Locking Down Layer 7

Pete Lindstrom, CISSP
Research Director

Spire Security, LLC
www.spiresecurity.com
petelind@spiresecurity.com

Web/Web Services: Bane or Panacea?

From monolithic mainframe, to two- and three- tier client server, to n-tier web, and on to *n-peer* Web Services

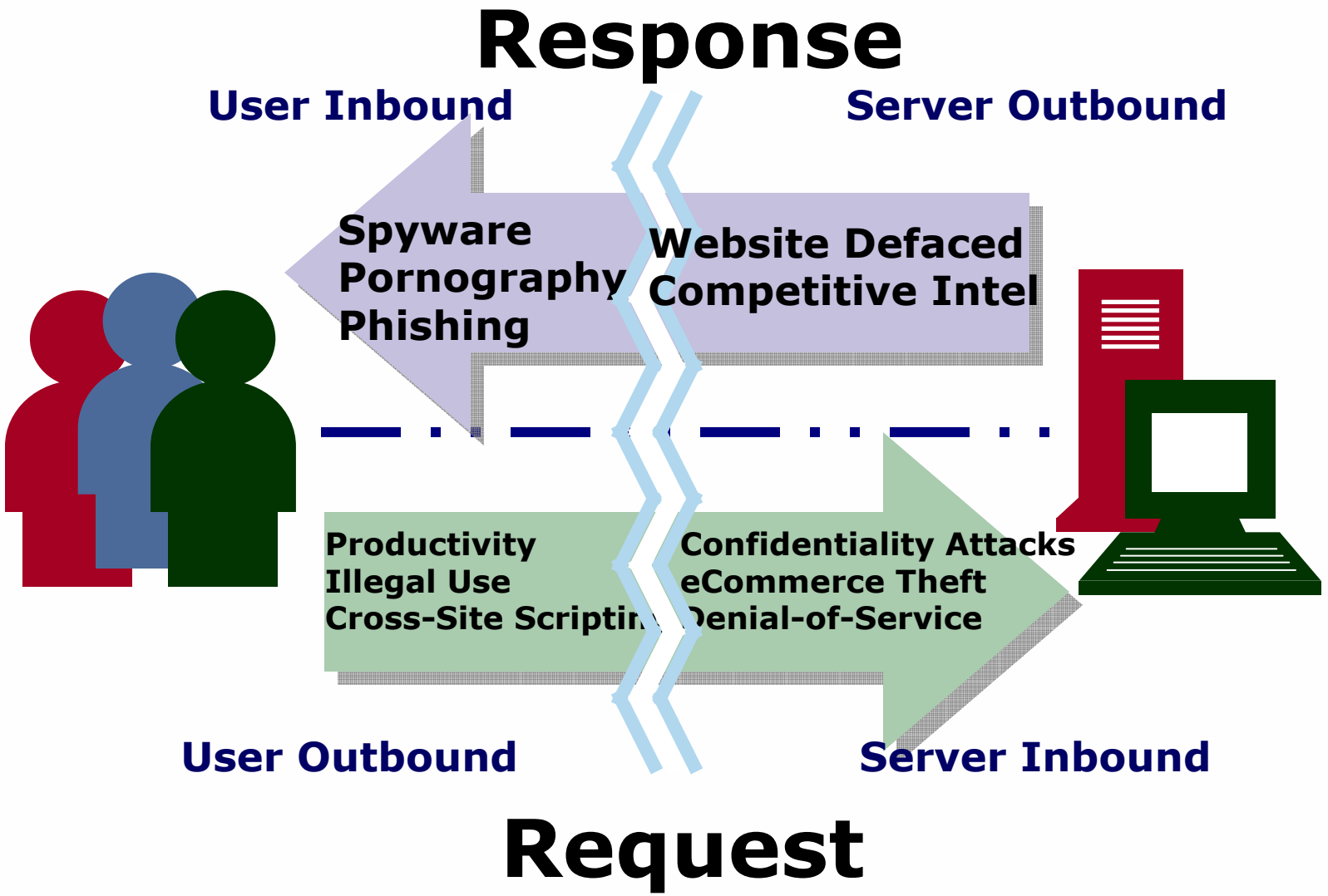
- **Standardization: common communication protocols**
  - **Easier to learn technology, higher likelihood of finding a target.**
- **Loose-coupling: flexible architecture**
  - **More uniquely addressable attack points.**
- **Federation: working together**
  - **More ways to "hide" amidst legitimate traffic.**

•Increased functionality brings increased risk, but it may be worth it.

Agenda

- **Web Architecture**
- **Web Attack Techniques**
- **Web Services Architecture**
- **Web Services Attack Techniques**

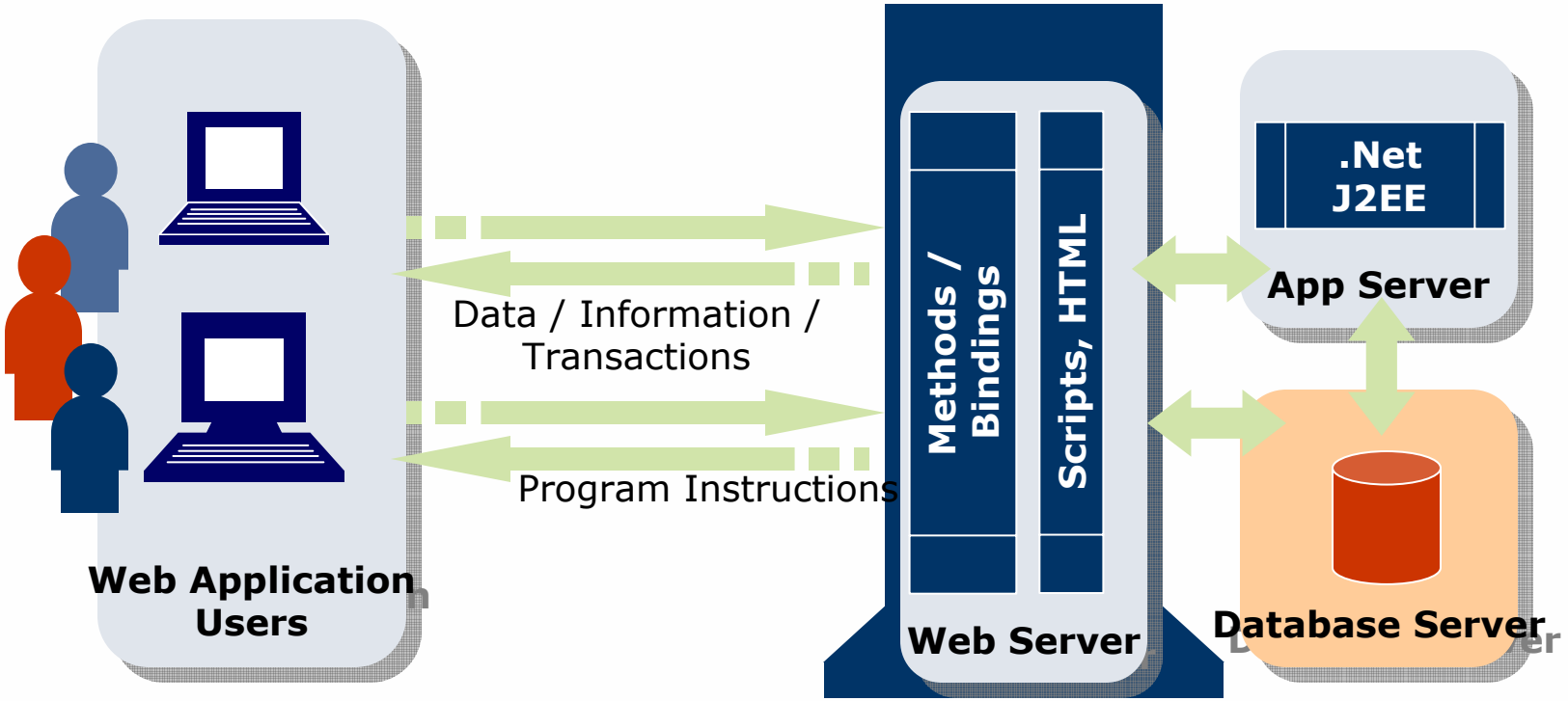
Web Security Round Trip



Threat Modeling Process

1. **Define Attack Points (Targets)**
2. **Define Attack Paths (Inputs)**
3. **Evaluate Attack Techniques (Exploits)**
4. **Apply Controls**
5. **[Constantly re-evaluate]**

Web Architecture



XML Documents

Attack Points (Targets)

- **Web Client**
- **Web Server**
- **Application Server**
- **Database Server (we'll ignore the operating system, network devices, message queues, federated systems, legacy systems, etc)**

Attack Paths (Inputs)

- **Web Client – client-side scripts; toolbars; drive-by installs**
- **Web Server – static html; CGI scripts.**
- **Application Server – dynamic html; database connection**
- **Database Server – stored procedures; SQL**

Web Client Attacks – Reasons

- **System resource usage**
  - **Participate in DoS attacks**
  - **Run as remote agent to perform other tasks**
- **Access to the private network**
  - **Conduit through VPN to enterprise**
- **Credentials**
  - **Identity theft and fraud**

Web Client Attacks - Techniques

- **Automated social engineering (Pop-ups)**
- **Obfuscated URLs (XSS, phishing)**
- **"Drive-by" installations**

Protecting the Web Client

- **Block Pop-ups; verify top level domain.**
- **Scan file system, registry, application configuration.**
  - **Search for "known" spyware.**
- **Run only approved applications.**
  - **Protects against drive-by installs.**

Server Attacks – Reasons

- **Merchandise Theft**
- **Site Defacement**
- **Information Theft (identity)**
- **Denial-of-Service**

Server Attacks – Techniques

- **Session/State Tampering**
  - **Cookie Poisoning, URLs**
- **Parameter Manipulation**
  - **Hidden form fields; HTML forms; URLs**
- **Known Vulnerabilities**
  - **Client, Web/App/DB Server**
- **SQL Injection**
  - **Database attack**

Protecting the Server

- **Patch**
- **Identify and validate inputs**
- **Harden Cookies**
- **Harden URL paths**
- **Consider Solutions**
  - **Web App Firewalls**
  - **Web App Shields**

Introducing Web Services

- **XML –EXtensible Markup Language creates a way to define many different data formats so that platforms can interoperate. XML documents and transactions are made up of elements within a multi-level hierarchical structure.**

- **UDDI – The Universal Description, Discovery, and Integration specification provides a registry for Web Services that can be searched for services and allows for dynamic updates.**

- **WSDL – The Web Services Description Language provides a way to describe interfaces for Web Services.**

- **SOAP – The Simple Object Access Protocol that provides a network protocol for transport of Web Services documents.**

Web Services Components

- **XML Documents / SOAP Messages**
- **Configuration Data (the setup)**
- **XML Processors**
  - **Legacy Apps**
  - **External Entities**
  - **Repositories**

XML Docs /SOAP Messages

- **Risk: Protocol Abuse**
- **XML Content as:**
  - **Protocol Conversations**
    - **Expected operations**
  - **Program Instructions**
    - **RPC / Commands (embedded code)**
    - **And variables, flags, attributes**
  - **Transactions - data**
  - **URIs - pointers**

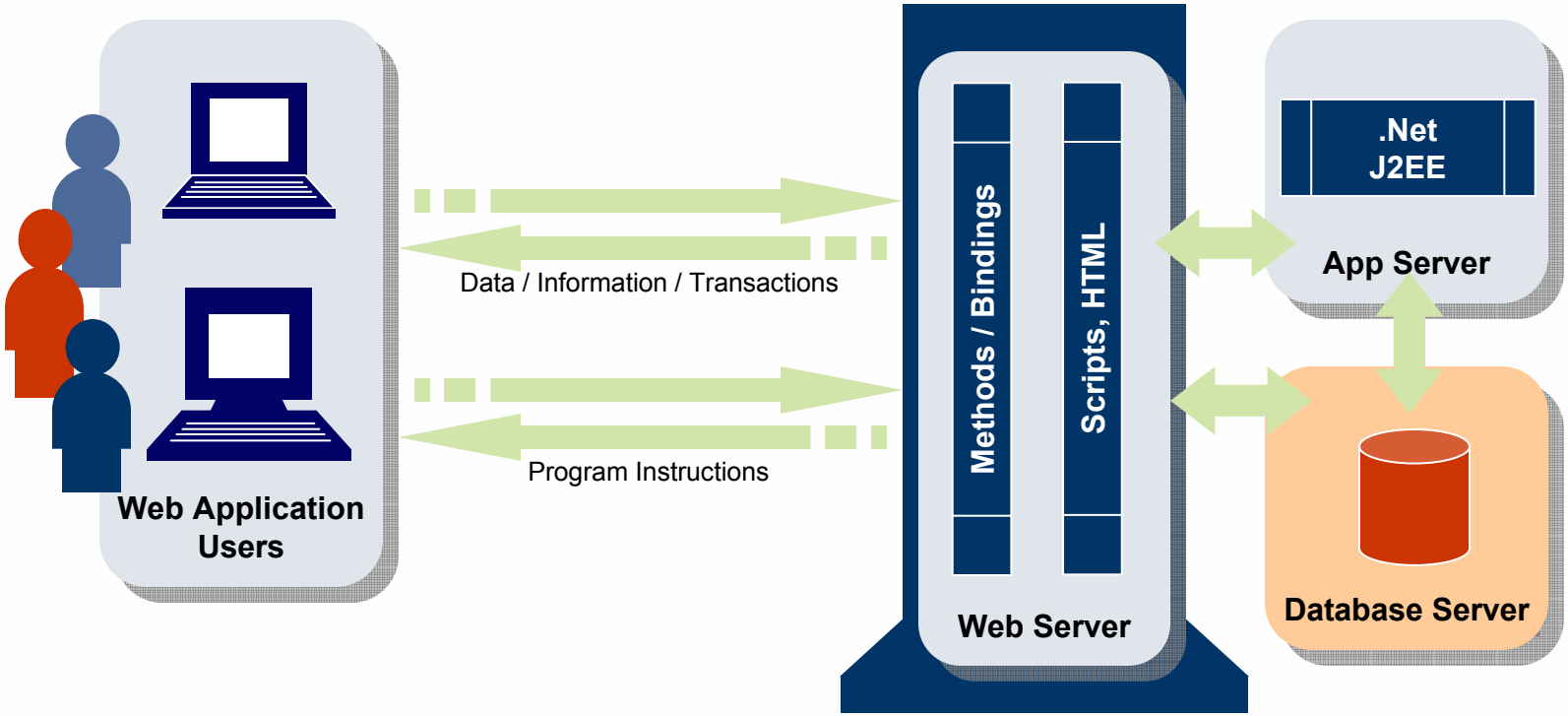
Web Services Configuration Data

- **Web Services Description Language (WSDL) Files**
- **XML Schemas**
- **XSLT Files**
- **WS-Policy information**

XML Processor

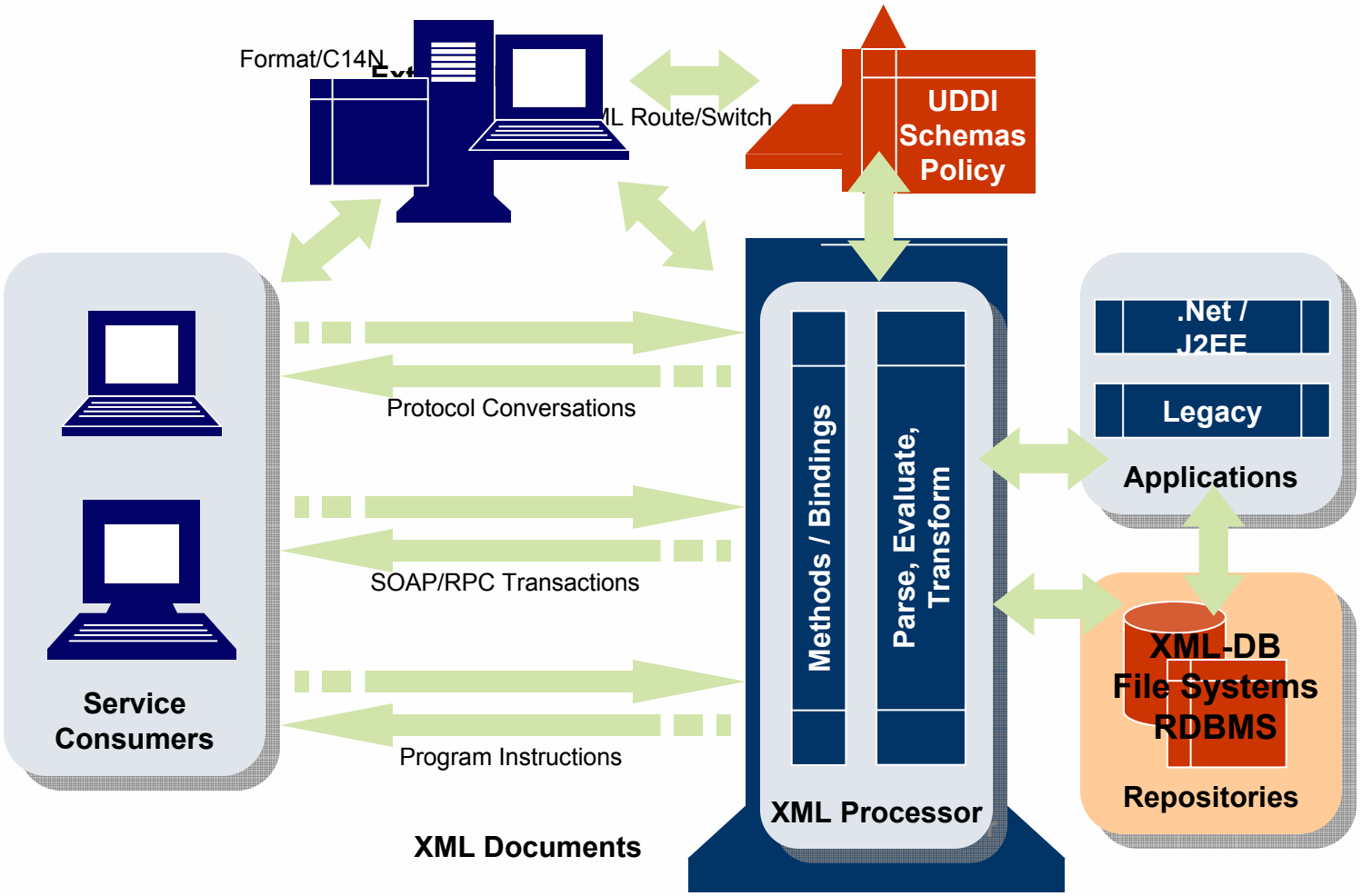
- **Standard operations**
  - **Parse XML**
  - **Aggregate data**
  - **Transform data**
  - **Canonicalize data**
- **All Legitimate manipulation of data after the source.**
- **Legacy bolt-ons**
- **Untrusted entities**

Remember: Web Architecture



XML Documents

Example: Web Services Threat Profile



Top Ten Attack Techniques

1. **XML Encapsulation**
2. **Coercive Parsing**
3. **Recursive Elements**
4. **Jumbo Payloads**
5. **Schema Poisoning**
6. **WSDL Enumeration**
7. **Routing Detours**
8. **External Entity Attacks**
9. **XQuery/XPath Injection**
10. **Malicious Morphing**

1. XML Encapsulation

- **Attacks legacy bolt-on xml processors.**
- **External operation of normally local functions.**
- **Uses "CDATA" feature in XML to "tunnel" through to app.**

XML Encapsulation Example

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="#?m$ux" ?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/TR/WD-
xsl">
<xsl:script>
<![CDATA[
x=new ActiveXObject("WScript.Shell");
x.Run("%systemroot%\\SYSTEM32\\CMD.EXE /C DIR C:\\
/a /p /s");
]]>
</xsl:script>
<msux>
msux
written by georgi guninski
</msux>
</xsl:stylesheet>
```

Source: http://www.guninski.com/ex$el2.html

2. Coercive Parsing

- **Attacks legacy bolt-on xml processors.**
- **Attacks old targets in new ways.**
- **External operation of normally local functions.**
- **Instead of using CDATA, uses XML parsing capability.**

3. Recursive Elements

- **Use XML within a document to reference another point in the document.**
- **Infinite loop**

4. Jumbo Payloads

- **XML is verbose by nature, but still expected to be of "reasonable" size files – measured in kb or mb.**
- **Jumbo payloads send a single file of hundreds of gigabytes (for example).**

5. Schema Poisoning

- **XML Schemas are used to define format and function of a document.**
- **Manipulating the schema can:**
  - **Execute denial-of-service attack.**
  - **Change formats from dates to numbers, for example**
  - **Obfuscate data.**

6. WSDL Enumeration

- **WSDL files provide detailed "API-like" information for anyone accessing them.**
- **Enumeration of WSDL files may expose debug information or private commands that aren't intended for public use.**

7. Routing Detours

- **XML by design is highly flexible.**
- **WS-Routing allows for inserting or appending in-process instructions that ensure a document gets to its anticipated destination (proxy-like).**
- **Inserting inappropriate or malicious routing instructions can allow for a man-in-the-middle attack or other attacks against conf, int, and avail.**

8. External Entity Attacks

- **Rather than attacking the xml document, attack the individual components of an architecture.**
- **Many waystations/interim queues of documents provide an opportunity to modify the contents.**

9. XQuery/XPath Injection

- **XQuery and XPath are ways to describe database queries using XML.**
- **XQuery injection is the same as SQL injection – insert commands into an element that gets interpreted inappropriately.**

10. Malicious Morphing

- **Web Services is malleable by design.**
- **Aggregate/Transform/C14N operations are expected.**
- **When they are performed by the wrong entities, they can be used as an attack.**

Top Ten Attack Techniques

1. **XML Encapsulation**
2. **Coercive Parsing**
3. **Recursive Elements**
4. **Jumbo Payloads**
5. **Schema Poisoning**
6. **WSDL Enumeration**
7. **Routing Detours**
8. **External Entity Attacks**
9. **XQuery/XPath Injection**
10. **Malicious Morphing**

Agree? Disagree?

Pete Lindstrom
petelind@spiresecurity.com
www.spiresecurity.com