An Overview of Distributed Security Architectures and Integration

Quadrāsis™

Don Flinn & Ted Burghart

Hitachi Computer Products (America) , Inc.

DOCSec March 19th, 2002

What's This Tutorial About?

- First Section
  - by Don Flinn
    - Technical Survey of the Security Solutions for Distributed Security (50,000 foot)
    - Discuss End to End Distributed Security
      - Describe the Problem and Where We Are Re: Solutions
- Second Section
  - By Ted Burghart
    - Dig Deeper Into Technology Integration Issues
    - Present Details of Selected Solutions

End-to-End Distributed Security Overview

Quadrāsis™

Don Flinn

Don.Flinn@Quadrasis.com

Distributed Security Problem

- “The Network is the Computer” (SUN)
  - No Longer Is Your Computer Isolated
  - Interaction is Within and Outside Your Company
- New Paradigm - Let Customers & Partners Into Your System
- Point Security Solutions Not Sufficient
  - Need End to End Responsibility
- Application Level Security Not Solution
  - Programmers Not Security Experts
  - Security Depends on the Application Call Chain
  - One Weak Point in the Chain Spells Disaster

Requirements for DOCsec

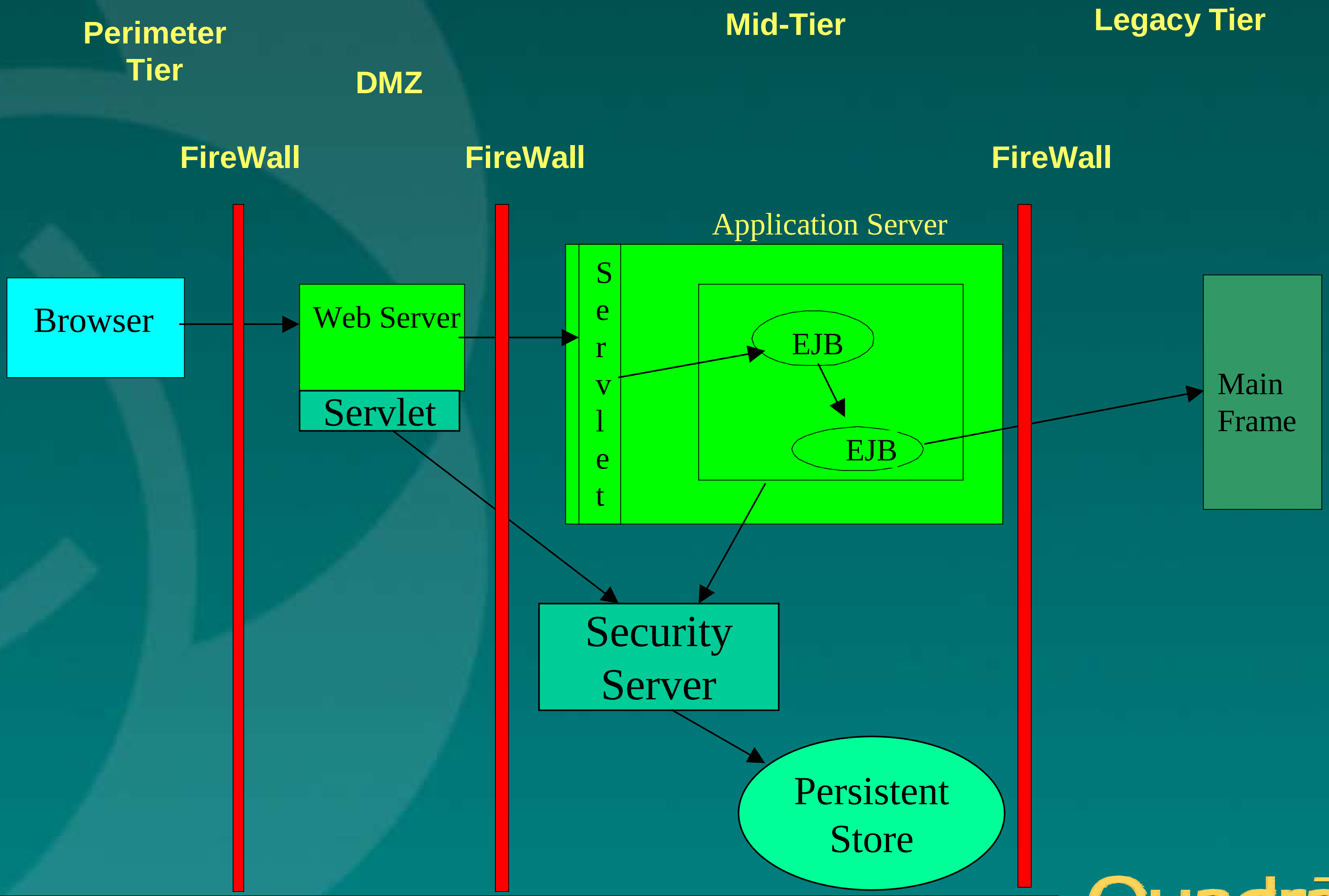
- Satisfy End-to-End Security Needs
- Satisfy Single SignOn (SSO)
- Access to internal information on a need-to-know basis, both for employees, partners and customers
- Simplified Administration (Security)
- Privacy
- Federation: interoperability across enterprises or departments
- 24 x 7 X 365 availability
- Easily Incorporate Third Party Security Solutions

Goals for DOCsec

- Unified Solution for Distributed Security
  - Security Cuts Across All Level of Enterprise
- Consistent Security Architecture
- Integrate Various Solutions to Give Corporations the Choice of Best of Breed
- Support for Frequent Change of Security Products
  - Distributed Security is Improving, i.e. Changing Rapidly

Quadrāsis™

End-to-End Security



Perimeter Security

- Today
  - It’s Mostly Proprietary
  - Rely Heavily on Perimeter Firewall
- Specification Work Under Way
  - SAML, XKML, dSig, XACML, XML Encryption
- Perimeter to Mid-tier & Mid-tier to Legacy-tier Boundaries
  - Neglected Today
  - Important in e-Commerce
- Today’s Browsers are Too Dumb re: Security

Problem with Browsers

- Designed to Work with HTTP
- Use SSL for Security
- Stateless
  - Want Session for SingleSignOn (SSO)
    - Cookies
- SSO Uses Cookies to Save “Credentials”
- Browser Can be Enticed to Give Up Cookies

Security Assertions Markup Language - SAML

- Framework for Defining XML Based Security
  - XML Document Format
    - XML Self Describing Data
  - Protocols
- Meant to Work with
  - Other Specifications, e.g. XACML, DSIG, SOAP, XML Encryption
  - Third Party Security Services
- SAML 1.0 Spec. to OASIS - Feb. 2002
- SAML v 1.0 An Important First Step
  - Standard and Consistent Security “Credential” Throughout All Tiers

SAML Specification Defines

- Assertions Which Contain
- Authentication Statement
  - Assertion by Trusted 3rd Party That the Subject of the Assertion has been Authenticated
- Attribute Statement
  - Assertion by Trusted 3rd Party That The Attributes Are Attributes for the Subject of that Assertion
- Authorization Statement
  - Assertion by Trusted 3rd Party That The Authenticated Subject with the Asserted Attributes Is or Is Not Authorized For a Given Action on a Resource
- Request/Response XML Message Exchange

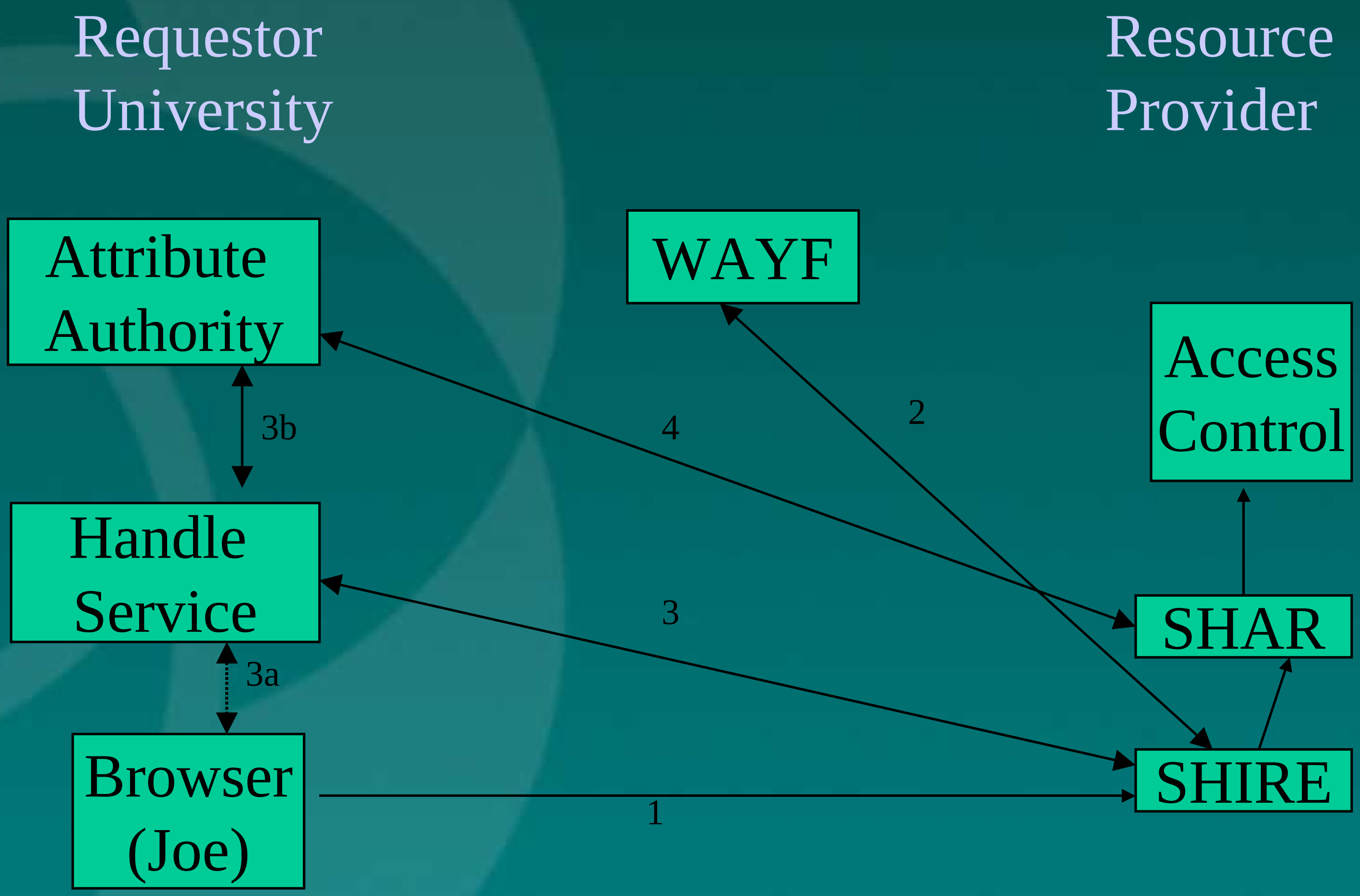
Simple Object Access Protocol (SOAP)

- SOAP is an XML Message Format Sent Over HTTP(S)
  - Other Communication Protocol May be Defined
- SOAP Message Consists of
  - Envelope
  - Header
    - Security Data Lives Here
  - Body
    - Contain RPC calls in an XML Format
- Securing the SOAP Document Itself
  - Digital Signatures
  - XML Encryption
- Transmit Authentication and Attribute
  - SAML Assertions in SOAP Header
  - Authorize the RPC requests

Shibboleth

- Joint Project of IBM & Internet2/MACE
  - Secure Interop Between Educational Institutions
    - Federation Problem
  - Security Data Management Problem
    - Managing Cooperating Universities' Users (Password or Certs) and Attributes by Target Administrator
  - Solution – Leave Administration to Source Site
    - Need to Securely Transfer Attributes From Source Site to Resource Provider Site
  - Solutions to Other Problems
    - Privacy
    - User's Choice of Attributes to be Used for a Given Target
    - Single SignOn (SSO)

Shibboleth Con’t



Mid-tier Security

- Leading Mid-tier Distributed Systems
  - Java
    - J2SE, J2EE
  - CORBA
  - Win2k, DCOM
- Business Systems Live Here
  - Multiple Interactions, Thus Complexity
- Neglected with respect to Security
  - Vulnerable to Insider Attacks
  - Outsiders Become Insiders

Quadrāsis™

JAVA Security

- Java 2 Platform Standard Edition (J2SE)
  - Applets
  - Java Client & Server Applications incl’d
  - APIs for Application Level Security
- Java 2 Platform Enterprise Edition (J2EE)
  - Defines the Enterprise Services
  - Enterprise Java Beans (EJB)
  - Server-side Security
    - Container Enforced Security
- Different Teams from SUN on Each
  - Different Emphasis
  - Not Complete Security Compatibility but Getting Better

Quadrāsis™

J2SE

- Original Sandbox Model
  - Local Code is Trusted
  - Downloaded Code is Not Trusted
  - General Security Characteristics (version 1.0)
    - Code Type-safe; bounds checking, memory management, garbage collection
    - Byte Code Verifier
    - Classloader Defines a Local Namespace
    - JVM Mediates Access to System Resources
    - 1.1 Introduced Signed Applet
      - Treated as Local Code if Signature Recognized

J2SE (Con't)

- Evolving Sandbox Model
  - JAAS
  - Protection Domains; System, Application, Principal
  - Configurable security policy
  - Extension of security checks to all Java programs
  - Policy File
    - Key Store
    - Grant <user> Permission name <target action signed-by>
    - grant principal javax.security.auth.x500.X500Principal "cn=Alice" { permission java.io.FilePermission "/home/Alice", "read, write"; };
  - Application Security is Left to the Programmer
    - Uses doPrivileged Method

Enterprise Java Beans (EJB)

- Has It's Own Specification (EJB 2.0)
- Client Accesses Application Server
- Principal Identity Transmitted to Container
- Principal is Authenticated
  - Web Server &/or
  - EJB Container
- EJB Container Checks Access Control
- EJB Container Permits or Deny Access
- Incorporates JAAS
  - Supports Kerberos

EJB Container Security

- Two Security Models
- Intra Container Security
  - Deployment Descriptor
    - XML Directive to the Container
    - Scaling Problems
- Inter Container Security
  - Common Secure Interoperability - CSIv2
    - Security Wire Protocol

Quadrāsis™

EJB Players

- Bean Provider
  - Minimum Security Involvement
    - Avoid Application Level Security
    - Push Security Down into the Middleware
    - Two “Escape” Container Security Calls
- Application Assembler
  - Define Logical Security Aspects of Bean
- Deployer
  - Adjust Security to Specific Environment

EJB Authorization

- Deployment Descriptor (DD)
  - XML Document Used by the Container
  - EJB DD XML Schema Defined by Specification
- Method Permissions
  - What Methods Can be Accessed By What Role
  - Improvement over ACLs
    - Use Wild Cards; Handle Overloaded Methods
    - Control Access to Naming
  - Still have Scaling Problems
    - Each Method Must Be Covered By a Method Permission

Delegation

- Two Flavors
  - Impersonation
    - Giving Some Entity the Right to Act as You
  - Restricted Delegation
    - Restricting Who Can Act as Delegatee
    - Restrict Actions They Can Perform (Future)
- Expressed in EJB through DD
  - Just Impersonation
  - Two Elements
    - use-caller-identity
    - runAs-specified-identity
  - Effects the Called Bean or Container
- Delegation Can Be Dangerous

Quadrāsis™

Web Services

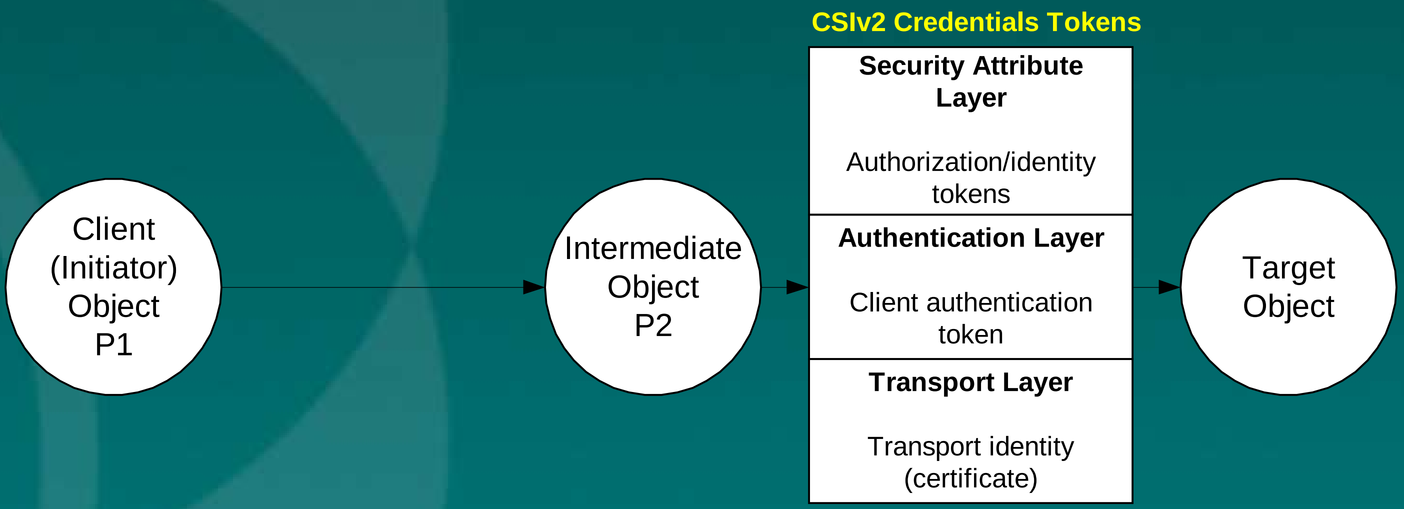
- Consist Of
  - Simple Object Access Protocol (SOAP) over HTTP
  - Universal Description, Discovery and Integration Service (UUID)
    - Trader & Registration Service for Business
    - Layered Over SOAP
    - White, Yellow, Green Pages
  - Web Services Definition Language (WSDL)
    - XML-grammar for dynamic cross-platform integration
- SAML Integration with Web Services
  - SOAP Uses SAML
  - dSig used to Sign SOAP Messages
  - XML Encryption to Encrypt Messages

Quadrāsis™

Common Object Request Broker Architecture(CORBA)

- Very Extensive Security Model
  - Specification – 400+ pages
- Defines Solutions for All Major Security Points
- Integrated with EJB re: EJB 2.0 Specification
  - Not yet Implemented by Major Container Providers
- Uses Internet Inter-ORB Protocol (IIOP)
  - Transmits Security Credentials in Service Context
- Common Secure Interoperability (CSIv2)
  - Also Mandated by EJB 2.0

Common Secure Interoperability (CSIV2)



Win2k Security

- Many Window's Only Security Models
- Strongest Based on Kerberos
  - Originated at MIT
  - Implemented by DCE Open Group
- Uses Remote Procedure Call (RPC)
- Active Directory
  - Supports Domains
    - In Turn Supports InterDomain Trust Relationships
- MS Pass Attributes in Kerberos Credential
  - Non-Standard

Kerberos (Simplified)

1 $c,tgs,time_{exp},n$

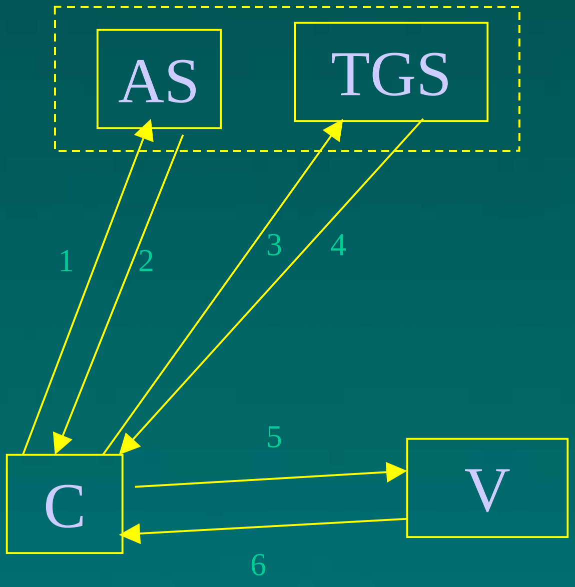
2 $\{K_{c,tgs},tgs,time_{exp},n,\ldots\}K_c,\ \{T_{c,tgs}\}K_{tgs}$

3 $\{ts,\ \ldots\}K_{c,tgs}\{T_{c,tgs}\}K_{ts},v,time_{exp},n$

4 $\{K_{c,v},v,time_{exp},n,\ldots)K_{c,tgs},\{T_{c,v}\}K_v$

5 $\{ts,ck,K_{subsession},\ldots\}K_{c,v},\{T_{c,v}\}K_v$

6 $\{ts\}K_{c,v}$(optional)



http://www.isi.edu/gost/publications/kerberos-neuman-tso.html

.NET My Services

- Moving Towards the XML Model
  - All Services are XML Web Services
- Kerberos Based Authentication
- . NET Passport
  - Authentication
  - Store Private Information
- SSO With Partners of .NET
- Sun et al - Liberty Alliance Project
  - As Yet - Vaporware

Legacy Tier

- Least Integrated
- Problem
  - Proprietary Security
  - No Knowledge of the Initiator
  - Usually Requires a Password From Initiator
- Need Proof of Initiator Authentication
  - May Have Traveled Through Many Servers
  - Often the Password Passed in Clear
- A Trusted Mid Tier Process
  - Makes the Request
  - Uses Delegation

Solution for DOCSec

- Multiple Lines of Defense in Each Tier
- Use Middleware Security
  - Security Unaware Applications
  - Security Policy Administratively Controlled
- Flexibility Towards New Solutions
  - New Requirements
  - Bad Guys Are Not Asleep
- Prerequisite – Host Based Security

An Architecture for Integrating Security across CORBA, J2EE and Web Services

Quadrāsis™

Ted Burghart

Ted.Burghart@Quadrasis.com

Introduction

- With the rapid pace of product development focused on Web Services and the abundance of security technologies being deployed throughout the enterprise, integration of these disparate products and services becomes increasingly important in order to maintain a manageable level of complexity.

Administrators, unable to manage the incompatible security technologies in any other way, resort to replication of data and effort across multiple security datastores.

Auditors are unable to establish the trustworthiness of the complete system.

Security Fundamentals

- Authentication
  - Proving the principal’s identity
  - Multi-factor authentication
- Authorization
  - Controlling access to protected resources
- Accountability
  - Who got access to what, when
  - Why - how the decision was made
- Availability
  - Rejection of attacks
  - Access to authorized clients assured

Quadrāsis™

System Requirements

- Administration
  - Common control point for most, if not all, policies
- Availability
  - Failover
  - Redundancy
  - Fault tolerance
- Scalability
  - Statefull or stateless?
- Standards Support
  - JCP
  - OASIS
  - OMG
  - W3C
  - The Next Big Thing ™

Platform Requirements

- Middleware
  - COM
  - CORBA
  - J2EE
  - Web Services
- Operating System
  - Unix
  - Windows
- Programming Language
  - C++
  - C#
  - Java

What do they have in common?

- Authentication
  - Translate evidence to credentials
  - Attributes
    - Translate credentials to privileges
- Authorization
  - Push vs. Pull
  - Granularity
- Is that it?
  - If it were only that simple …

Quadrāsis™

How do they differ …

- Authorization
  - Push vs. Pull
  - Granularity
    - Users
    - Groups
    - Roles
    - Privileges
    - Realms
    - Servers
    - Objects
    - Methods
    - Domains
    - Transports
    - Directories
    - Files
- Attribute Services
  - Translate some input to some output
- Accountability ?
- Administration ?
- Availability ?

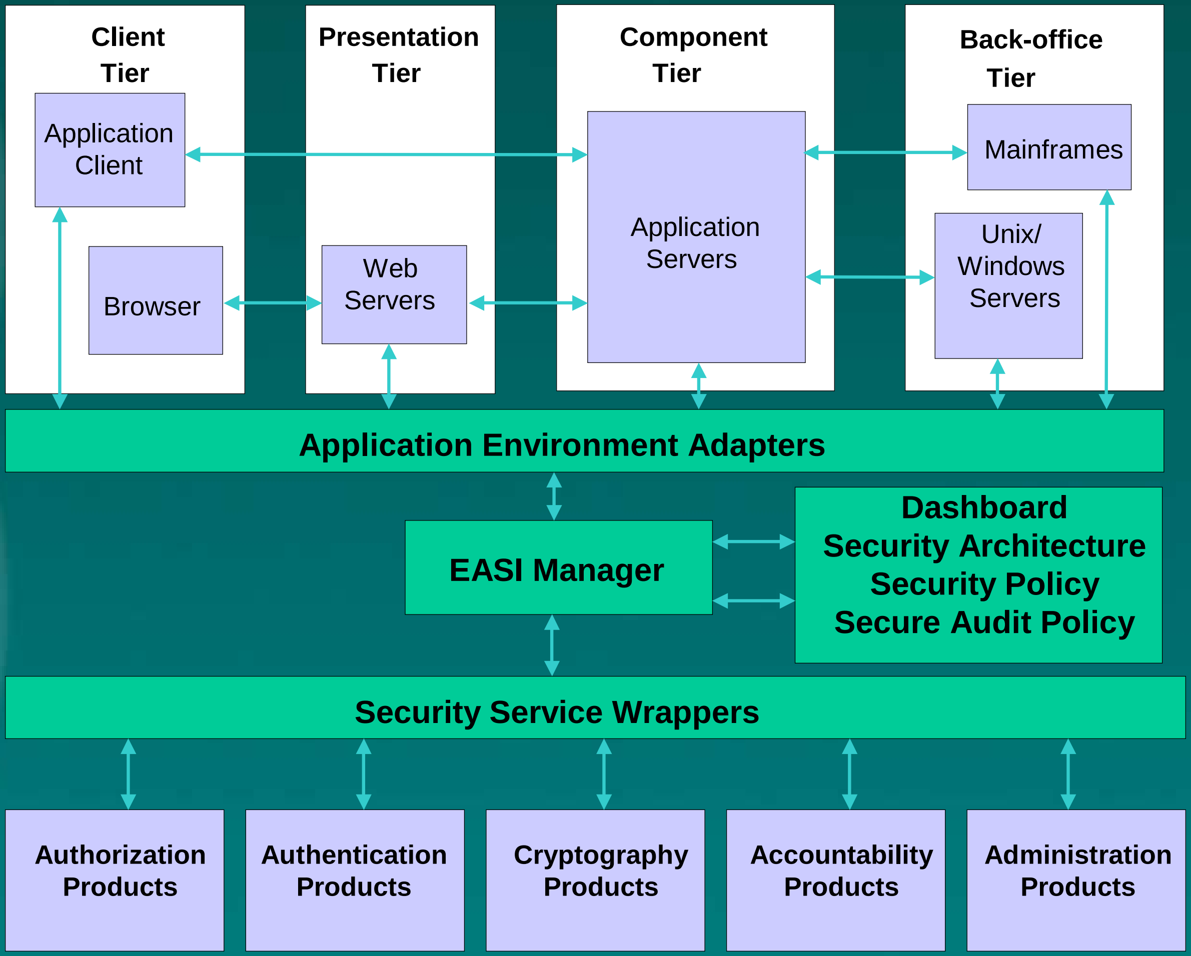
… and what can we do about it?

- Mapping
  - Credentials
  - Attributes
  - Decisions
  - Auditable events?
- Correlation
  - Event stream
- Policy
  - Is it even possible?
  - Ok, is it feasible?

Quadrāsis™

Enterprise Application Security Integration Framework

- Programming (APIs) – Collection of modular security service interfaces
- Policy – Security dashboard manages policies across security products
- Protocols – Driven by Security Assertion Markup Language (SAML) standard
- Products – Application environments and security services built by third parties and Quadrasis

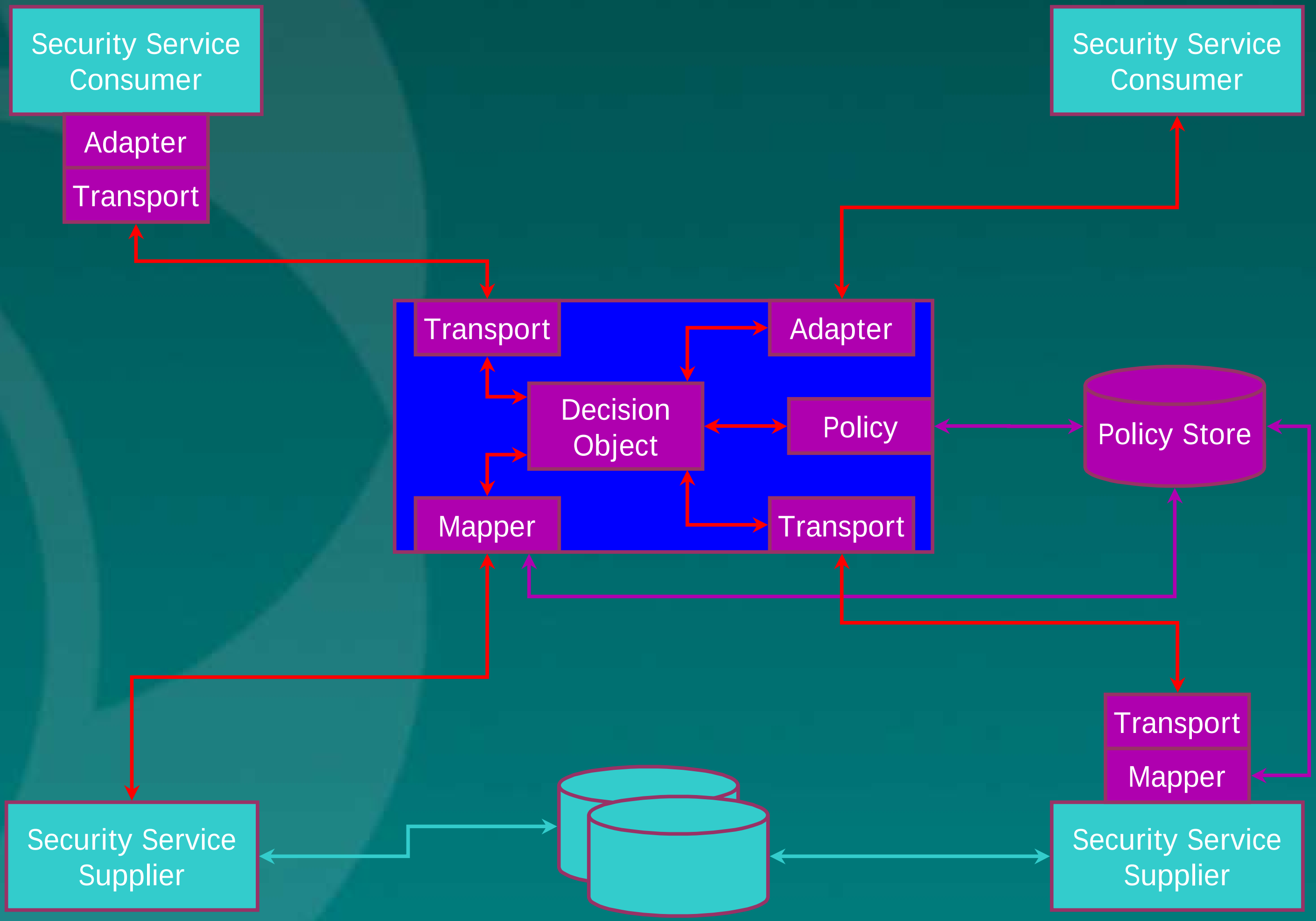


Quadrāsis™

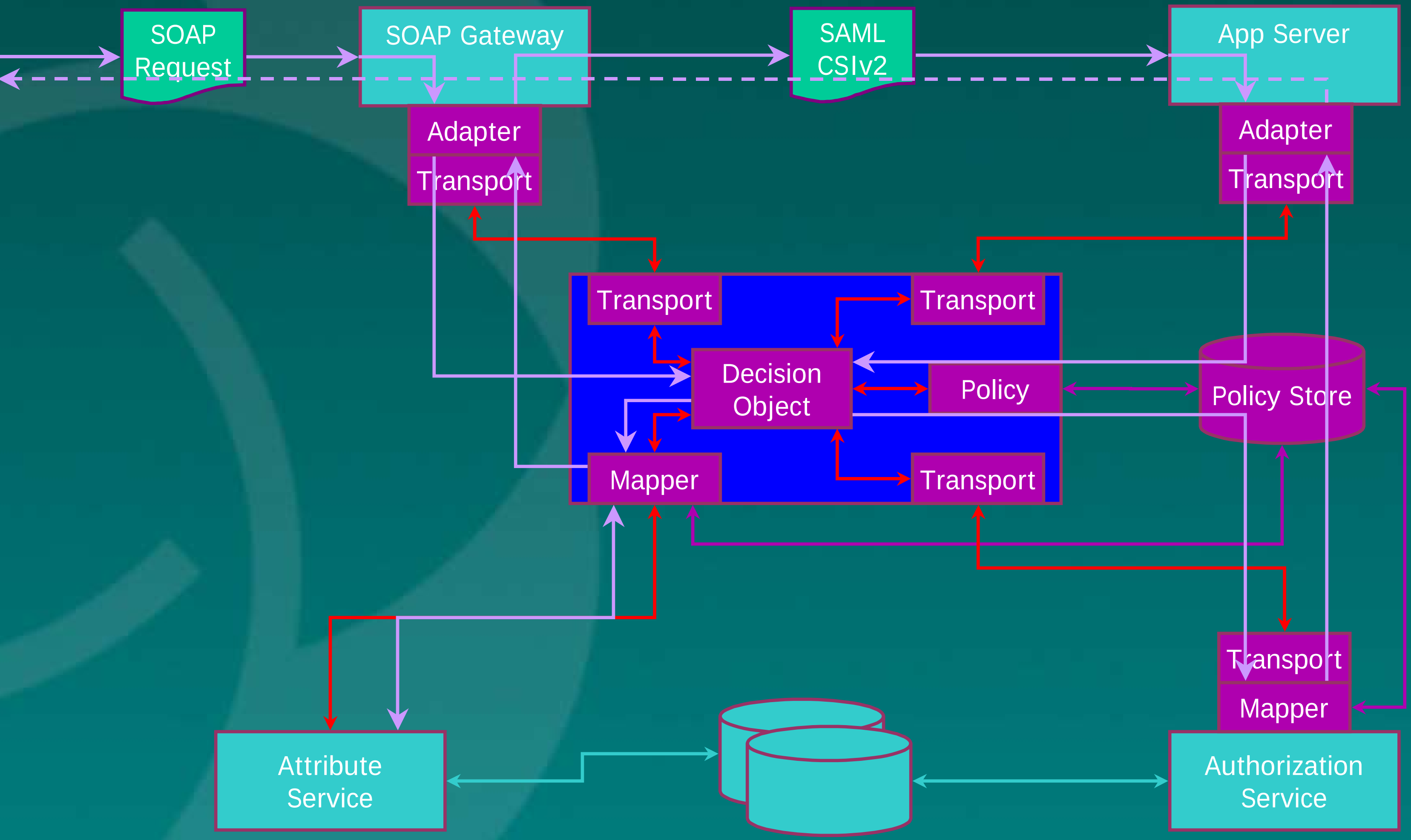
QPiK Plugin Architecture

- Loadable Services
  - Adapters
  - Mapping
  - Reporting
  - Routing
  - Translation
  - Transport
- Configuration
  - Transparency
  - Dependencies

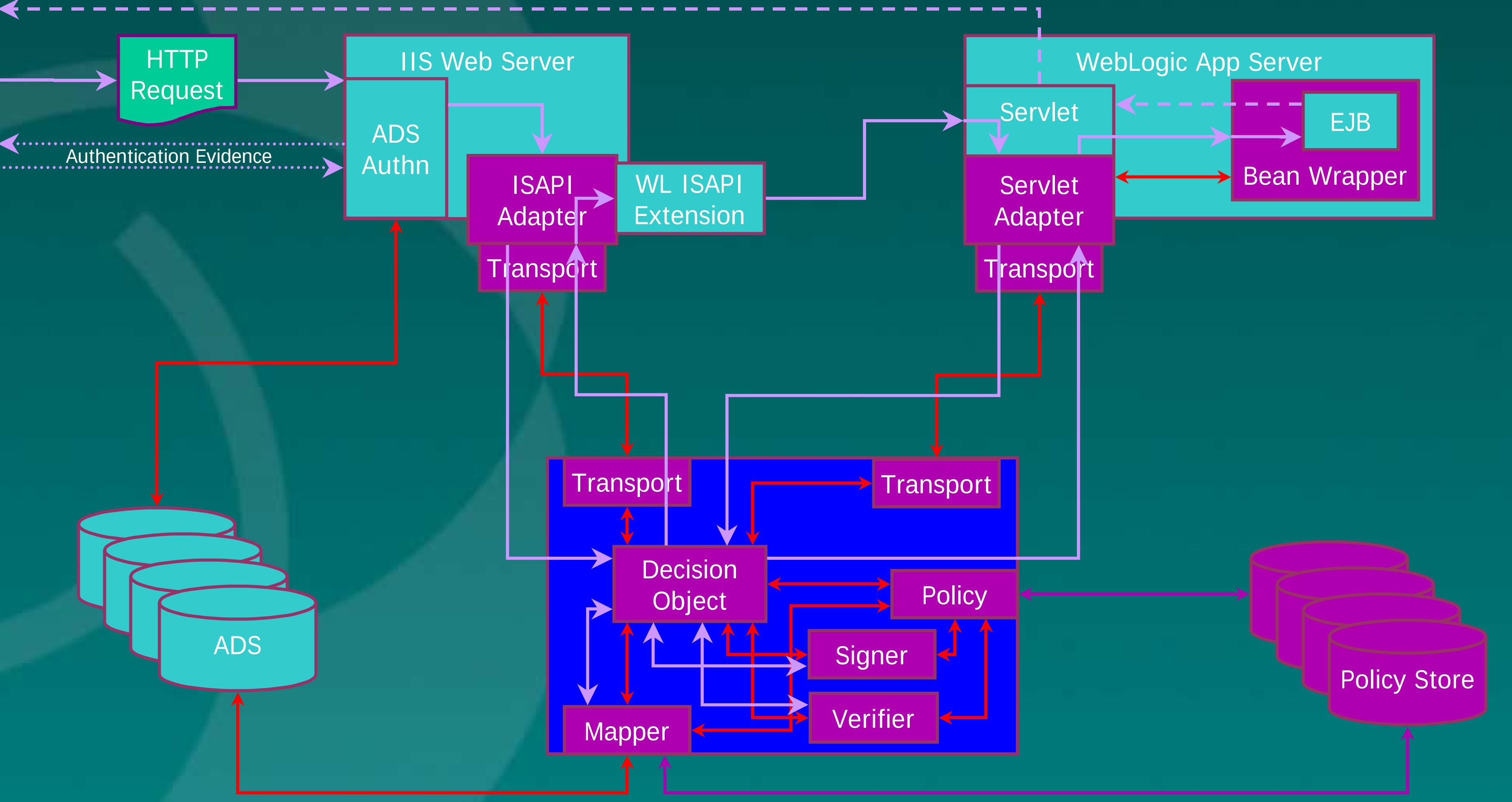
EASI Architecture


Security Service Consumer
Adapter
Transport
Security Service Consumer
Transport
Adapter
Decision Object
Policy
Mapper
Transport
Policy Store
Transport
Mapper
Security Service Supplier
Security Service Supplier

One Deployment Scenario ...



IIS/WL Deployment Scenario



Quadrāsis™

Trusting the trust decision

- Can we trust it when we're done?
  - Consolidated configuration
  - Secure transport
  - DSIGs on messages
- But …
  - Can't trust anything more than the least-trusted component
    - In many cases, that's the transport layer!
    - Where's the private key for *your* SSL server?
    - Where's the key encryption key for your private key?
  - DSIGs on decision modules
    - And where do the keys for that come from?