Scenario Description:

Test Registration SSO Login

Version Control

Version #	Date	Author	Description
1.0	04/23/2014	Ryan Marinello	Initial Draft

Test Scripts:

The following scripts, links and pages consist of sso and data warehouse

Test Components:

This test scenario covers the following high-level test requirements (see scripts below for specific requirements covered by each test script):

User Groups: example of groups

- **System Admin User**
- **TestReg Assessment Admin**
- **TestReg Entity Admin**
- **TestReg Student Admin**
- **TestReg User Admin**
- **TestReg Super Admin**
- **TestReg Accommodations Admin**
- **TestReg Eligibility Admin**
- **TestReg Student Group Admin**
- **TestReg Warehouse Admin**

Requirements Traceability:

RADTIB.15 RADTIB.19 RADTIB.24.1 RADTIB.31

Testing Requirements:

- **The Test Registration application will use the Smarter Balanced SSO component for user authentication and authorization**

Script #:1.1 Test Registration login

Purpose:

When Test Registration component is accessed users are re-directed through SSO login before Test Registration dashboard page displays

Assumptions:

- Test Registration, SSO, Test Spec Bank, ProgMan, Permission, Test Item Bank, Core Standard systems are up and running

Prerequisites:

- Test Registration is online and user is able to access URL to SSO login page and supported browsers are used

Test case Steps

Step #	Test Action	Expected Results	Pass/ Fail
L1.0	Open a supported browser and access link http://drcamp-dev.opentestsystem.org:8080/	SSO login page displays	
L1.1	Enter valid user ID and password based on the role type a user wishes to act on behalf of	Test Registration Dashboard page displays, appropriate access is granted and user is able to interact with the dashboard functional areas	

Test Execution

Date/Time	Tester	Test ID	Test Phase	Status
12/09/2013	RM, MR	UI	1	Pass

Requirements Traceability:

RADTIB 16, RADTIB 16.1, RADTIB 16.2, RADTIB 16.3, RADTIB 16.4, RADTIB 16.5, RADTIB 16.6,

Testing Requirements:

- **The Test Registration Components cache this information so that authorization and tenancy can be determined when a user is authenticated.**

- **Test data and dependent elements shall be maintained separately per tenant.**

- **Tests and test elements managed by a given user are done so on behalf of the tenant identified by the entity associated by a role.**

- **Users that do not have a valid role will not be given access to the application.**

- **Users with a valid role that are not associated to a valid tenant will not be given access to the application.**

- **Users with more than one valid role associated with more than one valid tenant will be given access to one and only one tenant at a time.**

Script #:1.1 SSO, Student data export and Data warehouse

Purpose:

- The application allows multi-level access based on the configuration in Program Management, known as crosswalk Mapping

- Data are maintained separately for a tenant
- User must have valid test registration role(s) and permission mappings. No access to the system if no valid role is defined. No access to the system if user is not associated with tenant even if that user has valid subscription, roles and permissions.
- User data is frequently sent to a remote SFTP.
- ALERT to MnA
- File Transfers of userChangeEvents as XML
- Callbacks SSO_USER_EXPORT
- SAML Filtering
- SFTP + Token Keys
- User data continuously is sent to a remote SFTP server. Data warehouse integration is a periodic export of data to a data warehouse landing zone. It incorporates pulling data out of the Test Registration database, compressing it, encrypting it, and then SFTPing the data
- AcademicYear is used to kick off the data warehouse export process
- Eligibility Calculation Fail triggers from callback
- Extracting from TR database, compression, encryption, then SFTP.
- LandingZone is SFTP.
- Requires established sessionUUID
- Assessment Eligibility Types for Export
- Data generated is JSON metadata file
- Data warehouse extract
- Data warehouse callback
- Data warehouse extract status

Step #	Test Action	Expected Results	Pass/ Fail
DW1.0	Open a supported browser and access link http://drcamp-dev.opentestsystem.org:8080/	SSO login page displays	
DW1.2	Enter valid user ID and password based on the role type a user wishes to act on behalf of	Test Registration Dashboard page displays, appropriate access is granted and user is able to interact with the dashboard functional areas	
DW1.3	Using a REST Client Method: POST	Host http://localhost:8080/rest/datawarehous	

Step #	Test Action	Expected Results	Pass/Fail
	Content Type: application/JSON	e/extract/status	
DW1.4	REST client Method: GET Content Type: application/json Response code: 200	Host http://localhost:8080/rest/datawarehouse/guid	
DW1.5	REST client Method: GET Response code: 204	Host http://localhost:8080/rest/datawarehouse/extract	
	Monitoring and Alerting		
M1.0	Open MnA Component	MnA dashboard page displays	
M1.1	Filter Alerts test_registration component	Alerts page displays	
M1.2	Filter Alert Type: SSO	Data should persist for SSO_USER_EXPORT	
M1.3	Sortable Grid Data	Time, Component, Severity, Alert Type, message	
M1.4	Filter Alert Type: Datawarehouse	Datawarehouse_callback Alerts should persist	
M1.5	Create /Modify Notification Rules	Add a Rule for SSO_USER_EXPORT and receive notification or Add a Rule for DATAWAREHOUSE_CALLBACK and also receive notification by your group	

Prerequisites:

- Users are associated with a tenant(s) in Program Management application
- Users have valid acting role(s) and permissions

Test Execution

Date/Time	Tester	Test ID	Test Phase	Status
03/06/2014	RM,SK,TH,SW	SSO Export	1	Pass