - Platform Hardening
 - Ongoing Assessment



Security Assurance Framework

- Security of ASP Environment
 - ISO17799
 - Compliance with regulations & standards
 - Review security policies and procedures
 - Review overarching information security processes
 - Physical security
- Threat Modeling
 - Identification
 - Classification
 - Quantify the risk
 - Validating compensating security controls
 - Identify test cases for penetration testing



Security Assurance Framework (Contd.)

- Source Code Analysis
 - Identify design and logic flaws
 - Automated review
 - Manual review
- Penetration Testing
 - Adopt a “hackers” view
 - Examine threat identified during Threat Modeling
 - Attempt to exploit potential vulnerabilities
 - Identify strengths and weakness
- API & Protocol Analysis
- Architecture & Design Review
- Platform Hardening
- Ongoing Assessment



Security Assurance Benefits

- Services to include security as a quality
- Raises the bar on security
- Secure products and services
- Lower security control cost
- Implement appropriate security measures
- Efficient and consistent means of evaluating and testing
- Risk based approach & solutions
- Reduced cost of fixing
- Prior to launch remediation



Questions?