GLOBALPLATFORM®
THE STANDARD FOR SECURE DIGITAL SERVICES AND DEVICES

# GlobalPlatform Technology

# Remote Application Management over HTTP

## Card Specification v2.3 – Amendment B

## Version 1.1.3.26 (target v1.2)

**Public Review**

**July 2021**

**Document Reference: GPC_SPE_011**

THIS SPECIFICATION OR OTHER WORK PRODUCT IS BEING OFFERED WITHOUT ANY WARRANTY WHATSOEVER, AND IN PARTICULAR, ANY WARRANTY OF NON-INFRINGEMENT IS EXPRESSLY DISCLAIMED. ANY IMPLEMENTATION OF THIS SPECIFICATION OR OTHER WORK PRODUCT SHALL BE MADE ENTIRELY AT THE IMPLEMENTER'S OWN RISK, AND NEITHER THE COMPANY, NOR ANY OF ITS MEMBERS OR SUBMITTERS, SHALL HAVE ANY LIABILITY WHATSOEVER TO ANY IMPLEMENTER OR THIRD PARTY FOR ANY DAMAGES OF ANY NATURE WHATSOEVER DIRECTLY OR INDIRECTLY ARISING FROM THE IMPLEMENTATION OF THIS SPECIFICATION OR OTHER WORK PRODUCT.

# Contents

# Figures



# Tables

# 1 Introduction

This document defines a mechanism for an Application Provider to perform Remote Application Management (RAM) according to ETSI TS 102 226 [102 226] (i.e. loading, installation, and personalization) using the HTTP protocol (RFC 2616 [HTTP]) and PSK TLS security Over-The-Air. A third party communication network may be used if the Application Provider has no OTA capability. This third party shall not be able to access clear text of any confidential data and code belonging to the Application Provider. This document describes:

- How to open an Over-The-Air connection with a remote server, based on HTTP and PSK TLS security
- How commands are sent to a Security Domain
- How responses of these commands are returned to the remote server
- How this mechanism can be used over a third party communication network
- A new key type for PSK TLS keys

[GPCAR] provides latest recommendations on cryptographic algorithms and protocols that should be implemented.

## 1.1 Audience

This amendment is intended primarily for card manufacturers and application developers developing GlobalPlatform card implementations.

It is assumed that the reader is familiar with smart cards and smart card production, and in particular familiar with the GlobalPlatform Card Specification [GPCS].

## 1.2 IPR Disclaimer

Attention is drawn to the possibility that some of the elements of this GlobalPlatform specification or other work product may be the subject of intellectual property rights (IPR) held by GlobalPlatform members or others. For additional information regarding any such IPR that have been brought to the attention of GlobalPlatform, please visit https://globalplatform.org/specifications/ip-disclaimers/. GlobalPlatform shall not be held responsible for identifying any or all such IPR, and takes no position concerning the possible existence or the evidence, validity, or scope of any such IPR.

## 1.3 References

**Table 1-1: Normative References**

| Standard / Specification | Description | Ref |
|---|---|---|
| GlobalPlatform Card Specification | GlobalPlatform Technology Card Specification v2.3.1 | [GPCS] |
| GP_TEN_053 | GlobalPlatform Technology Cryptographic Algorithm Recommendations (latest version) | [GPCAR] |
| ETSI TS 102 223 | Smart Cards; Card Application Toolkit (CAT), Release 10 | [102 223] |
| ETSI TS 102 226 | Smart cards; Remote APDU structure for UICC based applications, European Telecommunications Standards Institute Project Smart Card Platform (EP SCP), Release 10 | [102 226] |
| RFC 1035 | Domain Names - Implementation and Specification https://tools.ietf.org/html/rfc1035 | [DNS] |
| RFC 2616 | Hypertext Transfer Protocol – HTTP/1.1 | [HTTP] |
| RFC 2818 | HTTP over TLS | [HTTPS] |
| RFC 2246 | The TLS Protocol – Version 1.0 | [TLS 1.0] |
| RFC 4346 | The TLS Protocol – Version 1.1 | [TLS 1.1] |
| RFC 5246 | The TLS Protocol – Version 1.2 | [TLS 1.2] |
| RFC 8446 | The TLS Protocol – Version 1.3 | [TLS 1.3] |
| RFC 4366 | Transport Layer Security (TLS) Extensions | [TLS Extns] |
| RFC 4279 | Pre-Shared Key Cipher Suites for Transport Layer Security (TLS) | [PSK TLS] |
| RFC 5487 | Pre-Shared Key Cipher Suites for TLS with SHA-256/384 | [PSK 256] |
| RFC 4785 | Pre-Shared Key (PSK) Cipher Suites with NULL Encryption for Transport Layer Security (TLS) | [PSK NULL] |
| RFC 6655 | AES-CCM Cipher Suites for Transport Layer Security (TLS) | [PSK CCM] |
| OMA SCWS | Smartcard Web Server V1.1, Open Mobile Alliance™ | [OMA SCWS] |
| ISO/IEC 8825-1 | Information technology — ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER) | [8825-1] |
| Java Card Specifications | Java Card Specifications 3.0.x, Classic Edition, Oracle. | [JCS] |

## 1.4 Terminology and Definitions

Technical terms used in this document are defined in [GPCS].

## 1.5 Abbreviations and Notations

**Table 1-2: Abbreviations**

| Abbreviation | Meaning |
|---|---|
| AID | Application Identifier |
| AP | Application Provider |
| APDU | Application Protocol Data Unit |
| API | Application Programming Interface |
| APSD | Security Domain of the Application Provider |
| BIP | Bearer Independent Protocol |
| DNS | Domain Name System |
| FQDN | Fully Qualified Domain Name |
| HTML | Hypertext Markup Language |
| HTTP | Hypertext Transfer Protocol |
| HTTPS | Hypertext Transfer Protocol Secure |
| IP | Internet Protocol |
| ISD | Issuer Security Domain |
| KCV | Key Check Value |
| ME | Mobile Equipment |
| OTA | Over-The-Air |
| OTASD | Security Domain of the Over-The-Air platform operator |
| PIX | Proprietary Identifier extension |
| PSK TLS | Pre-Shared Key TLS |
| RAM | Remote Application Management |
| RAS | Remote Administration Server |
| RID | Resource Identifier |
| SCWS | Smart Card Web Server |
| SD | Security Domain |
| TAR | Toolkit Application Reference |
| TCP | Transport Control Protocol |
| TLS | Transport Layer Security |
| UDP | User Datagram Protocol |
| URI | Uniform Resource Identifier |

## 1.6 Revision History

GlobalPlatform technical documents numbered *n*.0 are major releases. Those numbered *n*.1, *n*.2, etc., are minor releases where changes typically introduce supplementary items that do not impact backward compatibility or interoperability of the specifications. Those numbered *n.n*.1, *n.n*.2, etc., are maintenance releases that incorporate errata and precisions; all non-trivial changes are indicated, often with revision marks.

**Table 1-3: Revision History**

| Date | Version | List of Modifications |
|---|---|---|
| Nov 2008 | 1.0 | Initial Release |
| June 2009 | 1.1 | **HTTP Header modification**<br>The "From" and "User-Agent" header fields are specified in the HTTP protocol ([HTTP]), but the content defined in version 1.0 for those headers were not compliant.<br>o Prefixed proprietary headers<br>The good practice of [HTTP] for custom headers is to prefix them by “X-”. All header names defined in this document (previously named Resume, Next-URI, Script-Status and Targeted-Application) are now prefixed by “X-Admin-”.<br>o "From" Header Field<br>[HTTP] specifies that the "From" request-header field, if given, shall contain an Internet e-mail address for the human user who controls the requesting user agent. Version 1.0 used the "From" header field in the HTTP post request to put the "Agent-ID" (identifier of the card). A custom “X-Admin-From” header field is now defined.<br>o "User-Agent" Header Field<br>[HTTP] specifies that the "User-Agent" request-header field contains information about the user agent originating the request. This is for statistical purposes, the tracing of protocol violations, and automated recognition of user agents for the sake of tailoring responses to avoid particular user agent limitations. In version 1.0, the "User-Agent" was used in the HTTP post request and in HTTP post response to identify the RAM over HTTP protocol. The "X-Admin-Protocol" header that will be used for the request and the response with the same value “globalplatform-remote-admin/1.0” is now defined.<br>**Content-Type Value**<br>[HTTP] only allows one slash in the value. Version 1.0 was inconsistent with this rule. A compliant Content-Type for POST request and response is now defined.<br>o POST request:<br>Content-Type: application/vnd.globalplatform.card-content-mgt-response;version=1.0 CRLF<br>o POST response:<br>Content-Type: application/vnd.globalplatform.card-content-mgt;version=1.0 CRLF<br>*(continues)* |

| Date | Version | List of Modifications |
|---|---|---|
| | 1.1<br>*(continued)* | **AID coding rules of the AID in the "X-Admin-Targeted-Application" header field is specified.**<br>**Agent-ID definition**<br>The value of the "Agent-ID" field is defined in the administration session triggering message or by the Security Domain parameters. In practice the remote admin server usually uses this field to identify the card instance (for example to keep an image of the card content) and not only the requesting application.<br>**Support of TLS protocol v1.1 and v1.2.**<br>**Support of Pre-Shared Key Cipher Suites for TLS with SHA-256.**<br>**Connection Parameters to configure the point to point TCP connection**<br>**Retry Policy**<br>Report mechanism has been added to have a status on the HTTP Administration session request.<br>o Report Failure Parameters.<br>These parameters allow an application to request the system to send a report through another communication channel than the one defined in this document.<br>o HTTPReportListener Interface.<br>This interface is added to notify the applet whether the requested HTTPAdministrationSession has completed successfully.<br>**HTTPAdministration Interface**<br>The object implementing this interface shall belong to the JCRE to have access to any object. This avoids requesting Global Arrays that are not always available. |
| March 2012 | 1.1.1 | • Added new section “Secure Channel Protocol Usage”.<br>• Clarified the meaning of “administration session”, the meaning of “communication breakdown”, and when the Retry Policy shall be used. See sections “Administration Session Start” and “Administration Session End”.<br>• Added precisions on the usage of PSK TLS keys. See section “Establishing a Secure Communication Channel”.<br>• Clarified the meaning of “command session”, and its relation with Secure Channel tunneling.<br>• Replaced section “PSK TLS key format” with section “Loading PSK TLS Keys” describing a suitable format for both PUT KEY and STORE DATA commands. |

| Date | Version | List of Modifications |
|---|---|---|
| May 2014 | 1.1.2 | • Section 3.3.4 – Clarified that a Security Domain may end an Admin Session if the Remote Administration Server has become non-responsive (i.e. an HTTP response is expected but none is received).<br>• Section 3.4.2 – Added precision that the Content-Type header should not be present when the HTTP response status is 204 (No Content). May be ignored or interpreted as an error.<br>• Section 3.4.2 – Added precision that the implementation shall not reject an HTTP POST response for the only reason that it contains an HTTP header field not described in this document.<br>• Section 3.7.1 – Added precisions regarding SD Admin Session Parameters, in particular (but not only) how parameters may be updated (using the STORE DATA) or audited (using the GET DATA).<br>• Section 3.7.4 – Clarified the meaning of the Retry Counter.<br>• Section 3.8.2 – New mandatory tag '95' (Key Usage) in DGI '00B9' + recommended value. New optional tag '96' (Key Access) in DGI '00B9'.<br>• New section 3.8.3 – Using the DEK of a PSK TLS Key Set. |
| May 2015 | 1.1.3 | This maintenance release provides some precisions regarding the implementation of the SCP '81' protocol and the HTTP admin protocol used in RAM over HTTP. In particular, the usage of the "`security-error`" status in HTTP POST requests becomes optional and can be replaced by the "`ok`" status. It also fixes some errors in the management of SD Administration Session Parameters using the STORE DATA and GET DATA commands (see section 3.7.1). |
| July 2020 | 1.1.3.4 | Committee Review |
| Feb 2021 | 1.1.3.12 | Member Review |
| July 2021 | 1.1.3.26 | Public Review |
| TBD | 1.2 | Public Release<br>• Updated for [TLS 1.3] changes which have been derived from RFC-8446.<br>• Updated with AEAD cipher suite for [TLS 1.2].<br>• Added optional capability to perform DNS Resolution to obtain the IP addresses of one or several Remote Administration Servers (RAS) from a fully qualified domain name (FQDN).<br>• Defined default parameter values. |

# 2 Use Cases and Requirements

This document defines a mechanism to handle Card Content Management as defined in [GPCS].

This document proposes a specification addendum to support the following requirements:

- It shall be possible to open an HTTPS connection between an Application Provider and its Security Domain (APSD).
- In this connection, the APSD acts as an HTTPS client, and the AP acts as an HTTPS server, called Remote Administration Server (RAS) in this document.
- This connection is used to exchange remote APDU format strings as specified in [102 226] with the APSD. It may also be used to send other content types, handled by other applications.
- The underlying transport protocol of this connection is out of scope of this specification.
- An intermediary OTA SD may be used. To ensure confidentiality, the targeted APSD may apply additional security to the exchanged remote APDU format strings.
- If supported, DNS resolution may be performed to obtain the IP addresses of one or several Remote Administration Servers (RAS) from a fully qualified domain name (FQDN).

This document also specifies an extension to the SCWS mechanisms which allows loading and installing applications using the same HTTPS channel. This enables the following additional use cases:

- Loading of static SCWS content as defined in [OMA SCWS]
- Loading of dynamic SCWS content generating applications
- Mapping these applications to a SCWS URL as defined in [OMA SCWS], within one session, all using the same HTTPS channel

# 3 Specification Amendments

## 3.1 PSK TLS Key Types

Two key types, '85' and '86', are defined in this amendment. Both key types refer to the master pre-shared secret that is used by all TLS cipher suites running in PSK mode to further derive required session key material. The difference between them resides only in the KCV computation method:

- For key type '85', the KCV shall be computed as the three most significant bytes of the SHA-1 digest of the key value.
- For key type '86', the KCV shall be computed as the three most significant bytes of the SHA-512 digest of the key value.

While support for key type '85' is mandatory, support for key type '86' remains optional.

## 3.2 Security Domain and Remote Administration Server

A Security Domain is responsible for establishing a connection with a remote HTTP server, called Remote Administration Server (RAS), in order to start (or resume) an Administration Session. Such an Administration Session is used to transport a set of APDU commands from the RAS to the Security Domain. If the card implements the OMA SCWS described in [OMA SCWS], the SD implements the OMA SCWS administration agent.

The connection is managed by the Security Domain and has the following characteristics:

- The physical link used for this connection is beyond the scope of the present document.
- It uses the industry standard security layer TLS protocol in order to secure communications (see RFC 2246 [TLS 1.0], RFC 4346 [TLS 1.1], RFC 5246 [TLS 1.2], and RFC 8446 [TLS 1.3]) and HTTPS (see RFC 2818 [HTTPS]). This specification references the TLS protocol as the GlobalPlatform Secure Channel Protocol '81' (SCP '81'). See section 3.3.2 for supported cipher suites.

During the Administration Session, the Security Domain:

- Acts as an HTTP Client and is in charge of managing connection establishment to the Remote Administration Server
- Encapsulates and transparently transports remote APDU format strings (as defined in [102 226])
- Is responsible for retry and reconnection management in case of communication breakdown

The Administration Session may be triggered either by external events or by internal events (i.e. internally generated by the card).

### 3.2.1 Secure Communication Configuration

For SCP '81', the “i” parameter (implementation options) is formatted as a 1-byte bitmap as defined in Table 3-1, indicating all the TLS versions supported by the Security Domain. A security domain may support one or multiple TLS versions.

**Table 3-1: Values of Parameter “i”**

| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Description |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | 1 | [TLS 1.0] supported |
| | | | | | | 1 | | [TLS 1.1] supported |
| | | | | | 1 | | | [TLS 1.2] supported |
| | | | | 1 | | | | [TLS 1.3] supported |
| | X | X | X | | | | | RFU (set to 0) |
| X | | | | | | | | Reserved |

NOTE: “i” is a sub identifier within an object identifier, and bit b8 is reserved for use in the structure of the object identifier according to ISO/IEC 8825-1 [8825-1].

A card implementation may support more TLS versions than those configured for a Security Domain via the “i” parameter. If a TLS “`ServerHello`” handshake message requests the usage of a TLS version not included in the set of TLS versions supported by the Security Domain that triggered the session, then the implementation shall abort the TLS handshake process. NOTE: According to [TLS 1.3] section D, the client shall send a “`protocol_version`” alert message and close the connection.

### 3.2.2 Updating the Secure Communication Configuration

The SCP '81' implementation options supported by a Security Domain may be updated using a STORE DATA command in DGI mode, conveying DGI 'C981' specifying a 2-byte value composed of:

- 1-byte SCP identifier ('81')
- 1-byte implementation options (“i” parameter)

The update shall be rejected if either the specified SCP is not currently supported by the Security Domain or the specified implementation options are not supported by the card implementation.

Notice that a successful update of these implementation options does not necessarily guarantee successful operation of the Secure Channel, as the new implementation options may require other data stored by the Security Domain to be updated.

## 3.3 Administration Protocol

### 3.3.1 Administration Session Start

An Administration Session starts when a SD receives an Administration Session Triggering Message (see section 3.7).

If the IP address of the Remote Administration Server (i.e. RAS IP address) can be determined from this triggering message and/or parameters stored by the SD/ISD (see sections 3.7 and 3.8), then the SD shall directly proceed according to section 3.3.1.1. Otherwise, if the implementation supports DNS Resolution and DNS Resolution Parameters can be determined from this triggering message and/or parameters stored by the SD/ISD (see sections 3.7 and 3.11), then the SD shall proceed according to section 3.3.1.2.

If an Administration Session Triggering Message is received while an Administration Session is already being processed, the SD shall store the new Administration Session triggering request and process it later upon completion of the current session. The number of Administration Sessions that may be buffered in this way remains out of scope, as well as the order in which they shall be processed if several requests have been stored. An implementation may attempt to process a queued Administration Session as soon as another Administration Session is set to wait and scheduled for later retry.

#### 3.3.1.1 Starting with a RAS IP Address

The SD shall set up a TCP connection with the RAS IP address and then establish secure communications using its own PSK TLS key (see section 3.3.2). It is assumed that the SD can resolve all the parameters needed to establish this connection and to manage its security from the triggering message and/or the parameters stored by the SD/ISD (see sections 3.7 and 3.8).

If the SD fails to connect to the RAS or if a communication breakdown occurs (i.e. a failure occurred over the communication channel), it shall retry later or terminate the session according to the Session Retry Policy described in 3.5. However, a failure of the underlying TLS session shall always be considered as a fatal error that shall fully terminate the Administration Session.

#### 3.3.1.2 Starting with a RAS FQDN

If the RAS IP address cannot be determined from the triggering message and/or parameters stored by the SD/ISD, and if the implementation supports DNS Resolution, then the SD shall perform the DNS Resolution procedure to obtain a list of RAS IP addresses from a Fully Qualified Domain Name of the Remote Administration Server (RAS FQDN). In this case:

- The RAS FQDN and other suitable DNS Resolution Parameters shall be determined from the triggering message and/or the parameters stored by the SD/ISD (see sections 3.7 and 3.11).
- The DNS Resolution procedure described in section 3.10 shall be applied to produce a suitable list of RAS IP addresses.

Once a list of RAS IP addresses has been produced:

- It is assumed that the SD may connect to any of the resolved RAS IP addresses, all corresponding to a Remote Administration Server able to serve the same Administration Session and contents. In particular, in case of session retry (see section 3.5), the SD may try to connect to any of the resolved RAS IP addresses.
- If the SD successfully connects to a RAS IP address, it shall then establish secure communications using its own PSK TLS key (see section 3.3.2). It is assumed that the SD can resolve all the parameters needed to establish this connection and to manage its security from the triggering message and/or the parameters stored by the SD/ISD (see sections 3.7 and 3.8).
- If the SD fails to connect to a RAS IP address, the SD shall retry to connect to the same RAS IP address after a specific time period and up to a specific number of times, as defined by the RAS IP Retry Policy (see section 3.8.7). This retry policy is independent and does not affect the retry counter of the Session Retry Policy globally defined for the Administration Session (see section 3.5). If all retries are exhausted, the SD shall try another RAS IP address (if any). See also Annex B.
- If several RAS IP addresses are available:
  - The SD shall try each address as specified in previous paragraphs until it successfully establishes a connection.
  - The first address tried by the SD shall be selected randomly in the list. To select and try another address, the SD shall then iterate over available addresses in order, starting from the first tried address and cycling over until it reaches that first tried address again. This algorithm balances the load over the available servers.

If all the RAS IP addresses have been tried unsuccessfully, the SD shall then try the same procedure using a Fallback RAS IP List (see section 3.11.8), if such a list can be provided for this RAS FQDN by the SD/ISD.

If no Fallback RAS IP List is available or all the Fallback RAS IP addresses have been tried unsuccessfully, the SD shall retry (or terminate) the session according to the Session Retry Policy (see section 3.5), and if retrying, shall start again with the list of RAS IP addresses initially produced by the DNS Resolution procedure.

See Annex B to get a global vision of how DNS resolution integrates within an Administration Session, and how multiple retry policies apply when DNS resolution is used.

### 3.3.2 Establishing a Secure Communication Channel

Once the communication channel has been set up, the Security Domain shall establish a secure communication channel with the Remote Administration Server.

The Security Domain processes the PSK TLS over this communication channel to enable mutual authentication, integrity, and possibly confidentiality, using one of the following cipher suites:

**Table 3-2: TLS Cipher Suites**

| TLS Version | Cipher Suites | As Defined in |
|---|---|---|
| TLS 1.0 and TLS 1.1 | TLS_PSK_WITH_3DES_EDE_CBC_SHA | [PSK TLS] |
| | TLS_PSK_WITH_AES_128_CBC_SHA | [PSK TLS] |
| | TLS_PSK_WITH_NULL_SHA | [PSK NULL] |
| TLS 1.2 | TLS_PSK_WITH_AES_128_CBC_SHA256 | [PSK 256] |
| | TLS_PSK_WITH_NULL_SHA256 | [PSK 256] |
| TLS 1.3 (operated in PSK mode) | TLS_AES_128_CCM_SHA256 | [TLS 1.3] |
| | TLS_AES_128_GCM_SHA256 | [TLS 1.3] |

An implementation is not required to support all the cipher suites listed above in all supported TLS versions. Consequently, the implementation may reject the choice made in the TLS "`Server Hello`" handshake message if it is deemed inconsistent or is not supported. The TLS "`Client Hello`" handshake message is allowed to present all the cipher suites supported by the implementation, irrespective of the TLS versions supported by the Security Domain that triggered the session.

Supporting cipher suites that are not listed above is allowed but out of scope of this specification.

The Key Version Number (KVN) and Key Identifier (KID) of the PSK TLS key, and the PSK Identity string that shall be used to initiate the PSK TLS session are read as part of Administration Session Security Parameters (see section 3.7).

The PSK TLS (SCP '81') key set consists of two keys: a PSK TLS key and a DEK (decryption/encryption) key. The DEK key may be used to decrypt or encrypt sensitive data using the `SecureChannel` interface (see section 3.3.3.1). It has the same KVN as the PSK TLS key, and a KID incremented by one. For example, when a PSK TLS session was opened using a PSK TLS key having a KVN of '40' and a KID of '01', then the DEK key that shall be used in this session is identified by a KVN of '40' and a KID of '02'.

The loading of new PSK TLS keys is described in section 3.9.

The Remote Administration Server shall support the Maximum Fragment Length Negotiation for TLS as defined in RFC 4366 [TLS Extns] and shall accept requests for a maximum fragment length down to 512 bytes. The Security Domain may use the Maximum Fragment Length Negotiation to request a maximum fragment length smaller than the default value of 16 Kbytes.

### 3.3.3 Fetching a Remote APDU Format String

Once the PSK TLS communication channel is established the Security Domain shall send an HTTP POST request in order to get a remote APDU format string. The targeted Security Domain shall verify the protection (if any) of each APDU read from the remote APDU format string as described in section 3.3.3.2.

When receiving the HTTP POST request from the Security Domain, the Remote Administration Server shall send an HTTP response, which encapsulates a remote APDU format string dedicated to a Security Domain. This dedicated Security Domain is defined as follows:

- If no "`X-Admin-Targeted-Application`" header is present in the HTTP POST response, then the targeted Security Domain is the one that provides the PSK TLS security of the communication channel.
- If a "`X-Admin-Targeted-Application`" header is present in the HTTP POST response, then the header value shall be read as the instance AID of the targeted Security Domain.

If requested, the Security Domain shall submit the remote APDU format string response in a new POST request to the Remote Administration Server over the PSK TLS secure channel.

The Remote Administration Server shall send the next remote APDU format string to the Security Domain over the PSK TLS channel, or send a final response requesting the end of the Administration Session in the POST response.

If the Security Domain receives a final response from the Remote Administration Server, it shall close the PSK TLS channel, and then close the underlying communication channel.

#### 3.3.3.1 Usage of the `SecureChannel` Interface

If the targeted Security Domain is handling the PSK TLS (SCP '81') secure channel session, the security of the script is successful and the following rules shall apply:

- `SecureChannel.getSecurityLevel()` may be used to verify the secure channel security level.
- `SecureChannel.processSecurity()` shall throw an ISO Exception with status code ISO7816.SW_INS_NOT_SUPPORTED.
- The `SecureChannel.unwrap()` method may be called and shall not return an error, but shall not perform any additional secure messaging processing.
- As the PSK TLS response will be secured implicitly according the PSK TLS security level, the `SecureChannel.wrap()` method may be called and shall not return an error, but shall not do any processing on the outgoing response message.
- `SecureChannel.encrypt()` and `SecureChannel.decrypt()` shall use the key having the same Key Version Number (KVN) and a Key Identifier (KID) incremented by one with respect to the key described in the Security Parameters of the current Administration Session (see section 3.8.2). The algorithm used is identified by the algorithm (3DES or AES) associated with the key. The CBC mode shall be used (with null ICV).
- The security level shall reflect the cipher suite used during the session:
  - AUTHENTICATED | C_MAC | C_DECRYPTION | R_MAC | R_ENCRYPTION
    - TLS_PSK_WITH_3DES_EDE_CBC_SHA
    - TLS_PSK_WITH_AES_128_CBC_SHA
    - TLS_PSK_WITH_AES_128_CBC_SHA256
    - TLS_AES_128_CCM_SHA256
  - AUTHENTICATED | C_MAC | R_MAC
    - TLS_PSK_WITH_NULL_SHA
    - TLS_PSK_WITH_NULL_SHA256
  - NO_SECURITY_LEVEL
    - SCP '81' not set up
- `SecureChannel.resetSecurity()` shall throw an `ISOException` with status code `ISO7816.SW_CONDITION_OF_USE_NOT_SATISFIED`.

If the targeted Security Domain is not handling the PSK TLS session, it shall apply its own secure channel to check the security of each command received in the remote APDU format string and the following rules shall apply:

- In this case the `SecureChannel.processSecurity()` method is used to set up the secure channel session.
- `SecureChannel.unwrap()` secures each APDU command string.
- The Security Domain shall explicitly wrap each command response of the remote APDU format string using its secure channel service `SecureChannel.wrap(byte[],short,short)`.

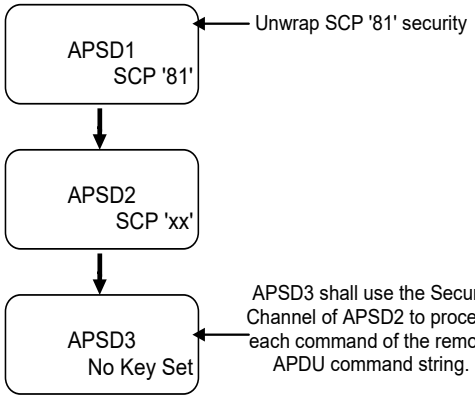
#### 3.3.3.2 Secure Channel Protocol Usage

When the targeted Security Domain is the one unwrapping the remote APDU command string, then the remote APDU command string is trusted and processed. Any attempt to initiate a Secure Channel session (according to another Secure Channel Protocol) within the remote APDU command string shall be rejected.

When the targeted Security Domain is not the one unwrapping the remote APDU command string, then the following rules apply:

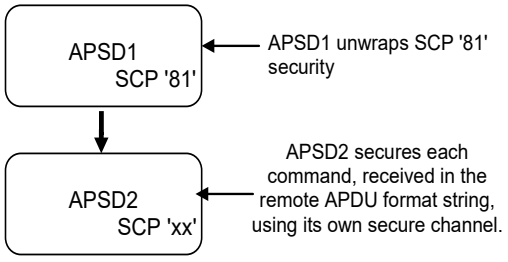
1. If the targeted Security Domain does not have any Secure Channel Key Set:
   a. If the Security Domain unwrapping the remote APDU command string is the Security Domain associated with the targeted Security Domain, then the remote APDU command string is trusted and processed by the targeted Security Domain.
   b. If the Security Domain unwrapping the remote APDU command string is not the Security Domain associated with the targeted Security Domain, then the remote APDU command string is not trusted and the targeted Security Domain shall request its associated Security Domain to verify the protection (SCP 'xx' with 'xx' in the range '01' to '7F') of the APDU commands received in the remote APDU command string.

**Figure 3-1: Targeted Security Domain without any Secure Channel Key Set**




2. If the targeted Security Domain has at least one complete Key Set for a Secure Channel Protocol, the remote APDU command string is not trusted, and:
   a. If the targeted Security Domain does not support SCP '81', then it shall use its own Secure Channel (SCP 'xx' with 'xx' in the range '01' to '7F') to verify the protection of the APDU commands received in the remote APDU command string.
   b. If the targeted Security Domain supports both SCP '81' and another Secure Channel Protocol (SCP 'xx' with 'xx' in the range '01' to '7F'), it shall use that other protocol to verify the protection of the APDU commands received within the remote APDU command string. Otherwise, the protection cannot be successfully verified and the APDU commands shall be rejected.

**Figure 3-2: Targeted Security Domain without SCP '81' Capability**



### 3.3.4 Administration Session End

An Administration Session ends:

- If all the HTTP messages sent by the Remote Administration Server (RAS) have been received and processed. The RAS explicitly ends the session by sending an HTTP response with no "`X-Admin-Next-URI`" header (see section 3.4.2), to which the Security Domain reacts by closing the Administration Session and the communication channel.
- Upon failure of the Secure Communication Channel (see section 3.3.2). In this case, the Security Domain immediately closes the Administration Session and the communication channel.
- If all retry policies have been applied and exhausted.

  If a communication breakdown occurs at some point during the HTTP dialog started with the RAS (NOTE: This situation would be reported as a specific event by the ME; see [102 223].), the Security Domain shall try to resume the Administration Session according to the Session Retry Policy described in section 3.5.

  The Security Domain may also decide to end the communication with the RAS and apply the Session Retry Policy if the RAS has become unresponsive (i.e. an HTTP response is expected but none is received). The RAS shall be considered unresponsive after expiration of the delay specified by the RAS Inactivity Timeout parameter (see section 3.8.8).

  When DNS Resolution is used, other retry policies apply when a communication breakdown occurs prior to the start of the HTTP dialog (see section 3.3.1.2).

## 3.4 Command Format

### 3.4.1 HTTP POST Request of Security Domain

The POST request is used by the Security Domain to fetch remote APDU format strings and to transmit response strings.

The POST request shall have the following format:

```
POST <URI> HTTP/1.1 CRLF
Host: <Administration Host> CRLF
X-Admin-Protocol: globalplatform-remote-admin/1.0 CRLF
X-Admin-From: <Agent ID> CRLF
[Content-Type: application/vnd.globalplatform.card-content-mgt-
response;version=1.0 CRLF]
[Content-Length: xxxx CRLF] or [Transfer-Encoding: chunked CRLF]
[X-Admin-Script-Status: <script-status> CRLF]
[X-Admin-Resume: true CRLF]
CRLF
[body-with-previous-response-string]
```

- The URI, the "`X-Admin-From`" value and the "`Host`" value to be used are defined in the Administration Session Triggering Message or by Security Domain parameters.
- The first POST request of a new Administration Session shall not contain any optional header field and no body.
- The "`X-Admin-Script-Status`" header value is used to return the delivery status of the previous remote APDU format string. The possible values are defined as follows:
  - "`ok`": This value is used if the previous remote APDU format string has been successfully delivered. A response string shall be sent.
  - "`unknown-application`": This value is used if the application targeted by the previous remote APDU format string could not be found. No response string shall be sent.
  - "`not-a-security-domain`": This value is used if the application targeted by the previous remote APDU format string is not a Security Domain. No response string shall be sent.
  - "`security-error`": This value may be used to indicate that the Security Domain targeted by the previous remote APDU format string is not able to check its security. In this case, the APDU format string may have been fully or partially processed but no response string shall be sent.

    The usage of the "`security-error`" status value is optional. Instead of using this status value, the targeted Security Domain may use the "`ok`" status value and send a response string.

    For instance, if the targeted Security Domain is not the one unwrapping the remote APDU command string and does not support any of SCP 'xx' with 'xx' in the range '01' to '7F', the "`security-error`" status value could be used to indicate that the Security Domain is not able to verify the protection of the commands received in the remote APDU command string.
- If the Administration Session is resumed from a previous interrupted session, the Security Domain shall use the "`X-Admin-Resume`" header with the value "`true`" in the first POST request of the resumed session. The "`X-Admin-Resume`" header shall not be used in the following POST requests. See section 3.5, Session Retry Policy.
- If a response string is to be sent, the Security Domain shall use:

- o "Content-Type" header with the value "application/vnd.globalplatform.card-content-mgt-response;version=1.0"
- o "Content-Length" header with the exact length of the body in bytes or "Transfer-Encoding" header with the value "chunked".
- o A body with the complete response string of the previous remote APDU format string, in binary format. The chunked Transfer-Encoding may be used. Expanded Remote response structure format as defined in [102 226] shall be used.

### 3.4.2 HTTP POST Response of Remote Administration Server

The POST response is used by the Remote Administration Server to transmit the next remote APDU format string to a Security Domain and possibly to inform about the next URI that must be used to request the following admin command.

The POST response shall have the following format:

```
HTTP/1.1 200 OK CRLF [or HTTP/1.1 204 No Content CRLF]
X-Admin-Protocol: globalplatform-remote-admin/1.0 CRLF
[X-Admin-Next-URI: <next-URI> CRLF]
[Content-Type: application/vnd.globalplatform.card-content-mgt;version=1.0
CRLF]
[X-Admin-Targeted-Application: <security-domain-AID> CRLF]
[Content-Length: xxxx CRLF] or [Transfer-Encoding: chunked CRLF]
CRLF
[body-with-command-string]
```

- The Remote Administration Server shall use a successful status (200 OK) if the response contains a body else it shall use the 204 (No Content) if no body is sent.
- If Content-Type and X-Admin-Protocol are inconsistent, the session shall be closed.
- If the Remote Administration Server was not able to process the last HTTP POST request (unexpected URI, invalid header…), then it shall use an error status. The Security Domain shall close the Administration Session.
- If a "X-Admin-Next-URI" header is present in the response, the Security Domain shall use the given URI in the next POST request. The "X-Admin-Next-URI" header may be replaced by the "SCWS-Next-URI" header without any functional modification.
- If no "X-Admin-Next-URI" header is present in the response and if the body is empty, this is the final response of the Remote Administration Server and the Administration Session shall be closed.
- If no "X-Admin-Next-URI" header is present in the response and if the body is not empty, the remote APDU format string shall be processed, no response string shall be returned to the Remote Administration Server, and the Administration Session shall be closed.
- If the Remote Administration Server indicates status 204 (No Content), then the "Content-Type" header should not be present. If present, it may be ignored or interpreted as an error.
- If the Remote Administration Server has remaining remote APDU format string to forward to a Security Domain it shall use a body with:
  - o "Content-Type" header with the value "application/vnd.globalplatform.card-content-mgt;version=1.0"
  - o "Content-Length" header with the exact length of the body in bytes, or "Transfer-Encoding" header with the value "chunked"

  - A body with a remote APDU format string in binary format to be forwarded to a Security Domain. The chunked Transfer-Encoding may be used. Expanded Remote command structure format as defined in [102 226] shall be used.

- Optionally, "`X-Admin-Targeted-Application`" header field with the representation of the targeted Security Domain AID as header value, if the targeted Security Domain is not the one in charge of the PSK TLS security.
  - The AID shall be coded as follows; //aid/<RID>/<PIX>, where <RID> and <PIX> are the two components of the application AID. All the bytes of the RID and PIX including any leading 0 byte values shall be represented in the character string notation.
  - A RID byte string is 5 bytes in length. Its character string equivalent shall be exactly 10 characters in length.
  - A PIX byte string can be from 0 to 11 bytes in length. A PIX byte string of N bytes in length shall have an equivalent character string representation of exactly 2*N characters in length.

Notice that the implementation shall not reject an HTTP POST response for the only reason that it contains an HTTP header field not described in this document.

### 3.4.3 Interworking with the SCWS

If RAM over HTTP on a card is used together with SCWS administration as defined in [OMA SCWS], the following additional provisions shall apply:

- The PSK TLS secure channel to be used for RAM over HTTP may also be opened as defined in [OMA SCWS].
- Independent of how the PSK TLS channel was opened, sequential switching between RAM over HTTP and SCWS administration shall be supported as defined in the next two bullet points.
- To switch from SCWS management to RAM over HTTP, the empty response that ends SCWS management shall be replaced by a response from the Remote Administration Server having content as defined in this document. This shall start an Administration Session as defined in this document.
- To switch from RAM over HTTP to SCWS management, the final response from the Remote Administration Server defined in this document shall be replaced by a response from the SCWS Remote Administration Server having content as defined for the SCWS. This shall end an Administration Session as defined in this document.

## 3.5 Session Retry Policy

The Security Domain handling the Administration Session is responsible for its completion and for attempting a number of retries in case of unexpected interruption.

The Session Retry Policy applies when at least one RAS IP address is available and:

- Either all attempts to connect to all available RAS IP addresses have failed
  - If the Administration Session started with a unique RAS IP address (see section 3.3.1.1), the SD shall try to connect to that unique RAS IP address and the RAS IP Retry Policy (see section 3.8.7) shall also apply to that RAS IP address.
  - If the Administration Session started with an FQDN and therefore uses DNS Resolution (see section 3.3.1.2), the SD shall try to connect to the resolved RAS IP addresses and the RAS IP Retry Policy (see section 3.8.7) shall apply to each RAS IP address.
- Or a communication breakdown occurs during the HTTP dialog started with a RAS

Notice that a failure of the Secure Communication Channel (see section 3.3.2) for security reasons shall always immediately end the Administration Session; i.e. with no retries (see also section 3.3.4).

The Session Retry Policy is composed of a waiting period (between two attempts) and a maximum number of retry attempts. These parameters are described in section 3.8.3. The Administration Session shall be ended if the maximum number of retries defined by the Session Retry Policy has been reached.

In case of communication breakdown during the HTTP dialog started with a RAS:

- If the Security Domain had received an HTTP response and processed a complete script before the breakdown occurred, and if a next URI had been provided, the Security Domain shall try to resume the HTTP dialog with the next HTTP request using the specified next URI and with the "`X-Admin-Resume: true`" header present. See section 3.4.1.
- At any other point in time after the first HTTP request has been sent, the Security Domain shall try to resume the HTTP dialog by repeating the last HTTP request with the "`X-Admin-Resume: true`" header present. See section 3.4.1.

If several administration requests have been started and need a retry, retries should be handled independently of each other (e.g. not block the other retry attempts if the current one is not successful).

## 3.6 Command Session

A command session consists of one or several remote APDU format string(s) for a single targeted Security Domain. An Administration Session may transport several command sessions for several targeted Security Domains.

At the beginning of the Administration Session, a command session is implicitly started, targeting the triggered Security Domain. Subsequently, a new command session shall be started if the Security Domain targeted by the current HTTP POST response is not the same as the one targeted by the previous HTTP POST response. That means:

- The value of the header "`X-Admin-Targeted-Application`" has changed,
- Or the value of the header "`Content-Type`" has changed,
- Or the previous HTTP POST response contains an "`X-Admin-Targeted-Application`" header while the current one does not contain this header,
- Or the current HTTP POST response contains an "`X-Admin-Targeted-Application`" header while the previous one does not contain this header.

During the command session, all APDU commands received in the APDU format string are forwarded to and processed by the targeted Security Domain.

If the targeted Security Domain is not the one unwrapping the remote APDU command string, and uses its own Secure Channel (SCP 'xx' with 'xx' in the range '01' to '7F') to verify the protection of the APDU commands received in the remote APDU command string, then that Secure Channel shall be terminated upon, and only upon, one of the following events: error detected in the protection of an APDU command, establishment of a new Secure Channel session, or end of the command session (as described hereafter).

A command session shall be closed if one of the following conditions occurs:

- The communication channel is closed.
- A new command session is started for another Targeted Application.
- A Card Reset occurs.

The internal notifications needed to implement the mechanisms described above are out of the scope of this document.

## 3.7 Administration Session Triggering Message

An Administration Session starts when a Security Domain receives an Administration Session Triggering Message, which may result from:

- An external event, for example a message sent by a remote or off-card entity
- An internal event, for example a timer
- An application using a dedicated API method (see section 3.12)

If the triggering message is sent to an SD, it may be sent to the TAR that processes the Expanded Remote Application data format according to [102 226].

The triggering message shall contain TLV-encoded Administration Session Triggering Parameters and DNS Resolution Parameters (if applicable) that the triggered SD shall use to perform DNS resolution (if applicable), set up the connection with the Remote Administration Server, establish the required security level, and prepare the content of the first request.

Any parameter needed to start the Administration Session and not present in this triggering message shall be completed with the corresponding parameter of the triggered SD. If the triggered SD can't provide this parameter, it shall be completed with the corresponding parameter of the ISD. If the ISD can't provide this parameter and no default value ultimately applies, the triggering message shall be rejected. These completion rules apply to all TLV-encoded parameters and sub-parameters. The parameters of the ISD are defined by the Card Issuer. See section 3.12 for a description of default values that apply when parameters cannot be provided by the ISD.

The following table identifies the TLV-encoded parameters that shall be used in the Administration Session Triggering Message.

**Table 3-3: Administration Session Triggering Parameters**

| Tag | Length | Name | | | | | | Presence |
|---|---|---|---|---|---|---|---|---|
| '81' | 0-n | Administration session triggering parameters | | | | | | Mandatory |
| | | **Tag** | **Len** | **Name** | | | | |
| | | '83' | 1-n | Administration Session Parameters | | | | Optional |
| | | | | **Tag** | **Len** | **Name** | | |
| | | | | '84' | 1-n | Connection Parameters | | Optional |
| | | | | '85' | 1-n | Security Parameters | | Optional |
| | | | | 'A5' | 1-n | Extended Security parameters (for TLS 1.3 or higher) | | Optional |
| | | | | '86' | 1-n | Session Retry Policy Parameters | | Optional |
| | | | | '89' | 1-n | HTTP POST Parameters | | Optional |
| | | | | | | **Tag** | **Len** | **Name** |
| | | | | | | '8A' | 1-n | Administration Host parameter |
| | | | | | | '8B' | 1-n | Agent ID parameter |
| | | | | | | '8C' | 1-n | Administration URI parameter |
| | | 'B3' | 1-n | DNS Resolution Parameters | | | | Optional |
| | | | | **Tag** | **Len** | **Name** | | |
| | | | | 'D6' | 1-n | FQDN: Fully Qualified Domain Name | | Optional |
| | | | | 'FA' | 1-n | DNS Connection Parameters | | Optional |

Administration Session Parameters and DNS Resolution Parameters are described in sections 3.8 and 3.11.

Tags '85' (Security Parameters) and 'A5' (Extended Security Parameters) are mutually exclusive. If both tags '85' and 'A5' are missing, security parameters shall be completed by looking within the triggered SD first for tag 'A5' then for tag '85' (if tag 'A5' was not found).

If a RAS IP address (i.e. “Data Destination Address” data object) is present within Connection Parameters (or if one can be completed according to completion rules), then DNS Resolution Parameters shall be ignored (whether they are present, can or cannot be completed). See section 3.3.1 and notes in section 3.8.1.

## 3.8 Security Domain Administration Session Parameters

The Administration Session Parameters of a SD may be set using tag '85' or 'A5' inside application specific install parameters (during installation) or within the STORE DATA command in TLV mode (during personalization) as defined in [GPCS].

**Table 3-4: TLV Security Domain Administration Session Parameters**

| Tag | Length | Name | | | | | Presence |
|---|---|---|---|---|---|---|---|
| '85' or 'A5' | 1-n | Security Domain Administration Session Parameters | | | | | Optional |
| | | **Tag** | **Len** | **Name** | | | |
| | | '84' | 1-n | Connection parameters | | | Optional |
| | | '85' | 1-n | Security parameters | | | Optional |
| | | 'A5' | 1-n | Extended Security parameters (for TLS 1.3 or higher) | | | Optional |
| | | | | **Tag** | **Len** | **Name** | |
| | | | | '85' | 1-n | Security parameters | Mandatory |
| | | | | … | … | … | Optional |
| | | | | '85' | 1-n | Security parameters | Optional |
| | | '86' | 1-n | Session Retry Policy parameters | | | Optional |
| | | '89' | 1-n | HTTP POST parameters | | | Optional |
| | | | | **Tag** | **Len** | **Name** | |
| | | | | '8A' | 1-n | Administration Host parameter | Optional |
| | | | | '8B' | 1-n | Agent ID parameter | Optional |
| | | | | '8C' | 1-n | Administration URI parameter | Optional |
| | | '8A' | 7 | RAS IP Retry Policy parameters | | | Optional |
| | | '8B' | 5 | RAS Inactivity Timeout parameter | | | Optional |

The STORE DATA command with tag '85' (in TLV mode) shall be used to create or update the complete set of SD Administration Session Parameters. Parameters may be removed all at once using tag '85' with a null length.

The STORE DATA command with tag 'A5' (in TLV mode) shall be used to create, update or remove sub-TLVs of SD Administration Session Parameters: '84', '85' or 'A5', '86', '89', '8A', '8B'. One or more of these sub-TLVs may be present. If a sub-TLV is not present, no creation/update shall be performed for the corresponding SD Administration Session Parameter. If a sub-TLV is present but has no value (i.e. length set to 0), then the corresponding SD Administration Session Parameter shall be removed from the SD. Otherwise, the value of the sub-TLV shall replace the value of the corresponding SD Administration Session Parameter within the SD.

As the value of tag '85' or 'A5' may be quite long, the implementation should support receiving this TLV spanning across (at least) two consecutive STORE DATA commands. Note that it is not possible to provide such long value at the time of Security Domain installation as Java Card Specifications [JCS] limits the length of Installation Parameters to 127 bytes.

The full set of SD Administration Session Parameters defined above may be retrieved using the GET DATA command as defined in [GPCS] with P1-P2 set to '00 85'. The response to the GET DATA command shall contain tag '85' encapsulating the full set of SD Administration Session Parameters (i.e. parameters that are currently stored).

A single SD Administration Session Parameter may be retrieved using the GET DATA command as defined in [GPCS] with P1-P2 set to '00 A5' and with a data field containing a Tag List indicating which parameter is requested. The Tag List (TLV '5C') shall contain only a single tag. Possible tags are: '84', '85' or 'A5', '86', '89', '8A', '8B'. The response to the GET DATA command shall contain tag 'A5' encapsulating only the TLV corresponding to the requested SD Administration Session Parameter.

### 3.8.1 Connection Parameters

The Connection Parameters TLV embeds all the needed parameters to establish a point to point TCP connection between the Administration Agent and the Remote Administration Server.

**Table 3-5: Connection Parameters**

| Description | Length |
|---|---|
| Connection parameters tag | 1 |
| Length (A) | 1 or 2 |
| Set of any comprehension TLV needed to open the TCP connection. | A |

This TLV contains the COMPREHENSION-TLV data objects that are defined for the OPEN CHANNEL proactive command in ETSI TS 102 223 [102 223] and that are required to establish the TCP connection between the Administration Agent and the Remote Administration Server over a BIP channel.

NOTE 1: Within Connection Parameters, the “Data Destination Address” data object specifies the RAS IP address.

NOTE 2: If the SD also stores DNS Resolution Parameters (see section 3.11), then its Connection Parameters should not specify any RAS IP address (i.e. the “Data Destination Address” data object should not be present), otherwise a RAS IP address missing in a Session Triggering Message targeting this SD (see section 3.7) would always be resolved into that stored RAS IP address and therefore DNS Resolution would never be triggered.

NOTE 3: If the ISD was to store Connection Parameters, these would be considered as the default Connection Parameters for the card. As a consequence, if the implementation supports DNS Resolution, the Connection Parameters of the ISD should not specify any RAS IP address (i.e. “Data Destination Address” data object), otherwise a RAS IP address missing in any Session Triggering Message (see section 3.7) may always be resolved into that default RAS IP address and therefore DNS Resolution would never be triggered.

### 3.8.2 Security Parameters

The security parameters are defined as follows:

**Table 3-6: Security Parameters**

| Description | Length | Presence |
|---|---|---|
| Security parameters tag | 1 | Mandatory |
| Length | 1, 2, or 3 | Mandatory |
| Length of PSK Identity | 1 | Mandatory |
| PSK Identity | 1-n | Mandatory |
| Length of Key version/Key identifier | 1 | Mandatory |
| Key version/Key identifier | 2 | Mandatory |
| SHA Type | 1 | Optional |

- PSK Identity is a string defined in [PSK TLS]. The administration agent shall support a PSK Identity length of at least 32 bytes.
- Key version and Key Identifier identify the PSK TLS key to be used for PSK TLS exchanges. The first byte is the key version number of the key. The second byte is the key identifier of the key.
- The SHA type is required if the PSK is configured for [TLS 1.3]. As per [TLS 1.3] section 4.2.11, for PSK key establishment, the client should advertise at least one cipher suite indicating a Hash associated with the PSK. If not present, it shall default to SHA-256. The SHA type shall be coded as described in [GPCS] appendix H.4, in the same way as “LFDBH algorithms”.

Multiple PSK identities may be specified using the structured Extended Security Parameters TLV instead of a simple Security Parameters TLV. As described in [TLS 1.3], multiple PSKs may be configured for TLS 1.3. This does not apply however to earlier versions of TLS.

### 3.8.3 Session Retry Policy Parameters

The Session Retry Policy parameters are defined as follows:

**Table 3-7: Session Retry Policy Parameters**

| Description | Length | Presence |
|---|---|---|
| Session Retry Policy parameters tag | 1 | Mandatory |
| Length | 1 | Mandatory |
| Retry counter | 2 | Mandatory |
| Retry waiting delay | 5 | Mandatory |
| Retry report failure | Var. | Optional |

- Retry counter: Maximum number of reconnection attempts
- Retry waiting delay: Definition of the time to wait between two reconnection attempts. This parameter is in the same format as the “timer” comprehension TLV defined in [102 223].

The implementation is expected to retry as soon as possible after expiration of the waiting delay. Indeed, it may not always be possible to retry immediately, e.g. if another session is being processed or network connectivity is unavailable.

Although this parameter shall be provided in "timer" format, there is no requirement for an implementation to actually use a timer (as defined in [102 223]) to implement the waiting delay.

- Retry Report Failure is typically used to send a message using another communication channel in case of an abort of an administration request. This data shall be encoded as described in [OMA SCWS] section 13.3.2.9.7.

If a retry has been scheduled and a card reset occurs, the card may not be able to correctly enforce the retry waiting delay. Therefore, following a card reset, an implementation may ignore this delay and retry as soon as possible after completion of the terminal initialization procedures (e.g. when network connectivity is detected).

### 3.8.4 Administration Host Parameter

This TLV defines the "`Host`" header value to be used by the Security Domain when sending a POST request. It is defined as follows:

**Table 3-8: Host Parameter**

| Description | Length |
|---|---|
| Administration Host parameter tag | 1 |
| Length | 1, 2, or 3 |
| "`Host`" header value | 1-n |

### 3.8.5 Agent Id Parameter

This TLV defines the "`X-Admin-From`" header value to be used by the Remote Administration Server to identify the requester when receiving a POST request. It is defined as follows:

**Table 3-9: Agent Id Parameter**

| Description | Length |
|---|---|
| Agent Id parameter tag | 1 |
| Length | 1, 2, or 3 |
| "`X-Admin-From`" header value | 1-n |

### 3.8.6 Administration URI Parameter

This TLV defines the URI value to be used by the Security Domain when sending the first POST request of the Administration Session. It is defined as follows:

**Table 3-10: Administration URI Parameter**

| Description | Length |
|---|---|
| Administration URI parameter tag | 1 |
| Length | 1, 2, or 3 |
| URI value | 1-n |

### 3.8.7 RAS IP Retry Policy Parameters

The RAS IP Retry Policy TLV contains the retry policy parameters for retrying a RAS IP address.

**Table 3-11: RAS IP Retry Policy Parameters**

| Description | Length | Presence |
|---|---|---|
| RAS IP Retry Policy tag | 1 | Mandatory |
| Length | 1 | Mandatory |
| Retry counter | 2 | Mandatory |
| Retry waiting delay | 5 | Mandatory |

The encoding of the retry counter and retry waiting delay is described in section 3.8.3.

### 3.8.8 RAS Inactivity Timeout Parameter

The RAS Inactivity Timeout TLV contains the delay after which a RAS may be considered unresponsive.

**Table 3-12: RAS Inactivity Timeout Parameter**

| Description | Length | Presence |
|---|---|---|
| RAS Inactivity Timeout tag | 1 | Mandatory |
| Length | 1 | Mandatory |
| Inactivity timeout | 5 | Mandatory |

The encoding of the inactivity timeout is the same as the retry waiting delay described in section 3.8.3.

## 3.9 Loading PSK TLS Keys

PSK TLS keys shall be loaded using either a PUT KEY command or the STORE DATA command. The capability to load a PSK TLS key using the STORE DATA command remains optional.

When sending PSK TLS keys, the following rules apply:

- PSK TLS keys shall be sent encrypted:
  - Before ciphering a PSK TLS key, the PSK TLS key shall be padded with as few (if any) random bytes to fill the last block required by the ciphering algorithm.
  - The padded PSK TLS key shall be ciphered with the Data Encryption Key (DEK) and associated encryption algorithm as defined by the Secure Channel Protocol used to protect the PUT KEY or STORE DATA command.
- A key check value shall be computed as described in section 3.1.

### 3.9.1 PSK TLS Key Loading with the PUT KEY Command

The key data field of a PSK TLS key shall be coded as follows:

**Table 3-13: PSK TLS Key Data Field**

| Name | Length | Value |
|---|---|---|
| Key type | 1 byte | '85' or '86' |
| Length of PSK key data | Variable | '01' – '80', or<br>'81 80' – '81 FF', or<br>'82 01 00' – '82 FF FF' |
| Length of PSK key (in bytes) | Variable | '01' – '80', or<br>'81 80' – '81 FF', or<br>'82 01 00' – '82 FF FF' |
| Ciphered PSK key | Variable | 'xxxx…' |
| Key Check Value length ('03') | 1 byte | '03' |
| Key Check Value | 3 bytes | 'xxxx…' |

### 3.9.2 PSK TLS Key Format for the STORE DATA Command

If the STORE DATA command is used to load PSK TLS keys, the CRT defined in the table below shall be used to describe the PSK TLS key sent to the Security Domain.

**Table 3-14: Data Content for DGI '00B9' – PSK TLS Key**

| Tag | Length | Description | Presence |
|---|---|---|---|
| 'B9' | Var. | CRT tag (CT) | Mandatory |
| '95' | 1 | Key Usage Qualifier according to [GPCS] section 11.1.9 | Mandatory |
| '96' | 1 | Key Access according to [GPCS] section 11.1.10 | Optional |
| '80' | 1 | '85' or '86' | Mandatory |
| '81' | 1 or 2 | Key Length, in bytes (unsigned integer value) | Mandatory |
| '82' | 1 | Key Identifier | Mandatory |
| '83' | 1 | Key Version Number | Mandatory |
| '84' | 3 | Key check value | Mandatory |

For tag '95' (Key Usage), a value of '3C' should be used.

DGI '8113' shall immediately follow DGI '00B9' and is used to populate the PSK TLS key:

**Table 3-15: Data Content for DGI '8113' – PSK TLS Key Value**

| DGI | Length | Data Content | Encrypt |
|---|---|---|---|
| '8113' | Var. – multiple of 8 | PSK TLS key | Yes |

### 3.9.3 Using the DEK of a PSK TLS Key Set

The DEK key of a PSK TLS Key Set (see section 3.3.2) shall be used to encrypt sensitive data (e.g. secret key values) according to the following rules:

- If the DEK algorithm is 3DES:
  - If the length of the sensitive data is not an integer multiple of the block length (8 bytes), padding of arbitrary bytes shall be appended prior to encryption to fill the last block.
  - 3DES CBC encryption with ICV set to zero is performed across the sensitive data and the result of the encryption of each block becomes part of the encrypted data. This encrypted data then becomes part of the “clear text” data field in the command message.
- If the DEK algorithm is AES:
  - If the length of the sensitive data is not an integer multiple of the block length (16 bytes), padding of arbitrary bytes shall be appended prior to encryption to fill the last block.
  - AES CBC encryption with ICV set to zero is performed across the sensitive data and the result of the encryption of each block becomes part of the encrypted data. This encrypted data then becomes part of the “clear text” data field in the command message.

The on-card decryption of key data is the exact opposite of the above operation.

## 3.10 DNS Resolution Procedure

DNS Resolution is to be performed when an Administration Session is started with a RAS FQDN instead of a RAS IP address. In this case, the purpose of the DNS Resolution procedure is to resolve the RAS FQDN into one or more RAS IP addresses. If several RAS IP addresses are obtained, then the Security Domain shall try to connect to each of them until the connection is successful.

### 3.10.1 RAS IP List Cache

The card may implement cache storage of RAS IP addresses previously resolved for a given RAS FQDN. If so and if a (non-empty) cached RAS IP address list is available for the RAS FQDN used by the current Administration Session, the SD may use this cached RAS IP address list before or after the DNS Resolution procedure as described hereafter:

- If the “Force DNS Resolution” parameter has not been set (or has been set to '00'; see section 3.11.1), then the SD shall first try the cached RAS IP addresses as described in section 3.3.1.2. If all cached RAS IP addresses are tried unsuccessfully, the SD shall then continue the DNS Resolution procedure (starting from section 3.10.2) to retrieve a fresh RAS IP address list. As this latter operation would typically refresh the content of the cache, the implementation may choose not to try the freshly retrieved RAS IP address list if it hasn’t changed (i.e. same as the list previously stored in the cache, which has already been tried at this point).
- If the “Force DNS Resolution” parameter has been set to '01', then the SD shall first try to perform the full DNS Resolution procedure (starting from section 3.10.2) in order to retrieve a fresh RAS IP address list. If the DNS Resolution procedure fails (i.e. no valid RAS IP address list could be retrieved), the SD shall then try the cached RAS IP address list.

The content of the cache may be initialized (or replaced, if the cache is not empty) using the RAS IP List Cache parameter (see section 3.11.5).

### 3.10.2 Fetching a DNS IP List

If the IP address of a DNS Server (i.e. DNS IP address) can be determined from the DNS Connection Parameters found in the Administration Session Triggering Message and/or parameters stored by the SD/ISD (see sections 3.7 and 3.11) then the SD shall directly proceed according to section 3.10.4 using this single DNS IP address. Otherwise, the SD shall first establish a list of IP addresses corresponding to one or several DNS Servers.

If supported by the Mobile Equipment (ME) in which the card is being used, the SD shall try to retrieve such a list from the ME using an OPEN CHANNEL proactive command as described in [102 223]. If the ME is unable to provide an answer at the time the request is made, the SD shall retry after a specific time period and up to a specific number of times, as defined by the DNS-From-ME Retry Policy (see section 3.11.2). This retry policy is independent and does not affect the retry counter of the Session Retry Policy globally defined for the Administration Session (see section 3.5).

If the above procedure is not supported or all retries have been exhausted, the SD shall then try to use a DNS IP List Cache (if supported, see section 3.10.3).

### 3.10.3 DNS IP List Cache

The card may implement cache storage of DNS IP addresses previously retrieved from the ME (if supported).

If such cache storage is supported, then:

- If the SD failed retrieving DNS IP addresses from the ME (for any reason), the SD shall try the cached DNS IP addresses (as described in section 3.10.4).
- If the SD successfully retrieved a fresh DNS IP address list from the ME, the SD shall try that freshly retrieved list and the content of the cache shall be updated accordingly.

The content of the cache may be initialized (or replaced, if the cache is not empty) using the DNS IP List Cache parameter (see section 3.11.6).

### 3.10.4 DNS Queries to DNS Server

A DNS query is sent to a DNS Server and is used to retrieve a list of RAS IP addresses for the RAS FQDN applicable for the session, which may be found in the Administration Session Triggering Message and/or parameters stored by the SD/ISD (see sections 3.7 and 3.11).

The SD shall establish a UDP connection with the DNS server at a given DNS IP address and send clear-text query frames. No TLS security is used here. In return, the DNS Server answers with clear-text response frames.

A first query shall be sent to query IPv6 addresses for the FQDN (query type: AAAA). Then a second query shall be sent to query IPv4 addresses for the (same) FQDN (query type: A). The overall attempt shall be considered successful if a non-empty RAS IP address list was successfully built from the responses to the two queries.

The format of DNS queries and responses is fully described in [DNS].

If the SD fails to establish an UDP connection with the DNS server, or the DNS server does not answer any of the queries, or any of the responses is malformed, the SD shall retry with the same DNS IP address after a specific time period and up to a specific number of times, as defined by the DNS IP Retry Policy (see section 3.11.3). This retry policy is independent and does not affect the retry counter of the Session Retry Policy globally defined for the Administration Session (see section 3.5). If all retries have been exhausted, the SD shall try another DNS IP address (if any). See also Annex B.

If several DNS IP addresses are available:

- The SD shall try each address as specified in previous paragraphs until the attempt can be considered successful. If the implementation supports a RAS IP List Cache (see section 3.10.1), an entry shall be located or allocated in this cache for this session's RAS FQDN and initialized to the freshly retrieved list.
- The first DNS IP address tried by the SD shall be selected pseudo-randomly in the list. To select and try another address, the SD shall then iterate over available addresses in order, starting from the first tried address and cycling over until it reaches that first tried address again. This algorithm balances the load over the available servers.

If all DNS IP addresses have been tried unsuccessfully, the SD may then try to use a RAS IP List Cache (if supported, see section 3.10.1).

## 3.11 Security Domain DNS Resolution Parameters

The DNS Resolution Parameters of a SD may be set using tag 'F4' or 'F5' inside application specific install parameters (during installation) or within the STORE DATA command in TLV mode (during personalization) as defined in [GPCS].

**Table 3-16: TLV Security Domain DNS Resolution Parameters**

| Tag | Length | Name | | | Presence |
|---|---|---|---|---|---|
| 'F4' or 'F5' | 1-n | Security Domain DNS Resolution Parameters | | | Optional |
| | | **Tag** | **Length** | **Name** | |
| | | 'D0' | 1 | Force DNS Resolution | Optional |
| | | 'D1' | 7 | DNS-From-ME Retry Policy | Optional |
| | | 'D2' | 7 | DNS IP Retry Policy | Optional |
| | | 'D6' | 1-n | FQDN: Fully Qualified Domain Name | Optional |
| | | 'F7' | 1-n | RAS IP List Cache | Optional |
| | | 'F9' | 1-n | DNS IP List Cache | Optional |
| | | 'FA' | 1-n | DNS Connection Parameters | Optional |
| | | 'FB' | 1-n | Fallback RAS IP List | Optional |

The STORE DATA command with tag 'F4' (in TLV mode) shall be used to create or update the complete set of SD DNS Resolution Parameters. Parameters may be removed all at once using tag 'F4' with a null length.

The STORE DATA command with tag 'F5' (in TLV mode) shall be used to create, update or remove sub-TLVs of SD DNS Resolution Parameters: 'D0', 'D1'…'FB'. One or more of these sub-TLVs may be present. If a sub-TLV is not present, no creation/update shall be performed for the corresponding SD Administration Session Parameter. If a sub-TLV is present but has no value (i.e. length set to 0), then the corresponding SD DNS Resolution Parameter shall be removed from the SD. Otherwise, the value of the sub-TLV shall replace the value of the corresponding SD DNS Resolution Parameter within the SD.

As the value of tag 'F4' (resp. 'F5') may be quite long, the implementation should support receiving this TLV spanning across (at least) two consecutive STORE DATA commands. Note that it is not possible to provide such long value at the time of Security Domain installation as Java Card Specifications [JCS] limits the length of Installation Parameters to 127 bytes.

Sub-tags 'F7' and 'FB' shall be present only if sub-tag 'D6' is present, as such RAS IP lists wouldn't make sense if not associated with a particular FQDN.

Tag 'F4' (resp. 'F5') shall be used in the GET DATA command to retrieve SD DNS Resolution Parameters in the same way as tag '85' (resp. 'A5') is used to retrieve SD Administration Session Parameters. See section 3.8.

### 3.11.1 Force DNS Resolution

The “Force DNS Resolution” TLV shall be used to specify the priority of using the DNS Resolution procedure over a cached RAS IP address list (if implemented and not empty).

If a RAS IP List Cache is implemented (as described in section 3.10.1):

- If this parameter is set to '01', the SD shall always try to perform the DNS Resolution procedure first, even if a non-empty cached RAS IP address list is available. Notice that successful DNS Resolution would typically refresh the content of this cache.
- If this parameter is set to '00', the SD shall try to use the cached RAS IP address list first (if not empty).

If no RAS IP List Cache is implemented, this parameter has no effect and shall be ignored.

### 3.11.2 DNS-From-ME Retry Policy Parameters

The DNS-From-ME Retry Policy TLV contains the retry policy parameters for retrying retrieval of DNS IP addresses from the ME.

**Table 3-17: DNS-From-ME Retry Policy Parameters**

| Description | Length | Presence |
|---|---|---|
| DNS-From-ME Retry Policy tag | 1 | Mandatory |
| Length | 1 | Mandatory |
| Retry counter | 2 | Mandatory |
| Retry waiting delay | 5 | Mandatory |

The encoding of the retry counter and retry waiting delay is described in section 3.8.3.

### 3.11.3 DNS IP Retry Policy Parameters

The DNS IP Retry Policy TLV contains the retry policy parameters for retrying a DNS IP address.

**Table 3-18: DNS IP Retry Policy Parameters**

| Description | Length | Presence |
|---|---|---|
| DNS IP Retry Policy tag | 1 | Mandatory |
| Length | 1 | Mandatory |
| Retry counter | 2 | Mandatory |
| Retry waiting delay | 5 | Mandatory |

The encoding of the retry counter and retry waiting delay is described in section 3.8.3.

### 3.11.4 Fully Qualified Domain Name (FQDN)

The FQDN TLV contains the Fully Qualified Domain Name of the Remote Administration Server (RAS).

**Table 3-19: Fully Qualified Domain Name (FQDN)**

| Description | Length |
|---|---|
| FQDN tag | 1 |
| Length (A) | 1 |
| Fully Qualified Domain Name in QNAME format as specified in [DNS] | A |

### 3.11.5 RAS IP List Cache

The RAS IP List Cache TLV embeds one or several RAS IP addresses and is used to initialize the RAS IP List Cache (see section 3.10.1) associated with the specified FQDN (see section 3.11.4).

**Table 3-20: RAS IP List Cache**

| Description | Length |
|---|---|
| RAS IP List Cache tag | 1 |
| Length (A) | 1 or 2 |
| Sequence (in any order) of ETSI IP TLV structures:<br>o '3E 05 21 xx xx xx xx' for IPv4<br>o '3E 11 57 xx xx … xx' for IPv6<br>NOTE: Tag 'BE' may be used instead of '3E'. | A |

If the RAS IP List Cache is not supported, this parameter has no effect and shall be ignored.

As the content of the cache may change over time (e.g. depending on the results of DNS resolution), the value of this parameter as later retrieved by the GET DATA command shall reflect the current content of the cache, hence providing a mean to audit the current content of the RAS IP List Cache.

### 3.11.6 DNS IP List Cache

The DNS IP List Cache TLV embeds the IP addresses of one or several DNS Servers and is used to initialize the DNS IP List Cache (see section 3.10.3).

The list of IP addresses shall be coded in the same way as for the RAS IP List Cache TLV (see Table 3-20).

If the DNS IP List Cache is not supported, this parameter has no effect and shall be ignored.

The DNS IP List Cache tag may also be used with the GET DATA command to audit the content of the cached list, which may change over time (e.g. depending on the result of DNS resolution).

NOTE: If this parameter hasn't been personalized or the DNS IP List Cache is not supported, and if the DNS-From-ME mechanism is not supported by the ME (Host Device), then the Administration Session Triggering Message shall include DNS Connection Parameters indicating a suitable DNS IP address for the DNS Resolution procedure to start.

### 3.11.7 DNS Connection Parameters

The DNS Connection Parameters TLV embeds all the needed parameters to establish a point to point UDP connection between the Administration Agent and the DNS Server.

**Table 3-21: Connection Parameters**

| Description | Length |
|---|---|
| DNS Connection Parameters tag | 1 |
| Length (A) | 1 or 2 |
| Set of any comprehension TLV needed to open the UDP connection. | A |

This parameter contains the COMPREHENSION-TLV data objects that are defined for the OPEN CHANNEL proactive command in ETSI TS 102 223 [102 223] and that are required to establish the UDP connection between the Administration Agent and the DNS Server over a BIP channel.

Within DNS Connection Parameters, the “Data Destination Address” data object would correspond to a DNS IP address and should not be present if a DNS IP List Cache is also present (see section 3.11.6) or if it is intended for the DNS IP address list to be retrieved from the ME as described in section 3.10.2. However, it may be present in the Administration Session Triggering Message (see section 3.7) if it is the intention of the triggering entity to force the usage of a particular DNS IP address.

### 3.11.8 Fallback RAS IP List

The Fallback RAS IP List TLV embeds one or several RAS IP addresses to be used if all the RAS IP addresses resolved for the specified FQDN (see section 3.11.4) were tried unsuccessfully.

The list of IP addresses shall be coded in the same way as for the RAS IP List Cache TLV (see Table 3-20).

## 3.12 Default Parameter Values

This section describes default parameter values (or behaviors) that shall apply whenever the ISD cannot provide the value of a given parameter (e.g. parameter not defined by the Card Issuer).

If a given parameter is required, and if no value can be found for this parameter (neither in the triggered SD, nor in the ISD), and no default value is defined (None), then the Administration Session Triggering Message shall be rejected (see section 3.7).

**Table 3-22: Default Administration Session Parameters**

| Tag | Name | | Default |
|---|---|---|---|
| '84' | Connection parameters | | None |
| '85' | Security parameters | | None |
| 'A5' | Extended Security parameters | | None |
| '86' | Session Retry Policy parameters | | No Retry |
| '89' | HTTP POST parameters | | None |
| | '8A' | Administration Host parameter | None |
| | '8B' | Agent ID parameter | None |
| | '8C' | Administration URI parameter | None |
| '8A' | RAS IP Retry Policy parameters | | No Retry |
| '8B' | RAS Inactivity Timeout parameter | | No Timeout |

**Table 3-23: Default DNS Resolution Parameters**

| Tag | Name | Default |
|---|---|---|
| 'D0' | Force DNS Resolution | False |
| 'D1' | DNS-From-ME Retry Policy | No Retry |
| 'D2' | DNS IP Retry Policy | No Retry |
| 'D6' | FQDN: Fully Qualified Domain Name | None |
| 'F7' | RAS IP List Cache | Empty List |
| 'F9' | DNS IP List Cache | Empty List |
| 'FA' | DNS Connection Parameters | None |
| 'FB' | Fallback RAS IP List | Empty List |

# 4 API for Administration Session Triggering

This document introduces new services, available in the `org.globalplatform` package beginning with version 1.3, in order to:

- Request the triggering of an Administration Session.

  The HTTP Administration service is accessible as a uniquely registered Global Service and can be retrieved using the **`FAMILY_HTTP_ADMINISTRATION (0x84)`** constant (identifying the HTTP Administration service family) and a service ID of **`0x00`**, as shown in the following call to the API:

  ```
  GPSystem.getService(null,(short)(FAMILY_HTTP_ADMINISTRATION<<8))
  ```

- Be notified of the outcome of the Administration Session triggering request.

# Annex A Examples

## A.1 Nominal Case

First request sent by the Security Domain:

```
POST /server/adminagent?cmd=1 HTTP/1.1 CRLF
Host: 172.96.0.1 CRLF
X-Admin-Protocol: globalplatform-remote-admin/1.0 CRLF
X-Admin-From: 0123456789 CRLF
CRLF
```

Command that shall be executed by the Security Domain in charge of the PSK TLS security:

```
HTTP/1.1 200 OK CRLF
X-Admin-Protocol: globalplatform-remote-admin/1.0 CRLF
X-Admin-Next-URI: /server/adminagent?cmd=2 CRLF
Content-Type: application/vnd.globalplatform.card-content-mgt;version=1.0
CRLF
Content-Length: xxxx CRLF
CRLF
[command-string]
```

Return of a command response:

```
POST /server/adminagent?cmd=2 HTTP/1.1 CRLF
Host: 172.96.0.1 CRLF
X-Admin-Protocol: globalplatform-remote-admin/1.0 CRLF
X-Admin-From: 0123456789 CRLF
Content-Type: application/vnd.globalplatform.card-content-mgt-
response;version=1.0 CRLF
Content-Length: xxxx CRLF
X-Admin-Script-Status: ok CRLF
CRLF
[response-string]
```

Last response of Remote Administration Agent, communication shall be closed:

```
HTTP/1.1 204 No Content CRLF
X-Admin-Protocol: globalplatform-remote-admin/1.0 CRLF
CRLF
```

## A.2 Nominal Case with an Intermediary Actor

First request sent by the OTA Security Domain:

```
POST /server/adminagent?cmd=1 HTTP/1.1 CRLF
Host: 172.96.0.1 CRLF
X-Admin-Protocol: globalplatform-remote-admin/1.0 CRLF
X-Admin-From: 0123456789 CRLF
CRLF
```

Command that shall be executed by another Security Domain (Application Provider Security Domain):

```
HTTP/1.1 200 OK CRLF
X-Admin-Protocol: globalplatform-remote-admin/1.0 CRLF
X-Admin-Next-URI: /server/adminagent?cmd=2 CRLF
Content-Type: application/vnd.globalplatform.card-content-mgt;version=1.0
CRLF
X-Admin-Targeted-Application: //aid/A000000018/0001 CRLF
Content-Length: xxxx CRLF
CRLF
[secured-command-string]
```

Return of a command response:

```
POST /server/adminagent?cmd=2 HTTP/1.1 CRLF
Host: 172.96.0.1 CRLF
X-Admin-Protocol: globalplatform-remote-admin/1.0 CRLF
X-Admin-From: 0123456789 CRLF
Content-Type: application/vnd.globalplatform.card-content-mgt-
response;version=1.0 CRLF
Content-Length: xxxx CRLF
X-Admin-Script-Status: ok CRLF
CRLF
[response-string]
```

Last response of Remote Administration Agent, communication shall be closed:

```
HTTP/1.1 204 No Content CRLF
X-Admin-Protocol: globalplatform-remote-admin/1.0 CRLF
CRLF
```

## A.3 Error Case

First request sent by the OTA Security Domain:

```
POST /server/adminagent?cmd=1 HTTP/1.1 CRLF
Host: 172.96.0.1 CRLF
X-Admin-Protocol: globalplatform-remote-admin/1.0 CRLF
X-Admin-From: 0123456789 CRLF
CRLF
```

Command that shall be executed by Application Provider Security Domain:

```
HTTP/1.1 200 OK CRLF
X-Admin-Protocol: globalplatform-remote-admin/1.0 CRLF
X-Admin-Next-URI: /server/adminagent?cmd=2 CRLF
Content-Type: application/vnd.globalplatform.card-content-mgt;version=1.0
CRLF
X-Admin-Targeted-Application: //aid/A000000018/0001 CRLF
Content-Length: xxxx CRLF
CRLF
[secured-command-string]
```

The targeted Security Domain may send the following response if it is not able to process the security of the commands received in the remote APDU format string (see section 3.4.1 for a description of the "`security-error`" status):

```
POST /server/adminagent?cmd=2 HTTP/1.1 CRLF
Host: 172.96.0.1 CRLF
X-Admin-Protocol: globalplatform-remote-admin/1.0 CRLF
X-Admin-Script-Status: security-error CRLF
X-Admin-From: 0123456789 CRLF
CRLF
```

## A.4 Communication Breakdown Case

Resume an Administration Session after a communication breakdown:

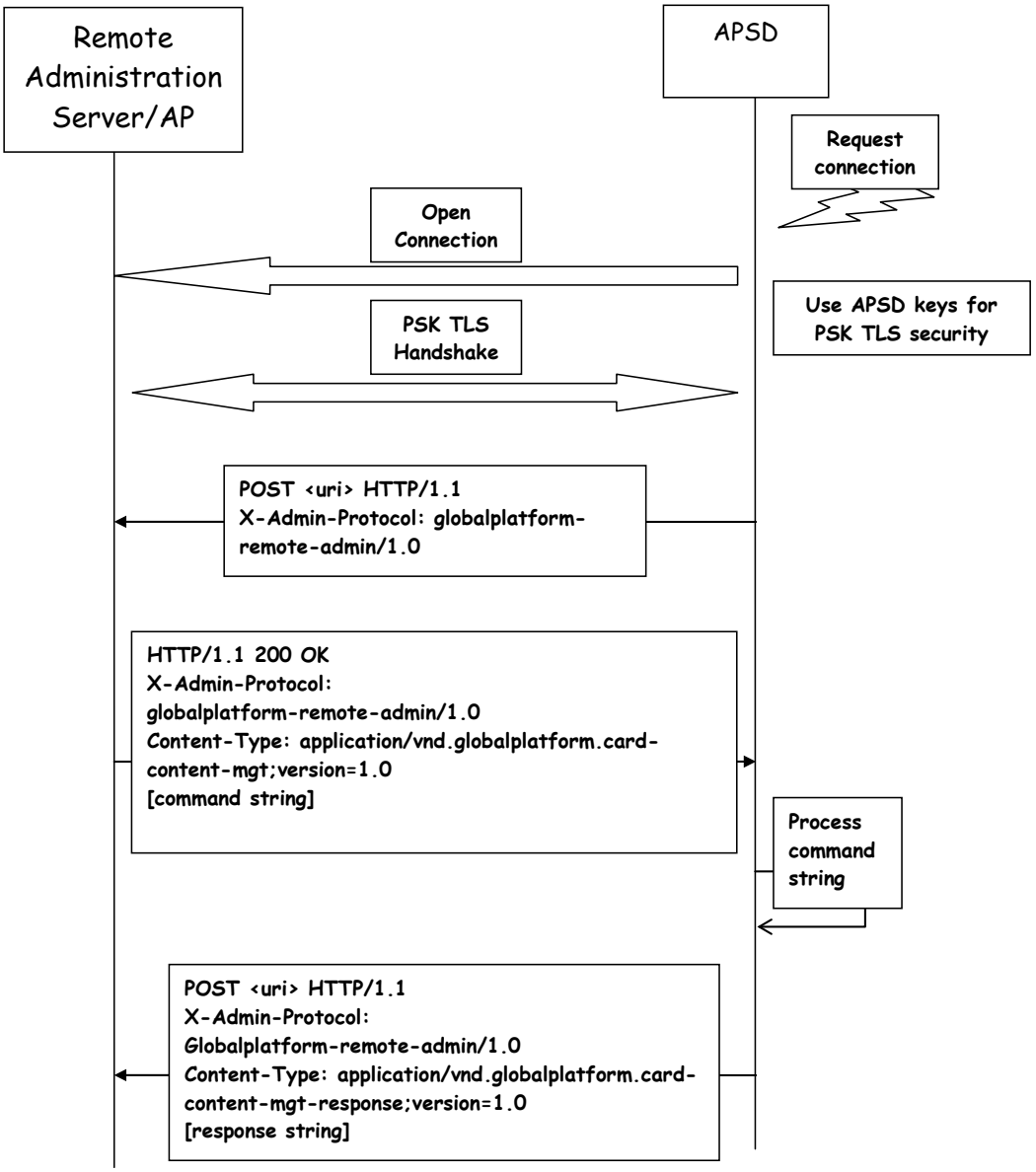
```
POST /server/adminagent?cmd=3 HTTP/1.1 CRLF
Host: 172.96.0.1 CRLF
X-Admin-Protocol: globalplatform-remote-admin/1.0 CRLF
X-Admin-From: 0123456789 CRLF
X-Admin-Resume: true CRLF
CRLF
```

## A.5 Communication Flow

The actors and on-card components involved in this scenario are

- The Application Provider (AP) owning a Remote Administration Server
- The Security Domain of the Application Provider (APSD), compliant with [102 226], and having PSK TLS keys

**Figure A-1: Communication Flow between an AP (owning a RAS) and its APSD**

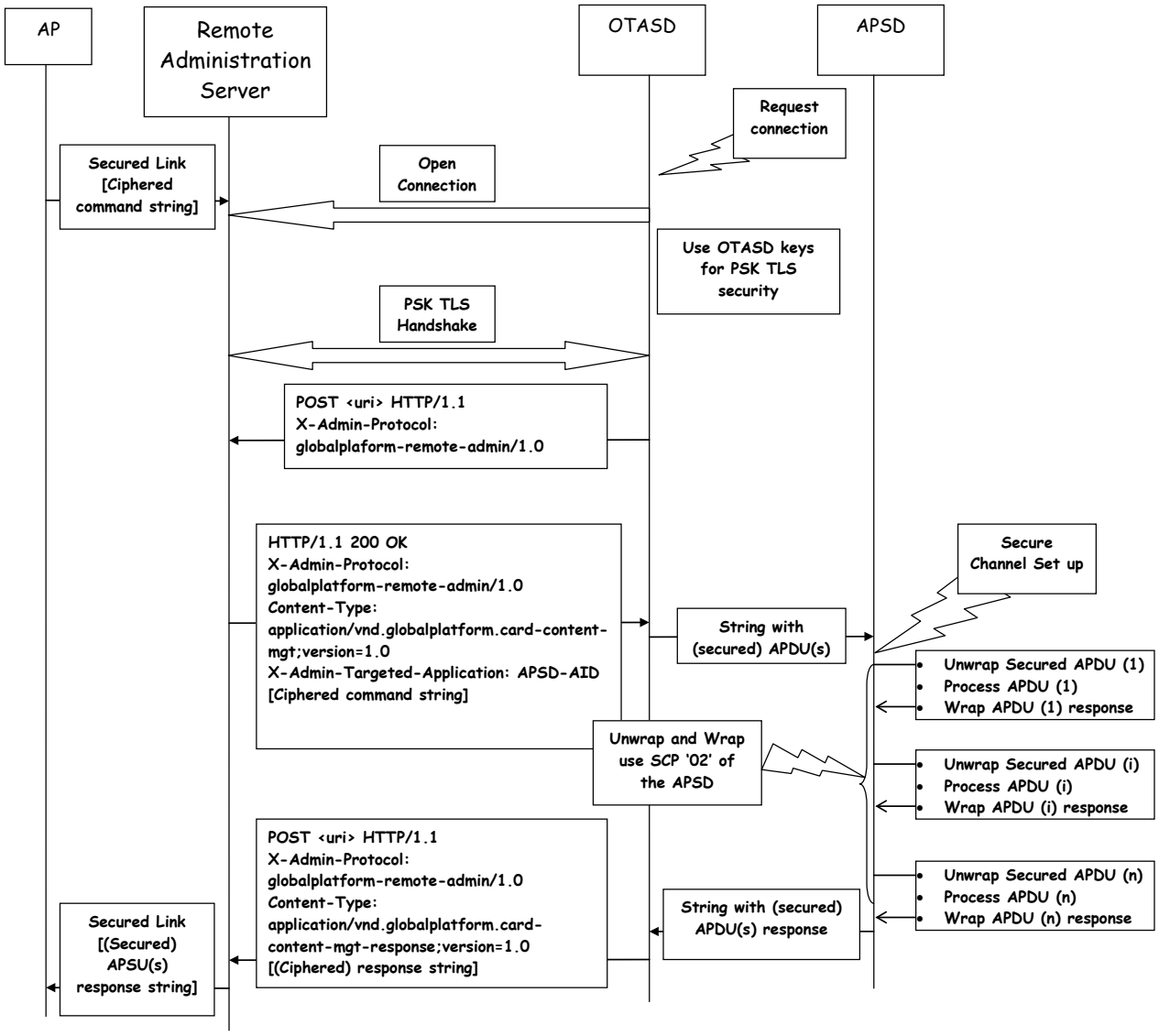


## A.6 Communication Flow through an Intermediary Actor

The actors and on-card components involved in this scenario are:

- The Application Provider (AP)
- The Remote Administration Server, owned by another entity
- The Security Domain in charge of the PSK TLS security, having PSK TLS keys (OTASD)
- The Security Domain of the Application Provider (APSD), compliant with [102 226], and if required supporting SCP '02' for securing the APDUs

**Figure A-2: Communication Flow between an AP and its APSD through a 3rd Party RAS**



# Annex B Administration Session with DNS Resolution

This annex contains flow diagrams showing how DNS resolution integrates within an Administration Session and how multiple retry policies apply when DNS resolution is used. For reference, see the following sections describing the use of the different retry policies: section 3.5 (Session Retry Policy), section 3.3.1.2 (RAS IP Retry Policy), section 3.10.2 (DNS-From-ME Retry Policy), and section 3.10.4 (DNS IP Retry Policy). Each retry policy maintains its own retry counter and uses its own retry period.

**Figure B-1: Administration Session started using DNS Resolution [MAIN]**

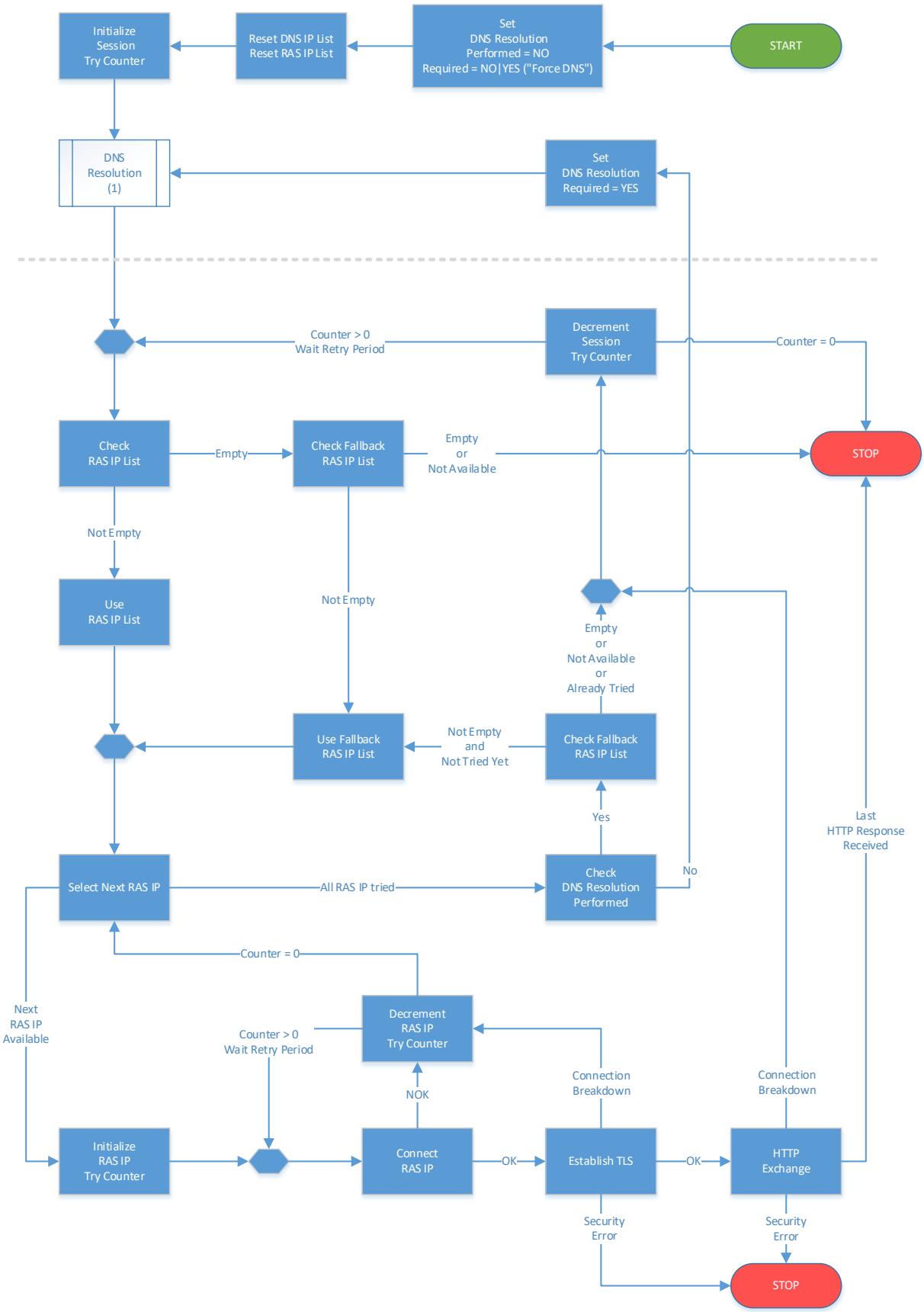


**Figure B-2: DNS Resolution [DNS1]**

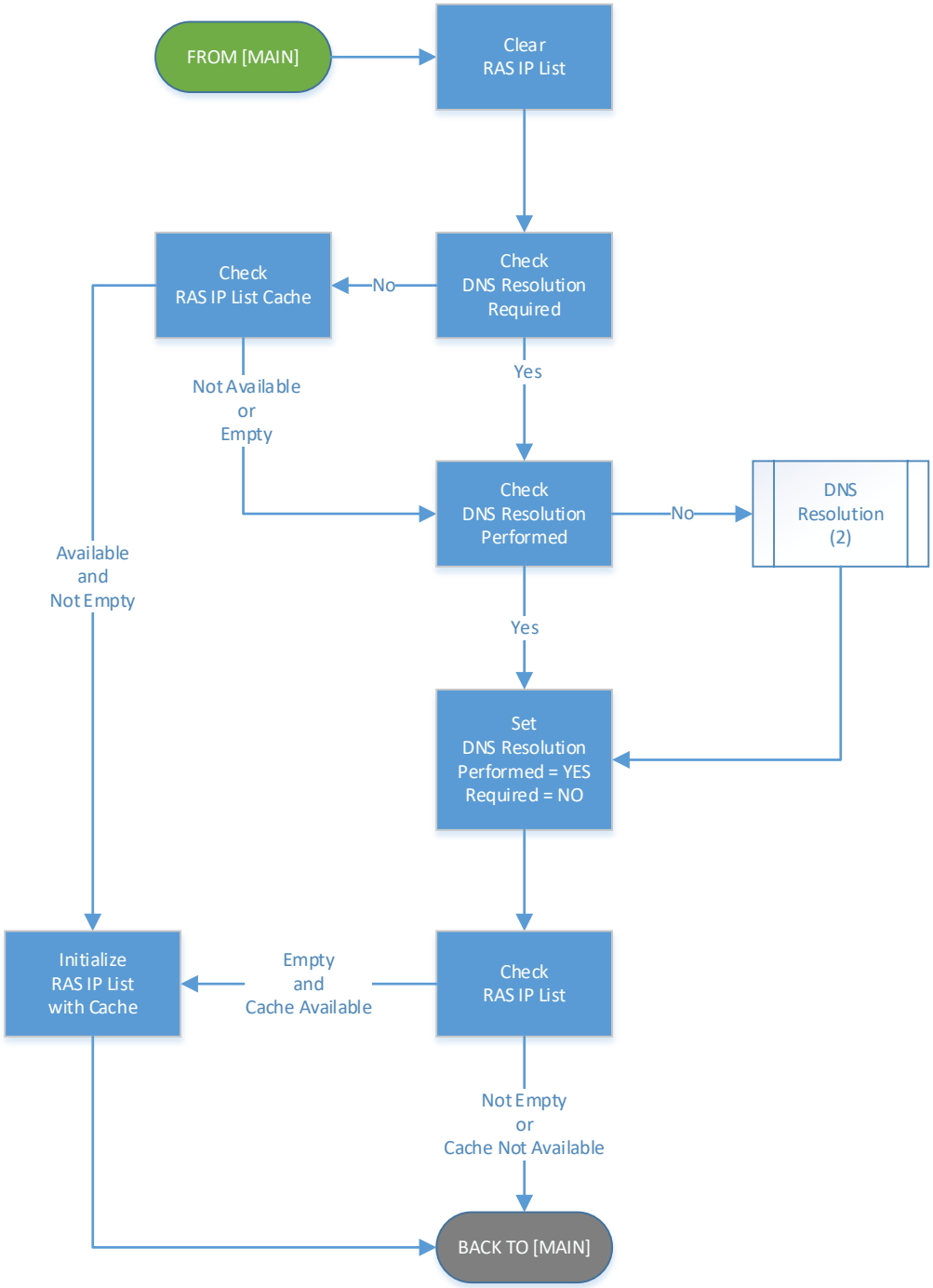


**Figure B-3: DNS Resolution [DNS2]**

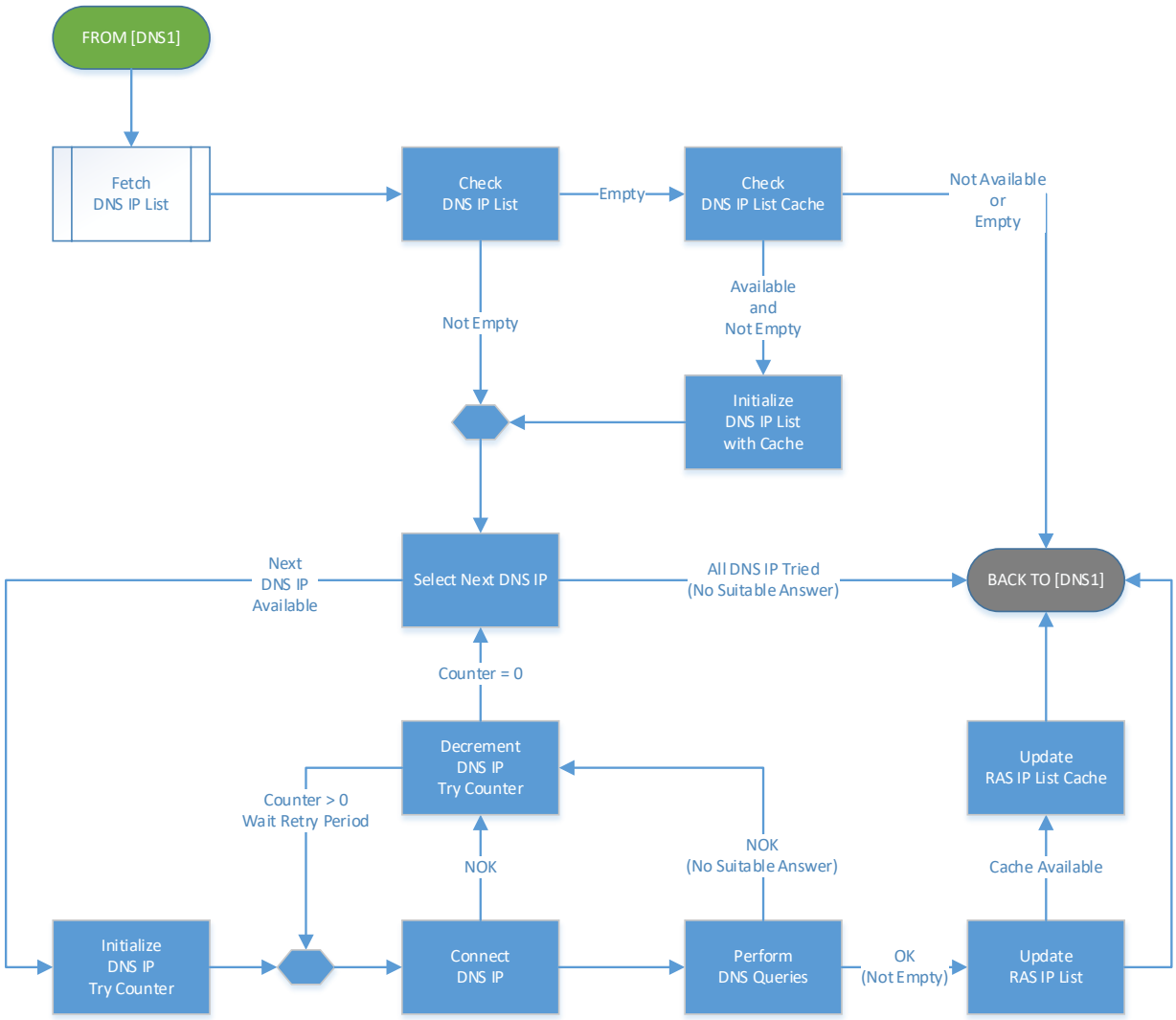


**Figure B-4: Fetching DNS IP List**