Designation: E2595 – 07 (Reapproved 2013)

An American National Standard

Standard Guide for Privilege Management Infrastructure[1]

This standard is issued under the fixed designation E2595; the number immediately following the designation indicates the year of original adoption or, in the case of revision, the year of last revision. A number in parentheses indicates the year of last reapproval. A superscript epsilon (ε) indicates an editorial change since the last revision or reapproval.

INTRODUCTION

This guide arises from the ongoing development and implementation of privilege management infrastructures (PMIs) within the healthcare environment. The healthcare environment supported by this guide is enterprise-wide and extends beyond traditional borders to include external providers, suppliers, and other healthcare partners. This guide supports privilege management within distributed computing as well as service-oriented architecture environments. This guide supports a distributed security environment in which security is also a distributed service.

The healthcare sector is continually improving the delivery of care by leveraging technical advances in computer-based applications. Health professionals are increasingly accessing multiple applications to schedule, diagnose, and administer patient care. These disparate applications are typically connected to a common network infrastructure that typically supports patient, business, and nonbusiness services, communications, and protocols. Because increased access is made possible through a common network infrastructure, secure access to these distributed, and often loosely coupled applications, is even more important than when these applications were accessed as stand-alone devices.

Secure access to legacy computer-based healthcare applications typically involves authentication of the user to the application using single-factor identification, such as a password, or multifactor identification, such as a password combined with a token or biometric devices. After authentication, the application determines the authority that user may have to use aspects of the application. Determining the level of authority a user has is typically done, if at all, by each application. The application may restrict operations (such as read, write, modify, or delete) to an application-specific group or role affiliation. Authenticated users are frequently associated with groups or roles using a local database or flat file under the control of an application administrator.

The use of a local mechanism for authorization creates a patchwork of approaches difficult to administer centrally across the breadth of a healthcare enterprise. That is, the software logic determining authorization is distinctive to each application. In some cases, applications can be adapted to use a network database that contains a trusted source of name-value pairs. This information allows applications to determine the user's group or role affiliation. This approach permits centralized control over a shared user base. However, the resulting granularity of control over user authorization is coarse and shall be interpreted by each application specialist. Granularity of user authority can only be improved by increasing the number of application-specific groups or roles in the shared database. Storing information specific to each application causes exponential growth of roles per user and results in provisioning difficulties. The better solution is to associate industry standard permissions to users. Each application can examine the permissions listed for a user and determine their level of authorization regardless of their group affiliation within the healthcare organization.

The resulting system is a PMI. By the nature of the problem, the privileges shall be defined in an industry standard way. This guide will discuss various aspects of identifying a PMI standard to vendors providing healthcare applications to the contemporary healthcare enterprise.

1. Scope

1.1 This guide defines interoperable mechanisms to manage privileges in a distributed environment. This guide is oriented towards support of a distributed or service-oriented architecture (SOA) in which security services are themselves distributed and applications are consumers of distributed services.

1.2 This guide incorporates privilege management mechanisms alluded to in a number of existing standards (for

Copyright © ASTM International, 100 Barr Harbor Drive, PO Box C700, West Conshohocken, PA 19428-2959. United States

Documents
Document Sharing Hub

Full document details: ASTM E2595-07(2013)

example, Guide E1986 and Specification E2084). The privilege mechanisms in this guide support policy-based access control (including role-, entity-, and contextual-based access control) including the application of policy constraints, patient-requested restrictions, and delegation. Finally, this guide supports hierarchical, enterprise-wide privilege management.

1.3 The mechanisms defined in this guide may be used to support a privilege management infrastructure (PMI) using existing public key infrastructure (PKI) technology.

1.4 This guide does not specifically support mechanisms based on secret-key cryptography. Mechanisms involving privilege credentials are specified in ISO 9594-8:2000 (attribute certificates) and Organization for the Advancement of Structured Information Standards (OASIS) Security Assertion Markup Language (SAML) (attribute assertions); however, this guide does not mandate or assume the use of such standards.

1.5 Many current systems require only local privilege management functionality (on a single computer system). Such systems frequently use proprietary mechanisms. This guide does not address this type of functionality; rather, it addresses an environment in which privileges and capabilities (authorizations) shall be managed between computer systems across the enterprise and with business partners.

1.6 *This standard does not purport to address all of the safety concerns, if any, associated with its use. It is the responsibility of the user of this standard to establish appropriate safety and health practices and determine the applicability of regulatory limitations prior to use.*

2. Referenced Documents

2.1 *ASTM Standards:*[2]

E1762 Guide for Electronic Authentication of Health Care Information

E1985 Guide for User Authentication and Authorization

E1986 Guide for Information Access Privileges to Health Information

E2084 Specification for Authentication of Healthcare Information Using Digital Signatures (Withdrawn 2009)[3]

E2212 Practice for Healthcare Certificate Policy

2.2 *ANSI Standards:*[4]

X9.45 Enhanced Management Controls Using Digital Signatures and Attribute Certificates

INCITS 359 Role-Based Access Control

2.3 *HL7 Standard:*[5]

Health Level 7 Context Management "CCOW" (Clinical Context Object Workgroup) Standard, Version 1.5

2.4 *IETF Standards:*[6]

RFC 3198 Terminology for Policy-Based Management

RFC 3280 Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile

RFC 3881 Security Audit and Access Accountability Message XML Data Definitions for Healthcare Applications

2.5 *ISO Standards:*[7]

ISO 9594-8 The Directory: Public-Key and Attribute Certificate Frameworks; also available as ITU-T X.509: 2000

ISO 10181-3-00 Security Frameworks for Open Systems: Access Control Framework; also available as ITU-T X.812: 1995

ISO/TS 21298 Functional and Structure Roles

ISO/TS 22600-2:2006 Health Informatics—Privilege Management and Access Control—Part 2: Formal Models

2.6 *OASIS Standards:*[8]

Security Assertion Markup Language (SAML) v2.0

SAML 2.0 Profile of XACML

Security Provisioning Markup Language (SPML) v2.0, (OASIS)

Web Services Business Process Execution Language (WS-BPEL v2)

WS-Trust (WS-Trust 1.3)

eXtensible Access Control Markup Language (XACML) v2.0

XACML Profile for Role Based Access Control (RBAC): Committee Draft 01

XACML Profile for Web Services (WS-XACML)

2.7 *NIST Standards:*

NIST Special Publication 800-33 Underlying Technical Models for Information Technology Security, (Stoneburner), December 2001

NIST Special Publication 800-95 (Draft) Guide to Secure Web Services, (Singhal, et al), September 2006

NIST Special Publication 800-100 Information Security Handbook: A Guide for Managers, (Bowen, et al), October 2006

FIPS PUB 66 Standard Industrial Classification (SIC) Codes[9]

3. Terminology

3.1 *Definitions:*

3.1.1 *access control decision function (ADF), n*—specialized function that makes access control decisions by applying access control policy rules to a requested action; see *policy decision point*.

[1] This guide is under the jurisdiction of ASTM Committee E31 on Healthcare Informatics and is the direct responsibility of Subcommittee E31.25 on Healthcare Data Management, Security, Confidentiality, and Privacy.

Current edition approved March 1, 2013. Published March 2013. Originally approved in 2007. Last previous edition approved in 2007 as E2595–07. DOI: 10.1520/E2595-07R13.

[2] For referenced ASTM standards, visit the ASTM website, www.astm.org, or contact ASTM Customer Service at service@astm.org. For *Annual Book of ASTM Standards* volume information, refer to the standard's Document Summary page on the ASTM website.

[3] The last approved version of this historical standard is referenced on www.astm.org.

[4] Available from American National Standards Institute (ANSI), 25 W. 43rd St., 4th Floor, New York, NY 10036, http://www.ansi.org.

[5] Available from Health Level Seven, Inc., 3300 Washtenaw Ave., Suite 227, Ann Arbor, MI 48104.

[6] Available from Internet Engineering Task Force, www.ieft.org/rfc.html.

[7] Available from International Organization for Standardization (ISO), 1 rue de Varembé, Case postale 56, CH-1211, Geneva 20, Switzerland, http://www.iso.ch.

[8] Available from www.oasis-open.org/specs/index.php.

[9] Withdrawn Feb. 8, 2005.

AVA Documents
Document Sharing Hub

Full document details: ASTM E2595-07(2013)

3.1.2 *access control enforcement function (AEF), n*—specialized function that is part of the access path between a requestor and a protected resource that enforces the decisions made by the ADF; see *policy enforcement point*.

3.1.3 *access control information (ACI), n*—any information used for access control purposes, including contextual information.

3.1.4 *attribute certificate (AC), n*—data structure that includes some attribute values and identification information about the owner of the attribute certificate, all digitally signed by an attribute authority (this includes certificates that an authority issues to itself) and this authority's signature serves as the guarantee of the binding between the attributes and their owner.

3.1.4.1 *Discussion*—Types: role specification and role assignment described in Ref (**1**).[10]

3.1.5 *attribute authority (AA), n*—authority, trusted by the verifier to delegate privilege, that issues attribute certificates.

3.1.6 *attribute authority revocation list (AARL), n*—revocation list containing attribute certificates issued to attribute authorities that are no longer considered valid by the certificate issuer.

3.1.7 *attribute certificate revocation list (ACRL), n*—revocation list containing attribute certificates issued to claimants that are no longer considered valid by the certificate issuer.

3.1.8 *authority, n*—entity responsible for the issuance of certificates.

3.1.8.1 *Discussion*—Two types are defined in this guide: certificate authority that issues public-key certificates and attribute authority that issues attribute certificates.

3.1.9 *authorization, n*—granting of rights that includes the granting of access based on access rights.

3.1.10 *authorization credential, n*—signed assertion of a user's permission attributes.

3.1.11 *authority revocation list (ARL), n*—revocation list containing public-key certificates issued to authorities that are no longer considered valid by the certificate issuer.

3.1.12 *authority certificate, n*—certificate issued to an authority (for example, either to a certification authority or to an attribute authority).

3.1.13 *business partner agreement, n*—document used to demarcate the legal, ethical, and practical responsibilities between subscribers to a privilege management infrastructure (PMI) and between cooperating PMI implementations.

3.1.14 *certificate revocation list (CRL), n*—signed list indicating a set of certificates that are no longer considered valid by the certificate issuer.

3.1.14.1 *Discussion*—CRLs may be used to identify revoked public-key certificates or attribute certificates and may represent revocation of certificates issued to authorities or to users. The term CRL is also commonly used as a generic term applying to all the different types of revocation lists, including CRLs, ARLs, ACRLs, and so forth.

3.1.15 *certificate validation, n*—process of ensuring that a certificate is valid, including possibly the construction and processing of a certification path, and ensuring that all certificates in that path have not expired or been revoked.

3.1.16 *claimant, n*—entity requesting that a sensitive service be performed or provided by a verifier based on the claimant's privileges as identified in its proffered attribute assertion, attribute certificate, or subject directory attributes extension of their public-key certificate.

3.1.17 *credential, n*—information describing the security attributes (identity or privileges or both) of a user or other principal.

3.1.17.1 *Discussion*—Credentials are claimed through authentication or delegation and used by access control.

3.1.18 *delegation, n*—conveyance of privilege from one entity that holds such privilege to another entity.

3.1.19 *delegation path, n*—ordered sequence of credentials that can be processed to verify the authenticity of a claimant's privilege.

3.1.20 *domain, n*—set of objects that a subject is allowed to access.

3.1.21 *environmental variables, n*—those aspects of policy required for an authorization decision that are not contained within structural structures but are available through some local means to a verifier (for example, time of day or current account balance).

3.1.22 *functional role, n*—job function within the context of an organization whose permissions are defined by operations on tasks, scenarios, aggregations, or data objects.

3.1.22.1 *Discussion*—Functional roles provide detailed permissions defining what a user can do within the context of an application. Examples include permissions to create an order, permission to sign a check, permission to read a database row, and so forth. A functional role applies to a workflow's individual process tasks.

3.1.23 *interoperable role, n*—as defined by HL7, a job function within the context of two or more organizations representing the lowest common level of interoperable permissions defined by a standardized vocabulary.

3.1.24 *owner, n*—entity to whom some privilege has been delegated either directly from the source of authority or indirectly through another attribute authority.

3.1.24.1 *Discussion*—An owner asserts its claim to that privilege by presenting authoritative credentials to a verifier and acting as a claimant for its privilege.

3.1.25 *permission, n*—approval to perform an operation on one or more protected objects.

3.1.26 *permission attributes, n*—operations and objects that define a permission.

3.1.27 *policy, n*—a set of rules, and an identifier for the rule-combining algorithm and (optionally) a set of obligations (OASIS XACML).

[10] The boldface numbers in parentheses refer to the list of references at the end of this standard.

AVA Documents
Document Sharing Hub

Full document details: ASTM E2595-07(2013)

3.1.28 *policy decision point (PDP), n*—system entity that evaluates applicable policy and renders an authorization decision.

3.1.28.1 *Discussion*—This term is defined in a joint effort by the IETF Policy Framework Working Group and the Distributed Management Task Force (DMTF)/Common Information Model (CIM) in RFC 3198. This term corresponds to access decision function (ADF) in ISO 10181-3-00.

3.1.29 *policy enforcement point (PEP), n*—system entity that performs access control by making decision requests and enforcing authorization decisions.

3.1.29.1 *Discussion*—This term is defined in a joint effort by the IETF Policy Framework Working Group and the Distributed Management Task Force (DMTF)/Common Information Model (CIM) in RFC 3198. This term corresponds to access enforcement function (AEF) in ISO 10181-3-00.

3.1.30 *privilege, n*—capacity assigned to an entity by an authority.

3.1.31 *privilege management infrastructure (PMI), n*—complete set of processes required to provide an authorization service.

3.1.32 *privilege policy, n*—policy that outlines conditions for verifiers to provide/perform sensitive services to/for qualified claimants.

3.1.32.1 *Discussion*—Privilege policy relates attributes associated with the service as well as attributes associated with claimants.

3.1.33 *public key infrastructure (PKI), n*—complete set of processes required to provide encryption and digital signature services.

3.1.34 *relying party, n*—user or agent that relies on data in an attribute assertion or certificate to make an access control decision.

3.1.35 *role, n*—job function within the context of an organization with some associated semantics regarding the authority and responsibility conferred on the user assigned to the role.

3.1.36 *role certificate, n*—certificate that assigns privileges to a role rather than directly to individuals.

3.1.36.1 *Discussion*—Individuals assigned to that role, through an attribute certificate or public-key certificate with a subject directory attributes extension containing that assignment, are indirectly assigned the privileges contained in the role certificate. Some PMIs use two types of X.509 attribute certificates: *(1)* role-specification AC held by the role containing privileges held by the role, and *(2)* role-assignment AC held by the user containing roles assigned to the user.

3.1.37 *sensitivity, n*—characteristic of a resource that implies its value or importance.

3.1.38 *source of authority (SoA), n*—special type of attribute authority upon which a verifier endows unlimited privilege.

3.1.38.1 *Discussion*—The verifier trusts the SoA to delegate that privilege to owners, some of whom may further delegate that privilege to other owners.

3.1.39 *security domain, n*—set of subjects, their information objects, and a common security policy (NIST Special Publication 800-33).

3.1.40 *security policy, n*—statement of required protection of the information objects.

3.1.41 *structural role, n*—job function within the context of an organization whose permissions are defined by operations on workflow objects.

3.1.41.1 *Discussion*—Structural roles provide authorizations on objects at a global level without regard to internal details. Examples include authorization to participate in a session, connect authorization to a database, authorization to participate in an order workflow, or connection to a protected uniform resource locator (URL). A structural role applies to the business process task as a group.

3.1.42 *target, n*—resource being accessed by a claimant.

3.1.43 *verifier, n*—entity responsible for performing or providing a sensitive service for/to qualified claimants.

3.1.43.1 *Discussion*—The verifier enforces the privilege policy. When validating certification paths, a verifier is a type of relying party.

3.1.44 *workflow, n*—representation of an organizational or business process in which documents, information, or tasks are passed from one participant to another in a way that is governed by rules or procedures; a workflow separates the various activities of a given organizational process into a set of well-defined tasks.

3.2 *Acronyms:*

3.2.1 *AA*—attribute authority

3.2.2 *AARL*—attribute authority revocation list

3.2.3 *ACRL*—attribute certificate revocation list

3.2.4 *ADF*—access control decision function

3.2.5 *AEF*—access control enforcement function

3.2.6 *AC*—attribute certificate

3.2.7 *ACI*—access control information

3.2.8 *ADF*—access decision function

3.2.9 *ADI*—access control decision information

3.2.10 *AEF*—access enforcement function

3.2.11 *ANSI*—American National Standards Institute

3.2.12 *ARL*—authority revocation list

3.2.13 *BPEL*—business process execution language

3.2.14 *CA*—certification authority

3.2.15 *CCOW*—clinical context object workgroup

3.2.16 *CIM*—common information model

3.2.17 *CORBA*—common object request broker architecture

3.2.18 *COTS*—commercial off the shelf

3.2.19 *CPU*—central processing unit

3.2.20 *CRL*—certificate revocation list

3.2.21 *DEA*—Drug Enforcement Agency

3.2.22 *DHHS*—Department of Health and Human Services

3.2.23 *DMTF*—Distributed Management Task Force

3.2.24 *DSA*—digital signature algorithm

3.2.25 *DTD*—document-type definition

AVA Documents
Document Sharing Hub

Full document details: ASTM E2595-07(2013)

3.2.26 *EHR*—electronic health record

3.2.27 *EIS*—entity identification service

3.2.28 *ER*—emergency room

3.2.29 *FIPS*—Federal Information Processing Standards

3.2.30 *HL7*—Health Level Seven

3.2.31 *ID*—identification

3.2.32 *IdM*—identity management

3.2.33 *IDS*—intrusion detection system

3.2.34 *IETF*—Internet Engineering Task Force

3.2.35 *INCITS*—International Committee for Information Technology Standards

3.2.36 *I/O*—input/output

3.2.37 *ISO*—International Organization for Standardization

3.2.38 *LDAP*—lightweight directory access protocol

3.2.39 *MDS*—multiple document signatures

3.2.40 *MOU*—memorandum of understanding

3.2.41 *NIST*—National Institute of Standards and Technology

3.2.42 *OASIS*—Organization for the Advancement of Structured Information Standards

3.2.43 *OCSP*—online certificate status protocol

3.2.44 *OMG*—object management group

3.2.45 *PA*—privilege allocators

3.2.46 *PAP*—policy administration point

3.2.47 *PDP*—policy decision point

3.2.48 *PEP*—policy enforcement point

3.2.49 *PHR*—personal health records

3.2.50 *PII*—personally identifiable information

3.2.51 *PKI*—public-key infrastructure

3.2.52 *PMI*—privilege management infrastructure

3.2.53 *PPS*—permission policy set

3.2.54 *RBAC*—role-based access control

3.2.55 *RIM*—reference implementation model

3.2.56 *RPS*—role policy set

3.2.57 *RuBAC*—rule-based access control

3.2.58 *SAML*—security assertion markup language

3.2.59 *S/MIME*—secure/multipurpose internet mail extensions

3.2.60 *SOA*—service-oriented architecture

3.2.61 *SoA*—source of authority

3.2.62 *SOAP*—simple object access protocol

3.2.63 *SPML*—security provisioning markup language

3.2.64 *UDDI*—universal description, discovery, and integration

3.2.65 *UHID*—universal healthcare identifier

3.2.66 *UML*—unified modeling language

3.2.67 *URI*—uniform resource identifier

3.2.68 *URL*—uniform resource locator

3.2.69 *WS-XACML*—XACML profile for web services

3.2.70 *XACML*—eXtensible access control markup language

3.2.71 *XML*—eXtensible markup language

3.3 *Terminology Comparison*—See Table 1 for terminology comparison.

4. Significance and Use

4.1 Motivation for the PMI comes from several organizational and application areas. For example:

4.1.1 Supporting a distributed heterogeneous application architecture with a homogeneous distributed security infrastructure leveraged across the enterprise; providing user and service identities and propagation; and providing a common, consistent security authorization and access control infrastructure.

4.1.2 Providing mechanisms to describe and enforce enterprise security policy systematically throughout the organization for consistency, maintenance, and ease of modification and to demonstrate compliance to applicable regulation and law.

4.1.3 Providing support for distributed/service-oriented architectures in which enterprise-wide services and authoritative sources are protected by providing security services that themselves are also distributed using common interfaces and communication protocols.

4.1.4 Providing "economies of scale" where it is desired to change the approach of individually managing the configuration of each point of enforcement to one that establishes a consolidated view of the safeguards in effect throughout the enterprise.

4.1.5 Providing centralized control, management, and visibility to security policy across the enterprise and when connecting to other organizations. This allows for additional key features such as delegated administration, centralized policy analysis, and consolidated reporting.

TABLE 1 Terminology Comparison

ISO	Non ISO
Example: (ISO) 10181-3-00	Example: OASIS XACML
Access Control Enforcement Function (AEF)	Policy Enforcement Point (PEP)
Access Control Decision Function (ADF)	Policy Decision Point (PDP)
Access Control Decision Information (ADI)	Request Context
Initiators	Access Requestor
Target	Resource
ADI Element (format not specified)	Attribute (XML Format)
Initiator ADI (format not specified)	Subject (XML Format)
Attribute Certificate	(not specified)
Rule Element (format not specified)	Condition
Claimant (not in 10181-3-00)	Access Requestor
Context ACI	Environment
Access Control Policy (format not specified)	Policy (XML Format)
Target	Target
(not in 10181-3-00)	Role Policy Set
(not in 10181-3-00)	Permission Policy Set
Role Specification (not in 10181-3-00)	
Role Assignment (not in 10181-3-00)	

AVA Documents
Document Sharing Hub

Full document details: ASTM E2595-07(2013)

4.1.6 Providing a distributed computing security architecture allowing for synchronized security services that are efficiently maintained across the enterprise while also allowing for centralized policy control and distributed policy decision-making/enforcement. Ensuring proper security controls are enacted for each service and when used in combination.

4.1.7 Provisioning incremental updates to policy and configuration data simultaneously across all distributed decision/enforcement points. Establishing and enforcing new policies not envisioned when individual applications were fielded and adapting to new requirements and threats. Managing identity and security implemented in a diverse mix of new and old technologies.

4.1.8 Permitting an organization to grant, suspend, or revoke centrally any or all ability to connect to or access enterprise resources either individually or collectively and with the capability to enforce these policies at run-time.

4.1.9 Supporting access decisions that are sensitive to a user's credentials in addition to identity. For example, the user may have to be a licensed healthcare professional to access a medical record.

4.1.10 *Supporting Delegation*—A user might delegate access for a resource to another user (for example, a physician might delegate access to his patient's records to a specialist). This shows the need for a delegation capability for some applications.

4.1.11 *Supporting Sender Verification*—When a user receives a signed document, he shall be sure the sender was, in some sense, authorized to sign and send the document. A simple example would be a prescription that shall be signed by a doctor. A simple identity certificate is insufficient, as it does not indicate the sender's credentials (that is, that he is a doctor).

4.1.12 *Supporting Document Cosigning*—Multiple examples exist in which more than one signature is required on a document (**2**). For example, a transcriptionist transcribes and signs a document, but it is not a valid part of the record until it is reviewed and signed by the primary care physician. Similar mechanisms can be used to provide cosignature controls when processing claims transactions. These types of applications require the ability to convey user authorizations (in assertions, credentials, authorization certificates, or possibly as extensions in identity certificates), to label documents and other objects with their security attributes (or to extract such attributes from the document), and to express authorization rules in machine-readable form.

4.2 Existing standards, including ANSI X9.45, ISO 9594-8, IETFRFC 3280 X.509, OASIS SPML, SAML, WS-*, and XACML, define a number of mechanisms that can be used to construct a healthcare-specific PMI specification. This would include the following features:

4.2.1 Privileges needed to access a target are conveyed in a claimant's authorization credential. The claimant's authorization credential may be an authorization certificate compliant with ISO 9594-8 (a particular form of attribute certificate) or a policy set description compliant with XACML or other referenced authorization standards.

4.2.2 The sensitivity or other properties of the target being accessed may be held in a local database or in a signed data structure. This guide does not define a standard way to represent this information, since this is a local matter. It does provide guidance on how such information might be represented and manipulated using common mechanisms such as ASN.1 and XML. For a given target object, there may be multiple operations that may be performed; each such operation may have a different set of sensitivity attributes.

4.2.3 The privilege policy may be held centrally, locally, or may be conveyed as a signed data structure. Different operations on a target may be subject to different privilege policies. This guide defines several standard policies, and applications may define additional policies.

4.2.4 In the document authorization paradigm, cosignature requirements may be associated with a user or document, such that the signed document is considered authorized only if all necessary signatures are attached.

4.2.5 Users may delegate privileges to other users.

4.2.6 Users may be assigned to roles that convey permissions.

4.2.7 Some authorizations may be sufficiently dynamic that it is not feasible to place them in an enterprise authorization infrastructure (that is, the cost of maintenance is too high given the short lifetime or rapid frequency of change of the privileges or constraints). Such authorizations may be kept in a local authorization server's database and accessed as environmental variables.

4.3 The remaining sections of this guide discuss mechanisms to convey privilege, sensitivity, and policy information in a distributed PMI.

5. Models

5.1 *General:*

5.1.1 Privilege management and authorization may be assigned to individual actors or to groups of individual actors playing the same role. Actors interacting with system components are called principals and can be a human user, system, device, application, component, or even an object.

5.1.2 Privilege management and access control management models typically contain entities and acts (**3**). Examples of which include:

5.1.2.1 *Entities:*

(1) Principals,
(2) Policies,
(3) Documents, and
(4) Roles.

5.1.2.2 *Acts:*

(1) Policy management,
(2) Principal management,
(3) Privilege management,
(4) Authentication,
(5) Authorization,
(6) Access control management, and
(7) Audit.

5.1.3 To obtain the above described structure and functionality, a number of models, mechanisms, processes, objects, and so forth, are needed. This section considers the following security models:

5.1.3.1 Domain model,

Documents
Document Sharing Hub

Full document details: ASTM E2595-07(2013)

5.1.3.2 Control model,
5.1.3.3 Delegation model,
5.1.3.4 Document model,
5.1.3.5 Policy model,
5.1.3.6 Role model,
5.1.3.7 Information distance model, and
5.1.3.8 Authorization model.

5.1.4 For enabling future-proof electronic health record (EHR) systems, all specifications made must be kept open, platform-independent, portable, and scalable. Therefore, the models provided will be described at the meta-model level and at the model level, keeping the instance level out of consideration.

5.2 *Domain Model:*

5.2.1 To manage and operate complex information systems that support shared care, principal-related components of the system are grouped into domains by common organizational, logical, and technical properties. Following Object Management Group's (OMG) definition, this could be done for common policies (policy domains), for common environments (environment domains), or common technology (technology domains). Any kind of interoperability internal to a domain is called an intradomain communication and cooperation, whereas interoperability between domains is called an interdomain communication and cooperation. For example, communication could be between departments of a hospital internally to the domain hospital (intradomain communication) but externally to the domain of a special department (interdomain communication). Real-world systems are most likely composed of multidomain information objects that cut across different information contexts.

5.2.2 As used here, a security domain is a set of subjects, their information objects, and a common security policy (NIST Special Publication 800-33). A domain is characterized by a domain identifier, domain name, domain authority, and domain qualifier (ISO/TS 22600-2:2006).

5.2.3 Within a security domain, all information objects exist at the same level of sensitivity. Members of a domain may have different security attributes, such as read, write, or execute permissions on information objects. Security domains are not bound by systems or networks of systems. A security domain's objects may reside in multiple systems.

5.2.4 A policy describes the legal framework including rules and regulations, the organizational and administrative framework, functionalities, claims and objectives, the principals involved, agreements, rights, duties, and penalties defined, as well as the technological solution implemented for collecting, recording, processing, and communicating data in information systems. For describing policies, methods such as policy templates or formal policy modeling might be deployed (**4**). For example, W3C WS-Policy provides a general purpose model and syntax to describe and communicate the policies of a web service. It specifies a set of common message policy assertions within a policy and attachment mechanisms for using policy expressions with existing XML service technologies.

5.2.5 In the domain model, a domain may consist of subdomains or participation in superdomains. Subdomains will inherit policies from their parent domain. The domain may be extended by chaining subdomains to superdomains, forming a common domain of communication and cooperation that is characterized by establishing an agreed upon security policy. Such transaction-concrete policy has to be negotiated between the communicating and cooperating principals, which is also called policy bridging.

5.2.6 Users may perceive a collection of objects from different security domains as a single object. This compound object is referred to as a multidomain information object. In compound security domains, additional policies shall be written that apply to the newly created multidomain information objects. The multidomain information security policy shall state the privileges that a user must have to view, print, create, delete, or transfer multidomain information objects between information systems. It cannot be assumed that the compound domain policies are simply inherited from the subdomains.

5.2.7 The general purpose of communication is the provision of services to a client requesting these services. Most of the services have to be provided by the functionality of the healthcare information system often combined with human users' interactions. Such application services are end-system services.

5.2.8 Middleware concepts are being increasingly introduced into newer versions of healthcare information systems. In this model, not only both principals, but also the middleware, can provide requested functionality and application security services. Such an architecture can be represented by chains of different domains.

5.2.9 From the security point of view, a domain ensuring intradomain communication according to its own policy is commonly considered to need protection only at its boundary to external domains, with their specific policies (or even the policy-free domain of the internet). This is done by firewalls, proxy servers, and so forth. Regarding the external environment, a domain is therefore often considered closed. The internal domain is mistakenly assumed to be secure, often neglecting internal threats and attacks that are the majority among all security attacks (NIST Special Publication 800-33).

5.2.10 Regarding the specific requirements and conditions of healthcare, the underlying security model shall consider the whole spectrum of security services and mechanisms that can be accomplished by secure microdomains.

5.2.11 *Interaction between Security Domains*—Separate security domains in the domain model can exchange privilege information by agreement of the parties. This interaction between security domains shall be coordinated on both a technical and documentary level. The creating and exchange of privilege sets should take into consideration organizational structure.

5.2.11.1 *Technical Basis*—Exchanges of privilege information shall be examined to ensure the meaning of privileges is consistent between security domains. This can be accomplished by creating a standard set of privileges. The standard set of privileges may include a mutually defined mapping of equivalent privileges between the domains. The equivalence of the exchanged privileges shall be reviewed on a technical basis to ensure the intended security implications are achieved.

Documents
Document Sharing Hub

Full document details: ASTM E2595-07(2013)

5.2.11.2 *Administrative Basis:*

(1) Privilege information exchanged between security domains may involve separate administrative entities (for example, distinct business partners or companies). An agreement as to the exchange of privileges and their use shall be documented, typically in a "business partner agreement." The use of a business partner agreement is required to distinguish the legal, ethical, and practical responsibilities between business partners and that may extend between other cooperating PMI implementations. An equivalent procedure is performed in a PKI through the use of a certificate practices statement and certificate policies. An alternative procedure uses "policy assertions" in WS-Policy.

(2) Multiple security domains may exist within a single company or organization. An agreement documenting responsibilities between such domains should also be set forth in a business partner agreement or a memorandum of understanding (MOU). The document should be periodically reviewed to ensure privileges extended across security domains exist only as long as required to meet the needs of the enterprise.

5.2.11.3 *Organizational Considerations*—Privilege information exchanged between security domains should be structured to reflect organizational considerations. Establishing a security domain that encompasses an organizational objective (for example, accounting or human resources) provides a coherent approach. The resulting standard set of privileges suitable for interdomain exchange is, as a result, highly cohesive. That is, the privilege set provides privileges to a subset of the organization (for example, accounting) without extending privileges required in an unrelated subset of the organization (for example, human resources). In addition, a cohesive privilege set provides all privileges that are required to meet a specific objective.

5.3 *Control Model:*

5.3.1 The control model illustrates how control is exerted over access to operations on protected objects. There are five components in the model: claimant, verifier, target, access control information, and access control policy. The claimant has privilege attributes assigned by enterprise domain authorities (contained in various forms such as an attribute certificate, SAML assertion, and so forth), pushed or pulled to the verifier (at run-time or through provisioning). The target gets various other access control information (for example, sensitivity attributes, environmental variables) that may be contained in a security label (separating policy between different information domains), attribute certificate, or in a local database. The model described here enables the verifier, who may be the owner of the target or an independent authority, to control access to the target by the claimant, in accordance with the control policy and other access control conditions (see Fig. 1).

5.3.2 To control access, the verifier must be able to do two things. First, it must be able to get all applicable control policies for the target. Then it shall compare these with the privileges known for the claimant. Access is granted if and only if the privileges owned by the claimant dominate the control and sensitivity policies of the target consistent with all other applicable access control information. In practice, the verifier consists of access control decision functions and access control enforcement functions that may also be combined or separated. If separated, the claimant's privileges may be available to the verifier in a wide variety of ways depending upon specific strategies.

5.3.3 The generality of this model makes the names of its parts appear somewhat abstract; however, with suitable interpretation, it can be applied to all the situations introduced in this guide.

5.3.4 *Use of Push or Pull:*

5.3.4.1 The control model uses a verifier that acquires access control information to make an access control decision. The claimant can provide the information (for example, in a token) along with the request to the verifier (push) or the verifier can get the required information from a trusted source (pull). In deciding whether to use a push or pull model, several factors should be considered:

(1) Push:

(a) Tokens should have a short time to live,

(b) Tokens shall be validated against an authentication service,

(c) Token delivery should be encrypted, and

(d) Tokens should include a nonce or a unique key within the encrypted token.

(2) Pull:

(a) Authentication services shall be accessible,

(b) Authentication service ACI repository shall be accessible,

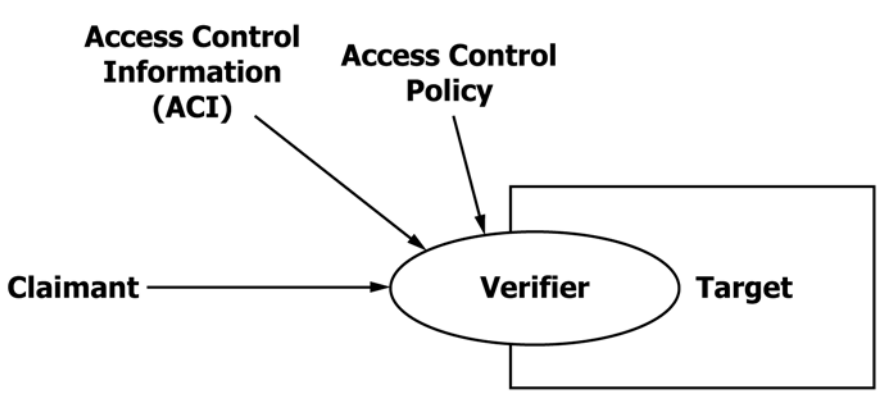

FIG. 1 Control Model

Documents
Document Sharing Hub
Full document details: ASTM E2595-07(2013)

(c) A trusted communication path to the ACI repository is required, and

(d) Authentication information should be current with ACI.

5.3.4.2 Tokens holding authentication information shall be sufficiently secure for the environment to guard against the possibility of replay attacks.

5.4 *Delegation Model:*

5.4.1 In addition to the control model, there is a need for a delegation model. There are three components of the delegation model: the verifier, the source of authority (SoA), and the claimant (see Fig. 2).

5.4.2 The verifier endows an entity known as the SoA with unlimited privilege. The SoA assigns privilege to claimants by issuing attribute credentials (for example, attribute certificates or SAML attribute assertions). The claimant asserts its delegated privilege by demonstrating its identity. This can be done by any mechanism consistent with the domain security policy, for example, by a password/identification or, increasingly, by proving knowledge of a private key whose public counterpart is contained in a public key certificate referenced by the claimed privilege. In a security distributed/service-oriented architecture, user authentication to the verifier would be provided by an authentication service directly.

5.4.3 Since an authentication service can also provide single sign-on capability, there shall be a relationship to the authorization service to ensure that claimant privileges are available to the verifier at the time when the claimant access request is made. In the case of an executable object, it may alternatively be done by demonstrating that the digest is the same as the "owner" value of an attribute certificate which includes the claimed privilege.

5.4.4 Optionally, the claimant may delegate its privilege to another claimant (delegate).[11] The verifier shall confirm that all entities in the delegation path possess sufficient privilege to access the target requested by the direct claimant. The verifier should determine that claimants possess the level of access required to access the target. Restrictions on the delegation may be established by the SoA, claimant, or by the target and include:

5.4.4.1 *Delegation Level*—For example, the claimant in delegate role may not be allowed to further delegate privilege.

5.4.4.2 *Delegation Context*—For example, delegate my "assigned-radiologist" privilege only for a given patient identity and only for a given set of X-ray images and only for a specified period of time.

5.4.4.3 *Delegate Set*—For example, no restrictions on number of levels of delegation, but all delegates shall be from a specified set of claimants.

5.4.4.4 *Reference Restriction*—The rights to use an object under specified circumstances are passed as part of the object reference to the recipient. For example, in privilege delegation, the initiating principal's access control information (that is, its security attributes) may be delegated to further objects in the chain to give the recipient the rights to act on the initiating principal's behalf under specified circumstances.

5.4.4.5 *Improper Delegation*—Restrictions that prevent a delegate from assigning rights to inappropriate delegates, for example, a clinician assigning rights to order medications to administrative staff.

5.5 *Document Model:*

5.5.1 Processes, entity roles, and so forth shall be documented and signed expressing the particular relations between entities and processes. The combination of processes and relations leads to multiple signatures (for example, in the case of delegation).

5.5.2 The document model shall support multiple document signatures (MDS). Cryptographic message syntax supports multiple signatures on a document. Each signature is computed over the document content and optionally a set of signed attributes specific to the particular signature. These attributes may include timestamps, signature purpose, and other information.

5.6 *Policy Model:*

5.6.1 A security policy is the complex of legal, ethical, social, organizational, psychological, functional, and technical implications for assuring trustworthiness of health information

[11] Distinguish this model based on discretionary access control from "delegated administration." For example, a physician may be granted some record access privilege by a hospital. The physician may assign the office manager responsibility to extend (delegate) that privilege to various roles which do not include the office manager. In this circumstance, there is an entity in the delegation path that does not possess access privilege. This delegation administration model implemented in some Personal Health Records (PHR) implementations (cf. iMetrikus).

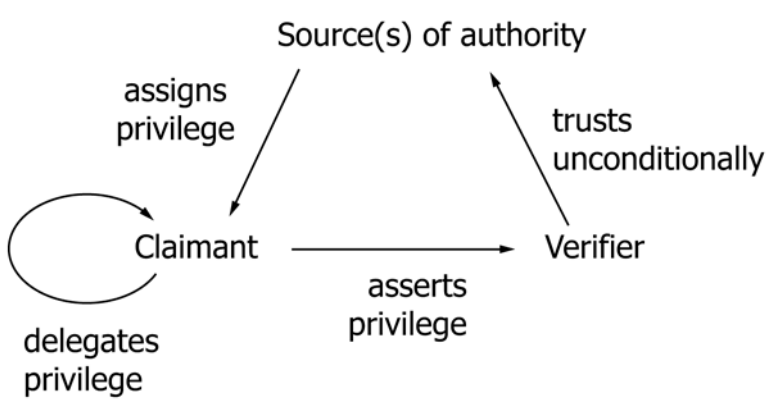

FIG. 2 Delegation Model

Documents
Document Sharing Hub
Full document details: ASTM E2595-07(2013)

systems. A policy is the formulation of the concept of requirements and conditions for trustworthy creation, storage, processing and use of sensitive information. A policy can be expressed:

5.6.1.1 Verbally unstructured,

5.6.1.2 Structured using schemata or templates, or

5.6.1.3 Formally modeled, for example, using Unified Modeling Language (UML).

5.6.2 For interoperability reasons, a policy shall be formulated and encoded in a way that enables its correct interpretation and practice (**5**). Therefore, policies have to be constrained regarding syntax, semantics, vocabulary, and operation of policy documents, also called policy statements or policy agreements (agreements between the partners involved).

5.6.3 *Policy Expression*—Several approaches to expression of policy in a formal manner have been developed. Some of these are presented here.

5.6.3.1 *XML Schema:*

(1) Fig. 3 presents a simple generic XML instance for a security policy statement. One common way to express constraints is the specification of a defined XML schema. This schema should be standardized for interoperability purposes mentioned in the previous section.[12]

(2) To refer to a specific policy reliably, the policy instance shall be uniquely identified. The same is true for all the policy components such as domain, targets, operations, and their policies, which have to be named and uniquely identified too. As with any other component, policy components can be composed or decomposed according to the generic component model. A policy is therefore characterized by a policy identifier, a policy name, a policy authority, a domain identifier, a domain name, a target list, a target identifier, a target name, and a target object with its operations allowed and related policies. This guide assumes the use of XACML for expressing XML security policies.

5.6.3.2 *Ponder-Language Base-Class:*

(1) The policy class can be specialized as basic policy, meta policy, and composite policy classes, as shown in Fig. 4.

(2) The specializations of the composite policy abstract class are interrelated in a complex way, which has been indicated in outlines as simple association.

5.6.3.3 *Object Management Group Security Services:*

(1) Another approach to policy decomposition has been provided by the OMG's Security Services Specification distinguishing between the following policies:

(a) Invocation access policy implementing access control policy for objects,

(b) Invocation audit policy controlling event type and criteria for audit, and

(c) Secure invocation policy specifying security policies associated with security associations and message protection.

(2) Policies regarding requirements for different object types include:

(a) Invocation delegation policy,

(b) Application access policy,

(c) Application audit policy, and

(d) Non-repudiation policy.

(3) Health information systems such as the EHR should at minimum have a Patient Policy, an Enterprise Policy, policies defined by laws and regulations, and one policy per Role.

(4) Every creation, access, or modification to an EHR component must be covered by one or more policies. The reference model of the EHR Extract includes a Policy ID attribute within the Record Component class to permit references to such policies to be made at any level of granularity within the EHR hierarchy. The policies that apply specifically to an EHR may be included within the EHR Extract, eventually including any bridged policies, as discussed in the domain model above.

5.7 *Role Model:*

5.7.1 Roles can be defined for managing relationships between principals (claimants) and objects. Roles are collections of permissions that allow principals to perform operations on protected information objects associated with work profile

[12] OASIS provides standardized schemata for implementing security policy in its eXtensible Access Control Markup Language (XACML) specification.

```
<policy>
        <policy_name/>
        <policy_identifier/>
        <policy_authority/>
        <domain_name/>
        <domain_identifier/>
        <target_list>
                <target_name/>
                <target_ID/>
                <target_object>
                        <operations/>
                        <policies/>
                </target_object>
        </target_list>
</policy>
```

FIG. 3 Policy Template Example

Documents
Document Sharing Hub
Full document details: ASTM E2595-07(2013)