Integrating Web Services into the Undergraduate Computer Science Curriculum

Alfred C. Weaver  
Department of Computer Science  
University of Virginia

1. Project Summary

This project will translate our research experience from a Microsoft Research-funded project (private and secure storage, access, and transmission of medical data) into a mainstream undergraduate web services course for computer science and computer engineering majors. Using a project-oriented seminar format, we will have students design, implement, publish, and validate web services that interface biometric devices, implement authentication services, make access authorization determinations, and create, distribute, and validate trust credentials in the larger federated enterprise. In addition to reviewing case studies, each student will create two web services—one that provides authentication services via a biometric device, and another (personalized to fit the student’s specific interests) that implements a useful web service in a medical, financial services, digital government, homeland security, or similar modern context.

All pedagogical materials developed will be treated as “open source” and will be publicly available. Dissemination will be accomplished via our public website, our annual participation in the SIGCSE, FIE, and ITiCSE conferences, and inclusion of our materials in the Microsoft knowledge base.

2. Research to be Incorporated

Microsoft Research is currently supporting a research project at the University of Virginia entitled "Federated, Secure Trust Networks for Distributed Healthcare IT Services" [1]. David Ladd is the project sponsor. The goal of this project is to develop a prototype information management system that is HIPAA-compliant [2] and that addresses these research issues:

- Access to patient information is now restricted under federal law. Electronic access to that information requires reliable identification of the individual seeking access. We must accommodate a range of identification technologies ranging from legacy, low-security techniques such as passwords to modern, higher-security systems such as fingerprints, iris scans, signature and voice recognition, and other biometric and non-biometric approaches.
- When any authentication technology identifies an individual, we must associate with the resulting identity token a "trust level" that is reflective of our confidence in that authentication. In the abstract, the trust levels $T()$ of identification techniques are ordered [3] (e.g., $T(\text{retina scan}) > T(\text{iris scan}) > T(\text{fingerprint scan}) > T(\text{password})$), although the $T()$ of any particular authentication device is implementation-dependent and can only be determined by field trials to assess its false accept and false reject ratios.
- Legitimate access to patient information will occur from wireless devices such as Tablet PCs, laptops, cell phones, and Pocket PCs. We must be able to authenticate user access from mobile devices with a reliability level equivalent to that of on-premises wired devices.
- When a person makes a service request (e.g., access to the electronic patient record, scheduling a diagnostic procedure) we must determine whether that individual, whose identity has been established with trust level $T()$, is allowed to perform the requested operation at this moment in time. Authorizations are always time-dependent as well as identity-dependent. We are using extensions to role-based access control [4,5,6] and a dynamic authorization rule engine to arbitrate access to data. Developing authorization rules for individuals is easy; handling groups is more difficult because groups have dynamic membership, complete with temporal exceptions. For example, there are multiple classifications of physicians (e.g., attending, referring, consultant, resident, fellow, student), each with different access privileges; there are multiple issues with group membership (e.g., doctors in equivalence classes such as surgeons or radiologists, one doctor temporarily covering for another); and there are some rare but important special cases (e.g., public health emergencies, war).
• Electronic transmission of data will occur across the public Internet (e.g., sending electronic prescriptions to pharmacies, filing insurance claims). HIPAA requires that sensitive data must be strongly encrypted and that the source and destination of all data transmissions must be known reliably.

• Data exchange outside the medical center network implies a system of federated trust. In the same way that ATMs from different banks share trust credentials, we must be able to share trust credentials across disparate parts of the medical enterprise (hospital, clinic, pharmacy, insurance, billing). We must be able to establish the identity of enterprises and people, and we must be able to verify the integrity of transmitted data, at a level of reliability that meets all legal standards. For example, a pharmacy must be able to verify an electronic prescription to a degree that would allow it to fill an order for narcotic drugs without fear of forgery; an insurance company must be able to verify the source and integrity of submitted claims.

The design of a distributed, federated, secure system for medical data exchange is a complex problem. Our research approach envisions the medical enterprise as a federated system (a cooperating collection of heterogeneous subsystems) and we implement it as a collection of web services. The web services model utilizes a loosely-coupled, language-neutral, platform independent framework for designing, publishing, promoting, registering, and initiating processes in a distributed environment—and this is the future of all new distributed computing systems, medical or otherwise.

3. Bodies of Research to be Integrated

This proposal integrates three bodies of research knowledge and experience:
(1) the form and function of web service discovery and data exchange;
(2) the seven main web service security components whose use is required to achieve the necessary level of privacy, security, and reliability; and
(3) our own experience in developing and fielding web services that provide the functionality required in the previously described medical scenarios.

3.1 Web Service Discovery and Data Exchange

The core components for web service discovery and data exchange that the student must master are:
• eXtensible Markup Language (XML) [7] – All data are exchanged in this universal SGML-style language that enables semantic tagging of the data. Semantic ontologies are being developed for specialized applications such as EM-XML for emergency medicine, ebXML for e-business, and MathML for mathematics.
• Simple Object Access Protocol (SOAP) [8] – SOAP is an envelope containing a message that is itself in another specific vocabulary such as HTTP or Java Message Service. It uses XML structure to create request/response messages and to hide application technology from users and other services.
• Web Services Description Language (WSDL) [9] – WSDL is an XML format for describing web services as end points that act on messages containing either documents or procedure calls. It describes the service, including who operates it, where it is located, and how it is accessed.
• Universal Description, Discovery, and Integration (UDDI) [10] – UDDI facilitates describing and discovering web services through the registration of service information. UDDI is sometimes referred to as the "yellow pages" of Internet software services.

3.2 Web Service Security

The building blocks of the web service security strategy are shown in figure 1. The foundation is the Simple Object Access Protocol (SOAP) that provides a uniform messaging service among web services. The WS-Security [11] specification enhances SOAP messaging to provide content protection through message integrity and message confidentiality; it describes how to attach signature and encryption headers and defines how to include security tokens (e.g., X.509 certificates, Kerberos tickets) within SOAP messages. Above WS-Security lie six more specifications, each providing additional functionality that is being defined by the World Wide Web Consortium,
the W3C [12]. We utilize the web services security architecture jointly recommended by IBM and Microsoft in [13].

<table>
<thead>
<tr>
<th>WS-SecureConversation</th>
<th>WS-Federation</th>
<th>WS-Authorization</th>
</tr>
</thead>
<tbody>
<tr>
<td>WS-Policy</td>
<td>WS-Trust</td>
<td>WS-Privacy</td>
</tr>
<tr>
<td>WS-Security</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SOAP</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 1. Web Services Security Strategy [13]

The Web Services Policy Framework [14] component describes the capabilities and constraints of the security policies on both intermediaries and endpoints. For example, it could specify whether security tokens are required, which encryption algorithms are supported, and which privacy rules are to be employed. The Web Services Trust specification [15] describes the framework for trust models such that web services may interoperate securely. The Web Services Privacy component, currently implemented as the Platform for Privacy Preferences Project 1.0 Specification (P3P1.0) [16], provides a model for how privacy preferences and organizational privacy practices are conveyed.

The Web Services Secure Conversation specification [17] details how to manage and authenticate message exchanges among parties. This includes the establishment and derivation of session keys. WS-Federation [18] explains how to manage and broker the trust relationships in a heterogeneous federated environment. WS-Authorization (forthcoming) will describe how to manage authorization data and policies.

Once the student has mastered the web services framework, its tools, techniques, languages, and standards, and then its multi-component security strategy, the student is ready to write useful web services.

3.3 Research Experience

Our research project currently has an operational electronic patient record with six components (demographics, patient history and symptoms, diagnosis and treatment plans, medical imagery, lab results, and psychiatric profile). All access from humans is achieved through a common medical data portal that self-configures to display the information needed (and allowed) by that individual once identity is established and the required trust level as been achieved. For example, a physician who has previously authenticated with a password and whose access device provides an identity token (cookie) to the medical portal can see the general set of services available to physicians, but no private details. If the physician then attempts to access a patient record the authorization web service will deny the request for lack of a sufficiently high trust level. In this example, hospital administrators have issued a rule that requires (a) identity to be established with a trust level of T(fingerprint) or higher and (b) the role of the requestor must be that of an attending, consulting, or referring physician). The authentication web service will offer the user the opportunity to re-identify himself or herself using whatever authentication technologies are available to the user (e.g., the built-in fingerprint scanner on an HP h5550 iPAQ). If the user provides a fingerprint (or any higher-trust attribute such as an iris scan) then the authentication server will validate the biometric and issue a new identity token (cookie) with a higher trust level.

We are developing an authentication web service that can determine identity, establish trust levels, and issue identity tokens for:

- Username and password
- Fingerprints using the Digital Persona U.are.U fingerprint scanner [19]
- Fingerprints using the HP h5550 iPAQ Pocket PC thermal fingerprint scanner [20]
- Iris scans using the Panasonic/Iridian Authenticam iris scanner [21]
- Electronic security keys using the Aladdin Knowledge Systems eToken (an encrypted USB device) [22]
- Signature recognition using Communication Intelligence Corporation's iSign software on a Tablet PC [23]
- Key fobs using the RSA SecurID unit (two-factor identification requiring a PIN and a pseudo-random number that changes every minute) [25]
- Combinations of other "who-you-are," "what-you-have," and "what-you-know" technologies.

We are implementing the authentication rule engine and rule database and a generic, secure interface for "outside" (off our network) entities such as a pharmacy. Strong encryption (AES-256 [26]) is used throughout. Electronic prescriptions are protected by encryption and validated via digital signatures.

4. Rationale for this Project

Undergraduate CS students rarely develop a distributed system with the complexity of the research project just described, and yet complex distributed systems are exactly what industry and government are building (e.g., financial services, digital government, homeland security). We want to move web services technology into the undergraduate curriculum. We propose to do that by creating an undergraduate seminar course that covers the basics described above (web services framework, tools, techniques, languages, standards, and security strategy), conducts case studies of successful implementations, and then challenges the student to create useful web services in a motivating context.

5. Goals and Objectives

(a) Each student masters the web services framework concept, the web services required for discovery and data exchange, and the seven web services needed for data privacy and security.  
(b) Each student experiences the design, implementation, validation, publication, and demonstration of a web service that implements a personal authentication task and of another web service that implements a portion of a larger application.  
(c) All pedagogical materials developed are maintained and shared from a common website. Our intent is that all material is treated as "open source" and that users will retrieve modules, use and improve them, and return the new and updated material to the common repository.  
(d) We will disseminate our work through our website; by annual reports at the SIGCSE, FIE, and ITiCSE conferences; by participation in the Microsoft Faculty Summit if so invited; and by linking our materials to the appropriate Microsoft libraries.

6. Course Outline

1. Research motivation—the need for authentication and authorization web services.  
2. Case studies taken from medical, financial, digital government and homeland security applications.  
3. Web services concept, framework, and tools.  
   a. Players and points of view: Microsoft, Oasis, Liberty Alliance  
   b. Our focus: Microsoft .NET and C#  
4. Web service discovery and data exchange  
   a. XML, SOAP, WSDL, UDDI  
5. Web service security architecture  
6. Authentication  
   a. Biometrics (fingerprint, iris scan, signature, voice, body geometry)  
   b. Digital (e-tokens, RFID, key fob)  
   c. Legacy (passwords, PINs, physical keys)  
7. Authentication rule engine  
   a. Design, authentication token, trust levels, reliability, encryption, digital signatures  
8. Authorization rule engine  
   a. Design, role-based access control, extensions to RBAC, fine-grained objects, meta-objects, rule database, performance issues
9. Federated Trust
   a. Trust credential design, content, establishment, certification; trust sharing; trust negotiation

10. Putting it all together in a working web services context

11. Demonstration of the resulting system

7. Timeline

Summer 2004: The PI and one graduate student will design the research seminar course and develop the instructional material needed for an undergraduate seminar on web services (framework, tools, techniques, languages, standards, security, and case studies). We will identify appropriate case studies and develop tutorial exercises that illustrate how a student should design, implement, test, publish, access, and utilize a web service that solves a particular problem.

Fall 2004: Revise two weeks of our extant CS453 Electronic Commerce Technologies course, taught each fall semester, to cover web services. Test some of the materials to be used in the research seminar course by using them as homeworks in CS453.

Spring 2005: Offer the full web services course as a research seminar, open to fourth-year undergraduates and first-year graduate students. Expected enrollment is 20-30 people.

8. Budget

<table>
<thead>
<tr>
<th>Year 1</th>
<th>6/1/04--5/31/05</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>A. Personnel</strong></td>
<td></td>
</tr>
<tr>
<td>1. A.C. Weaver, PI</td>
<td></td>
</tr>
<tr>
<td>25% effort, three months, 6/1/04—8/31/04, and 10% effort, nine months, 9/1/04-5/31/05</td>
<td>20650</td>
</tr>
<tr>
<td>2. Graduate Research Assistant</td>
<td></td>
</tr>
<tr>
<td>50% effort, one year, 6/1/04—5/31/05</td>
<td>18400</td>
</tr>
<tr>
<td>Subtotal Personnel</td>
<td>39050</td>
</tr>
<tr>
<td><strong>B. Fringe Benefits, 24.5% of Item A1</strong></td>
<td>5059</td>
</tr>
<tr>
<td><strong>C. Publications</strong></td>
<td>300</td>
</tr>
<tr>
<td><strong>D. Other Direct Costs</strong></td>
<td></td>
</tr>
<tr>
<td>1. Technical services—departmental computing services</td>
<td>3000</td>
</tr>
<tr>
<td>2. Other services—copying, communications</td>
<td>300</td>
</tr>
<tr>
<td>3. Tuition remission—in-state graduate students</td>
<td>8497</td>
</tr>
<tr>
<td>4 Travel -- SIGCSE, FIE, ITiCSE, MS Faculty Summit</td>
<td>5000</td>
</tr>
<tr>
<td>Total Direct Costs</td>
<td>61206</td>
</tr>
</tbody>
</table>
9. Synergy

An expanded version of this proposal, including a faculty partner at Longwood University, more development effort, and an academic evaluation plan has been submitted to the National Science Foundation’s CISE Educational Research and Curriculum Development program [27].

10. References