



IBM Software Group

Discovering the Value of Verifying Web Application Security Using IBM Rational AppScan

An IBM Proof of Technology



Agenda

- **Introductions & facilities**
- Security Landscape
- Vulnerability Analysis
 - ▶ Top Attacks Overview
 - ▶ Hands on Lab 1
- Vulnerability Analysis (continued)
 - ▶ Hands on Lab 2
- Automated Vulnerability Analysis
 - ▶ IBM® Rational® AppScan Overview
 - ▶ Hands on Lab 3



Welcome to the Technical Exploration Center

- Introductions
- Access restrictions
- Restrooms
- Emergency Exits
- Smoking Policy
- Breakfast/Lunch/Snacks – location and times
- Special meal requirements?



POT Objectives

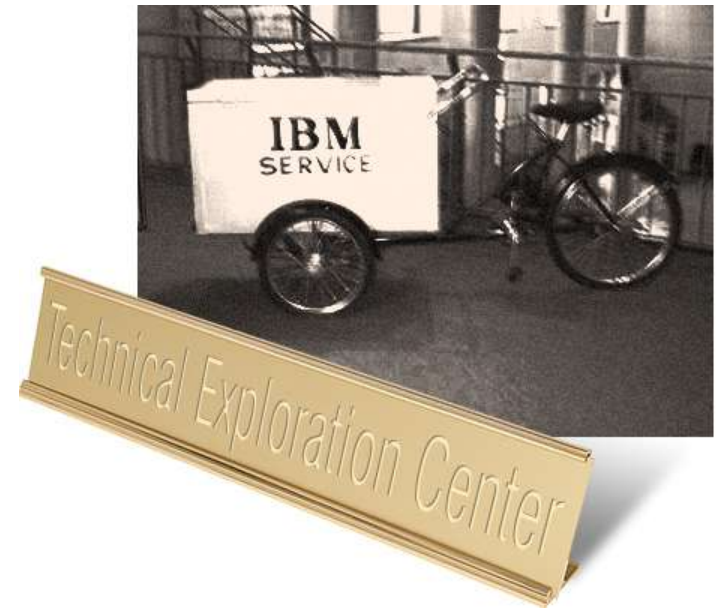
By the end of this session you will:

- Understand the Web application environment
- Understand and differentiate between network and application level vulnerabilities
- Understand where the vulnerabilities exist
- Understand how to leverage AppScan to perform an automated scan for vulnerabilities



Introductions

- Please introduce yourself
- Name and organization
- Current integration technologies/tools in use



What do you want out of this Exploration session?

Agenda

- Introductions & facilities
- **Security Landscape**
- Vulnerability Analysis
 - ▶ Top Attacks Overview
 - ▶ Cross Site Scripting
 - ▶ Hands on Lab 1
- Vulnerability Analysis (continued)
 - ▶ Hands on Lab 2
- Automated Vulnerability Analysis
 - ▶ AppScan Overview
 - ▶ Hands on Lab 3



The Alarming Truth

“Approximately 100 million Americans have been informed that they have suffered a security breach so this problem has reached epidemic proportions.”

Jon Oltsik – Enterprise Strategy Group

“Up to 21,000 loan clients may have had data exposed”

Marcella Bombardieri, Globe Staff/August 24, 2006

“Personal information stolen from 2.2 million active-duty members of the military, the government said...”

New York Times/June 7, 2006

“Hacker may have stolen personal identifiable information for 26,000 employees..”

ComputerWorld, June 22, 2006

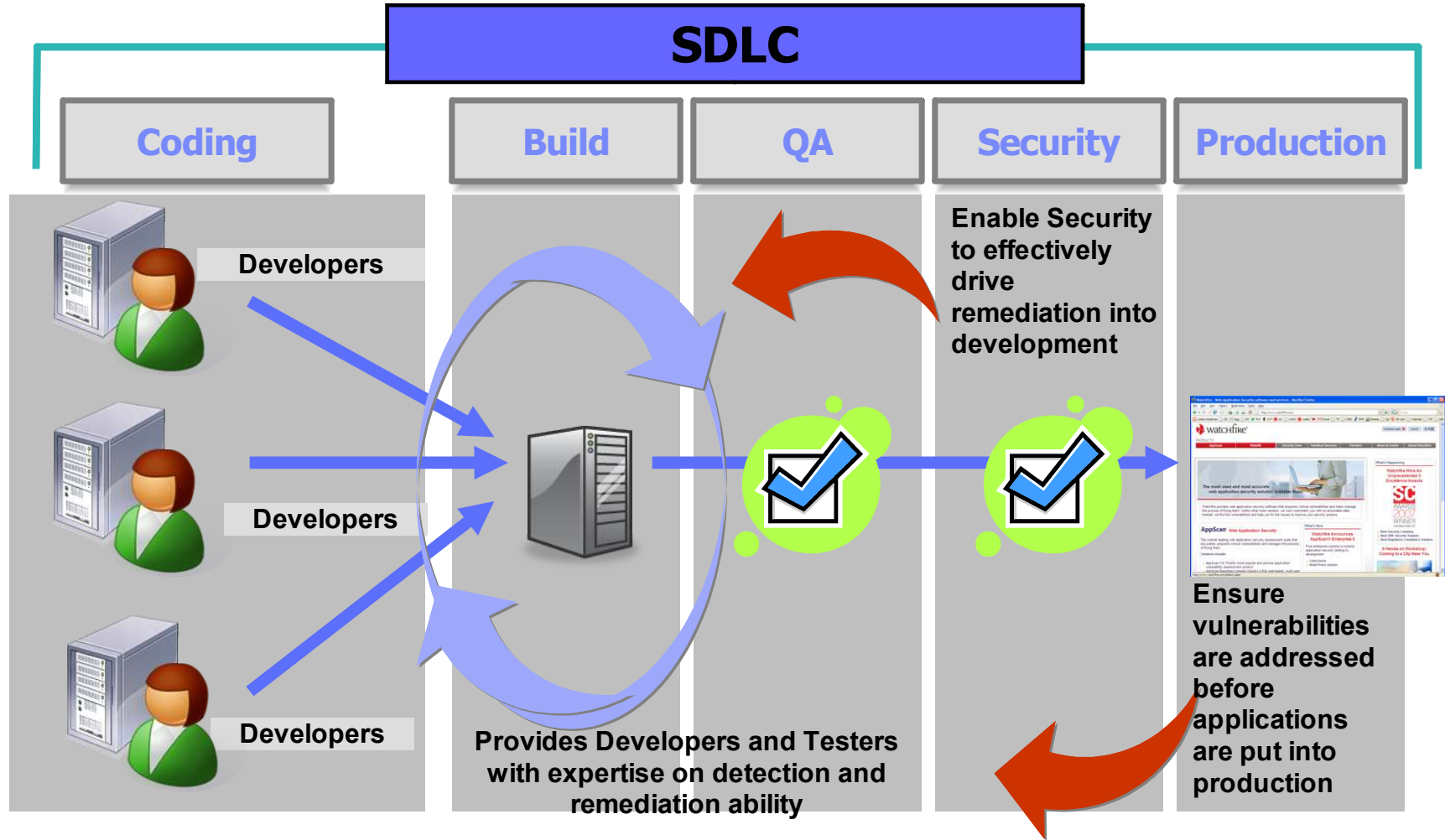


Why Application Security is a High Priority

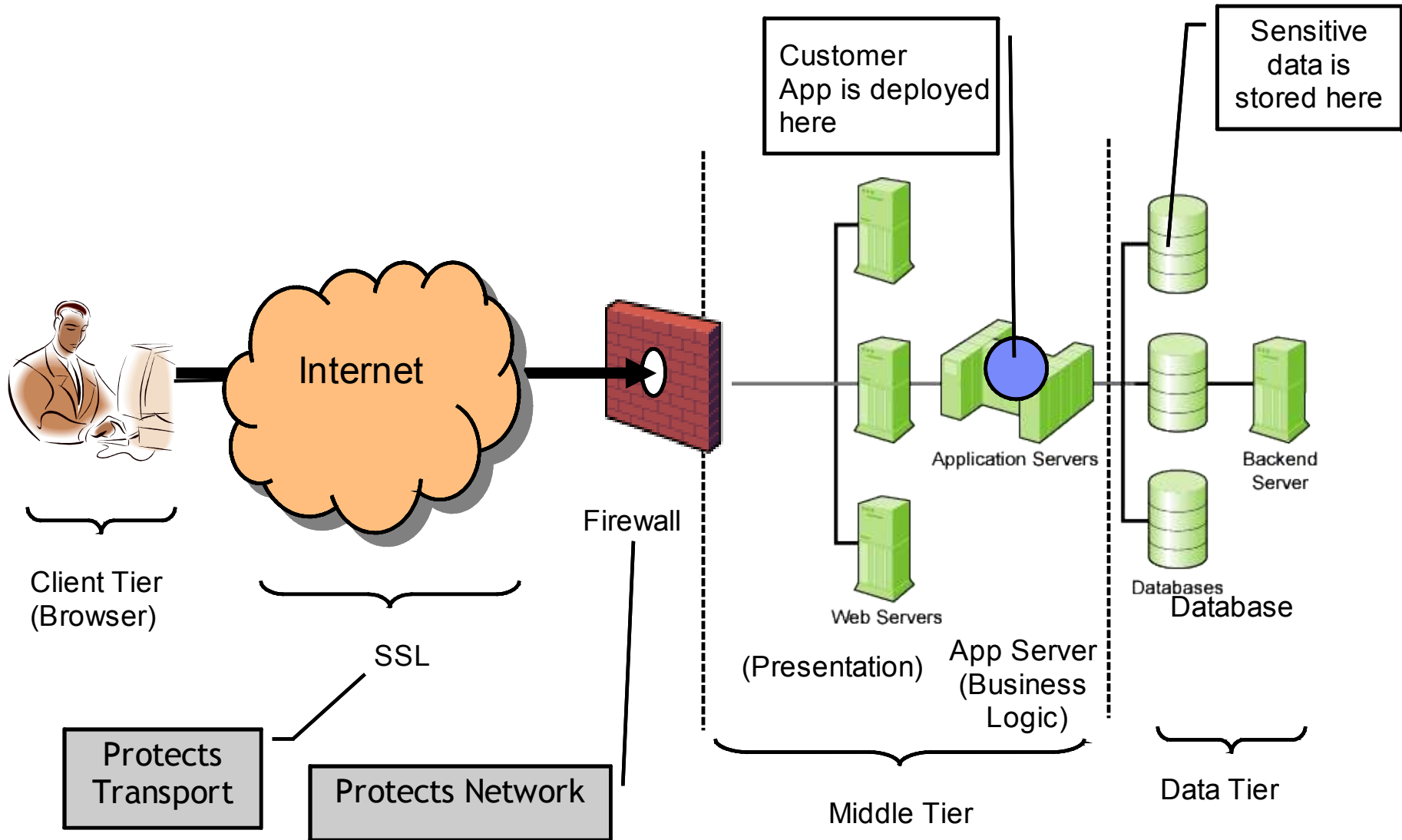
- **Web applications are the #1 focus of hackers:**
 - ▶ 75% of attacks at Application layer (Gartner®)
 - ▶ XSS and SQL Injection are #1 and #2 reported vulnerabilities (Mitre®)
- **Most sites are vulnerable:**
 - ▶ 90% of sites are vulnerable to application attacks (Watchfire®)
 - ▶ 78% percent of easily exploitable vulnerabilities affected Web applications (Symantec™)
 - ▶ 80% of organizations will experience an application security incident by 2010 (Gartner)
- **Web applications are high value targets for hackers:**
 - ▶ Customer data, credit cards, ID theft, fraud, site defacement, etc
- **Compliance requirements:**
 - ▶ Payment Card Industry (PCI) Standards, GLBA, HIPPA, FISMA,



Building Security & Compliance into the Software Development Lifecycle (SDLC)



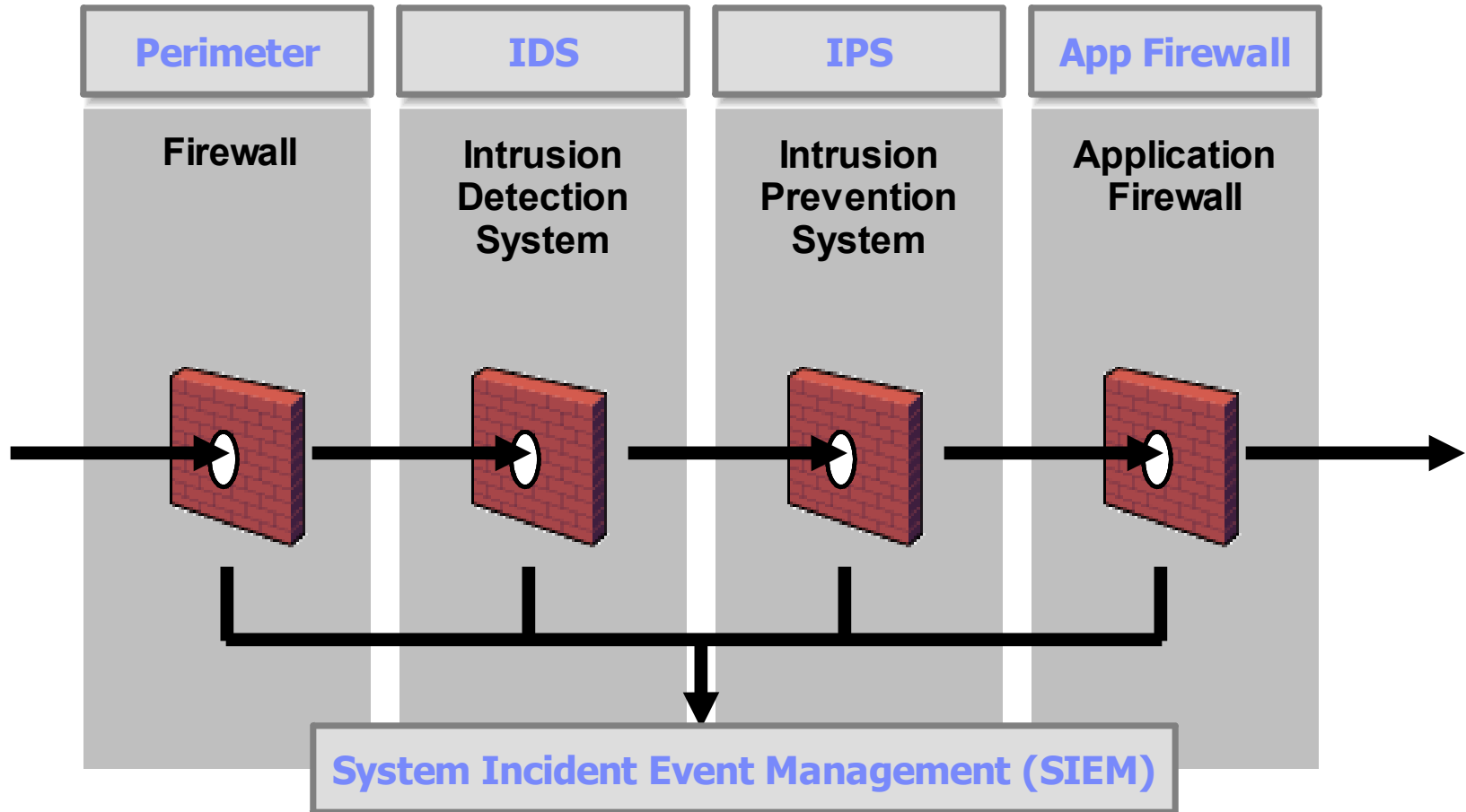
High Level Web Application Architecture Review



Network Defenses for Web Applications



Security



Agenda

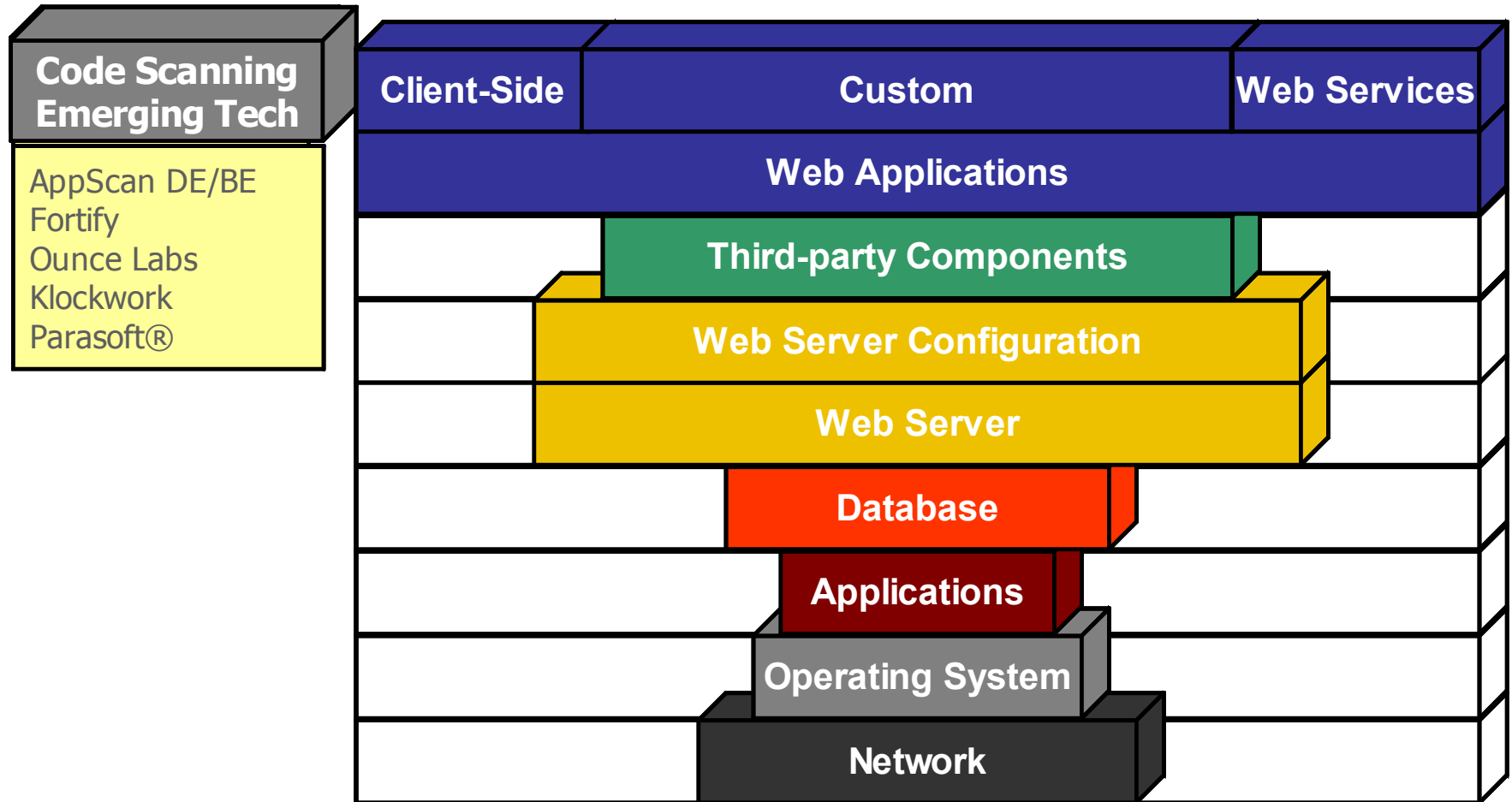
- Introductions & facilities
- Security Landscape
- **Vulnerability Analysis**
 - ▶ Top Attacks Overview
 - ▶ Hands on Lab 1
- Vulnerability Analysis (continued)
 - ▶ Hands on Lab 2
- Automated Vulnerability Analysis
 - ▶ AppScan Overview
 - ▶ Hands on Lab 3





Security

Where are the Vulnerabilities?





The Myth: "Our Site Is Safe"

Security

**We Have Firewalls
in Place**

**We Audit It Once a
Quarter with Pen Testers**

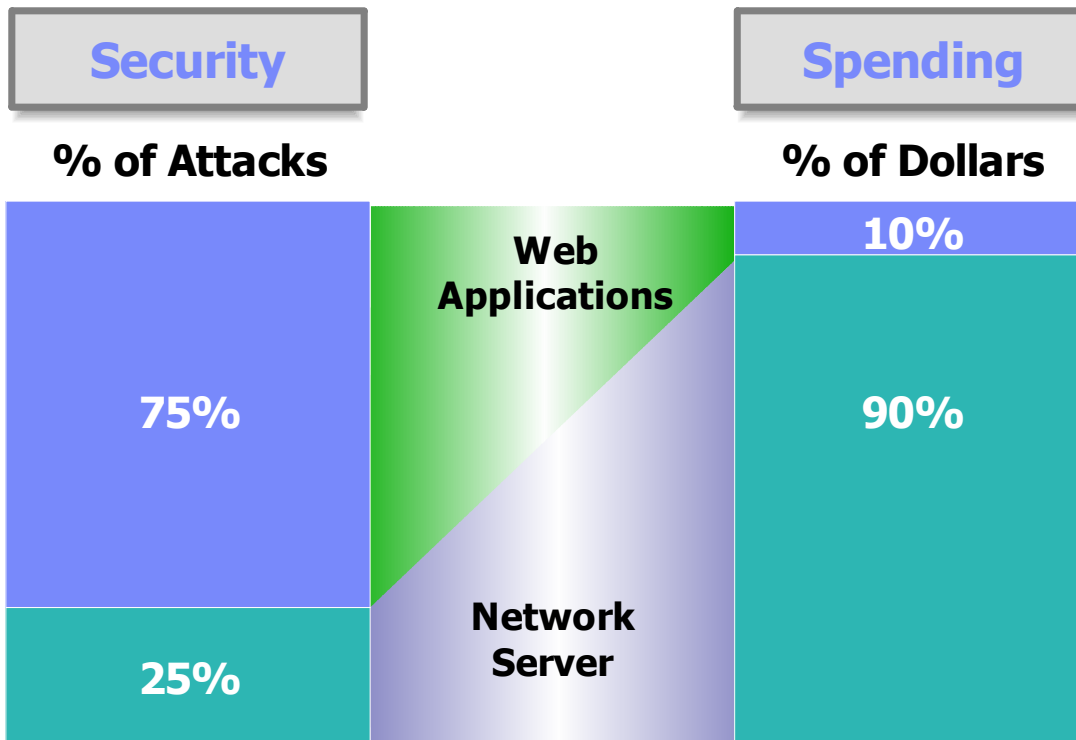
**We Use Network
Vulnerability Scanners**





Security

The Reality: Security and Spending Are Unbalanced

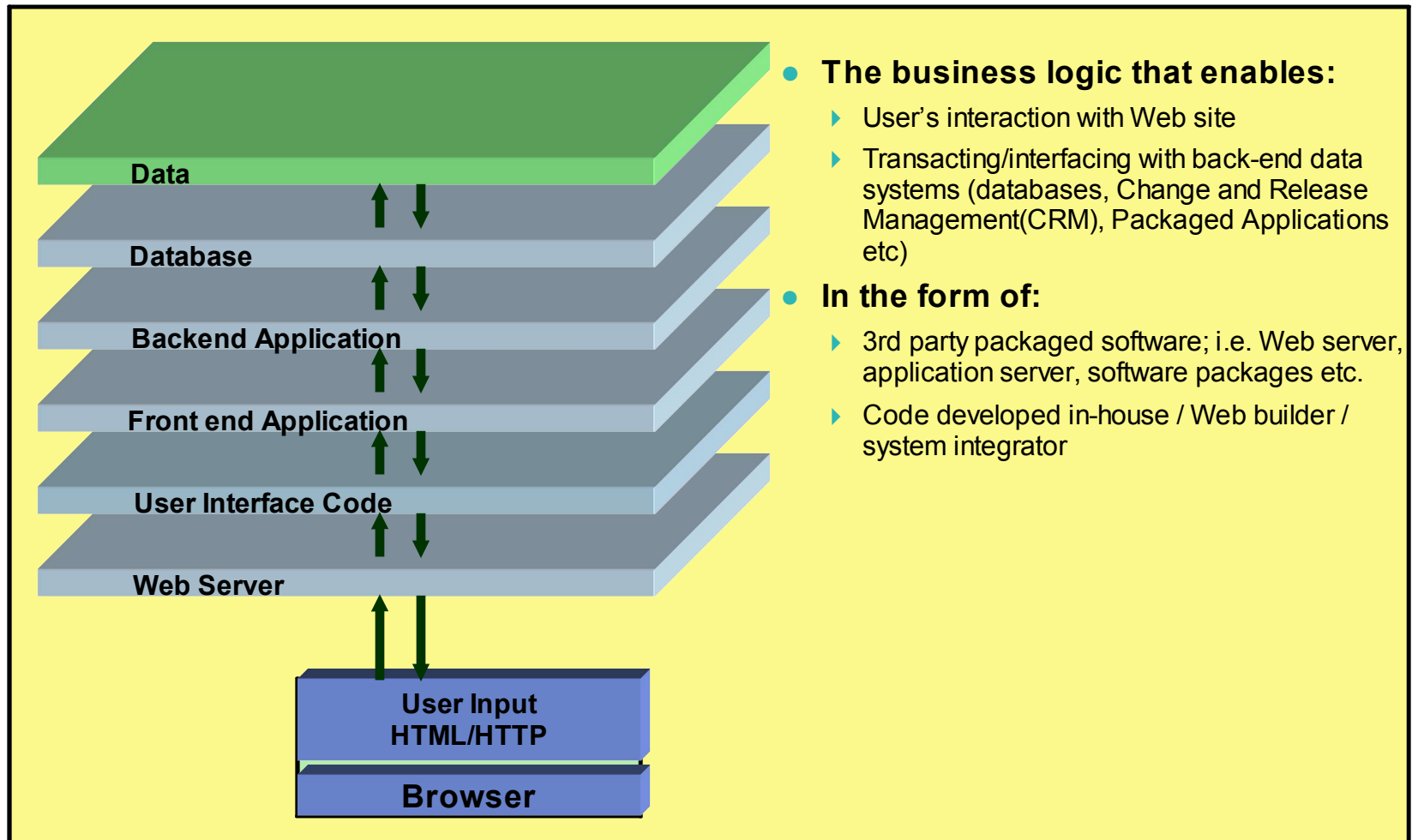


75% of All Attacks on Information Security Are Directed to the Web Application Layer

2/3 of All Web Applications Are Vulnerable

Gartner

What is a Web Application?



Input and Output flow through each layer of the application

A break in any layer breaks the whole application

Security Defects: Those I manage vs. Those I own

	Infrastructure Vulnerabilities or Common Web Vulnerabilities (CWVs)	Application Specific Vulnerabilities (ASVs)
Cause of Defect	Insecure application development by 3 rd party SW	Insecure application development In-house
Location within Application	3 rd party technical building blocks or infrastructure (Web servers,)	Business logic - dynamic data consumed by an application
Type(s) of Exploits	Known vulnerabilities (patches issued), misconfiguration	SQL injection, path tampering, Cross site scripting, Suspect content & cookie poisoning
Detection	Match signatures & check for known misconfigurations.	Requires application specific knowledge
Business Risk	Patch latency primary issue	Requires automatic application lifecycle security
Cost Control	As secure as 3 rd party software	Early detection saves \$\$\$

Open Web Application Security Project (OWASP) and the OWASP Top 10 list

- Open Web Application Security Project (OWASP) – an open organization dedicated to fight insecure software
- “The OWASP Top Ten document represents a broad consensus about what the most critical Web application security flaws are”
- We will use the Top 10 list to cover some of the most common security issues in Web applications



The OWASP Top 10 list

Application Threat	Negative Impact	Example Impact
Cross-Site[®] scripting	Identity Theft, Sensitive Information Leakage, ...	Hackers can impersonate legitimate users, and control their accounts.
Injection Flaws	Attacker can manipulate queries to the DB / LDAP / Other system	Hackers can access backend database information, alter it or steal it.
Malicious File Execution	Execute shell commands on server, up to full control	Site modified to transfer all interactions to the hacker.
Insecure Direct Object Reference	Attacker can access sensitive files and resources	Web application returns contents of sensitive file (instead of harmless one)
Cross-Site Request Forgery	Attacker can invoke "blind" actions on Web applications, impersonating as a trusted user	Blind requests to bank account transfer money to hacker
Information Leakage and Improper Error Handling	Attackers can gain detailed system information	Malicious system reconnaissance may assist in developing further attacks
Broken Authentication & Session Management	Session tokens not guarded or invalidated properly	Hacker can "force" session token on victim; session tokens can be stolen after logout
Insecure Cryptographic Storage	Weak encryption techniques may lead to broken encryption	Confidential information (SSN, Credit Cards) can be decrypted by malicious users
Insecure Communications	Sensitive info sent unencrypted over insecure channel	Unencrypted credentials "sniffed" and used by hacker to impersonate user
Failure to Restrict URL Access	Hacker can access unauthorized resources	Hacker can forcefully browse and access a page past the login page

1. Cross-Site Scripting (XSS)

- What is it?
 - ▶ Malicious script echoed back into HTML returned from a trusted site, and runs under trusted context
- What are the implications?
 - ▶ Session Tokens stolen (browser security circumvented)
 - ▶ Complete page content compromised
 - ▶ Future pages in browser compromised



Demonstration – Cross Site Scripting

- Main points covered in the demo:
 - ▶ Locating an a place where user input which is echoed back to the browser
 - ▶ Seeing if the user input is echoed back 'as-is' or if it is properly encoded
 - ▶ Exploiting the vulnerability

XSS Example I

Browser address bar: `http://www.testfire.net/search.aspx?txtSearch=asdf`

Search bar: `Search [asdf] Go`

Search Results

No results were found for the query:

`asdf`

HTML code:

```
<p>No results were found for the query:<br /><br />
<span id="_ct10_ct10_Content_Main_lblSearch">asdf</span>
```

Footer: [Privacy Policy](#) | [Security Statement](#) | © 2007 Altoro Mutual, Inc.

XSS Example II

Search Results

The page at http://www.testfire.net says:

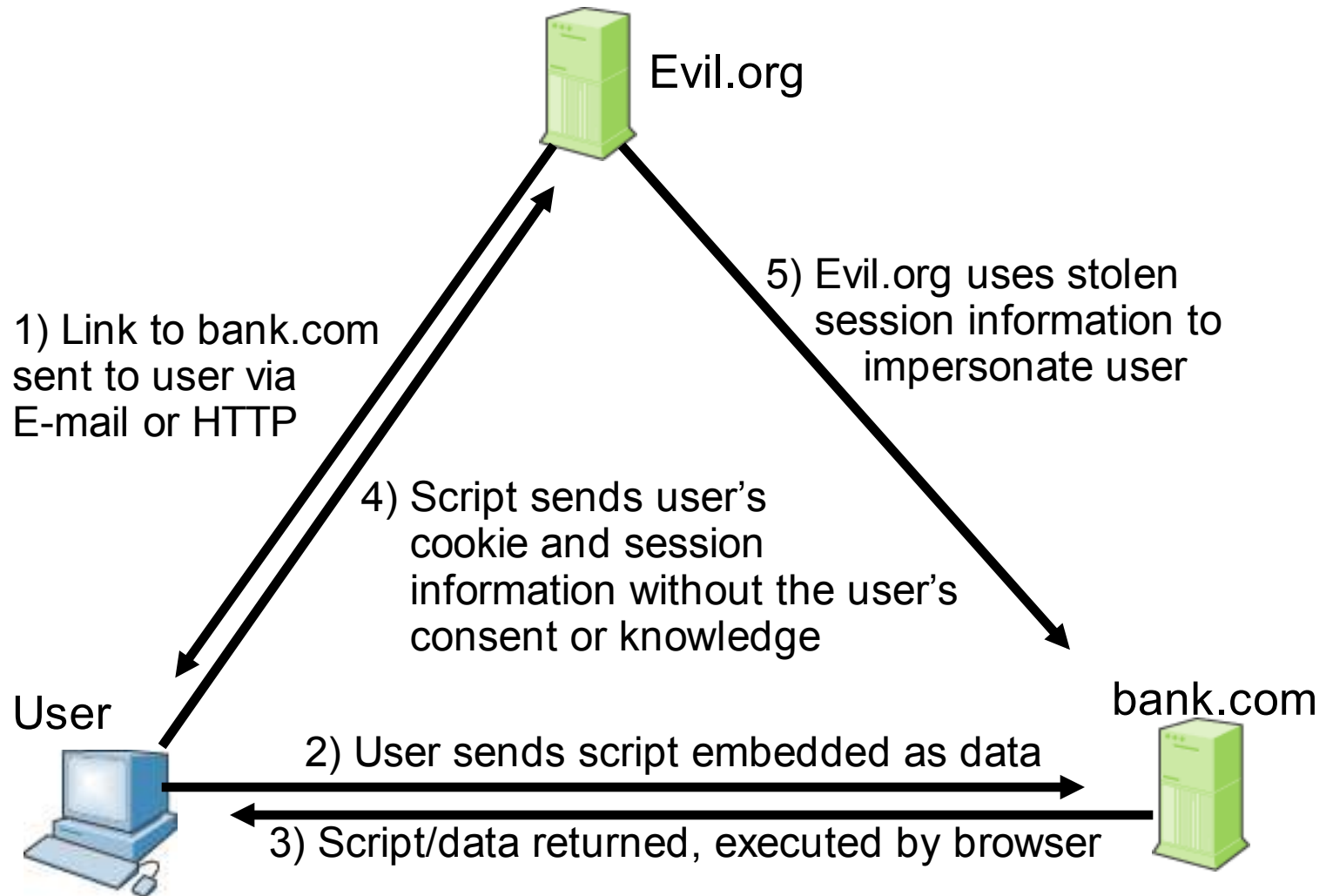
ASP.NET_SessionId=trohqq450cpi5r45rr2pl1fg; amSessionId=1824418181

OK

HTML code:

```
<p>No results were found for the query:<br /><br />
<span id="_ct10_ct10_Content_Main_lblSearch"><script>alert(document.cookie)</script></span>
```

Cross-Site Scripting – The Exploit Process



Agenda

- Introductions & facilities
- Security Landscape
- Vulnerability Analysis
 - ▶ Top Attacks Overview
 - ▶ **Hands on Lab 1**
- Vulnerability Analysis (continued)
 - ▶ Hands on Lab 2
- Automated Vulnerability Analysis
 - ▶ AppScan Overview
 - ▶ Hands on Lab 3

Lab 1 Profile Web Application and XSS

- The Goal of this lab is to:
 - ▶ profile the demo.testfire.net application
 - ▶ utilize a Cross-Site Scripting vulnerability on the demo.testfire.net application in order to access cookies on a target user's browser



Agenda

- Introductions & facilities
- Security Landscape
- Vulnerability Analysis
 - ▶ Top Attacks Overview
 - ▶ Hands on Lab 1
- **Vulnerability Analysis (continued)**
 - ▶ Hands on Lab 2
- Automated Vulnerability Analysis
 - ▶ AppScan Overview
 - ▶ Hands on Lab 3



2 - Injection Flaws

- What is it?
 - ▶ User-supplied data is sent to an interpreter as part of a command, query or data.
- What are the implications?
 - ▶ SQL Injection – Access/modify data in DB
 - ▶ SSI Injection – Execute commands on server and access sensitive data
 - ▶ LDAP Injection – Bypass authentication

SQL Injection

- User input inserted into SQL Command:
 - ▶ Get product details by id:
Select * from products where id='\$REQUEST["id"]';
 - ▶ Hack: send param id with value ' or '1'='1
 - ▶ Resulting executed SQL:
Select * from products where id=' or '1'='1'
 - ▶ All products returned

Demonstration – SQL Injection

- Main points covered in the demo or video:
 - ▶ How to find a SQL injection vulnerability
 - ▶ How to exploit a SQL injection vulnerability



SQL Injection Example I

http://www.testfire.net/bank/login.aspx

Sign In | Contact Us | Feedback | Search Go

Altoro Mutual

DEMO SITE ONLY

ONLINE BANKING LOGIN **PERSONAL** **SMALL BUSINESS** **INSIDE ALTORO MUTUAL**

PERSONAL

- [Deposit Product](#)
- [Checking](#)
- [Loan Products](#)
- [Cards](#)
- [Investments & Insurance](#)
- [Other Services](#)

SMALL BUSINESS

- [Deposit Products](#)
- [Lending Services](#)
- [Cards](#)
- [Insurance](#)
- [Retirement](#)
- [Other Services](#)

INSIDE ALTORO MUTUAL

- [About Us](#)
- [Contact Us](#)
- [Locations](#)
- [Investor Relations](#)
- [Press Room](#)
- [Careers](#)

Online Banking Login

Username:

Password:

Login

[Privacy Policy](#) | [Security Statement](#) | © 2007 Altoro Mutual, Inc.

The Altoro Mutual website is published by Watchfire, Inc. for the sole purpose of demonstrating the effectiveness of Watchfire

SQL Injection Example II

http://www.testfire.net/bank/login.aspx

Sign In | Contact Us | Feedback | Search Go

AltoroMutual

DEMO SITE ONLY

An Error Has Occurred

Summary:

Syntax error (missing operator) in query expression 'username = '' AND password = 'asdf'.

Error Message:

```
System.Data.OleDb.OleDbException: Syntax error (missing operator) in query expression 'username = '' AND password = 'asdf'. at System.Data.OleDb.OleDbCommand.ExecuteNonQueryForSingleResult(tagDBPARAMS dbParams, Object& executeResult) at System.Data.OleDb.OleDbCommand.ExecuteNonQuery(Object& executeResult) at System.Data.OleDb.OleDbCommand.ExecuteNonQuery(CommandBehavior behavior, Object& executeResult) at System.Data.OleDb.OleDbCommand.ExecuteReaderInternal(CommandBehavior behavior, String method) at System.Data.OleDb.OleDbCommand.ExecuteReader(CommandBehavior behavior) at System.Data.OleDb.OleDbCommand.System.Data.IDbCommand.ExecuteReader(CommandBehavior behavior) at System.Data.Common.DbDataAdapter.FillInternal(DataSet dataset, DataTable[] datatables, Int32 startRecord, Int32 maxRecords, String srcTable, IDbCommand command, CommandBehavior behavior) at System.Data.Common.DbDataAdapter.Fill(DataSet dataSet, Int32 startRecord, Int32 maxRecords, String srcTable, IDbCommand command, CommandBehavior behavior) at System.Data.Common.DbDataAdapter.Fill(DataSet dataSet, String srcTable) at Altoro.Authentication.ValidateUser(String uName, String pWord) in d:\downloads\AltoroMutual_v5\website\bank\login.aspx.cs:line 68 at Altoro.Authentication.Page_Load(Object sender, EventArgs e) in d:\downloads\AltoroMutual_v5\website\bank\login.aspx.cs:line 32 at System.Web.Util.CalliHelper.EventArgFunctionCaller(IntPtr fp, Object o, Object t, EventArgs e) at System.Web.Util.CalliEventHandlerDelegateProxy.Callback(Object sender, EventArgs e) at System.Web.UI.Control.OnLoad(EventArgs e) at System.Web.UI.Control.LoadRecursive() at System.Web.UI.Page.ProcessRequestMain(Boolean includeStagesBeforeAsyncPoint, Boolean includeStagesAfterAsyncPoint)
```


SQL Injection Example - Exploit

← → ↻ ⓧ 🏠 🔍 - 📄 ✕

[Sign In](#) | [Contact Us](#) | [Feedback](#) | Search

Altoro Mutual

ONLINE BANKING LOGIN | **PERSONAL** | **SMALL BUSINESS** | **INSIDE ALTORO MUTUAL**

PERSONAL

- [Deposit Product](#)
- [Checking](#)
- [Loan Products](#)
- [Cards](#)
- [Investments & Insurance](#)
- [Other Services](#)

SMALL BUSINESS

- [Deposit Products](#)
- [Lending Services](#)
- [Cards](#)
- [Insurance](#)
- [Retirement](#)
- [Other Services](#)

INSIDE ALTORO MUTUAL

- [About Us](#)
- [Contact Us](#)
- [Locations](#)
- [Investor Relations](#)
- [Press Room](#)
- [Careers](#)

Online Banking Login

Username:

Password:

[Privacy Policy](#) | [Security Statement](#) | © 2007 Altoro Mutual, Inc.

The Altoro Mutual website is published by Watchfire, Inc. for the sole purpose of demonstrating the effectiveness of Watchfire

SQL Injection Example - Outcome

http://www.testfire.net/bank/main.aspx

Sign Off | Contact Us | Feedback | Search Go

Altoro Mutual

DEMO SITE ONLY

MY ACCOUNT | PERSONAL | SMALL BUSINESS | INSIDE ALTORO MUTUAL

I WANT TO ...

- View Account Summary
- View Recent Transactions
- Transfer Funds
- Search News Articles
- Customize Site Language

Hello, John Smith

Welcome to Altoro Mutual Online.

View Account Details:

Congratulations!

You have been pre-approved for an Altoro Gold Visa with a credit limit of \$10000!
Click [Here](#) to apply.

[Privacy Policy](#) | [Security Statement](#) | © 2007 Altoro Mutual, Inc.

The Altoro Mutual website is published by Watchfire, Inc. for the sole purpose of demonstrating the effectiveness of Watchfire products in detecting web application vulnerabilities and website defects. This site is not a real banking site. Similarities, if any, to third party products and/or websites are purely coincidental. This site is provided "as is" without warranty of any kind, either express or implied. Watchfire does not assume any risk in relation to your use of this website. For additional Terms of Use, please go to <http://www.watchfire.com/statements/terms.aspx>.

Copyright © 2007, Watchfire Corporation. All rights reserved.

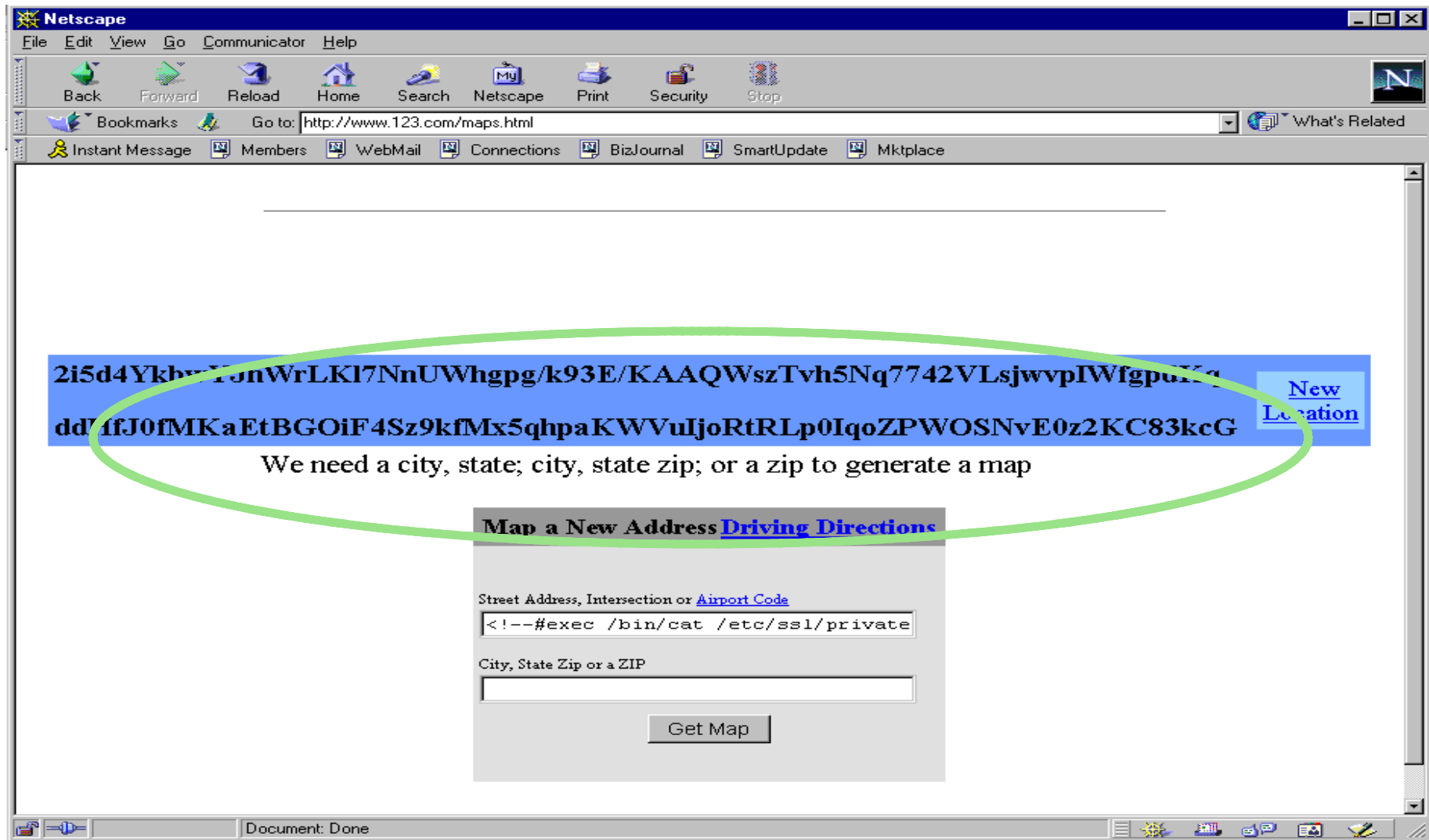
Find:

Injection Flaws (SSI Injection Example)

Creating commands from input

The screenshot shows a Netscape browser window with the address bar containing `http://www.123.com/maps.html`. The page displays a "Maps" section with a "New Location" button. A form titled "Map a New Address Driving Directions" is visible. The form has two input fields: "Street Address, Intersection or [Airport Code](#)" and "City, State, Zip or a ZIP". The first field contains the SSI payload `<!--#exec cat /etc/ssl/private.pem-`. A green oval highlights this payload, and a green arrow points from it to a green box below the form containing the same payload: `<!--#exec cat /etc/ssl/private.pem-`. The browser's status bar at the bottom shows "Document: Done".

The return is the private SSL key of the server



3 - Malicious File Execution

- What is it?
 - ▶ Application tricked into executing commands or creating files on server
- What are the implications?
 - ▶ Command execution on server – complete takeover
 - ▶ Site Defacement, including XSS option



Malicious File Execution – Example I

The screenshot shows a web browser window with the address bar displaying `http://www.testfire.net/feedback.aspx`. The page content includes a navigation menu with sections like 'ONLINE BANKING LOGIN', 'PERSONAL', 'SMALL BUSINESS', and 'INSIDE ALTORO MUTUAL'. A 'Tamper Popup' dialog box is overlaid on the page, showing the details of a request to `http://www.testfire.net/comment.aspx`.

The dialog box contains two tables:

Request Header Name	Request Header Value	Post Parameter Name	Post Parameter Value
Host	www.testfire.net	cfile	comments.txt
User-Agent	Mozilla/5.0 (Windows; U; Windov	name	asdf
Accept	text/xml,application/xml,applicat	email_addr	asdf
Accept-Language	en-us,en;q=0.5	subject	asdf
Accept-Encoding	gzip,deflate	comments	asdf
Accept-Charset	ISO-8859-1,utf-8;q=0.7,*;q=0.	submit	+Submit+
Keep-Alive	300		
Connection	keep-alive		
Referer	http://www.testfire.net/feedbak		
Cookie	ASP.NET_SessionId=adp4vz550		

The 'Post Parameter Value' for 'cfile' is 'comments.txt', which is highlighted with a red box. Below the dialog box, the original form's 'Submit' and 'Clear Form' buttons are visible.

Malicious File Execution – Example cont.

Tamper Popup

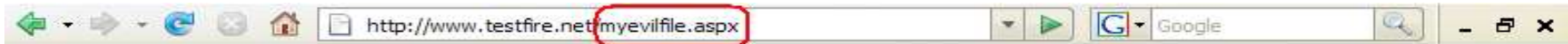
http://www.testfire.net/comment.aspx

Request Header Name	Request Header Value	Post Parameter Name	Post Parameter Value
Host	www.testfire.net	cfile	myevilfile.aspx
User-Agent	Mozilla/5.0 (Windows; U; Windov	name	asdf
Accept	text/xml,application/xml,applicat	email_addr	asdf
Accept-Language	en-us,en;q=0.5	subject	asdf
Accept-Encoding	gzip,deflate	comments	%3C%25%40+Page+Language
Accept-Charset	ISO-8859-1,utf-8;q=0.7,*;q=0.	submit	+Submit+
Keep-Alive	300		
Connection	keep-alive		
Referer	http://www.testfire.net/feedba		
Cookie	amUserInfo=UserName=JyBvciA		

```
<%@ Page Language="C#" %>  
<% Response.Write (System.IO.File.ReadAllText  
("c:/windows/system32/drivers/etc/hosts")); %>
```

OK Cancel

Malicious File Execution – Example cont.



```
asdf, asdf, asdf, # Copyright (c) 1993-1999 Microsoft Corp. # # This is a sample HOSTS file used by Microsoft TCP/IP for  
Windows. # # This file contains the mappings of IP addresses to host names. Each # entry should be kept on an individual line.  
The IP address should # be placed in the first column followed by the corresponding host name. # The IP address and the host  
name should be separated by at least one # space. # # Additionally, comments (such as these) may be inserted on individual #  
lines or following the machine name denoted by a '#' symbol. # # For example: # # 102.54.94.97 rhino.acme.com # source server  
# 38.25.63.10 x.acme.com # x client host 127.0.0.1 localhost
```


4 - Insecure Direct Object Reference

- What is it?
 - ▶ Part or all of a resource (file, table, etc.) name controlled by user input.
- What are the implications?
 - ▶ Access to sensitive resources
 - ▶ Information Leakage, aids future hacks



Insecure Direct Object Reference - Example

The screenshot shows a web browser window with the address bar containing the URL: `://www.testfire.net/default.aspx?content=business_deposit.htm`. The page displays the Altoro Mutual logo and navigation links: [Sign In](#), [Contact Us](#), [Feedback](#), and a search box. The main content area is titled "Deposit Products" and includes a list of services: Commercial Savings Accounts, Commercial Money Market Accounts, Time Deposits, and High Yield Investments. A sidebar on the left provides navigation for PERSONAL, SMALL BUSINESS, and INSIDE ALTORO MUTUAL sections. The footer contains links for Privacy Policy and Security Statement, and a copyright notice for 2007 Altoro Mutual, Inc. A dashed red box at the bottom of the page contains the following text: "The Altoro Mutual website is published by Watchfire, Inc. for the sole purpose of demonstrating the effectiveness of Watchfire".

Insecure Direct Object Reference – Example Cont.

http://www.testfire.net/default.aspx?content=../boot.ini

Sign In | Contact Us | Feedback | Search Go

Altoro Mutual

DEMO SITE ONLY

ONLINE BANKING LOGIN	PERSONAL	SMALL BUSINESS	INSIDE ALTORO MUTUAL
<p><u>PERSONAL</u></p> <ul style="list-style-type: none">• Deposit Product• Checking• Loan Products• Cards• Investments & Insurance• Other Services <p><u>SMALL BUSINESS</u></p> <ul style="list-style-type: none">• Deposit Products• Lending Services• Cards• Insurance• Retirement• Other Services <p><u>INSIDE ALTORO MUTUAL</u></p> <ul style="list-style-type: none">• About Us• Contact Us• Locations• Investor Relations• Press Room• Careers	Error! File must be of type txt or htm		

[Privacy Policy](#) | [Security Statement](#) | © 2007 Altoro Mutual, Inc.

The Altoro Mutual website is published by Watchfire, Inc. for the sole purpose of demonstrating the effectiveness of Watchfire

Insecure Direct Object Reference – Example Cont.

The screenshot shows a web browser window with the URL `http://www.testfire.net/default.aspx?content=../boot.ini%00.htm`. The page displays the Altoro Mutual logo and navigation menu. The main content area shows the following text:

```
[boot loader]timeout=30default=multi(0)disk(0)rdisk(0)partition(1)\WINDOWS[operating
systems]multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Microsoft Windows XP Professional"
/noexecute=optin /fastdetect
```

The text is highlighted with a red box, indicating a successful directory traversal attack that accessed a sensitive system file.

At the bottom of the page, there is a footer with the following text:

Privacy Policy | Security Statement | © 2007 Altoro Mutual, Inc.

The Altoro Mutual website is published by Watchfire, Inc. for the sole purpose of demonstrating the effectiveness of Watchfire

5 - Information Leakage and Improper Error Handling

- What is it?
 - ▶ Unneeded information made available via errors or other means.
- What are the implications?
 - ▶ Sensitive data exposed
 - ▶ Web App internals and logic exposed (source code, SQL syntax, exception call stacks, etc.)
 - ▶ Information aids in further hacks



Information Leakage - Example

The screenshot shows a web browser window displaying the Altoro Mutual website. The URL in the address bar is `http://www.testfire.net/bank/login.aspx`. The page features the Altoro Mutual logo and navigation links such as "Sign In", "Contact Us", and "Feedback". A "DEMO SITE ONLY" banner is visible in the top right. The main content area is titled "Online Banking Login" and contains input fields for "Username:" and "Password:", along with a "Login" button. A red box highlights a block of HTML source code, which includes a comment about contacting SiteOps and a span element with an ID of "ct10".

HTML Source Code (highlighted):

```
<h1>Online Banking Login</h1>
<!-- To get the latest admin login, please contact SiteOps at 415-555-6159 -->
<p><span id=" ct10| ct10 Content Main message"
```

At the bottom of the page, there is a footer with links for "Privacy Policy" and "Security Statement", and a copyright notice for "© 2007 Altoro Mutual, Inc.". A dashed box at the very bottom contains the text: "The Altoro Mutual website is published by Watchfire, Inc. for the sole purpose of demonstrating the effectiveness of Watchfire".

Improper Error Handling - Example

← → ↻ 🏠 🔍 Google

[Sign In](#) | [Contact Us](#) | [Feedback](#) | Search

AltoroMutual

DEMO SITE ONLY

An Error Has Occurred

Summary:

Syntax error (missing operator) in query expression 'username = '' AND password = 'asdf''.

Error Message:

```
System.Data.OleDb.OleDbException: Syntax error (missing operator) in query expression 'username = '' AND password = 'asdf''. at System.Data.OleDb.OleDbCommand.ExecuteNonQueryForSingleResult(tagDBPARAMS dbParams, Object& executeResult) at System.Data.OleDb.OleDbCommand.ExecuteNonQuery(Object& executeResult) at System.Data.OleDb.OleDbCommand.ExecuteNonQuery(CommandBehavior behavior, Object& executeResult) at System.Data.OleDb.OleDbCommand.ExecuteReaderInternal(CommandBehavior behavior, String method) at System.Data.OleDb.OleDbCommand.ExecuteReader(CommandBehavior behavior) at System.Data.OleDb.OleDbCommand.System.Data.IDbCommand.ExecuteReader(CommandBehavior behavior) at System.Data.Common.DbDataAdapter.FillInternal(DataSet dataset, DataTable[] datatables, Int32 startRecord, Int32 maxRecords, String srcTable, IDbCommand command, CommandBehavior behavior) at System.Data.Common.DbDataAdapter.Fill(DataSet dataSet, Int32 startRecord, Int32 maxRecords, String srcTable, IDbCommand command, CommandBehavior behavior) at System.Data.Common.DbDataAdapter.Fill(DataSet dataSet, String srcTable) at Altoro.Authentication.ValidateUser(String uName, String pWord) in d:\downloads\AltoroMutual_v5\website\bank\login.aspx.cs:line 68 at Altoro.Authentication.Page_Load(Object sender, EventArgs e) in d:\downloads\AltoroMutual_v5\website\bank\login.aspx.cs:line 32 at System.Web.Util.CalliHelper.EventArgFunctionCaller(IntPtr fp, Object o, Object t, EventArgs e) at System.Web.Util.CalliEventHandlerDelegateProxy.Callback(Object sender, EventArgs e) at System.Web.UI.Control.OnLoad(EventArgs e) at System.Web.UI.Control.LoadRecursive() at System.Web.UI.Page.ProcessRequestMain(Boolean includeStagesBeforeAsyncPoint, Boolean includeStagesAfterAsyncPoint)
```

Information Leakage – Different User/Pass Error

 ONLINE BANKING LOGIN	PERSONAL	SMALL BUSINESS	INSIDE ALTORO MUTUA
<p><u>PERSONAL</u></p> <ul style="list-style-type: none">• Deposit Product• Checking• Loan Products• Cards• Investments & Insurance• Other Services <p><u>SMALL BUSINESS</u></p> <ul style="list-style-type: none">• Deposit Products	<h2>Online Banking Login</h2> <p>Login Failed - Invalid Password</p> <p>Username: <input type="text" value="jsmith"/></p> <p>Password: <input type="password"/></p> <p><input type="button" value="Login"/></p>		

 ONLINE BANKING LOGIN	PERSONAL	SMALL BUSINESS	INSIDE ALTORO MUTUA
<p><u>PERSONAL</u></p> <ul style="list-style-type: none">• Deposit Product• Checking• Loan Products• Cards• Investments & Insurance• Other Services <p><u>SMALL BUSINESS</u></p> <ul style="list-style-type: none">• Deposit Products	<h2>Online Banking Login</h2> <p>Login Failed - Invalid Username</p> <p>Username: <input type="text" value="nouser"/></p> <p>Password: <input type="password"/></p> <p><input type="button" value="Login"/></p>		

6 - Failure to Restrict URL Access

- What is it?
 - ▶ Resources that should only be available to authorized users can be accessed by forcefully browsing them
- What are the implications?
 - ▶ Sensitive information leaked/modified
 - ▶ Admin privileges made available to hacker



Failure to Restrict URL Access - Admin User login

ONLINE BANKING LOGIN PERSONAL SMALL BUSINESS

PERSONAL

- [Deposit Product](#)
- [Checking](#)
- [Loan Products](#)
- [Cards](#)
- [Investments & Insurance](#)
- [Other Services](#)

SMALL BUSINESS

Online Banking Login

Username:

Password:

MY ACCOUNT PERSONAL SMALL BUSINESS

I WANT TO ...

- [View Account Summary](#)
- [View Recent Transactions](#)
- [Transfer Funds](#)
- [Search News Articles](#)
- [Customize Site Language](#)

ADMINISTRATION

- [View Application Values](#)
- [Edit Users](#)

Hello, Admin User

Welcome to Altoro Mutual Online.

View Account Details:

</admin/admin.aspx>

Simple user logs in, forcefully browses to admin page

ONLINE BANKING LOGIN **PERSONAL** **SMALL BUSINESS**

PERSONAL

- [Deposit Product](#)
- [Checking](#)
- [Loan Products](#)
- [Cards](#)
- [Investments & Insurance](#)
- [Other Services](#)

Online Banking Login

Username:

Password:

http://www.testfire.net/admin/admin.aspx

AltoroMutual [Sign Off](#) | [Contact Us](#) | [Feedback](#) | Search

MY ACCOUNT **PERSONAL** **SMALL BUSINESS** **INSIDE ALTORO MUTUAL**

I WANT TO ...

- [View Application Values](#)
- [Edit Users](#)

Edit User Information

Add an account to an existing user.

Users: Account Types:

Failure to Restrict URL Access: Privilege Escalation Types

- Access given to completely restricted resources
 - ▶ Accessing files that shouldn't be served (*.bak, "Copy Of", *.inc, *.cs, ws_ftp.log, etc.)
- Vertical Privilege Escalation
 - ▶ Unknown user accessing pages past login page
 - ▶ Simple user accessing admin pages
- Horizontal Privilege Escalation
 - ▶ User accessing other user's pages
 - ▶ Example: Bank account user accessing another's

Agenda

- Introductions & facilities
- Security Landscape
- Vulnerability Analysis
 - ▶ Top Attacks Overview
 - ▶ Hands on Lab 1
- Vulnerability Analysis (continued)
 - ▶ **Hands on Lab 2**
- Automated Vulnerability Analysis
 - ▶ AppScan Overview
 - ▶ Hands on Lab 3

Lab 2

Lab 1 – Profile Web Application, Steal Cookies

Lab 2 – Login without Credentials, Steal Usernames and Passwords, Logging into the Administrative Portal

Lab 3 – Automated Scan of Website

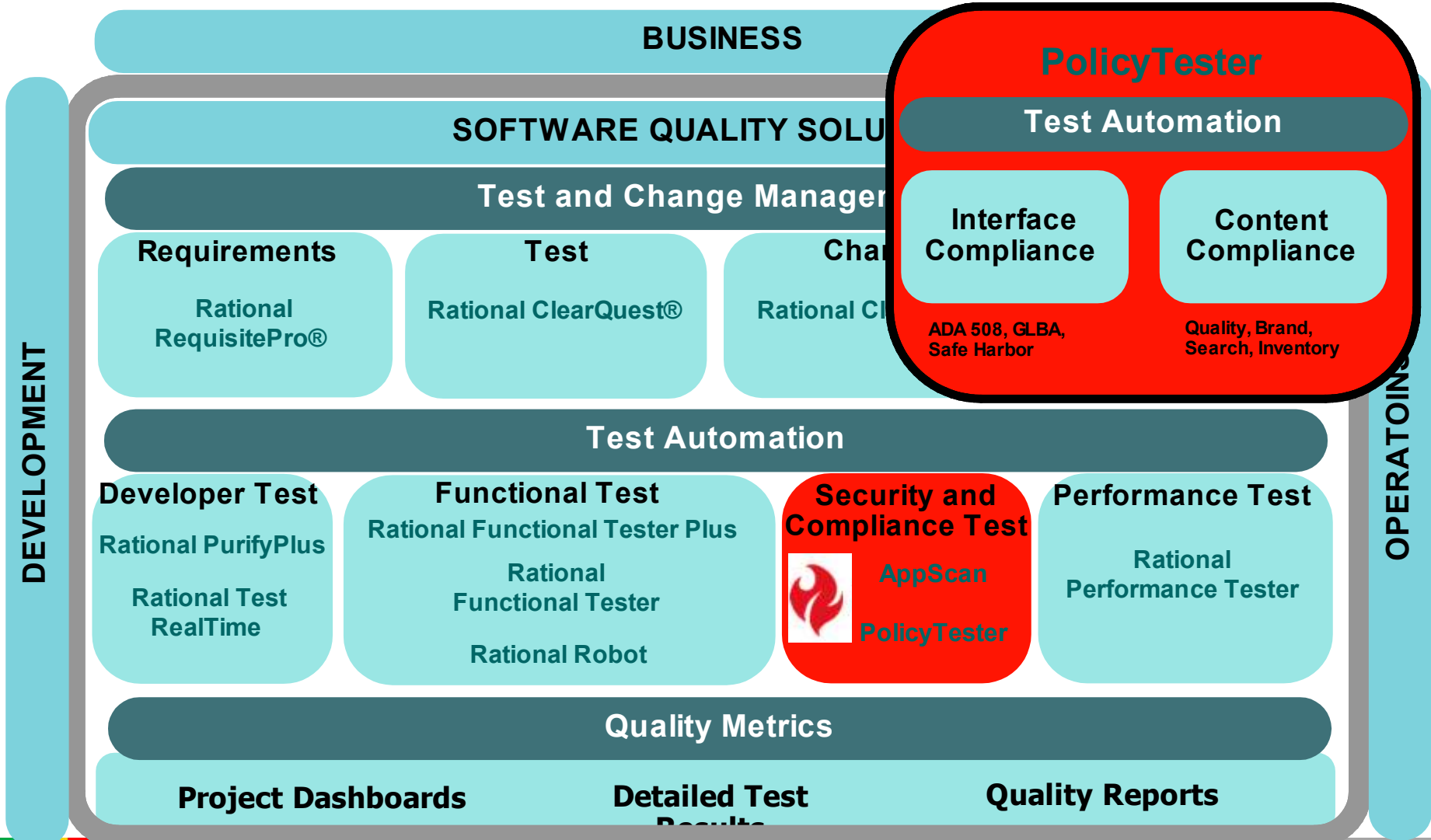




Agenda

- Introductions & facilities
- Security Landscape
- Vulnerability Analysis
 - ▶ Top Attacks Overview
 - ▶ Hands on Lab 1
- Vulnerability Analysis (continued)
 - ▶ Hands on Lab 2
- **Automated Vulnerability Analysis**
 - ▶ AppScan Overview
 - ▶ Hands on Lab 3

Watchfire in the Rational Portfolio

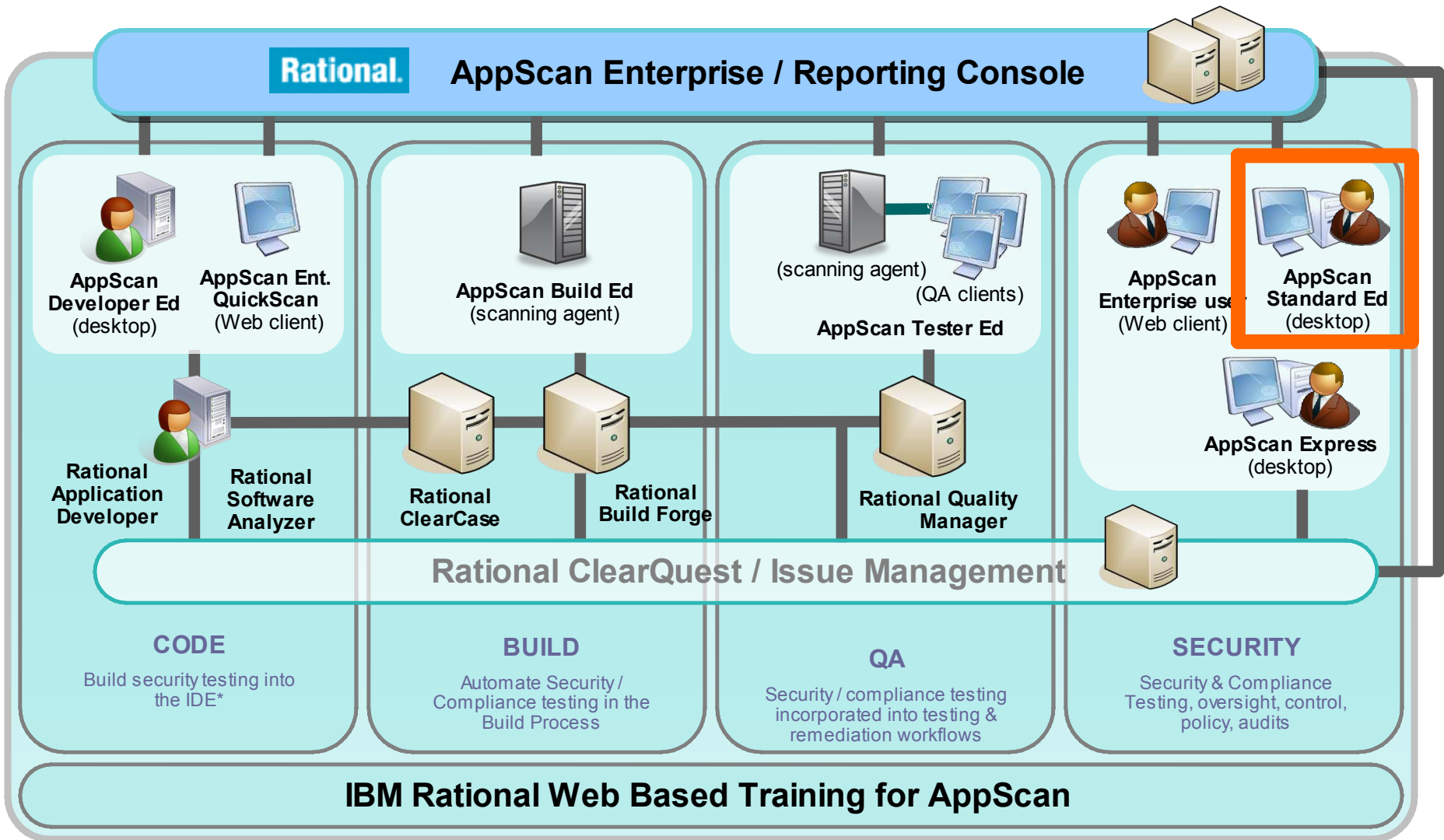


AppScan

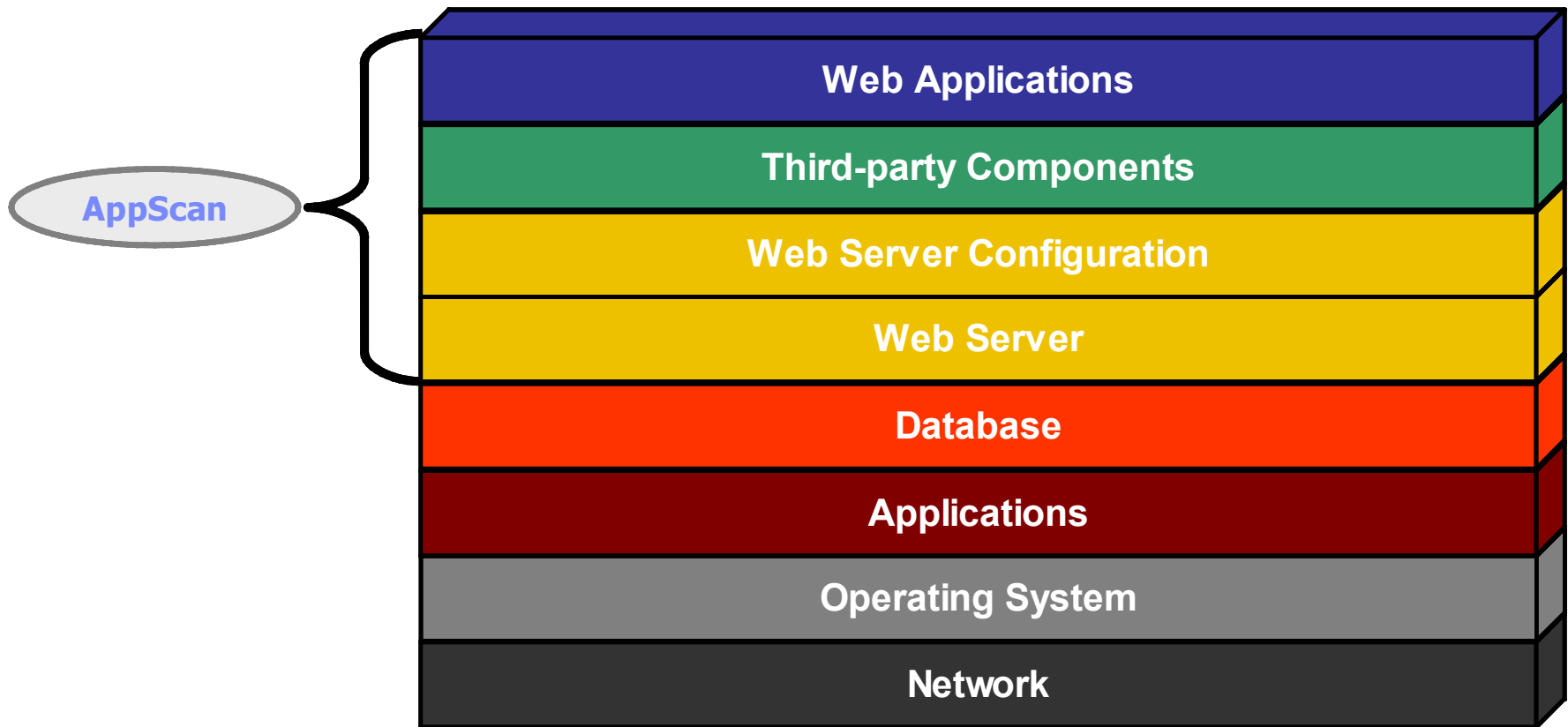
- What is it?
 - ▶ AppScan is an automated tool used to perform vulnerability assessments on Web Applications
- Why do I need it?
 - ▶ To simplify finding and fixing Web application security problems
- What does it do?
 - ▶ Scans Web applications, finds security issues and reports on them in an actionable fashion
- Who uses it?
 - ▶ Security Auditors – main users today
 - ▶ QA engineers – when the auditors become the bottle neck
 - ▶ Developers – to find issues as early as possible (most efficient)



IBM Rational AppScan Ecosystem

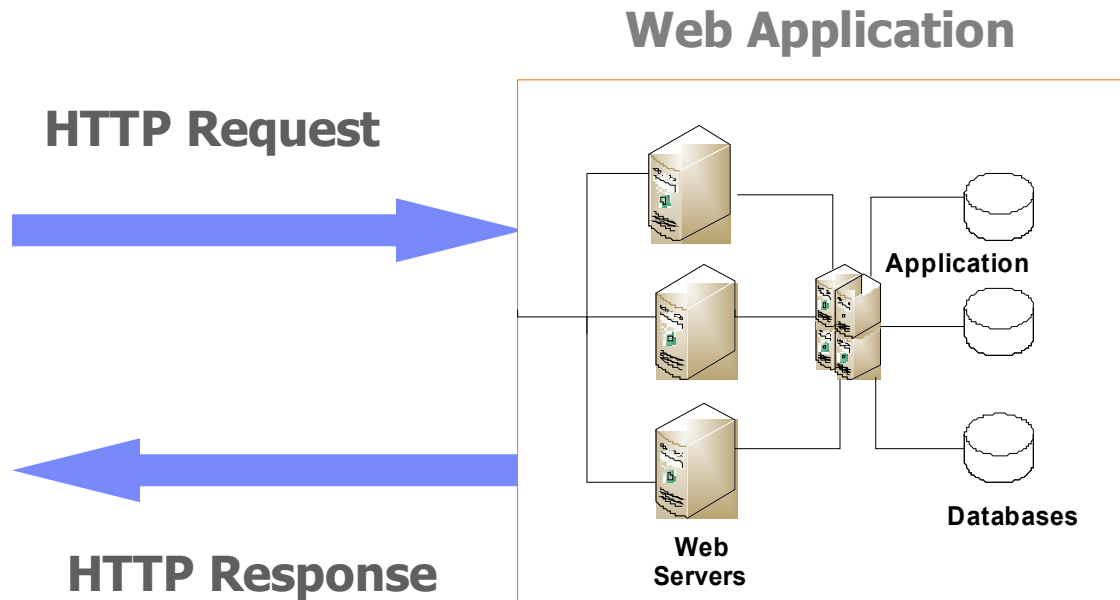


What does AppScan test for?

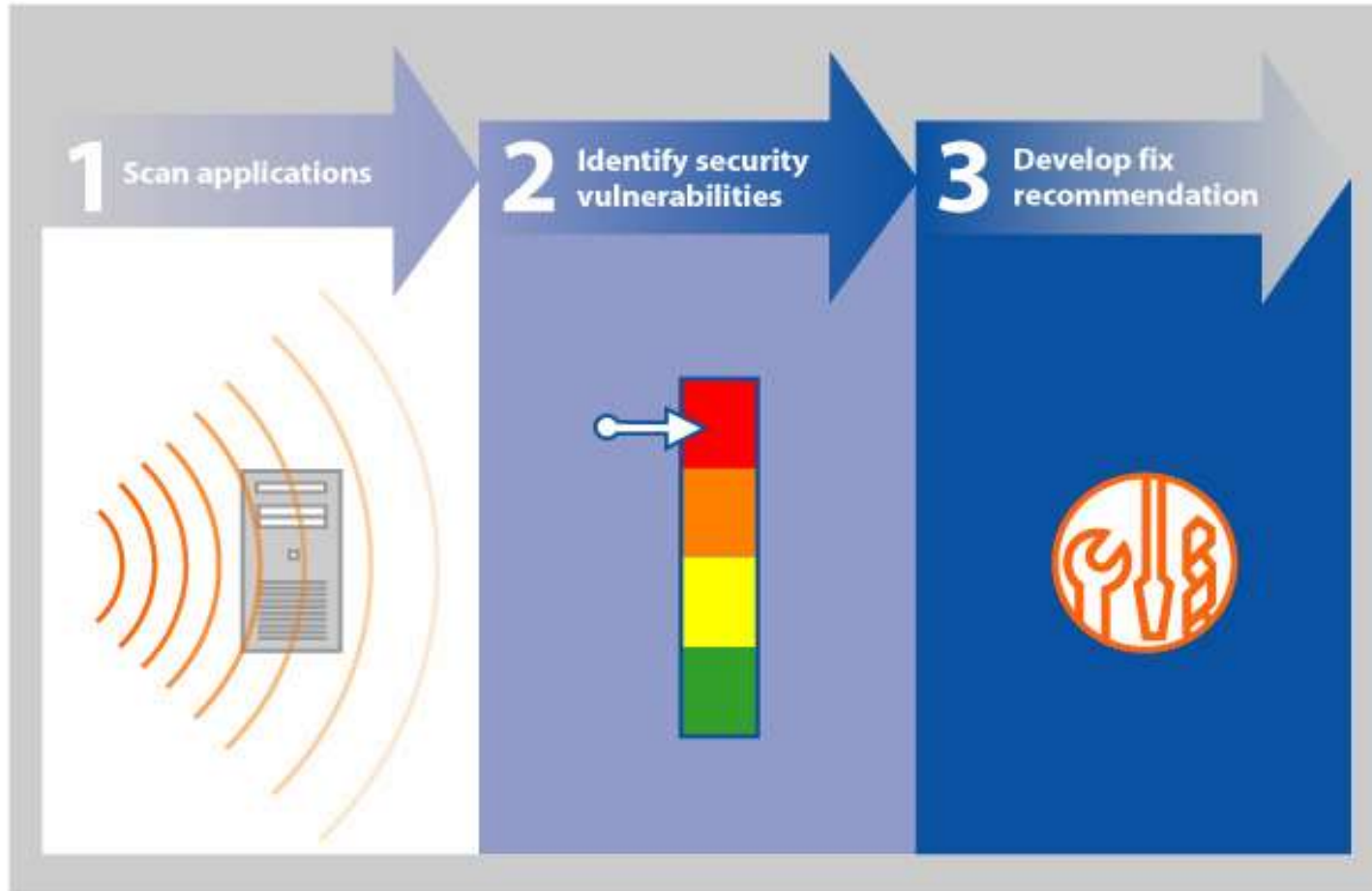


How does AppScan work?

- Approaches an application as a black-box
- Traverses a Web application and builds the site model
- Determines the attack vectors based on the selected Test policy
- Tests by sending modified HTTP requests to the application and examining the HTTP response according to validate rules



AppScan Goes Beyond Pointing out Problems



Actionable Fix Recommendations

The screenshot displays the IBM AppScan 7.5 interface. The main window shows a scan of 'My Application' with 53 security issues. The issues are arranged by severity, with the highest on top. The top issue is 'Blind SQL Injection' (4), which is expanded to show four specific instances on different URLs.

Blind SQL Injection (4)

- http://demo.testfire.net/bank/account.aspx (1)
- http://demo.testfire.net/bank/login.aspx (2)
- http://demo.testfire.net/bank/transaction.aspx (1)

The detailed view for 'Blind SQL Injection' shows a 'Fix Recommendation' section. Under the 'General' heading, it states: 'There are several issues whose remediation lies in sanitizing user input. By verifying that user input does not contain hazardous characters, it is possible to prevent malicious users from causing your application to execute unintended operations, such as launch arbitrary SQL queries, embed Javascript code to be executed on the client side, run various operating system commands etc.'

It is advised to filter out all the following characters:

- [1] | (pipe sign)
- [2] & (ampersand sign)
- [3] ; (semicolon sign)

The status bar at the bottom indicates 53 Security Issues, 18 Critical, 4 High, 22 Medium, and 9 Low severity issues.

AppScan with QA Defect Logger for ClearQuest

The screenshot shows the Watchfire AppScan interface. The main window displays a scan of 'AS7.5 Demo Scan 1.scan'. The left sidebar shows navigation options like 'Security Issues', 'Remediation Tasks', and 'Application Data'. The main area shows a list of security issues, with 'Cross-Site Scripting' selected. A context menu is open over this issue, with 'Log Defect to ClearQuest' highlighted. An arrow points from this menu item to a 'Defect Details' dialog box. The dialog box shows fields for Credentials (Username: admin, Password: [redacted]), Defect Details (Summary: SQL Injection in http://revelation/acmehackme/bank/login.aspx (Parameter passw)), and a dropdown for Priority (1-Critical, 2-Give High Attention, 3-Normal Queue, 4-Low Priority). The dialog also includes fields for Project, State, Keywords, Symptoms, and Owner. At the bottom, there are buttons for 'Cancel' and 'Log Defect'.

Lab 3 overview

- The goal of this lab is to use AppScan in order to automate the detection of vulnerabilities within a Web application





| IBM Software Group

Session summary

An IBM Proof of Technology



Session summary

- Understand the Web application environment
- Understand and differentiate between network and application level vulnerabilities
- Understand where the vulnerabilities exist
- Hands on exercises to understand types of vulnerabilities
- Hands on exercise to leverage automated scan for vulnerabilities



Questions



Next steps

- We can schedule a Vulnerability Assessment of one our your Applications -



Reference materials

- IBM.com

- ▶ <http://www-306.ibm.com/software/rational/welcome/watchfire/products.html>

© Copyright IBM Corporation 2009. All rights reserved.

The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, these materials. Nothing contained in these materials is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. This information is based on current IBM product plans and strategy, which are subject to change by IBM without notice. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way.

IBM, the IBM logo, and other IBM products and services are trademarks of the International Business Machines Corporation, in the United States, other countries or both. Other company, product, or service names may be trademarks or service marks of others.





Thank You

We appreciate your feedback.
Please fill out the survey form in
order to improve this educational
event.