Middleware for Scalable Real-time Multimedia Cyberinfrastructure

Jill Gemmill, Aditya Srinivasan, Jason Lynn, Samir Chatterjee,*
Bengisu Tulu, Tarun Abhichandani**
Departments of Computer and Information Sciences and IT Academic Computing
University of Alabama at Birmingham
**Network Convergence Laboratory*
Claremont Graduate University
USA
jgemmill@uab.edu, Samir.chatterjee@cgu.edu

Abstract

Middleware is a layer of software that is used by applications to locate people and resources and to provide security. Middleware in the form of authentication, authorization, and directory services is especially of interest for improving the scalability and security of managed multimedia applications. This paper describes the problems, challenges and solutions to creating middleware for real-time voice/videoconferencing applications. As part of the Internet2 Middleware and National Science Foundation Middleware Initiatives, we have designed, built and deployed middleware for scalable videoconferencing and other Internet multimedia applications. Two standards are currently used for signaling in such applications: H.323, the ITU-T signaling standard used in most of today's commercial video-conferencing systems; and Session Initiation Protocol (SIP), an IETF approved standard for voice and video communications. Specific outcomes of our research include the creation of the International ITU-T standard called H.350 Directory Services Architecture for Multimedia Conferencing; an advanced SIP-based videoconferencing client that is secure and H.350 enabled; a prototype H.323 endpoint that is H.350 enabled; a Video Middleware 'Cookbook' that fully documents the H.350 architecture and provides detailed examples for software developers as well as configuration and deployment instructions; a large-scale test bed; example local and global directory services; and an architecture supporting single-sign-on authentication that is consistent with the federated administration model.

Keywords: Multimedia, Videoconferencing, Middleware, Directory Services, VoIP.

1 Introduction

Research collaborations today are formed dynamically and cross many administrative and institutional boundaries. Such collaborations constitute Virtual Organizations (VO's) [1], real people doing real work, but who, as a group, happen to have affiliations that cross a number of different institutions. Internet video and voice applications have become essential multimedia tools supporting VO communications. Middleware, as described on the web site of the National Science Foundation Middleware Initiative (NMI) [2], "refers to an evolving layer of services that resides between the network and more traditional applications for managing security, access and information exchange". Middleware such as directory services can be used by applications to locate people and resources; middleware can interface an application with external, well-developed services such as authentication and authorization. Without middleware, application developers tend to create their own lists of users, passwords, configuration parameters, and so on, with results that are often haphazard because the application developer is expert in the core application function but inexpert in other areas. Thus, a VO seeking to use internet conferencing applications can expect to experience non-interoperability across implementations and poor or non-existent security. As video and voice-over-IP (VoIP) providers add users and provide services that cross into domains administered by other service providers the absence of standardized mechanisms for handling lists of users, their contact information, their configuration parameters and their passwords quickly becomes an obstacle to building large scale, vendor independent services.

Alan Blatecky suggests that a definition of middleware should begin with a look at cyberinfrastructure which he defines as "infrastructure for the electronic world" [3]. In that context, video middleware is software 'glue' that binds the videoconferencing technology to institutional identity management services, authentication and authorization infrastructures, and standardized information stores. Using the real-world experience of building ViDeNet [4], the global, managed videoconferencing/VoIP service, we examined how the Internet Engineering Task Force (IETF) [5] Session Initiation Protocol (SIP) [6] and the International Telecommunication Union Telecommunication

Standardization Sector (ITU-T) [7] Recommendation H.323 [8] protocols could utilize middleware for improved scalability, functionality, and security.

1.1 Background and Prior Work

The first videoconferencing standard was the ITU-T H.320 [9], for use in a circuit-switched, low bandwidth environment. The high cost of leasing dedicated Integrated Services Digital Network (ISDN) and T1 telephone lines was a natural limiting factor in building circuit-based videoconferencing services; participants built one special room per institution, if they could afford to. Packet based solutions for videoconferencing services began to develop as Internet bandwidth increased and researchers began to consider what new architectures services that could be offered with. The IBM Research Division's multimedia, multiparty teleconference (MMT) [10] system authors noted that whether the conferencing service was centralized or de-centralized, directory services were important because (a) the end-station is no longer a fixed telephone number but is instead an IP address that might have been assigned dynamically and (b) it would be convenient to translate a name to the correct type of dialing address. However, a workable directory service for videoconferencing was left as an open problem.

Integration of a directory service with a videoconferencing system was a component of the AT&T Research Lab's TOPS architecture [11]. The authors proposed use of an X.500 [12] directory service to map user names to terminals, for authentication of users and access by external servers to network services, and a query-handling profile to determine how to respond to incoming calls. Their own implementation of the TOPS architecture, however, did not include these described directory service components.

The Access Grid (AG) [13] was designed for group-to-group communication over the Web, and consists of multimedia display, presentation and interactions environments, interfaces to grid middleware, and interfaces to visualization environments. AG can support large-scale distributed meetings, collaborative work sessions, seminars, lectures, tutorials and training. A number of AG nodes have been deployed and are scheduled and used for various group collaborations. While certain of the AG's advantages are unique, there are some notable disadvantages including: high cost per node (around $60,000 US), difficulty in setting up and operating the node, difficulty in receiving IP Multicast at the AG location, and difficulty in obtaining university space to house the equipment. As we began our project the Access Grid used a machine-local login procedure that was not inter-domain aware and establishing a non-public meeting required that a special access control list must be built by hand. At time of this publication, the AG had progressed from machine-local login to a special AG login and password.

1.2 Current Standards-Based Multimedia Protocols

In the mid-to-late 1990's standardized Internet conferencing protocols began to appear, in particular H.323 and SIP, which are still relatively young protocols. Product vendors tended to emphasize interoperability over other features in early products and initially appeared oblivious to scalability and security. Even though standards for security in H.323 have existed since 1998 [14] no vendor that we have found has implemented them. Early versions of the H.323 protocol required manual configuration of each server in order to create a single namespace (dialing scheme) among a set of servers. Manually configuring each server and its prefix turned out to be a tiresome, error-prone process and once deployments grew beyond a very small number of servers, the scalability issues became obvious. This problem could have been solved using the Domain Name System (DNS) [15] SRV-Records but for a number of reasons, including the need to integrate with legacy H.320 equipment (telephone circuit based), the idea of a gatekeeper hierarchy was developed. An H.323 address space can be built via a hierarchy of gatekeepers, again requiring some manual configuration and limiting scalability.

Unlike H.323, SIP was designed for audio/video communication over Internet Protocol (IP), hence, (1) the addressing scheme was compatible with other IP standards and (2) resolving local as well as remote administrative domains utilized DNS-SRV records in native SIP protocol. Even without the need for name resolution, a managed SIP service requires use of a SIP Proxy server providing registration, configuration, and signaling that is very similar to the other gatekeeper functions.

1.3 Obstacles to Scalability

ViDeNet's early experience in operating a large, managed video/VoIP service identified four significant impediments to wider deployment of video/VoIP: (a) need for reliable authentication, (b) need for inter-institutional awareness, (c) need for directory services, and (d) improved manageability through integration with enterprise identity management.[16] These conclusions led to formation of the Internet2 [17] Video Middleware Working Group [18] and participation in the National Science Foundation Middleware Initiative (NMI) where our team has led the effort to design, build and deploy scalable middleware for videoconferencing and other multimedia applications. In August 2003 our activities resulted in the international standard ITU-T Recommendation H.350 "Directory Services Architecture for Multimedia Conferencing" [19].

1.4 Organization of this Paper

The remainder of the paper is organized in the following manner. In Section 2, we clearly explain the problems and challenges facing real-time voice/video middleware implementers. In Section 3, we summarize the H.350 directory services standard and how it can benefit both H.323 and SIP-based videoconferencing system. Security issues are discussed in Section 4. Section 5 describes the design and architecture of a SIP-based videoconferencing client that we have developed. This client was downloaded by over 200 users and is now being used on Internet2. We present call flows showing how to achieve user authentication using institutional directories and briefly discuss the open problem of authorization in a federated administrative domain. Section 6 describes the design and development of our videoconferencing meta-directory (global directory service or directory of directories). Finally, in Section 7 we summarize our results and briefly discuss our future work where we are exploring federated authentication technologies to address cross-domain authorization.

2 Problems and Challenges

As we started building ViDeNet, several interesting research challenges emerged. These challenges were centered on the need for standardized directory and security solutions for multimedia applications, interoperability of these solutions, and the need for clients and servers supporting them. These problems are summarized in this section.

2.1 Need for Standardized Directories

Large-scale, managed videoconferencing systems must be scalable. Adding and removing customers, upgrading equipment and migrating between manufacturers must be possible inside the system while it is operational. This type of scalability requires a uniform way to store and locate information about customers, addresses, and configurations. Collaborators in VO's should not need to purchase systems from the same vendor in order to communicate. Using a standardized representation of the underlying data, different multimedia brands and systems could be deployed together to create an overall application environment that avoids vendor lock. A white pages search engine developed by one vendor, for example, could serve directory information to IP telephones supplied by a second vendor, with signaling managed by a call server provided by yet a third vendor.

The big advantage of a standardized representation for customers is that they can easily publish their multimedia address so that anyone in the world can find them and place a call. What if you have some information about a person but don't know their institution or the address of the institution's directory service? What if you are looking for any contact who is a member of a particular organization? A meta-directory tool is useful for these reasons. Use of a meta-directory also builds scalability since each institution can manage and update information about its own community of users and the multimedia conferencing addresses under its administrative scope. Using a standardized representation, distributed directory information can be made to appear as a single directory service that would be too large and complex for any one organization to build and maintain on its own. We found that the meta-directory service we built also served as a convenient tool for checking interoperability of the various H.350 implementations in our test bed. In Section 3 we describe the standardized representation we developed (H.350) and our experience using it to implement institutional directory services for environments with varying local conditions. In Section 6 we describe a global white pages project, in which a meta-directory is built automatically from any number of local directories.

2.2 Multimedia Security Issues and Interoperability

Two important security issues for multimedia applications are authentication and confidentiality. ITU-T security standard H.235 describes both password and certificate-based methods for authentication and encryption, while the SIP protocol describes authentication using a digest hash mechanism.

Throughout this paper we will use the term 'endpoint' to refer to the SIP User Agent (UA) or H.323 Terminal, and the term 'call server' will be used to refer to an H.323 gatekeeper or SIP proxy/registrar. In H.323, communication from endpoint to call server is exclusively over User Datagram Protocol (UDP) [20], while call server to call server communication is over Transmission Control Protocol (TCP) [21]. SIP proposes the use of Transport Layer Security Protocol (TLS) [22] to secure hop-by-hop transmission and Secure/Multipurpose Internet Mail Extensions (S/MIME) [23] for achieving end-to-end confidentiality. Both SIP and H.323 use the same protocol for end-to-end media over UDP, and both SIP and H.323 optionally use signaling through any number of intermediary servers. Those intermediaries need to see certain information in the clear which prohibits the entire SIP or H.323 signaling message from being encrypted. We found it quite interesting that we could not find any commercial products implementing any of these security standards; application layer authentication and media encryption were not available.

We decided that our project should seek ways to implement the existing security standards rather than

inventing new methods, therefore one of our design goals was that our architecture would not require any changes or extensions to the existing H.323 or SIP protocols. Another design goal was that emerging middleware infrastructure such as authoritative enterprise identity management systems should be used for authentication and as the source for information about that user rather than requiring replication of those services inside the application. Our work in securing multimedia for this project focused on integrating the application with single sign-on infrastructures.

Interoperability issues arise when different users from different VO's authenticate differently and communicate through integrated services. Interoperability between H.323 and SIP has been established for the basic signaling and media transfers through the use of gateways. However, integration of authentication and authorization mechanisms used in disparate systems is still being researched. The challenge is to make communication possible between users of different protocols while utilizing the existing native security mechanisms and exchanging credentials of different types belonging to the respective protocol domain.

2.3 Need for advanced SIP-based Videoconferencing Clients

H.323 is the dominant technology for managed and multi-way videoconferencing. Conceptually, H.323 endpoints are devices (like telephones), not people, and some other aspects of H.323's derivation from H.320 leased-telephone line transport remain. SIP was designed by the IETF for multimedia communications over IP networks from its very beginning, and has the advantage that it is based on text messages very much like HTTP and email. Today, SIP is gaining momentum for voice over IP applications and has been recently approved as the signaling standard for 3G wireless networks, but SIP-based videoconferencing applications are rather sparse. Microsoft announced their SIP support in XP Messenger and is certain to be a market driver. This situation led us to design and develop a directory enabled SIP-based videoconferencing application, described in section 5.

3 H.350 Directory Services Architecture for Multimedia Conferencing

We designed an object class for uniformly storing and locating information related to any of the video and VoIP protocols, both standard and non-standard. Our extensive practical experience managing the ViDeNet network and also an interest in making use of the growing number of university person directories implemented using the Lightweight Directory Access Protocol (LDAP) [24] led us to develop the following design criteria for the directory object class:

(1) Store all endpoint information in a canonical data source (the Directory) rather than local to the call server. Manage all real-time multimedia devices from this single store, rather than using multiple protocol-specific or vendor-specific data stores.
(2) Enable endpoint information to be associated with people, or alternately, with resources such as conference rooms or classrooms.
(3) Support multiple instances of endpoints per user or per resource.
(4) Use authoritative sources for user identity directly and avoid unnecessary replication.
(5) Enable online searchable "white pages" where dialing information can be found along with other traditional directory information such as name, address, telephone, email, etc.
(6) Support the creation of very large-scale distributed directories. These include white pages "portals" that allow searching for users across multiple institutional directories.
(7) Represent endpoints that support more than one protocol, for example endpoints that are both H.320 and H.323 or endpoints that are both H.323 and SIP.
(8) Store enough information about endpoint configuration so that correct configuration settings can be documented to end users on a per-endpoint basis as a support tool, or loaded automatically into the endpoint.
(9) Be extendable to allow inclusion of implementation specific attributes.
(10) Be non-invasive to the enterprise person directory so that support for multimedia conferencing can be added in a modular fashion without requiring significant changes to the design of the enterprise directory.

Given these criteria, we chose LDAP as a suitable directory service. Another factor influencing our choice of LDAP was the NMI program and related Internet2 activities that were developing and popularizing standardized LDAP naming and organization for university identity management. LDAP also has the advantage of being a well-established directory services protocol and robust LDAP server software is readily available. Finally, some enterprises use LDAP authentication for single sign-on login service, and we planned to leverage the endpoints' LDAP capability to implement each protocol's defined authentication.

3.1 Overview of H.350

An overall view of the H350 architecture is shown in Figure 1. Note that the multimedia directory service (inside

the large oval area) involves two functionally different directories: the enterprise person directory and the application-specific H.350 directory. Through discussions with expert Internet2 directory architects we learned that directory managers were not favorable to including new, large object classes in their enterprise directories, especially classes that might require frequent writes and updates. In addition, conferencing management applications require privileged access to the directory in order to retrieve masked attribute values, but privileged access to an enterprise directory might be difficult to arrange. The directory experts preferred the approach of modifying a person entry by adding a simple Uniform Resource Identifier (URI) [25] pointing to an application specific directory entry that could be administered elsewhere.

As a result, the H.350 architecture consists of two core object classes: commURIObject for use in the person directory, and commObject for use in the multimedia directory. The relationship of these two classes is shown in Figure 2. The H.350 architecture is quite flexible; the person and H.350 directories can be two separate directories, or two branches of a single directory tree may be used instead. Each H.350 commURI has an associated user-friendly label such as "Ann's Office" and this label is used to associate a person with their communication attributes by pointing to that user's endpoint specific conferencing object in the H.350 directory.

Use of the standardized inetOrgperson object class [26] is recommended for the person directory; this class contains name, address, and other methods used to contact that person such as telephone number and email address. The multimedia directory contains one commObject entry for each instance of a multimedia communications endpoint, and each commObject contains at least one commOwner.

The commObject is an LDAP structural class, with attribute commOwner, a URI pointing to the directory entry of the person who owns this commObject. The commObject has a number of auxiliary object classes that are protocol-specific. Auxiliary classes have been defined for the H.323 (h323Identity [27]), SIP (sipIdentity [28]), H.320 (h320Identity [29]), and H.235 Security (h235Identity [30]) protocols. Non-standard conferencing protocols, for example Access Grid, can be represented using a generic auxiliary class (genericIdentity [31]). Each auxiliary class contains attributes that are specific to that protocol. The use of these auxiliary classes within a commObject is illustrated in Figure 3.

Each user in the enterprise directory may have multiple commURIs representing their use of multiple devices for voice and video communication. No matter what protocols different people may be using at a single institution, white pages lookups, based on the H.350 architecture, will provide all ways that each person can be contacted using their preferred multimedia endpoints.

Figure 3 also shows the location of each object's security credentials. For example, the H.350 endpoint object on the left contains the auxiliary class h235Identity. The H.235 protocol, part of the H.323 suite of protocols, is a security protocol with a profile defining a videoconferencing login and password to use when registering with the gatekeeper. These credentials are stored in the attributes h235IdentityEndpointID and h235IdentityPassword. On the right, the Person object class contains a password that could be the enterprise single-sign on password.

3.2 Endpoint Registration Example

One of the main benefits provided by H.350 to service providers is enterprise storage, management, and protection of a master data store of endpoint information. Prior to the development of H.350, the service administrator needed to load the registration information directly into the call server, usually in some proprietary format. If a single service provider happened to support multiple protocols they would end up with configuration information stored in many

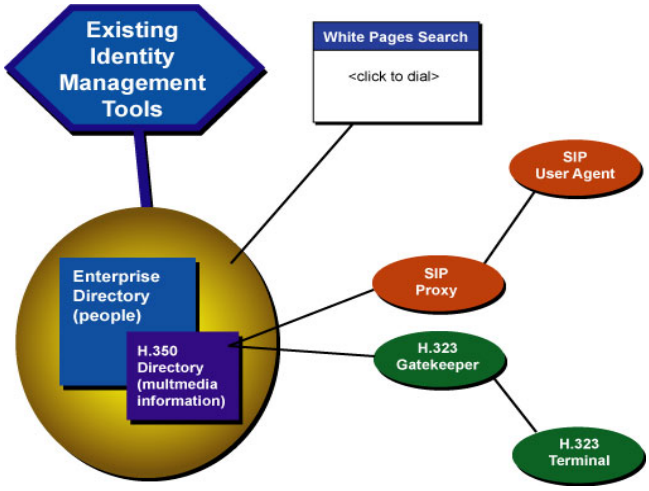


Figure 1 H.350 Architecture

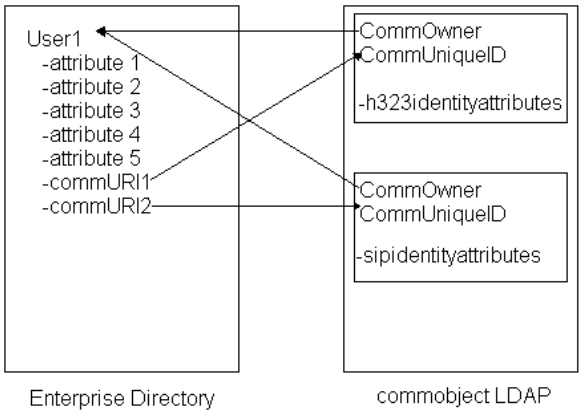


Figure 2 H.350 Object Classes relating person entries to communication entities

formats in multiple locations. This situation is an obvious barrier to scalability due to the amount of manual work needed to enter and track this information. H.350 can be implemented in endpoints as well as servers, and doing so makes it possible to automatically configure the endpoint properly at endpoint registration. This can save quite a bit of desktop support in the form of manual endpoint configuration. Even if the endpoint is not aware of H.350, there are management benefits.

As an example of how to use the H.350 architecture, Figure 4 provides a detailed call flow showing endpoint registration with the call server. To simplify the diagram, no error conditions or other failures are considered and User A and User B happen to be listed in the same Enterprise Directory and are registered with the same call server – this condition is not required.

Prior to placing calls through the managed multimedia infrastructure, the endpoint must register with the call server

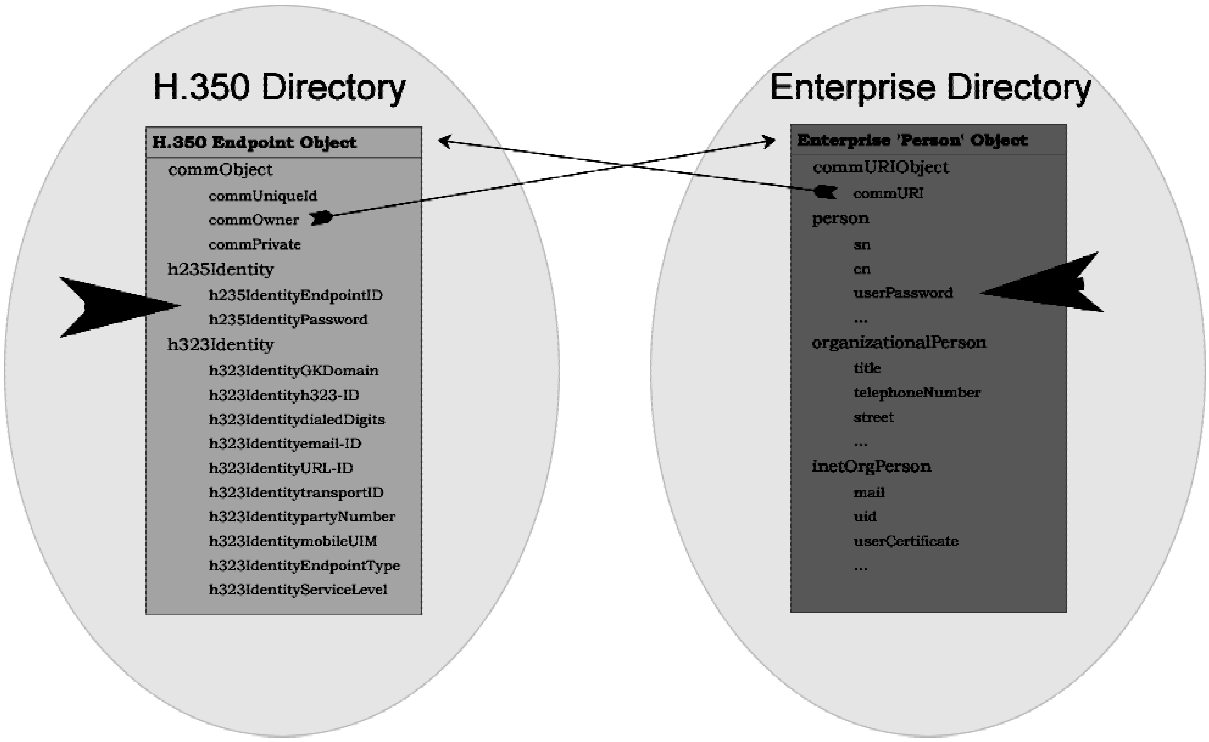


Figure 3 Auxiliary classes within a commObject. Thin arrows show linkage of commOwner to a person object, and commURI to an endpoint object. Large dark arrows indicate location of security credentials in each object class. h235Identity and h323Identity are auxiliary object classes

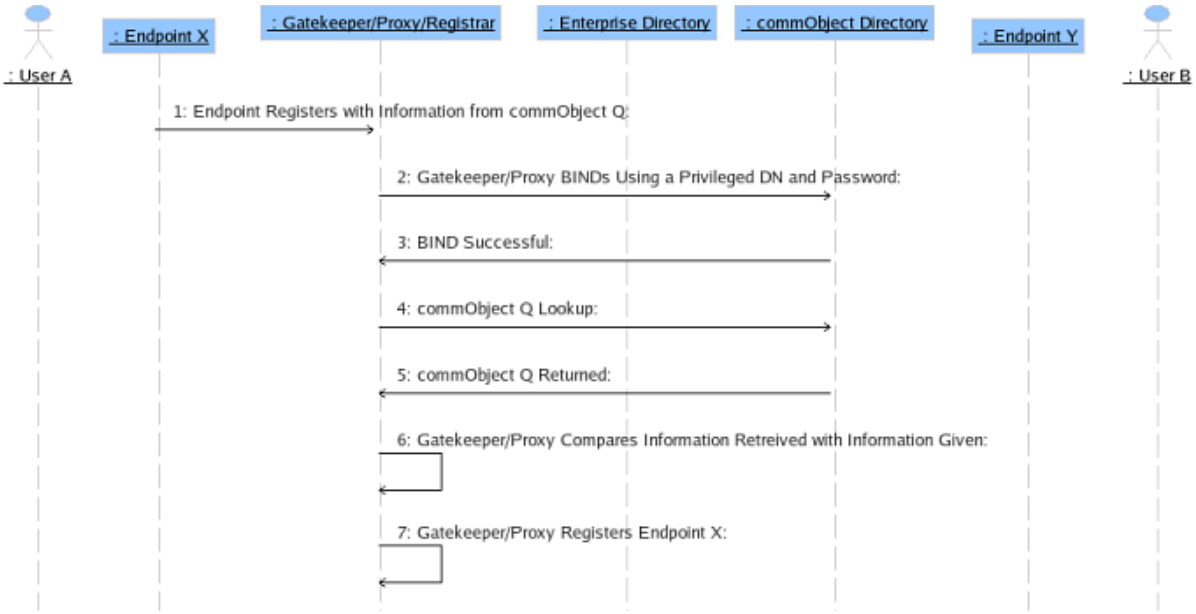


Figure 4 Call Flow for Endpoint Registration using H.350

(Step 1); this initial step is required in both H.323 and SIP. To validate the information presented by the endpoint during registration, the call server BINDs to the commObject directory with a privileged distinguished name and password permitting it to read any commObject in the directory (Steps 2 and 3). The call server can access the user's commObject (Steps 4 and 5) in order to compare the authoritative information to what has been presented (Step 6). If the information matches, the configuration is valid and the gatekeeper or registrar permits the endpoint to register (Step 7).

3.3 H.350 Enabled Applications and Interoperability Test Bed

Several variations of H.350 directory services have been implemented in our test bed, illustrated in Figure 5. The initial implementation of H.350 services used a single directory for both people and endpoints and was developed in PHP: Hypertext Preprocessor [32] as a stand-alone service for ViDeNet users. Deployment of H.350 directories was then extended to a small number of locations where there were separate enterprise and H.350 directories, or where integration with an actual enterprise directory could be achieved, using different combinations and versions of OpenLDAP and SunOne (iPlanet) directory servers. These efforts created a large interoperable, managed multi-media services test bed utilized by multiple institutions in Internet2.

We have deployed the following H.350-capable components in our test bed. Our commercial partner RADVISION [33] developed a proof-of-concept H.323 software endpoint that is H.350-capable and is available to ViDeNet test bed participants. We also conducted beta testing with the RADVISION H.350-enabled Gatekeeper. An H.350-enabled SIP Proxy server was developed by HCL Technologies [34] and installed in our test bed. The H.350-enabled HCL proxy is capable of managing the endpoints' address and authentication information "behind the scenes" in accordance with enterprise policy, without requiring any modification to the endpoint. These components, in addition to the previously mentioned CGUsipClient, were demonstrated at the Fall 2003 Internet2 meeting. Figure 5 shows the H.350 implementation domains and specialized application areas of each deployment. The test bed also includes certain entities accessed for testing purposes including a Simple Traversal Over UDP Network (STUN) server [35] and a SIP/PSTN gateway.

3.4 Successful Standardization

The commObject architecture was presented to ITU-T Study Group 16 in the Fall of 2002 and was ratified as an

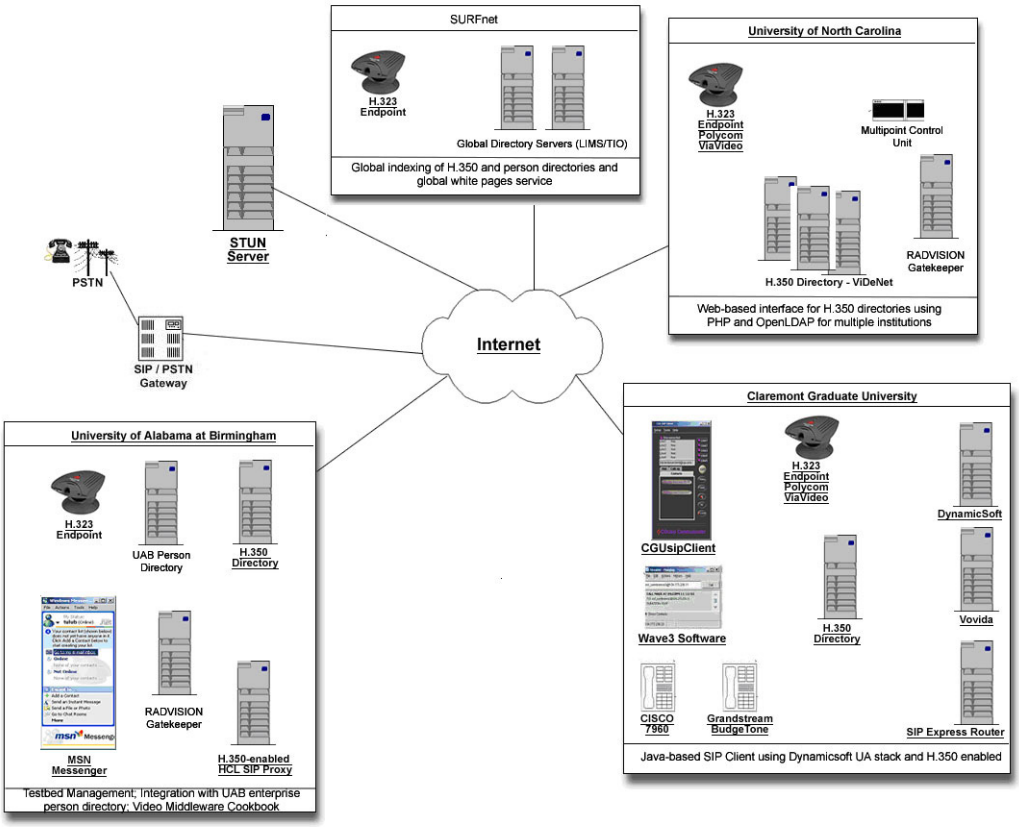


Figure 5 The ViDeNet Test Bed

international standard in September 2003. The H.350 standard has also been reported by the ITU-T to the IETF and its SIPPING group via established ITU-IETF communication channels.

4 Identity Management and Security for Multimedia Applications

SIP and H.323 are not easy protocols to secure. Their use of many trust boundaries, modes of operation, intermediaries, expected usage between elements with no trust at all, and user-to-user operation together makes security mechanisms very challenging. It should be noted from the outset that there is no such thing as "absolute" security. There are different levels of risk that various deployments are willing to tolerate given the spectrum of risk versus operational and ease of deployment. Adding to the confusion, there are many existing security mechanisms inside and outside the application, with overlapping functionality.

4.1 Security in SIP

SIP messages face all the typical security threats that are faced by other IP applications. These include registration hijacking, tampering with message bodies, tearing down a call, impersonating a server and distributed denial of service attacks. Note that the security of SIP signaling itself has no bearing on the security of protocols used in concert with SIP such as Real Time Protocol (RTP) [36]. Any media associated with a session can be encrypted end-to-end regardless of any associated SIP signaling. For media encryption, secure distribution of the shared encryption keys can be difficult; either out-of-band delivery must be arranged in advance, or a secure channel must be established on request.

A Digest Authentication scheme [37], based on a cryptographic hash function is one of the recommended authentication mechanisms for SIP and it is illustrated in Figure 6. Once a SIP User Agent initiates a REGISTER request to a proxy, it receives back an authentication required response with challenge parameters like realm, nonce value, etc. The SIP UA is expected to respond back with a new REGISTER message including a hashed value of the required parameters such as user name, password, nonce value, and realm in addition to the information provided in the first REGISTER message. Hence, only a hash value is exchanged between UA and servers instead of the SIP password itself. The Proxy/Registrar then generates the hash using the user information it retrieves from the storage and compares these two values. Registration happens once this comparison is completed successfully.

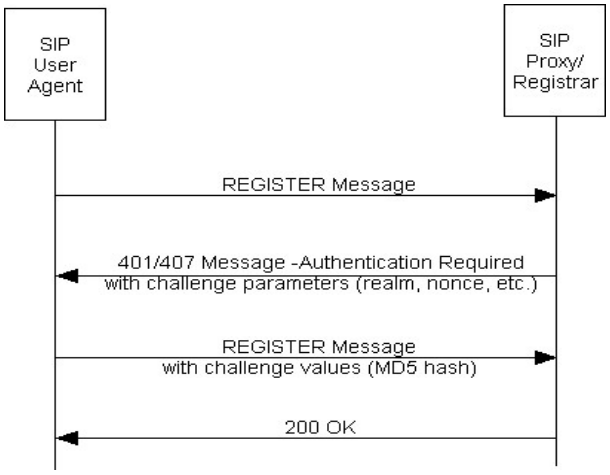


Figure 6 SIP Authentication Mechanism

4.2 Security in H.323

To solve authentication issues in the H.323 standard, ITU-T H.235 Annex D uses a password shared by the endpoint and gatekeeper to encrypt and decrypt messages between components in a manner similar to the SIP approach. As shown in Figure 7, the sending endpoint used the shared secret to encrypt the address/ID of the recipient, a timestamp, and optionally its own address and a random number. The Registration Request signal carries a Cryptotoken. (A Cryptotoken is the name of a message field used in the Registration, Admission, and Status signaling protocol.) The endpoint nests the encrypted token (indicated by CryptoPwdEncr) inside the Registration Request's Cryptotoken and sends the request to the Gatekeeper. signaled, including the Upon receiving the CryptoToken, the Gatekeeper uses its copy of the shared secret to encrypt its own address and the received timestamp (which is sent in the clear as well as in the encrypted data). The computed value is compared to encrypted value, and if they match the endpoint receives a Registration Confirmation message. The H.235 standard specifies additional approaches for authentication, included shared-secret based hash and use of

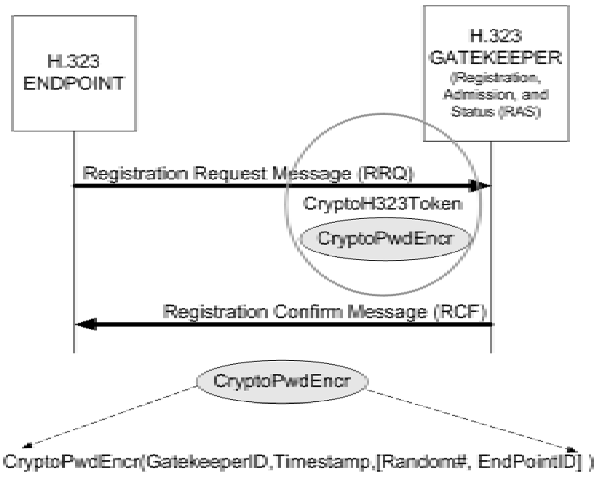


Figure 7 H.235 Registration with a Gatekeeper Using Authentication

digital signatures.

As mentioned previously, in spite of these protocol-specific security specifications, we could find no products implementing these approaches. We believe some key reasons for this absence of basic security are (a) storage locations for security credentials vary by application or operating system platform, (b) enough variety in enterprise authentication systems exists such that application developers find it overwhelming to support well-developed interfaces to all of these authentication options, and (c) some licensing, export, and other non-technical issues associated with certificate-based solutions are prohibitive [38].

4.3 Improving Identity Management and Security Using H.350

We decided that a first and important step in managing and securing multimedia was to utilize an authoritative listing of persons associated with the institution rather than recreating that information in non-standard format for each application. Today most universities and other large organizations store person related information in a central directory, and make that single store of information available to numerous applications via LDAP. The information located in the central directory is authoritative because the data sources are the entities responsible for creating and maintaining that data, such as the Human Resources (personnel office) or university Registrar's office.

For H.323, the H.350 architecture improves security by providing standardized storage for both password and certificate based authentication credentials needed by multimedia applications. The h235Identity object class contains attributes to support H.235 Annex D password authentication, and also attribute userCertificate (an attribute of the inetOrgPerson object class) to support H.235 Annex E and F digital certificate-based authentication. Likewise, the sipIdentity object class contains attributes defined in Digest Authentication. Now multimedia applications have a standardized storage location for credentials needed when verifying user registrations. The H.350 storage locations for security credentials are shown in Figure 3.

5 SIP Client Design and Architecture

The Session Initiative Protocol (SIP) is gaining momentum for voice over IP applications and has been approved as the signaling standard for 3G wireless network, but SIP-based videoconferencing applications are rather sparse. Given our need for directory-enabled software we needed to design and develop an advanced SIP-based videoconferencing application that would demonstrate proof of concept.

5.1 Design and Development of CGUsipClientv1.x

The Network Convergence Lab (NCL) at Claremont Graduate University (CGU) developed CGUsipClientv1.1.1 using the Dynamicsoft commercial SIP stack, which provides authentication mechanisms described in the SIP standard. Voice and video capabilities were incorporated using Java Media Framework (JMF) 2.1.1 Sun libraries. The CGUsipClientv1.1.1 architecture is presented in Figure 8. As illustrated in the figure, there are two main Java packages – *cgusip.client* and *cgusip.utils* – that structure CGUsipClientv1.1. The *utils* package handles the existing instances of SIP connections and calls. The client package has three main components: *gui*, *sip*, and *media*. The *gui* package handles all aspects of the client user interaction. The *sip* package handles the necessary interaction between the CGUsipClientv1.1.1 and the Dynamicsoft SIP stack for creating and terminating sessions and initiating and receiving calls. The *media* package is in charge of making media connections using JMF and Dynamicsoft SDP libraries. Supported video formats are h.263, h.261, and jpeg; supported audio formats are G.723, GSM, DVI, and μLAW. In case of multiple calls, the video and audio of the active call is projected. For additional description of the specific components of the client refer to [39]. This version of the client is available for download [40].

In addition to providing facilities for registering to the CGU SIP proxy and registrar, CGUsipClientv1.1.1 provides a private address book and is capable of searching any H.350-compatible directory. The software also uses the H.350 directory to implement "click-to-call". By clicking on a contact name listed, the application automatically places a SIP call instead of the user having to type the desired SIP URI manually. CGUsipClientv1.2, which was successfully demonstrated at the Fall 2003 Internet2 Member Meeting [41], uses the H.350 directory to accomplish enterprise-integrated authentication.

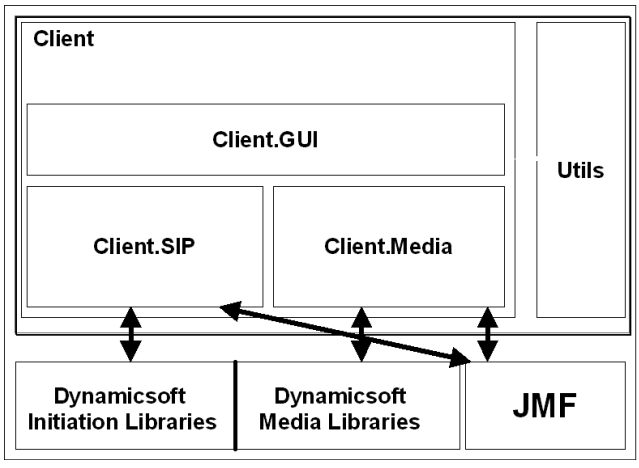


Figure 8 CGUsipClientv1.1.1 Architecture

5.2 Single Sign-On using H.350

Since H.350 application code needed to be LDAP-enabled, our implementation of enterprise-based single sign-on leveraged this opportunity and incorporated LDAP-based authentication at the enterprise directory to load and verify videoconferencing security credentials. We are not stating that LDAP authentication is the best solution; what is important here is that we demonstrated integration of single sign-on for applications that have "legacy" authentication design. In Figure 9, User A begins by entering his or her enterprise username and password (Step 1); the endpoint is pre-configured with the address of the enterprise LDAP service. Endpoint X searches for User A in the directory to find the user's distinguished name (dn); if located, Endpoint X then binds with that dn using the password supplied (Steps 2-4). If more than one commURI is associated with that user, the list of commURI's is presented by Endpoint X and the user is asked to choose which profile should be used (Steps 5-9). Endpoint X again binds with the user's distinguished name and password, but this time it binds to the commObject directory (Step 10). The commObject directory in this instance is set up to for pass-through authentication so that it "passes" the bind on to the enterprise directory. The pass-thru steps are needed in order to assure that only the user that owns a particular commObject is able to read the private commObject attribute values (Steps 10-13). Once the BIND succeeds, the endpoint can then read the commObject associated with the commURI and can configure itself with attribute values provided by the commObject from this authoritative source, including obtaining the videoconferencing application-specific authentication credentials (Steps 14-16). Using a single enterprise identity and login or user profile, a user can select the appropriate service provider for the endpoint employed (the address of the proxy server or registrar is loaded from the directory server), or the user could borrow

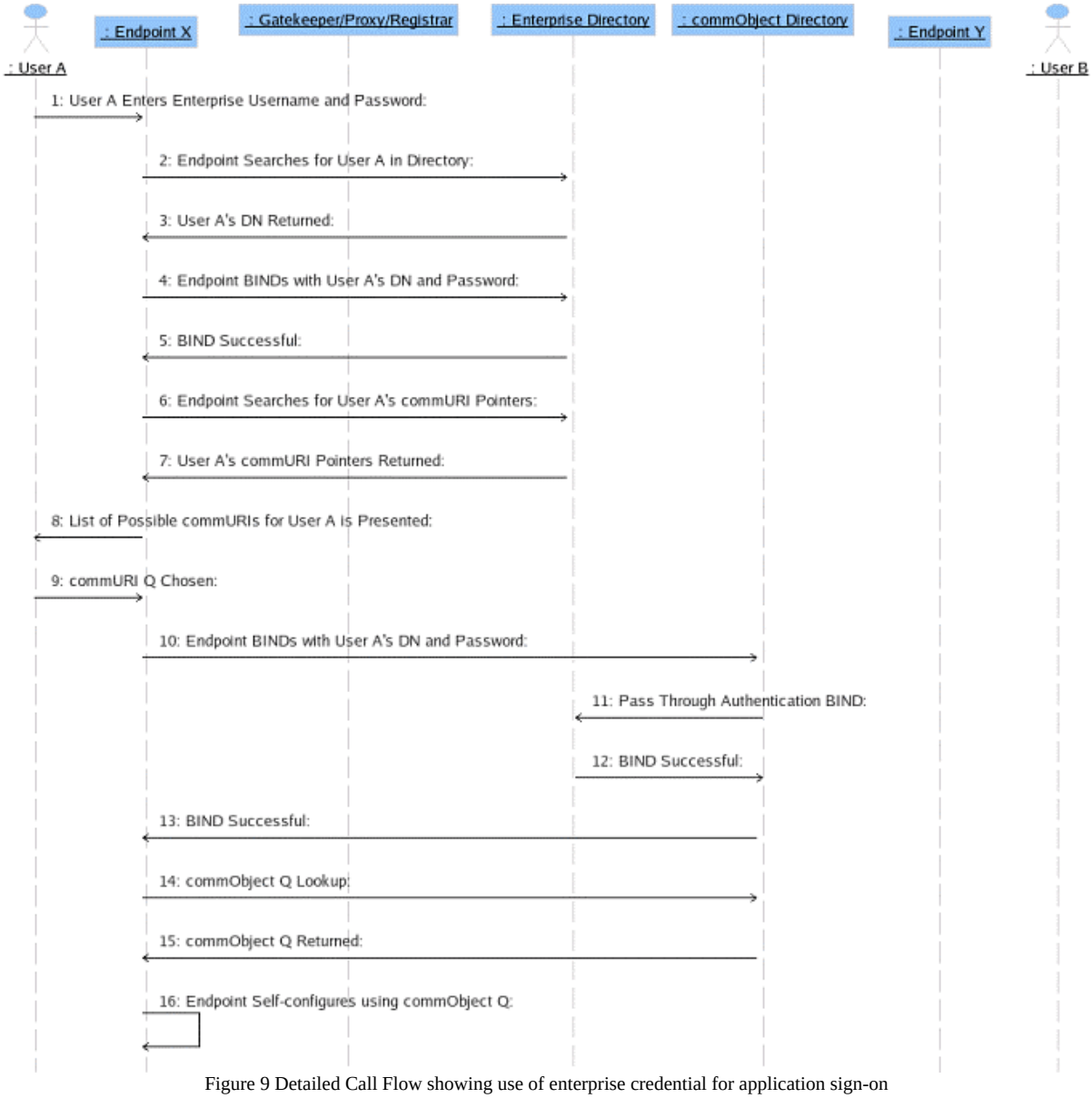


Figure 9 Detailed Call Flow showing use of enterprise credential for application sign-on

someone else's SIP phone while maintaining their own identity and service provider. Embedded systems vendors have considered approaches for getting correct address information into their endpoint. In cases like this where there is no user, an H.350 management system could do lookups on behalf of the endpoint and then transmit that information to the endpoint.

5.3 Federated Administration is of future interest

At the beginning of our H.350 design work we became quite interested in the "federated administration" [42, 43] model for authentication, especially given our motivation of supporting VO's. Federated administration is a method for one administrative domain to provide authentication and user attributes to other domains that are willing to trust that information source, and is similar to the approach taken by the Liberty Alliance [44]. The Shibboleth software [45] developed to demonstrate federated administration was particularly intriguing because any number of authentication schemes – such as UNIX or LDAP password, or Kerberos [46] – could be used locally but exchange information globally. Using federated administration, a message stating "Jane Dixie from University XYZ is calling" received anywhere outside of UAB through any service provider could be believed. Unfortunately, the early Shibboleth software developed for federated administration was written for web-based services only and lacked an API. As a result of our discussions with the Shibboleth developers, a Shibboleth API is under development.

6 Global Directory Services for Multimedia

A global directory (also called meta-directory or a directory of directories) can be built in many ways. The simplest approach that we tested and deployed first involved a set of Java based servlets written using the Java JNDI API that search across multiple directories in parallel. A configuration file is created listing the LDAP URI and base DN for each server to be searched. A query sent to the servlet spawns a thread for each directory and waits a (customizable) period of time for each thread to return results back to the controlling thread. The results include each user at any institution that matched the search criteria and their multimedia endpoints in the form of a clickable hyperlink. Clicking any link invokes another set of servlets that use the LDAP URI in the commURI to query the appropriate H.350 directory. Using Java to query hundreds of servers simultaneously is not very scalable, so we tried using the *meta* backend available with OpenLDAP that directly provides similar capability. We tested this approach and found that it worked well; our configuration is documented in the cookbook.

Eventually, when the number of servers to be searched becomes too large coupled with slow directory servers, the OpenLDAP solution does not scale and response time becomes unacceptably slow. A more elegant solution is the IETF LIMS directory server solution [47, 48], which was implemented for this project by our colleagues at SURFnet [49]. This global directory can be searched at http://directory.vide.net/. The LIMS server is a directory search engine that functions as an LDAP v3 compliant server. If the LDAP client making the request has been set to automatically chase referrals it will automatically end up at the correct directory and can then query for the commURI and use it to locate the person's videoconferencing endpoints. We found that LIMS requires each client to be fully LDAPv3 compliant, which unfortunately excluded most browsers. Other difficulties were encountered due to variations in indexing and LDAP directory tree design. These problems are the sorts of issues that must be addressed and standardized in order for middleware to become seamless and transparent.

7 ViDe H.350 Cookbook

A full discussion of the H.350 architecture, object classes, attributes, and choices when implementing the H.350 architecture are included in the ViDe H.350 Cookbook: Directory Services Middleware for Multimedia Conferencing [50]. The cookbook explains how to install and configure H.350 directory services, modify multimedia services to take advantage of this service, and integrate with enterprise directories. Source code examples in several programming languages are provided to help developers understand the interaction of multimedia components with the H.350 directory server.

8 Conclusions

Based on current work to build a middleware that can support voice and video over IP applications, it has become clear that the technologies and tools we have developed fill a critical need as larger deployments begin to happen and awareness of the need for security deepens among end users. Vendors are already implementing the H.350 directory services standard, including the RADVISION ECS Gatekeeper; Tandberg [51] TMS management software, VCON [52] MXM management system, and the HCL SIP Proxy mentioned earlier. Polycom [53] has announced future support for H.350 in its products.

Many interesting opportunities remain for enhancing the security of H.323 and SIP-based systems. The use of digital certificates is a natural next step. We are aware that

one contributing factor for the slow adoption of Public Key Infrastructure (PKI) is the cost of deploying individual user certificates that can be easily and believably validated, to users that understand proper management of their private key. Non-Certificate Authority-based approaches for security exists, such as Pretty Good Privacy (PGP), which uses the notion of self-signed certificates and our ability to trust another person's voice and image; we are presently developing PGP-style certificate based security within SIP clients.

The notion of a federated system for authentication and authorization is an open field. We are especially interested in server-level digital certificates, where the server is trusted to use non-PKI means for identifying a particular set of users and then issues a temporary X.509 [54] style certificate in a proxy fashion for the user, or S/MIME signs an assertion about the user's identity or attributes. Technologies such as Shibboleth, Security Assertion Markup Language (SAML) [55], and KX.509 [56], are all possible candidates to explore in this arena.

LDAP access control rules are a complex issue. Securing information in any directory server is a challenge since there are no standards [57] for LDAP Access Control Instructions (ACI). Lacking standards, ACI functionality varies significantly across directory servers so developing a set of mechanisms to protect data across implementations while simultaneously maintaining single sign on required careful thought. Our architecture allows non-trusted, non-privileged services to obtain protected information in a user's directory entry by leveraging the use of enterprise authentication and pass-thru authentication to obtain that information appropriately. This architecture is also useful from a more abstract perspective. Our architecture, without the videoconferencing protocol specific object classes, works for any other non-trusted application: use enterprise authentication and then leverage use of the authoritatively established user to gain his or her attributes. General purpose abstract object classes supporting this architecture could be designed to allow service specific object classes to inherit from these abstract classes.

The growing awareness of a need for security in cyberinfrastructure has actually placed many roadblocks in the path of real-time multimedia applications, specifically firewalls and network address translators (NAT). If a desktop is behind a NAT then it is using a private, non-routable IP address. While a NAT device handles the address translation at the network layer, the H.323 and SIP applications actually signal using the desktop device's IP address as the return IP address – the application is unaware of the NAT and fails because of that. Some endpoints now allow a knowledgeable user to specify a return IP number corresponding to the address appearing at the other side of the NAT; however, the user must be aware of the difference between addressable and non-addressable addresses, understand how to obtain the correct IP address to use, and know where to modify the application. Firewalls typically close most inbound ports, perhaps allowing only web traffic and email to enter unrestricted. Attempting to define firewall rules for multimedia is especially difficult since the H.323 and SIP applications typically generate dynamic ports for the media such as voice and video. If all the correct return ports are not open through the firewall the protected desktop may be able to place outgoing calls but will not be able to receive the acknowledgement back so the call signaling is broken and the call is terminated. The range of problems described in this paragraph provide example cases where an application fails because it is not aware of things occurring in the underlying infrastructure and which are not addressed by the protocols' internal rules for security.

As part of the Internet2 Middleware and National Science Foundation Middleware Initiatives we have designed, built and deployed middleware for scalable real-time multimedia deployments over the Internet. Specific outcomes of our research includes the creation of an International ITU-T standard called H.350 Directory Services Architecture for Multimedia Conferencing; an advanced SIP-based Videoconferencing Client that is secure and H.350 enabled; a multimedia meta-directory; a large-scale test bed; and an architecture supporting single-sign-on authentication that is consistent with the federated administration model. In addition, The ViDe H.350 Cookbook version 0.5 was introduced in NMI Release 4 and version 1.0 was included in NMI Release 5.

Acknowledgements

This material is based upon work supported by the National Science Foundation (NSF) under ANI-0222710 to the University of Alabama at Birmingham. We acknowledge the contribution of colleagues Tyler Miller Johnson, University of North Carolina Chapel Hill (UNC) and Editor of ITU-T Study Group 16 document H.350; Nadim El-Khoury of UNC; Egon Verharen, Director of Innovations at SURFnet; and Karen Krivaa and others of RADVISION. Additional support was provided by NSF EPSCoR EPS-0096193 and NSF Middleware Initiative ANI-0123937 via SURA-2002-103 Subcontract. H.350 was introduced in NMI Release 2 of the NSF Middleware Initiative as *commObject* and the ViDe H.350 Cookbook was introduced in NMI Release 4. Both are NMI-EDIT components, in part supported by NSF ANI-0123937. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Additional support has been provided by the University Corporation for Advanced Internet Development (Internet2). We acknowledge and thank the support of all our colleagues at Internet2 Video Middleware working group. Special thanks to Ken Klingenstein, Tom Barton, Michael Gettes, John-Paul Robinson, Bob Morgan, Scott Cantor, Sasha Ruditsky, and Art Vandenberg for their contributions to commObject.

References

[1] Foster I., Kesselman C., and Tuecke S. "The Anatomy of the Grid", *International Journal of High Performance Applications*, Vol. 15, No. 3, 2001.

[2] National Science Foundation Middleware Initiative Accessed at http://nsf-middleware.org/ Retrieved May 15, 2004.

[3] Blatecky, A., West, A., Spada, A. "Middleware: The New Frontier", *EDUCAUSE Review*, Vol. 37, No. 4, 2002 pp. 24-35.

[4] ViDeNet Access at https://videnet.unc.edu/my_ videnet.phtml Retrieved May 15, 2004.

[5] Internet Engineering Task Force Accessed at http://www.ietf.org/ Retrieved May 15, 2004.

[6] Rosenberg J., Schulzrinne H., Camarillo G., Johnston A., Peterson J., Sparks R., Handley M., Schooler E. "SIP: Session Initiation Protocol", Internet Engineering Task Force RFC 3261, 2002.

[7] International Telecommunication Union Accessed at http://www.itu.int/ Retrieved May 15, 2004.

[8] Recommendation H.323 "Packet-based Multimedia Telecommunication Systems". International Telecommunication Union, Telecommunication Standardization Sector of ITU, Geneva, Switzerland, 1998.

[9] Recommendation H.320 "Narrow-band visual telephone systems and terminal equipment", International Telecommunication Union, Telecommunication Standardization Sector of ITU, Geneva, Switzerland, July 1990.

[10] Willebeek-LeMair, Marc H; Shae, Zon-Yin. "Distributed video conferencing systems", *Computer Communications* Vol. 20, 1997, pp. 157-168.

[11] Anerousis, N; Gopalakrishnan, R.; Kalmanek, C. et al, "TOPS: An Architecture for Telephony over Packet Networks", *IEEE Journal on Selected Areas in Communications*, Vol. 17, No. 1, January 1999, pp 91-108.

[12] Recommendation X.500 "Open Systems Interconnection - The Directory: Overview of concepts, models and services" International Telecommunication Union, Telecommunication Standardization Sector of ITU, Geneva, Switzerland, (also ISO (ISO/IEC 9594)).

[13] Stevens, R., Papka, M., Disz, T. "Prototyping the Workspaces of the Future", *IEEE Internet Computing*, Vol. 7, No. 4, July/August 2003, pp. 51-58.

[14] Recommendation H.235 "Security and Encryption for H-series (H.323 and H.245-based) multimedia terminals", International Telecommunication Union, Telecommunication Standardization Sector of ITU. Geneva, Switzerland, 2000.

[15] Mockapetris, P. "DOMAIN NAMES – IMPLEMENTATION AND SPECIFICATION" Internet Engineering Task Force RFC 1035, November 1987.

[16] Gemmill, J., Chatterjee, S., Johnson, T., Verharen, E., "ViDe.Net Middleware for Scalable Video Services for Research and Higher Education", *Proceedings of the ACM Southeastern Conference, Savannah, GA.* 2003, pp. 463-468.

[17] Internet2 Accessed at http://www.internet2.edu/ Retrieved May 15, 2004.

[18] Internet2 Video Middleware Working Group Accessed at http://middleware.internet2.edu/video/ Retrieved May 15, 2004.

[19] Recommendation H.350 "Directory services architecture for multimedia conferencing,", International Telecommunication Union, Telecommunication Standardization Sector of ITU, Geneva, Switzerland, August 2003.

[20] Postel, J. "User Datagram Protocol", Internet Engineering Task Force RFC 768 August 1980.

[21] Information Sciences Institute, "DOD Standard Transmission Control Protocol", Internet Engineering Task Force RFC 761, January 1980.

[22] Dierks, T.; Allen, C. "The TLS Protocol", Internet Engineering Task Force RFC 2246, January 1999.

[23] Dusse, S.; Hoffman, P.; Ramsdell, B., et al "S/MIME Version 2 Message Specification", Internet Engineering Task Force RFC 2311, March 1998.

[24] Hodges, J., Morgan, R. "Lightweight Directory Access Protocol (v3): Technical Specification", Internet Engineering Task Force RFC 3377, September 2002.

[25] Berners-Lee, T.; Fielding, R., et al. "Uniform Resource Identifiers (URI): Generic Syntax", Internet Engineering Task Force RFC 2396, August 1998.

[26] Smith, M. "Definition of the inetOrgPerson LDAP Object Class", Internet Engineering Task Force RFC 2798. 2000.

[27] Recommendation H.350.1 "Directory services architecture for H.323", International Telecommunication Union, , Telecommunication Standardization Sector of ITU. Geneva, Switzerland, August 2003.

[28] Recommendation H.350.4, "Directory services architecture for SIP", International Telecommunication Union, , Telecommunication Standardization Sector of ITU. Geneva, Switzerland, August 2003.

[29] Recommendation H.350.3, "Directory services architecture for H.320", International Telecommunication Union, Telecommunication Standardization Sector of ITU. Geneva, Switzerland, August 2003.

[30] Recommendation H.350.2, "Directory services architecture for H.235", International Telecommunication Union, Telecommunication Standardization Sector of ITU. Geneva, Switzerland, August 2003.

[31] Recommendation H.350.5, "Directory services architecture for Non-Standard Protocols", International Telecommunication Union, Telecommunication Standardization Sector of ITU. Geneva, Switzerland, August 2003.

[32] PHP: Hypertext Preprocessor is a project of the Apache Software Foundation. Accessed at http://www.php.net/ Retrieved May 15, 2004.

[33] RADVISION Accessed at http://www1.radvision.com/corporate/radvision Retrieved May 15, 2004.

[34] HCL Technologies Accessed at http://www.hcltech.com/ Retrieved May 15, 2004.

[35] Rosenberg, J. Weinberger, C. Huitema, and R. Mahy, "STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)," Internet Engineering Task Force RFC 3489, March 2003.

[36] Schulzrinne, H., Casner, C., Frederick, R., and Jacobson, V. "RTP: A Transport Protocol for Real-Time Applications," Internet Engineering Task Force RFC 1889, January 1996.

[37] Franks, J., Hallem-Baker, P., Hostetler, J. et al, "HTTP Authentication: Basic and Digest Access Authentication" Internet Engineering Task Force RFC 2617, 1999.

[38] Gemmill, J. "Secure Videoconferencing Today", Internet2 Fall Member Meeting, Indianapolis Indiana, November 30, 2003. Accessed at http://www.internet2.edu/presentations/fall-03/20031014-Video-Gemmill.ppt and http://www.researchchannel.org/program/displayevent.asp?rid=1991 Retrieved May 15, 2004.

[39] Tulu B., Abhichandani T., Chatterjee S.,Li H. "Design and Development of a SIP-based Video Conferencing Clienr", *6th IEEE International Conference on High-Speed Networks and Multimedia Communications*, Estoril, Portugal. 2003.

[40] CGU Networked Convergence Lab and SIP software Accessed at http://ncl.cgu.edu Retrieved May 15, 2004.

[41] Chatterjee, S. "Directory Enabled SIP Video Conferencing Client", Internet2 Fall Member Meeting, Indianapolis Indiana, 2003. Accessed at http://events.internet2.edu/2003/fall-mm/sessionDetails.cfm?session=1075 Retrieved May 15, 2004.

[42] Klingenstein N., Shibboleth FAQ", 2002. Accessed at http://shibboleth.internet2.edu/shib-faq.html. Retrieved May 15, 2004.

[43] Erdos, M., Cantor, S., "Shibboleth Architecture Draft V05", 2002. Accessed at http://shibboleth.internet2.edu/docs/draft-internet2-shibboleth-arch-v05.pdf Retrieved May 15, 2004.

[44] Liberty Alliance Accessed at http://www.projectliberty.org/ Retrieved May 15, 2004.

[45] Shibboleth software available at http://shibboleth.internet2.edu/ Retrieved May 15, 2004.

[46] Kohl, J. and C. Neumann, "The Kerberos Network Authentication Service (V5)", Internet Engineering Task Force RFC 1510, September 1993.

[47] Hedberg, R., Greenblatt, B. *et al*, "A Tagged Index Object for use in the Common Indexing Protocol", Internet Engineering Task Force RFC 2654, 1999.

[48] Allen, J., Leach, P., Hedberg, R. "CIP Transport Protocols", Internet Engineering Task Force RFC 2653, 1999.

[49] SURFnet Netherlands Research Network. Accessed at http://www.surfnet.net/ Retrieved May 15, 2004.

[50] Gemmill, J., Lynn, J.L.W., editors. *ViDe H.350 Cookbook, Version 1.0* Accessed at http://lab.ac.uab.edu/vnet/ and was included as a National Science Foundation Middleware Initiative Component in NMI Releases 4 and 5.

[51] Tandberg Accessed at http://www.tandberg.net/ Retrieved May 15, 2004.

[52] VCON Accessed at http://www.vcon.com/ Retrieved May 15, 2004.

[53] Polycom Accessed at http://www.polycom.com/ Retrieved May 15, 2004.

[54] Recommendation X.509 “X.509 (2000) | ISO/IEC 9594-8:2000”, International Telecommunication Union, Telecommunication Standardization Sector of ITU, Geneva, Switzerland, 2004.

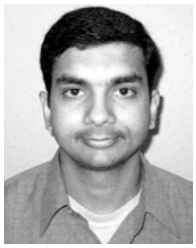
[55] Security Assertion Markup Language (SAML), an OASIS standard Accessed at http://www.oasis-open.org/ Retrieved May 15, 2004.

[56] Doster, W., Watts, M., Hyde, D., "The KX.509 Protocol", University of Michigan (undated) KX.509 is an NMI component. Accessed at http://www.citi.umich.edu/techreports/reports/citi-tr-01-2.pdf. Retrieved May 15, 2004.

[57] Donley, C., *LDAP Programming, Management and Integration*, Manning Publications, Greenwich CT pp. 268-270, 2003, p. 269.

Biographies

Jill Gemmill is Research Assistant Professor in the Department of Computer and Information Sciences at the University of Alabama at Birmingham (UAB) and also Assistant Director of Information Technology Academic Computing (ITAC). Jill received her B.A. in History from Antioch College, her M.S. in Computer and Information Sciences from UAB, and her MSEE in Electrical Engineering from UAB. Jill founded and directs the ITAC Advanced Technology Laboratory. Current interests include single-sign-on authentication architectures, transparent authorization across domains, secure videoconferencing, end-to-end application performance, 3D graphics and visualization, grid computing, and developing better collaboration tools for virtual organizations. She has received research funding from the National Science Foundation, National Institutes of Health, National Library of Medicine, Southeastern Universities Research Association, and the UAB Health Services Foundation. Jill's professional experience includes real-time data collection, computer graphics, and data representation with expertise in three-dimensional modeling and visualization for neuroscience. She has managed research computing facilities and created UAB's first "Virtual Help Desk". Jill is recognized for significant contributions to Alabama's Internet2 high-speed research and education network backbone. She is Past-Chair of ViDe, the video development initiative; Chair of the Internet2 Middleware-Enabled Mailing List Working Group; and a member of the Internet2 Health Sciences Leadership Team. She has been program chair of the 2002, 2003, and 2004 SURA/ViDe Digital Video Workshops and co-organizer of the 2001 Managing Digital Video Content Workshop.



Aditya Srinivasan works as a software engineer for Macquarium Intelligent Communications, Inc. in Atlanta, Georgia and is primarily focused on creating custom web solutions. He has extensive experience developing LDAP based applications. Aditya holds an M.S. in Computer and Information Sciences from the University of Alabama at Birmingham.

Jason L. W. Lynn is currently employed by the University of Alabama at Birmingham as an Information System Specialist II in IT Academic Computing's Advanced Technology Laboratory. He is the Testbed Manager for "ViDe.Net: Middleware for Scalable Video Services for Research and Higher Education", a National Science Foundation funded project. He is co-editor and an author of the Video Middleware Cookbook, the leading information source on deploying and developing H.350 components. Jason holds a Bachelor's Degree in Computer Engineering from Auburn University.

Samir Chatterjee received the B.E (Hons.) degree in Electronics Engineering from Jadavpur University, India and M.S. and Ph.D degree in computer science from the School of Computer Science, University of Central Florida, Orlando. He is an Associate Professor of Computer networking and telecommunications at Claremont Graduate University's School of Information Science. He joined CGU in July of 2001. He is the founding director of Network Convergence Lab at CGU. Prior to joining CGU, he was an Associate Professor of CIS at Georgia State University's Robinson College of Business in Atlanta. He had been on the faculty at GSU for seven years. His research interests are in designing algorithms and protocols for Quality-of-Service, Voice/Video over IP, and next-generation secured cyberinfrastructures. He is currently working on developing efficient Grid-enabled tools for bioinformatics. He has published over 40 articles in leading journals such as Communications of the ACM, Computer Communications, Computer Networks, Communications of the AIS, Information System Frontier and several ACM and IEEE conferences. He has been on the program committee of

several IEEE and ACM conferences and a regular participant at EntNet@Supercom. He has served as Principal Investigator on various NSF, BellSouth, Northrop-Grumman, Hitachi, nternet2 funded projects and serve as a consultant to top industry. He is secretary of IEEE Communication Society's EntNet and was the Program Chair for IEEE Healthcom 2003 and HICSS-36 Chair on Network Convergence. He is an Associate Editor of International Journal of Business Data Communications and Networking.

Bengisu Tulu is currently a doctoral student in Management Information Systems at Claremont Graduate University, School of Information Science.Her research interests lie in computer security, medical informatics, and specifically telemedicine. She is currently working on Secure Video Conferencing and Digital Signatures in the healthcare domain. Ms. Tulu holds a MS in MIS (Claremont Graduate University) as well as a BS in Mathematics (Middle East Technical University, Turkey). Ms. Tulu is currently a Research Associate at the Network Convergent Lab at Claremont Graduate University.

Tarun Abhichandani is a PhD student at School of Information Science in Claremont Graduate University (CGU) and a Research Associate at the Network Convergence Laboratory in CGU. He has extensive background in databases, software engineering and programming. In Network Convergence Laboratory, he has been involved in multi-institutional research centering on creating middleware for Video-Conferencing applications. In the past, he has held various positions while designing and administering organization-wide networking infrastructure, database applications and ERP systems. He holds a Masters degree in Management of Information Systems from CGU and a Masters degree in Finance from Mumbai University, India.