Rational. software

Watchfire, an IBM Company: an introduction to web application security

Presented by:
Paul Murray
Sr. Technical Specialist
pmurray@uk.ibm.com

IBM Rational Software Development Conference 2007

What keeps me **Rational**?

9 October – Johannesburg & 11 October – Cape Town

Agenda

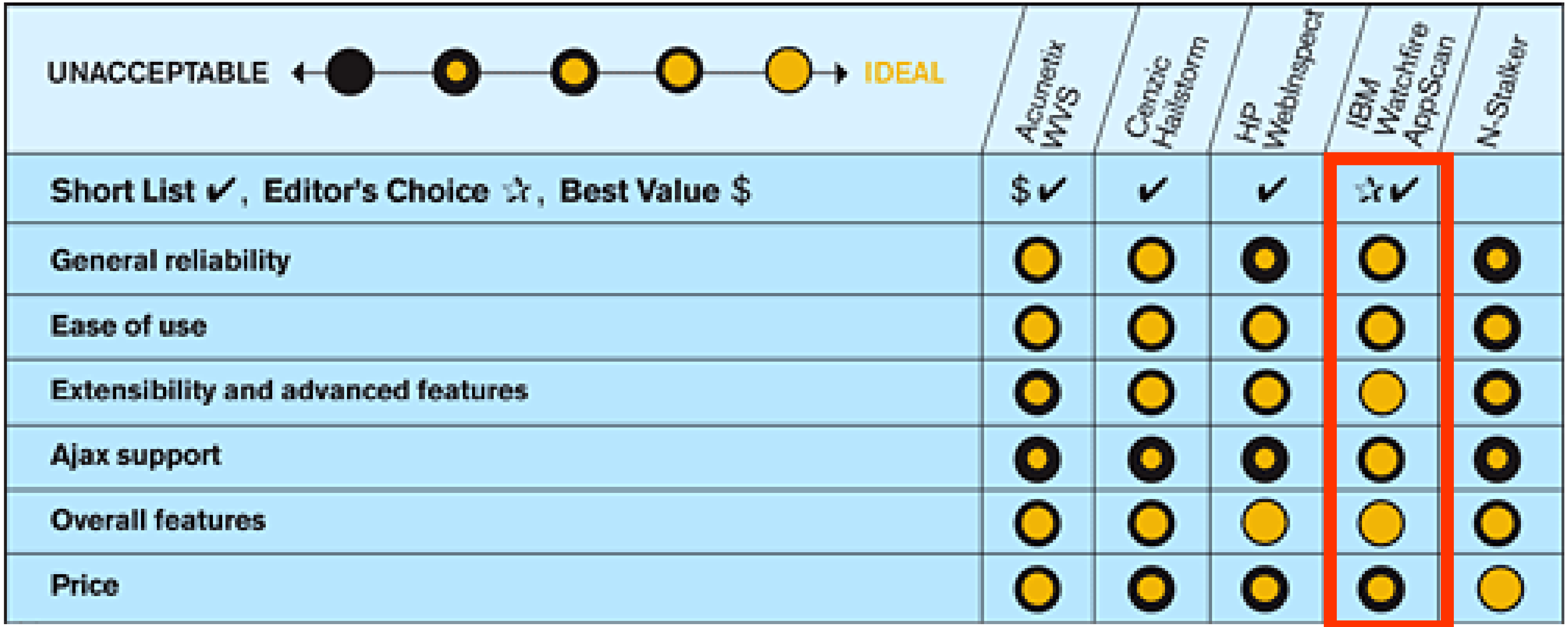
- Rational AppScan
- The Security Landscape
- An Overview of Security Vulnerabilities

What is Rational AppScan

- AppScan helps ensure the security and compliance of Web applications throughout the software development lifecycle.
- AppScan provides complete vulnerability scanning including the latest Web 2.0 technologies: Flash, advanced JavaScript, AJAX support

IBM Rational AppScan

- Recent Review of Web Scanning tools done be Information Week
- Rational AppScan is the editors choice
  - *http://ibmforums.ibm.com/forums/thread.jspa?threadID=354942*



UNACCEPTABLE ← ● — ◕ — ◑ — ◔ — ○ → IDEAL

Short List ✓, Editor's Choice ☆, Best Value $	Acunetix WVS	Cenzic Hailstorm	HP WebInspect	IBM Watchfire AppScan	N-Stalker
	$ ✓	✓	✓	☆ ✓	
General reliability	◔	◔	◕	◔	◕
Ease of use	◔	◔	◔	◔	◑
Extensibility and advanced features	◑	◔	◔	○	◑
Ajax support	◕	◕	◕	◔	◕
Overall features	◔	◔	○	○	◔
Price	◔	◑	◑	◑	○

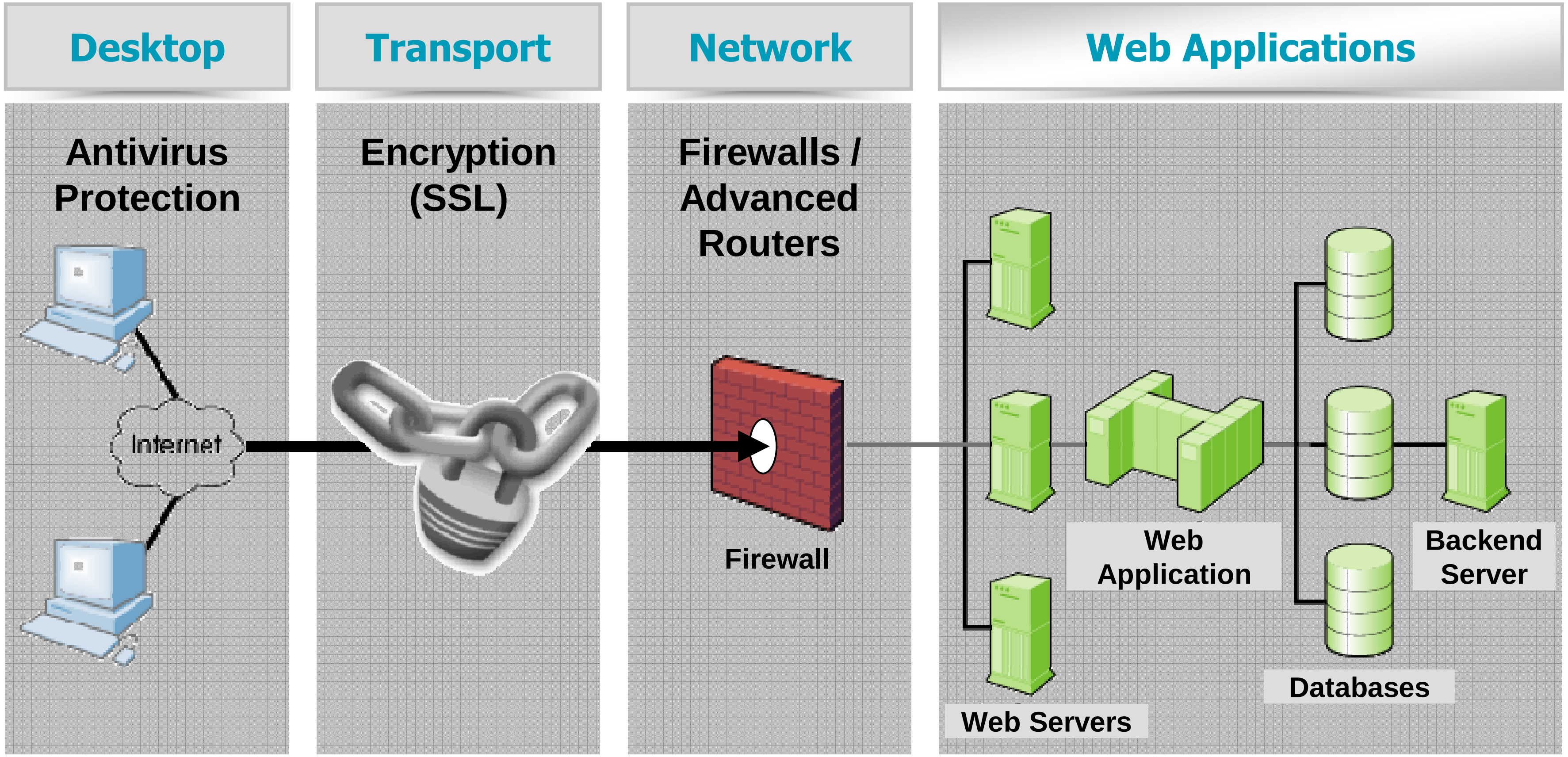
Web Application Security Landscape

IBM Rational Software Development Conference 2007

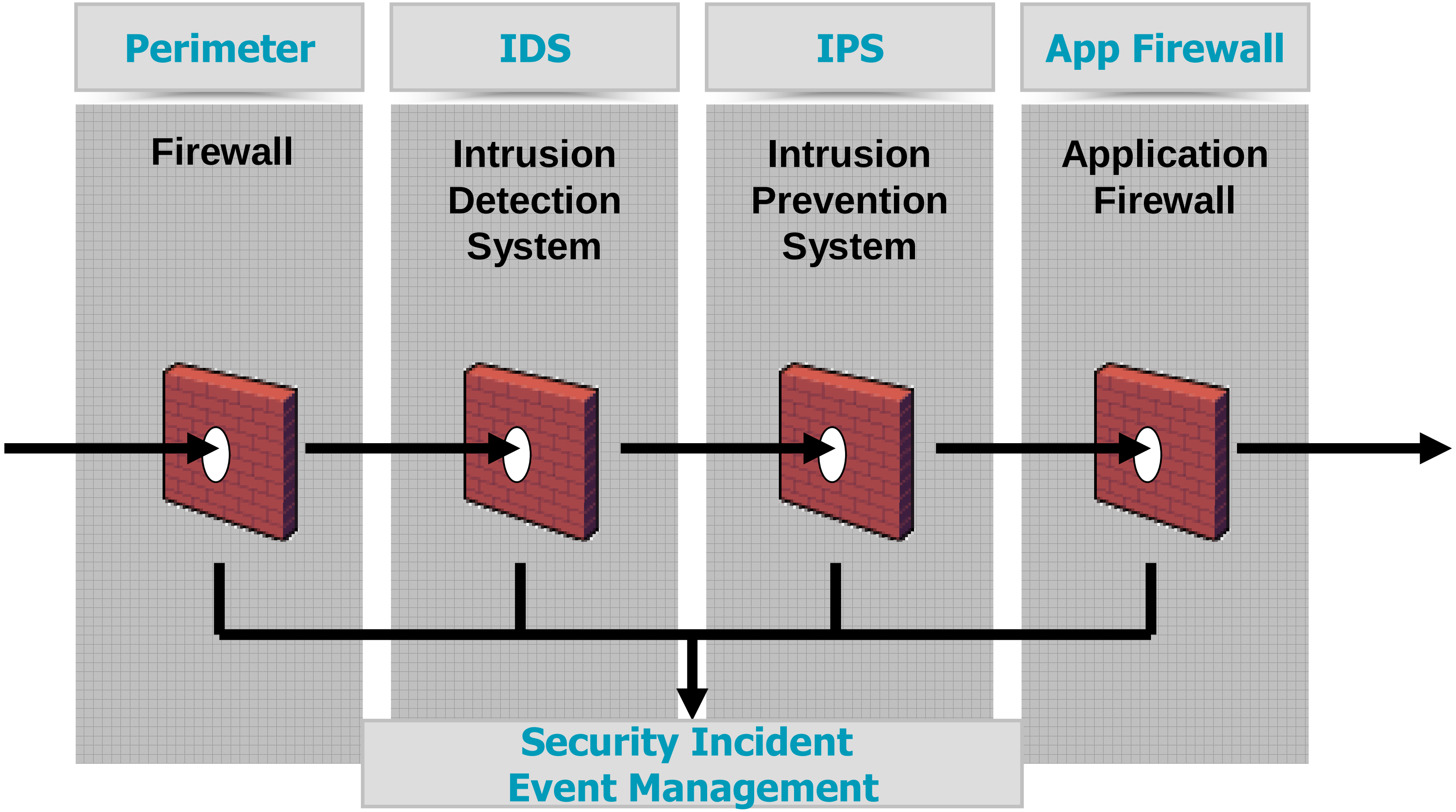
What keeps me **Rational**?

9 October – Johannesburg & 11 October – Cape Town

High Level Web Application Architecture



Network Defenses for Web Applications


Perimeter
IDS
IPS
App Firewall
Firewall
Intrusion Detection System
Intrusion Prevention System
Application Firewall
Security Incident Event Management

Web Application Security Business Drivers

IBM Rational Software Development Conference 2007

What keeps me **Rational**?

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

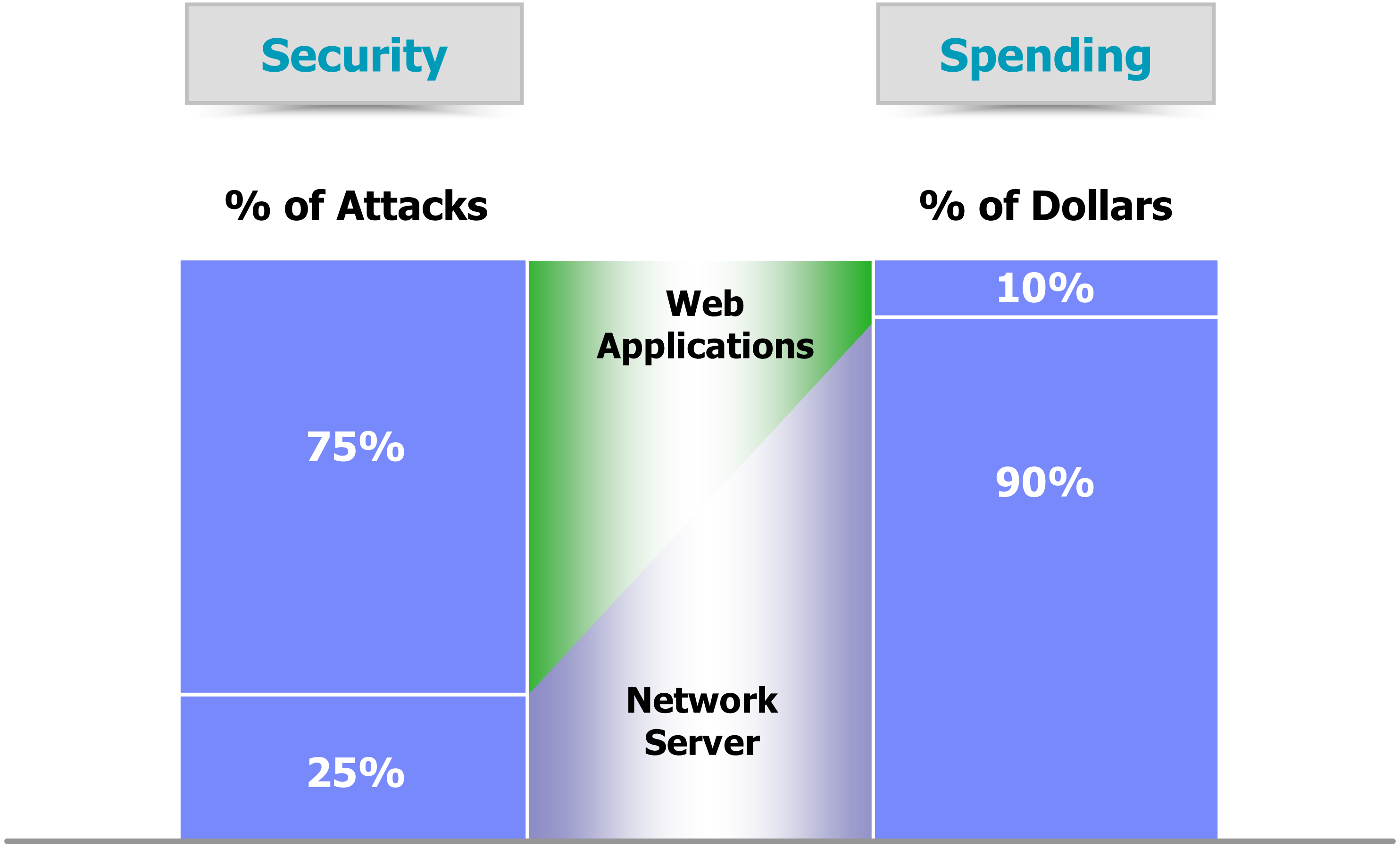
“Hacker may have stolen personal identifiable information for 26,000 employees..”

ComputerWorld, June 22, 2006

Why Web Application Security

- Protect sensitive customer, employee & business data
- Meet regulatory and corporate compliance requirements
- Defend against the high cost of attack
  - Media attention, brand damage, shareholder value
  - FTC fines and penalties
  - Audits
  - Lawsuits

The Challenge for Organizations



Sources: Gartner, IDC, Watchfire

Why Application Security is a High Priority

- **Web applications are the #1 focus of hackers:**
  - 75% of today's attacks occur at Application layer (Gartner)
  - XSS and SQL Injection are rated #1 and #2 vulnerabilities (Mitre)
- **Most sites are vulnerable:**
  - 90% of sites are vulnerable to application attacks (Watchfire research)
  - 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,

Top Attacks Overview

Security

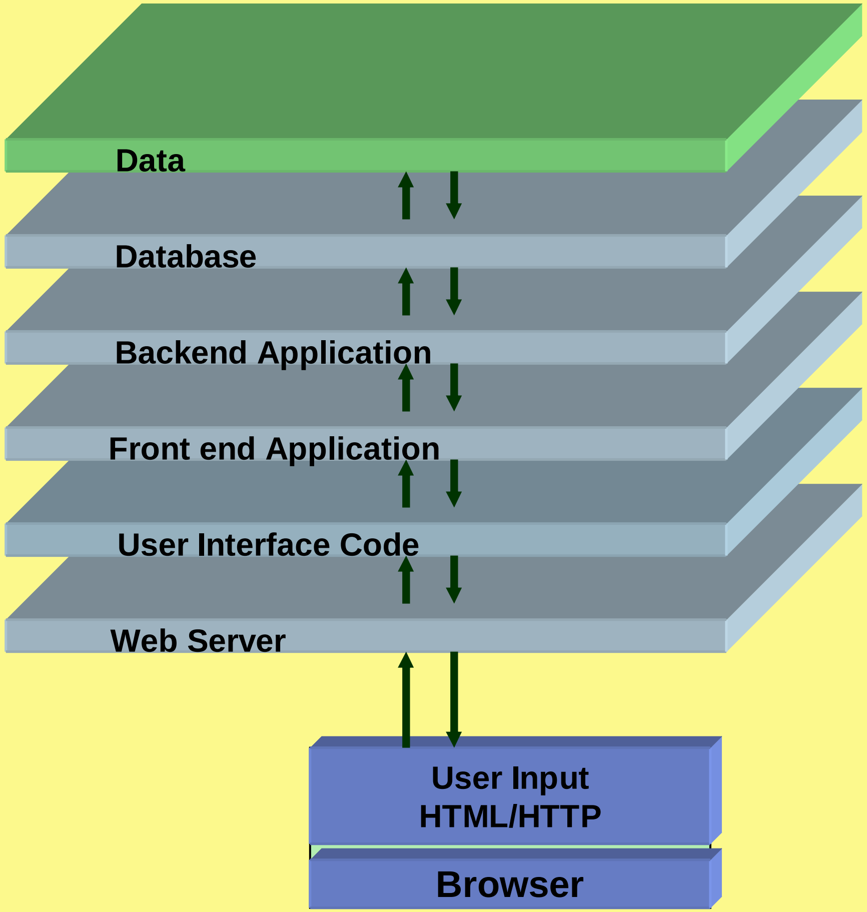
The Myth: “Our Site Is Safe”

We Have Firewalls in Place

We Audit It Once a Quarter with Pen Testers

We Use Network Vulnerability Scanners

What is a Web Application?



- **The business logic that enables:**
  - User’s interaction with Web site
  - Transacting/interfacing with back-end data systems (databases, CRM, ERP etc)
- **In the form of:**
  - 3rd party packaged software; i.e. web server, application server, software packages etc.
  - Code developed in-house / web builder / system integrator

Input and Output flow through each layer of the 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 **3rd party SW**	Insecure application development **In-house**
Location within Application	3rd 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 3rd party software	Early detection saves $$$

OWASP and the OWASP Top 10 list

- Open Web Application Security Project – 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)

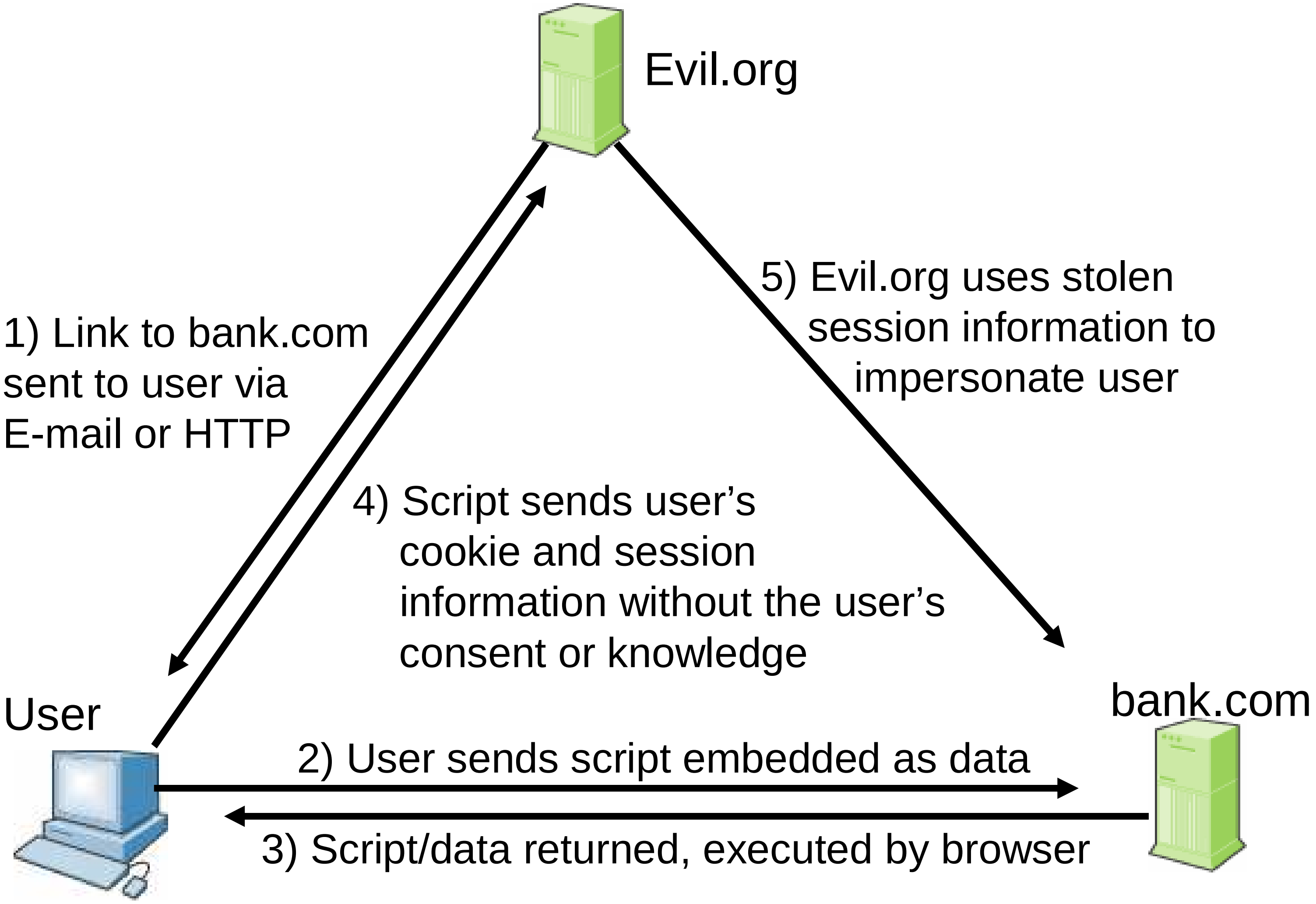
- What is it?
  - Malicious script echoed back into HTML returned from a trusted site, and runs under trusted context

- What are the implications?
  - Session Tokens stolen (browser security circumvented)
  - Complete page content compromised
  - Future pages in browser compromised

XSS – Details

- Common in Search, Error Pages and returned forms.
  - But can be found on any type of page
- Any input may be echoed back
  - Path, Query, Post-data, Cookie, Header, etc.
- Browser technology used to aid attack
  - XMLHttpRequest (AJAX), Flash, IFrame…
- Has many variations
  - XSS in attribute, DOM Based XSS, etc.

Cross Site Scripting – The Exploit Process



Evil.org

1) Link to bank.com sent to user via E-mail or HTTP

5) Evil.org uses stolen session information to impersonate user

4) Script sends user's cookie and session information without the user's consent or knowledge

User

bank.com

2) User sends script embedded as data

3) Script/data returned, executed by browser

Exploiting XSS

- If I can get you to run my JavaScript, I can…
  - Steal your cookies for the domain you're browsing
  - Track every action you do in that browser from now on
  - Redirect you to a Phishing site
  - Completely modify the content of any page you see on this domain
  - Exploit browser vulnerabilities to take over machine
  - …
- XSS is the Top Security Risk today (most exploited)

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

3. Malicious File Execution

- What is it?
  - Application tricked into executing commands or creating files on server

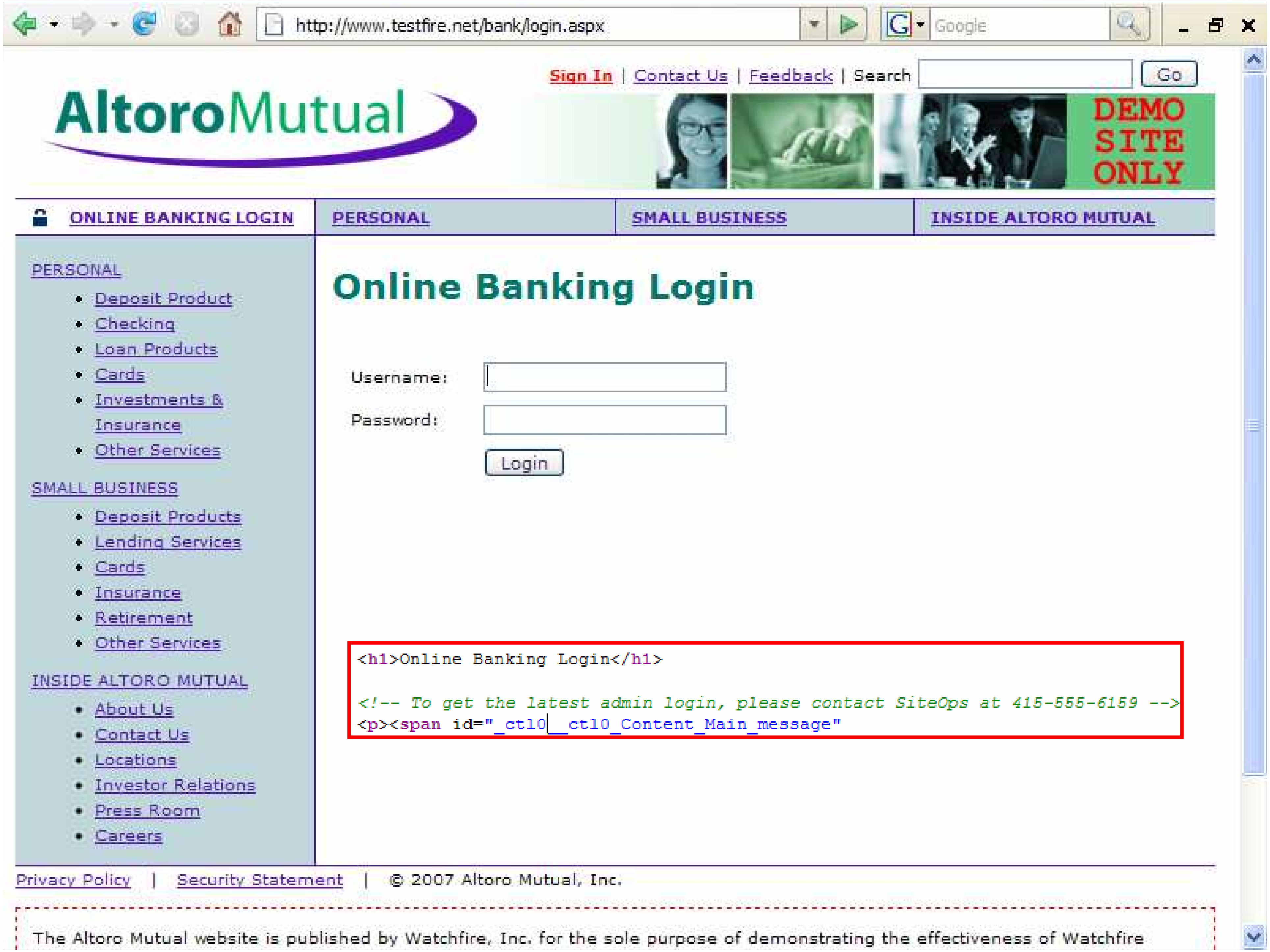
- What are the implications?
  - Command execution on server – complete takeover
  - Site Defacement, including XSS option

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

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



Online Banking Login

Username:

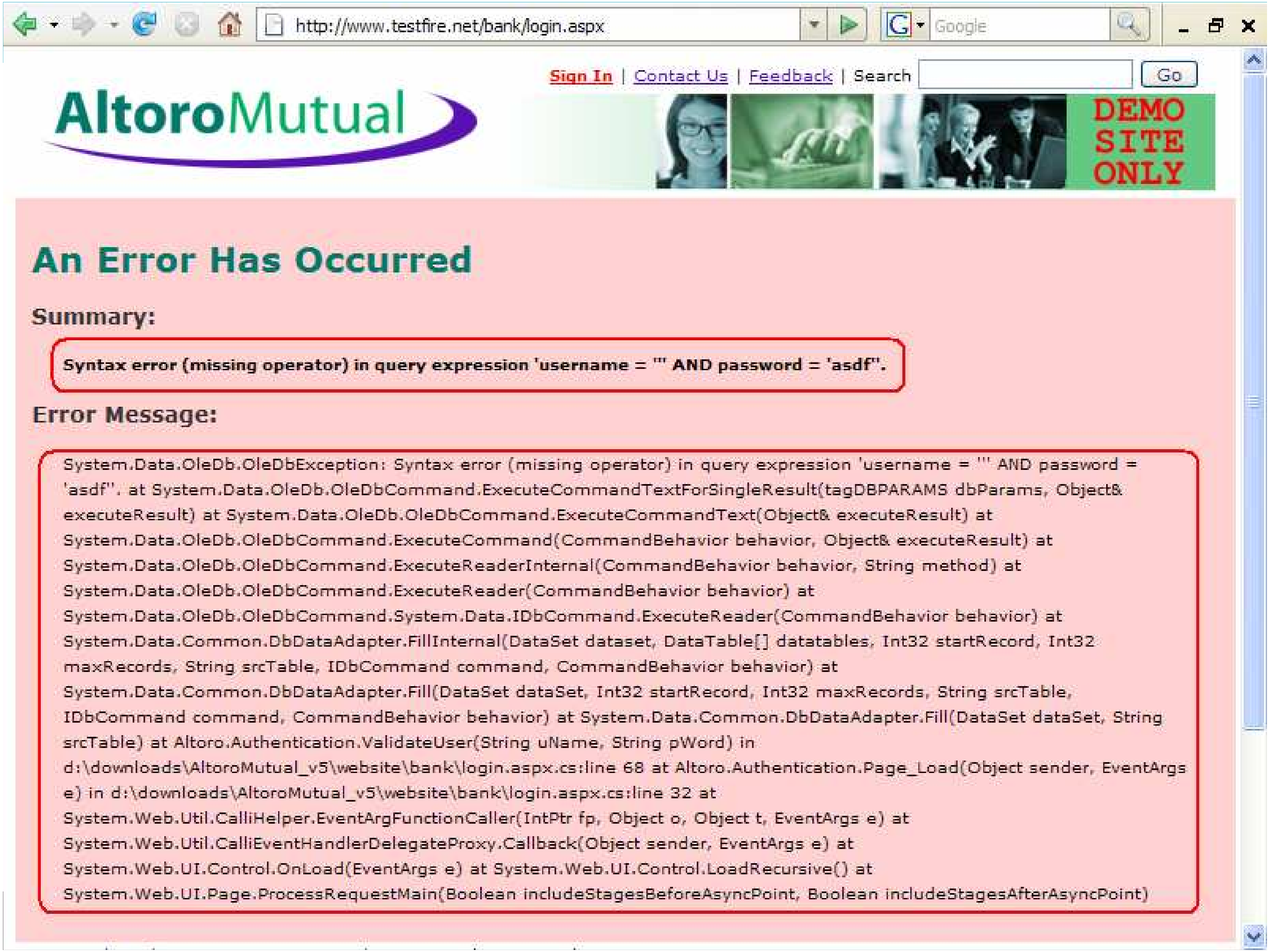
Password:

Login

```
<h1>Online Banking Login</h1>

<!-- To get the latest admin login, please contact SiteOps at 415-555-6159 -->
<p><span id="_ctl0__ctl0_Content_Main_message"
```

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


http://www.testfire.net/bank/login.aspx
Google
Sign In | Contact Us | Feedback | Search
Go
AltoroMutual
DEMO SITE ONLY

An Error Has Occurred

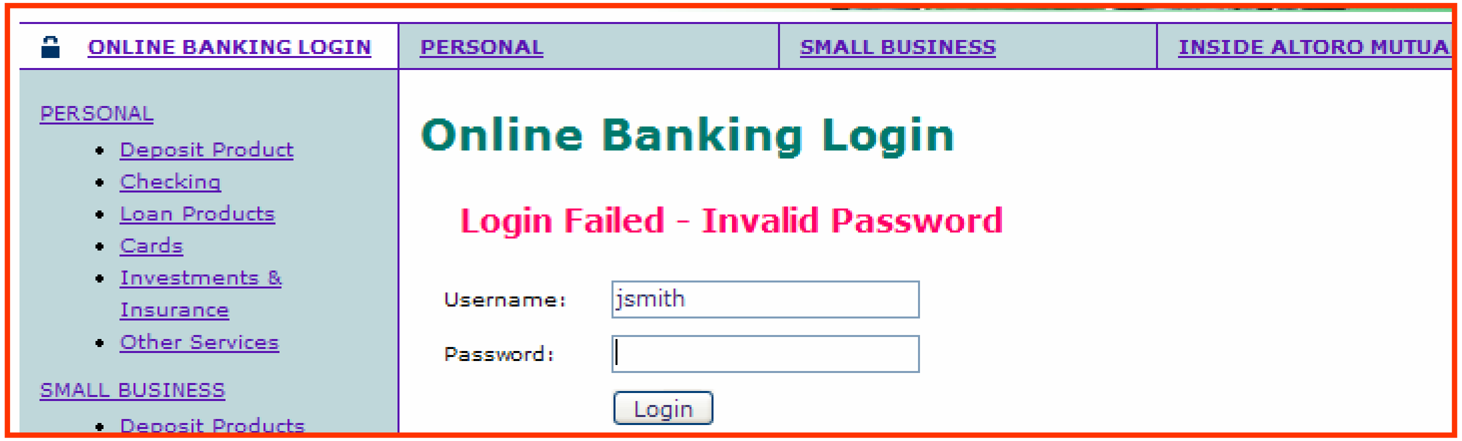
Summary:

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.ExecuteCommandTextForSingleResult(tagDBPARAMS dbParams, Object&
executeResult) at System.Data.OleDb.OleDbCommand.ExecuteCommandText(Object& executeResult) at
System.Data.OleDb.OleDbCommand.ExecuteCommand(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



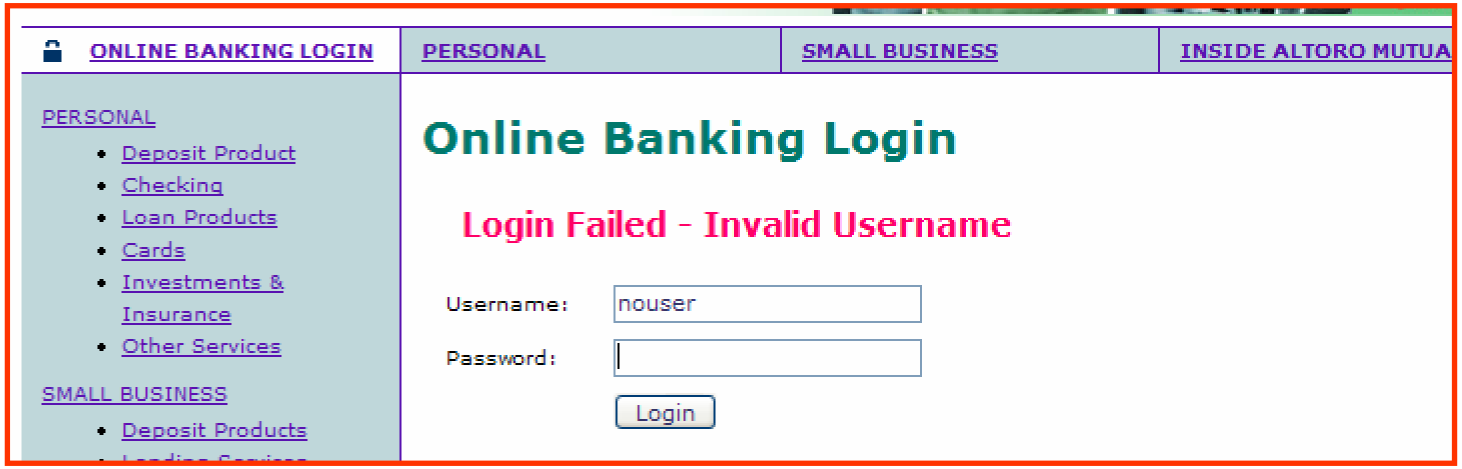
Online Banking Login

Login Failed - Invalid Password

Username: jsmith

Password:

Login



Online Banking Login

Login Failed - Invalid Username

Username: nouser

Password:

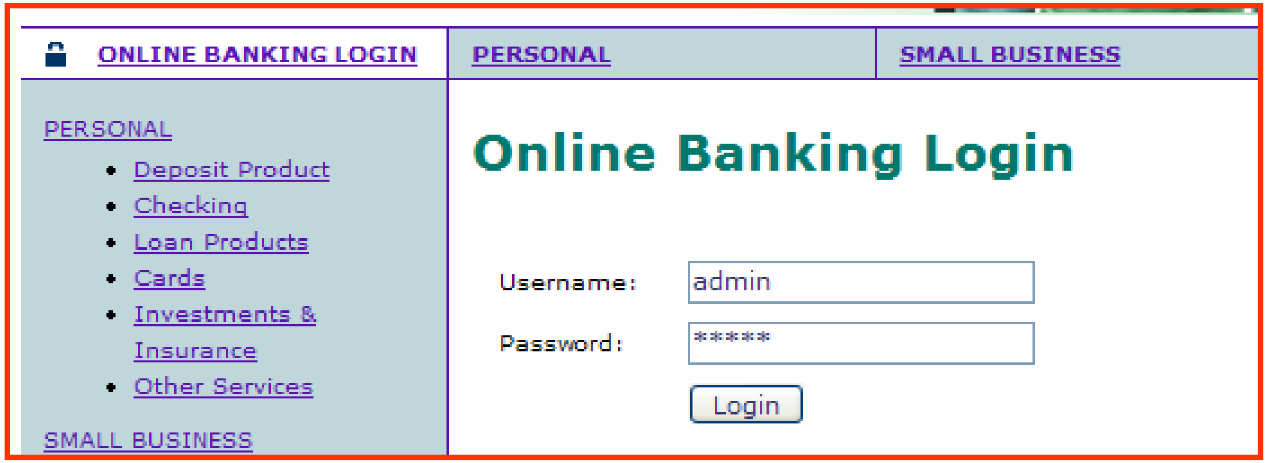
Login

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

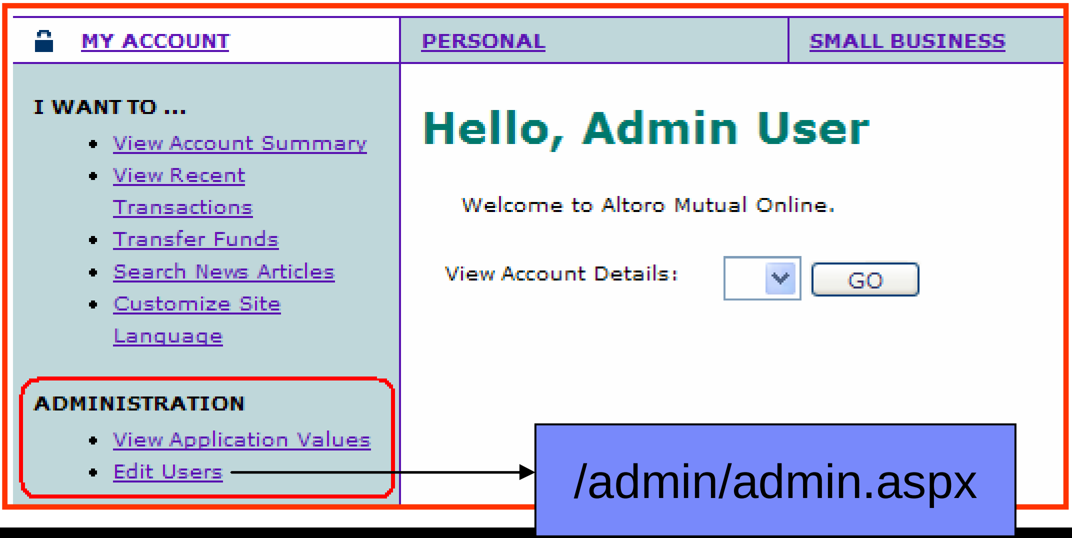
Failure to Restrict URL Access - Admin User login



Online Banking Login

Username: admin

Password: *****



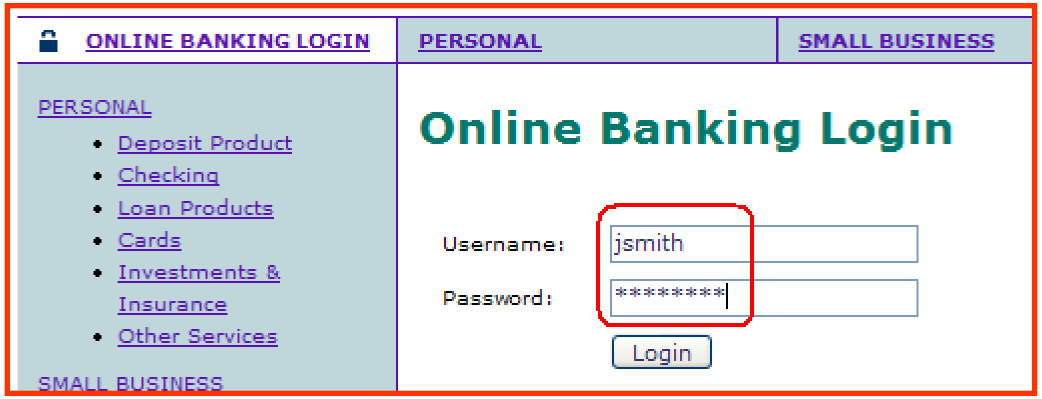
Hello, Admin User

Welcome to Altoro Mutual Online.

View Account Details:

/admin/admin.aspx

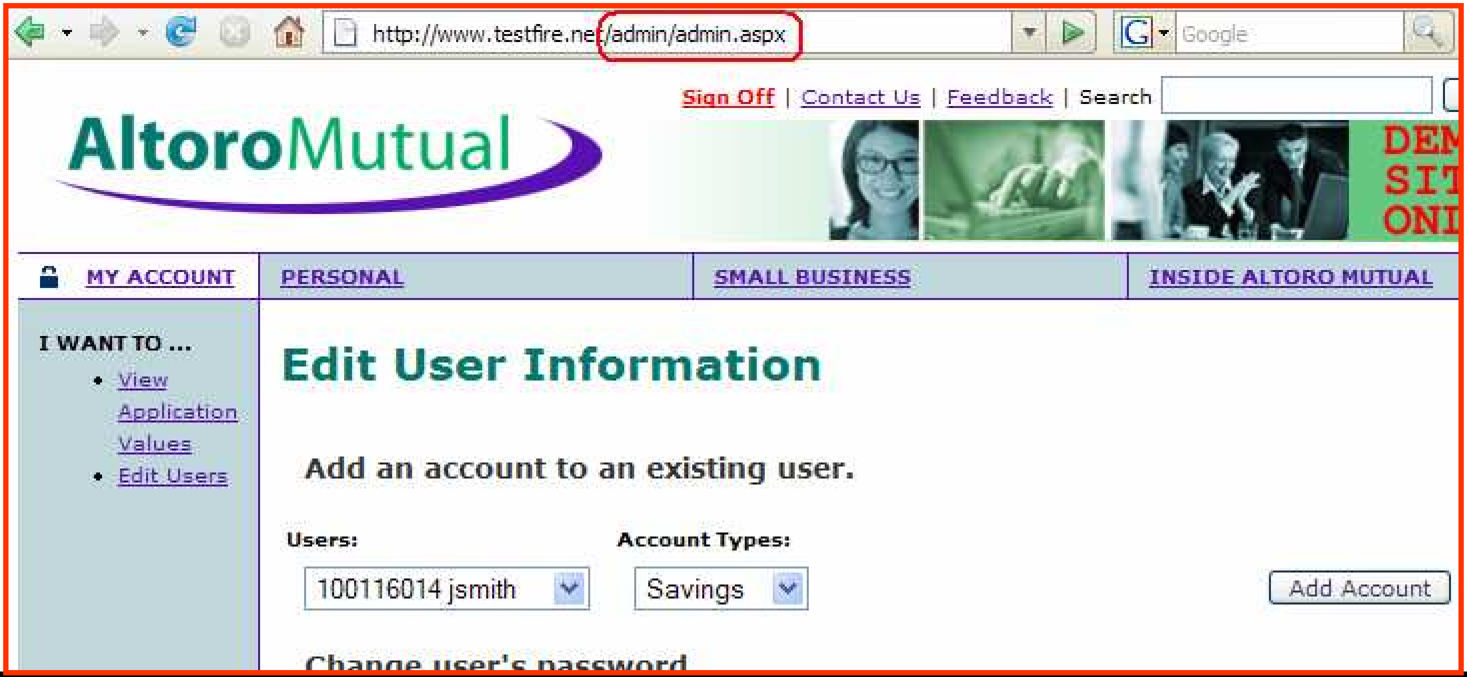
Simple user logs in, forcefully browses to admin page



Online Banking Login

Username: jsmith

Password: ********



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

Edit User Information

Add an account to an existing user.

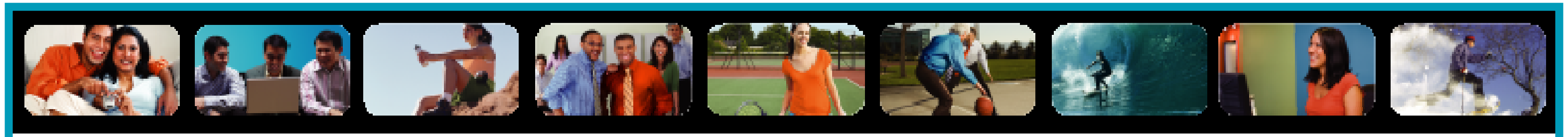
Users: 100116014 jsmith

Account Types: Savings

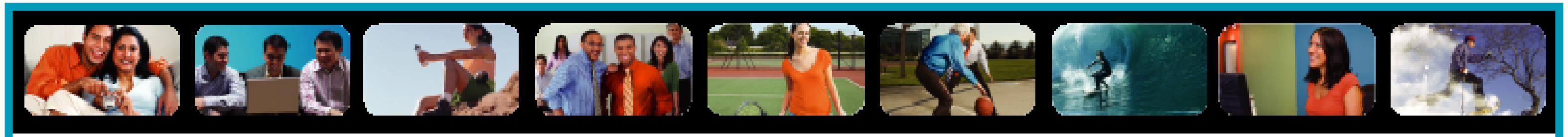
Change user's password

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



Demo



Questions

Resources

- Download AppScan 7.0 - http://www.watchfire.com
- Latest whitepapers visit: http://www.watchfire.com/news/whitepapers.aspx
- Visit Watchfire at one of our upcoming shows http://www.watchfire.com/news/events.aspx
- Register for upcoming web seminars visit http://www.watchfire.com/news/seminars.aspx
- Contact us at sales@watchfire.com

Thanks for joining me today!

IBM Rational Software Development Conference 2007

What keeps me Rational?