

ENCRYPTION AND DIGITAL SIGNATURE

Pre-requisites:

Channel has to generate a pair of keys (Private and Public) and Public Key need to be shared with EIS.

Algorithm Specification:

Advanced Encryption Standard (AES) for Payload Encryption.

- Cipher Mode Operation : Cipher Block Chaining (CBC) with PKCS5Padding
- Cryptographic Key : 256 bits
- IVector : First 16 byte of cryptographic key (Secret Key)

Rivest-Shamir-Adleman (RSA) for AES Key Encryption.

- Cipher Mode Operation : Electronic Codebook (ECB) with OAEPPadding
- Cryptographic Key : 2048 bit X509 Certificate

SHA256-Rivest-Shamir-Adleman (RSA) for Digital Signature.

- Hashing Algorithm : SHA 256
- Cryptographic Key : 2048 bit X509 Certificate

Implementation and Process Flow:

- 1) Channel has to generate a 32 character plain text dynamic key (AES 256 encryption key) for the payload encryption.
- 2) Encrypt the plain JSON request with above AES 256 encryption key.
- 3) SHA256-RSA algorithm has to be used to sign the plain request payload with the help of Channel Private Key.
- 4) RSA algorithm has to be used to encrypt the above AES 256 encryption key with the help of shared EIS Public Key.
- 5) All the three values (obtained from Steps 2, 3, 4) must be in **Base64 encoding** format and shared with EIS at the time of request in the below mentioned format.

```
{  
    "REQUEST_REFERENCE_NUMBER": "SBISI25111900000000000006",  
    "REQUEST": "[Outcome of Step (2)]",  
    "DIGI_SIGN": "[Outcome of Step (3)]"  
}
```

Add the secret key in the Http Header request

AccessToken - [Outcome of Step (4)]

6) EIS will validate the request received and decrypt the AccessToken with RSA algorithm and then proceed with decryption of REQUEST using the AES encryption key obtained from the decryption of AccessToken and verification of DIGI_SIGN with the shared Channel Public Key.

7) Failure in Validation of request received will return ERROR_CODE **SI011** with **401** HTTP code and the failure in decryption/verification will return ERROR_CODE **SI051** with **401** HTTP code.

```
{
  "REQUEST_REFERENCE_NUMBER": "SBISI2511190000000000011",
  "ERROR_CODE": "SI051",
  "ERROR_DESCRIPTION": "Unable to process due to technical error!!"
}
```

Response HTTP Header

X-Original-HTTP-Status-Code - 401

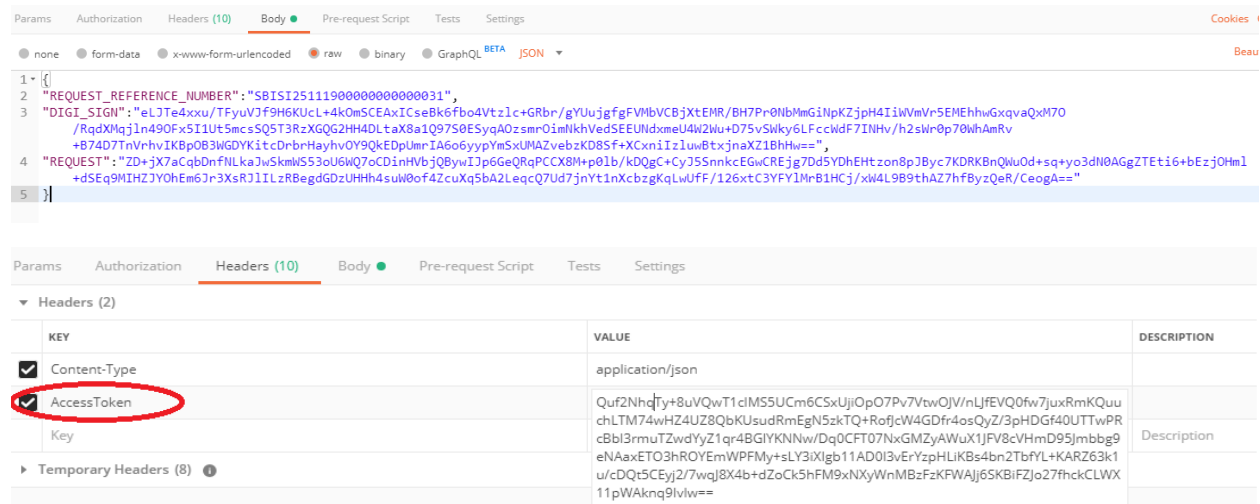
8) On Successful decryption/verification EIS will move on for valid request processing. Depending on success/failure EIS will forward the response to the Channel with **200** HTTP code.

```
{
  "RESPONSE": "[Encrypted Response (Encrypted with the key obtained from the decryption of AccessToken)]",
  "REQUEST_REFERENCE_NUMBER": "SBISI2511190000000000010",
  "RESPONSE_DATE": "26-11-2019 13:10:17",
  "DIGI_SIGN": "[Signed Data (Signed on Plain Response)]"
}
```

9) Finally channel must decrypt the RESPONSE using the same AES 256 encryption key which was used in for REQUEST and verify the DIGI_SIGN with SHA256 RSA algorithm with shared EIS Public Key.

REFERENCE SCREENSHOTS:

➤ Prepare JSON request using encrypted payload and encrypted key as shown below



NOTE: - Payload is passed in body as *REQUEST* field, Signature is passed in body as *DIGI_SIGN* field and encrypted key in header as *AccessToken*.

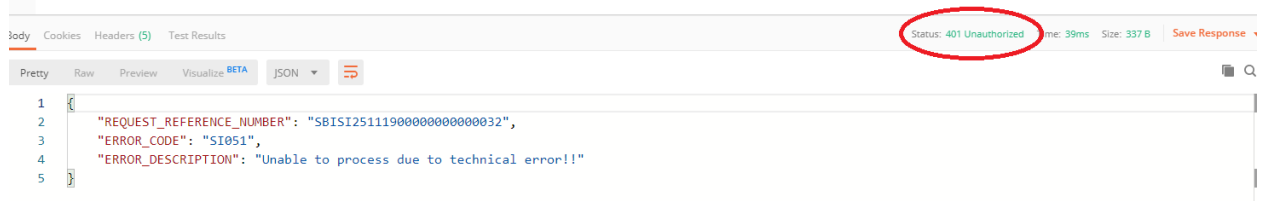
➤ Encrypted response format after successful processing of the request.



The screenshot shows the 'Test Results' tab in a browser's developer tools. The status bar at the top right indicates 'Status: 200 OK', 'Time: 122ms', and 'Size: 960 B'. The response body is displayed in a 'Raw' view, showing a JSON object with the following fields:

```
1 {
2   "RESPONSE": "yb9zH+IzPrRrog2zfVstUv89RMIj0vuiKy5Y80UNMoLyEz7otA0KHA00eTume2i6xXa4B1AdSj03jcBgJmcZ2GFvQB/104TPhyDyO36pUm6eZLZB/
3     EKHV0NZd7k6hd4aIx6FhB6BxgIEP+4TKrSu+9BEWFzBwk1EovX7G1hZPU1mFKFITXUg0452K9nV7bKtCcj1zoxMLyFvO51Y+d7E6VsNqp0/
4     sLEq4wOuVsBpJGG87ITn9gYt0XtFvAXVzCgdcrcbGzr6r32FL1s4yDZvmZvczSaFxp4iM93PaXiC1M=",
5   "REQUEST_REFERENCE_NUMBER": "SBISI25111900000000000031",
6   "RESPONSE_DATE": "26-11-2019 16:18:04",
7   "DIGI_SIGN": "GR0mDMHv/UjLH9FoeXACQqMx+APuHLz+hI55F5AgRMSvgc6/
8     Ag7b8LWQaCfossNWQ1Tbh1T0FaV6iHmVOCxebEG0jvm7KdIIDcCtmYBUZgxa8kYJ6qZQYkYy51xpZ0mI8e14SwHrLfrQ8i9BGZ2r6P9BISgtzUUh+1PKG+svwFHTt5070mwzTE//
9     WAEv9Nq4EH718wqF+ZjXQs8wqj59G/Dd0YjAdhj95i25GQXg3vg8rpe7BJI8AI3288pp6H1t+YqfneaBxHf05x+MbgvgyAZwt3ocgYiFCHhvWHQo8jdGtJ6MA8hK13HwqZNGUKhb1e
10    +DJxNoqI4kbYDw="
11 }
```

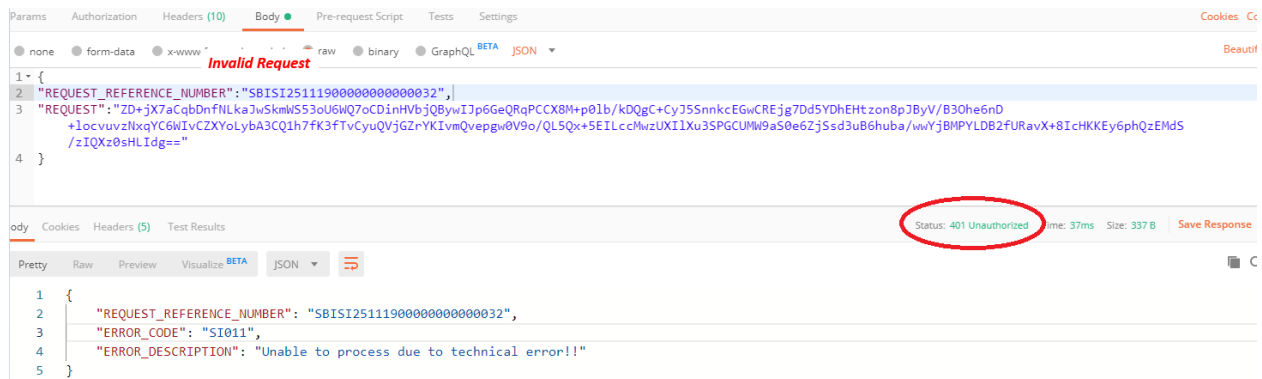
➤ Authorization/Authentication Failure response format before processing the request.



The screenshot shows the 'Test Results' tab in a browser's developer tools. The status bar at the top right indicates 'Status: 401 Unauthorized', 'Time: 39ms', and 'Size: 337 B'. The response body is displayed in a 'Pretty' view, showing a JSON object with the following fields:

```
1 {
2   "REQUEST_REFERENCE_NUMBER": "SBISI25111900000000000032",
3   "ERROR_CODE": "SI051",
4   "ERROR_DESCRIPTION": "Unable to process due to technical error!!"
5 }
```

➤ Validation Failure response format before processing the request.



The screenshot shows the 'Params' tab in a browser's developer tools. The status bar at the top right indicates 'Status: 401 Unauthorized', 'Time: 37ms', and 'Size: 337 B'. The response body is displayed in a 'Raw' view, showing a JSON object with the following fields:

```
1 {
2   "REQUEST_REFERENCE_NUMBER": "SBISI25111900000000000032",
3   "REQUEST": "ZD+jX7aCqbDnfnLkaJwSkmWS53oU6WQ7oCDInHVbjQBywIjP6GeQRqPCCX8M+p0lb/kDQgC+Cy755nkkEGwCREjg7Dd5YDhEhtzon8pJByV/B30he6nD
4     +1ocvuvzNxxqYC6WIVCZXyOlybA3CQ1h7fK3fTvcyUQVjGZrYKIvmQvepgw0V9o/QL5Qx+5E1LccMwzUXI1Xu3SPGCUMM9a50e6Zj5sd3uB6huba/wwYjBMPYLD2FURavX+8IcHKKEy6phQzEMdS
5     /zIQxz0sHLIdg="
6 }
```