ES2+ Interface Standard and Proprietary APIs Specification_v002.03

Table of contents

1. ES2+ Proprietary API

2.1 ES2+ Profile Generation API

Objective: This document is designed to understand the generate profile API at real time upon request of customer.

Description: This API will trigger when customer wants to generate the profile in real time.

Below are the technical specifications of the generate profile API provided by Monty Mobile. This includes input parameters, output parameters, possible error messages and any restrictions.

Path

https://<domain name>/gsma/rsp2/es2plus/ generateProfile

Mandatory Parameters:

API Implementation at Customer End:

•   The customer must implement these APIs at their end to initiate the request.

ES2+ Interface Implementation:

•   For sending the download.order request after profile updates, the customer must implement the ES2+ interface at their end.

Variable Agreement:

•   Prior discussion and agreement between the customer and us are mandatory. This includes determining which variables will be shared and which ones should be generated by SM-DP+. The data definition will be configured accordingly to generate the values.

Adherence to Format:

•   The format specified by Monty Mobile is mandatory. Customers must follow this format precisely when sending requests to SM-DP+.

Input Batch Name Requirement:

•   It is mandatory to include the Input Batch name when sending the request.

Providing Variables and Values:

•   Customers must provide a list of variables along with their corresponding values when sending the request.

These parameters are crucial for the smooth operation of the system, ensuring consistency and accuracy in data transmission and processing.

Input Parameters:

In addition to the standard header parameter sent in the standard ES2+ APIs the below should be sent:

Input data name	Description	Type	MOC
input_batch_name	Name of the input batch to which the profile will be linked	String	M
profile_variables	Object containing key value pairs of variables to be used in profile generation (iccid, imsi, ki, opc …)	Object	M
encrypted_variables	List of keys (names) of variables that were sent in an encrypted format	Array of strings	C

NOTE 1: In case any of the variables are encrypted the encrypted_variables parameter is mandatory to specify the names of those variables.

NOTE 2: Encryption should be through the K4 key shared with Monty, and only for AES keys.

NOTE 3: Only imsi (imsi2, imsi3 in case needed), ki and opc can be sent in an encrypted format.

The available variables to send using the "profile_variables" parameter are the following:

```
[

"iccid", "ki", "opc", "imsi", "imsi2", "imsi3", "pin1", "pin2", "second_pin_appl1", "second_pin_appl2",
"puk1", "puk2", "second_puk_appl1", "second_puk_appl2", "adm1", "adm2", "adm3", "adm4", "adm5",
"acc", "msisdn",

"kic1", "kid1", "kik1", "kic2", "kid2", "kik2" ,"kic03->0F", "kid03->0F", "kik03->0F"

]
```

Table 1 Variables options.

```
{

"type”: "object",

"properties”:

{

"input_batch_name”:

{

"type”: "string",
```

```
"description": "name of the input batch that the generated profile should be linked to (this value is
provided by Monty)"

}

"profile_variables":

{ "type": "object",

"description": "an object (dictionary) of key value pairs, where the key is one of the variables presented in
Table 1"

},

"encrypted_variables":

{ "type": "array", "description": "an array of strings representing the keys of values sent in an encrypted
format in the request body"

}

}

}
```

Output Data:

In addition to the standard header parameter returned in the standard ES2+ APIs the below data will be
returned in the case of a successful call:

Output Data Name	Description	Type	MOC
profile	An object containing all the variables on the SM-DP+ in a key value pair format	Object	M

NOTE: the keys of the variables will follow the naming specified in Table 1.

NOTE: In case the OTA keys kic kid kik 03->0F were not shared in the variables parameter in the request
body their generation will depend on the profile configuration if it needs to have these keys or not, so
whether these variables are returned or not will depend on the need for their generation. This allows for
smaller response payloads (in case they're not needed we wouldn't share an extra 39 variables) and
prevents the need to generate variables that will not be used.

Request/Response JSON Examples:

Sample request:

```
{
"header": {
"functionRequesterIdentifier": "fri",
"functionCallIdentifier": "fci"
},
"input_batch_name": "input_batch_name",
"profile_variables": {
"iccid": "01234567899876543210",
"ki": "ki",
"opc": "opc",
"imsi": "imsi",
"msisdn": "msisdn"
},
"encrypted_variables": ["ki", "opc", "imsi"]
}
```

HTTP Response for a successful execution:

```
{
"header": {
"functionExecutionStatus": {
"status": "Executed-Success"
}
},
"profile": {
"iccid": "iccid",
"ki": "ki_encrypted",
```

```
"opc": "opc_encrypted",

"imsi": "imsi_encrypted",

"pin1": "pin1",

"pin2": "pin2",

"second_pin_appl1": "second_pin_appl1",

"second_pin_appl2": "second_pin_appl2",

"puk1": "puk1",

"puk2": "puk2",

"second_puk_appl1": "second_puk_appl1",

"second_puk_appl2": "second_puk_appl2",

"adm1": "adm1",

"acc": "acc",

"msisdn": "msisdn",

"kic1": "kic1",

"kid1": "kid1",

"kik1": "kik1",

"kic2": "kic2",

"kid2": "kid2",

"kik2": "kik2",

}

}
```

HTTP Response for a failed execution:

```
{

"header": {

"functionExecutionStatus": {

"status": "Failed",

"statusCodeData": {
```

```
"subjectCode": " 8.2.1",

"reasonCode": " 1.2",

"message": " Indicates that the function caller is not allowed to perform this function on the target
Profile."

}

}

}

}
```

NOTE1: The subjectCode, reasonCode, and message are all sent as specified in the SGP.22-v2.5, in addition to some custom messages to align with the usage of this API.

NOTE2: This format applies for all failed executions of the messages that will be mentioned below, the subjectCode, and reasonCode will change per the use case following the definition in the SGP.22-v2.5.

Specific status code:

Subject Code	Subject	Reason code	Reason	Description
8.2.1	Profile ICCID	1.2	Not Allowed (Authorization)	Indicates that the function caller is not allowed to perform this function on the target Profile.
8.2.1	Profile ICCID	3.3	Already in Use (Uniqueness)	Conflicting ICCID value.
8.2.11	Input Batch	2.2	Mandatory Element Missing	Indicates that the input_batch_name is missing in the context of this order.
8.2.11	Input Batch	3.9	Unknown	Indicates that the Input Batch identified by this Input Batch name is unknown to the SM-DP+.
8.2	Profile	2.2	Mandatory Element Missing	Indicates that the profile_variables parameter is missing in the context of this order.
8.2	Profile	3.9	Unknown	Indicates that the variable identified by {variable_name} is unknown to the SM-DP+.
8.2	Profile	2.1	Invalid	Indicates that the format of { variable_name } is invalid.

NOTE: the {variable_name} will be replaced by the name of the variable as shared in the "profile_variables" to which the error relates .

Error Messages

Error Message	Description
Indicates that the function caller is not allowed to perform this function on the target Profile.	Indicates that Monty didn't grant the operator the permission to call this function, or that one of the functionCallerIdentifier or the functionRequesterIdentifier shared in the header parameter are invalid
Indicates that the input_batch_name is missing in the context of this order.	Indicates that one of the elements marked as mandatory in the API specifications (input_batch_name, profile_variables) is missing in the request body.
Indicates that the profile_variables parameter is missing in the context of this order.	
Conflicting ICCID value	Indicates that a profile has already been generated for this ICCID
Indicates that the Input Batch identified by this Input Batch name is unknown to the SM-DP+.	Indicates that no input batch with the given name has been configured at SM-DP+ side.
Indicates that the format of {variable_name} is invalid.	Indicates that the provided variable using the key "variable_name" has an invalid format, meaning either a wrong length or a wrong type, e.g. if pin1 is configured to be of 4 digits this error will be returned in case a pin1 of length ≠ 4 is provided, same applies for all variables.

2.4 Delete Profile API

Below are the technical specifications of the reset profile API provided by Monty Mobile. This includes input parameters, output parameters, possible error messages and any restrictions.

Path:

https://<domain name>/gsma/rsp2/es2plus/deleteProfile

Input Parameters

In addition to the standard header parameter sent in the standard ES2+ APIs the following should be sent:

Input data name	Description	Type	MOC
iccid	Identification of the Profile to download and install in the eUICC.	String	M

```
{
"type": "object",

        "properties": {

        "iccid": {

        "type": "string",

        "pattern": "^[0-9]{19}[0-9F]?$",

        "description": "String representation of 20 digits, the 20th digit could be the padding
character F or a check digit"

        }

        }

}
```

Output Data

No additional output data besides functionExecutionStatus.

Request/Response JSON Examples

Sample request:

```
{
"header": {
```

```
"functionRequesterIdentifier": "{{fri}}",

"functionCallIdentifier": "{{fci}}"

},

"iccid": "01234567899876543210",

}
```

HTTP Response for successful execution:

```
{

"header": {

"functionExecutionStatus": {

"status": "Executed-Success"

}

}

}
```

HTTP Response for a failed execution:

```
{

"header": {

"functionExecutionStatus": {

"status": "Failed",

"statusCodeData": {

"subjectCode": " 8.2.1",

"reasonCode": " 1.2",

"message": " Indicates that the function caller is not allowed to perform this function on the target Profile."
```

```
    }

  }

  }

}
```

NOTE1: The subjectCode, reasonCode, and message are all sent as specified in the SGP.22-v2.5, in addition to some custom messages to align with the usage of this API.

NOTE2: This format applies for all failed executions of the messages that will be mentioned below, the subjectCode, and reasonCode will change per the use case following the definition in the SGP.22-v2.5.

Error Message:

Subject code	Reason code	Error Message	Explanation
8.2.1	1.2	Indicates that the function caller is not allowed to perform this function on the target Profile.	Indicates that Monty didn't grant the operator the permission to call this function, or that one of the functionCallerIdentifier or the functionRequesterIdentifier shared in the header parameter are invalid
1.2	2.2	Indicates that a mandatory input parameter of the function is missing.	Indicates that one of the elements marked as mandatory in the API specifications is missing in the request body.
8.2.1	3.9	Indicates that the Profile identified by this ICCID is unknown to the SM-DP+.	Indicates that the provided iccid value does not exist in the operator's inventory.