HL7 Role-Based Access Control (RBAC) Role Engineering Process

Version 1.3

HL7 Security Technical Committee

September 2007

Review History

Date	Version	Description	By
09/17/2007	1.2	Last reviews completed July 2007 w/Out-of-Cycle Ballot, November 2005	Suzanne Gonzales-Webb
09/19/2007	1.3	QA review / revision	Craig M. Winter

Table of Contents

Section **Page**

List of Figures

Figure **Page**

List of Tables

Table **Page**

1 Introduction

This scenario-based role engineering process has been adopted by the HL7 Security Technical Committee (TC) as of January 2005 for the purpose of defining healthcare-specific permission standards. Based largely upon earlier work by Gustaf Neumann and Mark Strembeck [Neumann/Strembeck], HL7 adopted engineering and role definition content models compliant with those of the ANSI RBAC standard [ANSI-RBAC].

The *fundamental objective of the role engineering approach is to create a set of standard re-useable permissions*. In this way, HL7 will create a set of common building blocks capable of being assigned to an arbitrary number of functional roles to meet the needs of the healthcare community.

Currently, there is widespread international interest in Role-Based Access Control (RBAC) as one form of policy-based access control. RBAC standards and numerous related papers from organizations such as the National Institute of Standards and Technologies (NIST), Association of Computing Machinery (ACM), Organization for the Advancement of Structured Information Standards (OASIS), and the International Organization for Standardization (ISO) are establishing fundamental principals for RBAC implementations. At the same time, efforts to develop essential domain-specific role content have not kept pace with the growth of standards.

HL7 has taken the lead in promoting role engineering processes supporting standard permission definitions for healthcare organizations. HL7 role-engineering efforts began in May 2004 with Board approval of a proposal to add permission definitions to the HL7 set of standards. In January 2005 the HL7 Security Technical Committee was approved with the responsibility for developing and maintaining the HL7 Permission Catalog and other related artifacts.

2 Motivation

Should this person or a person who performs this job function typically be allowed to access this type of data?

RBAC is a method to control access to resources on an information system. It was developed to overcome the complexities of managing individual user permissions and their assignments. The HL7 effort is motivated by concurrent efforts to:

- Simplify authorization management,
- Reduce administrative costs,
- Improve security,
- Enhance partner interoperability,
- Enable new network-level RBAC services, and
- Improve service to members/clients/patients.

3 Role Engineering Models

HL7 will use role engineering models to assist it in carrying out its activities. Models illustrate relationships between components of an abstract role system to the components of the role engineering process. The discussion below defines the various HL7 role engineering models. While HL7 is not defining roles, this section describes the application of HL7 permissions to them.

3.1 Role Types

There are two types of high-level healthcare role models: Functional and Structural.

> ***Functional Roles*** consist of all the permissions (operations on health information system objects) needed to perform a task. Functional role names are associated with groups of permissions for convenience in assigning to users. A user may be assigned one or more functional roles, and thereby be assigned all of the permissions associated with a corresponding set of healthcare tasks (healthcare workflow). Permissions will ultimately be used to set the system operations (create, read, update, delete, execute, etc.) for data and software applications. *Functional roles may be found as entries in a user attribute certificate or stored in a distributed authorization directory*.

Figure 1 illustrates the User, Functional Role, Permission, and Operation and Object relationships. This figure is an adaptation of the ANSI RBAC Core RBAC reference model. It does not include the concept of sessions/session roles, which is not part of the HL7 role engineering process.

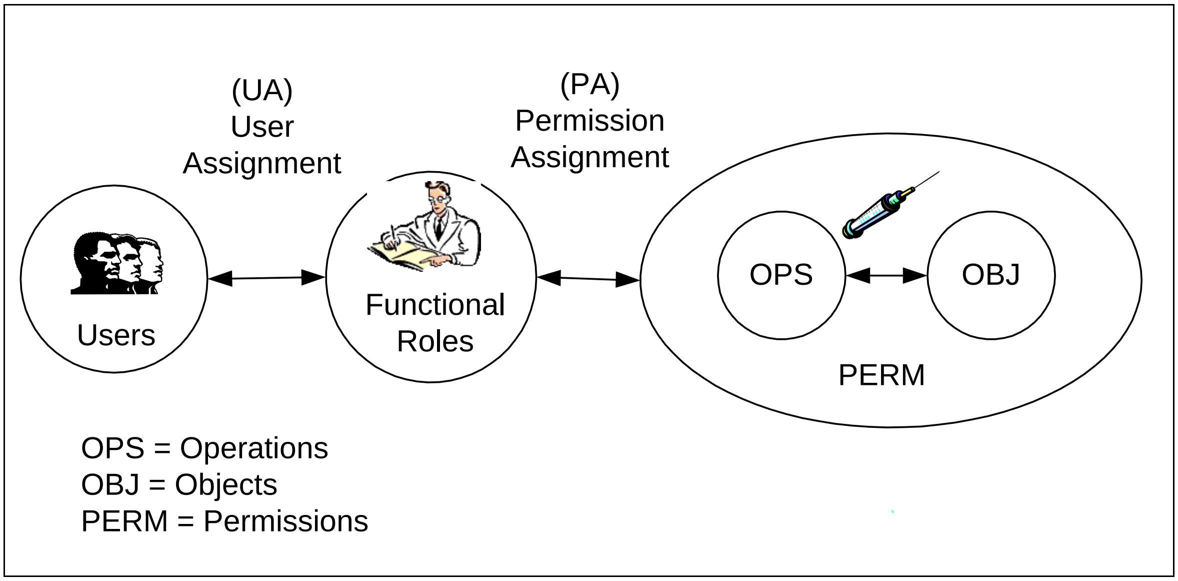


Figure 1: Role Structure
(Adapted from ANSI RBAC Core RBAC Model)

> ***Structural Roles*** place people in the organizational hierarchy as belonging to categories of healthcare personnel warranting differing levels of access control.[1] Structural roles allow users to participate in the organization's workflow (e.g., tasks) by job, title, or position, but do not specify detailed permissions on specific information objects. Structural roles allow a user to "connect" to a resource, but do not grant authorizations. Some structural role examples include: Physician, Pharmacist, Registered Nurse Supervisor, and Ward Clerk. *Structural role names may be found as non-critical certificate extensions entries to an X.509 certificate as specified in ASTM 2212-00.*

Structural roles define what specific healthcare workflows users are allowed to participate in while functional roles define authorizations granted to entities to allow access to protected health information.

ANSI-RBAC describes two other models: Hierarchical RBAC and Constrained RBAC. Although the role engineering process described herein recognizes these models, HL7 does not plan to use them in its initial phase.

[1] See ASTM E1986-98 for a listing of healthcare personnel that warrant differing levels of access control.

3.2 Work Profiles, Tasks, Scenarios, Steps, and Permissions

The Scenario Model, as shown in Figure 2, illustrates the hierarchy of work profiles, tasks, scenario, and steps. Permissions are defined relative to steps (described in the role engineering process to follow).

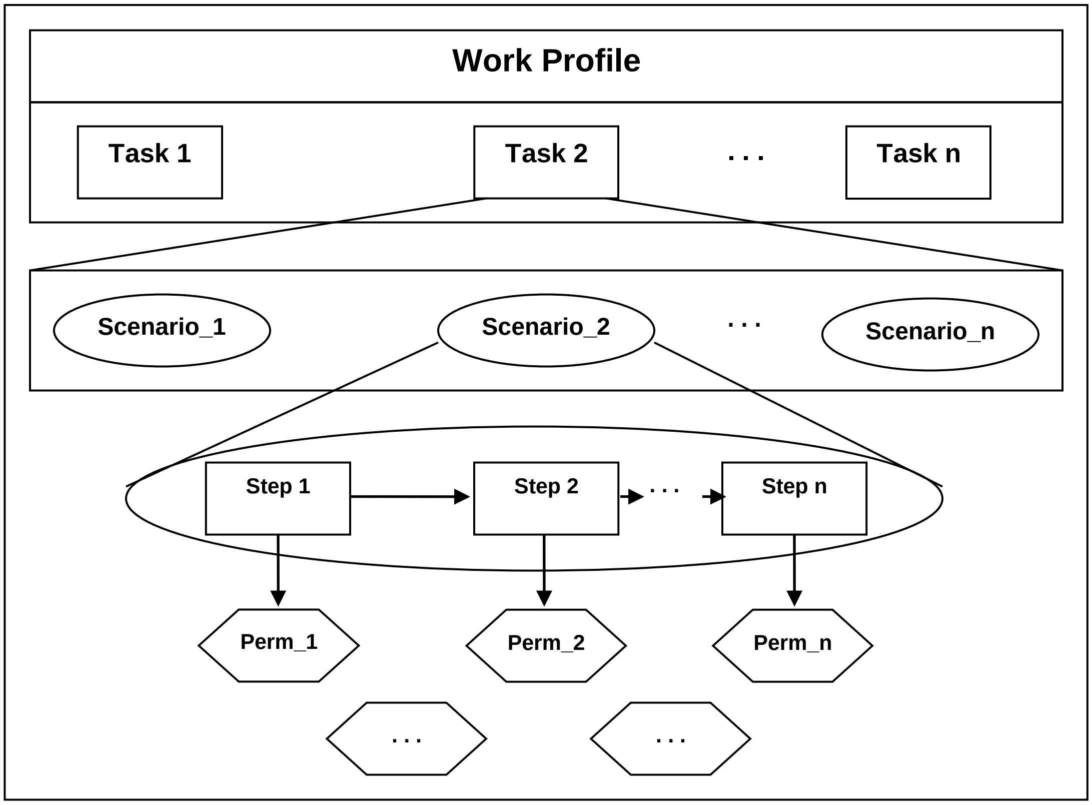


Figure 2: Scenario Model
[Neumann/Strembeck]

In the scenario-based role-engineering approach, each action and event within a scenario can be seen as a step that is associated with a particular access operation. Scenarios, which are applied in a particular order to reach a predefined task goal, act as sources for the derivation of permissions. The user performing a scenario must own all permissions that are needed to complete every step of the scenario.

Permissions need to be defined at an appropriate level to be useful "building blocks." Their definition should not be too low or too high in the hierarchy. If permissions are defined too low, (e.g., at the data element or table level), then we are effectively defining a universal healthcare schema to which all organizations must comply. Table 1 below shows the various levels in the hierarchy, and highlights the "aggregate" level, which is the lowest hierarchical level for permission interoperability. The aggregate level describes a "conceptual" healthcare object or function (e.g., a prescription) without reference to its structural components.

Table 1: Permissions

Action	Role Engineering Object	Role
Participate	Work Profile	Structural Role
Execute	Task	
Execute	Scenario	
Perform	Step	
Create, Read, Update, Delete[2]	Aggregation	Functional Role
Execute	Function	Functional Role
Create, Read, Update, Delete	Data Table	
Create, Read, Update, Delete	Data Element	

System owners, architects, and vendors are all able to describe such objects more concretely in reference to their own more proprietary objects. Because aggregates are viewed as the objects subordinate to steps of the role-engineering process, they will also logically be defined by this process.

The HL7 role engineering process creates abstract healthcare work profiles, permissions, and information model components (see Table 1 shaded rows). Using the model, HL7 members identify or create work profiles or tasks (e.g., HL7 storyboards=scenarios, etc.) to derive abstract permissions. These permissions further define operations on HL7 information objects.

[2] Many healthcare organizations do not "delete" objects, but instead add a new object that replaces the older one. In this case, "deleted" may be effectively implemented as "addend."

4 Role Engineering Process

The fundamental objective of the role engineering approach is to create a set of standard re-useable permissions.

4.1 Assumptions

[Neumann/Strembeck] provides a basis for defining roles using scenarios. Within this context, the following clarifications are made:

- Tasks reflect an organization's job functions and can be used to deduce permissions,
- Structural roles determine a user's authorization to participate in specific workflows and to connect to specific protected resources,
- Permissions determine what operations a user is permitted on health information system protected resources,
- Permissions are assigned to functional roles (currently out of scope of HL7 permission process), and
- Standard functional roles consisting of grouped standard permissions are defined to support inter-domain data transfer (currently out of scope of HL7 permission process).

4.2 High-level View of the Role Engineering Process

4.2.1 Model Interrelations

Developing the usage scenarios (i.e., the Scenario Model) is the first step in the process.[3] Figure 3 illustrates how the remaining components of the role engineering process are related to the Scenario Model.

[3] Establishing functional Work Groups prior to developing usage scenarios will provide a mechanism for grouping and categorizing scenarios.

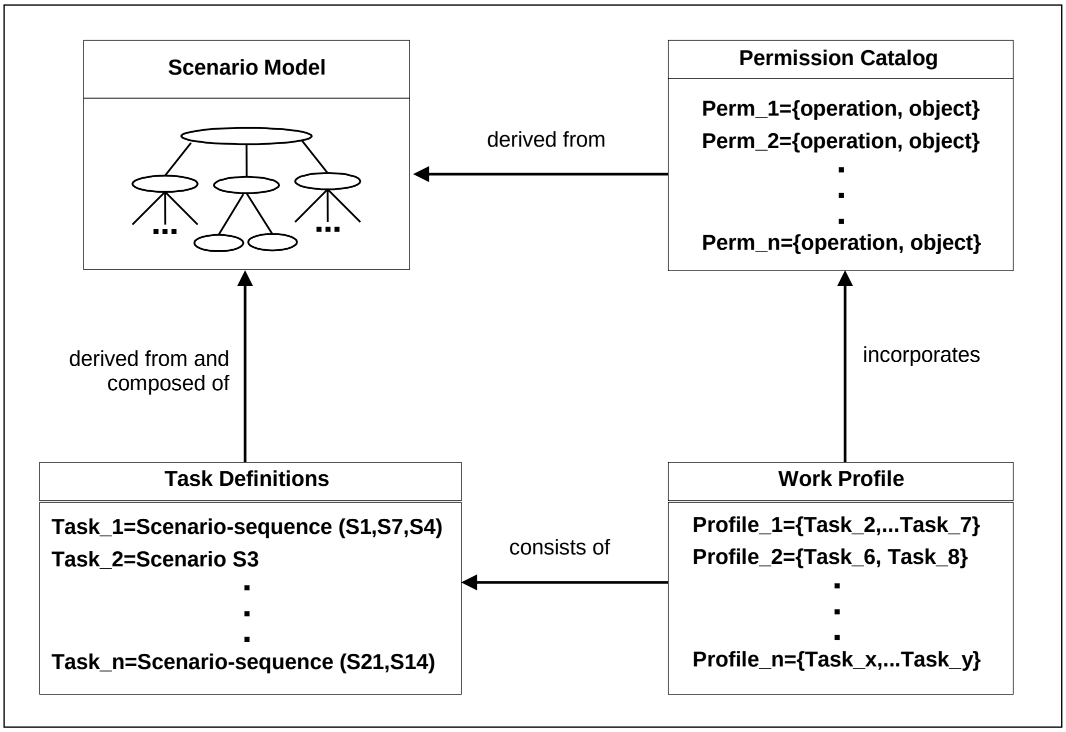
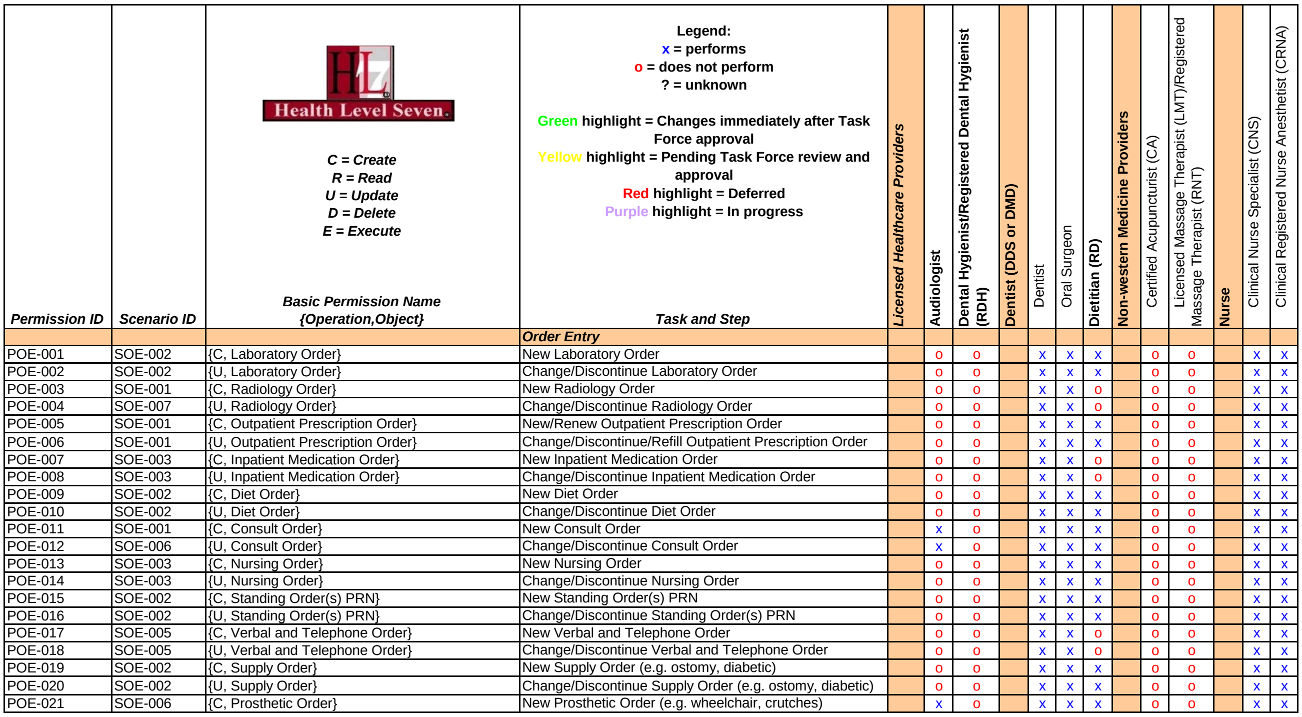

Scenario Model
Permission Catalog
Perm_1={operation, object}
Perm_2={operation, object}
.
.
.
Perm_n={operation, object}
derived from
derived from and composed of
incorporates
Task Definitions
Task_1=Scenario-sequence (S1,S7,S4)
Task_2=Scenario S3
.
.
.
Task_n=Scenario-sequence (S21,S14)
consists of
Work Profile
Profile_1={Task_2,...Task_7}
Profile_2={Task_6, Task_8}
.
.
.
Profile_n={Task_x,...Task_y}

Figure 3: Interrelations of Scenario Model and Documents
(Adapted from [Neumann/Strembeck])

4.2.2 Healthcare Scenario Roadmap

The effort of identifying work tasks and profiles for all healthcare personnel (licensed, non-licensed, and non-caregiver) is daunting. To scope and manage the effort, the development of a spreadsheet called the "Healthcare Scenario Roadmap" provides a useful tool.

Figure 4 illustrates an example of a Healthcare Scenario Roadmap that contains a list of high level scenarios mapped to [ASTM 1986] structural healthcare roles. Licensed and non-licensed structural healthcare roles are derived from [ASTM 1986], the *Standard Guide for Information Access Privileges to Health Information*. The list of healthcare system work tasks is derived from documentation, system access patterns, user interaction, and input from healthcare professionals and domain experts.



Permission ID	Scenario ID	C = Create R = Read U = Update D = Delete E = Execute Basic Permission Name {Operation,Object}	Legend: x = performs o = does not perform ? = unknown Green highlight = Changes immediately after Task Force approval Yellow highlight = Pending Task Force review and approval Red highlight = Deferred Purple highlight = In progress Task and Step	Licensed Healthcare Providers	Audiologist	Dental Hygienist/Registered Dental Hygienist (RDH)	Dentist (DDS or DMD)	Dentist	Oral Surgeon	Dietitian (RD)	Non-western Medicine Providers	Certified Acupuncturist (CA)	Licensed Massage Therapist (LMT)/Registered Massage Therapist (RNT)	Nurse	Clinical Nurse Specialist (CNS)	Clinical Registered Nurse Anesthetist (CRNA)
			Order Entry													
POE-001	SOE-002	{C, Laboratory Order}	New Laboratory Order		o	o		x	x	x		o	o		x	x
POE-002	SOE-002	{U, Laboratory Order}	Change/Discontinue Laboratory Order		o	o		x	x	x		o	o		x	x
POE-003	SOE-001	{C, Radiology Order}	New Radiology Order		o	o		x	x	o		o	o		x	x
POE-004	SOE-007	{U, Radiology Order}	Change/Discontinue Radiology Order		o	o		x	x	o		o	o		x	x
POE-005	SOE-001	{C, Outpatient Prescription Order}	New/Renew Outpatient Prescription Order		o	o		x	x	x		o	o		x	x
POE-006	SOE-001	{U, Outpatient Prescription Order}	Change/Discontinue/Refill Outpatient Prescription Order		o	o		x	x	x		o	o		x	x
POE-007	SOE-003	{C, Inpatient Medication Order}	New Inpatient Medication Order		o	o		x	x	o		o	o		x	x
POE-008	SOE-003	{U, Inpatient Medication Order}	Change/Discontinue Inpatient Medication Order		o	o		x	x	o		o	o		x	x
POE-009	SOE-002	{C, Diet Order}	New Diet Order		o	o		x	x	x		o	o		x	x
POE-010	SOE-002	{U, Diet Order}	Change/Discontinue Diet Order		o	o		x	x	x		o	o		x	x
POE-011	SOE-001	{C, Consult Order}	New Consult Order		x	o		x	x	x		o	o		x	x
POE-012	SOE-006	{U, Consult Order}	Change/Discontinue Consult Order		x	o		x	x	x		o	o		x	x
POE-013	SOE-003	{C, Nursing Order}	New Nursing Order		o	o		x	x	x		o	o		x	x
POE-014	SOE-003	{U, Nursing Order}	Change/Discontinue Nursing Order		o	o		x	x	x		o	o		x	x
POE-015	SOE-002	{C, Standing Order(s) PRN}	New Standing Order(s) PRN		o	o		x	x	x		o	o		x	x
POE-016	SOE-002	{U, Standing Order(s) PRN}	Change/Discontinue Standing Order(s) PRN		o	o		x	x	x		o	o		x	x
POE-017	SOE-005	{C, Verbal and Telephone Order}	New Verbal and Telephone Order		o	o		x	x	o		o	o		x	x
POE-018	SOE-005	{U, Verbal and Telephone Order}	Change/Discontinue Verbal and Telephone Order		o	o		x	x	o		o	o		x	x
POE-019	SOE-002	{C, Supply Order}	New Supply Order (e.g. ostomy, diabetic)		o	o		x	x	x		o	o		x	x
POE-020	SOE-002	{U, Supply Order}	Change/Discontinue Supply Order (e.g. ostomy, diabetic)		o	o		x	x	x		o	o		x	x
POE-021	SOE-006	{C, Prosthetic Order}	New Prosthetic Order (e.g. wheelchair, crutches)		x	o		x	x	x		o	o		x	x

Figure 4: Healthcare Scenario Roadmap Example

4.2.2.1 Roadmap Columns

The HL7 Role Engineering Process has specified the existence of two distinct role types: structural and functional roles.

Structural roles can be viewed as precursor roles that give a person access to a "session" or "connection." Structural roles allow a user possessing that role to participate in a work profile. HL7 has chosen to use a set of data elements found in the Standard Guide for Information Access Privileges to Health Information, "ASTM E 1986 – 98 Healthcare Personnel that Warrant Differing Levels of Access Control" for the definition of structural role names used in the roadmap to illustrate the breakdown or partitioning of health information into data sets that warrant differing levels of access. The Table list is comprised of data elements and entities that may, but are not required to be collected, utilized, stored, or maintained, or a combination thereof, in the process of providing healthcare administrative and clinical services.

The entities placed in the column headings of the roadmap are structural role names.

Functional roles reflect the essential business functions that need to be performed. They are closely related to Work Profiles in the Scenario model. Functional roles define what an actor can do once connected to a protected resource. The roadmap does not define functional roles; however, analysis of the roadmap-derived scenarios will lead to defined permissions that can be used to create functional roles.

4.2.2.2 Roadmap Rows

The aggregate activities in the rows of the Roadmap have been suggested by knowledgeable healthcare personnel possessing the structural roles listed in the columns of the roadmap. The elements in the rows reflect aggregate activities performed by the corresponding basic role. An “x” in the intersection means that the activity is performed; an “o” means that it is not.

The combination of all activities for which an “x” exists for a given structural role defines a work profile for that role.

The highlighted horizontal activities of the roadmap are tasks.

Note that if the subordinate activities can be described by a single scenario by a single actor, then the highlighted horizontal tasks collapse to a scenario and the subordinate activities become steps. Role engineering efforts focus on writing scenarios for defined roadmap tasks.

The subordinate task activities represent, in part or as a whole, components of scenarios performed by a single actor.

The “x” activities of the roadmap represent “least privilege” abstract permissions for the corresponding objects of the rows.

4.3 Process Steps [Neumann/Strembeck]

This section describes the detailed HL7 role engineering process steps used to create standards-compliant permissions.

The HL7 process described here is adapted from and closely follows sections of Neumann and Strembeck’s *A Scenario-driven Role Engineering Process for Functional RBAC Roles*, June 2002. The process itself is not healthcare specific, but HL7 permissions are because they are derived from healthcare domain knowledge applied to the general process. The major difference is the adaptation of the [Neumann/Strembeck] process to define “permissions” and not “roles.” This distinction of the goal of the HL7 RBAC Role Engineering Process replaces [Neumann/Strembeck] where applicable and as indicated. Where an apparent conflict exists, the *HL7 RBAC Role Engineering Process* shall be authoritative.

4.3.1 Identify and Model Usage Scenarios [Neumann/Strembeck 4.1]

4.3.1.1 Preliminary

Scenarios are the key to a systematic engineering approach leading to the development of permissions on protected health information objects. HL7 defined scenario names and appropriate methodologies are used to document the scenario steps. Initial scenarios are suggested from HL7 storyboards, Electronic Health Record (EHR) scenarios, healthcare professionals, and existing systems.

4.3.1.2 Detailed Process

Identify sensible usage scenarios.

Start each candidate scenario by writing down a short sentence, for example, “Create a new patient record.” Give each scenario a unique name, and follow up by adding detailed text and diagrams. Involve assistance from healthcare domain experts like physicians, nurses, and hospital clerks. Finally, record the consolidated list of scenarios; this is the scenario model.

STEP 1 ➡ Gather an initial list of healthcare scenarios using HL7 storyboards and actual system access patterns.

STEP 2 ➡ Assign each scenario a name using the HL7 nomenclature. Create structured text (steps) and a sequence diagram for each scenario.

STEP 3 ➡ Validate and complete scenarios with input from healthcare domain experts.

STEP 4 ➡ Record consolidated list of scenarios.

4.3.2 Permission Derivation from Scenarios [Neumann/Strembeck 4.2]

4.3.2.1 Preliminary

HL7 permissions exhibit specific attributes as follows:

- Provide access rights to protected information (least privilege, need-to-know, or separation of duties),
- Describe security-relevant events, and
- Contain at least one operation on one object.

4.3.2.2 Detailed Process

Identify scenario permissions and populate the permission catalog. Review each scenario step and decide which operation needs to be performed on which object to complete the step. For each of these {operation, object} pairs, create a record in the permission catalog.

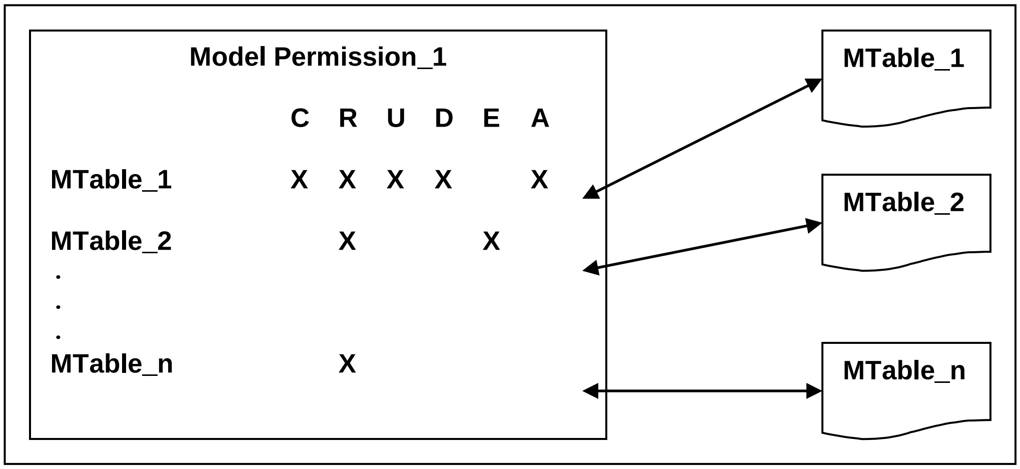
STEP 1 ➡ Review scenario and identify the actors and steps in the scenario.

STEP 2 ➡ Identify the operations and objects required to perform each step.[4]

STEP 3 ➡ For each scenario step, record the associated {operation, object} pairs.

Figure 5 depicts a model permission catalog.

[4] This process may suggest clarifications needed to the scenario. If so, make the update and catalog the results.



Model Permission_1

	C	R	U	D	E	A
MTable_1	X	X	X	X		X
MTable_2		X			X	
.						
.						
.						
MTable_n		X				

Figure 5: Model Permission Catalog

4.3.3 Identification of Permission Constraints [Neumann/Strembeck 4.3]

Constraints are not currently used to create the HL7 permission catalog. At this time, constraints are considered to be local rules affecting roles not permissions. Constraints that are inter-organizational in scope will be added to this process at a future date. This section is a place-holder for possible future work and is described here for completeness.

Constraints are restrictions that are enforced upon access permissions. They can include contextual properties such as separation of duties, time-dependency, mutual exclusivity, cardinality, or location, etc. Constraints may be enterprise specific.

Examples of permission constraints could include:

- Head Nurse permission functions can be accessed only by one Registered Nurse per 12-hour shift on a hospital floor at any given time (cardinality of 1, time-dependency),
- Only one Physician may have access to the Chief of Staff permissions (cardinality of 1),
- A laboratory user can co-sign another Lab Technician's results, but cannot co-sign their own even if logged on as the Lab Technician Supervisor (separation of duties),
- Provider's access to a remote hospital that is not his/her primary workplace (location), and
- A physician working scheduled clinic hours (time-dependency) vs. physician working in a 24 hour Emergency Room (no time-dependency).

4.3.4 Scenario Model Refinement [Neumann/Strembeck 4.4]

Review and refine the initial scenario model using two essential activities:

Concretion. Each step within each scenario is reviewed to see if it is complex enough to be described in more detail through its own sub-scenario.

STEP 1 ➡ For each complex scenario, define sub-scenarios, as necessary.

STEP 2 ➡ Update the scenario model.

Generalization. Review the current scenario model.

STEP 1 ➡ Search the scenario model for similar {operation, object} pairs.

STEP 2 ➡ Consolidate the list of similar steps and {operation, object} pairs, eliminating duplicates.

STEP 3 ➡ Define an abstract type for the scenario, if necessary.

STEP 4 ➡ Group the similar scenarios and derive a common abstract type.

4.3.5 Definition of Tasks and Work Profiles [Neumann/Strembeck 4.5]

4.3.5.1 Preliminary

HL7 is concerned with defining standard model permissions for the healthcare community. To accomplish this goal, scenarios must be developed and (optionally) grouped into tasks for analysis. This process does not presume the order of scenario and task definition.

- A task is a collection of scenarios that can be combined to perform a complex operation.
- A work profile consists of one or more tasks. Therefore, each work profile is a job description for a certain position within the organization under consideration.

The specifications for task and work profiles are defined with assistance from healthcare domain experts. The most challenging part is to select the correct group of scenarios for a particular task.

4.3.5.2 Detailed Process

Combine logically related scenarios together into tasks. Use the result to define work profiles.

STEP 1 ➡ Identify scenarios that logically belong together.

STEP 2 ➡ Group the scenarios into tasks.

STEP 3 ➡ For each entity and permission, record a corresponding “x” or “o” in the healthcare scenario roadmap.

4.3.6 Derivation of a Preliminary Role-hierarchy [Neumann/Strembeck 4.6] / RBAC Model Definition [Neumann/Strembeck 4.7]

Roles are not currently part of the HL7 permission catalog definition. At this time, roles are considered to be locally defined by organizations that build them using HL7 standard permissions. Roles that are inter-organizational in scope may be added to this process at a future date. This section is a place-holder for possible future work.

5 References

[ASTM 1986] American Society for Testing and Materials. *ASTM Standard E1986-98: Standard Guide for Information Access Privileges to Health Information*, 1998.

[Collins] A. Collins, T. Cooper, MD, G. Martin and E. Powers. *Role-based Access Control: A Sensible Approach.* Healthcare Information and Management Systems Society, 2000.

[HL7] L. Dailey-Evans. *HL7 Lab Frequency Order Storyboard.* Health Level 7 Orders Technical Committee, 2002.

[IETF] Marshall, *Security Audit and Access Accountability Message Data Definitions for Healthcare Applications*, Internet Draft, Internet Engineering Task Force, December 2003.

[Neumann/Strembeck] G. Neumann and M. Strembeck. *A Scenario-driven Role Engineering Process for Functional RBAC Roles*, June 2002.

[ANSI-RBAC]. Information Technology Industry Council. *American National Standard for Information Technology - Role-Based Access Control*, ANSI INCITS 359-2004, 2004.

[SAIC] SAIC Security Engineering. *Working Paper – Implementing a Role-Based Access Control Policy for CHCS II*, 1999.

[SAIC2] Staggs, David. *XACML in the VHA Development Environment Version 1.0*, Science Applications International Corporation Secure Business Solutions Group, March 2004.

[XACML] OASIS. *eXtensible Access Control Markup Language (XACML) Version 1.0.* OASIS Standard. 18 February 2003.

[XACML RBAC] OASIS. *XACML Profile for Role Based Access Control (RBAC) Committee Specification 01*, 13 February 2004.

6 Annex A - Terminology

6.1 Term Harmonization

Table 2 illustrates relationships among the ANSI RBAC standard, Neumann/Strembeck, OASIS, and the HL7 terminology. Wherever possible the ANSI RBAC term is preferred. HL7 terms harmonize ANSI RBAC and Neumann/Strembeck semantics into a single composite set.

Table 2: Term Harmonization

ANSI RBAC	Neumann/ Strembeck	OASIS	HL7
User	NA	NA	NA
NA	NA	Subject	Actor
Role	Functional Role	Attribute/Role	Functional Role
Role	Organizational Role	NA	Structural Role
Permission	Permission	Rule/Permission	Permission
Operation	Operation	Action	Operation
Object	Object	Resource	Object
NA	Work Profile	NA	Work Profile
NA	Task	NA	Task
NA	Scenario	NA	Scenario
NA	Step	NA	Step
Session Roles	NA	NA	NA

6.2 Definitions

Table 3 presents the terms that have been adopted for use within this document. The definition and source of each term are also listed.

Table 3: Definitions

Term	Definition	Source
Access	*Access* is performing an action.	[XACML]
Access Control	*Access control* is controlling access in accordance with a policy.	[XACML]
Action	An *action* is an operation on a resource.	[XACML]

Table 3: Definitions

Term	Definition	Source
Actor	An *actor* is defined as a healthcare worker involved in a step within a scenario. The actor and associated step are labeled in a sequence diagram. Actors are classes that define roles that objects external to a system may play. They are used to model users outside of a system that interact directly with the system as part of coherent work units. This includes person or human users and other systems.	UML
Functional Role	*Functional roles* reflect the essential business functions that need to be performed. *Functional roles* are defined by a set of standard healthcare tasks (e.g., Neurologist).	[Neumann/Strembeck]
Object	An *object* is an entity that contains or receives information. The *objects* can represent information containers (e.g., files or directories in an operating system; and/or columns, rows, tables, and views within a database management system), or *objects* can represent exhaustible system resources, such as printers, disk space, and CPU cycles. The set of *objects* covered by RBAC includes all of the objects listed in the permissions that are assigned to roles.	[ANSI-RBAC]
Operation	An *operation* is an executable image of a program, which upon invocation executes some function for the user. Within a file system, *operations* might include read, write, and execute. Within a database management system, *operations* might include insert, delete, append, and update. An *operation* is also known as a privilege.	[ANSI-RBAC]
Organizational Role	*Organizational roles* correspond to the hierarchical organization in a company in terms of internal structures.	[Neumann/Strembeck]
Permission	A *permission* is an approval to perform an operation on one or more RBAC protected objects.	[ANSI-RBAC]
Policy	A *policy* is a set of rules, an identifier for the rule-combining algorithm, and (optionally) a set of obligations. A *policy* may be a component of a policy set.	[XACML]

Table 3: Definitions

Term	Definition	Source
Resource	A *resource* can be data, service, or system component.	[XACML]
Role	A *role* is a 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.	[ANSI-RBAC]
Rule	A *rule* is the most elementary unit of a policy. A *rule* has a target, an effect, and a condition.	[XACML]
Scenario	A *scenario* is an example of system usage in the form of action and event sequences. *Scenarios* are recorded as UML sequence diagrams.	[Neumann/Strembeck]
Session Roles	A *session role* is the role activated by a user session.	[ANSI-RBAC]
Step	A *step* is an action or event within a scenario.	[Neumann/Strembeck]
Storyboard	A *storyboard* is an HL7 healthcare scenario. See *Scenario*.	[HL7]
Structural roles	*A structural role* is a type of healthcare personnel warranting differing levels of access control. Also known as "basic role," "organizational role," or "role group." For a listing of healthcare structural roles see ASTM E 1986-98 (e.g., Attending Physician).	Adapted from [ASTM 1986]
Subject	A *subject* is an actor whose attributes may be referenced by a predicate.	[XACML]
Task	A *task* is a collection of one or more scenarios.	[Neumann/Strembeck]
User	A *user* is defined as a human being, but can be extended to include machines, networks, or intelligent autonomous agents.	[ANSI-RBAC]
Work Profile	A *work profile* is a processing event that consists of all tasks performed by a user.	[Neumann/Strembeck]

6.3 Acronyms

Table 4 lists the acronyms used in this document.

Table 4: Acronyms

Acronym	Definition
ANSI	American National Standards Institute
ANSI	American National Standards Institute
ASTM	American Society for Testing and Materials
EHR	Electronic Health Record
HL7	Health Level Seven
INCITS	InterNational Committee for Information Technology Standards
ISO	International Organization for Standardization
ITEF	Internet Engineering Task Force
NA	Not Applicable
NIST	National Institute of Standards and Technologies
OASIS	Organization for the Advancement of Structured Information Standards
RBAC	Role-based Access Control
SAIC	Science Applications International Corporation
TC	Technical Committee
VHA	Veterans Health Administration
XACML	eXtensible Access Control Markup Language

7 Attachment 1 – Applied Example

Refer to the document, *HL7 Role-based Access Control (RBAC) Role Engineering Process - Applied Example*, to see how this HL7 RBAC Role Engineering Process can be applied to sample scenarios.