Cloudy with a chance of hack

Lars Ewe
CTO / VP of Eng.
Cenzic
lars@cenzic.com

OWASP
November, 2010
Agenda

- Weather Trends & 6-Day Forecast
- Clouds Everywhere!
- Why So Little Sunshine?
- How To Best Dress For Bad Weather
- Q & A
Web Security Trends

75% of cyber attacks & Internet security violations are generated through Internet applications
Source: Gartner Group

87% of Websites are vulnerable to attack
Source: SearchSecurity – January 2009

75% of enterprises experienced some form of cyber attack in 2009
Source: Symantec Internet Security Report – April 2010

90% of Websites are vulnerable to attack

$6.6 Million is the average cost of a data breach
Source: Ponemon Institute – January 2009
Web Application Vulnerabilities (as a percentage of total)

Q1-Q2 2009: 78%
Q3-Q4 2009: 82%
Q1-Q2 2010: 66%

Source: Cenzic Q1-Q2, 2010 Application Trends Report
Web Vulnerabilities by Class (commercial applications)

Source: Cenzic Q1-Q2, 2010 Application Trends Report
Web App Other Category

- 64% Permissions-Privileges-Access Control
- 20% Buffer Errors
- 7% Input Validation
- 5% Code Injection
- 2% Command Injection
- 2% Information Leak-Data Disclosure

Source: Cenzic Q1-Q2, 2010 Application Trends Report
Web Vulnerabilities by Class (proprietary applications)

Source: Cenzic Q1-Q2, 2010 Application Trends Report
Percentage of Vulnerabilities

- Information Leaks and Exposures: 90%
- Cross Site Scripting: 80%
- Unauthorized Directory Access: 68%
- Authorization and Authentication: 68%
- Cross Site Request Forgery: 20%
- Insecure Resource Location: 14%
- Session Management: 13%
- SQL Injection: 8%
- Remote Code Execution: 3%

Source: Cenzic Q1-Q2, 2010 Application Trends Report
Information Leaks and Exposures

- Application Errors / Exceptions
- Forms that cached sensitive user information
- Passwords submitted without utilizing SSL
- Sensitive information passed as a URL parameter
- Password auto-complete attribute
- HTML / Javascript comments
- ...

Source: Cenzic Q1-Q2, 2010 Application Trends Report
Authorization and Authentication Flaws

- Brute force login
- Unauthorized resource access
- Privilege escalation
- ...

Source: Cenzic Q1-Q2, 2010 Application Trends Report
Session Management

- Weak session randomness
- Ineffective session termination
- Session Fixation
- ...

Source: Cenzic Q1-Q2, 2010 Application Trends Report
Cross Site Scripting

- Q1-Q2 2010: 68%
- Q3-Q4 2009: 81%
- Q1-Q2 2009: 73%

Source: Cenzic Q1-Q2, 2010 Application Trends Report
Cross Site Request Forgery

<table>
<thead>
<tr>
<th>Period</th>
<th>Percentage of Vulnerable Sites</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1-Q2 2010</td>
<td>20%</td>
</tr>
<tr>
<td>Q3-Q4 2009</td>
<td>14%</td>
</tr>
<tr>
<td>Q1-Q2 2009</td>
<td>14%</td>
</tr>
</tbody>
</table>

Source: Cenzic Q1-Q2, 2010 Application Trends Report
SQL Injection

<table>
<thead>
<tr>
<th>Period</th>
<th>Percentage of Vulnerable Sites</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1-Q2 2010</td>
<td>14%</td>
</tr>
<tr>
<td>Q3-Q4 2009</td>
<td>32%</td>
</tr>
<tr>
<td>Q1-Q2 2009</td>
<td>27%</td>
</tr>
</tbody>
</table>

Source: Cenzic Q1-Q2, 2010 Application Trends Report
Robert'); DROP TABLE Students;--
And The 6-Day Forecast?

Cloudy with a chance of hack
Cloud Security
Cloud Security – A Big Issue

Q: Rate the challenges/issues ascribed to the 'cloud'/on-demand model
(1=not significant, 5=very significant)

- Security: 74.6%
- Performance: 63.1%
- Availability: 63.1%
- Hard to integrate with in-house IT: 61.1%
- Not enough ability to customize: 55.8%
- Worried on-demand will cost more: 50.4%
- Bringing back in-house may be difficult: 50.0%
- Regulatory requirements prohibit cloud: 49.2%
- Not enough major suppliers yet: 44.3%

Source: IDC Enterprise Panel, August 2008  n=244
Cloud Security – A Big Issue

Security Defects in the technology itself
Unauthorized access to or leak of proprietary information
Unauthorized access to or leak of our customer’s information
Application or system performance
Business viability of provider: risk company will fail
Business continuity or disaster recovery readiness of provider
Features and general maturity of technology
Vendor Lock-in
Other

Source: Information Week Analytics (547 respondents)
Cloud And Security

- Exposure is similar to any Web apps – but on a potentially massive scale
- Security boundaries and attack surfaces are often only partially understood
- Proliferation of Mashups and ‘open’ APIs that favor ‘experience’ over security
- Does security ownership transfer to the cloud infrastructure / platform provider?
- What happens in case of a breach? Who’s responsible?
- Often organizations are still figuring out the “Functionality / Usability” aspects of their cloud strategy...

“Security is usually the last component added to any new technology, and cloud computing is no exception.” – Mark Nicolett, Gartner
Top 5 Myths of Web Application Security

1. We use SSL so that’ll protect my Web site
   ▸ SSL ≠ App Security

2. We have never been hacked
   ▸ How do you know?

3. We’re PCI compliant
   ▸ Heartland, Hannaford...

4. We test some of our Web applications once a year
   ▸ Any vulnerable site is your weakest link

5. Too expensive
   ▸ Many flexible options to get you jump started

Learn more: App Security MythBusters Videos
http://www.cenzic.com/resources/videos/mythbusters/
The Hacker World
Hackers: What Motivates Them?

- Hackers stole **$1.2 million in 30 minutes** from Sugarland Corporation & **$9M in a few hours** from RBS World Pay
- Hackers get paid ~ **$10,000 / week**


<table>
<thead>
<tr>
<th>Overall Rank</th>
<th>Item</th>
<th>Percentage 2009</th>
<th>Percentage 2008</th>
<th>Range of Prices</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Credit card information</td>
<td>19%</td>
<td>32%</td>
<td>$0.85–$30</td>
</tr>
<tr>
<td>2</td>
<td>Bank account credentials</td>
<td>19%</td>
<td>19%</td>
<td>$15–$850</td>
</tr>
<tr>
<td>3</td>
<td>Email accounts</td>
<td>7%</td>
<td>5%</td>
<td>$1–$20</td>
</tr>
<tr>
<td>4</td>
<td>Email addresses</td>
<td>7%</td>
<td>5%</td>
<td>$1.70/MB–$15/MB</td>
</tr>
<tr>
<td>5</td>
<td>Shell scripts</td>
<td>6%</td>
<td>3%</td>
<td>$2–$5</td>
</tr>
<tr>
<td>6</td>
<td>Full identities</td>
<td>5%</td>
<td>4%</td>
<td>$0.70–$20</td>
</tr>
<tr>
<td>7</td>
<td>Credit card dumps</td>
<td>5%</td>
<td>2%</td>
<td>$4–$150</td>
</tr>
<tr>
<td>8</td>
<td>Mailers</td>
<td>4%</td>
<td>3%</td>
<td>$4–$10</td>
</tr>
<tr>
<td>9</td>
<td>Cash-out services</td>
<td>4%</td>
<td>3%</td>
<td>$0–$600 plus 50%–60%</td>
</tr>
<tr>
<td>10</td>
<td>Website administration credentials</td>
<td>4%</td>
<td>3%</td>
<td>$2–$30</td>
</tr>
</tbody>
</table>
Why So Little Industry Progress?

- Functionality & Usability tend to almost always win over security
- Time-to-market is the name of the game
- Security continues to be an afterthought
- Very limited security related education
- Experts are still hard to find (compared to other disciplines)
- Many organizations still struggle to find a scalable and persistent security approach
- Stakeholders still “don’t always get it” ...
I DISCOVERED A HOLE IN OUR INTERNET SECURITY.

WHAT?!

GOOD GRIEF, MAN! HOW COULD YOU PUT A HOLE IN OUR INTERNET?

ACTUALLY, THAT’S NOT MY JOB. BUT I’LL INFORM OUR NETWORK MANAGEMENT GROUP.

PASSING THE BUCK!!! YOU’RE A BUCK PASSER!!!

I DIDN’T PUT IT THERE. I FOUND IT...AND IT’S NOT...

FORGET IT! THERE’S NO HOLE! IT GOT BETTER!

THAT’S MORE LIKE IT.

IT’S YOUR JOB TO FIX THAT HOLE. I WANT YOU TO WORK 24-7!

I FIXED THE INTERNET.
How To Best Dress For Bad Weather
Best App Security Practices

- Analyze and know your security boundaries and attack surfaces
- Beware of reliance on client-side security measures
  - Always implement strong server side input & parameter validation (black & whitelisting)
  - Test against a robust set of evasion rules
- Remember: The client can never be trusted!
- Assume the worst case scenario for all 3rd party interactions
  - 3rd parties can inherently not be trusted!
Best App Security Practices (contd.)

- Implement anti-CSRF defenses
- Escape special characters before sending them to the browser (e.g. `<` to `&lt;`)
- Leverage HTTPS for sensitive data, use `HTTPOnly` & `Secure` cookie flags
- Use parameterized SQL for any DB queries
- Don't disclose any stack trace, debug log, or path information or failed SQL statements to users
- Use strong tokens with strong randomness
Best App Security Practices (contd.)

- Implement a comprehensive, solid exception handling architecture
  - Default error handler which returns sanitized error message for all error paths
  - Do not disclose any stack trace, debug log, or path information or failed SQL statements to users
Best App Security Practices (contd.)

- Beware of weak / faulty session management
  - Use strong authentication mechanism (e.g. two factor)
  - Avoid weak passwords & weak change / forgot password mechanisms
  - Implement strong logout functionality (with invalidation of session tokens & deletion of session & state on server)
  - Implement session expiration with same results as strong logout (after e.g. 5 or 10 minutes)
Beware of weak / faulty session management (contd.)

- Ideally do not allow concurrent logins
- Terminate sessions when attacks are detected
- And always remember: The strongest authentication won't help if session management vulnerabilities exist!

Also see owasp.org and OWASP dev guide
Security In The Real World …

It’s true, you might not be able to outrun the bear, but let’s not forget, all you have to do is outrun your competition!
Things to Remember

- Attackers can be extremely creative and overcome various defense mechanisms
- Never assume you’re safe just because you’ve implemented a few basic defenses
- Never underestimate your opponent!
<table>
<thead>
<tr>
<th></th>
<th>Areas of Testing / People involved</th>
<th># of Attacks</th>
<th>Testing Freq</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>No areas tested &gt; No People</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>2</td>
<td>Intermittent testing of Dev, QA &gt;&gt; InfoSec (or just 1 person)</td>
<td>Basic 5 – 10 attacks</td>
<td>Test once or twice</td>
</tr>
<tr>
<td>3</td>
<td>Dev, QA Tested, Testing pre-prod apps &gt; InfoSec, Mgmt (few people)</td>
<td>Intrusive attacks</td>
<td>Test every year</td>
</tr>
<tr>
<td>4</td>
<td>Dev, QA &amp; Safe testing of Production apps &gt; Execs, InfoSec, Dev (more people, but no standardization)</td>
<td>Infrastructure + (non)-intrusive</td>
<td>Testing every 6 mo</td>
</tr>
<tr>
<td>5</td>
<td>Dev, QA, and full production Tested &gt; Execs, InfoSec, Dev, QA (most of the company is security driven)</td>
<td>Application logic tests + all others</td>
<td>Continuous Testing / monthly</td>
</tr>
</tbody>
</table>
Application Security Maturity Model

People & Process

Tools & Technology

Low

High

Panic Scramble

Pit of Despair

Security as Core Business Process

1

2

3

4

5
Website Testing: Best Practices

Low RISK SCORE

High

Test All Apps For HealthCheck

Strong Testing for Important Apps

Deep Testing for Critical Apps

CENZIC HAILSTORM ENTERPRISE ARCTM
Dashboard

Website Testing: Best Practices
Risk Management Dashboard

Tells which apps have been tested

Finds and lists all applications

Web Interface

Tells vulnerability levels

Quantitatively tells how severe the risk is for each app
Sophistication of Hackers ...
Meets Unprepared Users …

© Scott Adams, Inc./Dist. by UFS, Inc.