6th INTEGRATION TEST CAMP
Review

INTERNATIONAL DATA SPACES ASSOCIATION

SQS SOFTWARE QUALITY SYSTEMS

How was the Integration Test Camp carried out?

It was performed as a virtual event where participants could test the interoperability of their components within an IDSA implementation composed of connectors, DAPS and Broker. Each participant was assigned with a slot of two hours in which the SQS lab was entirely dedicated for them.

This Integration Test Camp mainly focused on the IDS Specification criteria for the connector to verify the different test cases it must meet for the base security profile. In addition, a first contact was also made with a version of an Appstore currently under development.

The SQS team kept the same environment as the 5th Integration Test Camp. We are actively looking for an Appstore and a Clearing House.

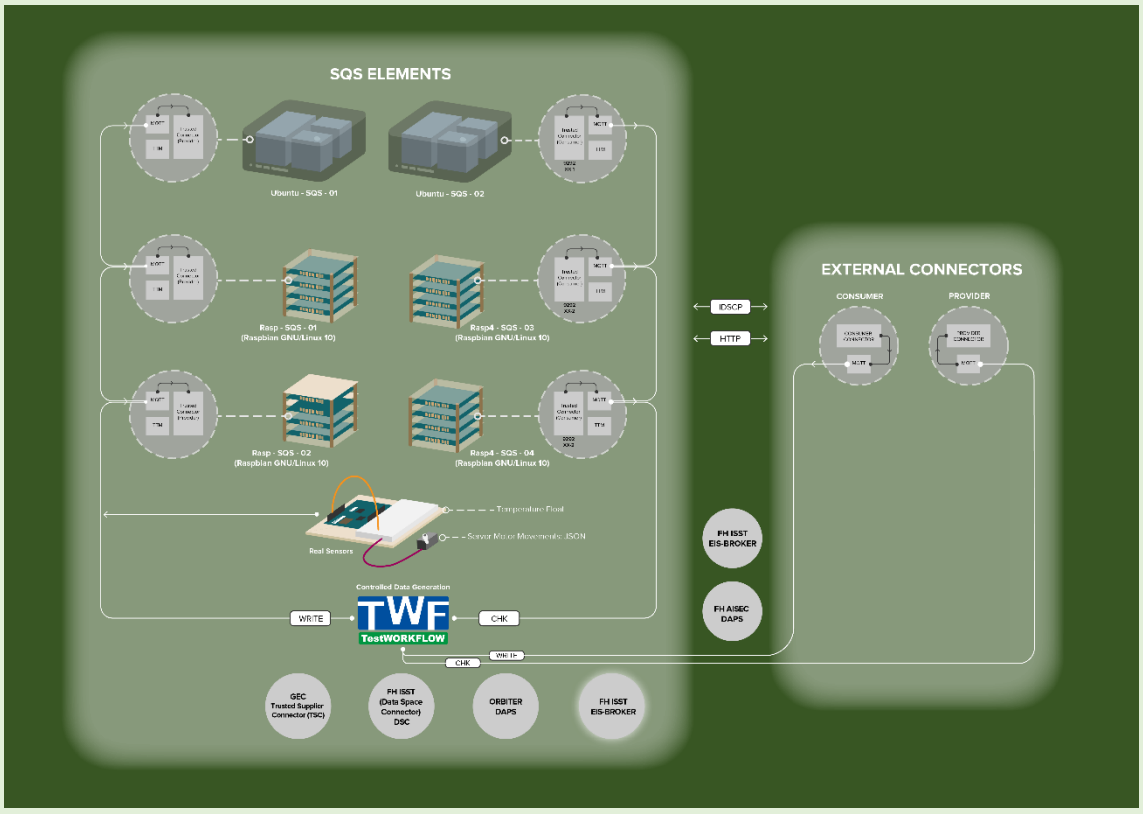

Figure 1 - SQS lab environment (5th edition)

With this architecture, the next scenarios were offered:

1. Connector Under Test (CUT) as Provider and SQS lab connector as Consumer, with IDSCP, IDSCPv2 or HTTP communication protocol
2. Connector Under Test (CUT) as Consumer and SQS lab connector as Provider, with IDSCP, IDSCPv2 or HTTP communication protocol

3. Connector Under Test (CUT) and lab Orbiter DAPS
4. Connector Under Test (CUT) and lab Broker (FH IAIS Broker)

The DAPS in SQS lab (Orbiter DAPS) and the DAPS from FH were offered for all the scenarios. This Integration Test Camp mainly focused on the FH IAIS Broker and a DAT obtained from IAIS because we went deeper into the connector criterion for the base security profile, conducting multiple test cases for each of the criterion to make sure the connector under test correctly meets all the IDS specification requirements.

Additionally, during the Connector and Broker interoperability testing we found out a connector is able to successfully register to the FH IAIS Broker with an Orbiter DAT.

Who participated?

The schedule for this Integration Test Camp:

22-Feb	25-Feb	26-Feb
15:00-16:30 TeraLab	15:00-17:00 TeraLab (2)	15:00-17:00 FIWARE

What were the participants able to do there?

Participants were able to test the interoperability of their pre-commercial components within a real IDSA architecture. They could interact with real IDSA infrastructure components and verify how their components will act in the real world.

For that, SQS lab team proposed the next Test Scenarios:

- Step 1. Environment configuration

  The participants receive the required information to connect to the Integration Test Camp components

- Step 2. Connection to the DAPS (Orbiter and Fraunhofer)
  The participants are able to obtain a valid DAT from the DAPS and have the ability to verify it

- Step 3. Connection to the DAPS (Orbiter and Fraunhofer) (fail)
  The participants try to obtain the DAT with invalid certificate or self-description

- Step 4. Data interchange

  After a successful initial setup, the CUT is able to send/receive data correctly

- Step 5. Data interchange (fail)

  Setup the connector in various ways, such that they all individually get rejected. Invalid certificate, DAT, self-description...

- Step 6. Connection with more than one provider/consumer (optional)

  The CUT provides for more than one consumer
  The CUT receives from more than one provider

- Step 7. Connection with more than one provider/consumer (optional) (fail)

  While the CUT is sending/receiving data, one of the lab connectors is turned off. 2 minutes later, it is turned on. Connection is re-established and data flows correctly. Then, both connectors are turned off. 2 minutes later, they are turned on. The connections are re-established and data flows correctly.

- Step 8. Connector registers to the Broker with both Orbiter and FH DAPS

  Connector performs all the current available Broker calls: register, update, query, …

- Step 9. Connector registers to the Broker with both Orbiter and FH DAPS (fail)

  The CUT is not able to register with an invalid certificate, DAT, self-description…

What have been achieved during the Test Camp?

In the next tables the results of the Test Camp are shown:

IDS Specification			IDS Arch (Connector)		Fraunhofer (AISEC/AISECv2)	
Criterion group	Criterion Identity	Name	Test Case Approach		FIWARE Provider	FIWARE Consumer
Communication Integrity	COM 01	Protected connection	**Get DAT** Component gets a DAT from the DAPS, requesting it with a valid certificate.	Check authenticity	OK	OK
				Check encryption	OK	OK
				Check integrity	OK	OK
			Get DAT – Negative answer Component with an expired or invalid certificate, requests a DAT to the DAPS. The DAPS sends a negative response.	Check authenticity	OK	OK
				Check encryption	OK	OK
				Check integrity	OK	OK
	COM 02	Mutual authentication	**Validate DAT** Component providing data requests DAPS to verify an invalid or expired DAT given by a component requesting for data. DAPS gives a negative response.	Check DAT validity	OK	OK
			Validate DAT – Negative answer Component providing data requests DAPS to verify an invalid or expired DAT given by a component requesting for data. DAPS gives a negative response.	Check DAT is not valid	OK	OK
	COM 03	State of the art cryptography	**Cryptography** What type of encryption? Communication transfer protocol?	Check encryption	OK	OK
Data Usage Control	USC 01	Definition of usage policies	**Usage policy** Stablish with the provided data and usage policy and receive this usage policy attached	Check valid usage policy	NOT TESTED	NOT TESTED
				Check wrong usage policy	NOT TESTED	NOT TESTED
			Re-establishment of usage policy Stablish new usage policy for the data provided	Check new valid usage policy	NOT TESTED	NOT TESTED
				Check new wrong usage policy	NOT TESTED	NOT TESTED

6th INTEGRATION TEST CAMP
Review

	USC 02	Sending of usage policies	**Send usage policy** The provider sends a usage policy and the connector accepts it	Check the consumer accepts the usage policy	NOT TESTED	NOT TESTED
				Check the consumer rejects the usage policy	NOT TESTED	NOT TESTED
			Negotiate usage policy Send usage policy and the consumer rejects it. The consumer renegotiates the usage policy	Check how the consumer renegotiates the usage policy	NOT TESTED	NOT TESTED
Information model	INF 01	Self-description at connector	Create self-description	Check self-description	OK	OK
			Share valid self-description	Check valid self-description	OK	OK
			Share invalid self-description	Check wrong self-description	OK	OK
			Receive valid self-description	Check it is valid	OK	OK
			Receive invalid self-description	Check it is invalid	NOT TESTED	NOT TESTED
	~~INF 02~~	~~Self-description at broker~~	~~Currently deleted from the criterion~~		NA	NA
	INF 03	Self-description content	Contains the following information: a) Cryptographic hash of Connector certificate b) Connector operator c) Data endpoints offered by Connector d) Log format of data endpoints offered e) Security profile of connector f) Connector ID	Check the information that is included in the self-description	a, b, c, e, f -> OK d -> NOK Ask point d	a, b, c, e, f -> OK d -> NOK Ask point d

6th INTEGRATION TEST CAMP
Review

	INF 04	Self-description evaluation	Connector is able to verify the self-description of another connector	Check it is a valid self-description	NOT TESTED	NOT TESTED
	INF 05	Dynamic attribute tokens	Data interchange Consumer connector requests data, with a valid DAT, to the provider connector. Provider connector verifies DAT, and the data is interchanged.	Check validity of both DATs	OK	OK
			One invalid DAT	Check message one invalid	OK	OK
			Both invalid DATs	Check message both invalid	NOT TESTED	NOT TESTED
			Share DAT every connection, this is being able to validate yourself at every connection	Check your identity with connection	OK	OK
Identity and access management	IAM 01	Connector identifier	Get identification. The connector is unambiguously identified by means of an identifier derived from a X.509 certificate	Check X.509 certificate valid	NOT TESTED	NOT TESTED
				Check X.509 certificate not valid	NOT TESTED	NOT TESTED
	IAM 02	Time Service	Certificate is valid	Check valid	NOT TESTED	NOT TESTED
			Certificate is not valid (expired)	Check expired	NOT TESTED	NOT TESTED
			Switch off time service	Check message	NOT TESTED	NOT TESTED
			Modify time (forward)	Check message	NOT TESTED	NOT TESTED
			Modify time (actual)	Check message	NOT TESTED	NOT TESTED
			Modify time (backward)	Check message	NOT TESTED	NOT TESTED

6th INTEGRATION TEST CAMP
Review

	IAM 03	Online certificate status check	Get online certificate status What type of online certificate status is used in the CUT?	Check valid -> Request valid/invalid	NOT TESTED	NOT TESTED
				Check invalid -> Request valid/invalid	NOT TESTED	NOT TESTED
	IAM 04	Attestation of dynamic attributes	Refer to COM 02 This criterion is already checked with the test cases in COM 02	Check COM 02	OK	OK
Broker service	BRK 01	Broker service inquiries	**Register** CUT is registered at the broker	Check registration	OK	OK
		Broker service inquiries	Query for connector with valid self-description	Check CUT at broker	NOT TESTED	NOT TESTED
		Broker service inquiries	Query for connector without a self-description or with a wrong self-description	Check CUT at broker	OK	OK
	BRK 02	Broker registration	Check CUT registration at broker	Check properly registered	NOT TESTED	NOT TESTED
				Check registration is denied	NOT TESTED	NOT TESTED
		Broker registration	Check CUT self-description at broker	Check valid self-description	NOT TESTED	NOT TESTED
				Check invalid self-description	OK	OK
	BRK 03	Broker registration update	Update connector information CUT updates its information to a broker	Check valid self-description	NOT TESTED	NOT TESTED
				Check invalid self-description	NOT TESTED	NOT TESTED
		Broker registration update	Update valid self-description Does it show? Does it show unavailable? Does it even update?	Check if it is implemented right	NOT TESTED	NOT TESTED
		Broker registration update	Unregister CUT The connector gets unregistered from a broker	Check the CUT is not available at the broker	NOT TESTED	NOT TESTED

6th INTEGRATION TEST CAMP
Review

Operating System	OS 01	Container support	Connector supports installation and execution of containers	Check installation and execution	OK	OK
Apps and App Store Connection	APS 01	App signature	CUT supports only apps possessing a valid signature	Check signature	NA	NA
	APS 02	App signature verification	CUT verifies signature after downloading and before installation	Check signature certification	NA	NA
	APS 05	App installation	CUT supports apps delivered and installed as independent software containers	Check app installation and containerization	NA	NA
	APS 06	App Store	CUT receives apps from a central App Store	Check central App Store	NA	NA
Data Usage Transparency	AUD 01	Access control logging	Connector logs each access control decision in the form of an integrity protected entry in its domain	Check access control logs	NOT TESTED	NOT TESTED
	AUD 02	Data access logging	Connector logs every access to data in the form of an integrity protected entry in its domain	Check access to data logs	NOT TESTED	NOT TESTED
	AUD 03	Configuration changes logging	Connector logs any changes made to its configuration in the form of integrity protected entries	Check in logs changes in configuration	NOT TESTED	NOT TESTED

6th INTEGRATION TEST CAMP
Review

INTERNATIONAL DATA SPACES ASSOCIATION

SQS SOFTWARE QUALITY SYSTEMS

TeraLab Appstore

- TeraStore is the IDS implementation of an Appstore based on the TeraLab marketplace. It is based on docker and has two groups of information based on JSON documents:
    1. Metadata: helping with defining and classifying Apps.
    2. Deployment: necessary information to deploy a docker container in a configurable way.
- During the Integration Test Camp, it was checked that the TeraStore Appstore was able to obtain a valid DAT from the orbiter DAPS provided by SQS. This will initiate the first step of communication with the outside network regarding future Integration Test Camps.
- In this scenario the following test cases were carried out:
    - Get the self-description of the AppStore
    - Get the catalogue (collection of metadata)
    - Request an artifact for an App
    - Create a new App
    - Check that the new App appears in the catalogue

What difficulties have been encountered?

- In the first session with TeraLab, the AppStore was not setup to obtain a DAT from the locally installed Orbiter DAPS.
    - After going over the communication guide, handshake documentation and the information model, the issue was fixed. By the second session, the participant was able to successfully obtain a DAT and verify its validity.
- Connector was not checking the validity of the incoming connector's DAT. The policies are sent in the self-description.
    - This is being worked on to align with the guidelines established for the IDS Information Model.
- Going more in depth into some of the connector's expected functionality, we were able to identify issues that did not align with the agreed IDS certification criterions.
    - Some issues are being worked on; some have been redirected to the WG Certification for further clarification.

Discovered

We were able to discover that a **connector with a valid Orbiter DAT can register to the Broker**. We were testing the CUT and Broker interoperability when we came across this. This was a test case that has not been performed before. The CUT obtained a valid DAT from the Orbiter DAPS and tried registering to the Broker (ConnectorAvailableMessage). The expected outcome was a rejection message as the Broker responds to AISEC DATs. To make sure the Broker was working as intended, we then registered a connector with an AISEC valid DAT. A query request (QueryMessage) revealed the connector registered with the Orbiter DAT in the list.

The Broker developers have been contacted to ensure this is how they intend it to work as there is no information about this being the Broker's behaviour.

Conclusions

SQS wants to thank all the participants that have taken part. It has been a great opportunity to meet each participant and to understand their component better. It allowed us to get a deeper knowledge of what is being developed in IDSA environment. The Integration Test Camp has been a great opportunity to collaborate and face the needs of the components to work with each other.

With the feedback received and the lessons learned from the problems faced in the session, SQS will keep improving the lab environment and the organization for the next Integration Test Camp.