A Call for Drastic Action

A Survey of Web Application Firewalls

OWASP
The Open Web Application Security Project
• Jaeson Yoo, Head of Global Business
• Responsible for Penta’s international business interests, for all regions/countries outside of Korea and Japan
• Management Consulting and Investment Banking background, with experience in IT security industry
AGENDA

• Penta Security Systems Inc.
• Web Application Firewalls: An Overview
• Why Aren’t WAFs More Prevalent?
• The Traditional Signature-Based Approach is Failing
• What WAF Solution is Needed?
- **Over 2,000 clients including:** government, small and medium businesses, enterprise, academic organizations, and financial organizations (including banks, securities, insurance, and credit card companies).

- **Leading technology provider of** Web Application Firewalls, DBMS Encryption Systems, and PKI Solutions.

- **Our appliance based WAF holds 70%+ market share in the growing Korean WAF market, and we hold 60% of the Korean market share in the DBMS encryption market.**

<table>
<thead>
<tr>
<th>Founded</th>
<th>July 1997</th>
</tr>
</thead>
<tbody>
<tr>
<td>CEO/Founder</td>
<td>Seokwoo Gregory Lee</td>
</tr>
<tr>
<td>Headquarters</td>
<td>Seoul, Korea</td>
</tr>
<tr>
<td>Revenues</td>
<td>US$ 18.4M (2012)</td>
</tr>
<tr>
<td>Employees</td>
<td>120+</td>
</tr>
<tr>
<td>Overseas</td>
<td>Penta Security Systems, K.K., Tokyo, Japan (subsidiary)</td>
</tr>
<tr>
<td></td>
<td>Thailand, Singapore, Australia, Malaysia, Indonesia (distributors)</td>
</tr>
<tr>
<td>Products</td>
<td>WAPPLES (Web Application Firewall)</td>
</tr>
<tr>
<td></td>
<td>D’Amo (DB Encryption Solution)</td>
</tr>
<tr>
<td></td>
<td>ISSAC (PKI Solution)</td>
</tr>
<tr>
<td></td>
<td>ISign Plus (SSO Solution)</td>
</tr>
</tbody>
</table>
## Company History

### 2008 ~ Present

**2012**
- 01. WAPPLES granted Japanese patent for its algorithm-based, application-layer attack detection engine.
- 12. Joined the VMware TAP program and became a VMware Solution Provider

**2011**
- 12. Launched KT WAF service with WAPPLES V-Series
- 12. D’Amo granted US patent for its query processing system and method for DB query encryption transformation
- 08. D’Amo received v2.3 SAP Certification SAP in Germany
- 06. Became an OWASP sponsor
- 02. Received the Competitive Power Software Company award from KISA
- 03. Green-Biz Authorized

**2010**
- 07. Received the International Security Leadership Achievements (ISLA) Leadership Award
- 02. Received the Competitive Power Software Company award from KISA

**2009**
- 12. Acquired certification of PCI-DSS conformance for WAPPLES
- 11. Received the 2009 award for Deployment of Innovative Technology
- 04. Received the New Software Grand Award for WAPPLES
- 01. Penta Security Systems K.K.(Japan) was established

**2008**
- 11. WAPPLES received the Intelligence Award
- 10. GS Caltex, SSO and DB security system built
- 07. CIS-CC received Validation Certification from NIS
- 07. D’Amo for DB2 was introduced
- 06. WAPPLES received Security Compatibility Certification
- 05. D’Amo SG was introduced
- 04. QUEST in Japan, became a contract agent of WAPPLES
- 01. WAPPLES became CC (Common Criteria) Certified

**2010 ISLA Award**
**2009 New Software Grand Award**
**2009 Intelligence Award**
**2007 GS Certification**

### 1997 ~ 2007

**2007**
- 04. WAPPLES received GS (Good Software) certification
- 03. DB Encryption Established in the Ministry of Foreign Affairs and Trade

**2006**
- 10. D’Amo for SQL Server was introduced
- 07. Local office in Japan was established
- 06. Selected as an Innovative Business (INNO-BIZ)
- 03. SCP, PKI-based Security Library, was introduced
- 01. D’Amo received GS (Good Software) Certification

**2005**
- 05. WAPPLES, Web Application Firewall, was introduced

**2004**
- 12. Initiated sales of D’Amo in Japan
- 03. D’Amo, DB Database Security Solution, was introduced

**2003**
- 08. CIS received encryption certification from NIST
- 01. Received an award from the Korean government for realizing the e-Government project

**2002**
- 09. Established GPKI system with the Korean government

**2001**
- 09. ISign EAM Solution was introduced

**2000**
- 06. Received the Grand Award in the Information field of Venture business

**1999**
- 07. Established the first DB security business in Korea

**1998**
- 09. Penta Security Lab was founded
- 08. Siren, Intrusion Detection System, was introduced
- 01. ISSAC, PKI Solution, was introduced

**1997**
- 08. Information Security Technical cooperation with POSTECH
- 07. Penta Security Systems was founded
Intelligent Web Application Firewall, WAPPLES

Leading WAF product in Korea with 70%+ market share
• Unique Logic Based Detection Engine
• Near-zero false positive detection rate, and immunity to unknown attacks
• Security Compatibility Certification from the Korean National Intelligence Service
• PCI-DSS Certificate of Conformity

PKI Security Solution, ISSAC

The first PKI product in Korean meeting international technology standards
• User authentication is provided through the NPKI and GPKI systems
• Supports web session encryption and user authentication

PKI Development Toolkit, SCP

Cryptographic Module, CIS

Cryptographic function and security service development tool
• Supports embedded environment
• Able to support from PC to Mainframe

Integrated DB Security, D’Amo

No.1 Database Encryption Software in Korea since 2004 with 500+ customers of all sectors
• Runs on Oracle (Enterprise/Standard), MS SQL Server, DB2, and MySQL
• Supports various security configuration through D’Amo SG and D’Amo SCP
• Enables realization of strong, comprehensive DB security

SSO & EAM Solution, Isign Plus

The first SSO solution in Korea
• Single Sign On (SSO) solution for enterprise security since 2001
• Extranet Access Management based on RBAC (Role Based Access Control)

MySQL Encryption, MyDiamo

The world’s first advanced encryption & access control solution for MySQL
• Open-source Software
• Offered as freeware to individuals with non-commercial interests, educational institutions
• One column of free encryption for all other users
<table>
<thead>
<tr>
<th>Industry</th>
<th>Customer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Online Shopping</td>
<td>AEON Mall</td>
</tr>
<tr>
<td>Data Center</td>
<td>DSR</td>
</tr>
<tr>
<td>Department Store</td>
<td>Shibuya109 Net Shop</td>
</tr>
<tr>
<td>Finance</td>
<td>Nissay Asset Management</td>
</tr>
<tr>
<td>Finance</td>
<td>Visa Credit Card Japan</td>
</tr>
<tr>
<td>Service Provider</td>
<td>Sanwa Comtech</td>
</tr>
<tr>
<td>Service Provider</td>
<td>ST-NET</td>
</tr>
<tr>
<td>Outsourcing</td>
<td>TECSEED Compass Inc</td>
</tr>
<tr>
<td>Education</td>
<td>Tokyo University of Agriculture</td>
</tr>
<tr>
<td>Government</td>
<td>Malaysia Ministry of Finance</td>
</tr>
</tbody>
</table>

**Korea**

<table>
<thead>
<tr>
<th>Industry</th>
<th>Customer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manufacturing</td>
<td>BMW Korea</td>
</tr>
<tr>
<td>Banking</td>
<td>HSBC Korea</td>
</tr>
<tr>
<td>Insurance</td>
<td>ING Life</td>
</tr>
<tr>
<td>Service</td>
<td>JW Marriott Seoul</td>
</tr>
<tr>
<td>Manufacturing</td>
<td>Honda Korea</td>
</tr>
<tr>
<td>Electronics</td>
<td>LG Electronics Inc.</td>
</tr>
<tr>
<td>IT</td>
<td>Samsung SDS</td>
</tr>
<tr>
<td>Insurance</td>
<td>MetLife Korea</td>
</tr>
<tr>
<td>Manufacturing</td>
<td>NIKE Sportswear Korea</td>
</tr>
<tr>
<td>Distribution</td>
<td>LG International Corp.</td>
</tr>
</tbody>
</table>
Web Application Firewalls: An Overview
A web application firewall (WAF) secures web applications!

What is a WAF?

- **Web Application Firewall**
  - It executes a security **analysis of the OSI 7** layer between all messages of the web server and the web client.
  - It protects against attacks aimed at **web applications**.

- **Roles**
  - It protects web servers from **external attacks** (service in)
  - It **protects against leakage** of important information from the web server (service out)
• Protects against web attacks
• Prevents leakage of personal, confidential, and/or proprietary information
• Enables regulatory compliance
Web application firewalls enable regulatory compliance.

**Payment Card Industry Data Security Standard** (PCI DSS, 2004) is an international information security standard for companies dealing with electronic payment transactions (credit cards, debit cards, etc.).

- Requires secure management of cardholder data.
- Fines for violations can range from $5000-$100,000 USD per month.

Other well-known regulations which can be followed with accurate web application security include the **Health Insurance Portability and Accountability Act** (HIPAA, USA, 1996), and the **Federal Information Security Management Act** (FISMA, USA, 2002).
Network Firewalls and IDS/IPS cannot protect web applications against the OWASP Top 10 Threats, but a Web Application Firewall (WAF) can!

<table>
<thead>
<tr>
<th>OWASP Top 10 (2010)</th>
<th>Network Firewall</th>
<th>IDS / IPS</th>
<th>WAF</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1: SQL Injection</td>
<td>X</td>
<td>△</td>
<td>✓</td>
</tr>
<tr>
<td>A2: Cross Site Scripting (XSS)</td>
<td>X</td>
<td>△</td>
<td>✓</td>
</tr>
<tr>
<td>A3: Broken Authentication and Session Management</td>
<td>X</td>
<td>△</td>
<td>✓</td>
</tr>
<tr>
<td>A4: Insecure Direct Object References</td>
<td>X</td>
<td>X</td>
<td>✓</td>
</tr>
<tr>
<td>A5: Cross Site Request Forgery (CSRF)</td>
<td>X</td>
<td>X</td>
<td>✓</td>
</tr>
<tr>
<td>A6: Security Misconfiguration</td>
<td>X</td>
<td>X</td>
<td>✓</td>
</tr>
<tr>
<td>A7: Failure to Restrict URL Access</td>
<td>X</td>
<td>X</td>
<td>✓</td>
</tr>
<tr>
<td>A8: Insecure Cryptographic Storage</td>
<td>X</td>
<td>X</td>
<td>✓</td>
</tr>
<tr>
<td>A9: Insufficient Transport Layer Protection</td>
<td>X</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>A10: Unvalidated Redirects and Forwards</td>
<td>X</td>
<td>X</td>
<td>✓</td>
</tr>
</tbody>
</table>
Comparing Network Firewalls, IDS/IPS, and WAFs:

<table>
<thead>
<tr>
<th>Network Firewalls</th>
<th>IDS/IPS</th>
<th>Web Application Firewall</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Network Firewalls cannot detect or analyze attacks targeting web applications.</td>
<td>• Cannot typically analyze application layer attacks.</td>
<td>• Can inspect all traffic between the Web client and Web server</td>
</tr>
<tr>
<td>• Network Firewalls generally cannot protect Port 80.</td>
<td>• Easy for attackers to bypass</td>
<td>• Can inspect protocol</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Can inspect encrypted traffic</td>
</tr>
</tbody>
</table>

OSI Network Model (7 Layers)
Why Aren’t WAFs More Prevalent?
A Survey of Web Application Firewalls
The threats to web applications are clear. So why have so many large businesses been reluctant to install web application firewalls?

The first generation web application firewalls was based on a pattern-matching detection mechanism.

- **First Generation WAFs: Adoption of the Black List**
  - The administrator added a known attack pattern (black list) to block malicious traffic.
  - First generation WAFs compared web traffic to the updated pattern by analyzing them at the application level.
  - No detection system for new or modified attacks against the web application layer.
The First Generation WAF was not particularly successful in the IT security market. Not only did it place a significant burden on system administrators, they were not seen as particularly effective.

- **Heavy Workload & HR Requirements**
  - Administrators needed to update known attack patterns on the black list.
  - Attempts to add new conceivable patterns meant a heavy workload for the WAF administrative team.
  - The high costs of manpower were significant, especially for smaller companies or IT security providers with limited budgets.
Vulnerabilities with Poor System Performance

- No protection against new or modified attacks
- Significant increase in false positives
- Attempts to add patterns for all conceivable attacks led to a deterioration in web application performance

Decrease in Web Application Performance
The second generation WAF attempted to remedy the flaws and limitations of its predecessor.

- **Second-Generation WAFs: The Advent of the automated policy**
  - By analyzing the web application(s) protected by the WAF, the second-generation WAF was able to automate the establishment of “white lists” for security policies.
  - Combined with a black list containing known and conceivable attacks against the application layer, the second generation WAF could figure out what was permissible traffic, and what was not permissible.
Unfortunately, the second generation WAF did little to increase application layer security, or alleviate manpower requirements.

- **Heavier Workload!**

  - The automatically established policies, or the white lists, could take up to two weeks to identify and implement.

  - Additionally, while the policies were established automatically, they still required manual configuration by an administrator, thereby increasing (not reducing) the administrative burden.

  - The security administrator still has to maintain the black list of known web attacks.
Unfortunately, the second generation WAF did little to increase application layer security, or alleviate manpower requirements.

- **Still Vulnerable with Poor System Performance**
  - Still a pattern-matching solution
  - Still an inability to protect against unknown attacks
  - Still a tendency to produce false positives
  - Still promoted slow system performance

Decrease in Web Application Performance
The Traditional Signature-Based Approach is Failing
“There is widespread agreement that advanced attacks are bypassing our traditional signature-based security controls and persisting undetected on our systems for extended periods of time.

The threat is real. You are compromised; you just don’t know it.”

Source: Gartner, 2012
# The Exploit Database

The Exploit Database (EDB) - an ultimate archive of exploits and vulnerable software. A great resource for penetration testers, vulnerability researchers, and security addicts alike. Our aim is to collect exploits from submittals and mailing lists and concentrate them in one, easy to navigate database.

## Remote Exploits

<table>
<thead>
<tr>
<th>Date</th>
<th>D</th>
<th>A</th>
<th>V</th>
<th>Description</th>
<th>Plat.</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012-08-21</td>
<td></td>
<td></td>
<td>✓</td>
<td>Sysax Multi Server 5.64 Create Folder Buffer Overflow</td>
<td>381</td>
<td>metasploit</td>
</tr>
<tr>
<td>2012-08-20</td>
<td></td>
<td></td>
<td>✓</td>
<td>Adobe Flash Player 11.3 Font Parsing Code Execution</td>
<td>2356</td>
<td>metasploit</td>
</tr>
<tr>
<td>2012-08-20</td>
<td></td>
<td></td>
<td></td>
<td>Sysax Multi-Server 5.64 Create Folder Buffer Overflow</td>
<td>395</td>
<td>Matt Andreko</td>
</tr>
<tr>
<td>2012-08-18</td>
<td></td>
<td></td>
<td>✓</td>
<td>Apple Quicktime plugin - Windows 4.1.2 (Japanese) Remote Overflow Vulnerability</td>
<td>937</td>
<td>UNYUN</td>
</tr>
<tr>
<td>2012-08-16</td>
<td></td>
<td></td>
<td>✓</td>
<td>IE Time Element Memory Corruption Exploit (MS11-050)</td>
<td>2671</td>
<td>Ciph3r</td>
</tr>
<tr>
<td>2012-08-16</td>
<td></td>
<td></td>
<td>✓</td>
<td>E-Mail Security Virtual Appliance (ESVA) Remote Execution</td>
<td>971</td>
<td>iJoo</td>
</tr>
<tr>
<td>2012-08-15</td>
<td></td>
<td></td>
<td>✓</td>
<td>TestLink v1.9.3 Arbitrary File Upload Vulnerability</td>
<td>1051</td>
<td>metasploit</td>
</tr>
</tbody>
</table>

## Local Exploits

<table>
<thead>
<tr>
<th>Date</th>
<th>D</th>
<th>A</th>
<th>V</th>
<th>Description</th>
<th>Plat.</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012-08-15</td>
<td></td>
<td></td>
<td>✓</td>
<td>globalISCAPE CuteZIP Stack Buffer Overflow</td>
<td>558</td>
<td>metasploit</td>
</tr>
<tr>
<td>2012-08-15</td>
<td></td>
<td></td>
<td>✓</td>
<td>Windows Service Trusted Path Privilege Escalation</td>
<td>1622</td>
<td>metasploit</td>
</tr>
<tr>
<td>2012-08-13</td>
<td></td>
<td></td>
<td></td>
<td>OS X Local Root Exploit for Viscosity OpenVPN Client</td>
<td>1565</td>
<td>zx2c4</td>
</tr>
<tr>
<td>2012-08-11</td>
<td></td>
<td></td>
<td>✓</td>
<td>Tunnelblick Local Root Exploit</td>
<td>1752</td>
<td>zx2c4</td>
</tr>
</tbody>
</table>
Example of misdetection with a Signature-Based Engine

In the case that a WAF has the below signatures:

**Example of SQL Injection Signature**
- Below ‘part’ means substring-searching target. ‘rgxp’ means a regular expression.
- After finding a string of ‘part’, ‘rgxp’ is applied.

<table>
<thead>
<tr>
<th>Signature</th>
<th>Signature Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>part=&quot;or 'a' = 'a&quot;</td>
<td>SQL Injection WHERE Statement Manipulation</td>
</tr>
<tr>
<td>part=&quot;or 'a'= 'a&quot;</td>
<td>SQL Injection WHERE Statement Manipulation 1</td>
</tr>
<tr>
<td>part=&quot;or 'a'= 'a&quot;</td>
<td>SQL Injection WHERE Statement Manipulation 2</td>
</tr>
<tr>
<td>part=&quot;or 'a'=='a&quot;</td>
<td>SQL Injection WHERE Statement Manipulation 3</td>
</tr>
<tr>
<td>part=&quot;or 'a' ='a&quot;</td>
<td>SQL Injection WHERE Statement Manipulation 4</td>
</tr>
<tr>
<td>part=&quot;or 'a'=='a&quot;</td>
<td>SQL Injection WHERE Statement Manipulation 5</td>
</tr>
<tr>
<td>part=&quot;or 'a'=='a&quot;</td>
<td>SQL Injection WHERE Statement Manipulation 6</td>
</tr>
<tr>
<td>part=&quot;or a=a&quot;</td>
<td>SQL Injection WHERE Statement Override</td>
</tr>
<tr>
<td>part=&quot;or 1=1&quot;</td>
<td>SQL Injection WHERE Statement Override 1</td>
</tr>
</tbody>
</table>

If a SQL Injection source is modified from ‘a’=‘a’ to ‘b’=‘b’, the regular expression cannot detect the modified SQL Injection attack.
Example of a false-positive with a Signature Based Engine

In the case that a WAF has the below signatures:

<table>
<thead>
<tr>
<th>Signature</th>
<th>Signature Name</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>part=&quot;waitfor&quot;, rgxp=&quot;[^a-zA-Z]waitfor's*delay&quot;</code></td>
<td>SQL Injection - Waitfor</td>
</tr>
<tr>
<td><code>part=&quot;having&quot;, rgxp=&quot;[^A-Za-z]having[^\&amp;](0,20)\=[^\&amp;](0,20)&quot;</code></td>
<td>SQL Injection - &quot;having&quot; statement injection attempt</td>
</tr>
<tr>
<td><code>part=&quot;opendatasource&quot;, rgxp=&quot;select.*from.*opendatasource&quot;</code></td>
<td>SQL Injection – opendatasource</td>
</tr>
</tbody>
</table>

If HTTP message includes string “… **having a good time**… ––…” such as shown below, a signature based WAF would generate a false-positive.
WebShell Hacking...What is It?

• A broad term for a type of file that operates similarly to an injection
• Once successfully uploaded, the WebShell file is capable of exploiting server vulnerabilities and eventually providing the attacker with access to the web server structure
• These files are often masked as harmless, and can be quite difficult to detect until damage has already been done

Implications?

• WebShell attacks pose a very threat because they can potentially lead to complete server access and privileges for the hacker
• As with most APTs, the conventional security systems in place are ill-equipped to handle the threats of WebShell attacks, and cannot keep up with how quickly their methods are modified.
What Solution is Needed?
What WAF Solution is Needed?

The threats against web applications continue to evolve. The web application remained vulnerable and a frequent target of hackers. So what solution is needed?

- A Whole New Breed of WAF
  - An “intelligent” WAF, based on an entirely new concept
  - Capable of analyzing web traffic
  - Detect attacks, analyzing and classifying them
  - Apply appropriate countermeasures to block detected attacks
  - Be able to perform all of these functions without the continual involvement of administrative staff

A new solution would be able to execute all of the functions above, in order to protect web applications in a stable manner, while easing the administrative workload and management costs.
Thank You.

Trust for an Open Society ———

Penta SECURITY