**Contributor: Nilesh Khambekar** *(khambekar@ieee.org)*

1. **System Administration and Security**
   1. **Administration**

Administrative functions on the SDs and SSMs are largely assumed to be implementer-specific, and out of scope for the standard, but recommendations are included in the informative annex.

This interface shall have a secure mechanism to administer the system, allowing:

* performing of SD calibrations
* updating firmware of SDs
* changing configuration of SD, and making associated changes to the SD configuration file
* triggering SD reboots and other SD hardware maintenance functions.

Additionally, administrator shall have a mechanism allowing

* pause or delete particular scanning task and corresponding message-queue/topic.
* flush a message-queue and restart task with the existing/new taskID
* dissociate a TA and all scheduled/ongoing tasks from the TA
* dissociate a SD
* migrate all sensing schedules for an SD to other SD(s).

The administration interface must be a secure interface with key exchange, and these keys must not be the same as keys used for TASKING AGENT<>SM<>SD authentication.

The standard suggests using special administration mode (AdminCmd and AdminCmdResponse) in the message header. There may be multiple response messages for a single command indicating the execution status of the command such as scheduled, in-progress, and done. Following table identifies administration commands for various interfaces.

* + 1. **SD Administration**

SM-SD Interface Summary *(Actual messages need to be detailed upon high level agreement over the content.)*

|  |  |  |  |
| --- | --- | --- | --- |
| Administration Command | SD Action | Administration Response | Details |
| NAME: admCalibrate  DATA TYPE: String | SD performs CALif no time specified else schedules CAL. | NAME: admCalibrateStatus  DATA TYPE: String | The time may be optionally specified with this command. The status from the SD could be SD\_CAL\_IN\_PROGRESS, SD\_CAL\_SCHEDULED, SD\_CAL\_DONE, or SD\_CAL\_ERR |
| NAME: admFwUpdate  DATA TYPE: String | SD updates firmware | NAME: admFwUpdateStatus  DATA TYPE: String | The SM provides the firmware path, name, and update time. The SD status could be SD\_FW\_UPDATE\_SCHEDULED, SD\_FW\_UPDATE\_IN\_PROGRESS, SD\_FW\_UPDATE\_DONE, or SD\_FW\_UPDATE\_ERR |
| NAME: admConfig  DATA TYPE: String | SD applies suggested configuration | NAME: admConfigStatus  DATA TYPE: String | The SM provides the config path, filename, and update time. The SD status could be SD\_CONF\_UPDATE\_SCHEDULED, SD\_CONF\_UPDATE\_IN\_PROGRESS, SD\_CONF\_UPDATE\_DONE, or SD\_CONF\_UPDATE\_ERR |
| NAME: admPowerCycle  DATA TYPE: String | SD performs power cycle | NAME: admPowerCycleStatus  DATA TYPE: String | The SM optionally provides time. The SD status could be SD\_POWER\_CYCLE\_SCHEDULED, SD\_POWER\_CYCLE\_DONE, or SD\_POWER\_CYCLE\_ERR |

In addition to these commands, it is possible to define custom commands for administration. The standard may include more common use-cases in its next revision.

* + 1. **Sensing Task Administration**
       1. **Pause/Resume a sensing task**

It may be essential to pause a sensing task due to administrative or technical issue. In this case, a message-exchange is initiated from the SM to notify corresponding TA and SDs. SM coordinates this action with DM and a message exchange is initiated from DM to DCs. The SCOS administrator may need to resume the sensing task after the issue is resolved. Table <#> identifies associated interactions on the SM<->SD, SM <->TA, SM<->DM, and DM<-> DC interfaces.

* + - 1. **Restart a sensing task**

It may be essential to restart a sensing task due to administrative or technical issue. Here, in certain use-cases, TAs may prefer that a scanTaskID generated. TAs may communicate this preference during association (useNewTaskforRestart). If TA prefers a new task ID, old task ID is also communicated for associating the two scanning tasks.

A message-exchange is initiated from the SM to notify corresponding TA and SDs. SM coordinates this action with DM and a message exchange is initiated from DM to DCs. Table <#> identifies associated interactions on the SM<->SD, SM <->TA, SM<->DM, and DM<-> DC interfaces.

* + - 1. **Migrate all the sensing tasks associated with an SD**

It may be essential to migrate all sensing tasks to other SCOS resources (due to technical or security issue) . SM may assign a new task ID depending on TAs preferences.

A message-exchange is initiated from the SM to notify corresponding TA and SDs. SM coordinates this action with DM and a message exchange is initiated from DM to DCs. Table <#> identifies associated interactions on the SM<->SD, SM <->TA, SM<->DM, and DM<-> DC interfaces.

* + - 1. **Delete a sensing task**

It may be essential to delete a sensing task (due to technical, resource, or security issue) .

A message-exchange is initiated from the SM to notify corresponding TA and SDs. SM coordinates this action with DM and a message exchange is initiated from DM to DCs. Table <#> identifies associated interactions on the SM<->SD, SM <->TA, SM<->DM, and DM<-> DC interfaces.

* + - 1. **Delete all sensing tasks from a TA**

It may be essential to delete a sensing task (due to technical, resource, or security issue) .

A message-exchange is initiated from the SM to notify corresponding TA and SDs. SM coordinates this action with DM and a message exchange is initiated from DM to DCs. Table <#> identifies associated interactions on the SM<->SD, SM <->TA, SM<->DM, and DM<-> DC interfaces.

* + - 1. **Summary of the message interactions for sensing task administration**

SM-TA Interface messages summary *(Actual messages need to be detailed upon high level agreement over the content.)*

SM-SD Interface messages summary *(Actual messages need to be detailed upon high level agreement over the content.)*

SM-DM Interface messages summary *(Actual messages need to be detailed upon high level agreement over the content.)*

DM-DC Interface messages summary *(Actual messages need to be detailed upon high level agreement over the content.)*

* + 1. **SCOS Platform Behavior Specification**

In addition to explicit administration, SCOS operator could define behavior of SCOS platform using policy policy construct. Following are a few examples of specifying SCOS platform behavior

* Allowed data distribution modes - streaming and/or store-forward
* Protocol choice for each of the interfaces - An SCOS operator may prefer MQTT for DM-DC interface and another operator may choose RESTful HTTP interface.
* Security mode for each of the interfaces - An SCOS operator may choose to enforce secure transport for all the interfaces (SD-DM, SD-SM, DM-SM, TA-SM, and DM-DC)
  1. **Security Systems**
     1. **Scope**

The SCOS standard includes security measures toward the maintaining the integrity and confidentiality of the sensing tasks and sensing data. Also, SCOS standard includes measures for ensuring authenticity of the messages. The standard makes provision for the security features and these are highly recommended however, it is upto the SCOS administrator to enforce the security mechanisms on most interfaces.For SM-DM interface, the SCOS platform must include security mechanisms for maintaining the integrity and confidentiality of the sensing tasks and sensing data.

Another part of platform security is authorization. Administrators need to ensure that only authorized users can issue sensing tasks, only authorized users have access to the sensing data published by the platform, and only authorized users can issue certain privileged scans. The standard in the current draft provides an approach to implementing administrative policies. Currently, in this version of the standard, The approach is not a mandatory approach and SCOS administrators may choose to implement policy using an alternate approach.

* + - 1. **Out of scope**

Following security aspects are out of scope of the standard

* Physical security of the infrastructure (SDs, SM, DM)
* Availability of the sensing data - The sensing tasks would typically generate enormous amount of sensing data and the SCOS standard does not require the SCOS platform implementation to make sensing data available past it has been received by the data clients. If data clients happen to lose the data, sensing needs to be performed again.
* Redundancy model for SM and DM - SM and DM are key entities and administrators may include a redundancy model for SM and DM to improve SCOS platform availability. The redundancy model is out of scope of the standard,
  + 1. **Authentication, Confidentiality, and Integrity**

The standard requires implementing following procedure on SM-DM interface. On the remaining interfaces, it is highly recommended.

* TLS mutual authentication shall be performed per [n.1]
* EnityA (For example,TA) communicates with EntityB (For example, SM). TLS-v1.2 as specified in [n.3] shall be used to perform authentication. Previous versions of TLS (e.g., TLS-v1.1 per RFC-4346, TLS-v1.0 per RFC-2246 or SSL-v3.0) shall not be used.
* During the TLS exchange, mutual authentication shall be performed. The EntityA (For example, TA) initiating the TLS connection shall authenticate the EntityB (For example, SM), and the EntityB (For example, SM) shall authenticate the EntityA (For example, TA).
* During the TLS message exchange, the EntityA (For example, TA) shall authenticate EntityB (For example, SM) according to the procedures defined in [n.4]. Server certificate validation shall be performed according to the procedures in [n.5].
* A EntityA (For example, TA) which is unable to successfully authenticate an EntityB (For example, SM) shall abort the TLS connection establishment procedure. It is implementation specific when the EntityA (For example, TA) should re-attempt the TLS connection establishment procedure.
* During the TLS message exchange, the EntityA (For example, TA) provides its client certificate to the EntityB (For example, SM). The EntityB (For example, SM)shall perform client certificate validation according to the procedures in [n.5]. The EntityB (For example, SM) which is unable to successfully authenticate a EntityA (For example, TA) shall abort the TLS connection establishment procedure.
  + 1. **Authorization**

The standard suggests implementing authorization using policy construct for SCOS control plane and data plane.

* + - 1. **Control Plane Authorization**

Control plane authorization is implemented at the SM. It includes the

* ability to enforce a regulatory policy to determine if the location, time, frequency specified in the requested scan are compliant with regulations
* ability to check if the user is authorized to issue the requested sensing task
* ability to define priority for all scans from a specific user
* ability to define priority for a certain set of SD resources
  + - 1. **Data Plane Authorization**

Data plane authorization is implemented at the DM. It includes the

* ability to check if the data client is authorized to subscribe to the requested sensing data.
* rules for how data is distributed for various data clients. Only privileged data clients may be able to request store and forward interface.
* ability to define max storage space in case of store and forward mode

The SCOS security mechanisms and certain administration mechanisms (as described in Section 7.1.3) are implemented using a policy file which is securely installed by the SCOS operator.