# The Harvard Integrated Management System - Project Integer[1] Authentication and Authorization Model

| Version | Date | Description |
|---|---|---|
| 1.0 | 11/10/2013 | Initial version. |
| 1.1 | 11/12/2013 | Added section on integration with enterprise attribute service. |
| 1.2/3 | 11/13/2013 | Typos, format and changes for operator vs. administrator roles. Added temporary event-based permissions. |
| 1.4 | 11/18/2013 | Added diagram on credential use by Integer. |

## Introduction

Hardware, software and cloud providers use different technologies for authentication and authorization functions in the software they provide for the management of their products/services. Some are proprietary, some are based on standards, and others use combinations. These differences are also driven by the specific technology that is to be managed and the way the vendor has implemented that hardware or software product and the protocols they use to manage the devices/software.

### Impact

An important element of the cost associated with multiple stove piped systems for management of different technology elements is additional cost associated with the separate authentication and authorization functions they use. For example:

1. Cost with adds, moves, and changes in users and their roles since there are multiple systems that must be configured and differences from one system to the next.
2. Differences in the granularity of authorization in several dimensions such as scope of systems/resources a user can is granted access, functions they are permitted to perform on each resource and the management objects on which they are allowed to perform these functions (create, delete, modify).
3. Managing separate mechanisms for authentication for each management application.
4. Managing separate mechanisms for authorization and the granularity of the authorizations in each application.
5. Separate logging facilities and how/if activities are logged.
6. Increased security risk as a result of ‘dangling’ user level accounts on one or more management systems or managed elements.

---

[1] The project, Integer, is an attempt to create a unified whole from the separate protocols, data elements and software systems we use to operate our increasingly complex computing environments. See: http://www.thefreedictionary.com/integer. Also see: http://en.wiktionary.org/wiki/integer#Latin

7. Delay in granting ac hoc access to experts during critical periods since multiple systems may have to be provisioned.
8. Integration authentication technologies are used and the way(s) external authentication systems can be accesses.

# Project Integer Authorization Model Requirements

Project Integer is designed to use standard external authorization systems. As a result, the requirements of those systems and protocols are taken as a given and not changeable. For example, if a CAS (Central Authentication Service) is used, then requirements for interaction with the CAS server are defined by the CAS specification.

In short, Integer will rely on services of external systems like CAS for authentication functions. A 'local' option will be supported mostly for testing and other purposes but it is not recommended as a general operating approach. In addition, the system will not assume that CAS is the only method. It will designed to support operation in other environments, for example as would be the case if it were 'fronted' by a Shibboleth SP server.

# Authorization Requirements/Approach

The key challenge for determining which users will have different kinds of access to specific resources is that deployment environments are becoming more rather than less complex. One example is the increased usage of third party cloud providers and the multiple virtual locations they offer. This is further complicated by the fact that in some cases, it is necessary to host part of an application in a local data center while other portions are hosted in a could environment. Add to this complexity a move to a continuous Development to Operations (DevOps). The following sections describe the facilities that Integer provides to determine authorization to resources.

## Access Policy Object

The system uses an Access Policy Object to capture details of which users are permitted different types of access to which set of resources (Service Elements[2]). These objects are associated with different roles. System users may be associated with multiple roles.

## User Access - Which Users are Authorized

The complexity of modern environments dictates that three axes be used to determine is a specific user is to be granted access to a resource:

1. Role - A role is an arbitrary function or set of functions that are entered into Integer. Note that a user may have more than one role at any given time. Examples include:

[2] A service element in project integer represents any accessible resource. This could be a router, name server, database instance, Tomcat server, etc. It is anything that may be accessed via Integer. Each service element with have one and potentially many management objects that can be configured or monitored through a variety of access methods from SSH to SNMP, to vendor proprietary RESTful or other interfaces.

1. Network Engineer
2. Network Operator
3. Database Administrator
4. Application Engineer
5. Software Engineer
6. Application on-call engineer
7. Server/System engineer
8. Server/System operator
9. Tier 1-N support
10. Internal Consultant
11. External Consultant

2. Authentication methods - in some cases, the sensitivity of the information that traverses a service element or is stored on a service elements, or the criticality of the service element indicate one type of authentication or another be used. Integer will be able to discriminate access based on access method[3].

3. Temporal restrictions - In some cases, there may be a need to restrict access to resources (service elements) by users/roles based on time of day. In other cases, the role may need to be active on a recurring basis for a specific duration. Or it may be active once for a specific period of time. In other cases, the permission may be granted based on an event (such as a failure) and access is granted while that condition exists.

## Which Operations are Authorized?

The system must provide some granularity of control with regard to what operations are performed. This is controlled by permissions:

1. Permissions are granted on a per Access Policy Object basis and can be one of three:

   - Read-only - management objects in service elements within the policy scope are only available to people that have been assigned a role associated with this access policy object.

   - Read-write - objects that are writeable as defined by the management protocols and service element configuration may be modified or deleted by users associated with this role and Access Policy Object.

   - Read-(write)-Create - some service elements/instances may be expensive to create, for example spinning up another DB server or allocating large amounts

[3] Note that this approach does not necessarily provide restrictions based on subnet or IP address from which the user accesses the system. If the user is properly authenticated, using the required authentication mechanism(s), they will be deemed to have been properly authenticated and granted access - assuming they meet the other authorization requirements.

of backup storage. For this reason, certain operations may be restricted to read-create.

## Access Policy Scope - Service Element Access

The previous sections described which users may be permitted certain functions based on roles. Policy scope, the third axis, is used to identify which service element instances a user with a specific role is permitted to access.

1. Organization and Organization Location - in modern environments, many organizations may be involved in the delivery of services in an IT environment. This facility allows access to service elements to be controlled based on the organizational membership of users. In addition, some organizations will have multiple locations, access may be further restricted based on location the facility in a particular organization (including virtual cloud locations).

2. Service Element Criticality - not all service elements are equal. This attribute allows the system to control access based on the relative criticality of the service element (even within a service element type such as a router or firewall).

3. Service Element Types - some system users have specialties and work only with certain types of service elements. Examples include database servers, name server systems, mail servers, or routers and firewalls. This attribute allow people with specific roles to be restricted to accessing only certain types of system.

4. Domain Restrictions - A domain is a specific function type used in a technology area. For example, in the Service Technology area of routing, there are the domains of OSPF and BGP. One could also use quality of service which would be a different Domain. In this case, Quality of service might use one or more of several specific mechanisms to achieve the desired QoS. For example WFQ or RED.

5. Instance Level Restrictions - the finest-grained control is the specific enumeration of service element instances that are under the control of the Access Policy Scope instance.

## Integration with Organization Attribute Services

Some of the attributes used by Integer may be centrally managed by an attribute service operated by the enterprise. In cases where updates (e.g., to contact phone number or organization) are performed centrally, Integer will provide a method for integration with that system - but it will not be required for system operation.

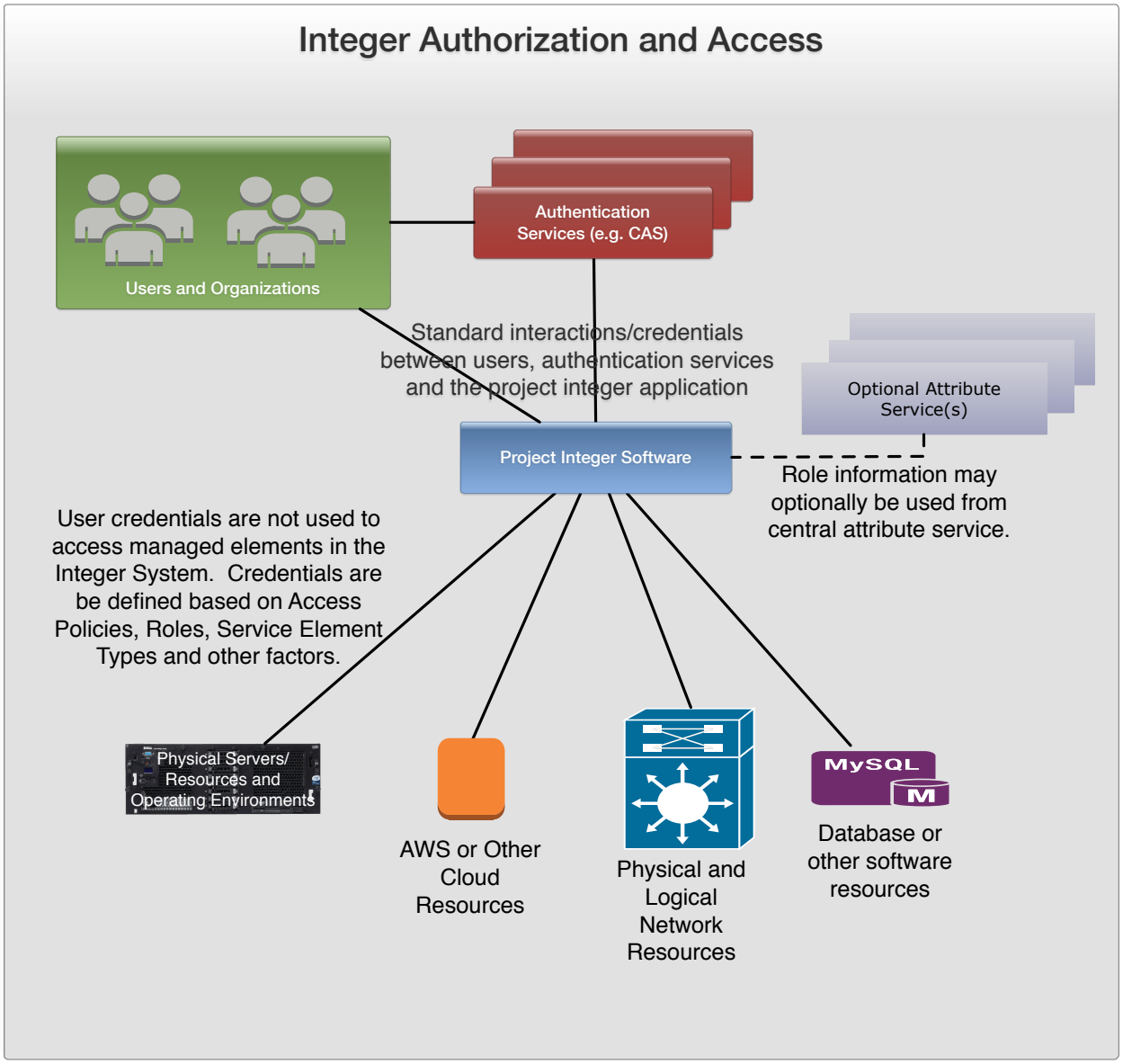
## Security/Logging

To ensure backdoors are not created for unauthorized access, only a very few users should be permitted to have access policies associated with a role that allows creation/modification of access policy objects and roles. For these functions

additional restrictions about the type of authentication, organization or other factors may be applied.

Data that represents all security elements of the system will be encrypted at rest and in transit. This includes user selectable logging levels.

## Interaction Model




Users will authenticate with the designated authentication system(s) and once authenticated[4] will be signed into the system. At this point, the facilities described

[4] Inactivity and other protections can be defined in the authentication system.

above will control access. The system will interact directly with each service element using it's own credentials, not the users.

The previous diagram illustrates that user credentials gain access to Integer. The Integer administrator can determine how fine grained they wish credentials to be between Integer and each service element. For example, all operational read access via SNMP might be configured using a single SNMPv3 user that represents all roles that have read access to a range of service elements. The same user may have a role that permits them access (even if only temporarily) to certain network infrastructure elements like a Cisco router. In this case again the credentials are between the Integer system and the managed element.

In all cases, actions taken by integer on behalf of a user can an should be logged, though Integer administrators will have flexibility in the log levels they set.

This approach can significantly reduce the amount of credential management on each managed element and still preserve detail logging of each user's activity. It also means that in those environments where an operator has had access