

Web Services Attacks and Countermeasures

K. Scott Morrison
Director, Architecture



May 2004



Bio – K. Scott Morrison

- Director, Architecture at Layer 7 Technologies
 - <http://www.layer7tech.com>
 - Layer 7 is based in Vancouver BC, Canada
- Co-author of Sams *Java Web Services Unleashed* & Wrox's *Professional JMS*
 - Over 35 other publications in academic journals and trade magazines
- Frequent speaker on Web services, XML, mobile/wireless computing systems, distributed systems architecture, and Java design issues



Agenda and Theme

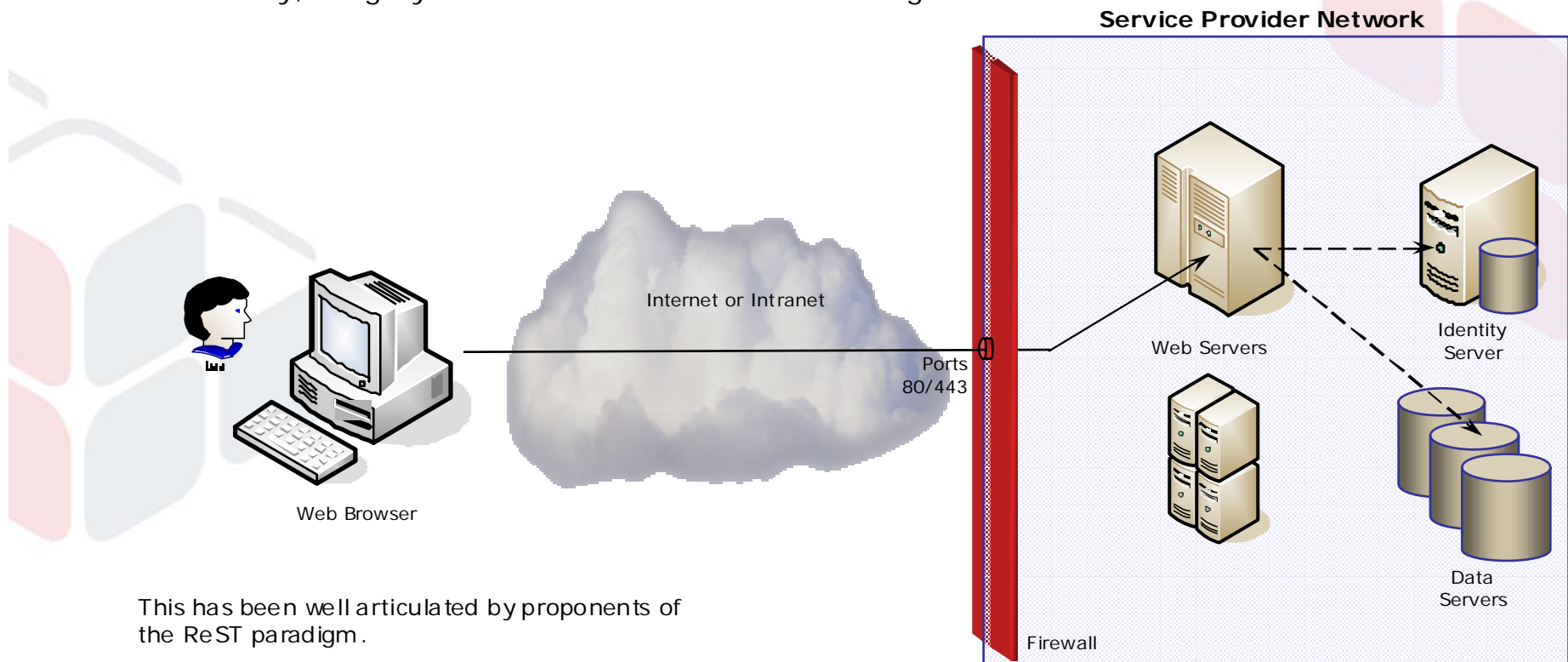
- What are the philosophical underpinnings of security the conventional Web?
- Why this fails for Web services
 - We'll look at this from an architectural, not a technological perspective
- How to fix it
 - Policy Engines
 - Strong Identity Model
- What this looks like in deployment
 - Policy Decision Points, Enforcement Point, and Application Points

Theme: The business problem is *flexible integration*. Security technology has to be a mechanism to address this, not an end in itself



Security on the Web

- A limited set of verbs (GET, POST, PUT, DELETE, etc), but many nouns (employee_MORRISON, order_123, etc)
 - Nevertheless, we've had many problems with this.
- Authentication using simple HTTP mechanisms (basic, digest, certificate, etc)
- Authorization using ACL tied to an URL (<http://www.layer7tech.com/orderEntry>)
- Confidentiality, integrity and client-cert authentication using SSL



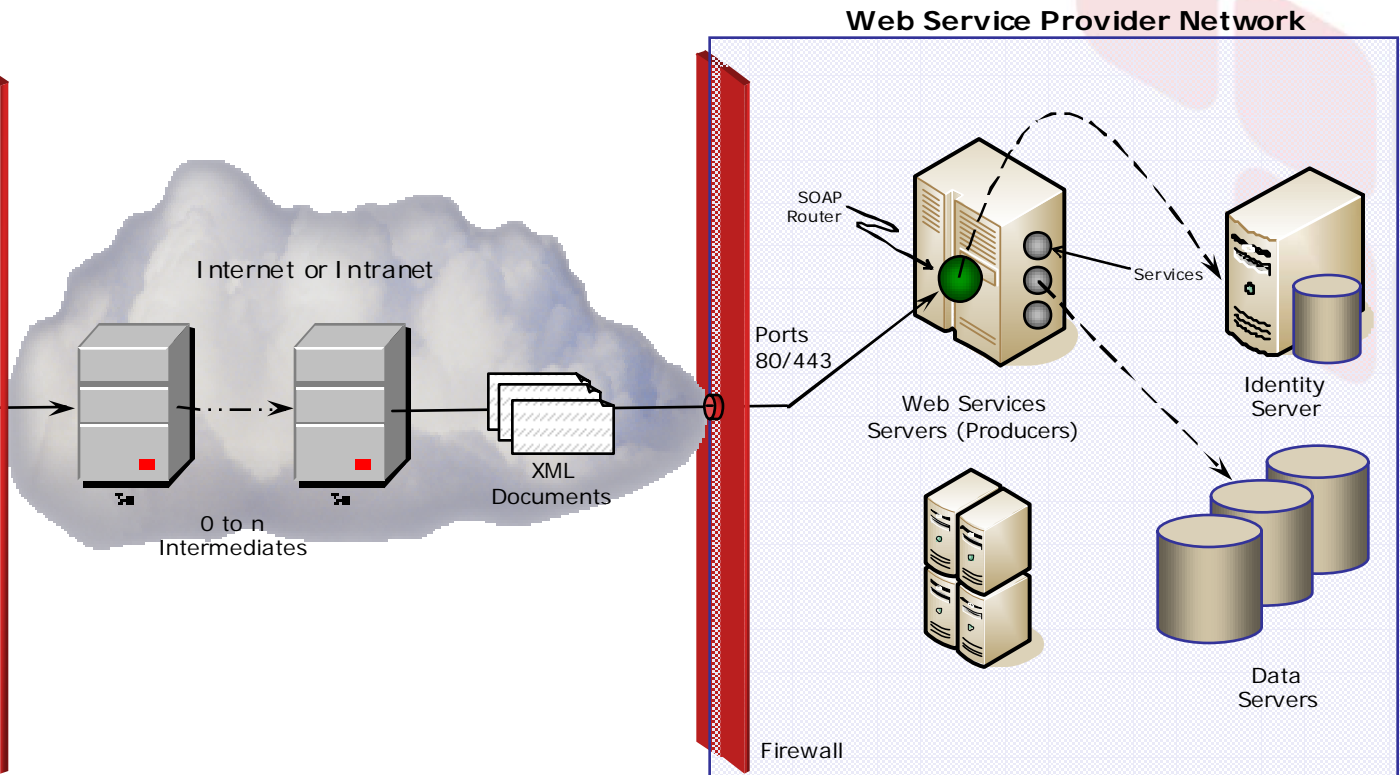
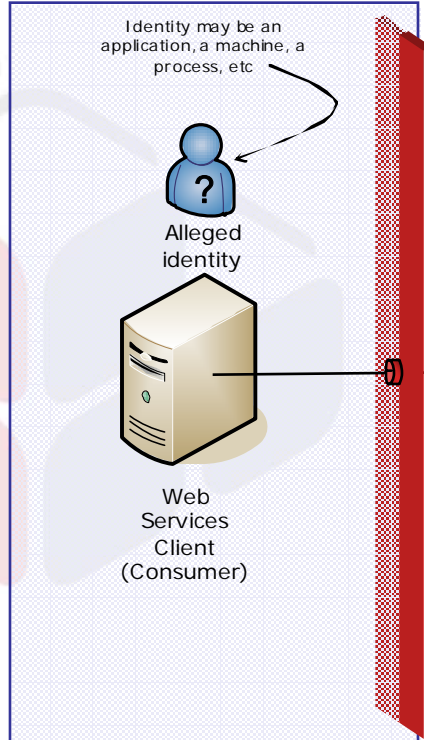
This has been well articulated by proponents of the ReST paradigm.



Why Conventional Web Security Fails for Web Services

- An infinite set of verbs (`deleteEmployee`, `makeOrder`, etc), and infinite nouns (`employee_MORRISON`, `order_123`, etc)
- Authentication needs to be on a message-by-message basis
 - No sense of an authenticated and secure channel anymore
- Authorization can't be tied to an URL (<http://www.layer7tech.com/soapRouter>)
- Confidentiality and integrity should be decoupled from transport to support intermediates or other transports

Web Service Client Network



How Can We Deal with This?

- We need two critical technologies

1. Rich Policy

- This must be much more than simple authentication and authorization

2. A Strong Identity Model

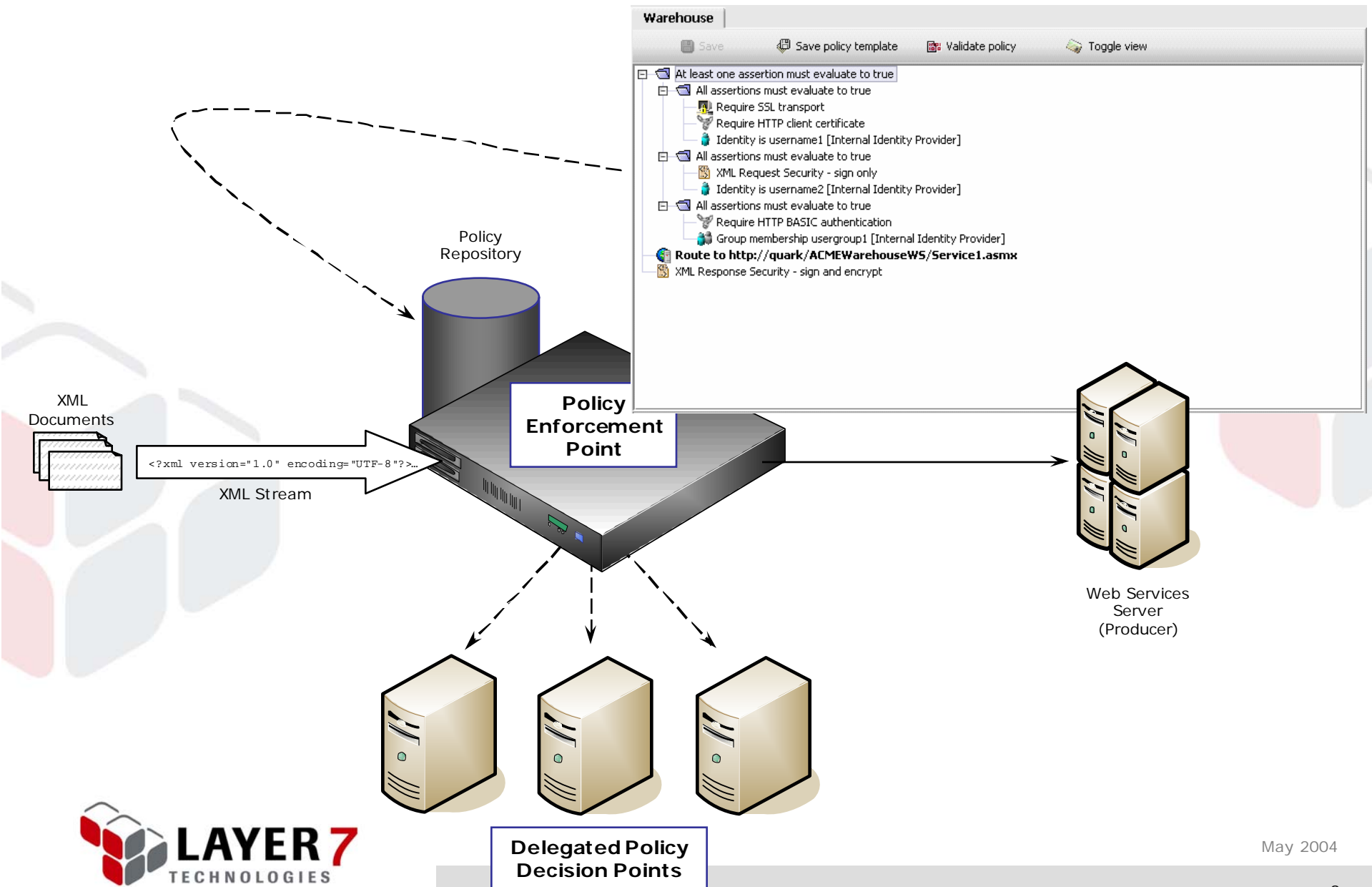
- Fully integrated into Web services infrastructure
- (Hint: Web services finally gives PKI something to do...)

Policy: It's More Than Access Control

- Authentication and authorization are certainly a component
- Consider also:
 - Cryptography and integrity expectations
 - Reliability
 - Logging and audit
 - Transactional boundaries
 - etc...

Essentially, policy encompasses all of the variants that can be managed across a transaction

Policy is Associated with Services

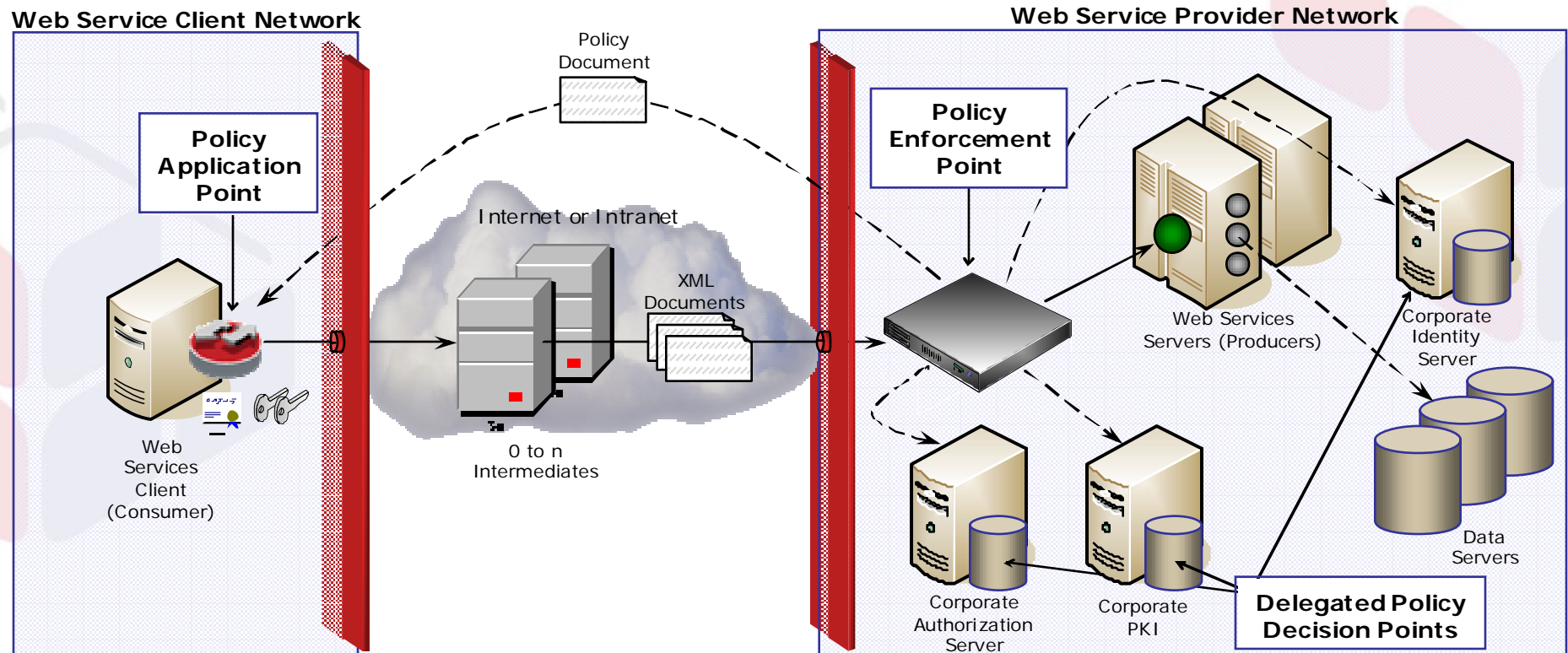


A Strong Identity Model

- Remember the *Principle of Least Privilege*
 - Also known as the *Need to Know*
 - This is only relevant under a strong identity model
- We would like to associate policy against services *and* identities
 - E.g. smorrison can only access service getQuote from 09:00 to 17:00 hrs, M-F
- Binding identity to messages is less trivial than to a channel, but more rewarding
 - The beginnings of true non-repudiation
- The problem is, this really needs PKI fully integrated into Web services infrastructure

Deployment of a Complete Solution

- Need a *Policy Enforcement Point* that delegates appropriate *Policy Decisions* to existing infrastructure
- We need an effective PKI infrastructure to build the identity model, and to provide confidentiality and integrity
 - Harder than it sounds. Consider the great PKI failure on the web
- To avoid baking policy into code (thus eliminating loose coupling), we need a *Policy Application Point*



For further information:

K. Scott Morrison

Layer 7 Technologies

Suite 501 – 858 Beatty St.

Vancouver, BC V6B 1C1

Canada

(800) 681-9377

smorrison@layer7tech.com

<http://www.layer7tech.com>



May 2004

