The Consultative Committee for Space Data Systems

Report Concerning Space Data System Standards

# SPACE DATA LINK SECURITY PROTOCOL—EXTENDED PROCEDURES—SUMMARY OF CONCEPT AND RATIONALE

INFORMATIONAL REPORT

CCSDS 350.11-G-1

GREEN BOOK
July 2024

The Consultative Committee for Space Data Systems

Report Concerning Space Data System Standards

# SPACE DATA LINK SECURITY PROTOCOL—EXTENDED PROCEDURES—SUMMARY OF CONCEPT AND RATIONALE

INFORMATIONAL REPORT

CCSDS 350.11-G-1

GREEN BOOK
July 2024

# AUTHORITY

| | |
|---|---|
| Issue: | Informational Report, Issue 1 |
| Date: | July 2024 |
| Location: | Washington, DC, USA |

This document has been approved for publication by the Management Council of the Consultative Committee for Space Data Systems (CCSDS) and reflects the consensus of technical panel experts from CCSDS Member Agencies. The procedure for review and authorization of CCSDS Reports is detailed in *Organization and Processes for the Consultative Committee for Space Data Systems* (CCSDS A02.1-Y-4).

This document is published and maintained by:

CCSDS Secretariat
National Aeronautics and Space Administration
Washington, DC, USA
Email: secretariat@mailman.ccsds.org

# FOREWORD

This document is a CCSDS Report, which contains background, rationale and a concept of operation to support the CCSDS Recommended Standard on the Space Data Link Security Protocol (reference [1]).

Through the process of normal evolution, it is expected that expansion, deletion, or modification of this document may occur. This Report is therefore subject to CCSDS document management and change control procedures, which are defined in *Organization and Processes for the Consultative Committee for Space Data Systems* (CCSDS A02.1-Y-4). Current versions of CCSDS documents are maintained at the CCSDS Web site:

http://www.ccsds.org/

Questions relating to the contents or status of this document should be sent to the CCSDS Secretariat at the email address indicated on page i.

At time of publication, the active Member and Observer Agencies of the CCSDS were:

Member Agencies

- Agenzia Spaziale Italiana (ASI)/Italy.
- Canadian Space Agency (CSA)/Canada.
- Centre National d'Etudes Spatiales (CNES)/France.
- China National Space Administration (CNSA)/People's Republic of China.
- Deutsches Zentrum für Luft- und Raumfahrt (DLR)/Germany.
- European Space Agency (ESA)/Europe.
- Federal Space Agency (FSA)/Russian Federation.
- Instituto Nacional de Pesquisas Espaciais (INPE)/Brazil.
- Japan Aerospace Exploration Agency (JAXA)/Japan.
- National Aeronautics and Space Administration (NASA)/USA.
- UK Space Agency/United Kingdom.

Observer Agencies

- Austrian Space Agency (ASA)/Austria.
- Belgian Science Policy Office (BELSPO)/Belgium.
- Central Research Institute of Machine Building (TsNIIMash)/Russian Federation.
- China Satellite Launch and Tracking Control General, Beijing Institute of Tracking and Telecommunications Technology (CLTC/BITTT)/China.
- Chinese Academy of Sciences (CAS)/China.
- China Academy of Space Technology (CAST)/China.
- Commonwealth Scientific and Industrial Research Organization (CSIRO)/Australia.
- Danish National Space Center (DNSC)/Denmark.
- Departamento de Ciência e Tecnologia Aeroespacial (DCTA)/Brazil.
- Electronics and Telecommunications Research Institute (ETRI)/Korea.
- Egyptian Space Agency (EgSA)/Egypt.
- European Organization for the Exploitation of Meteorological Satellites (EUMETSAT)/Europe.
- European Telecommunications Satellite Organization (EUTELSAT)/Europe.
- Geo-Informatics and Space Technology Development Agency (GISTDA)/Thailand.
- Hellenic National Space Committee (HNSC)/Greece.
- Hellenic Space Agency (HSA)/Greece.
- Indian Space Research Organization (ISRO)/India.
- Institute of Space Research (IKI)/Russian Federation.
- Korea Aerospace Research Institute (KARI)/Korea.
- Ministry of Communications (MOC)/Israel.
- Mohammed Bin Rashid Space Centre (MBRSC)/United Arab Emirates.
- National Institute of Information and Communications Technology (NICT)/Japan.
- National Oceanic and Atmospheric Administration (NOAA)/USA.
- National Space Agency of the Republic of Kazakhstan (NSARK)/Kazakhstan.
- National Space Organization (NSPO)/Chinese Taipei.
- Naval Center for Space Technology (NCST)/USA.
- Netherlands Space Office (NSO)/The Netherlands.
- Research Institute for Particle & Nuclear Physics (KFKI)/Hungary.
- Scientific and Technological Research Council of Turkey (TUBITAK)/Turkey.
- South African National Space Agency (SANSA)/Republic of South Africa.
- Space and Upper Atmosphere Research Commission (SUPARCO)/Pakistan.
- Swedish Space Corporation (SSC)/Sweden.
- Swiss Space Office (SSO)/Switzerland.
- United States Geological Survey (USGS)/USA.

# DOCUMENT CONTROL

| Document | Title | Date | Status |
|---|---|---|---|
| CCSDS 350.11-G-1 | Space Data Link Security Protocol—Extended Procedures—Summary of Concept and Rationale, Informational Report, Issue 1 | July 2024 | Original issue |
| EC 1 | Editorial change 1 | August 2024 | Modifies title |

# CONTENTS

Section Page

## CONTENTS (continued)

Figure Page

# 1 INTRODUCTION

## 1.1 PURPOSE

This Report has been developed to present the concept and rationale of the CCSDS Recommended Standard on the Space Data Link Security (SDLS) Protocol Extended Procedures. This Green Book will enable mission designers and protocol implementers to:

a) understand the purpose and usage of the SDLS Extended Procedures;

b) select appropriate procedures and parameters for the mission;

c) cover nominal and contingency scenarios;

d) understand the performance and limitations of the Extended Procedures.

The SDLS Protocol (reference [1]) is a protocol that implements user-selected security services to the data transported by the Space Data Link (SDL) protocols (references [4], [5], [6], and [7]). The SDLS protects the Service Data Units (SDUs) transported by the SDL protocol and, in addition, selected SDL protocol data structures, taking into account compatibility constraints with SDL and Space Link Extension (SLE) services.

The Recommended Standard for SDLS Extended Procedures (EP) (reference [2]) extends the SDLS protocol with a standardized set of auxiliary services for managing an implementation of the SDLS protocol. These EP services are categorized into Key Management, Security Association (SA) Management, and SDLS Monitoring & Control. Further, reference [2] specifies service interfaces and data structures for transport of EP service messages within the SDL protocols along with a security unit status reporting mechanism.

SDLS Extended Procedures encompass well-known procedures such as Over-The-Air Rekeying (OTAR), which are fully documented in the Blue Book (reference [2]). Furthermore, this report describes the concept of operations and illustrates normal and contingency scenarios so that mission designers and protocol implementers can make optimal use of the SDLS EP Recommended Standard.

## 1.2 SCOPE

The information contained in this Report is informative, and not a normative part of the CCSDS Recommended Standards on the Space Data Link Security Protocol (references [1] and [2]). In the event of any conflict between the Recommended Standards and the material presented herein, the Recommended Standards are peremptory.

## 1.3 ORGANIZATION OF THIS REPORT

Section 2 presents an overview of the Extended Procedures, the rationale for their development, and the major design goals and constraints.

Section 3 provides a concept of operation for using the protocol's security services; in particular, the transmission of EP Protocol Data Units (PDUs) within CCSDS protocol stacks, the logical order of EP operations, and the data structures, fields, and functions are given.

Section 4 provides a discussion of key design concepts of the protocol, including handling of EP signaling errors and execution failures, implementation of redundancy, and off-nominal operations. It also discusses the use of SDLS within several example mission scenarios.

Annex A elaborates on the baseline implementations.

Annex B provides a list of acronyms and abbreviations.

## 1.4 CONVENTIONS AND DEFINITIONS

Generic definitions for the security terminology applicable to this and other CCSDS documents are provided in reference [3]. This document uses the following additional definitions:

**Initiator**: An entity initiating a procedure (a proactive participant).

**Recipient**: An entity reacting to an initiated procedure (a reactive participant).

## 1.5 REFERENCES

The following documents are referenced in this Report. At the time of publication, the editions indicated were valid. All documents are subject to revision, and users of this Report are encouraged to investigate the possibility of applying the most recent editions of the documents indicated below. The CCSDS Secretariat maintains a register of currently valid CCSDS documents.

[1] *Space Data Link Security Protocol*. Issue 2. Recommendation for Space Data System Standards (Blue Book), CCSDS 355.0-B-2. Washington, D.C.: CCSDS, July 2022.

[2] *Space Data Link Security Protocol—Extended Procedures*. Issue 1. Recommendation for Space Data System Standards (Blue Book), CCSDS 355.1-B-1. Washington, D.C.: CCSDS, February 2020.

[3] *Information Security Glossary of Terms*. Issue 3. Recommendation for Space Data System Practices (Magenta Book), CCSDS 350.8-M-3. Washington, D.C.: CCSDS, February 2024.

[4] *TM Space Data Link Protocol*. Issue 3. Recommendation for Space Data System Standards (Blue Book), CCSDS 132.0-B-3. Washington, D.C.: CCSDS, October 2021.

[5] *TC Space Data Link Protocol*. Issue 4. Recommendation for Space Data System Standards (Blue Book), CCSDS 232.0-B-4. Washington, D.C.: CCSDS, October 2021.

[6] *AOS Space Data Link Protocol*. Issue 4. Recommendation for Space Data System Standards (Blue Book), CCSDS 732.0-B-4. Washington, D.C.: CCSDS, October 2021.

[7] *Unified Space Data Link Protocol*. Issue 3. Recommendation for Space Data System Standards (Blue Book), CCSDS 732.1-B-3. Washington, D.C.: CCSDS, June 2024.

[8] *Symmetric Key Management*. Issue 1. Recommendation for Space Data System Practices (Magenta Book), CCSDS 354.0-M-1. Washington, D.C.: CCSDS, December 2023.

[9] *Space Missions Key Management Concept*. Issue 1. Report Concerning Space Data System Standards (Green Book), CCSDS 350.6-G-1. Washington, D.C.: CCSDS, November 2011.

[10] *Space Data Link Security Protocol—Summary of Concept and Rationale*. Issue 2. Report Concerning Space Data System Standards (Green Book), CCSDS 350.5-G-2. Washington, D.C.: CCSDS, January 2024.

[11] *Communications Operation Procedure-1*. Issue 2. Recommendation for Space Data System Standards (Blue Book), CCSDS 232.1-B-2. Washington, D.C.: CCSDS, September 2010.

[12] *CCSDS Cryptographic Algorithms*. Issue 2. Recommendation for Space Data System Standards (Blue Book), CCSDS 352.0-B-2. Washington, D.C.: CCSDS, August 2019.

[13] *Security Threats against Space Missions*. Issue 3. Report Concerning Space Data System Standards (Green Book), CCSDS 350.1-G-3. Washington, D.C.: CCSDS, February 2022.

[14] *Security Guide for Mission Planners*. Issue 2. Report Concerning Space Data System Standards (Green Book), CCSDS 350.7-G-2. Washington, D.C.: CCSDS, April 2019.

[15] Morris Dworkin. *Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC*. National Institute of Standards and Technology Special Publication 800-38D. Gaithersburg, Maryland: NIST, November 2007.

[16] Nigel Smart, ed. *ECRYPT II Yearly Report on Algorithms and Keysizes (2011-2012)*. Revision 1.0. ICT-2007-216676. Luxembourg: CORDIS, 30 September 2012.

[17] Markku-Juhani O. Saarinen. *GCM, GHASH and Weak Keys*. Cryptology ePrint Archive: Report 75490220. Reno, Nevada: IACR, 2012.

# 2 OVERVIEW AND RATIONALE

## 2.1 SPACE DATA LINK SECURITY

The SDLS Protocol (reference [1]) is a protocol that implements user-selected security services to the data transported by the SDL protocols (references [4], [5], [6], and [7]). The SDLS protects the SDUs transported by the SDL protocol and, in addition, selected SDL protocol data structures, taking into account compatibility constraints with SDL and SLE services. Figure 2-1 depicts the basic SDLS capabilities defined in reference [1].

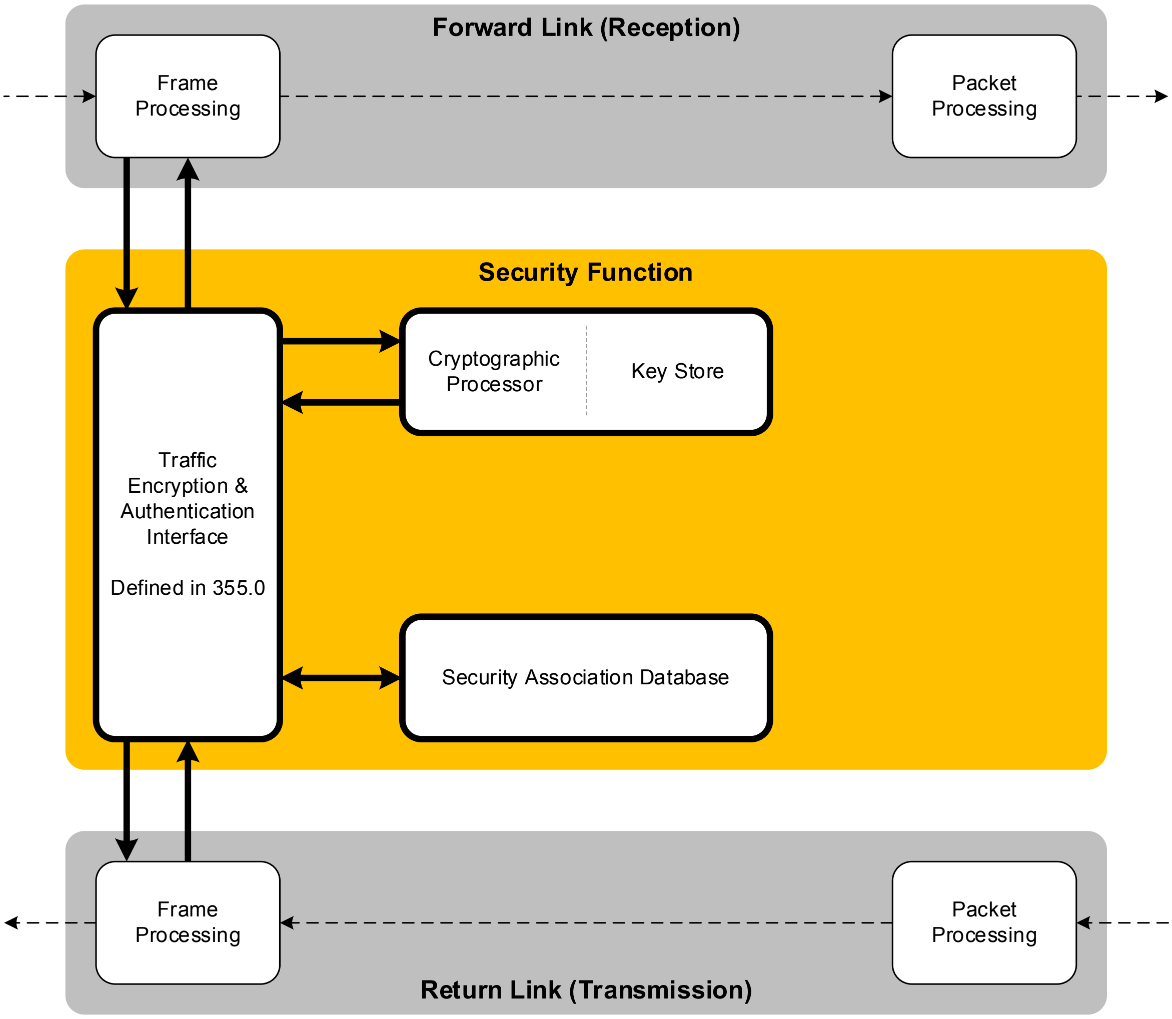


**Figure 2-1: Traffic Encryption & Authentication Interface**

## 2.2 EXTENDED PROCEDURES

The Recommended Standard for SDLS Extended Procedures (reference [2]) extends the SDLS protocol (reference [1]) with services for managing the security parameters of the space link. The purpose of SDLS Extended Procedures is to provide a standardized set of auxiliary services for managing an implementation of the SDLS protocol. These EP services are categorized into Key Management, SA Management, and SDLS Monitoring & Control. The SDLS EP specification also includes service interfaces and data structures for transport of EP service messages within the SDL protocols and a security unit status reporting mechanism. Figure 2-2 depicts the additional capabilities defined in reference [2].

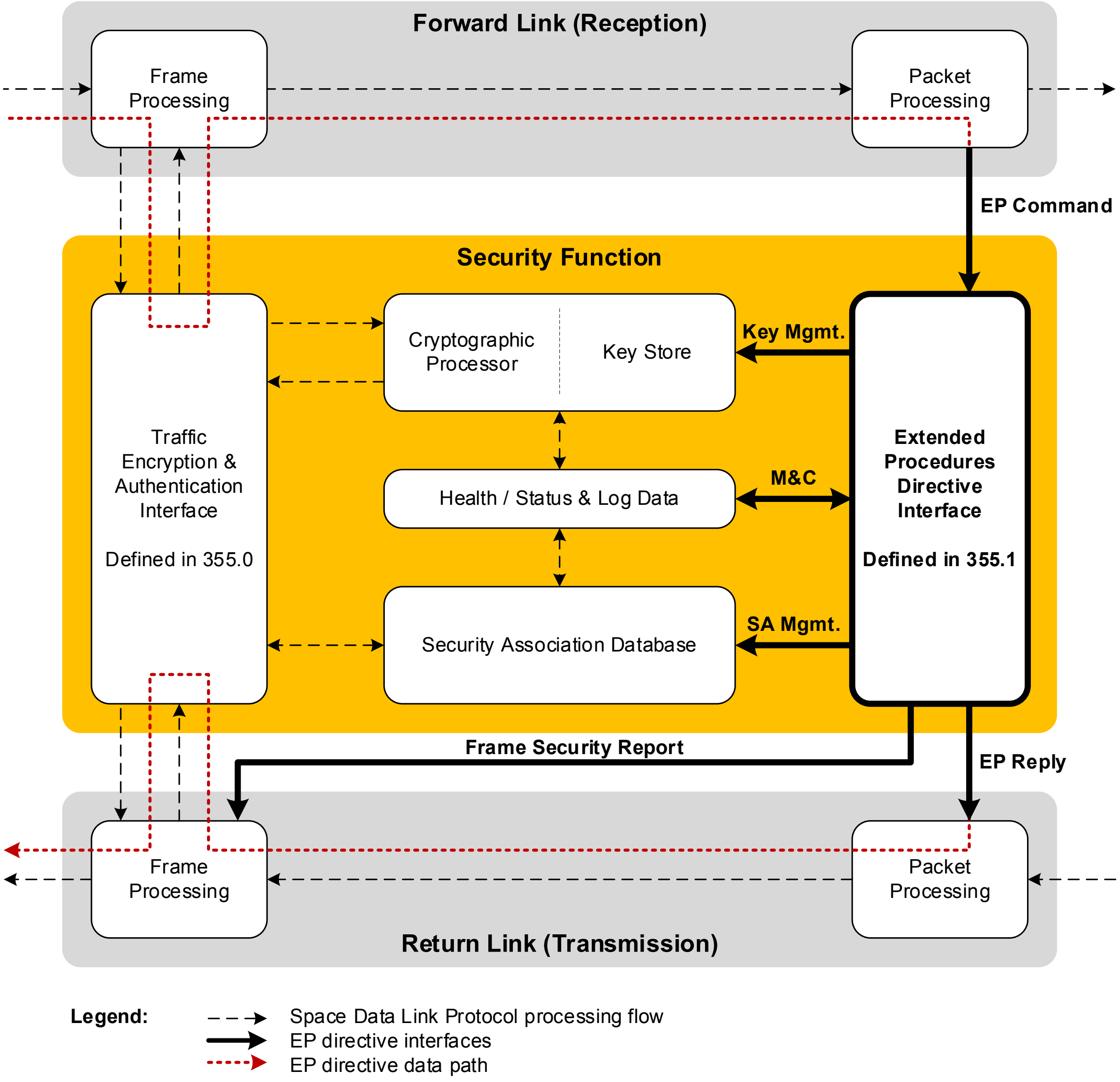


Figure 2-2: Extended Procedures Directive Interface

## 2.3 DESIGN GOALS AND CONSTRAINTS

### 2.3.1 COMPATIBILITY WITH SDL SERVICES

The SDLS standards (references [1] and [2]) have been developed for use with the existing CCSDS Telemetry (TM), Telecommand (TC), Advanced Orbital Systems (AOS), and Unified Space Data Link Protocol (USLP) Space Data Link Protocols defined in references [4], [5], [6], and [7].

As depicted in figure 2-2, SDLS Extended Procedures specify a separate logical interface for managing the security of the space data link. They neither replace nor modify the behavior of the SDLS traffic encryption and authentication services defined in reference [1]. EP commands are, in essence, transmitted alongside user data within the SDL SDU.

### 2.3.2 REQUIREMENTS

SDLS Extended Procedures are designed to operate in a master-slave configuration. For nominal ground-to-space and space-to-ground links, the master is the mission operations center. It is also possible to use the Extended Procedures for managing space-to-space links that operate in a master-slave configuration. In a space-to-space configuration one of the endpoints needs to be designated as the master.

### 2.3.3 PROTOCOL DATA UNITS

#### 2.3.3.1 Use of Packet Service

All SDLS Extended Procedures directives (both Command and Reply PDUs) are transmitted using the Packet Service (VC Packet in AOS and TM, or Multiplexer Access Point [MAP] Packet in TC and USLP) of each supported CCSDS Space Link Protocol. Figure 2-3 depicts the insertion and extraction of EP PDUs by the supporting Space Link Protocols.

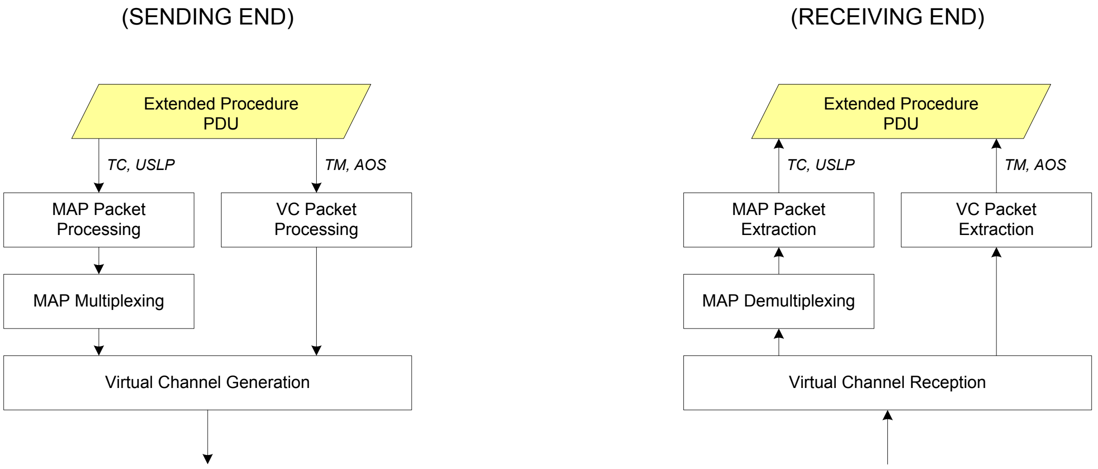


**Figure 2-3: EP PDU Relation to Data Link Layer Processing**

Specification of the Packet Service for the transmission of SDLS Extended Procedures directives does not mandate the design for the PDU interface implementation. Figure 2-2 depicts the SDLS EP Command and Reply interface as directly attached to forward and return link processing. This could illustrate a potential architecture for embedding a security function in the onboard baseband signal processor (e.g., as an integrated hardware unit or a software-defined radio).

As depicted in figure 2-4, it is possible (and certain missions may prefer it) to route SDLS EP packets through the onboard computer's packet processing function for simplicity of implementation and validation.

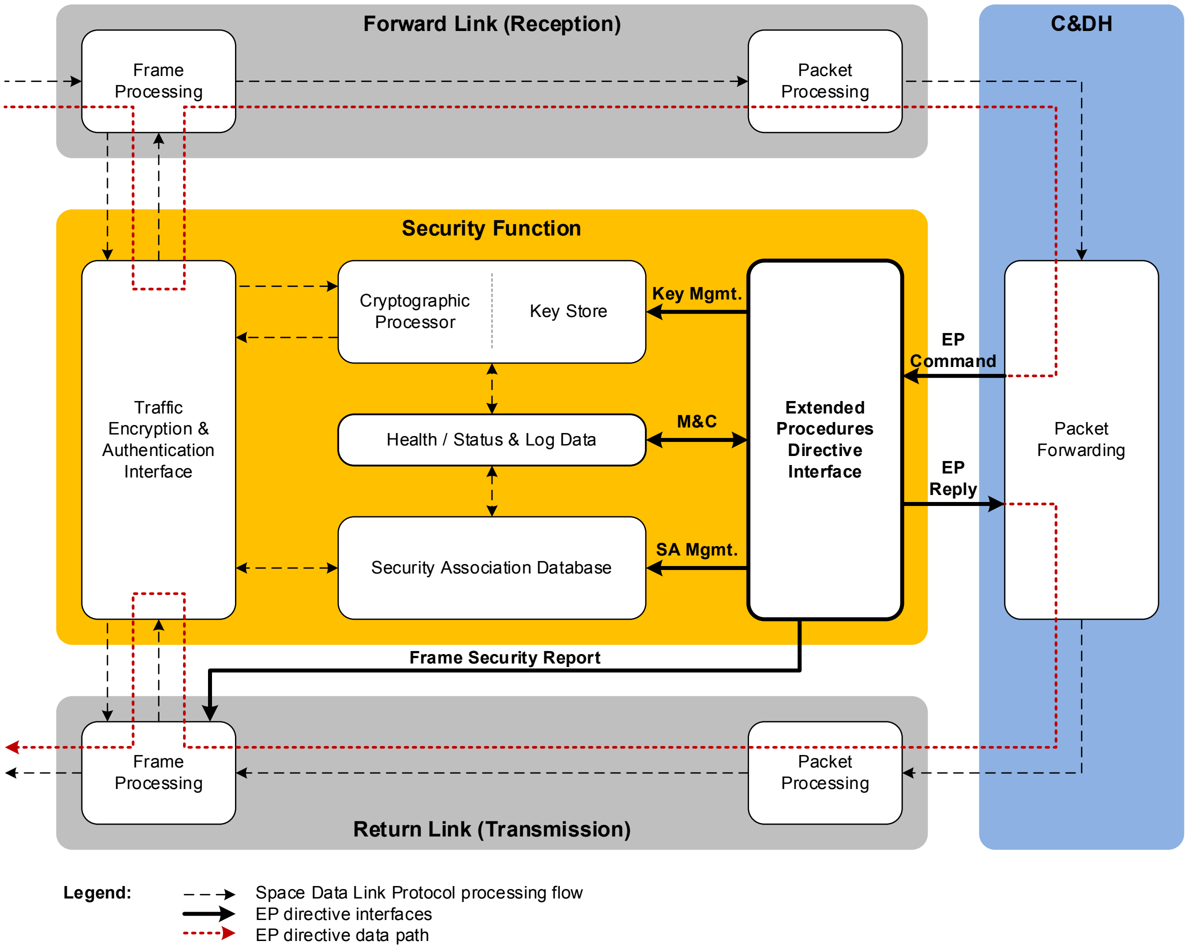


**Figure 2-4: EP PDU Interface via Onboard Computer**

### 2.3.3.2 Delivery of Protocol Data Units

SDLS EP PDU exchanges do not contain any mechanisms for assuring reliable delivery. Directives lost in transmission will go undetected unless the Initiator receives telemetry reporting from the Recipient of EP Command PDUs as they are received and executed.

Telemetry acknowledgement of EP Command PDUs received by the Recipient is required to maintain the integrity of SDLS, but its implementation is mission specific.

#### 2.3.3.3 Protection of Protocol Data Units

Because SDLS Extended Procedures provide an in-line capability to modify operational attributes of the onboard security function, it is necessary to protect against the insertion of unauthorized Command PDUs. All EP PDUs are transmitted over an SDLS channel protected by authentication or authenticated encryption. Specification of the VC to be used for SDLS control and reporting is done as a managed parameter. The implementer may choose either to encrypt and authenticate the SDLS VC, or only authenticate.

While a few directives include authentication and/or encryption within the PDU (e.g., OTAR, Key Verification), all EP PDU channels should be equally protected across all EP PDU exchanges. (See the note in 4.4.1.2 related to Initialization Vector usage for Key Verification.) The SDLS Extended Procedures Blue Book (reference [2]) identifies 'reserved' Security Associations available for securing EP PDUs.

The decision by a mission to implement authenticated encryption versus authentication-only for the transmission of EP Service PDUs should be based upon an overall analysis of threats and risks to the mission. Certain PDUs, for example, Key Inventory or Dump Log Reply PDUs, or Set Anti-Replay Sequence Number (ARSN) Command PDU, could reveal information that should remain private to a hostile third party. Since the same threats are generally applicable to other spacecraft monitoring and control data exchanges, any risk-based decision is likely to be also applicable to the protection of telecommands and/or telemetry.

EP Commands received on board (depicted via the red dashed line in figure 2-4) pass through the SDLS frame encryption/authentication interface before they are routed to the SDLS EP directive interface on board. As a result, they are potentially viewable by other onboard components during this interval.

Based on a security assessment, additional protection of EP PDUs while they are routed on board or across the ground segment may be needed for governmental or agency missions that are deemed to be of strategic value. Reference [13] provides the rationale and methodology behind such an assessment and can help to understand/determine the required level of cryptosecurity needed in a particular case. OTAR and Key Verification directives are protected within their own EP PDUs. Whenever EP PDUs are unprotected outside the confines of an SDLS-protected link, a secure channel is needed at another layer (i.e., network or application). This 'tunneling' approach should also be considered if the Reply PDU path is unprotected (illustrated in figure 2-5).

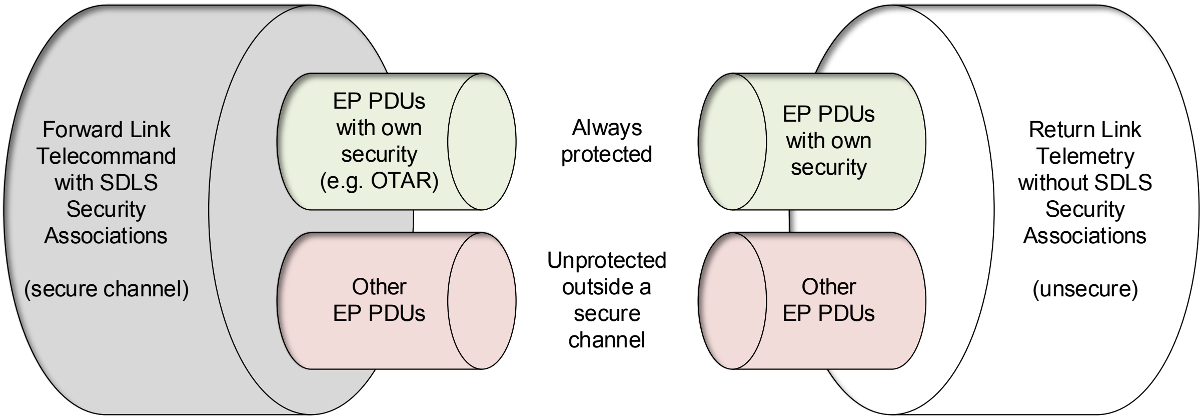


**Figure 2-5: Additional Protection of EP PDUs within Normal Traffic**

## 2.4 KEY MANAGEMENT

### 2.4.1 JUSTIFICATION

CCSDS recommends a standard cryptographic key lifecycle and a set of key management procedures to enable proper generation, distribution, and handling of cryptographic keys for space missions (reference [8]). The SDLS Extended Procedures for key management represents a specific implementation of this recommendation. In addition, CCSDS has produced general guidelines and practices on key management (reference [9]).

### 2.4.2 SUMMARY OF CAPABILITIES

The SDLS security services rely on symmetric cryptosystems in order to operate properly. The following general key management schemes listed in reference [10] are supported by the Extended Procedures:

- Scheme 1: all session keys are pre-loaded on spacecraft before launch and cover the entire mission lifetime.
- Scheme 2: a subset of keys (master keys/Key Encryption Keys [KEKs]) and session keys/traffic encryption keys) are pre-loaded on spacecraft before launch; additional session keys are uploaded in encrypted form during spacecraft operation (OTAR).

## 2.5 SECURITY ASSOCIATION MANAGEMENT

### 2.5.1 GENERAL

The SDLS protocol provides encryption, authentication, or authenticated encryption for data link layer services of the TC, TM, Advanced Orbital Systems AOS, and USLP protocols. Central to the operation of SDLS is the SA, a data schema used at the sending and receiving ends of a space link for managing the session state of cryptographic parameters.

All Transfer Frames that share the same SA on a physical channel constitute a Secure Channel. A Secure Channel consists of one or more Global Virtual Channels (GVCs) or Global MAP IDs (TC and USLP only) assigned to an SA at the time of its creation. Certain EP directives have immediate effects on the state of a secure channel.

### 2.5.2 JUSTIFICATION

The Security Association Management Service for the SDLS protocol is designed to carry out the Security Association setup, activation, status, and control necessary to configure the Security Association parameters of a remote (slave) system's SDLS implementation into an operational state.

### 2.5.3 SUMMARY OF CAPABILITIES

The SA Management Service is designed to support an operational state model that may be either simple or complex as indicated by mission needs. Many missions of ordinary duration with low data rates can be satisfied using statically defined Security Associations and pre-loaded cryptographic keys and algorithms. For these missions, it is sufficient to choose an SA for use on a particular virtual channel along with all of its pre-loaded attributes.

High data rate or long-duration missions may require the capability to reuse and/or reconfigure Security Associations as the SAs and keys loaded into the system prior to the mission are used up over time. As a result, the SA Management Service state model includes optional directives supporting OTAR or instantiation of Security Associations on demand. Figure 2-6 illustrates the state model and related directives for Security Associations.

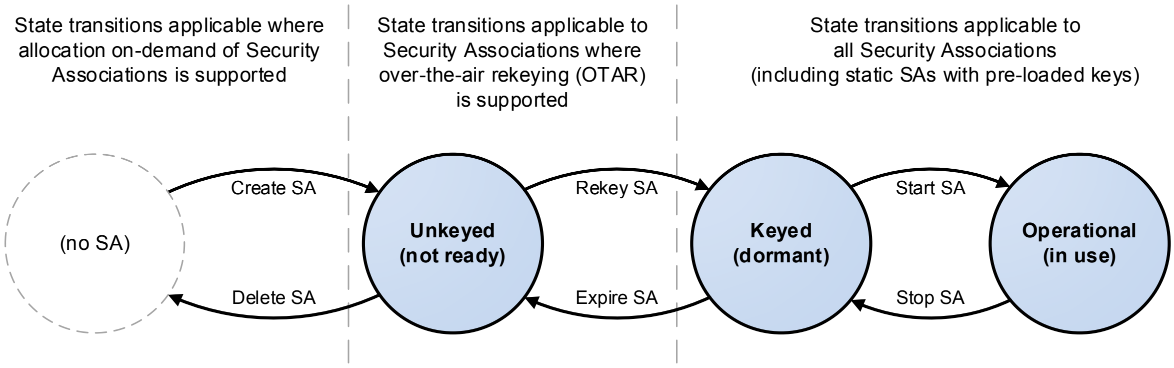

State transitions applicable where allocation on-demand of Security Associations is supported
State transitions applicable to Security Associations where over-the-air rekeying (OTAR) is supported
State transitions applicable to all Security Associations (including static SAs with pre-loaded keys)
(no SA)
Create SA
Delete SA
Unkeyed (not ready)
Rekey SA
Expire SA
Keyed (dormant)
Start SA
Stop SA
Operational (in use)

**Figure 2-6: Variable State Model for Security Associations**

If a mission needs the capability to generate or upload new cryptographic keys (or sets of keys) during the mission lifetime (as in the case of OTAR), it also needs the capability to change individual Security Associations parameters to use new keys rather than any pre-loaded keys.

Although it is not expected to be common practice, long-duration missions may need to replace existing Security Associations. For example, this capability could be used in conjunction with reprogrammable cryptographic systems and redundant security units to retire obsolete algorithms and transition to use newer ones.

## 2.6 MONITORING & CONTROL

### 2.6.1 GENERAL

The SDLS Extended Procedures specify a set of service procedures for the monitoring and control of the slave security functions, which are typically on board. The onboard security unit is an implementation of the SDLS security functions in hardware and/or software. The Monitoring & Control service procedures provide for nominal and contingency scenarios.

### 2.6.2 JUSTIFICATION

The master (typically a mission operations center) needs to know the state of the slave (typically onboard) security unit. This state includes the history of security events, in order to enable the investigation of encountered anomalies and detect potential attacks. These are communicated using the EP Monitoring & Control directives.

### 2.6.3 SUMMARY OF CAPABILITIES

The set of Extended Procedures Monitoring & Control service procedures covers several aspects:

– security unit health monitoring (e.g., Ping, Self-Test);

– management of the log of security events (e.g., Log Status, Dump Log, Erase Log); and

– general-purpose monitoring and control of the onboard security unit (e.g., Alarm Flag Reset).

Several on-demand or on-event reporting mechanisms and corresponding messages are specified in the Extended Procedures. They provide non-real-time or non-systematic reporting of frame verification status at the receiving end of an SDLS-secured forward link. They enable further investigation of security events occurring on board.

## 2.7 FRAME SECURITY REPORT

### 2.7.1 GENERAL

The SDLS Extended Procedures specify a new type of telemetry frame Operational Control Field (OCF), which is used for frame security reporting and is fully compatible with the existing SDL protocols (TM, AOS, and USLP) (references [4], [6], and [7]) for reporting of the onboard security unit status.

### 2.7.2 JUSTIFICATION

This Frame Security Report (FSR), which is the PDU transmitted (in an OCF) from the Recipient to the Initiator of an SDLS secured TC, AOS, or USLP forward link, provides the systematic, real-time mechanism allowing the SDLS function at the receiving end to report the status of forward link TC, AOS, or USLP frame verification to the sending end.

The FSR is similar to the COP-1 Communications Link Control Word (CLCW) (reference [11]), which provides real-time reporting of the status of forward link TC, AOS, or USLP frame acceptance by the COP-1 function to the sending end.

### 2.7.3 SUMMARY OF CAPABILITIES

The reporting capabilities of the FSR are:

- a persistent Alarm Flag that will signal a frame rejection by an onboard SDLS function;

NOTE – This flag can be reset by the user once the rejection has been taken into account by the Initiator (most often the Mission Operation Center [MOC]).

- non-persistent Security Event Flags to characterize security violations detected on the last received frame such as an invalid Sequence Number, an invalid MAC (failed authentication), or an invalid SA;
- a Security Parameter Index (SPI) of the last received frame;
- a Sequence Number (SN) of the last received frame.

The Alarm Flag enables the systematic detection by the Initiator of any forward frame rejection by SDLS. The detection latency is minimized by the transmission of the FSR in the OCF that is carried in every return link frame.

The Security Event Flags and the associated SPI/SN information enable the characterization of major security events occurring on a forward frame provided that the FSR is transmitted to the Initiator for every forward frame received (see 3.5 for discussion of FSR transmission rate).

# 3 CONCEPT OF OPERATION

## 3.1 OVERVIEW

The SDLS Extended Procedures enable optional capabilities for managing data link security and are organized into three functional areas: Key Management, SA Management, and SDLS Monitoring & Control. Figure 3-1 illustrates the complete set of Extended Procedures.

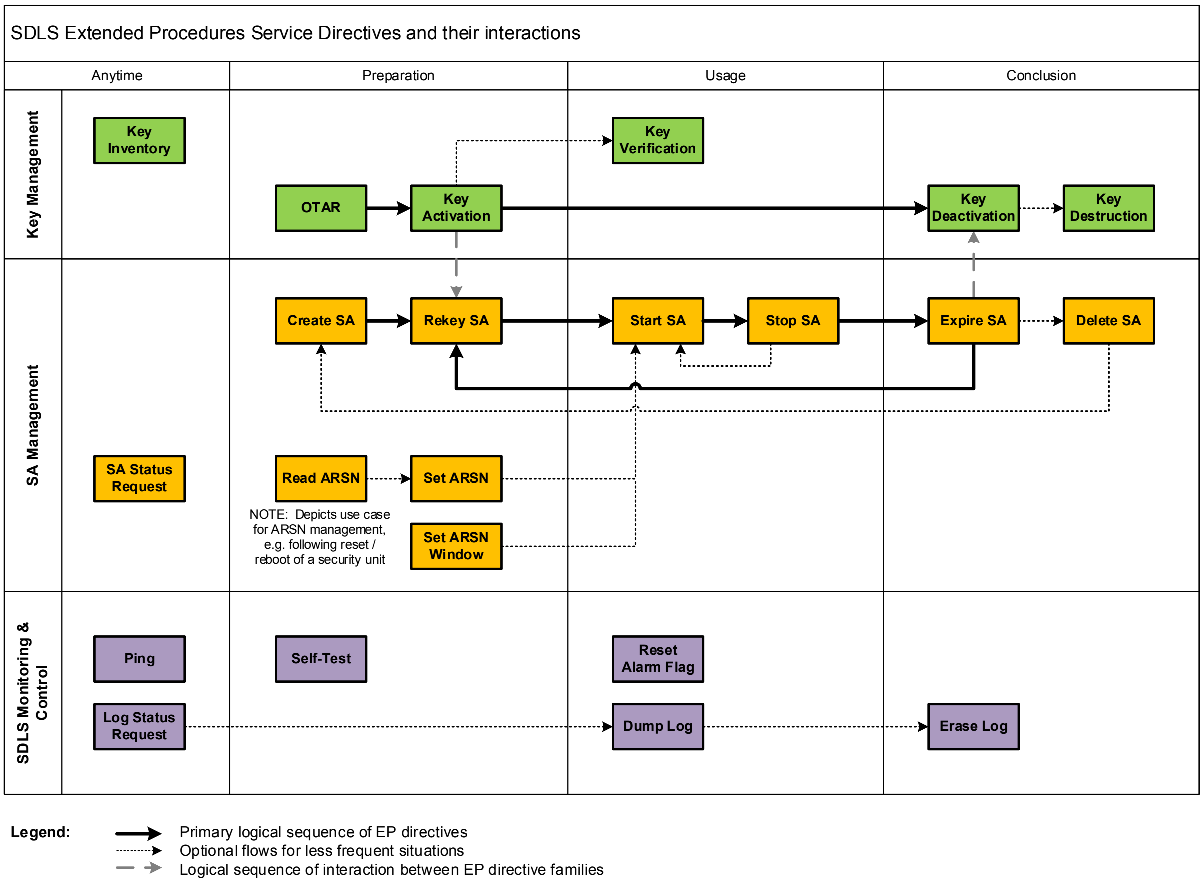

SDLS Extended Procedures Service Directives and their interactions
Anytime
Preparation
Usage
Conclusion
Key Management
SA Management
SDLS Monitoring & Control
Key Inventory
Key Verification
OTAR
Key Activation
Key Deactivation
Key Destruction
Create SA
Rekey SA
Start SA
Stop SA
Expire SA
Delete SA
SA Status Request
Read ARSN
Set ARSN
Set ARSN Window
NOTE: Depicts use case for ARSN management, e.g. following reset / reboot of a security unit
Ping
Self-Test
Reset Alarm Flag
Log Status Request
Dump Log
Erase Log
Legend:
Primary logical sequence of EP directives
Optional flows for less frequent situations
Logical sequence of interaction between EP directive families

**Figure 3-1: Extended Procedure Directives**

It is not necessary to implement the entire set of EP directives. Implementation of Extended Procedures can be tailored to the needs of a mission. However, within each functional area, many of the directives correspond to others and the presence of related directives is logically expected. The Extended Procedures specification (reference [2]) Protocol Implementation Conformance Statement (PICS) provides further detail about which EP directives should be implemented together.

## 3.2 KEY MANAGEMENT

### 3.2.1 OVERVIEW

This subsection describes the concept of operation for the key management SDLS Extended Procedures (reference [2]).

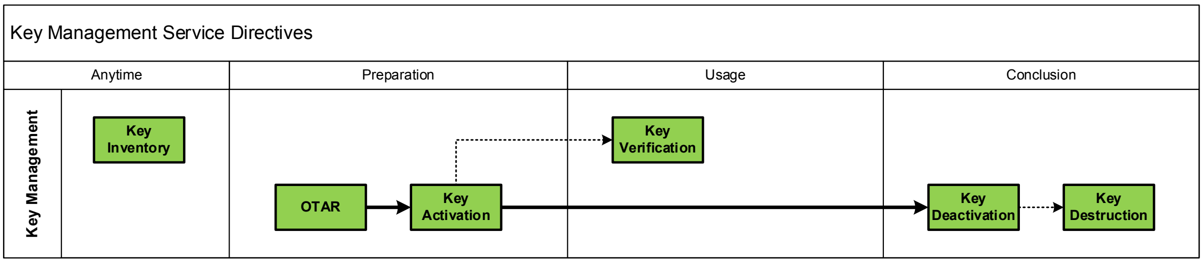


**Figure 3-2: Key Management Directives**

### 3.2.2 CRYPTOGRAPHIC KEY LIFECYCLE

The symmetric cryptographic key is a core component in every cryptographic operation. It represents the shared secret used between communication partners and thus forms the basis for authentication, integrity, and confidentiality services that the communication partners agree to implement.

A cryptographic key is governed by a state-based lifecycle as defined in 3.2 of reference [8]. Key usage is dependent on its state in the key's lifecycle. Key states are applicable system-wide and not only to a single cryptographic module. SDLS Extended Procedures support the key states and transitions from reference [8] depicted in figure 3-3.

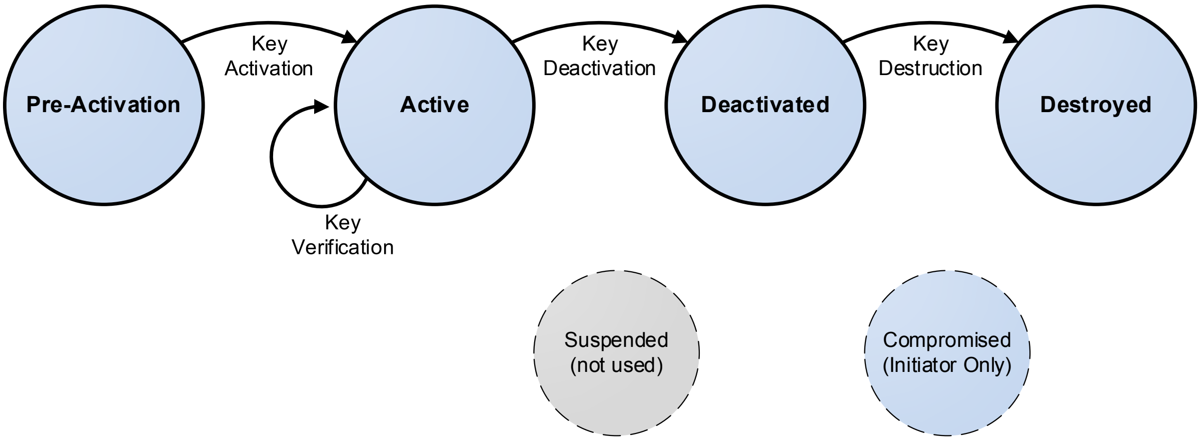


**Figure 3-3: Key States and Transitions**

A detailed description of the key states may be found in the CCSDS Symmetric Key Management Recommended Practice (reference [8]). The SDLS Key Management Extended Procedures do not support:

- the optional Suspended state: a key suspension does not represent a credible operational scenario for SDLS;
- the Compromised state: the Compromised state applies only to the Initiator for SDLS.

### 3.2.3 KEY IDENTIFIER

The Key Identifier (Key ID) provides the unique and abstract identification of a key. In the SDLS Extended Procedures, the Key ID is an integer number. An implementation of the SDLS Extended Procedures maps the Key ID to the location of the cryptographic key (e.g., in onboard memory). It is important to note that as a consequence of a key management operation, a Key ID could be reassigned to a different key (e.g., in the case of a newly uploaded key overwriting an old key with the same Key ID).

If a two-tier key hierarchy of master and session keys is being used (see reference [8]), the master and session keys share the same Key ID range; that is, there is only one Key ID type. The partitioning of the Key ID range into Key IDs for master keys and session keys is implementation specific. If partitioning is done, it provides a potential safeguard against the inadvertent selection of master keys for use as session keys.

### 3.2.4 PROCEDURES IMPLEMENTING LIFECYCLE TRANSITIONS

#### 3.2.4.1 Overview

This subsection discusses the subset of the key management-related Extended Procedures that are directly related to state transitions in the key lifecycle. These procedures are:

- Key Activation;
- Key Deactivation;
- Key Destruction.

#### 3.2.4.2 Key Activation

The Key Activation procedure implements the transition of one or more cryptographic keys from pre-activation to active state. This transition is a prerequisite for the operational use of a key.

The only Key Activation parameter is the set of Key IDs that indicate to the Recipient which cryptographic keys should transition from pre-active to active state. The Key Activation commanding directive is an atomic transaction; that is, either all referenced keys are successfully transitioned, or none are transitioned.

#### 3.2.4.3 Key Deactivation

The Key Deactivation procedure implements the transition of one or more cryptographic keys from Active to Deactivated state. Keys that undergo this transition cannot be used for cryptographic operations on new data but remain physically present at the Initiator and Recipient. This provides the means to decrypt legacy data that has been encrypted with deactivated keys.

The only Key Deactivation parameter is the set of Key IDs that indicate to the Recipient which keys should transition from Active to Deactivated state. The Key Deactivation commanding directive is an atomic transaction: that is, either all referenced keys are successfully transitioned, or none are transitioned.

#### 3.2.4.4 Key Destruction

The Key Destruction procedure implements the transition of one or more cryptographic keys from the Deactivated to Destroyed state. As a result of this transition, the cryptographic keys on the Recipient side are physically removed/deleted or overwritten with random data. The Initiator may keep a copy of the key for legacy purposes. However, the key is no longer part of the active SDLS Extended Procedures implementation. The Key ID of any destroyed key is considered available and can be reassigned to a new key.

The only Key Destruction parameter is the set of Key IDs that indicate to the Recipient which cryptographic keys should transition from deactivated to destroyed state. It should be noted that the Key Destruction commanding directive is an atomic transaction; that is, either all referenced keys are successfully destroyed, or none are transitioned.

### 3.2.5 KEY RENEWAL SCHEMES

#### 3.2.5.1 Overview

As described in 2.4, the SDLS Extended Procedures support two main key renewal schemes:

- Scheme 1: no key regeneration during the lifetime of the mission;
- Scheme 2: OTAR.

The decision of which scheme will be implemented by a mission is based on a risk assessment and trade-off analysis. The CCSDS Security Guide for Mission Planners (reference [14]) provides background information on performing such risk assessments.

#### 3.2.5.2 Scheme 1: No Key Regeneration during the Lifetime of the Mission

In this scheme, all cryptographic keys that will be used during the lifetime of the mission are stored at the Initiator and Recipient side (in pre-activation state) before the mission becomes operational. No refresh or replacement of keys is foreseen during the lifetime of the mission. As a consequence, any key that reaches the deactivated or destroyed state will not be available for the remainder of the mission lifetime. Any key that is corrupted, or suspected to be corrupted, cannot be replaced with a new key.

For many missions, this scheme is attractive because of its operational simplicity. Any mission implementing scheme 1 will not need to implement OTAR. However, it needs to be noted that this scheme comes with a significant risk. Any weakness that is discovered with keys after launch (up to and including corruption or compromise) can no longer be mitigated.

The key hierarchy under this scheme is typically flat; that is, all cryptographic keys are session keys (see reference [8]).

#### 3.2.5.3 Scheme 2: Over-the-Air-Rekeying

The OTAR scheme provides the ability to transmit newly generated keys from the Initiator to the Recipient during mission operations. OTAR requires a two-tier key hierarchy composed of master keys at the higher level and session keys at the lower level. Both key types and the hierarchy concept are described in reference [8].

The OTAR concept involves the secure uploading of newly generated session keys from the Initiator to the Recipient using a master key as the shared secret. The OTAR procedure allows multiple session keys to be uploaded at the same time. The maximum number of keys possible for each OTAR command directive is limited only by the packet size of the Space Packet.

The OTAR procedure assumes that the Initiator has the ability to securely generate any number of session keys. The key generation procedures are outside the scope of the SDLS Extended Procedures. The availability of the number of session keys to send to the Recipient is one of the preconditions for the OTAR procedure. Another precondition is the availability of an active master key to use in the authenticated encryption of the session keys so they are protected during transmission.

Each uploaded session key is accompanied by a Key ID. This is necessary for the Recipient to know which Key ID needs to be assigned to a newly updated key. As previously mentioned, it is possible that a newly uploaded session key may replace a destroyed key reusing the Key ID. The (session key, Key ID) pairs are authenticated and encrypted using a master key (identified by the Master Key ID). Depending on the cryptographic algorithm being used (e.g., AES-GCM in baseline mode), an Initialization Vector (IV) may need to be transmitted as part of the directive.

When an OTAR directive is received on the Recipient side, the (session key, Key ID) pairs are first authenticated and decrypted, and then the individual new session keys are stored in pre-activation state in the memory slots identified by their Key ID.

The OTAR Command PDU always contains a Message Authentication Code (MAC). If transmitted over a communication channel using an SDLS Authenticated SA, a MAC would also be present at the transfer frame (data link) layer. While this ‘double’ authentication appears redundant, it is necessary because the two occur at different layers (as depicted in figure 2-5) and could (depending on the implementation) be added or removed at different points along the end-to-end path. If, as shown in figure 2-4, EP Command PDUs are routed through the onboard computer, then any transfer frame MAC would be removed before the Command PDU arrived at the Extended Procedures directive interface. It should be noted that the OTAR PDU contents would lack integrity protection during its onboard routing and verification processing within the security unit. Similarly, integrity and confidentiality protection is needed for the OTAR PDU during its routing on the ground in the mission operations center.

While OTAR is expected to be executed infrequently, its inherent complexity is expected to significantly drive the design of the onboard SDLS EP security function and its interface with the communications and data handling functions. It may also impact flow control, latency, and throughput of the space link. Care should be taken to balance design complexity, flow control, performance impact, and operations concept.

### 3.2.6 PROCEDURES FOR CONFIRMING KEY INFORMATION

#### 3.2.6.1 Overview

This subsection discusses the subset of the key management-related Extended Procedures for maintaining the accuracy of the remote Recipient’s key database. These procedures are:

- Key Verification;
- Key Inventory.

#### 3.2.6.2 Key Verification

The Extended Procedures Key Verification mechanism provides the means to verify one or more keys stored at the Recipient side. In space missions, this is important because the space environment may cause bit flips or other unwanted modifications of keys stored on board the spacecraft. Therefore it is imperative to verify a key before using it for the first time.

The Key Verification directive uses a challenge-response approach to verify keys stored at the Recipient. A random number is generated and sent to the Recipient with the respective Key ID for each key to be verified. Upon reception, the Recipient will encrypt the random number using the key designated by the Key ID and the IV, and send the result back to the Initiator.

NOTE – The IV has to be carefully selected taking into account the constraints of the cryptographic algorithm selected. Further discussion of this topic can be found in 4.4.1.2 of this document as well as 3.4.2.3 of reference [10].

The Initiator uses its copy of the respective keys to perform the same encryption operation and compares the results. A match means that the Initiator and Recipient keys are identical. A mismatch indicates a problem that would require investigation by the operator. In order to minimize session key exposure, it is recommended to verify a session key immediately prior to its use for cryptographic operations.

NOTE – The Key Verification procedure necessitates over-the-air use of the specified key(s). For this reason, it is carried out only on keys already in the Active state, as shown in figure 3-2. A mission could, if desired, explicitly tie the Key Activation and Key Verification procedures together such that a single spacecraft command would result in both Command PDUs being issued in sequence to the onboard security function. It should be noted that Key Verification will imply the start of the verified key's cryptoperiod.

#### 3.2.6.3 Key Inventory

In missions in which SDLS EP are used for performing remote key management tasks, it is useful for the Initiator to obtain key status information from the Recipient to compare the status against its own local key database. If the Recipient end is out of synchronization with the Initiator, that is, the Recipient key states do not match the ground's expectation, it needs to prompt the issuance of EP Command PDUs to direct the onboard end into the desired state and correct the mismatch.

The Key Inventory directive is used to query the Recipient for its local key state information for a range of Key IDs. The Initiator provides a numerical range of Key IDs. The Recipient replies with a list of Key ID-and-state pairs corresponding to that range. The returned pairs indicate one of the SDLS EP-supported key states (e.g., pre-activation, active, deactivated, destroyed) for each Key ID in the specified range. Nonexistent Key IDs within the range are omitted from the reply. The values corresponding to each defined key state are mission-specific metadata. Details of how keys are stored in onboard memory are mission specific and beyond the scope of this document.

### 3.2.7 KEY MANAGEMENT CONCEPT OF OPERATIONS

#### 3.2.7.1 Interaction between Key Management and SA Management

SDLS Security Associations are dependent upon the existence of cryptographic keys eligible for operational use. Determining which keys are eligible for operational use is the task of key management.

Because it is not mandatory to implement the entire set of SDLS EP directives, there are directives in the Key Management and Security Association Management service groups that are logically related, yet separate because they carry out distinct functions. When the two service groups are implemented together, there is a sequential relationship between them as depicted in figure 3-1.

A precondition of the Rekey SA directive is that its specified key is in the Active state. The Key Activation directive enables a key for operational use, and therefore it would be expected to precede any Rekey SA directive to associate the same key with a SA. Likewise, the Key Deactivation directive disables a key for operational use and would therefore be expected to follow the Expire SA directive. A mission could, if desired, explicitly tie these operations together in implementation such that a single spacecraft command resulted in both Command PDUs being issued in sequence to the onboard security function.

#### 3.2.7.2 Use of Master Keys

Master and traffic (session) keys may be indistinguishable from one another in terms of key format, but each serves a distinct purpose. To avoid potential compromise, keys of one type should not be used in place of another type. SDLS Extended Procedures only use master keys for the OTAR procedure. SDLS Security Associations use keys only for traffic encryption and/or authentication.

The loss or corruption of a master key is a serious occurrence. It is catastrophic if OTAR cannot be used to upload new keys, including a new master key, because there is no usable master key currently on board. To protect against this contingency, it is recommended that several master keys are installed on board in advance.

## 3.3 SECURITY ASSOCIATION MANAGEMENT

### 3.3.1 OVERVIEW

This subsection describes the concept of operations for the SDLS Extended Procedures (reference [2]) Security Association management.

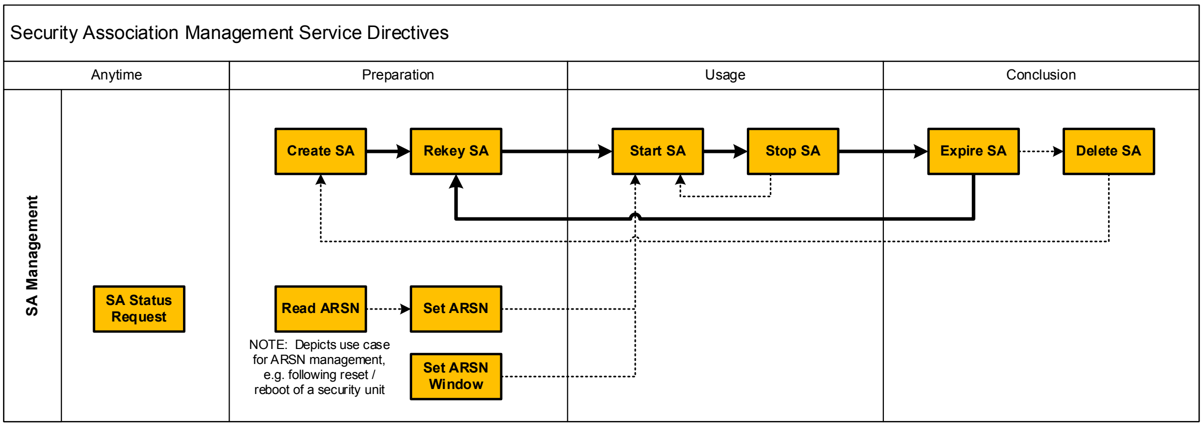


**Figure 3-4: SA Management Directives**

### 3.3.2 GUIDELINES ON PLANNING AND ASSIGNING SECURITY ASSOCIATIONS

#### 3.3.2.1 SAs for Nominal Traffic

The SPI is a 2-byte field in the Security Header. Therefore the number of SAs per Master Channel has an upper bound of $2^{16}$ (65,536). The actual number of SAs that an implementation needs to assign and prepare for use is notionally equivalent to the number of keys that the spacecraft's security unit is capable of storing simultaneously.

Specific ranges of SPI values may be assigned for operational convenience to mission-specific operational use cases when there are use cases whose traffic protection requirements are not interchangeable (e.g., subdivision into SAs used for spacecraft housekeeping and SAs used for private payload data, or SAs used for testing in 'clear mode').

#### 3.3.2.2 SAs for SDLS EP Traffic and Other Special Uses

SDLS EP traffic can be carried over the same SAs used by normal traffic and routed by normal VC or MAP packet processing to the correct remote security unit for PDU processing. However, it is critical that SDLS EP directives never modify the SA currently being used to transmit their own EP PDUs. Doing so could interrupt the processing of EP PDUs in the middle of a sequence of EP operations or cause loss of cryptographic synchronization between sender and receiver.

Alternatively, SDLS EP traffic can be carried over different SAs not used by normal traffic. In the SDLS protocol specification (reference [1]), two SPI values were reserved for future use. These values (0 and 65535) were intentionally reserved to remain available for special SDLS management use cases. VC or MAP packet processing is still necessary for routing EP traffic to the security unit for PDU processing. The baseline mode of SDLS EP (reference [2], annex D) mandates the use of the two reserved SPIs for exchanging EP services PDUs over the space link.

### 3.3.3 NORMAL PROCEDURES FOR SA MANAGEMENT

#### 3.3.3.1 Instantiating an SA

Many SA service parameters are managed. Each SA needs to specify the values of these parameters, whether implicitly via pre-loaded static definition or explicitly via EP directive (Create SA). This information is collectively known as the SA database, although that term does not imply a Relational Database Management System (RDBMS)-type implementation.

The following SA parameters are fixed at the time of creation and do not change thereafter:

a) SPI;

b) SA Service Type;

c) the field lengths for Security Header and Security Trailer fields;

d) encryption cipher suite length and identifier;

e) IV length;

f) authentication cipher suite length and identifier;

g) authentication bit mask length and value;

h) ARSN length; and

i) ARSN Window length.

The following SA parameters change during use but need to be provided with initial values:

j) IV initial value;

k) ARSN initial value; and

l) ARSN Window value.

Static pre-loading initializes all of the above managed parameters. If implemented, the Create SA directive accomplishes the same function. The Create SA directive instantiates a new SA in the unkeyed state containing initial parameters and context supplied in the directive. The authentication bit mask needs to be tailored to the particular Space Link Protocol to which it will be applied. For further considerations, refer to 3.2.5 of reference [10].

Static pre-loading also commonly associates cryptographic keys with SAs. Since the Create SA directive does not associate cryptographic keys with the SA, the Create SA directive should be followed by the Rekey SA directive to transition from Unkeyed to Keyed state so that the SA is ready for activation via the Start SA directive.

To replace a Security Association, the two EP directives Delete SA and Create SA are needed as depicted in figure 2-6. The Delete SA directive erases all existing parameters of the SA and its state information so that the specified Security Parameter Index no longer references any defined SA. The Create SA directive is used to instantiate a new SA that reuses the SPI previously belonging to the deleted SA.

### 3.3.3.2 Changing Cryptographic Keys Associated with a SA

If a mission requires the capability to generate or upload new cryptographic keys (or sets of keys) during the mission lifetime, it needs the capability to change individual Security Associations parameters to use new keys.

As depicted in figure 2-6, to associate a new key with a Security Association, the two EP directives Expire SA and Rekey SA are used. The SA’s existing key is removed from the SA via the Expire SA directive. This transitions the SA from Keyed state into Unkeyed state.

The new key is associated with the SA via the Rekey SA directive, which transitions the SA from Unkeyed state to Keyed state.

#### 3.3.3.3 Switching between SAs on a Channel

As depicted in figure 2-6, the most basic operation for Security Associations is to change which SA is used on a channel.

This is carried out via the two EP directives 'Stop SA' and 'Start SA'. The Stop SA directive transitions the current ('old') SA from its Operational (in use) state into the Keyed (dormant) state. In this state, the secure channel is stopped and further communications through the associated GVC/MAP IDs are blocked. Data is rejected by the security function, unless or until another SA is in operation.

The Start SA directive transitions the 'new' SA from the Keyed state to the Operational state. It is expected that most implementations will carry out key changes during normal operation by iterating through a set of SAs configured in advance, as depicted in figure 3-5. As illustrated, the applicable channel (GVC Identifier [GVCID] or Global Multiplexer Access Point Identifier [GMAPID]) remains constant, while the SA used on the channel is replaced at every key change event.

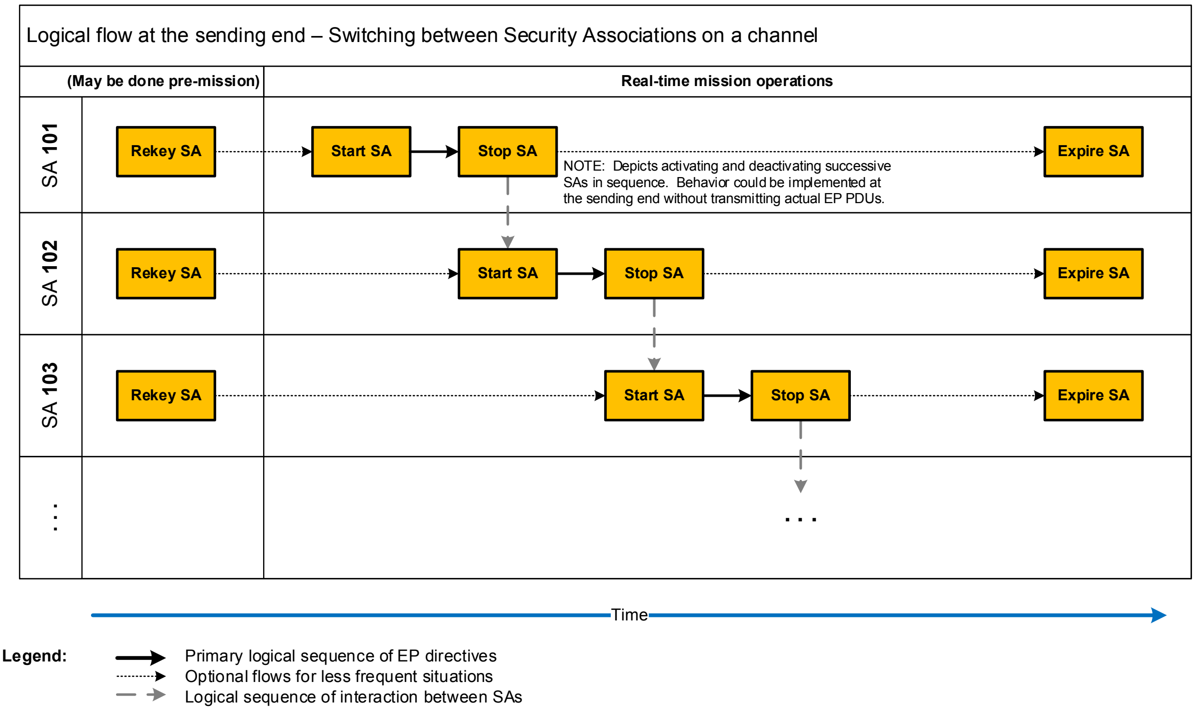
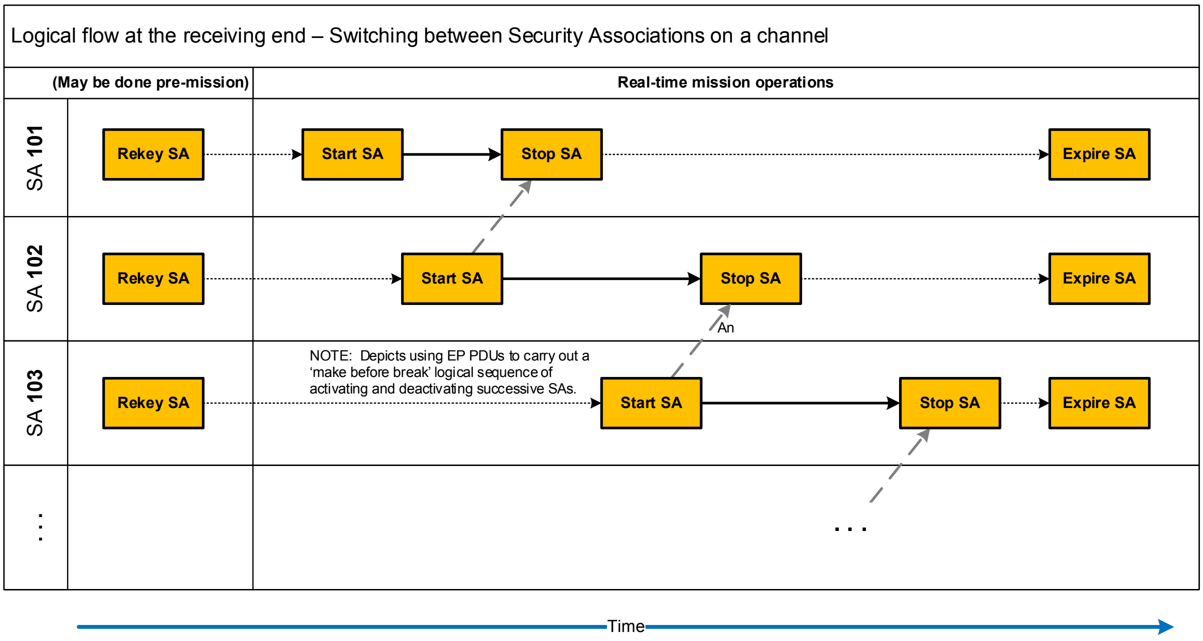

Logical flow at the sending end – Switching between Security Associations on a channel
(May be done pre-mission)
Real-time mission operations
SA 101
SA 102
SA 103
. . .
Rekey SA
Start SA
Stop SA
Expire SA
NOTE: Depicts activating and deactivating successive SAs in sequence. Behavior could be implemented at the sending end without transmitting actual EP PDUs.
Time
Legend:
Primary logical sequence of EP directives
Optional flows for less frequent situations
Logical sequence of interaction between SAs

**Figure 3-5: Operational Key Change Scenario (Sending End)**

### 3.3.3.4 Seamless Key Change

Depending on the capabilities of the security units at the sending and receiving ends, it is possible for the sending end to change which SA is used on a channel (and which key is in effect) from one frame to the next, without the receiving end dropping frames during the transition.

If frame-upon-frame key change is to be supported, both the sending end's and receiving end's security units should be capable of handling more than one active cryptographic session and key simultaneously. The receiving end's security unit should be capable of supporting more than one SA in the Operational state on a given VC or MAP. When newly arrived frames indicate SAs different from previous frames, the security unit can correctly process without delay in transition. The 'new' SA should be transitioned into the Operational state at the Recipient end before the Initiator starts sending frames using the 'new' SA.


Logical flow at the receiving end – Switching between Security Associations on a channel
(May be done pre-mission)
Real-time mission operations
SA 101
Rekey SA
Start SA
Stop SA
Expire SA
SA 102
Rekey SA
Start SA
Stop SA
Expire SA
An
SA 103
Rekey SA
Start SA
Stop SA
Expire SA
. . .
. . .
Time

NOTE: Depicts using EP PDUs to carry out a 'make before break' logical sequence of activating and deactivating successive SAs.

**Figure 3-6: Seamless Key Change Scenario (Receiving End)**

### 3.3.3.5 Query and Modify SA Parameters

EP directives are provided to allow mission operations staff to adjust SA parameters in response to observed performance or unexpected behavior on the space link.

The SA Status Request directive queries the Recipient to report the current state of a specified SA. The SA Status Request Reply PDU returns the Procedure ID of the last executed state transition directive (e.g., Start SA, Stop SA, Expire SA) for the requested Security Association. The defined Procedure ID values ordinarily returned by this directive, as illustrated in figure 5-10 of reference [2], implicitly embed both the previous ('from') and current ('to') states during the last state transition for the applicable SA. In the case of a mission that initializes SAs through static pre-loading prior to the mission, the 'last state transition' for SAs that have not yet received EP directives is undefined unless it is initialized to a default state after power up.

SDLS SAs providing Authentication service protect against 'replay attacks' (the potential for an unauthorized party to record and retransmit previously transmitted frames such as commands to a spacecraft) by making use of a transmitted sequence counter and a managed 'window' indicating how close a sequence number has to be to its expected value to be accepted as valid. The Set ARSN and ARSN Window directives are used to adjust the ARSN and ARSN Window, respectively. In case of synchronization loss, or when switching to a previously used SA, it may be necessary to use the Read ARSN directive to obtain the stored onboard value of the Sequence Number.

### 3.3.4 CONTINGENCY AND OFF-NOMINAL SCENARIOS

#### 3.3.4.1 Recovery SA in Telecommand

A SA can only cover a single VC of the TC Space Link Protocol. However, SDLS does not exclude the duplication of SAs over a given Telecommand VC. Experience acquired with previous ad hoc implementations of security functions for the protection of Telecommand has shown that the existence of a redundant SA, only to be called as a last resource, could be very beneficial. When the 'nominal' SA has failed and possibly left the spacecraft telecommanding unavailable, this 'redundant' SA will allow restoration of telecommanding without jeopardizing security. This special SA is labelled *Recovery SA*.

Special care should be taken to store and segregate the context of this SA at both ends of the space link. This Recovery SA should not be used for regular operations. Preferably the onboard keys associated with this Recovery SA should be neither erasable nor reloadable nor revocable, in order to maximize operational safety. For the same safety reason, the set of EP directives authorized for this Recovery SA can be limited.

## 3.4 MONITORING & CONTROL

### 3.4.1 OVERVIEW

This subsection is outlining the concept of operations for the SDLS Monitoring & Control part of the SDLS Extended Procedures (reference [2]).

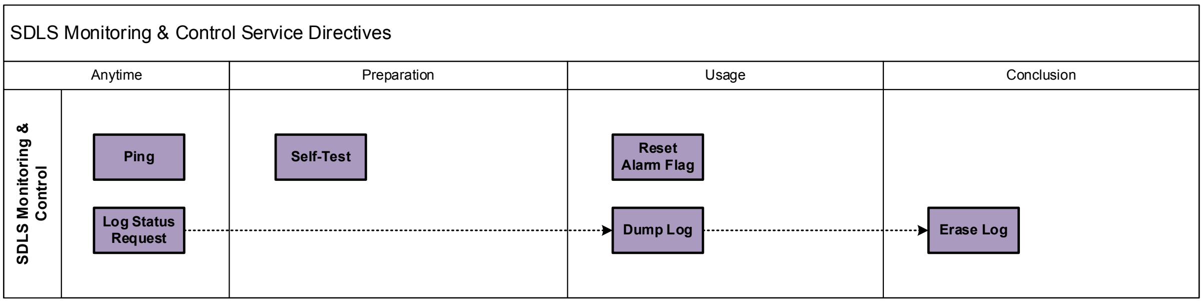


**Figure 3-7: SDLS Monitoring & Control Directives**

## 3.4.2 MONITORING & CONTROL PROCEDURES

### 3.4.2.1 General

The Extended Procedures define the following Monitoring & Control procedures:

- Ping;
- Log Status;
- Dump Log;
- Erase Log;
- Self-Test;
- Alarm Flag Reset.

### 3.4.2.2 Ping

The Ping procedure is a simple way to test that the onboard security unit is alive and able to process EP directives. The Ping procedure also provides a simple test of the uplink and the downlink.

Upon reception of a Ping command, the onboard security unit generates a reply and sends it to the ground. Neither the Ping command nor the Ping reply transmits a parameter.

### 3.4.2.3 Log Status

The Log Status procedure is related to the management of the security log on the Recipient side.

The security log contains a set of security event messages. The format of such messages is implementation specific. In reply to the Log Status command, the onboard security unit generates a PDU containing the number of security event messages stored in the log and the remaining amount of space left in the log for storing new security event messages. The remaining space can be expressed as a value in octets or a percentage of the total log space available (the choice of which is left to the implementer).

The Log Status procedure is used by the mission operations center to monitor the usage level of the security log and take appropriate measures (i.e., Dump Log, then Erase Log).

(See 3.4.3 for more information on the security log.)

#### 3.4.2.4 Dump Log

The Dump Log procedure is related to the management of the security log on the Recipient side.

When necessary, the Initiator can use the Dump Log procedure to order the Recipient security unit to send the complete security log to ground. The Dump Log procedure does not affect the contents of the security log, which remains unchanged.

It should be noted that the security log may contain sensitive information.

(See 3.4.3 for more information on the security log.)

#### 3.4.2.5 Erase Log

The Erase Log procedure is related to the management of the security log on the Recipient side.

When the mission operations center has successfully received the security log (by using the Dump Log procedure), it can erase the onboard security log with the Erase Log procedure. Upon reception of the Erase Log command, the onboard security unit erases all the content of the security log, freeing the memory for new security event messages. It then replies to the command with a PDU containing the number of security event messages stored in the log (zero unless new events have occurred) and the remaining amount of space in the log.

(See 3.4.3 for more information on the security log.)

#### 3.4.2.6 Self-Test

The Self-Test procedure is used by the mission operations center to verify the health state of the onboard security unit.

(See 3.4.4 for more information on the Self-Test.)

#### 3.4.2.7 Alarm Flag Reset

The Alarm Flag Reset procedure is associated with the FSR management.

The FSR contains a persistent Alarm Flag that indicates that at least one forward link Transfer Frame has been rejected by the onboard SDLS function since the last reset of the Alarm Flag. When the Alarm Flag has been taken into account by the mission operations center, the Alarm Flag Reset command is sent to the onboard security unit to order it to reset the Alarm Flag.

(See 3.5.3 for more information on the Alarm Flag.)

### 3.4.3 SECURITY LOG

The security log is a means for recording important events seen by the onboard security unit. These events may affect the security of the protected links and are called security events. They are generated by the onboard security unit, either on its own or in reply to a received command. They can be generated when an error occurred or simply to log an important routine event. Such events could be but are not limited to:

- Frame received with a bad Sequence Number value (replay attack);
- Frame received generating a MAC error;
- Frame received pointing to an inactive SA (bad SPI);
- Sequence Number in use reaching its maximum value (need to change the key);
- Key corrupted;
- New key uploaded (OTAR monitoring);
- SA creation or deletion.

Each Security Event Message provides, in addition to the type of event encountered, the minimum information necessary for forensic investigation (SA/SPI used, Virtual Channel Identifier [VCID], Sequence Number, time-tag, …).

As the onboard memory is limited, the security log may reach its maximum length. This length is implementation-dependent. When the security log is full, the onboard security unit can deal with new Security Events Messages in two different ways:

- New Security Event Messages are stored, and the oldest are lost; or
- New Security Messages are lost, and the oldest are kept in the log.

The behavior is not defined by CCSDS; the choice is left to the implementer but should be documented.

The list of security events is not defined by CCSDS and is left to the implementer. It should also be documented. Simple onboard security units may have very simple security log or no log at all. Complex ones may have more verbose security logs.

In reply to a Dump Log command, the onboard security unit sends the complete set of Security Event Messages stored in the log. The format of a Security Event Message is implementation specific, but it is transmitted via the Dump Log command in a Tag, Length, Value (TLV) format (hence allowing for future events definition while maintaining compatibility). However, the precise definition of the T LV fields is left to the implementer.

### 3.4.4 SELF-TEST

The Self-Test Command is intended to initiate a series of predefined tests on the on-board security unit. These tests are not defined by CCSDS and left to the implementer's choice. They are intended to cover the overall security unit's functionality and give confidence that the security unit is alive and performs well.

For example, the Self-Test for a security unit providing authenticated encryption on a TM link could be to compute an authenticated encrypted TM frame using a set of predefined test frame, Key, and ARSN value. The result is then compared with a reference frame. If the computed frame and the reference frame are the same, the test passes. If not, the test fails.

Another complementary test could be a key database test, for example, by associating a Cyclic Redundancy Check (CRC) with each key and storing it with the key in the key database. The key database test consists of computing each CRC sequentially and comparing it with stored CRC.

NOTE – The CRCs never go out of the security unit, as they give information on the key values. Generally speaking, no result of a computation involving a key (except the result of a crypto algorithm) can go outside the security unit as this would be a security breach.

The Self-Test Command PDU is self-contained and does not allow passing of any parameter to the security unit. The Self-Test Reply PDU, however, has seven bits left in the Self-Test Result field that are not defined by CCSDS and may be used by the implementer to give more information on the test result.

The onboard security unit may or may not continue to process normal traffic while performing a Self-Test. This is not defined by CCSDS and is left to the implementer's choice.

## 3.5 FRAME SECURITY REPORT

### 3.5.1 GENERAL

The FSR is a real-time report of the onboard SDLS function at the receiving end of a TC, AOS, or USLP forward link. FSR is transmitted over a TM, AOS, or USLP return link in the transfer frames' OCF using the Master Channel OCF (MC_OCF) or Virtual Channel OCF (VC_OCF) services as defined in references [4], [6], and [7].

When operating a SDLS secured forward link, the sending end (MOC) needs to detect as promptly as possible major security events occurring on board the spacecraft at the receiving end that result in TC, AOS, or USLP frames verification failure. Forward frame rejections (e.g., due to uncorrectable transmission errors) or SDLS frame verification failures have a major impact on operations. While transmission errors will be dealt with by the COP-1 (reference [11]) and trigger automatic retransmission of rejected frames, frames rejection by the onboard SDLS function will need to be swiftly detected and analyzed by MOC to decide a course of action. The FSR provides a real-time synthetic report of the SDLS function's receiving end status and operation, enabling it to inform MOC of forward frame rejections as soon as they occur. MOC can then further investigate the related security event(s) by using the EP SA Management and Monitoring & Control directives (e.g., Dump Log, Self-Test, Read Sequence Number).

## 3.5.2 RELATION TO SPACE LINK PROTOCOLS

### 3.5.2.1 General

Since a frame cannot contain two OCFs at the same time, insertion of the FSR needs to be multiplexed with the insertion of the CLCW (report word from COP-1, reference [11]) in implementations in which both COP-1 and SDLS are used on the uplink. The multiplexing scheme is mission specific, though the following considerations need to be taken into account:

### 3.5.2.2 Transmission Rate of the FSR

Since all fields of the FSR (apart from the Alarm flag, discussed below) are non-persistent and updated with each forward frame received by the onboard SDLS function, it is desirable that the transmission rate of the FSR on the return link is equal to or higher than the forward link frame rate. This would guarantee that any security event detected by the onboard SDLS function is reported real-time to ground and that each security event can be associated unambiguously with the forward frame that caused it (except for spurious frames transmitted by an attacker). This would enable limited investigation of security events by a mission operations center without the need for dumping and analysis of the onboard security log (if implemented) or usage of appropriate Extended Procedures directives.

### 3.5.2.3 Constraints When COP-1 CLCW Is Present

This constraint on FSR transmission rate can only be met if the return link frame rate is significantly higher than the forward frame rate, since some return link frames’ OCF will be reserved for CLCW. Although it is not necessary that the CLCW reporting rate (from the receiving end to the sending end) match the Transfer Frame rate (from the sending end to the receiving end), some minimum CLCW sampling rate is necessary for the proper operation of the COP.

#### 3.5.2.4 Alarm Flag Persistence

In case the above-mentioned constraint on the FSR transmission rate cannot be met, FSR contains a persistent Alarm flag that will in all cases inform the Initiator that a security event has occurred on a forward frame since the Alarm Flag was last reset. Determination of the type of security event(s) and of the forward frame(s) involved will require sending appropriate EP directives (see Monitoring & Control and SA management sections).

### 3.5.3 HOW TO INTERPRET THE FLAGS

Since the purpose of the FSR is real-time reporting of the onboard SDLS function at the receiving end of a TC, AOS, or USLP link, all fields but one (the Alarm Flag) are non-persistent and are updated at each forward frame processed by the SDLS function.

The various information carried by the FSR can be interpreted as follows:

– Alarm flag (persistent): indicates that at least one forward Transfer Frame has been rejected by the onboard SDLS function since the last reset of the Alarm Flag. This flag can be reset from the ground by sending the Alarm Flag Reset Command PDU. This flag being persistent guarantees that no security event detected on board will go unnoticed by the Initiator, irrespective of the FSR transmission rate.

– Security event flags (non-persistent): indicates the type of security event triggered by the last received forward frame by the onboard SDLS function. The flags are updated at each forward Transfer Frame processed by the onboard SDLS function; the forward frame to which those flags relate is identified by the SPI and SN values transmitted in the last part of the FSR. Three generic types are reported:

  - Invalid SN: indicates that the SN carried by the last received Transfer Frame by the onboard SDLS function was invalid (i.e., outside the SN window).

  - Invalid MAC: indicates that the MAC carried by the last received Transfer Frame by the onboard SDLS function was invalid (i.e., did not match the MAC computed over the received Transfer Frame). This flag signals an integrity or authentication error on the related frame.

  - Invalid SA: indicates whether the last Transfer Frame received by the SDLS onboard function failed SA verification or carried an SPI pointing to a non-Operational SA or an Operational SA associated with an non-Active key.

  NOTE – SA verification consists of checking that the SPI carried by the received frame is pointing to an SA that is associated with the GVCID/GMAPID of that frame.

– Last SPI used (non-persistent): indicates the SPI carried in the last received Transfer Frame by the onboard SDLS function. This information, combined with the SN information, enables unambiguous identification of the last received transfer frame to which the above-mentioned Security Event Flags relate.

- SN value (non-persistent): contains the eight Least Significant Bits (LSBs) of the SN carried in the last received Transfer Frame by the onboard SDLS function. This SN is related to the Security Association that is pointed to by the SPI. SN, in combination with SPI, unambiguously identifies the last received transfer frame to which the above-mentioned Security Event Flags relate.

### 3.5.4 CONCEPT OF OPERATIONS FOR HANDLING ALARM FLAGS

While operating a secured forward link to a spacecraft with limited contact time, it is of the utmost importance to detect as promptly as possible any link disruption and to be able to discriminate between the two main causes of disruption, namely:

- transmission problems causing outage or frame rejection due to transmission errors;
- security events/attacks causing frame rejection by the onboard SDLS function.

At the forward link receiving end, two frame validation processes operate in sequence:

- the Frame Acceptance & Reporting Mechanism (FARM) of the Communications Operation Procedure (COP-1) specified in reference [11]. This mechanism checks the validity of the transfer frame based on:
  - the results of the decoding of the channel code (presence of uncorrectable errors),
  - the results of the check of the frame CRC,
  - the result of the check of structure of the frame and validity of its header fields;
- the onboard SDLS function specified in reference [1]. This mechanism checks the validity of the transfer frame based on:
  - validity of the MAC, which, if valid, guarantees the integrity and authenticity of the frame,
  - validity of the SN, which, if valid, guarantees that the frame is not a replay from a previously sent frame,
  - validity of the SPI, which, if valid, guarantees that an appropriate active key and SA have been used to protect the frame.

The COP FARM and the SDLS function will reject/discard any frame that fails their respective checks. Both FARM and SDLS functions have their real-time reporting message that will enable the Initiator to detect and discriminate between transmission errors and security events/attacks:

- CLCW for the COP;
- FSR for SDLS.

Both types of report messages (CLCW and FSR) will be multiplexed in the Operational Control Field of downlink TM, AOS, or USLP frames. In most cases, downlink frame rate being significantly higher than uplink frame rate, at least one CLCW and one FSR can be transmitted for each uplink frame received enabling full real-time reporting of any communication (COP)- or security (SDLS)-related discarding of an uplink frame.

SDLS can secure forward link (e.g., uplink using TC, AOS, or USLP) and/or return link (e.g., downlink using TM, AOS, or USLP). Nevertheless, FSR will only be generated at the Recipient (typically the spacecraft) and sent to the Initiator (typically the control center) to report the status of the Recipient security unit and security events detected at the Recipient. For the return link, there is no reporting mechanism from the Initiator (ground) to the Recipient (spacecraft) for security events detected on the return link.  This is operationally not needed.

# 4 DESIGN CONCEPTS

## 4.1 ERROR HANDLING

### 4.1.1 SIGNALING ERRORS

As noted in 2.3.3.1, SDLS Extended Procedures PDU exchanges do not contain any built-in mechanisms for assuring reliable delivery.  EP PDUs provide a limited set of directives and replies.  The Extended Procedures do not define the mechanism for acknowledging that EP PDUs are received.  Reception acknowledgements are expected to be communicated using spacecraft telemetry.

### 4.1.2 EXECUTION ERRORS

#### 4.1.2.1 General

SDLS Extended Procedures rely for operation on the exchange between the Initiator (e.g., the mission control center) and the Recipient (e.g., spacecraft) of Command PDUs and Reply PDUs. In the course of execution of those procedures, a number of failure conditions can occur, for example:

- Command PDU received with incorrect syntax or erroneous parameters;
- Command PDU received out of sequence.

Those failure conditions will, in general, prevent the safe execution of the procedure by the Recipient. The SDLS Extended Procedures standard (reference [2]) does not specify any specific behavior for the Initiator or the Recipient in the case of failure conditions. Nevertheless, the following general considerations should be considered by mission implementers:

- An invalid directive should not be executed.
- Rejected EP commands or procedure failures should be reported from the Recipient to the Initiator through housekeeping telemetry of the spacecraft. No specific error reporting format is specified by the SDLS Extended Procedures standard (reference [2]).
- The execution of EP directives following EP procedure failure or EP command rejection is to be handled through conditional logic as in any other conditional commanding.

#### 4.1.2.2 Key Management

The directives for Key Activation, Key Deactivation, and Key Destruction are explicitly stated to be atomic operations: if any part fails, the entire operation should be rolled back and treated as failed.  The Key Verification directive, on the other hand, provides individual challenge responses for each key and so is successfully executed even if verification fails for any individual key.

The Key Inventory directive may be useful for troubleshooting discrepancies after errors are encountered.

#### 4.1.2.3 SA Management

Several of the SA Management Procedures direct the Recipient to verify preconditions before commencing any execution of operations. Whenever preconditions cannot be verified on board, the operation should be halted and treated as failed. Status communicated back to the Initiator using telemetry should report the failed directive.

The SA Status directive may be useful for troubleshooting discrepancies after errors are encountered.

#### 4.1.2.4 Monitoring & Control

The Log Status and Dump Log directives may be useful for troubleshooting discrepancies after errors are encountered.

## 4.2 REDUNDANCY

### 4.2.1 GENERAL

Most spacecraft implementing SDLS will also have redundancy of frame processing and associated security units. It is possible to manage security units through the SDLS Extended Procedures such that secure communications is maintained while the security unit is actively being managed. Two typical implementation scenarios are discussed below.

In both scenarios, the range of available SPIs (i.e., SAs) should be partitioned between Nominal and Redundant strings (each side of a redundant prime/backup pair) to guarantee uniqueness across strings, since SA states are not shared across strings.

### 4.2.2 PHYSICAL CROSS-STRAPPING

Scenario 1: Redundancy is provided when each communications ‘string’ (i.e., each side of a redundant prime/backup pair) has its own independent virtual channel(s) or MAP(s) so that RF data link traffic is directed explicitly to use a specific string (‘Side A’ vs. ‘Side B’).

In Scenario 1, each security unit is addressed using the virtual channels, or MAPs, and SAs which belong to that string. Nominal RF traffic is addressed to one string ‘Side A’ using Side A’s virtual channels or MAPs at the same time SDLS EP directives are addressed to the other string ‘Side B’ using Side B’s virtual channels or MAPs. There is no ambiguity about which security unit is addressed by a specific SDLS EP directive.

In this case, assignment of separate virtual channels (not used by nominal traffic) and/or SAs for each side’s security unit will prevent ambiguity about which security unit is addressed by a specific SDLS EP directive. Use of the two reserved SPI values (0 and 65535) to address separate security units is one possible method of accomplishing this.

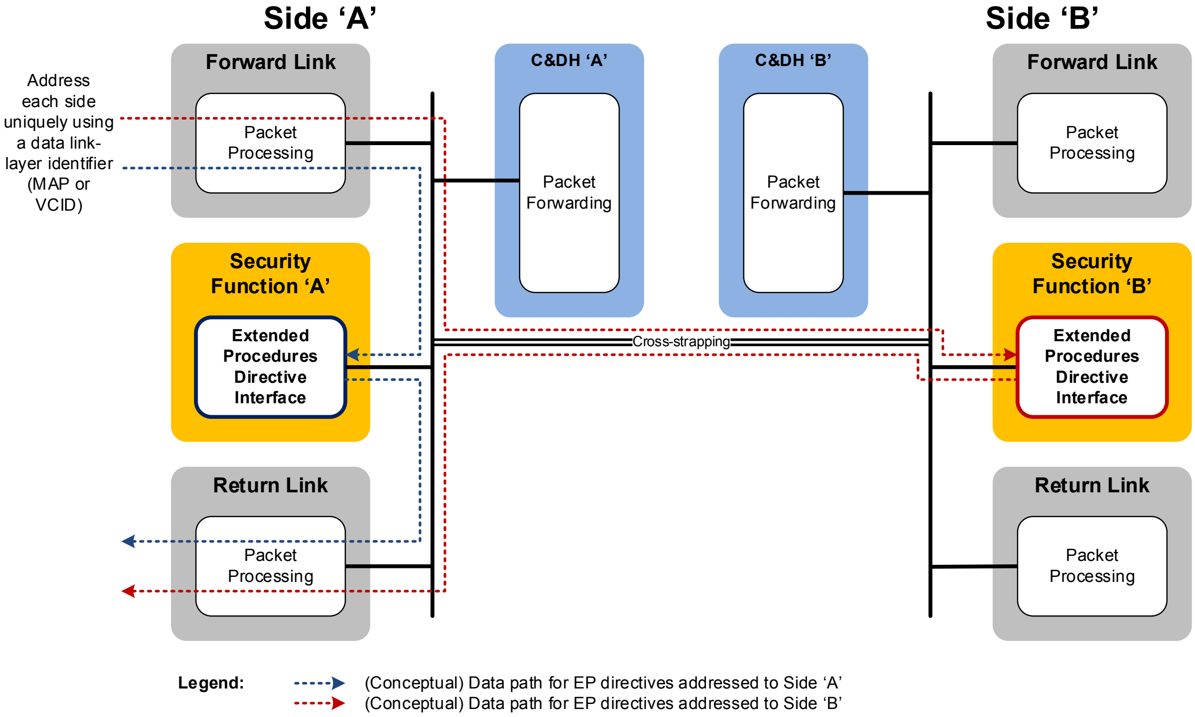

Side 'A'
Side 'B'
Address each side uniquely using a data link-layer identifier (MAP or VCID)
Forward Link
Packet Processing
C&DH 'A'
Packet Forwarding
C&DH 'B'
Packet Forwarding
Forward Link
Packet Processing
Security Function 'A'
Extended Procedures Directive Interface
Cross-strapping
Security Function 'B'
Extended Procedures Directive Interface
Return Link
Packet Processing
Return Link
Packet Processing
Legend:
(Conceptual) Data path for EP directives addressed to Side 'A'
(Conceptual) Data path for EP directives addressed to Side 'B'

**Figure 4-1: Physical Cross-Strapping**

### 4.2.3 LOGICAL CROSS-STRAPPING

Scenario 2: Redundancy is provided when both communications strings of a redundant pair share the same virtual channel(s), processing traffic in parallel so that RF traffic is output by whichever specific string currently acts as prime.

In Scenario 2, even though nominal RF traffic may continue along the virtual channel(s) shared by both strings, it is necessary that each string's security unit be addressable using an identifier (e.g., unique Application Process Identifier [APID]) that belongs to it alone. It is further necessary that, in addition to each communications string being able to route SDLS EP directives to its own security unit, that it also be able to route SDLS EP directives to the security unit belonging to the other string.

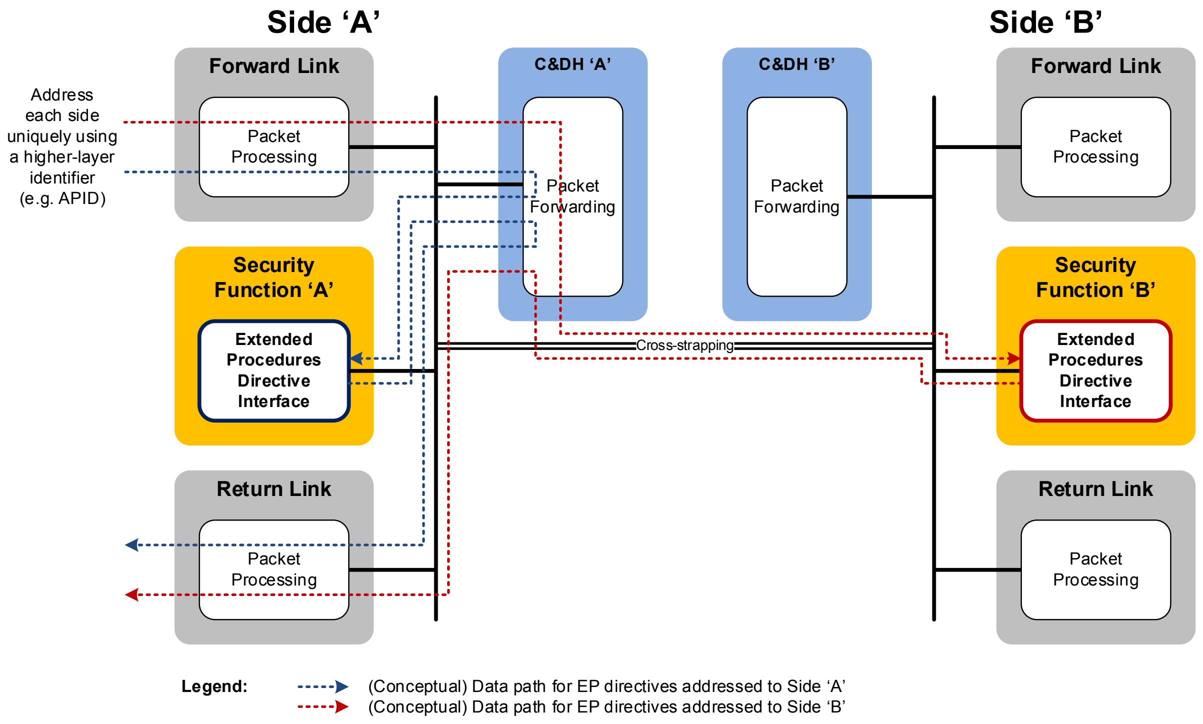

Side ‘A’
Side ‘B’
Forward Link
Packet Processing
C&DH ‘A’
Packet Forwarding
C&DH ‘B’
Packet Forwarding
Forward Link
Packet Processing
Address each side uniquely using a higher-layer identifier (e.g. APID)
Security Function ‘A’
Extended Procedures Directive Interface
Cross-strapping
Security Function ‘B’
Extended Procedures Directive Interface
Return Link
Packet Processing
Return Link
Packet Processing
Legend:
(Conceptual) Data path for EP directives addressed to Side ‘A’
(Conceptual) Data path for EP directives addressed to Side ‘B’

**Figure 4-2: Logical Cross-Strapping**

## 4.3 MISSION SCENARIOS

### 4.3.1 ‘CLASSICAL’ GROUND-SPACE SCENARIO

For space-to-ground links in which a mission operations center controls a single spacecraft, the ‘Initiator’ (master) and source of all uplinked EP Command PDUs is the mission operations center. The ‘Recipient’ (slave unit being managed) is the spacecraft, and any applicable Reply PDUs and FSRs are sent via the telemetry downlink path.

Depending on the mission design, both the uplink and the downlink may implement SDLS, and therefore it is necessary to indicate which directional link (up or down) is being managed by specific Security Association Management directives. The Initiator signals SA Management direction using the two-bit Service Group field in the Extended Procedures PDU. One Service Group value is provided for directives managing SAs that secure the ground-to-space link from Initiator to Recipient, and a separate Service Group value is provided for directives managing SAs that secure the space-to-ground link from Recipient to Initiator.

### 4.3.2 SINGLE SPACECRAFT, MULTIPLE LINKS TO GROUND

For space-to-ground links in which a single spacecraft has multiple downlink paths (e.g., separate high- and low-rate links), the two-bit Service Group field in the Extended Procedures PDU is insufficient for the Initiator to specify which of the available downlinks is applicable to a given SA Management directive. In this case, the applicable security unit should be explicitly addressed using one of the methods described in 4.2, for example, using a unique Space Packet APID for carrying all Command and Reply PDUs to a given security unit.

The FSR is not used for downlink not addressed to the MOC.

### 4.3.3 SPACE-TO-SPACE LINKS

A third mission network topology that can be considered is space-to-space links (e.g., within a constellation of satellites). SDLS protocol and its associated Extended Procedures are designed to operate in a master-slave configuration, but it is also possible to use the SDLS protocol and its Extended Procedures for securing and managing space-to-space links.

In all cases, there is no negotiation between endpoints as to which is the master: all directives are issued from a predetermined master (referred to as Initiator in reference [2]) to a predetermined slave (referred to as Recipient in reference [2]). Therefore, in the case of space-to-space links, a hierarchy needs to be established among the communicating satellites so that for each connection, a master (Initiator) and a slave (Recipient) are unambiguously defined. This hierarchy usually does not exist for space-to-space links and within a constellation.

Moreover, a key management scheme needs to be implemented across the constellation. Possible examples are:

- a dedicated set of keys managed for each possible connection, which might become impractical when the number of communicating satellite pairs increases;
- a common set of keys shared throughout the constellation while still ensuring that the cryptographic algorithm and mode of operation requirements are met (e.g., key-IV uniqueness for AES-GCM).

As shown above, SDLS protocol and its Extended Procedures can be used to secure inter-satellite links. However, for larger constellations, other approaches should be considered, such as security at a higher layer (e.g., network or application layer).

## 4.4 RELATIONSHIP TO OTHER CCSDS STANDARDS

### 4.4.1 CRYPTOGRAPHIC ALGORITHMS (352.0-B)

#### 4.4.1.1 General

SDLS was designed to be compatible with a variety of algorithms. Neither the core SDLS protocol nor the SDLS Extended Procedures mandate the use of a specific cryptographic algorithm. The (non-normative) baseline modes specified in references [1] and [2] to foster interoperability use the AES-GCM algorithm with 256-bit keys, 96-bit IV, and 128-bit MAC. In the case of TC Space Link Protocol, the baseline modes use the Advanced Encryption Standard Cipher-based Message Authentication Code (AES-CMAC) algorithm with 256-bit keys and 128-bit MAC.

#### 4.4.1.2 Implications for SA Creation

a) ARSN: When AES-GCM is used (as in the baseline mode for TM, AOS, and USLP), SDLS uses the IV and ARSN as a single field. The Create SA procedure initializes the length and initial values of the ARSN, which also serves a double function as the IV.

b) IV: As an AES-GCM IV, it comprises a ‘fixed’ field (a value static to the originating device/context) and an ‘invocation’ field (a value different with every invocation). This field therefore limits both the number of distinct devices/contexts that can call the GCM authenticated encryption function with a single key, and the number of times each one can call it. For example, a 32-bit fixed field implies a limit of $2^{32}$ on the number of distinct devices/contexts, and a 64-bit invocation field implies a limit of $2^{64}$ on the number of invocations of the GCM authenticated encryption function. (See reference [15] for more detail.)

c) MAC: The length of the authentication tag constrains the safe number of operations over the lifetime of the key. Up to half the length of the MAC is thought to be a reasonable limit; in other words, a 128-bit MAC would provide authentication assurance for up to $2^{64}$ frames. However, message size provides an additional constraint as the MAC length is shortened. Most supported CCSDS transfer frame sizes qualify as ‘short’ messages for AES-GCM algorithm considerations. Reference [15] states that when the MAC is 64 bits long, and the maximum combined length of ciphertext and Additional Authenticated Data (AAD) in a message is $2^{15}$ bytes, the maximum invocations of the authenticated decryption function should not be greater than $2^{32}$.

#### 4.4.1.3 Implications for Key Verification

For each Key ID passed by the Key Verification procedure, a plaintext challenge is provided in the Command PDU, and the same challenge is returned as ciphertext in the Reply PDU along with an IV and a MAC. Because of this, the algorithm used needs to be resistant to known-plaintext attacks. AES-GCM has no known vulnerability to known-plaintext attacks, as long as the rule against ever repeating the use of an IV with the same key is followed. This rule needs to be obeyed, even within a single Reply PDU. The challenge should also be a random pattern to increase the difficulty of this type of attack.

NOTE – Implementation of any challenge/response directives (e.g., Key Verification) that require protection of reply PDUs needs to ensure that key/IV pairs are not reused between the Initiator and the Recipient. Prevention of key/IV pair collisions is implementation specific, but it can be managed, for example, by allocating to each Initiator and Recipient its own pool of available IVs.

### 4.4.2 SYMMETRIC KEY MANAGEMENT (354.0-M)

#### 4.4.2.1 Key Management Schemes Not Implemented

There is a third key management scheme listed in reference [8] but not directly supported by the SDLS Extended Procedures:

> Scheme 3: a subset of keys (master keys/KEKs and session keys) are pre-loaded on satellite before launch; session keys are generated on board from master keys and an uploaded non-secret seed.

The complexity of this scheme necessitates autonomous procedures operating beyond the scope of the SDLS Extended Procedures. It is heavily dependent upon a subset of cryptographic operations, including random number generation and key derivation algorithms, for which no CCSDS Recommended Standard currently exists. As such, any implementation of this scheme would be mission specific. For additional procedures to support such a scheme, the reader is referred to reference [8], 4.3.7.

#### 4.4.2.2 Key States Not Implemented

The SDLS Extended Procedures do not implement the full range of key states described in reference [8].

First, there is an optional Suspended state in reference [8] that anticipates a temporary operational restriction on the use of previously activated keys. One such use case would be for setting aside a set of still-unused keys that had been activated in the expectation of their being used in the near term, and which (for whatever reason) are no longer anticipated to be needed soon. This key state makes sense only for key management systems capable of storing a very large quantity of keys. Since the size constraints of onboard key storage typically preclude storing very many keys at any one time, use of the Suspended state in space systems is not anticipated. Even if ground-based key management systems supported the Suspended state, its implementation on board would be superfluous, and no SDLS Extended Procedures are provided to support it.

Second, the Compromised state in reference [8] prevents the operational use of keys that are unfit due to their having been disclosed. In the SDLS Extended Procedures, the Compromised state is listed as a state applicable only to the Initiator. Ground-based key management systems will often preserve Compromised keys in storage for recordkeeping. The use case is not applicable to space systems, so no SDLS Extended Procedures are provided to support it. In the event the Initiator (master) needs to transition keys stored locally into the Compromised state, it would issue the Key Destruction directive to the Recipient (slave) to destroy the same keys.

# ANNEX A

# BASELINE MODES

## A1 INTRODUCTION

This annex provides the rationale for the baseline implementation mode specified in annex D of the SDLS Extended Procedures Blue Book, reference [2].

## A2 FRAME SECURITY REPORT

The FSR is the PDU transmitted from the Recipient to the Initiator of an SDLS secured TC, AOS, or USLP uplink. It provides the systematic, real-time mechanism by which the SDLS function at the receiving end reports the status of frame acceptance to the sending end.

The baseline implementation mode specified for integrating the FSR into the TM, AOS, and USLP transfer service is as follows:

a) The FSR is reported as Operational Control Field (OCF Type 2).

b) In case COP-1 is reporting on the same virtual channel, the FSR reporting alternates with the Communications Link Control Word (OCF Type 1) reporting.

The purpose of the Operational Control Field is to provide a standardized mechanism for reporting a small number of real-time functions, such as supporting the reporting mechanism for the onboard SDLS security function. Two types of OCF have been specified in TM, AOS, and USLP Space Data Link Protocol:

- Type 1 for the CLCW of the COP-1 retransmission protocol;
- Type 2 for the FSR of the SDLS security protocol.

Both reporting mechanisms are usually needed on TC, AOS, or USLP uplinks. Therefore OCF needs to be shared between COP-1 and SDLS reporting as specified in the baseline mode. Ideally, a CLCW and an FSR should be transmitted to ground for each received uplink TC, AOS, or USLP transfer frame. If the downlink frame rate is at least twice the uplink frame rate, this is feasible by interleaving the two types of reports in the OCF of the downlink TM, AOS, or USLP frames. If this condition is not met, subsampling of CLCW, FSR, or both will need to be done. In that case, all Security Event Flags in the FSR being non-persistent, the initiator (Mission Control Center) will not be able to relate a given security event to a specific frame directly from the FSR analysis. The occurrence of a Security Event will be signaled to the initiator by the Alarm Flag, which is a persistent flag in the FSR. The initiator will then have to investigate through analysis of the recipient (onboard security unit) telemetry to determine which frame has triggered the security event.

## A3 PROTOCOL DATA UNITS

SDLS Extended Procedures commands and reports share a common message format, based on the TLV concept. The Tag field uniquely identifies the command or the report. The Length field indicates the length of the Value field (may be zero). The (optional) Value field contains additional data pertaining to the message.

The TLV concept allows nesting: the Value field can itself be composed of one or more TLV messages. Given the procedures selected for the baseline mode, there is no need for nested TLV PDUs.

## A4 RESERVED SPI/SA

Sensitive EP Service PDUs need to be communicated over a SDLS channel protected by authenticated encryption to guarantee integrity, authenticity, and confidentiality. All other EP Service PDUs need to be at least authenticated to guarantee integrity and authenticity before execution or processing.

In the baseline implementation mode, the two SDLS reserved SPIs (values of 'all zeros' [0] and 'all ones' [65535]) defined in reference [1] are used for exchanging EP Service PDUs. This allows for the use of dedicated SAs to protect the transmission of EP Service PDUs and therefore the use of different SAs from the one being affected by the EP Service Command PDU, preventing unintentional loss of control of an SA.

## A5 KEY MANAGEMENT SERVICE

### A5.1 GENERAL

The baseline implementation mode includes all EP key management procedures except Key Destruction and Key Inventory:

- Key Destruction is not needed in most mission scenarios. Key revocation at both ends of the Secure Channel by the Key Deactivation procedure is sufficient to guarantee that a compromised key or 'burnt' key (all IV or ARSN used) can no longer be used for cryptographic operations. A deactivated key can only be used to decrypt previously encrypted data.
- Key Inventory is not absolutely necessary to manage the onboard set of keys. Onboard key states can be inferred from command execution verification reports of other baseline mode directives.

## A5.2 SECURITY ALGORITHM AND KEY CONFIGURATION

### A5.2.1 Selection of Cryptographic Algorithm for OTAR and Key Verification

The cryptographic algorithm is selected from the CCSDS Standard on Cryptographic Algorithms (reference [12]), in particular from the recommended algorithms for Authenticated Encryption. Therefore AES-GCM is the recommended algorithm for the OTAR and Key Verification operations that require authentication and encryption of the uploaded keys.

Recent cryptographic research on AES-GCM has identified a weakness concerning certain keys (references [16] and [17]). The user is invited to carefully consider the key generation and selection process in order to avoid the use of 'weak' keys.

### A5.2.2 Design of Cryptographic Algorithm Parameters: MAC and Key Lengths

With the selection of AES-GCM, the selection of MAC and key length is as follows:

– The MAC length is automatically set to 128 bits, which is the maximum possible value. This value is considered sufficiently secure for civilian missions as justified by the security analysis in A2.3 of reference [10].

– The key length is limited to three possible values: 128, 192, and 256 bits. A value of 128 bits is considered sufficient for civilian missions as justified by the security analysis in A2.3 of reference [10], but a margin on key length is necessary to anticipate the threat of quantum computers. This leads to a selection of a 256-bit key for the SDLS Extended Procedures.

### A5.2.3 IV Construction

AES-GCM requires an Initialization Vector. There are two specified approaches for constructing an IV for AES-GCM (see 8.2 of reference [15]). The recommended construction is the following: deterministic with 96 bits in total length.

To maintain security, it is essential to avoid a repetition of the IV with the same cryptographic key. Failure to meet this requirement will imply a security leakage. Further details can found in reference [15].

## A6 KEY MANAGEMENT SERVICES PARAMETERS

### A6.1 GENERAL

The baseline implementation mode includes 4 Key Management procedures: OTAR, Key Verification, Key Activation, and Key Deactivation. *Not* selected for the baseline mode are the Key Destruction and Key Inventory procedures.

### A6.2 OTAR

The format of the OTAR Command PDU is shown in figure A-1.

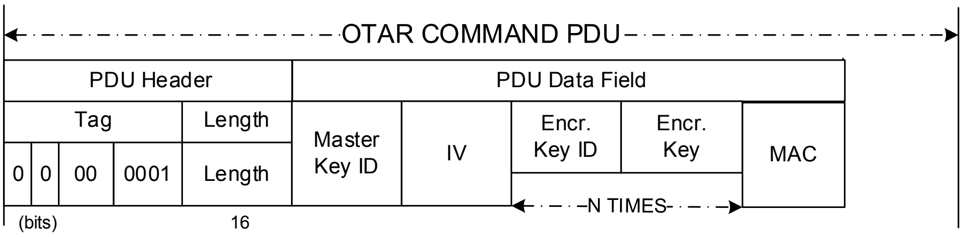


**Figure A-1: OTAR Command PDU**

The baseline implementation configuration selected for the OTAR procedure operation is:

a) The Master Key ID field of the OTAR Command PDU has a size of 16 bits.

   It is up to the implementer to decide if master keys are assigned a special range from the total Key ID range. A 16-bit ID for the master key and session keys allows for 65536 keys in total, which is largely sufficient for most missions.

b) The Initialization Vector field of the OTAR Command PDU has a size of 96 bits.

   This size of 96 bits derives from the choice of AES-GCM as cryptographic algorithm for OTAR. (See justification in A5.2.3.)

c) Each Encrypted Key Block of the OTAR Command PDU has a size of 272 bits, consisting of

   – the Key ID field has a size of 16 bits;

   – the Session Key field has a size of 256 bits.

   The size of the Key ID and Session Key fields (respectively 16 and 256) derive from the settings of the SDLS baseline mode defined in annex E of reference [1] and justified in annex A of reference [10].

d) The MAC field of the OTAR Command PDU has a size of 128 bits.

   This size of 128 bits for the MAC derives from the choice of AES-GCM as the cryptographic algorithm for OTAR. (See justification in A5.2.2.)

e) In baseline mode, the OTAR Command PDU allows for the transfer of up to 16 session keys (N<=16). This limitation, while acceptable operationally, allows for the complete Command PDU to fit into one TC frame.

## A6.3 KEY ACTIVATION

The format of the Key Activation Command PDU is shown in figure A-2.

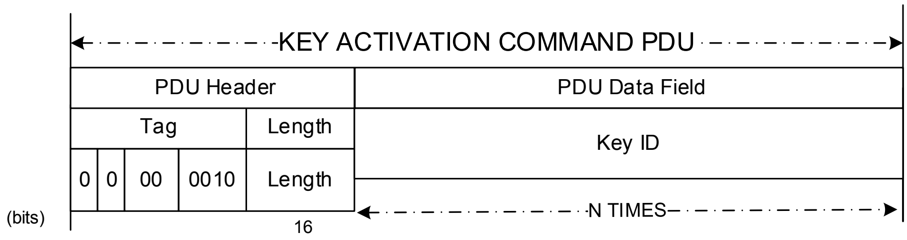


**Figure A-2: Key Activation Command PDU**

a) In baseline mode, the Key ID length is 16 bits, which allows for 65536 keys in total (session keys + master keys). This is largely sufficient for most missions, especially with the possibility to upload new keys in flight with the OTAR procedure.

b) Baseline mode allows for up to 32 keys to be activated with a single Command PDU (N<=32). This limit is coherent with the OTAR procedure and acceptable operationally.

## A6.4 KEY DEACTIVATION

The Key Deactivation Command PDU uses the same parameters as the Key Activation Command PDU. (See A6.3.)

## A6.5 KEY VERIFICATION

The format of the Key Verification Command PDU is shown in figure A-3.

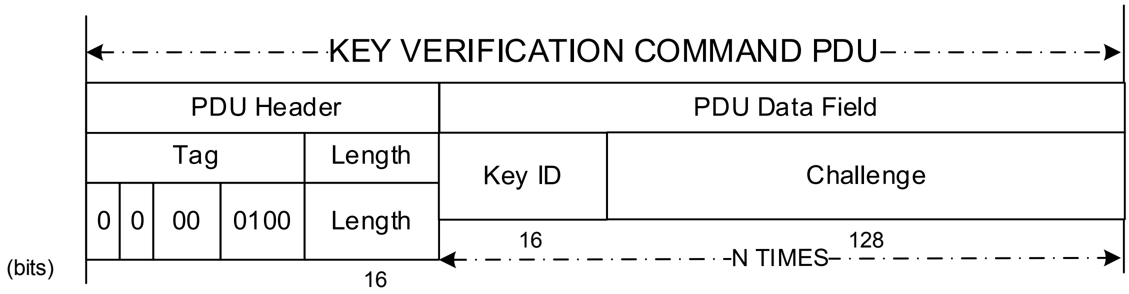


**Figure A-3: Key Verification Command PDU**

a) In baseline mode, the Key ID length is 16 bits, which allows for 65536 keys in total (session keys + master keys). This is largely sufficient for most missions, especially with the possibility to upload new keys in flight with OTAR procedure.

b) Challenge size is 128 bits, which is coherent with the algorithm selected for Key Verification, which is AES-GCM operating on 128-bit blocks. The challenge should be a pure random pattern to avoid clear-cipher text attacks, the potential attacker being able to intercept both the challenge (clear text) in the Command PDU and the encrypted challenge (in the reply PDU), even though AES-GCM has no known vulnerability to clear-cipher text attacks.

c) Baseline mode allows for up to 32 keys to be verified with a single Command PDU (N<=32). This limit is coherent with OTAR procedure and acceptable operationally.

The format of the Key Verification reply PDU is shown in figure A-4.

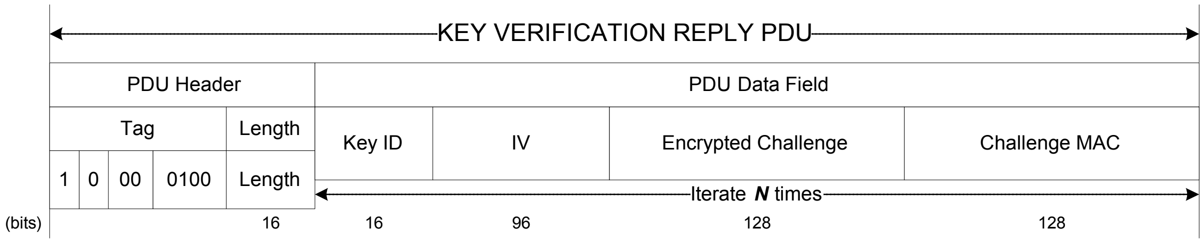


**Figure A-4: Key Verification Reply PDU**

a) In baseline mode, Key ID length is 16 bits, allowing for 65536 keys in total (session keys + master keys), which is largely sufficient for most missions, especially with the possibility to upload new keys in flight with OTAR procedure.

b) The Initialization Vector field of the Key Verification Reply PDU has a size of 96 bits. This size of 96 bits derives from the choice of AES-GCM as cryptographic algorithm for Key Verification. (See justification in A5.2.3.)

c) The encrypted challenge size is 128 bits, coherent with the algorithm selected for Key Verification, which is AES-GCM operating on 128-bit blocks. The encrypted challenge enables the initiator to check the integrity of each of the keys loaded through the OTAR procedure. AES-GCM has no known vulnerability to clear and cipher text attacks at this current time. Therefore it is acceptable to transmit unencrypted between initiator and recipient both challenge and encrypted challenge.

d) The challenge MAC is 128 bits, which is coherent with AES-GCM.

e) Baseline mode allows for up to 32 keys to be verified with a single Command PDU (*N*≤32).

f) Before verifying a key, this key needs to be activated, which starts its operational lifetime.

## A7 SECURITY ASSOCIATION MANAGEMENT SERVICE PARAMETERS

### A7.1 INTRODUCTION

Baseline mode of Extended Procedures retains six SA Management procedures: Start SA, Stop SA, Rekey SA, Expire SA, Set ARSN, and Read ARSN. Create SA, Delete SA, Set ARSN Window, and SA Status Request are not selected for baseline mode for the following reasons:

- In most missions, there is no need to create or to delete an SA in flight. All SAs needed for the mission duration are preloaded on board. Up to 65536 SAs can be loaded on board before launch, which is largely sufficient to cover the lifetime.
- The ARSN Window can be selected statically for the mission. Most missions will select a window of maximum size allowing any up counting ARSN. This protects against replay while allowing for any type of gaps in the reception of frames at the recipient.
- SA Status Request: SA status can in most cases be managed from the ground.

NOTE – Concerning ARSN length, the Rekey SA and Set ARSN procedures specified in the baseline mode allocate 96 bits to be able to carry an ARSN for any of the supported Space Link Protocols.

a) As used in the baseline mode for TM, AOS, and USLP, the ARSN is 96 bits in length. Since the ARSN is identical to the IV for the baseline mode AES-GCM algorithm, executing this procedure also sets the IV.

b) As used in the baseline mode for TC, the ARSN is 32 bits in length. If this ARSN field carries an ARSN for TC SAs, the left-most 64 bits are zeroed.

### A7.2 START SA

The format of the Start SA Command PDU is shown in figure A-5.

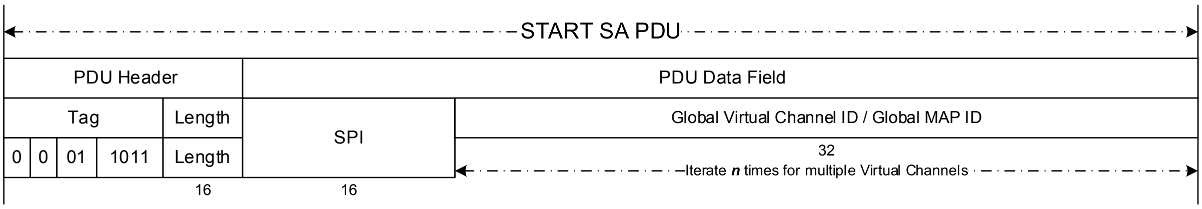

START SA PDU
PDU Header
PDU Data Field
Tag
Length
0
0
01
1011
Length
16
SPI
16
Global Virtual Channel ID / Global MAP ID
32
Iterate *n* times for multiple Virtual Channels

**Figure A-5: Start SA Command PDU**

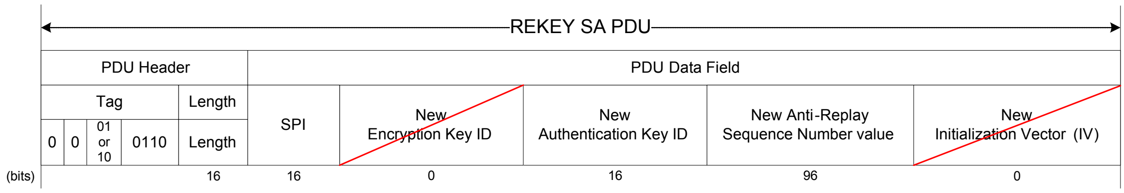
The baseline implementation configuration selected for Start SA procedure operation is:

a) The SPI field is 16 bits in length. It allows for 65536 security associations in total, which is largely sufficient for most missions.

b) Each GVCID/GMAPID can fit into a 32-bit length field. Each GVCID/GMAPID listed is associated with the SA identified by the SPI. GVCID and GMAPID are specified in the relevant Space Data Link Protocol (SDLP) Recommended Standards (references [4], [5], [6], and [7]) for the space data link protocol used on the space link associated with the SPI. The GVCID is the concatenation of: GVCID = Transfer Frame Version Number (TFVN) + SpaceCraft IDentifier (SCID) + VCID. GMAPID is the concatenation of: GMAPID = TFVN + SCID + VCID + MAPID. In all cases (whichever the SDLP used), GVCID, and GMAPID can each be coded on less than 32 bits, the leftmost bits being filled with '0' to complete the 32-bit field.

## A7.3 REKEY SA

The format of the Rekey SA Command PDU is shown in figure A-6.



| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | REKEY SA PDU | | | | | | | | | |
| | PDU Header | | | | | PDU Data Field | | | | |
| | Tag | | | | Length | SPI | ~~New Encryption Key ID~~ | New Authentication Key ID | New Anti-Replay Sequence Number value | ~~New Initialization Vector (IV)~~ |
| | 0 | 0 | 01<br>or<br>10 | 0110 | Length | | | | | |
| (bits) | | | | | 16 | 16 | 0 | 16 | 96 | 0 |

**Figure A-6: Rekey SA Command PDU**

The ARSN field length needs to be 96 bits to be able to carry either ARSN (TC SAs) or IV (TM, AOS, or USLP SAs). If this ARSN field carries an ARSN (TC SAs) and not an IV, the left-most 64 bits are zeroed.

Since the ARSN is identical to the IV for the SDLS baseline mode AES-GCM algorithm, executing this procedure will set the IV.

The cryptographic algorithms selected for the baseline mode are:

- AES-GCM for TM, AOS, and USLP;
- AES-CMAC for TC.

Therefore in all cases, only one single key and Key ID are required for operation in baseline mode, whichever the Space Data Link Protocol used on the secured link.

### A7.4 SET ARSN, READ ARSN

The formats of the Set ARSN command and Read ARSN reply PDUs are shown in figure A-7.

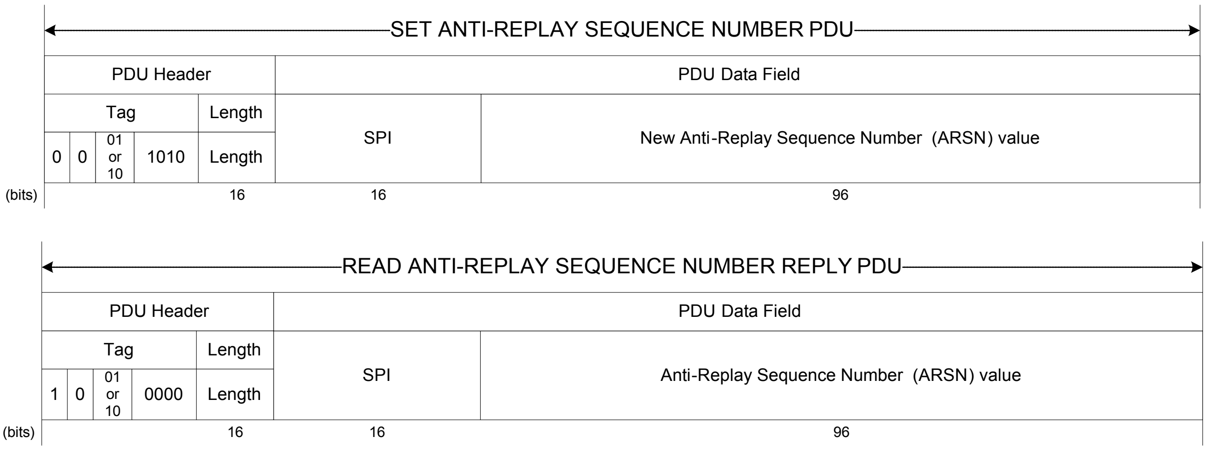

SET ANTI-REPLAY SEQUENCE NUMBER PDU
PDU Header
PDU Data Field
Tag
Length
0
0
01 or 10
1010
Length
SPI
New Anti-Replay Sequence Number (ARSN) value
(bits)
16
16
96
READ ANTI-REPLAY SEQUENCE NUMBER REPLY PDU
PDU Header
PDU Data Field
Tag
Length
1
0
01 or 10
0000
Length
SPI
Anti-Replay Sequence Number (ARSN) value
(bits)
16
16
96

**Figure A-7: Set ARSN Command PDU and Read ARSN Reply PDU**

The ARSN field length needs to be 96 bits to be able to carry either ARSN (TC SAs) or IV (TM, AOS, or USLP SAs). If this ARSN field carries an ARSN (TC SAs) and not an IV, the left-most 64 bits are zeroed.

Since the ARSN is identical to the IV for the SDLS baseline mode AES-GCM algorithm, executing the Set ARSN procedure will set the IV.

## A8 MONITORING AND CONTROL SERVICE PARAMETERS

The baseline implementation mode of Extended Procedures includes the Ping and Alarm Flag Reset procedures.

*Not* selected for the baseline mode are:

a) The Log Status, Dump Log, and Erase Log procedures. In most missions, an onboard security log is not needed. The FSR provides enough observability to record on the ground all security events, provided that the FSR is sampled at each received TC frame.

b) The Self-Test procedure. Self-Test is usually implemented in the security unit in a mission-specific way for which interoperability is not needed.

# ANNEX B

# ACRONYMS AND ABBREVIATIONS

This annex lists the acronyms and abbreviations used in this Report.

| | |
|---|---|
| AAD | additional authenticated data |
| AES | Advanced Encryption Standard |
| AOS | Advanced Orbiting Systems |
| APID | application process identifier |
| ARSN | anti-replay sequence number |
| CCSDS | Consultative Committee for Space Data Systems |
| CLCW | communications link control word |
| COP-1 | Communications Operation Procedure-1 |
| CRC | cyclic redundancy check |
| EP | Extended Procedures |
| FARM | frame acceptance & reporting mechanism |
| FSR | frame security report |
| GMAPID | global multiplexer access point identifier |
| GVCID | global virtual channel identifier |
| ID | identifier |
| IP | Internet Protocol |
| ISO | International Standards Organization |
| IV | initialization vector |
| KEK | key encryption key |
| MAC | message authentication code |
| MAP | multiplexer access point |
| MC | master channel |

| | |
|---|---|
| OCF | operational control field |
| OTAR | over-the-air rekeying |
| PDU | protocol data unit |
| RDBMS | Relational Database Management System |
| SA | security association |
| SCID | spacecraft identifier |
| SDL | Space Data Link |
| SDLS | Space Data Link Security Protocol |
| SDLP | Space Data Link Protocol |
| SDU | service data unit |
| SN | sequence number |
| SPI | security parameter index |
| TC | telecommand |
| TFVN | transfer frame version number |
| TLV | tag, length, value |
| TM | telemetry |
| USLP | Unified Space Link Protocol |
| VC | virtual channel |
| VCID | virtual channel identifier |