



ACE Web Application Firewall



Ong Poh Seng

ongps@cisco.com

31st Oct 2008

Topics

- Secure Data Center Transformation
- Application Security Trends and Concerns
- Web Application Attack
- Introducing Cisco Web Application Firewall WAF
- Q&A

Secure Data Center Transformation



Cisco: Transforming Data Centers



Data Center Assurance Program

Edge to Disk Data Center systems testing and validation



Application Networking

- Empowering applications thru network
- WAN Optim, App Switching, XML Security/Offload
- Broad Portfolio of Application Networking Technologies



DC Automated Provisioning

- Link LAN, SAN, AFE, Security together w/ compute and storage
- Visibility to Business Process Execution



Secure DC Infrastructure

Purpose built infrastructure and transport systems designed for tomorrow's data centers



Unified Network Fabric

Evolution of data link to allow a single network in the data center for all traffic types

SDN Secured Data Center: big picture and where does ACE WAF play?

Data Center Edge

- Firewall & IPS
- DOS Protection
- App Protocol Inspection
- Web Services Security
- VPN termination
- Email & Web Access control

Web Access

- Web Security
- Application Security
- Application Isolation
- Content Inspection
- SSL Encryption/Offload
- Server Hardening

Apps and Database

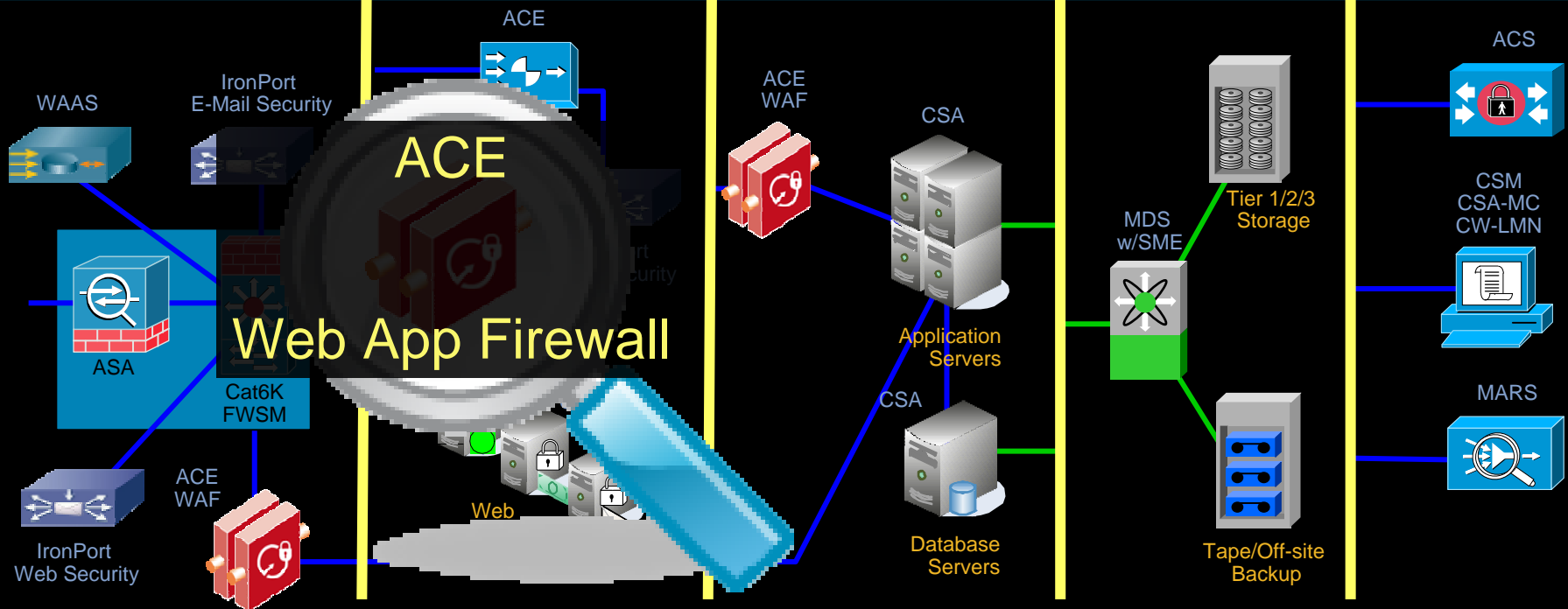
- XML, SOAP, AJAX Security
- XDoS Prevention
- App to App Security
- Server Hardening

Storage

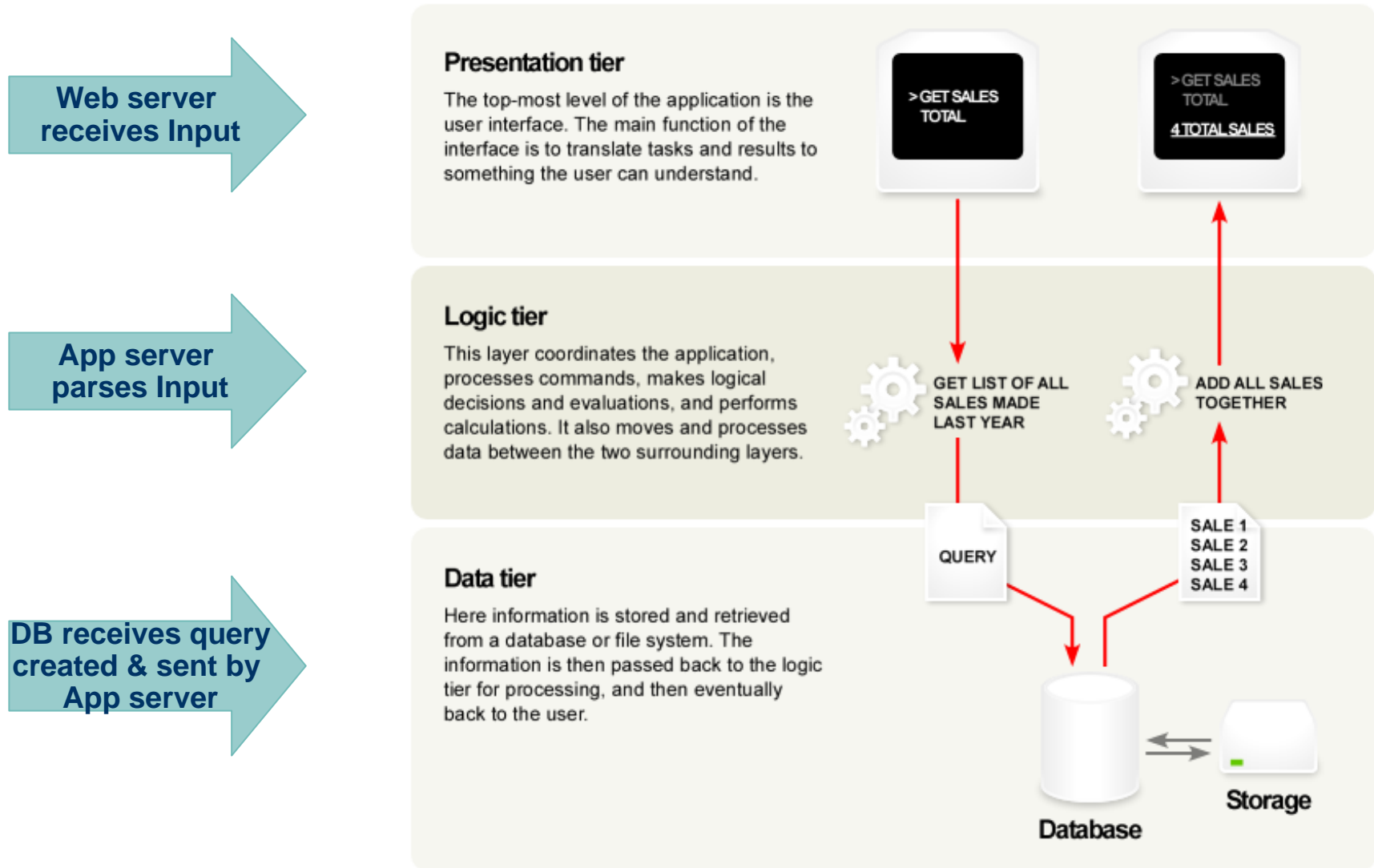
- Data Encryption
 - In Motion
 - At Rest
- Stored Data Access Control
- Segmentation

Mgmt

- Tiered Access
- Monitoring & Analysis
- Role-Based Access
- AAA Access Control



Typical Web Application Architecture



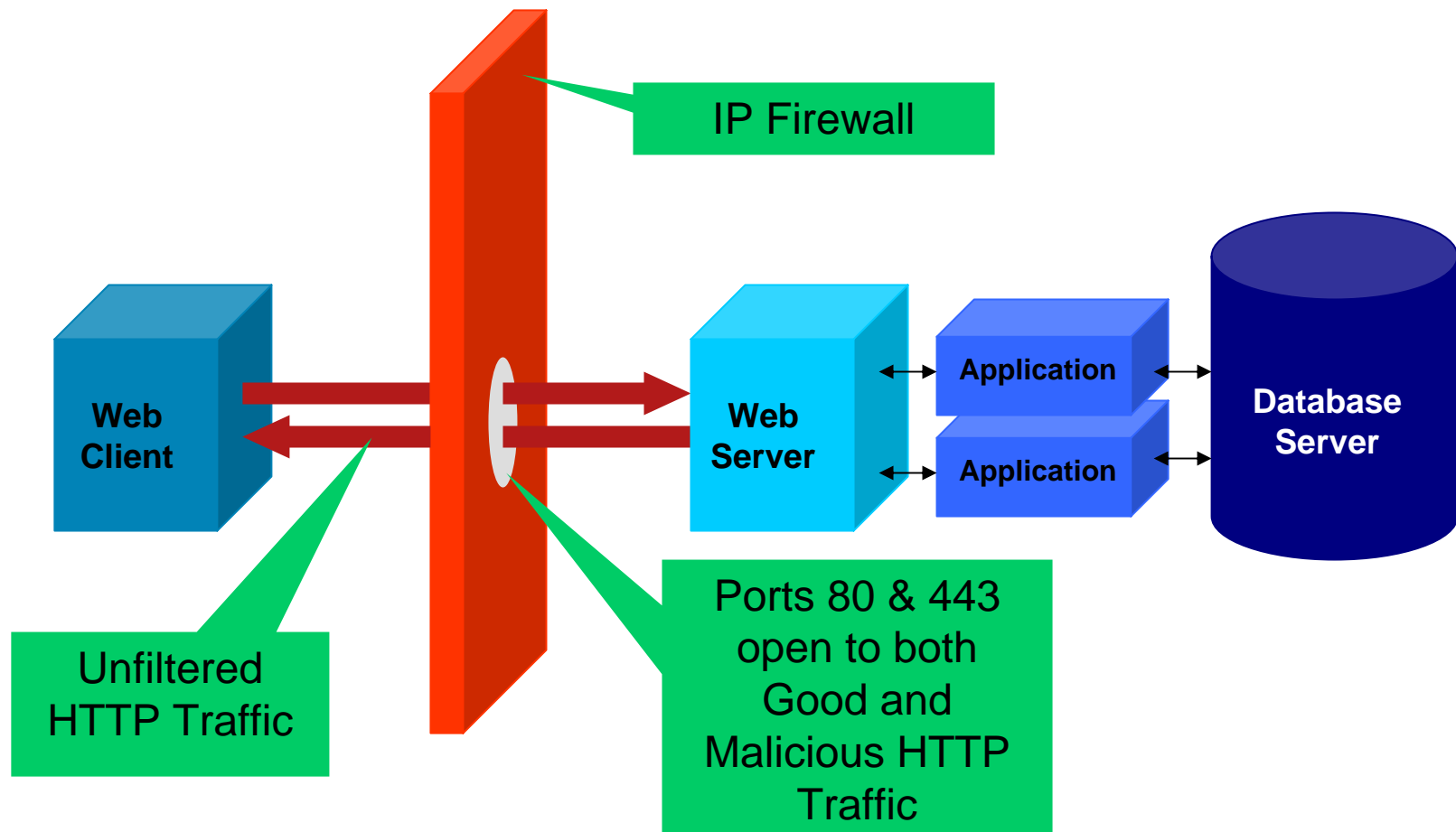
Application Security Trends and Concerns



Off the press

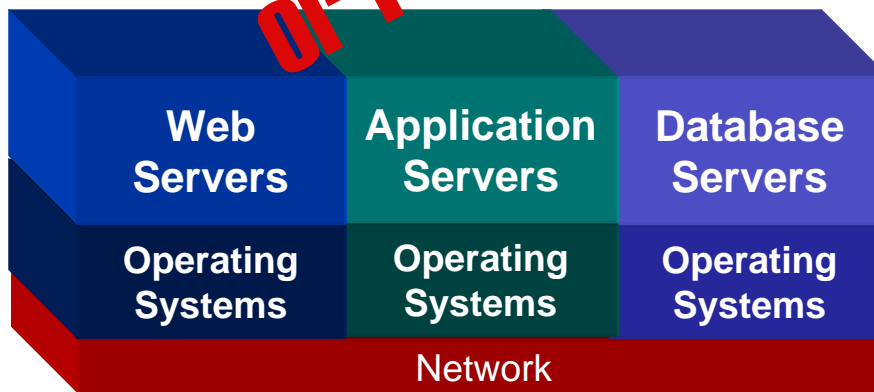
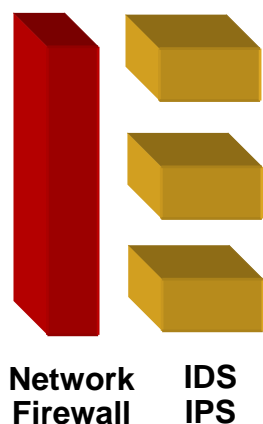
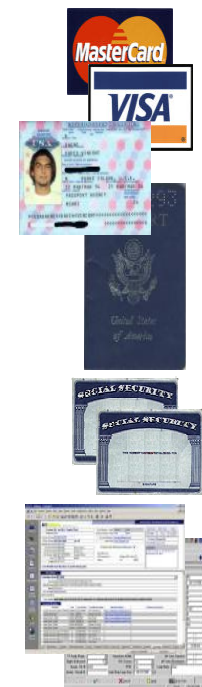
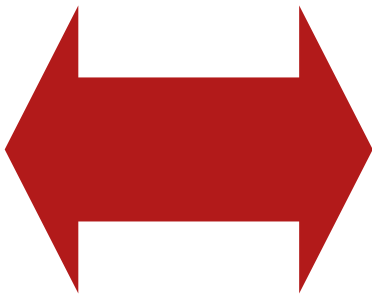
... more than 45 million credit and debit card numbers have been stolen from its IT systems ...

Traditional Network Firewalls ill-equipped to protect Web Applications



Focus of today's attacks

75% of Attacks Focused Here



No magic signatures or patches for your custom PHP script

What sort of attacks are we talking about?

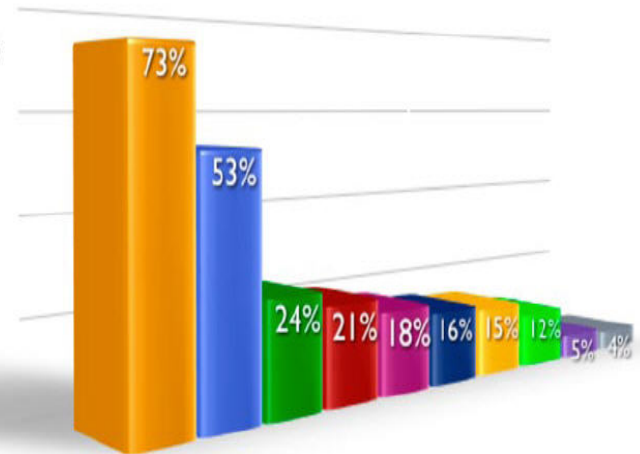
<http://www.owasp.org>

Top 10

- A1 – Cross Site Scripting (XSS)
- A2 – Injection Flaws.....
- A3 – Malicious File Execution
- A4 – Insecure Direct Object Reference
- A5 – Cross Site Request Forgery (CSRF)
- A6 – Information Leakage and Improper Error Handling
- A7 – Broken Authentication and Session Management
- A8 – Insecure Cryptographic Storage.....
- A9 – Insecure Communications
- A10 – Failure to Restrict URL Access.....

How widespread these attacks are

- Cross-Site Scripting
- Information Leakage
- Content Spoofing
- Predictable Resource Location
- SQL Injection
- Insufficient Authentication
- Insufficient Authorization
- Abuse of Functionality
- Directory Indexing
- HTTP Response Splitting



Top 10 vulnerability classes by percentage likelihood.

Source: WhiteHat Security, 2007

Industry Response



- Visa, American Express, Master Card and others (the Payment Card Industry)
 - Created a Data Security Standard (PCI DSS)
- Section 6.6:
 - Must conduct code reviews or
 - Install a Web Application Firewall**
- Every company that processes credit cards must comply or face fines
- Compliance deadline is June 30 2008
- April 15 revision added XML security to the list of requirements; recommends WAF and secure coding practices



Why Not Fix Current Applications?



Every 1000 lines of code averages 15 critical security defects

(US Dept of Defense)

The average business app has 150,000-250,000 lines of code

(Software Magazine)

The average security defect takes 75 minutes to diagnose and 6 hours to fix

(5-year Pentagon Study)

Even if you consider those figures are exaggerated (positively or negatively) the cost of fixing applications is prohibitive

WAF always a very financially sound option!

Web Attacks



Cross-Site Scripting (XSS) attacks

- **What is it?**

A malicious script is echoed back into HTML returned from a trusted web site. The scripts executes locally on the client.

Extremely widespread – some experts estimate 70%-80% of websites are vulnerable

- **What are the implications?**

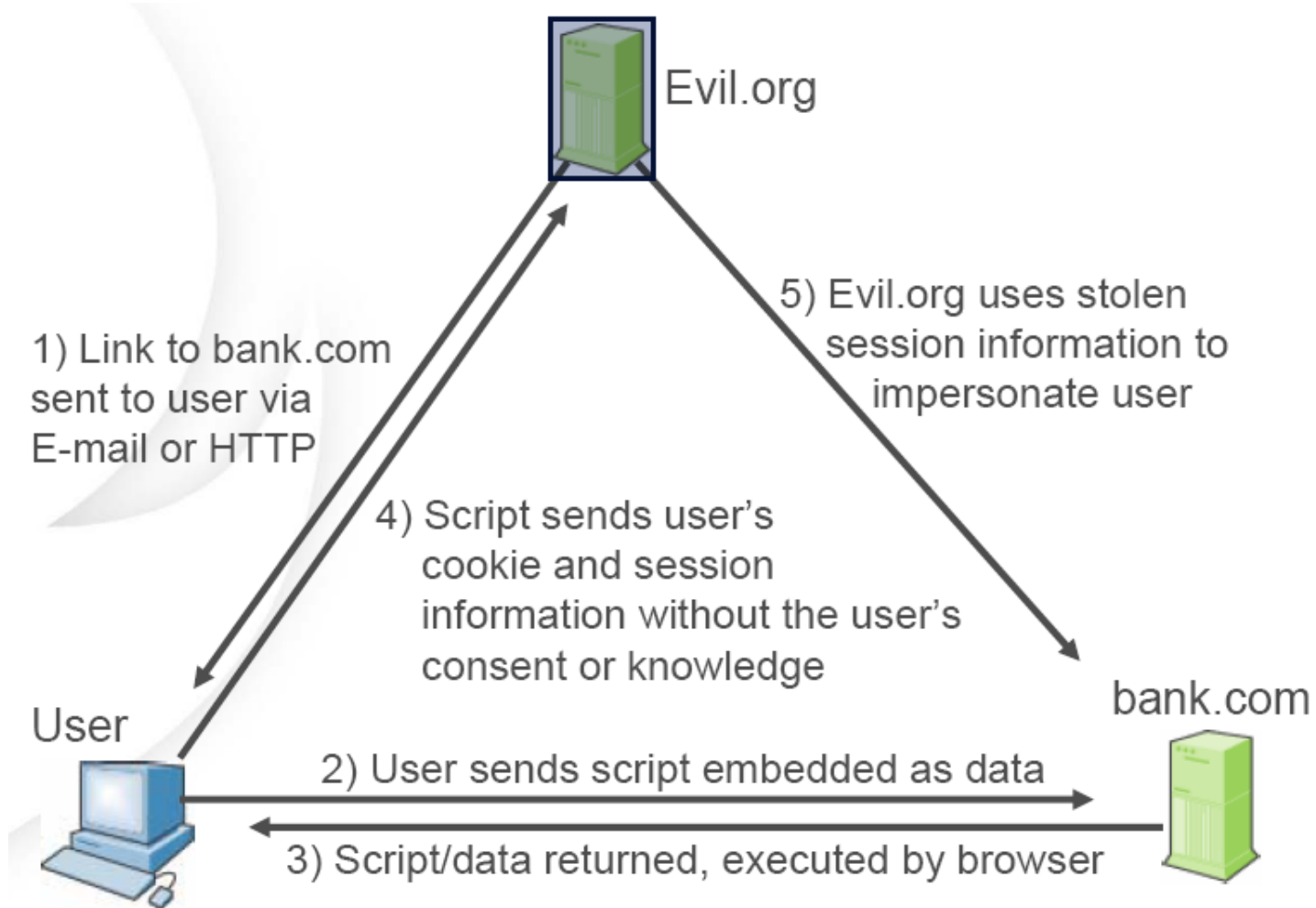
Web Site Defacement

Session IDs stolen (cookies exported to hacker's site)

Browser security compromised – control given to hacker

All data sent between client and server potentially hijacked

The XSS attack process



SQL Injection

- SQL stands for **Structured Query Language**
- Allows applications to access a database
- SQL can:
 - Execute queries against a database
 - Retrieve data from a database
 - Insert new records in a database
 - Delete records from a database
 - Update records in a database
- Many applications take user input and blindly send it directly to SQL API!

Response Message Rewrite

- Search for and replace questionable content in responses from server

Data you submitted

The first param is **4444 4444 4444 4444**

The second param is

Data you submitted

The first param is **xxxxxxxxxxxxxxxxxxxxxxxx**

The second param is

Cross Site Request Forgery

- “Whereas cross-site scripting exploits the **trust** a user has in a website, a cross-site request forgery exploits the trust a Web site has in a user by **forging** a request from a trusted user.” (source: Wikipedia)
- How does it work:
 - Bob is logged into his bank’s website
 - Bob is also chatting/reading a blog at the same time
 - Hacker posts a comment in the blog inviting Bob to click a link
 - The link performs an action on Bob’s bank
 - As Bob is logged in, the action has the potential to succeed
- Simple example: <http://www.google.com/setprefs?hl=ga>
- Note that Bob doesn’t even have to click a link – a simple [](http://example.org/buy.php?item=PS3&qty=500) on a web page could suffice!

Introducing the ACE Web Application Firewall



Introducing...

The ACE Web Application Firewall (WAF)



Drop-in solution for

PCI Compliance, Virtual App Patching, Data Loss Prevention

- **Secure** – Deep packet protection of the most common vulnerabilities
- **Fast** – Processes 3,000+ TPS and 10,000+ concurrent connections
- **Drop-in** - Does not require recoding applications, deployable in under an hour
- **PCI 6.5/6.6 compliance is just a few clicks away**

Key Release 6.0 Features

Threat Protection

- Extensive Threat Signatures
- HTTP Input Normalization
- Application Cloaking
- Encrypted & Tamperproof Cookies
- SSL client and server decryption
- Data overflow protection
- Data Theft Prevention
- Custom error remapping
- Egress content rewrite

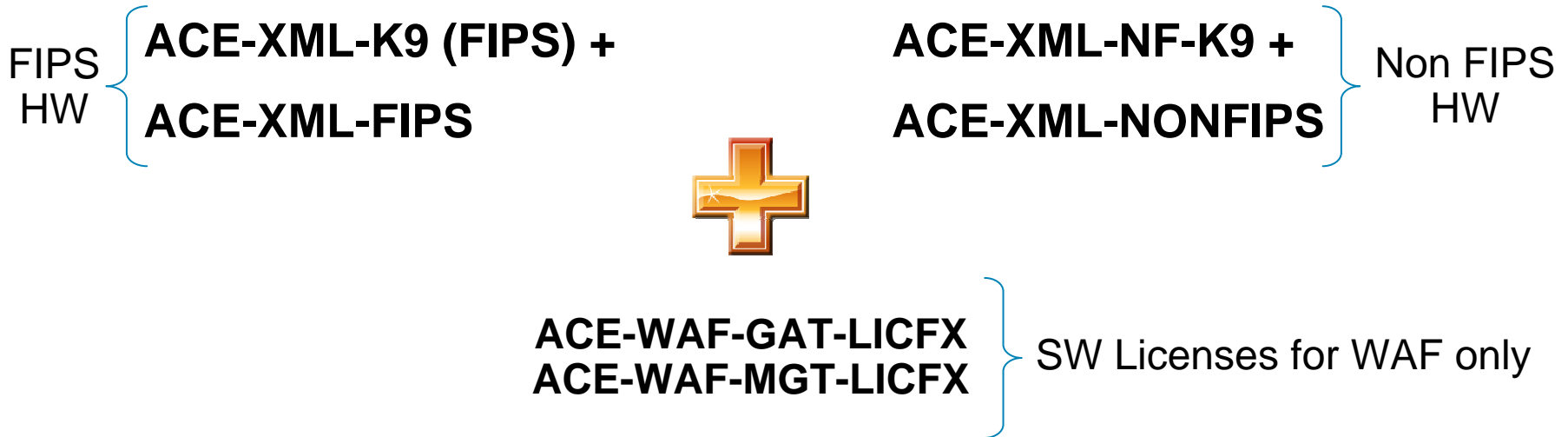
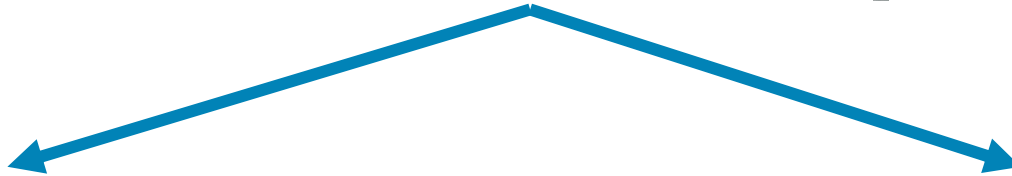
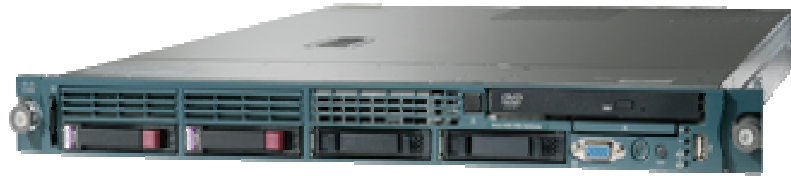
Usability

- Powerful yet simple GUI
- Seamless Signature Updates
- Human-assisted site learning
- MIB & Statistics
- Instant alerting and reporting
- Change control and audit log
- Extensive Security Logging

Addresses All Key PCI Requirements

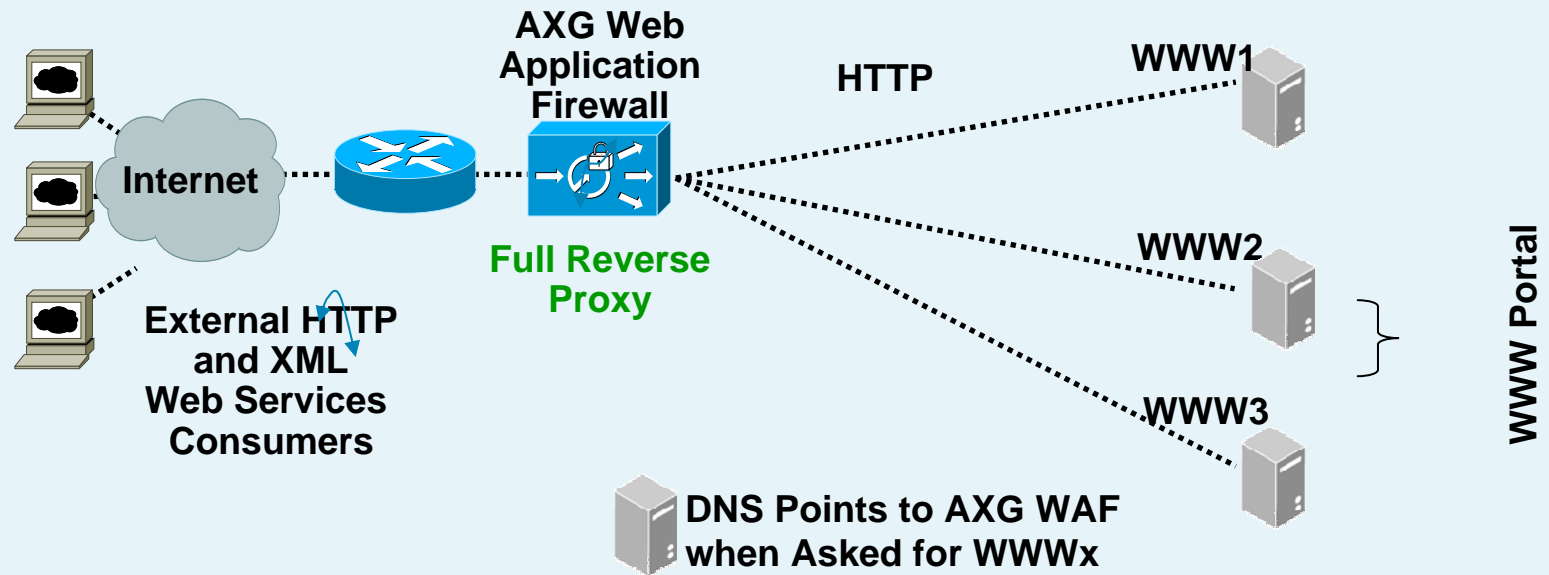
1. Cross-Site Scripting	2. Injection Flaws
3. Malicious file execution	4. Insecure direct object ref
5. CSRF	6. Improper error handling
7. Broken authentication	8. Insecure cryptographic storage
9. Insecure cryptographic comms	10. Failure to restrict URL access

Hardware and Software Part Numbers



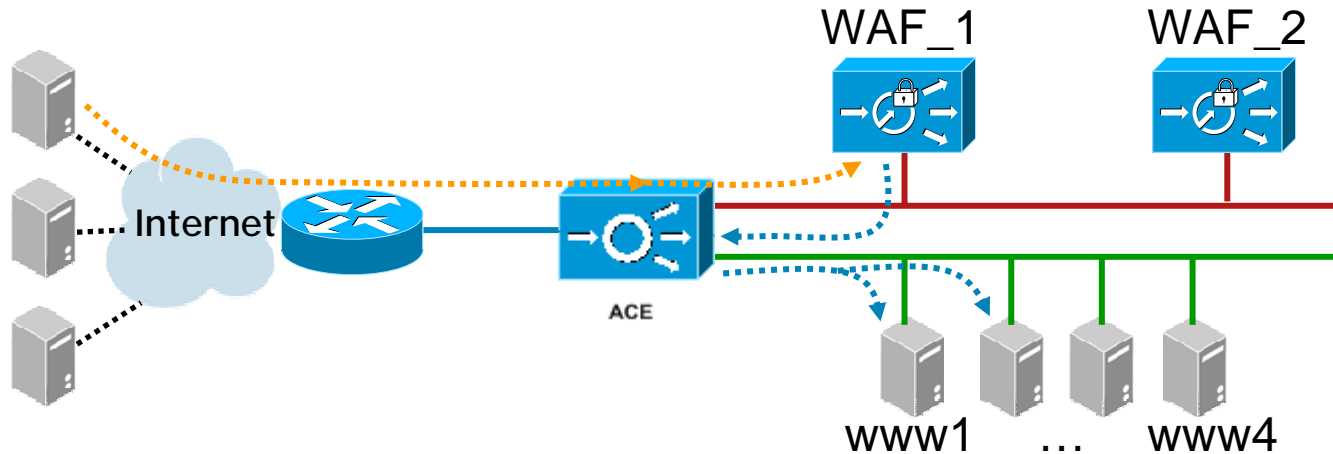
There is also a “full” license which contains both XML/Web Services and WAF feature sets

Alternative Network Deployment Model



- The ACE Web Application Firewall is a full reverse proxy
- In other words, you can have the DNS server point to the IP address of the WAF to represent the actual Web server
- At that point, the WAF accepts all requests destined to the Web server, filters them, and sends them out; the response comes back to the WAF as well for total control of the session

Typical Network Deployment



- Clients resolve www.site.com to a VIP residing on the ACE
- The ACE picks a WAF and sticks the session to it
- The WAF chooses a policy based on the Host header
- When done with the inspection, the WAF sends the packet out to an internal VIP
- That internal VIP represents the actual www servers, ACE performs the LB decision and sticks the WAF session to one real server

The Website Is Under Attack

13. We Are Launching a XSS Attack Against the Website

Web App Firewall Incidents

Group by Virtual Web App -- all records --

[Update View](#) Incident records are available for the last 12 minutes CSV [Export Raw Data](#)

Description	Incidents	%	
Incidents By Virtual Web App at Feb 13 2008 01:50:35 AM PST	4	100.0%	
test	4	100.0%	[events]
http://*/	4	100.0%	[events]
CrossSiteScripting	1	25.0%	[events]
Data Overflow	2	50.0%	[events]
SqlInjection	1	25.0%	[events]

Immediate Incident
Report View

Let's Drill Down

14. Let's See What the Attack Looks Like

Event Log Viewer

Current Manager Event Logging alert, error, warning, notice [\[edit \]](#)
Current ACE XML Gateway Event Logging alert, error, warning, notice, info, debug [\[edit \]](#)

During
search events logged on for events of type
with message GUID
category (e.g., /policy/access)
component (e.g., core or console)
description

Display a maximum of events per page
[Update](#)

EVENT LOG SEARCH RESULTS AT FEB 18 2008 09:30:39 AM PST

First < Prev Displaying events 1 - 8 Next > (more recent events are shown at the top)

Time (PST)	Description	Message GUID	Host	Component	Category
Feb 18 2008 09:29:41.714 AM	W CROSSSITESCRIPTING.CrossSiteScripting1:52:REQUEST_POSTPARAM['name'] detected by rule; returning error.	45ABFA2D000014292D980A4F08849B2D	ciscowaf	reactor	/waf/incic
Feb 18 2008 09:29:41.714 AM	W Terminating HTTP session: 500 An error occurred	45ABFA2D000014292D980A4F08849B2D	ciscowaf	reactor	/session
Feb 18 2008 09:29:41.714 AM	W An error occurred for this request: An error occurred while handling the request.	45ABFA2D000014292D980A4F08849B2D	ciscowaf	reactor	/error

ID of the Rule that Caused the Alert

The Name of the Attack Vector Is Provided

Detailed Security Event Drill-Down

15. Detailed Forensics Are Available for Each Attack

EVENT LOG SEARCH RESULTS AT FEB 18 2008 09:34:51 AM PST	
First < Prev Displaying events 1 - 14 Next > (more recent events are shown at the top)	
Time (PST)	Description
Feb 18 2008 09:29:41.714 AM	D Awaiting new request on inbound connection
Feb 18 2008 09:29:41.714 AM	W CROSSSITESCRIPTING.CrossSiteScripting1:52:REQUEST_POSTPARAM['name'] detected by rule; returning error.
Feb 18 2008 09:29:41.714 AM	W Terminating HTTP session: 500 An error occurred
Feb 18 2008 09:29:41.714 AM	W An error occurred for this request: An error occurred while handling the request.
Feb 18 2008 09:29:41.714 AM	I No policy-specific error handler for WAF.CROSSSITESCRIPTING.CrossSiteScripting1:\${SIG_MATCH_SIGID}:\${SIG_MATCH_INPUT_NAME}:
Feb 18 2008 09:29:41.713 AM	I Checking limit 1
Feb 18 2008 09:29:41.713 AM	I Checking limit 0
Feb 18 2008 09:29:41.713 AM	I Checking 3 limits
Feb 18 2008 09:29:41.713 AM	I Accepted a new HTTP POST request from 171.69.141.0 for /SCRIPTS/xss.php
Feb 18 2008 09:29:41.713 AM	I HTTP POST request for /SCRIPTS/xss.php from 171.69.141.0 matched Port 'Default HTTP port'; checking for handler
Feb 18 2008 09:29:41.713 AM	I Performing normalization on '/SCRIPTS/xss.php' with mode 7211
Feb 18 2008 09:29:41.713 AM	D HTTP Trace IN: Content-Type: application/x-www-form-urlencoded Content-Length: 58 name=%3Cscript%3Ealert%28document.cookie%29%3C%2Fscript%3E
Feb 18 2008 09:29:41.711 AM	D HTTP Trace IN: POST /SCRIPTS/xss.php HTTP/1.1 Host: foobarfoo2k.cisco.com User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.8.1.12) Gecko/20080201 Firefox/2.0.0.12 Accept: text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,image/png,*/*;q=0.5 Accept-Language: en-us,en;q=0.5 Accept-Encoding: gzip,deflate Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7 Keep-Alive: 300 Connection: keep-alive Referer: http://foobarfoo2k.cisco.com/SCRIPTS/xss.php Cookie: cec_user_id=cpaggen; SMIDENTITY=zv5hvychtfb9t4nGfXC5vsozQJXocY3BBaVM0802pEHDUDb4mPyEBkbj0dOq+xQ//TR.Ziz0UpMcRa3IWnmLDn3PaHSz74dXFY3ILU

Full Dump of Incoming Request

ACE Web Application Firewall Summary

- **Future proof application security** – Full featured Web Application firewall with integrated XML Firewall
 - Extend protection for traditional HTML-based web applications to modern XML-enabled Web services applications.
- **Access enforcement**
 - AAA enforcement mechanism to secure applications from unauthorized access
- **Positive and Negative security enforcement**
 - Best of both worlds by keeping bad traffic patterns out and allowing only good traffic through
- **Human assisted learning**
 - Deploy policies and profiles in monitoring mode to prevent application downtime due to false positives typical in an automated learning environment.
- **Policy-based provisioning**
 - Increases developer productivity and ease of deployment with sophisticated GUI, rollback and versioning capabilities.

Defense-in-Depth should include a web application firewall that can quickly, effectively and cost-effectively block attacks at layers 5-7

