Distributed Access Management in Multimedia IDCs

Future Internet data centers that provide multimedia content will face security challenges requiring appropriately designed policies to manage resource access. An example from the healthcare domain shows how an access management framework can address these challenges.

Rafae Bhatti
Basit Shafiq
Mohamed Shehab
Arif Ghafoor
Purdue University

Advancements in the Internet and related technologies combined with the rapid proliferation of multimedia data on the Web have created tremendous opportunities for the business community to provide ubiquitous multimedia services.

Businesses can use the Web-based e-enterprise model not only to offer their services to a diverse and distributed clientele from a single online location but also to simplify the administration of such services. An underlying *Internet data center* architecture supports the storage and delivery of the massive amounts of multimedia data from this single virtual location to a huge clientele. Generally, third parties such as Akamai[1] own the IDC storage, computational, and networking infrastructure and charge the content provider for the hosted services.

While using IDCs lets the original source provide services to its subscribers through a third-party infrastructure, this environment's dynamic nature raises serious concerns regarding the management of access-control policies across heterogeneous enterprise domains. Using an IDC to serve multimedia data exacerbates the problems associated with providing complex access-control mechanisms that ensure secure dissemination of multimedia content on the Web. Since a manifold increase in the use of multimedia data will likely occur, this complexity threatens to affect the future use of IDCs to provide such services.

A practical scenario is a healthcare digital government initiative to provide online healthcare that several states have undertaken. Using IDC technology to archive, manage, and securely disseminate electronic clinical records of patients—from digital X-rays to videos of certified health practitioners performing diagnostic exams—can make a statewide healthcare system possible. For example, in the near future, the state of Indiana plans to set up a digital infrastructure to advance the provision of clinical services that includes, among other things, using high-end bioimaging facilities to remotely monitor and diagnose patients. However, the accessibility and provision of this type of multimedia data presents specific security and privacy concerns.[2]

IDC SECURITY CHALLENGES AND CONFIGURATION

The security challenges of an IDC that disseminates secure multimedia content vary depending on its configuration and the services it offers. For example, a simple data dissemination facility such as Yahoo Maps might not require explicit access-control. In other domains such as government, military, and healthcare, however, ensuring information confidentiality and integrity and enabling distributed collaboration are both paramount. These concerns require establishing elaborate access-management mechanisms based on the data's sensitivity level and timeliness.

The healthcare digital government initiative provides a useful example that demonstrates the need for secure provisioning of multimedia IDC-based services without adversely affecting the enterprise's

Published by the IEEE Computer Society
0018-9162/05/$20.00 © 2005 IEEE

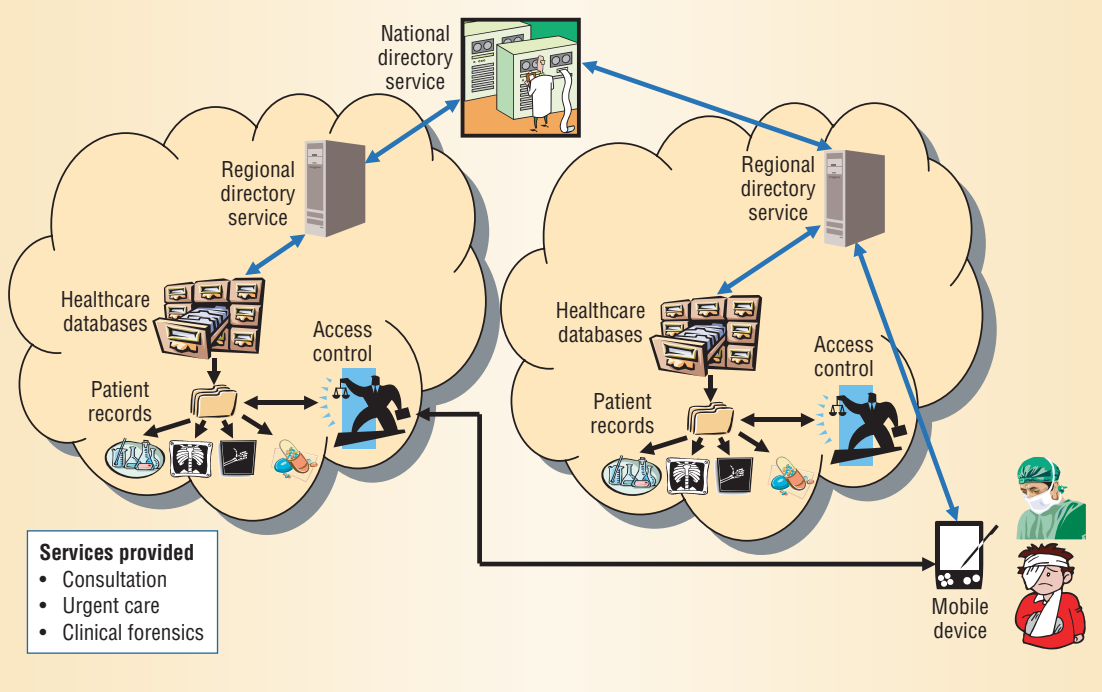

Figure 1. Multimedia IDC-based healthcare system. A network of collaborating IDCs is distributed across various regions in the state. The statewide directory service routes a user request for healthcare services from the local IDC to the appropriate regional IDC.

functional objectives. Combining a multimedia environment with strict access-management policies and cross-domain collaboration requirements in the IDC context is a challenging task requiring a framework that adequately addresses security concerns.[2]

To address this challenge, we propose a software architecture that abstracts the application-specific details and provides a generic design for implementing a framework that introduces the notion of service class descriptions as sources for composing access-control policies. The healthcare example illustrates the system configuration and service classification that drive our framework.

Figure 1 provides a high-level view of a multimedia IDC-based healthcare system. This statewide configuration of collaborating IDCs—extensible to a nationwide system—routes requests for access to healthcare resources to the appropriate IDC.

Each IDC consists of a *regional directory service* (RDS) and a *patient record database* (PRD). When the regional IDC receives a request, it queries its component RDS for the patient's records. If the record exists within this IDC, the RDS returns its location; otherwise the IDC reroutes the request to the *statewide directory service* to forward to the appropriate regional IDC, which eventually returns the resource's location. When it identifies the resource location, the original IDC submits an access request to the target IDC housing the resource. The target IDC follows its resource access-control policies with regard to releasing the requested information.

The security challenges in the context of this system configuration include multimedia content management, context-aware access control, and cross-domain collaboration.

Multimedia content management

The inherent complexity and varying levels of confidentiality of the multimedia data comprising a patient's clinical records pose several access-management challenges. The clinical data could include X-rays, electrocardiograms, or videotaped clips of medical examinations. Additionally, the patient's records could include prescription information or identification data stored as a barcode or a subscriber identity module card that serves as a biometric identifier.

Suppose that the situation depicted in Figure 1 involves a medical clinic offering a remote specialty consultation service to patients treated in a less-specialized clinic under a prior collaboration agreement. The specialized clinic's staff needs to access the patient's records and perform an evaluation and diagnostic procedure. Given the data's sensitive nature, however, the IDC should release it only under strict privacy policies. For example, the user's identity appearing on an X-ray should not be disclosed to the remote physician; in addition, it might be desirable to avoid revealing the patient's face if this physician needs to receive a clinical exam video clip to offer an expert opinion.

These concerns motivate providing a fine-grained content-based access-control mechanism that allows specification of access privileges within a media object. If the depicted situation is an instance of a clinical forensics service, a medical investigator might request a similarity match from an IDC's PRD based on collected evidence. Similarity-based queries require data classification based on features the object contains.

The IDC must classify, catalog, and process these multimedia data records appropriately on the fly

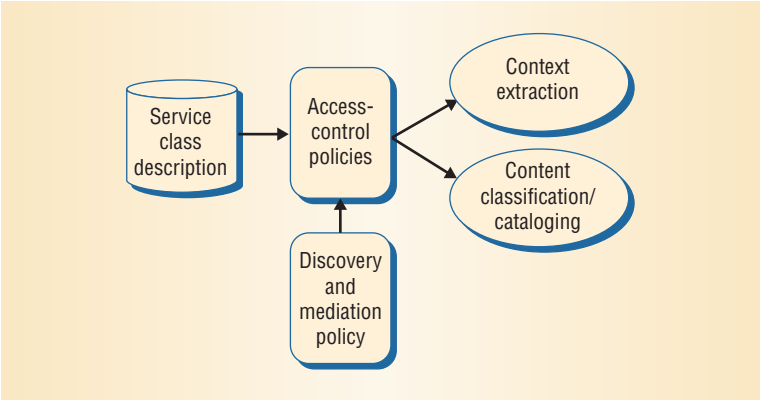

Figure 2. Design methodology. The service class description drives the access-control policy design. The access-control policy incorporates the discovery and mediation rules for enabling interoperation with policies from other domains. The policy also determines the context and content necessary for access management within the corresponding service class.

to ensure the release of only the appropriate information. Although mechanisms exist to handle multimedia content management issues,[2,3] the lack of a single mechanism that integrates classification, cataloging, and access management is an important concern.

Context-aware access control

The widely adopted ubiquitous computing paradigm places increasing demands on future IDCs to provide context-aware services tailored to the subscribers' current context. The context has two aspects: the user and the environment. The former establishes the identity of the users involved—such as doctor and patient—while the latter controls access to the information.

For example, the statewide healthcare system in Figure 1 can provide access to patients' information even outside their home area if required in, say, an urgent care situation. In such a scenario, the patient's user context identifies the patient's healthcare provider and locates the needed clinical records, while the doctor's user context provides access to the records. The urgent care environmental context has implications in the access-control and privacy policies controlling the release of sensitive information.

While emerging security frameworks[4] incorporate support for the context-aware access control necessary for such applications, researchers must address additional challenges. Proposed mechanisms for context collection, dissemination, and inference in a distributed environment are still in the initial stages.[5] Industry professionals now widely agree that context awareness will be a top priority in computing for the next decade, offering challenges that the research community is actively pursuing.[6]

Cross-domain collaboration

In the healthcare example's statewide system, the collaboration between the distributed enterprises responsible for sharing and disseminating the information content raises another important issue.[7] While collaboration and resource sharing expand information accessibility over multiple domains and thus enhance an enterprise's capability and marketability, they also raise serious security concerns.

As Figure 1 shows, healthcare providers use the directory services at each level to communicate. The infrastructure design must be scalable and flexible enough to avoid a bottleneck in the collaborative process. Additionally, such cross-domain collaboration requires a mediation policy governing access to sensitive data initiated from outside the domain housing the data. Enabling secure cross-domain collaboration is a key challenge in distributed environments.[8,9]

PROPOSED DESIGN METHODOLOGY

Figure 2 depicts the methodology that drives the design of our access management framework.

The *service class description* is a fundamental component of the overall design. The service classification refers to the categories of service the system provides to subscribers. This classification is important because the service class has implications for other system components' design. We therefore focus on the service class description as guiding data in the design process.

Based on the service offered, system designers must design an access-control policy for the resources needed to accomplish that service; this policy in turn drives the demand for the context information required to implement the policy. The requesting user's context determines whether it can access the resource. Additionally, the degree of access control depends on additional context collected when the system issued the request. Fine-grained access-control mechanisms can be used to control the requested content's release, making it necessary to catalog the data to support the access management requirements.

A component-based approach helps to identify the five main components in our framework:

- service class description,
- access-control policy,
- context set,
- content catalog, and
- mediation policy.

The last component occurs in a cross-domain collaborative environment.

Table 1. Criteria for healthcare-domain service class description.

Service class	Access control and privacy protection	Context awareness	Content classification/cataloging	Cross-domain collaboration mechanism
Consultation	*Intensive*–Patient data access must be protected; the privacy of patient data released for consultation must be preserved in accordance with the privacy policy.	*Moderate importance*–The user context must be known precisely; the environmental context blocks the release of information and is exercised at will.	*Moderate effort*–A basic cataloging of data based on a patient's identity attributes works because the data is always accessed specific to a patient whose identity is known.	*Hybrid*–The consultation services can be provided either through prearranged collaboration or on an ad hoc basis; hence, the system configuration could support both federated and loosely coupled collaborative environments.
Urgent care	*Concessional*–Access to patient data must be protected, but the privacy policy governing the release of patient data must have special allowances in emergency situations to avoid preventing access to critically needed records.	*High importance*–The user context must be known precisely. The environmental context authorizes the release of information and is mandatory; its integrity becomes very critical.	*Moderate effort*–Same as above.	*Loosely coupled*–Because no prearranged collaboration requirements can be posed in this service class, the system must support loosely coupled collaborative environments.
Clinical forensics	*Organizational*–Since legal and government domains are involved, the access protection for patient data depends on the criticality of the forensics process and the sensitivity level of the records.	*Low importance*–It is not necessary to precisely know the user's context; the environmental context can be supplied at will to refine the results.	*High effort*–Identity-based data cataloging does not suffice because forensics involves searching multiple patients' records for a similarity match based on a given piece of evidence.	*Federated*–The forensic services are typically provided only on a need-to-know basis through prearranged collaboration among medical institutions, government, and civic agencies; hence, the system must support a federated environment.

SERVICE DESCRIPTION

The service description incorporates the features necessary to identify a service. Assessing the significance level of the system components needed to describe a service requires careful analysis.

Table 1 describes the criteria for three representative service classes from the healthcare domain—consultation, urgent care, and clinical forensics—each of which puts a different emphasis on addressing a particular security challenge.

ACCESS-CONTROL POLICY

System designers create access-control policy specifications according to the service category. The policy design would therefore include the following procedures:

- specify and extract the relevant context and evaluate and incorporate it in the access-control decision;
- express fine-grained access constraints on the release of the requested content, possibly incorporating information privacy policies; and
- use a mediation policy to manage cross-domain accesses.

The access-control policy drives the design of the content extraction and content cataloging components. Based on a particular service category, each IDC composes its own access-control policy defining the context set and content catalog relevant to the target application. Since the access-control policy is the architecture's most idiosyncratic component, its standardization is a challenging task.

Because relatively well-known services and resources characterize domain-specific environments such as multimedia systems,[2] Web services,[4] and enterprise systems,[10] developers can capture their attributes in the access-control policy at design time. However, since the IDC system configuration could offer a multitude of service possibilities, the service description is not generally available. This introduces a considerable challenge because the service description guides the access-control policy design.

One possible mechanism for generalizing the service description uses standardized Web-based service description protocols, such as WSDL, and designs access-control policies based on these protocols.[4] However, designers must revisit this approach in the context of the service and resource heterogeneity challenges that highly distributed collaborative environments pose.

CONTEXT EXTRACTION

Designing a context-aware framework begins with defining the context relevant to the applica-

Table 2. An example of context inference.

Purported user ID	Location (zip code)	Time (est.)	Accessed resource	Inference
(Dr.) Smith	47906	11:22 p.m.	Jack's ECG	Normal
(Dr.) Smith	98101	11:41 p.m.	Jill's ECG	MLOH

tion. The IDC's access-control policy incorporates both the user and environmental contexts. The context information can include the location, time, bandwidth, available memory, and running processes.

Because the context describes the situation relevant to the application,[11] context identification and collection depend on the target application. Based on the service description and access-control policy, developers identify the relevant context and use a schema to represent it.

The terms *referring user* and *referred user* identify the users in a multiuser environment. In our healthcare example, the referring user is the physician, and the referred user is the patient. Each user's context is necessary to handle different aspects of the access-control problem. The referred user context locates the source of requested information and determines the information release policy, while the referring user context determines the level of access allowed on the requested content.

The user context can be represented using well-known forms of credentials, such as public-key certificates or other forms of authentication and authorization tokens. In fact, our software architecture uses this approach. However, few approaches address the challenges that the representation of environmental context poses.

One approach uses the notion of context sets that comprise parameters of interest for a given service class. It then uses this set to define context-based conditions in the access-control policy.[4] The access-control system evaluates these conditions at runtime by comparing the context parameters' predesignated values with the supplied context. However, to be usable with multiple service classes, the mechanism must be generalized, as described in the "Context awareness" column in Table 1. In addition, such a mechanism must facilitate the adequate collection, dissemination, and inference of context information.

Context collection and dissemination

Context collection and dissemination are particularly challenging in a multidomain environment in which various sensors and agents distributed all over the network monitor the context. Discovery and verification of the context information is a significant issue.

In the proposed IDC architecture, the host IDC maintains a list of relevant context sources (wherein the context schema provides relevance) to obtain regularly updated context information. These sources can include weather channels, the Global Positioning System, and the like. The collected context can be represented using well-known credential formats, and the source can digitally sign it for subsequent verification.

Since context information can include sensitive parameters that should not be released unconditionally, privacy policies can govern context dissemination. A credential-based format allows expressing these policies as rules on credential attributes. The proposed IDC architecture stores such policies in the context filter module in the host IDC domain.

Context inference

The raw information available from context sources may not be directly meaningful to the remote IDC, and it may be necessary to infer high-level context abstractions based on the service class.

The context inference process involves determining the correlation between multiple context parameters to establish a meaningful context—the *inferred context*. The inferred context can include user preferences, behavior, access patterns, and relationships with other users and system entities. The inferences are derived from the rules in the knowledge base that are specific to the target application domain (IDC), and they establish the relationship between context parameters.

Dependency models have been proposed for specifying the relationship among various context parameters.[12] Table 2 provides an example of a context-dependent access-control policy requiring context inference. Here, an accumulated set of context parameters is used to infer the high-level context abstraction (or access pattern) "hasAccessedFromMultipleLocationsInOneHour"(MLOH). The access-control policy could use this information, for example, to block subsequent access by the same user because of concerns about a security breach.

CONTENT CLASSIFICATION

Applying fine-grained access control on IDC resources requires a mechanism for content classification and cataloging. Developers generally consider content representation to be independent of the service description in traditional systems; however, this clearly is not appropriate for multimedia content management. The classification and cataloging process must take into account the service's needs and appropriately arrange a content catalog

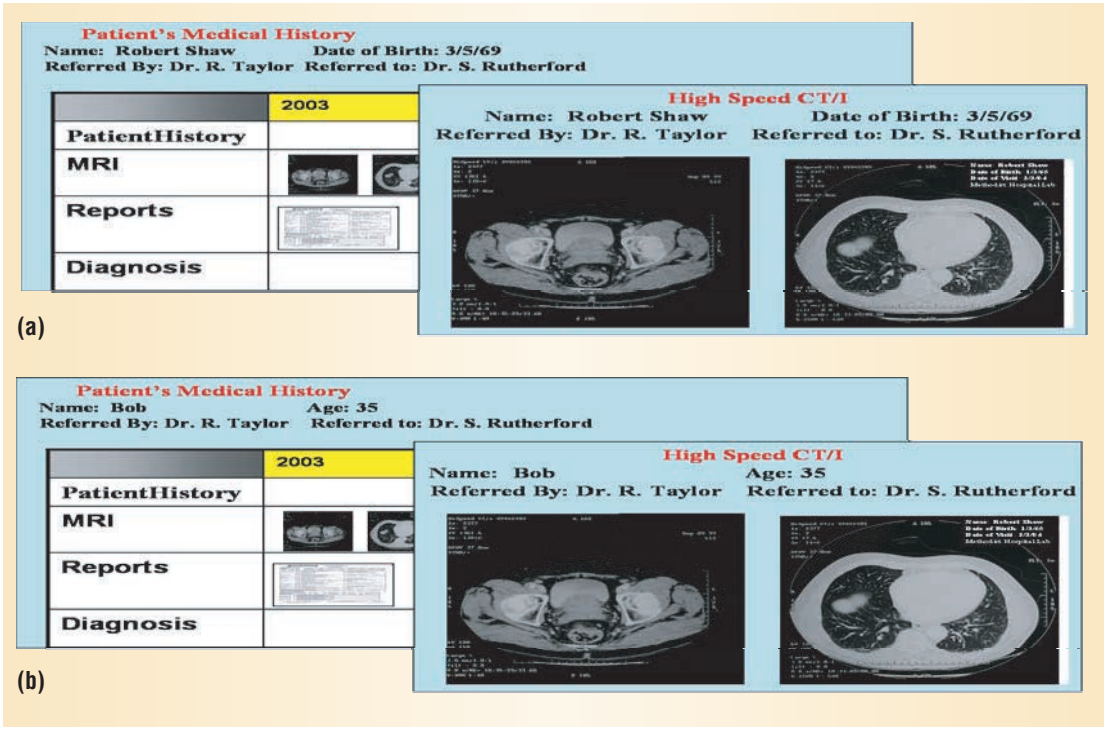

Figure 3. Two views of the same multimedia content. (a) A patient record that includes the patient's name and date of birth. (b) A filtered view of the record in which the patient's name has been masked and the date of birth has been changed to an age attribute.

for efficient data retrieval depending on the information release policies.

From the information retrieval perspective, a multimedia object—video, audio, image—is considered to be a monolithic entity with access privileges defined for the entire object. This monolithic view of multimedia content can significantly reduce information availability because of the coarser access granularity. Thus, the media content classification must be based on the information's sensitivity level. One such application is the KMed knowledge-based image retrieval system, which supports medical image retrieval based on different features and attributes embedded within an image.[3]

Various domain-specific standards have also evolved for content-based extraction and representation of information from media objects. Examples include the Picture Archiving and Communications System[13] and the Digital Imaging and Communications in Medicine[14] standards that define the requirements for archival and secure transmission of medical images. The DICOM standard also specifies the guidelines and associated data structure for cataloging semantic medical image information.

The information catalog also can include the corresponding information's sensitivity level. Accordingly, different users can view a given media object differently depending on the user's authorization. An access-management framework design for multimedia content can be based on multiple views for multiple access levels.[2]

Figure 3 illustrates the generation of multiple views from the same media content based on the access policy. In this scenario, Robert Shaw's primary physician is authorized to see all of his medical records. The primary physician could consult another physician for an expert opinion. However, the patient's privacy policy might not allow releasing any information that can identify the patient to anyone other than the primary physician. Therefore, the system creates a filtered version of the original view. For example, in Figure 3b, the patient's name changes from Robert Shaw to Bob, and the exact date of birth changes to an age attribute. Hence, a fine-grained multimedia access-control framework requires unification of the earlier approaches for classifying, cataloging, and filtering multimedia content.

DISCOVERY AND MEDIATION POLICY

Resource discovery and access mediation are critical components in a distributed collaborative IDC environment.

To address the discovery issue, a vertical hierarchy of directory services must be defined to locate the information source efficiently. At the lowest level, a directory service records the information that the IDC contains at a finer granularity. Directory components higher in the hierarchy track the information content of multiple domains at a much coarser granularity. As Figure 4 shows, resource accesses across heterogeneous IDCs follow a path along a horizontal hierarchy.

The information flow across these hierarchies must overcome any semantic heterogeneity issues, and developers must formulate languages that the system can use for discovery and mediation.

The query language for resource discovery is the

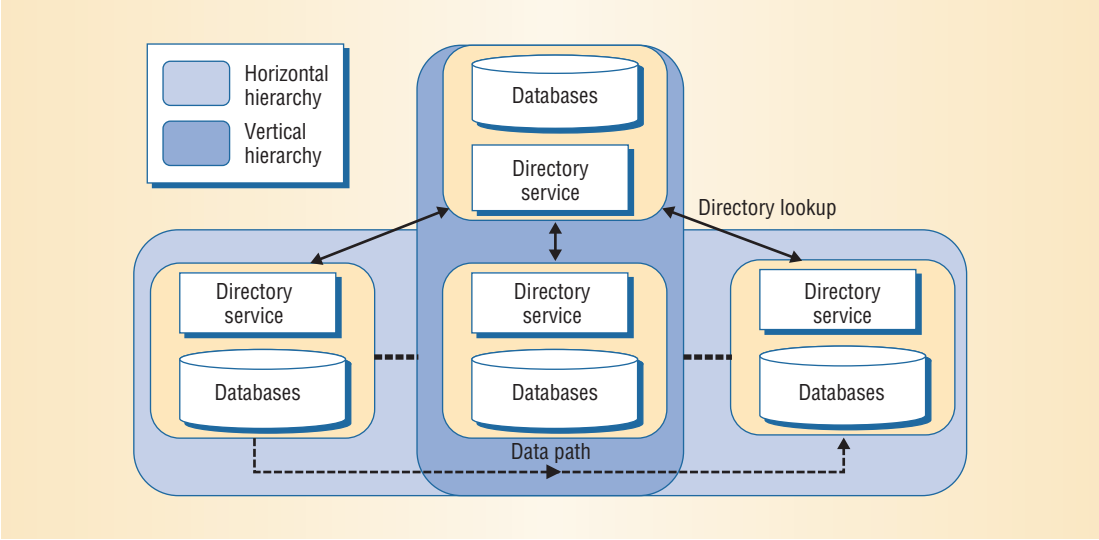

Figure 4. A system configuration representing vertical and horizontal IDC hierarchies. The resource discovery request takes a path along the vertical hierarchy to locate the appropriate IDC, whereas the actual resource access occurs along the horizontal hierarchy.

simpler of the two languages. It serves as an interface across the vertical chain of directory servers to facilitate information retrieval and can comprise a standardized format such as the Lightweight Directory Access Protocol. The mediation policy language, however, cannot be formulated independently of the service class, since the latter would influence the access mediation component design at each individual IDC across the horizontal hierarchy.

Access mediation deals with resolving conflicts arising due to semantic heterogeneity and policy integration. Based on the service description, the system configuration can either be closely coupled (federated), loosely coupled, or a mixture of the two (hybrid). While resolving semantic heterogeneity issues can be handled similarly in both environments, resolving policy-level conflicts in a loosely coupled system requires a different mediation mechanism. A combination of the two approaches can address conflicts in a hybrid environment.

Researchers have resolved semantic heterogeneity issues in the context of schema integration in databases,[8] and schema mismatch which is characterized by metamodel conflicts.[9] A comprehensive approach is necessary to address the semantic heterogeneity issue, possibly requiring the unification of the various approaches. Additionally, ontology-based mediation can play a significant role in this area.[15]

Conflicts arise when local access-control policies of collaborating domains are combined in an integrated policy governing interdomain information and resource exchanges between the various IDC domains. The resulting policy should not violate the principles associated with the interoperation environment, such as security and autonomy requirements of the constituent domains.[16]

These requirements can be well defined in a federated environment, and the mediation can be carried out through a static analysis. However, a loosely coupled environment requires a dynamic mediation mechanism because all knowledge of interoperation is not available in advance. Approaches to handle this scenario include using trust management to establish trusted interoperation among untrusted domains.[4] Once the interoperation is defined, the mediation policy can proceed as in a federated system.

SOFTWARE ARCHITECTURE AND PROTOCOL

A dedicated functional component implements each of the architecture's mechanisms. Figure 5 shows the overall architecture and provides an expanded view of an individual IDC from Figure 4.

The IDC collaboration protocol (ICP) describes the functionality of the architecture's components. System designers can use the protocol to incorporate support for secure interoperation and information dissemination between multimedia IDCs. The numbered arrows in Figure 5 represent the steps in our ICP.

Step 1. The user requesting data from an IDC must obtain well-recognized credentials to assist in the multicentric access control that occurs in distributed collaboration among IDCs.

Establishing the identity and capability of foreign entities—a key for mediation to proceed in such environments—requires a scalable identity and authorization management infrastructure. For this purpose, the architecture employs an authentication manager that is similar to any public-key infrastructure certification authority and an authorization manager that issues certificates to users.

Step 2. The authorization manager receives the authentication token, along with the authorization information pertaining to the user's local domain, such as identity or capability. The authorization manager issues an authorization token that uniquely and globally binds this user's identity with

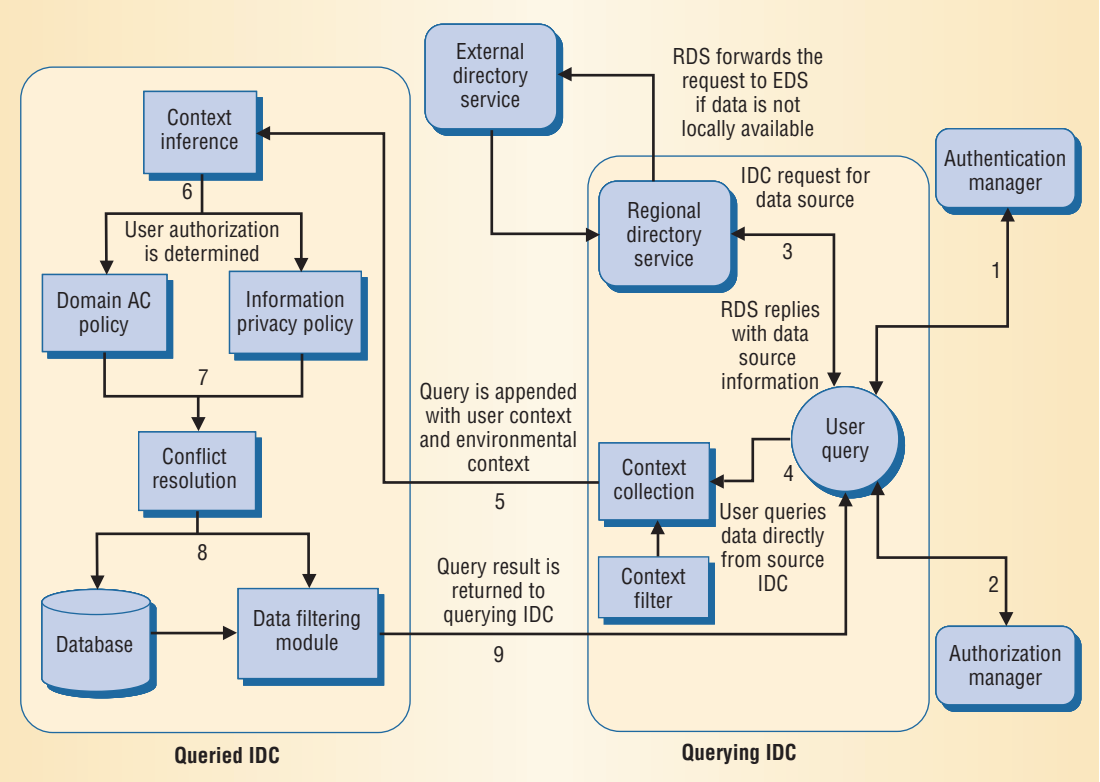

Figure 5. A partitioned view of the IDC architecture. The collaboration protocol supports secure interoperation and information dissemination among multimedia IDCs.

the capability in the user's parent domain. Because the authorization manager is a well-known entity, all collaborating domains accept the authorization token it issues.

Using trusted tokens allows verification of the foreign user's identity or capability, which provides the scalable delegation support that our system configuration requires.

In cases in which the authorization manager has limited functionality, the credential management task can be outsourced to existing credential management mechanisms designed for such purposes, like the W3C's XML Key Management Specification (www.w3.org/TR/xkms/).

Following steps 1 and 2, the system submits the user query to the IDC.

Steps 3-4. The IDC that receives the query requests information about the requested resource's location from its component RDS. If the record exists within this IDC, the RDS returns the resource's location; otherwise, the RDS forwards the request to the IDC's vertical hierarchy until it eventually receives the requested information.

In step 4, upon identifying the resource location, the original, querying IDC then submits the query to the queried IDC, which houses the resource.

Step 5. The IDC obtains the environmental context relevant to this request as indicated by the context schema prescribed by the applicable access-control policy for the requested service. It then appends the user context and environmental context to the access request and submits it to the queried IDC. This step assumes that the IDC is equipped with well-defined mechanisms for context collection and dissemination.

Step 6. The queried IDC's access-control module evaluates the query embedded with the context information. This consists of two phases: First, it checks the domain access control for the referring user's authorizations based on the supplied user and environmental context; second, it checks the referred user's information privacy policy for any restrictions on the requested content's release.

This step assumes that the IDC is equipped with an adequate context inference mechanism.

Step 7. Depending on the collaboration environment—federated, loosely coupled, or hybrid—the queried IDC can invoke a conflict-resolution module to mediate between the access-control policies of the two interacting IDC domains. The mediation mechanism uses the information in the referring user's authorization token to appropriately translate the user's access rights within the target domain defined by the queried IDC. The conflict-resolution manager implements this functionality.

Step 8. Based on the privacy policy, the system can omit information violating user preferences from the data's returned view. The user might be required to digitally sign the privacy policies to ensure integrity. The system retrieves the requested content from the IDC database and sends it to the data filtering module, which generates an appropriate view of the content.

Step 9. The final step returns the requested content's resulting view to the querying IDC.

Our architecture integrates access management mechanisms into the design of multimedia IDCs for secure dissemination of information content in a distributed collaborative environment. It also provides a service-driven, context-aware policy-design methodology that fits the host IDC's organizational needs.

This proposed architecture does not explicitly address networking issues related to IDC-based service provision mechanisms, including resource allocation and load balancing across multiple IDC domains, which are required to function in the widely distributed Internet environment.[1] However, we believe that context information will help provide a better mechanism for resource allocation, keeping in view user preferences and traffic patterns.

Several interesting issues related to our framework remain to be explored, especially in the emerging area of context awareness. Of particular significance will be mechanisms to generate representations of context schema to identify the relevant context for a given service class. A related challenge is to automate derivation of access-control policies from service class descriptions. We believe that the service class-driven design methodology will provide a viable solution to access-management challenges in multimedia IDCs and other distributed service provisioning architectures.

Our future research goals include addressing remaining design challenges and prototyping and evaluating our system architecture. ■

Acknowledgments

Portions of this work have been supported by the Center for Education and Research in Information Assurance and Security (CERIAS) at Purdue University and the National Science Foundation under NSF grants no. IIS-0242419 and IIS-0209111.

References

1. J. Dilley et al., "Globally Distributed Content Delivery," *IEEE Internet Computing*, Sept./Oct. 2002, pp. 50-58.
2. J. Joshi et al., "A Model for a Secure Multimedia Document Database System in a Distributed Environment," *IEEE Trans. Multimedia*, June 2002, pp. 215-234.
3. C. Chu et al., "Knowledge-Based Image Retrieval with Spatial and Temporal Constructs," *IEEE Trans. Knowledge and Data Eng.*, Nov. 1996, pp. 872-888.
4. R. Bhatti, E. Bertino, and A. Ghafoor, "A Trust-Based Context-Aware Access Control Model for Web Services," *Distributed and Parallel Databases*, July 2005, pp. 83-105.
5. H. Lei et al., "The Design and Applications of a Context Service," *ACM Sigmobile Mobile Computing and Communications Rev.*, Oct. 2002, pp. 45-55.
6. V. Kumar and S. Zdonik, "Workshop Report," *Proc. NSF Workshop Context-Aware Mobile and Sensor Information Management*, Jan. 2002; www.sice.umkc.edu/nsfmobile/wshop.html/report.pdf.
7. J. Joshi et al., "Digital Government Security Infrastructure Design Challenges," *Computer*, Feb. 2001, pp. 66-72.
8. A.P. Sheth and J.A. Larson, "Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases," *ACM Computing Surveys*, Sept. 1990, pp. 183-236.
9. R. Pottinger and P. A. Bernstein, "Merging Models Based on Given Correspondences," *VLDB J.*, 2003, pp. 826-873.
10. R. Bhatti et al., "X-GTRBAC: An XML-Based Policy Specification Framework and Architecture for Enterprise-Wide Access Control," *ACM Trans. Information and System Security*, vol. 8, no. 2, 2005, pp. 187-227.
11. A.K. Dey, "Understanding and Using Context," *Personal and Ubiquitous Computing*, vol. 5, no. 1, 2001, pp. 4-7.
12. M. Khedr and A. Karmouch, "ACAI: Agent-Based Context-Aware Infrastructure for Spontaneous Applications," *J. Network & Computer Applications*, vol. 28, no. 1, pp. 19-44.
13. K. Huang and R.K. Taira, "Infrastructure Design of a Picture Archiving and Communication System," *Am. J. Roentgenology*, vol. 158, 1992, pp. 743-749.
14. R. Noumeir, "DICOM Structured Report Document Type Definition," *IEEE Trans. Information Technology in Biomedicine*, Dec. 2003, pp. 318-328.
15. T. Tsai et al., "Ontology-Mediated Integration of Intranet Web Services," *Computer*, Oct. 2003, pp. 63-71.
16. L. Gong and X. Qian, "Computational Issues in Secure Interoperation," *IEEE Trans. Software Eng.*, Jan. 1996, pp. 43-52.

Rafae Bhatti *is a PhD candidate in the Purdue University School of Electrical and Computer Engineering. His research interests include information systems security, with emphasis on policy-based access management and secure interoperation in open collaborative systems. Bhatti received an MS in electrical and computer engineering from Pur-*

due. He is a student member of the IEEE and the ACM. Contact him at rafae@purdue.edu.

***Basit Shafiq** is a PhD candidate in the Purdue University School of Electrical and Computer Engineering. His research interests include information security, access-control management in distributed systems, and multimedia information systems. Shafiq received an MS in electrical and computer engineering from Purdue University. He is a student member of the IEEE. Contact him at shafiq@ecn.purdue.edu.*

***Mohamed Shehab** is a PhD candidate in the Purdue University School of Electrical and Computer Engineering. His research interests include information security, distributed access control, and distributed secure collaboration. Shehab received a BSc in electrical engineering from the United Arab Emirates University. Contact him at shehab@purdue.edu.*

***Arif Ghafoor** is a professor in the Purdue University School of Electrical and Computer Engineering. His research interests include multimedia systems, databases, distributed computing systems, and broadband multimedia networking. Ghafoor received a PhD in electrical engineering from Columbia University. Contact him at ghafoor@ecn.purdue.edu.*