PSU Response To Registry Questionnaire

Background

As part of a university Identity and Access Management project, Penn State was tasked with implementing a new Central Person Registry, CPR. The CPR is an intelligent registry for managing person information and represents a fundamental change to business and systems. The creation and management of person information in one central place will position Penn State to support to the increase of cloud services and the increase of systems for HR, Finance, Outreach and World Campus, Student Systems, College of Med and so forth. One single individual can be represented in many systems but should only need to change personal information in one place.

When it came to designing the Central Person Registry, a service-oriented approach was taken. To date, a total of 80+ SOAP services have been developed to address the following registry areas:

- Person Management
  - Names, Addresses, Phones, Email Addresses, Gender, and Date of Birth
- Identity Management
  - Credentials, Penn State Identifiers, ID Card
- Linkages
  - Person and Account
- Affiliations
- Identity Assurance Profiles
- And other things like: Confidentiality, Comments and Security

Each of these major areas are described in detail in the CPR's high-level functional overview. Along with the overview, a request flow from user initiation through service fulfillment is also available for review.

Penn State is considering offering this CPR as an open source. The responses below are solely based on an institutional perspective for managing and governance. Penn State is interested in licensing and supporting this registry for the OSIdM4HE - higher education community. The level of support and type of licensing is still under consideration and up for discussion.

Response to Questions

IAM Registry questions to evaluate features and functionality against standard business requirements.

Category	Description or Question for solution provider	Response	Link(s) to Documentation
General architecture	Describe how ID match capability is provided by the registry solution. For example, is it (a) an integral part of the solution as provided or (b) must it be integrated with an external ID match engine or (c) can it be provided in some other way?	The Penn State registry solution for matching has two parts, an external engine which generates match codes and an algorithm that is part of the registry, both A and B. It's flexible to accommodate other solutions. With regards to the match codes that are generated by the appliance, they take into account variations in name, and address. So a match code for Bill Smith, William Smith and Billy Smith would be the same thing. When the matching process is done, an exact match is attempted using either our Penn State Identifier Number (PSU ID Number), Social Security Number or the userid. If the exact match fails, a near match is done using the match codes for name, date of birth, and address. The result of which is a ranking of the match between 1 and 550. For Penn State a match is a score of at least 330. There are two match algorithms one for domestic and a second for international. In addition to the identity match, the registry is responsible for cleansing the data when possible. To support the cleansing of data, an external product was purchased to validate addresses against the USPS. Products exist to validate addresses for other countries as well. With the near match logic, the CPR will be able to decrease the number of duplicate records. All of this does hinge on consistent data collection, so for the CPR we are going to require new person records to have a name, address and partial/full date of birth. If any part of that information is missing, the data will not be included in matching. The record could be stored in the registry as a orphan. From a matching perspective the flow is: 1. Generate match codes for the input data using the matching appliance. 2. Determine if an exact match can be performed using one of the following: a. Userid, Name Match Code, and DOB (partial or full). b. PSU ID Number, Name Match Code, and DOB (partial or full). c. SSN, Name Match Code, DOB (partial or full). 3. If the exact match is successful information about the person is returned back to the caller. 4. Otherwise, the near match process is executed using the criteria specified in the links in the next column. The near match process will examine the input data against the registry data (active and inactive records). 5. If the records are found that are above the cut off of 330 all of them are returned to the caller in rank order. The data returned will include the PSU ID Number, Userid and the score.	Matching Criteria - Standard Matching Criteria - International
	Describe how groups management (for use with authZ controls and other purposes) is provided. For example, is it (a) handled internally by the solution or (b) integrated with an external group management engine such as Grouper or (c) provided in some other way?	The PSU Central Person Registry has integrated Grouper for access management control to the registry. All Systems of Record (Registration Authorities) are represented as groups. The Registration Authority Agents are assigned roles with permissions. Within the person registry, Grouper is utilized to control authorization to the web services and the data they control. Registration authorities are represented as a group, which is then assigned a role. The role is assigned permissions like execute service or update data. This enables us to remove the authorization from outside the registry.	
Data model	Describe how the registry solution supports an extensible set of attributes about (a) persons, (b) applications or other external resources, and (c) other, arbitrary entities?	The data model is flexible to support additional person attributes by the addition of new database tables and the establishment of the linkages between the person entities and their new attributes. Additionally, attributes such as name and address have types. Types can be added to support specific requirements by various systems of record. The current design for the registry is scoped to people. Entities will be supported in the future either in the existing CPR or in a separate registry appropriately linked to the person registry.	
AuthZ support	Describe how the registry data model supports defining arbitrary user roles in support of authZ functions.	Roles are an integral part of any Access Management solution. The CPR will be used to provide information in the construction of roles, however the roles themselves will not live in the registry, as they will reside in an access management solution such as Grouper.	

Features	Describe how the registry solution supports audit logging of sensitive transactions, including support for the recording of historical changes made to sensitive data. Describe how this log includes the requester and authorizer identities, and transaction timestamps.	The registry has various levels of auditing, the first of which is database logging. All transactions are logged to a service log table. In addition, for each database table we maintain a history of changes to records. Whenever a record is changed, the existing record is marked inactive and a new record is cut. For each database table we have the following fields that determine our history: start_date - the timestamp the record was "started". end_date - will be NULL for active records, otherwise it will be the date the record was made inactive. last_update_by - contains the identity (service or person) which updated the record. last_update_on - contains the timestamp of when the record was last updated. created_by - contains the identity (service or person) which updated the record. created_on - contains the timestamp of when the record was created. Remember whenever there is a change a new record in the particular database is cut. We opted to go this route as opposed to having either a single audit table or multiple audit tables. The changes for records are contained within the tables themselves. In addition to database logging, log4j is used as part of all the Java code.	Refer to the data model at the CPR Design Wiki.
	Describe how the registry solution supports the secure storage of security questions and answers for use in password recovery.	At this time, the data for password security questions and answers are stored in a separate database schema that is outside of our normal registry. The data can only be accessed via the password reset application using a separate database userid and password. The Penn State database vendor, Oracle, does provide the facility to encrypt the answers, which we have implemented.	
	Is there support for multiple name and address types as well as history? If yes, please describe.	Yes, the registry does support multiple types of names, addresses, phones, and email addresses. The types will be A type is associated with each record stored in our names, addresses, phones and email address types. So for example, for a Name record it can either be a legal name, preferred name or a documented name. For a documented name which is obtained from a legal document, we also record the document type (which can be password, driver's license, state identification card or a military identification card). In addition the types are used as part of our authorization decisions. We have designed our authorization scheme to allow RAs to only assign particular types of data, if need be. This authorization is controlled using Grouper. Types could be extended to represent various system of records formatting requirements.	
Identity Assurance	Are registration events captured as they occur? Do these events automatically trigger assignment /deassignment of an IAP	Yes, vetting and proofing data is captured during registration events. The data is accumulative, once the user has met the necessary requirements for a particular IAP, it is automatically assigned. On the flip side, other events like account misuse will trigger a downgrade of the user's IAP. When it came to designing our implementation to IAP, we took the approach of developing a locally defined Penn State IAP and map it to an InCommon one. Why did we do this? Flexibility if InCommon would change the requirements for a particular IAP, our locally defined IAPs can still work. All we would need to do is create a new Penn State IAP that would map to the changed InCommon IAP. As it stands the data requirements for our Penn State IAPs currently exceed the InCommon requirements. That is because we are collecting a consistent set of data necessary for identity matching. Because of that, Penn State Bronze is collecting more data than InCommon. The data collected for Bronze includes: Legal Name, Address of Record, and Date of Birth (full/partial). The data required for Silver includes: Legal Name, Address of Record = physical address is required, valid US address or international address, and Date of Birth (full/partial). ID Proofing required using a Legal Id. Names, Address of Record (if available), and DOB must be proofed in a single event. So how is the data proofed? Penn State IAM has created the notion of an Identity Assurance Agent (IAA), whose job will be to proof identity data for the purposes of IAP assignment. An IAA has to take a training course and then an examination to be certified. They will be using the IAA Web Site to perform the proofing activities.	OSIdM4HEteam: Identity Assurance Data Model
	Is there support for real time provisioning of Identities/services	Yes, the person registry supports the notification of provisioning requests using JMS. When a service and/or batch process is executed that requires a service/identity to be provisioned, a JMS JSON message is sent to the appropriate service provisioner. The results of the provisioning event are retrieved by a standalone daemon which is then used to update the registry. From an implementation perspective, we have selected ActiveMQ as our message bus. We are using a separate message queue for each service provisioner. The CPR will write JSON JMS messages to the provisioners queue based on the type of service that initiated the requested and the data associated with the user. The service provisioners are only authorized to read messages from their appropriate queue. Upon completion of the provisioning event, a response is sent back to the CPR's main message queue with the status of the event. The CPR then determines what to do next based on the status received in the message.	
	Describe how data is processed (batch, web services)	Data within the person registry can be processed in either batch or web services. The web services are SOAP-based. Future plans are for the development of RESTful services. The common core code is isolated in a .jar file that can be shared between the service and batch processes.	
	Is registry dependent on other open source or vendor products? If yes, please provide details.	The registry is built using open source products; Apache Tomcat, Apache CXF, Java, Hibernate, Apache ActiveMQ, JBoss Drools and JAX-WS. The only commercial product that is used by the registry is the matching appliance, which is isolated by the use of a service. So any matching solution can be dropped in with minimal changes.	
	Where is the business logic stored? Is there support for delegation to maintain these rules?	Core business logic for the person registry is stored in a rules engine, Drools Expert. To isolate changes from business logic from rule updates, the rule engine is encapsulated in a service that applications call to process rules. The benefit to this approach is that applications do not need to be redeployed when the rules are changed. With regards to maintenance a future release of the person registry will utilize Drools Guvnor for rules maintenance. Using the Drools Guvnor features and Grouper, the registry will be able to delegate the maintenance of rules.	
	How does the registry notify external entities of data changes? (for example name is changed)	External entities, Service Provisioners, will register their preferences for message receipt with the person registry. Based on service and/or batch execution if there is a change to a data element they have subscribed to receive, a JMS message is sent to them with the change information. Currently, we are doing point-to-point JMS messaging, but plan on looking at push/pull in the second quarter of 2012.	
	Is code located in public repository	No, not at this time. A snapshot of the code is available at a shibboleth protected web site.	
	How are changes, marketing, etc communicated to public? (wiki, lists, web presence)	Our registry is currently not in production. Development will be complete on 3/31/2012 and running on production hardware. Currently communication is all internal to Penn State.	
	Is there proper OSS license?	The open source license is still under discussion but will align with those chosen for other HE open source efforts. Discussions with the IP Office are planned for final approval.	
	Is there a clear project lead?	Penn State's IAM project is sponsored by the VP of IT, Kevin Morooney. He has created an IAM team within Information Technology Services and assigned Renee Shuey as Principal Lead of this effort.	
	Is there an existing project steering committee/governance?	Yes. There is an existing IAM Governance which is sponsored by the Provost and VP of Information Technology. Plans are to introduce an IT Leadership Council working group to provide steering and develop new policies which will be presented to the executives for approval.	