Federated, Secure Trust Networks for Distributed Healthcare IT Services

Alfred C. Weaver, Samuel J. Dwyer III*, Andrew M. Snyder,
James Van Dyke, James Hu, Xiaohui Chen, Timothy Mulholland, Andrew Marshall
Department of Computer Science
*Department of Radiology
University of Virginia
Charlottesville, VA 22904

Abstract - **Recent federal regulations mandate the privacy and security of healthcare data at a level never previously contemplated; compliance with those requirements will require a complete rethinking of how data is utilized, stored, and transmitted. To better understand these issues in the context of a modern distributed system, our research group is developing a prototype healthcare IT system and medical data portal based upon a web services approach consistent with Microsoft's .NET framework. An authentication web service manages trust levels, issues authorization tickets, and uses biometric devices to establish identity; an authorization web service determines what data may be accessed, in what way, and by whom. Hospital administrators set access privileges for recursively-defined groups, subgroups, and individuals. All patient records and medical images are protected using AES encryption with 256-bit keys. Off-network entities such as pharmacies, insurance companies, and other medical service providers participate through a federated trust-sharing arrangement. Electronic prescriptions are transmitted securely to participating pharmacies and pick-up notifications are provided to the patient using the preferred notification method (email, alerts to a PDA, automated telephone call) stored in the patient's profile.**

I. HIPAA REQUIREMENTS

Today's healthcare computer systems rely upon a disparate collection of legacy services, all developed separately and independently for patient records, radiological images, scheduling, billing, and administration. Recent federal laws and healthcare trends have prompted industry-wide efforts to integrate these distributed systems securely, thereby increasing operating efficiency, workflow, and the quality of patient-centered care. Effective in 2003, the federal Health Insurance Portability and Accountability Act [1] subjects all medical data to stringent privacy regulations; security regulations have likewise been published and will take effect in 2005.

To comply with HIPAA, encryption services will be needed for the storage and transmission of patient information; authentication services will be needed to verify the identity of those who access healthcare records; authorization services will be needed to programmatically determine what data rights are granted to which individuals and IT subsystems. Furthermore, to overcome the deficiencies of the existing patchwork system of healthcare systems, the entire medical IT enterprise needs to be accessed through a single portal that adapts itself to the needs of the doctor, staff member, patient, or allied enterprise (e.g., a pharmacy) that is engaged in authorized access. The portal must also accommodate and adapt to the type and means of access, e.g. large-screen desktop machines connected over an in-hospital LAN or small-screen Pocket PCs connected over an external wireless network.

The design of a distributed, federated security system for healthcare services is a complex issue. Legitimate access to healthcare web portals and web services will occur from a variety of devices (desktops, laptops, Pocket PCs, Tablet PCs, cell phones) and the authentication service must handle multiple identification technologies ranging from username/password to biometrics such as fingerprints and iris scans. The authorization rules must be programmable to respond to dynamic situations, and therefore this system requires a rule engine that can implement context-dependent authorization based upon identity and the data requirements of the requested task.

Our research project is using the Microsoft .NET framework to build a prototype healthcare IT system based entirely on the concept of web services. It is our intent to utilize existing web standards wherever possible, as well as to make our own contributions to emerging standards such as federated trust-sharing among dissimilar organizations.

II. WEB SERVICES

The modern approach to a problem of this magnitude is to envision the computing enterprise as a federated system (a cooperating collection of heterogeneous subsystems) and to integrate them via a collection of web services. As promoted by the World Wide Web Consortium [2], web services are seen as the preferential way to link applications both within and without an organization in a loosely-coupled, language-neutral, platform-independent way. "Web services" is a new model for distributed systems that enables designing, publishing, promoting, registering, and initiating processes dynamically in a distributed environment. While the utility of web services is couched in terms of convenient, location-independent access to data through web portals, this efficiency extends to other enterprises as well—for example, off-net entities such as pharmacies and insurance companies can access the web services without going through the

hospital's medical portal. Web services provide the common interface that will allow disparate systems to access data and services, either through the medical portal or directly, while ensuring the necessary (and soon to be federally mandated) data safeguards.

The fundamental components include:

- eXtensible Markup Language [3] is a data format description language from W3C.

- Simple Object Access Protocol [4] is an XML-based protocol that defines a vocabulary for electronic message exchange. SOAP is an envelope containing a message that is itself encoded in another specific vocabulary such as HTTP or Java Message Service [5]. It uses XML structure to create request/response messages and to hide application technology from users and other services.

- Web Services Description Language [6] is another W3C product. It 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 [7] facilitates describing and discovering web services and businesses through the registration of business identity information. UDDI is sometimes called the web services "yellow pages."

None of XML, SOAP, WSDL, or UDDI directly implements security; other proposed services are responsible for providing it. The six core security components are:

a) identification – which client (human or software) is making the request?
b) authentication – how does the system know the client is who it says it is?
c) authorization – is the client allowed to perform its requested task?
d) integrity – is the exchanged data reliable?
e) confidentiality – how does the system provide data only to authorized entities?
f) auditing – how does the system log all data accesses, both to fulfill the HIPAA requirement and to provide a traceable record in case of misuse?

III. WEB SERVICE STANDARDS

Multiple groups have been active in developing web service specifications and implementations. There are at least five key groups participating in the race to develop web security standards, some competitive and some cooperative.

- Microsoft has already released its Passport .NET service [8]. Passport is a generic authentication web service with built-in support for security concerns. After a user registers once, Passport acts as a proxy with cooperating entities (e.g., services such as calendaring software, companies such as the online travel service Expedia). Passport increases its user community daily because it simplifies trust sharing (in this case by supporting automatic login services by providing usernames and passwords) among cooperating system.

- The W3C has issued three XML-based standards:

  1. XML Digital Signatures [9] – digital signatures to authenticate a message's source, all done within XML.
  2. Encryption [10] – implements message privacy within XML.
  3. XML Key Management Services [11] – public key registration and validation.

- OASIS (Organization for the Advancement of Structured Information Systems) [12] is a not-for-profit, global consortium (600 corporate and individual members in 100 countries) that drives the development, convergence and adoption of e-business standards. OASIS promotes SAML, the Security Assertion Markup Language [13], an XML-based framework for exchanging security information, especially authentication and authorization. SAML provides a "single sign-on" for heterogeneous environments. SAML version 1.0 was released July 15, 2002.

- The Liberty Alliance [14] (primarily driven by Sun Microsystems, but with 62 active global business members as of July 2002 when its version 1.0 specification was released) seeks to establish an open standard for federated network identity through open technical specifications. These open standards support a broad range of identity-based products and services, allowing the customer to choose his identity providers, link accounts through account federation, utilize a single sign-on, and access a network of connected services and devices.

- WS-Security [15] extends and subsumes previous web service security specifications published individually and jointly by IBM, Microsoft, and Verisign. The specification defines a set of foundational SOAP extensions used to implement integrity and confidentiality. It describes how to exchange signed and encrypted messages in a web services environment, using multiple security approaches including the federal public key infrastructure [16], MIT's network authentication protocol [17], SAML, the eXtensible Rights Markup Language [18], Secure Sockets Layer [19], and others.

Although each of these approaches has its advantages, WS-Security is likely to become an enduring standard because it provides a flexible framework that does not lock the user into specific security choices; instead it interacts with multiple web service specifications above it, and SOAP below it, to implement a dynamic and programmable security strategy for arbitrary user applications. This is exactly what is needed in a distributed healthcare environment.

IV. WEB SERVICES SECURITY STRATEGY

The building blocks of the web service security strategy are shown in fig. 1. The foundation is the Simple Access Protocol (SOAP) that provides a uniform messaging service among web services. The WS-Security 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 [20], Kerberos tickets) within SOAP messages. SOAP and the Web Services Security standards are both defined and initial implementations have been fielded. Above WS-Security lie six more specifications, each providing additional functionality, that are being defined by W3C. An excellent introduction to the security architecture of web services is provided in a joint Microsoft/IBM whitepaper on that topic [21].

The Web Services Policy Framework [22] 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 [23] describes the framework for trust models such that web services may interoperate securely. The Web Services Privacy component will provide a model for how privacy preferences and organizational privacy practices are conveyed.

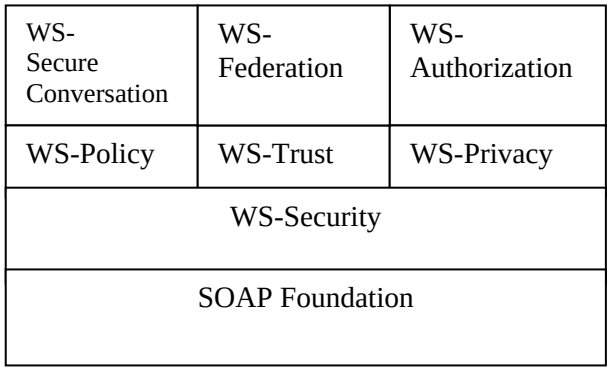

Fig. 1. Web Services Security Strategy

The Web Services Secure Conversation specification [24] details how to manage and authenticate message exchanges among parties. This includes the establishment and derivation of session keys. WS-Federation (not yet published) will explain how to manage and broker the trust relationships in a heterogeneous federated environment. WS-Authorization (not yet published) will describe how to manage authorization data and policies. Our prototype will implement this service as a programmable rule engine that can respond rapidly to dynamic changes in the permission access matrix.

V. AN EXAMPLE ENCOUNTER

The use of these various web services is best illustrated by example; see fig. 2.

Using an HP5455 PDA with built-in fingerprint scanner, Dr. Smith decides to order a new prescription for her patient Mr. Jones. She accesses the medical portal (arrow 1 in figure 2) with a request to access her patients' records. When the portal requests protected information from a web service, that web service requires that access to be associated with a certain trust level that is set by hospital administration. If the client already has a valid authentication ticket that meets or exceeds the required trust level, the portal obtains the requested data from the web service and displays it. However, if a valid authentication ticket does not exist, or if a ticket exists but contains a lower trust level than is required for the requested service, the client is redirected to a login page. Furthermore, the authentication ticket must not have expired (*leases* are used to implement time-dependent authentication), and the ticket must be signed by a trusted authentication web service. This provides a foundation for federated trust; even an authentication ticket signed by an outside authentication web service such as Passport might be satisfactory for lower levels of trust.

For this example, assume that this request is the doctor's first access of the day and hence no authentication ticket is provided. In general, the authentication service will now permit the physician to identify herself using some combination of username/password, fingerprint, iris scan, smartcards, or other supported identification modalities. In this case, the authentication service knows from the type of the access device which modalities are supported (username/password and fingerprints on the HP5455), and it knows from its authentication rule engine what level trust is required for the requested service. Hospital administrators define the trust level required for each service available, e.g., "access to patient records requires a trust level equal to or greater than that provided by fingerprints." Given that this access is coming from a PDA with fingerprint support, and that the rule set allows patient data access if identification is established via fingerprint, that is the identification modality required for Dr. Smith to gain access to Mr. Jones' records from a PDA. Had the access device supported iris scans, that modality would also have been an acceptable option because the trust level of iris scans exceeds that of fingerprints.

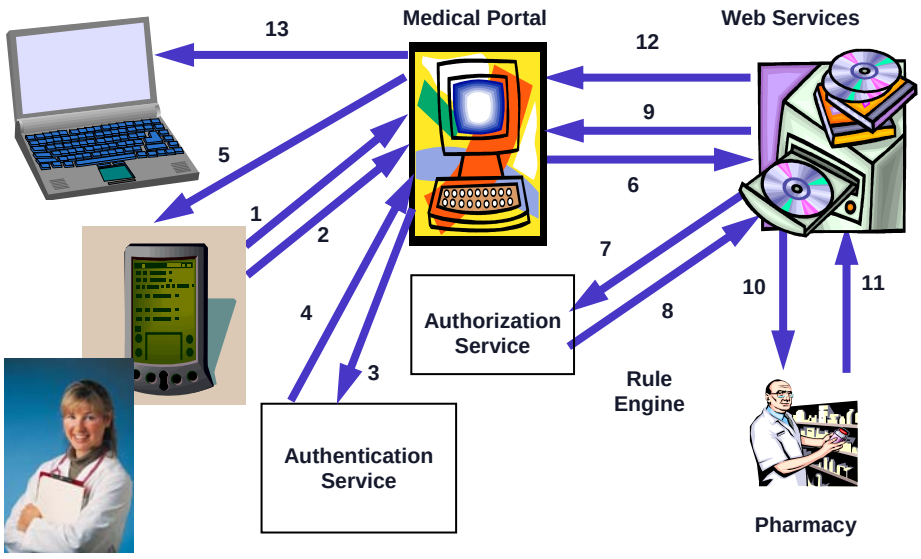

Fig. 2. System Architecture

Dr. Smith provides her fingerprint (2). The PDA transmits her fingerprint data to the web portal which in turn forwards it to the authentication web service (3); this service then attempts to confirm identity by comparing fingerprint data against its database of registered users. If identity is reliably established and if the trust level of the identification methodology equals or exceeds that required for the requested operation, then the authentication service so informs the portal (4), which then instructs the client to generate and store a cookie containing the authentication ticket data (5).

Although identity has now been established, there is still the question of authorization; Dr. Smith's request to access Mr. Jones' records (6) is forwarded to the authorization service (7). Dr. Smith's authorization to retrieve patient records can be changed at any time by hospital administration; therefore, authorization is always a dynamic operation and is always a hotspot in terms of the information dataflow. If the authorization rule engine determines that Dr. Smith is authorized to view Mr. Smith's records, then authorization is returned (8) and access to the data record is granted and the portal displays the requested data (9). After reviewing the patient record, Dr. Smith uses her PDA to write an electronic prescription that is forwarded (along with her authentication ticket) through the medical portal and appended to the patient record. Because this is a new prescription, the prescription data is also encrypted and then transmitted (10), along with the hospital's trust credentials, to the patient's preferred pharmacy (as determined from the patient's profile).

The pharmacy must first determine whether the "trust credentials" provided are sufficient for it to accept any prescription; the rules by which an outside entity (e.g., pharmacy, insurance) can express what credentials are needed and how they are to be interpreted is one of the open research questions. While the framework of shared trust is present in the WS-Trust specification, all the details are yet to be determined by the research community.

Assuming that the pharmacy chooses to trust the electronic transmission, standard digital signature techniques are used to assure the origin and integrity of the data, and then AES decryption techniques are applied to reveal the prescription. If the nature of the prescription requires it (e.g., narcotics), the pharmacy may make additional electronic inquiries (e.g., the physician's federal ID number) to further ensure the integrity of the process. When the prescription is filled, the pharmacy makes electronic notification, along with its trust credentials, to the hospital. If these trust credentials are accepted, the notification is decrypted and the patient records is updated (11), which in turn updates the medical portal database (12), which in turn sends a .NET Alert message to the patient (13) notifying him that the prescription has been filled and is awaiting pickup. The nature of the Alert message (e.g., email, telephone call) is controlled by the patient preferences stored in the patient profile.

VI. RESEARCH QUESTIONS

A. Authentication Service

The authentication web service is interrogated whenever a user needs to establish his or her identity. In times past, when all network access was wired and all accesses occurred from within the medical center, passwords or challenge-response approaches sufficed for establishing identity. But

as medicine progresses toward mobile devices, the opportunity for mischief, error, or misrepresentation increases. This risk can be mitigated by the use of biometric devices (e.g., fingerprint, iris and retina scanners, smartcards keyed to personal data, Subscriber Identity Module (SIM) for a PDA) and/or a requirement for multiple authentications.

Even so, the trust associated with an identification method varies based upon the method itself. For instance, username/password provides a weaker level of assurance than does fingerprints, which are in turn weaker than iris scans, which are weaker than retina scans, etc. Hospital administration is responsible for determining the trust levels associated with any particular type of identification technology, and for programming the authentication rule engine to record what trust levels are required to establish identity within any specific group (e.g., doctor, patient, technologist). The definition, specification, ordering, and management of trust levels is an open research question, but our use of trust levels is one of the unique characteristics of our approach to security. Federated trust (establishing and exchanging trust across independent domains) is an even harder problem, but one where our research project can make important contributions.

Assume an arbitrary mix of access devices and authentication technologies. How does one use WS-Policy to describe the level of authentication required as a function of the access device in use and that device's capabilities? For example, how might one define an institutional rule such as "access via wireless PDA requires a fingerprint scan if that device is available, or if not then it requires the insertion of the physician's unique SIM card and a typed password." Once an individual is authenticated, for how long should the resulting authentication ticket be valid?

B. Authorization Service

Like authentication, the authorization service must inspect and approve the access rights of the requester against the destination and nature of the data access. WS-Authorization is responsible for specifying the access rights of authenticated individuals. While access rights to patient records is one obvious example, there are additional access rights (e.g., scheduling examinations, ordering laboratory tests, reporting lab test results, filing diagnostic imagery) that extend beyond the physicians to cover nurses, technologists, and other hospital staff. How should those rights be encoded? Clearly, the cross-product of all staff members against all patients, procedures, and tests is impractical. Furthermore, some access rights should be based on groups (e.g., reading a CT can be done by any radiologist), whereas some should be based on specific individuals (e.g., the patient should receive his diagnosis from his primary care physician), and some are based on context (e.g., when the primary care physician is on vacation, his authority should be extended to his replacement temporarily).

C. Federation and Trust

Once a participant has been authenticated and authorized in the hospital system, how can that trust be represented and exported to other systems? For instance, in the example of the physician prescribing medication in the previous example, it would be impractical to have the physician physically re-authenticate with each and every off-network pharmacy for each and every prescription that is to be transmitted.

Various web sites, web services, and authentication web services have implemented different security techniques. How can federated trust be established between these disparate systems? How can digital signatures be trusted across domains when using different security techniques to sign them? Can trust be exchanged such that only a single sign-in or authentication process is required for the user? What limitations arise as a result?

D. Secure Data Storage and Transmission

HIPAA requires that all "open systems" (e.g., those with Internet connections) protect their data with encryption technology. While encrypting one digitized x-ray before storage and decrypting it before display is unlikely to disrupt the hospital's workflow, this requirement has the potential for serious unintended consequences. CT and MR examinations typically consist of hundreds of images ("slices"), each of which is identified, stored, and retrieved individually. If a radiologist has to decrypt 500 images to see one MR of a patient's knee, what will that do to her workflow?

Our university's radiology department conducts some 380,000 examinations and produces 9 TB of digital data annually. Images are initially collected and stored in the industry standard DICOM (Digital Imaging and Communications in Medicine) format. HIPAA will require that these images be stored and communicated in encrypted form. Which encryption method is best suited to HIPAA’s requirements? How should large (encrypted) images be transmitted over the Internet? What is the impact on the radiologist's workflow if every image has to be encrypted before storage and decrypted before viewing?

VII. PROJECT STATUS

The design and implementation of the major portions of the prototype has begun. Representative "access groups" include doctors, medical staff such as technologists, patients, research groups, administrators, medical records authorities and external entities. Each group has its own access privileges and rules as defined by the authorization rule engine (e.g., a doctor can see all aspects of the electronic patient record, a patient can see all of his electronic patient record except psychological evaluations, radiology technologists can work with radiology images but not with cardiology images).

The electronic patient record has six major sections: patient identification and demographic information, medical history, physician notes, lab results, prescriptions, and medical images. Administrators can grant access privileges to any group or subgroup using a recursive definition language, and access can be granted or denied to types of data (e.g., physician's notes, images) and to individual patient data (Mr. Jones' electronic patient record). Using this scheme the hospital can quickly and easily assign access privileges to all physicians, or to subgroups (radiologists, surgeons), or to individuals (Dr. Smith), or to temporary groups (Dr. Smith and all physicians who are on call while Dr. Smith is on vacation).

The authentication rule engine enforces the strength of the identification technology needed to access data. At the moment four techniques are supported (username/password, fingerprints at desktop/laptop machines, fingerprints from PDAs, and iris scans at fixed locations). Hospital administrators can create rules that define the relative strength of the various identification technologies (e.g., iris scan is more reliable than fingerprint, fingerprint is more reliable than username/password), as well as the number and strength of authentication methods needed to access specific data or resources. One could, for example, implement a rule that requires stronger authentication from a mobile device than from a fixed location device within the hospital. This process is formalized and standardized using our concept of authentication trust levels; all data in our system specifies a trust level that must be attained to access that data.

After identity has been verified by the authentication service and an authentication ticket has been provided by the client, the authorization rule engine determines whether access to a particular type of data is permitted to a person whose identity has been established. The authorization rule engine implements those policies defined by hospital administration (e.g., a patient can see all of his own medical record except for physician notes; only a certified member of the medical records group can change a medical entry).

Encryption is a crucial component of the HIPAA security scheme. The first master's thesis to emerge from our project [25] explains the background and requirements of HIPAA, discusses the operation of four encryption algorithms, conducts performance measurements of software encryption in a .NET environment, and constructs a workflow model for our university's radiology department that predicts how the added work of encrypting and decrypting all medical images will affect patient throughput. All data and images will be stored and transmitted using the Advanced Encryption Standard [26] with 256-bit keys.

Trust sharing among ancillary healthcare services is an open research question. Banks have been able to create trust-sharing that interoperates among different ATMs, different banks, and different credit cards. Similar trust-sharing capabilities are required among hospitals, pharmacies, and insurance companies.

VIII. CONCLUSIONS

HIPAA's security provisions will require radical changes in the way healthcare data is protected and administered; our web services approach is one way to achieve that goal. One key concept is the use of trust levels to define how stringently an individual must be identified, and only if the required level of trust is established does the system even proceed to the second question of whether access to data or other resources is authorized. Trust may be defined across a spectrum from none (public access) to highly trusted (multiple biometric devices). Our project will define those trust levels and provide a language for describing them, and then will advance to the more complicated problem of how to exchange trust levels across disparate networks.

Once trust is established to the level required, an authorization rule engine is used to control access to resources. The authorization rules are dynamic, and thus there must be a simple way to describe and enforce the rules of access. All patient data and medical images are stored in encrypted form using 256-bit key AES encryption, and all network transmissions are likewise secured.

ACKNOWLEDGEMENTS

Our Internet Commerce Group is grateful to Microsoft Corporation for funding this research effort, and to Mr. David Ladd of Microsoft's University Research Program who is our liaison and project sponsor. InterCom also thanks the U.Va. Department of Radiology for its many years of continuous interaction and support.

REFERENCES

[1] Health Care Portability and Accountability Act, Public Law 104-191, http://aspe.hhs.gov/admnsimp/pl104191.htm
[2] World Wide Web Consortium, http://www.w3.org/
[3] eXtensible Markup Language, http:// www.w3.org/XML/
[4] Simple Object Access Protocol, http://www.w3.org/TR/SOAP/
[5] Java Message Service Application Programming Interface, http://java.sun.com/products/jms/
[6] Web Service Definition Language, http://www.w3.org/TR/wsdl
[7] Universal Description, Discovery, and Integration of Web Services, http://www.uddi.org/ specification.html
[8] Microsoft Passport, http://www.microsoft.com/netservices/passport/overview.asp
[9] XML Digital Signature Working Group, http://www.w3.org/Signature/
[10] XML Encryption Working Group, http://www.w3.org/Encryption/2001/
[11] XML Key Management Specifications, http://www.w3.org/TR/xkms/

[12] Organization for the Advancement of Structured Information Systems, http://www.oasis-open.org.
[13] Security Assertion Markup Languages, http:// xml.coverpages.org/aml.html
[14] The Liberty Alliance, http://www.projectliberty.org
[15] "Web Service Security," Microsoft, IBM, and Verisign joint specification, http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnglobspec/html/ws-security.asp
[16] Federal Public Key Infrastructure, http://csrc.nist.gov/pki/
[17] Kerberos, the Network Authentication Protocol, http://web.mit.edu/kerberos/www/
[18] eXtensible Rights Markup Language, http://www.xrml.org/
[19] Secure Sockets Layer, http://wp.Netscape.com/eng/ssl3/charter.html
[20] Directory of many X.509 components, http://www.ietf.org/html.charters/pkix-
[21] "Security in a Web Services World: An Architecture and Roadmap," Microsoft and IBM joint white paper, see http://msdn.microsoft.com/library/default.asp?url=/library/enus/dnwssecur/html/securitywhitepaper.asp
[22] Web Services Policy Framework, http://msdn.microsoft.com/webservices/default.aspx?pull=/library/en-us/dnglobspec/html/ws-policy.asp
[23] Web Services Trust Language, http:// msdn.microsoft.com/library/default.asp?url=/library/en-us/dnglobspec/html/ws-trust.asp
[24]Web Services Secure Conversation Language, http://msdn.microsoft.com/webservices/default.aspx?pull=/library/en-us/dnglobspec/html/ws-secure conversation.asp
[25] Andrew M. Snyder, "Performance Measurement and Workflow Impact of Securing Medical Data using HIPAA-Compliant Encryption in a .NET Environment," master's thesis, Department of Computer Science, University of Virginia, August 2003.
[26] "Advanced Encryption Standard," FIPS Publication 197, November 26, 2001, http://csrc.nist.gov/publications/ fips/fips197/fips-197.pdf