X-Header Insertion and Encryption

Feature Summary and Revision History

Summary Data

Table 1: Summary Data

Applicable Product(s) or Functional Area	5G-UPF
Applicable Platform(s)	VPC-SI
Feature Default Setting	Disabled – Configuration Required
Related Changes in This Release	Not Applicable
Related Documentation	*UCC 5G UPF Configuration and Administration Guide*

Revision History

Revision Details	Release
Added support for the anti-spoofing functionality through X-header enrichment.	2023.02.0
First introduced.	2020.02.x

Feature Description

The process of X-header insertion and encryption is collectively known as Header Enrichment (HE). The UPF supports HE to allow deletion and modification of the existing X-header fields in the HTTP header of the URL requests.

This feature enables appending the headers to HTTP or WSP GET and POST Request packets, and HTTP Response packets for use by end applications, such as mobile advertisement insertion (MSISDN, IMSI, IP address, user-customizable, and so on).

How it Works

HE includes two operations:

1. X-Header Insertion
2. X-Header Encryption

X-Header Insertion

Extension header (X-header) fields are fields that are added to protocol headers for specific purposes such as user identification, device detection, etc. The X-header mechanism allows additional entity-header fields to be defined without changing the protocol. However, these fields are assumed to be unrecognizable by the recipients.

X-header insertion inserts X-headers in HTTP or WSP GET and POST Request packets and HTTP Response packets. You can configure rules to insert X-headers in HTTP or WSP Request and HTTP Response packets. The charging-action associated with the rules contain the list of X-headers to be inserted in the packets.

Note

- Flows for which the X-header is inserted in a packet are not offloaded.
- With the X-header configuration, all TCP OOO packets irrespective of transmit order CLI, will be buffered and sent after re-ordering.

X-Header Encryption

X-header encryption enables encryption of X-header fields before insertion. If X-header insertion has already happened for an IP flow (because of any X-header format) and the current charging-action has the first-request-only flag set, the X-header insertion will not happen for that format. If the first-request-only flag is not set in a charging-action, the insertion for that X-header format continues happening in other suitable packets of that IP flow.

Changes to the X-header format configuration do not trigger re-encryption for existing calls. The changed configuration will however, be applicable for new calls. The changed configuration will also apply at the time of the next re-encryption, to the existing calls for which the re-encryption timeout is specified. If encryption

is enabled for a parameter while data is flowing, since its encrypted value will not be available, insertion of that parameter stops.

Note This feature does not support flow recovery.

Limitations

This section highlights the limitations existing in the current release:

- X-header insertion in the Response packet is not supported.
- X-header encryption with RSA and RC4MD5 is supported, but not supported with AES.
- Monitor protocol for X-header is not supported.
- Insertion of the following X-header fields is not supported in a packet:
  - QoS
  - UIDH
  - Customer ID
  - Hash Value
  - Time of the Day
  - RADIUS String
  - Session-Id
  - Congestion Level
  - User-Profile
- Parsing and buffering of the HTTP header will be done only if the rulebase has any x-header format configured with x-header fields explicitly set to delete-existing. For operators not using x-header delete-existing, no overhead for parsing or buffering will be seen.

Configuring X-Header Insertion and Encryption

Configuring X-Header Insertion

Step 1	Create or configure a ruledef to identify the HTTP packets in which the X-headers must be inserted.
Step 2	Create or configure a rulebase and configure the charging-action that inserts the X-header fields into the HTTP packets.
Step 3	Create the X-header format as described in *Creating the X-Header Format*.

Step 4	Configure the X-header format based on the message type in the charging action, as described in *Configuring the X-Header Format*.

Creating the X-Header Format

To create an X-header format, use the following configuration:

```
configure
   active-charging service ecs_service_name
      xheader-format xheader_format_name
      end
```

Configuring the X-Header Format

To configure an X-header format, use the following configuration:

```
configure
   active-charging service ecs_service_name
      xheader-format xheader_format_name
        insert xheader_field_name string-constant xheader_field_value | variable
{ bearer { 3gpp { apn | charging-characteristics | charging-id | imei |
imsi | qos | rat-type | s-mcc-mnc | sgsn-address } | acr | customer-id |
 ggsn-address | mdn | msisdn-no-cc | radius-string |
radius-calling-station-id | session-id | sn-rulebase |
subscriber-ip-address | username } [ encrypt ] [ delete-existing ] | http
 { host | url } }
        end
```

Configuring X-Header Encryption

Step 1	Configure the X-header insertion as described in *Configuring X-Header Insertion*.
Step 2	Create or configure a rulebase, and the encryption certificate to use and the re-encryption parameter as described in *Configuring X-Header Encryption*.
Step 3	Configure the encryption certificate to use as described in *Configuring Encryption Certificate*.

Configuring X-Header Encryption

To configure X-header encryption, use the following configuration:

```
configure
   active-charging service ecs_service_name
      rulebase rulebase_name
         xheader-encryption certificate-name certificate_name
         xheader-encryption re-encryption period re-encryption_period
         end
```

NOTES:

- This configuration enables X-header encryption for all subscribers using the specified rulebase.

- If the certificate is removed, ECS continues using the copy that it has. The copy is set free once the certificate name is removed from the rulebase.
- Changes to x-header format configuration won't trigger re-encryption for existing calls. The changed configuration will however, be applicable for new calls. The changed configuration will also apply at the next reencryption time to those existing calls for which reencryption timeout is specified. If encryption is enabled for a parameter while data is flowing, since its encrypted value won't be available, insertion of that parameter stops.

Configuring Encryption Certificate

To configure the encryption certificate, use the following configuration:

```
configure
  certificate name certificate_name pem { { data pem_certificate_data private-key
 pem [ encrypted ] data pem_pvt_key } | { url url private-key pem { [
encrypted ] data pem_pvt_key | url url } }
  end
```

Verifying X-header Insertion and Encryption Configuration

Use the following command in the Exec Mode to verify your configuration:

xheader-format *xheader_format_name*

Monitoring and Troubleshooting X-Header Insertion and Encryption Configuration

show active-charging charging-action statistics name

The output of this command displays statistics for X-header information.

- X-header information:
  - XHeader Bytes Injected
  - XHeader Pkts Injected
  - IP Frags consumed by XHeader
  - XHeader Bytes Removed
  - XHeader Pkts Removed

show active-charging rulebase statistics name

The output of this command displays the header enrichment statistics.

- HTTP header buffering limit reached

Anti-Spoofing Support

Feature Description

The UPF supports spoofing detection and mitigation using header enrichment. The anti-spoofing functionality allows the operators and their subscribers from spoofing threats posed by malicious UE devices. This feature detects the fraudulent activities when an external portal is used for subscriber or content authorization.

The anti-spoofing functionality is supported only for HTTP Requests. This functionality allows deletion and modification of the existing X-header fields in the HTTP header of the URL request.

If the user-generated header name is similar to the header name used in x-header format, you can search, delete, and modify the X-header fields. The X-header field names are case-insensitive. For example, if x-MSISDN and x-msisdn fields are present in an incoming HTTP request, both fields are processed.

If the user HTTP header already has the fields to be inserted, the corresponding values for those fields are replaced with the modified values from the gateway at the end of the header. In case of multiple entries for one field, which is already present in the header, a single instance of the field is selected along with the value inserted from the gateway.

In case of multiple HTTP GET requests, the packet header replaced with X-header fields for all GET requests. If encryption is enabled, it is performed after inserting the X-header.

Limitations

The following are the known limitations of the anti-spoofing feature:

- If an HTTP GET or POST header is not complete in 3 packets, the anti-spoofing is performed only for the last packet in which the header completes. Otherwise, the packets are sent out without HE.
- Though fields without having 'x-' in the field name are allowed, we do not delete the extension header fields that do not start with x-.

Enabling Anti-Spoofing in X-Header

To enable anti-spoofing in the X-header, use the following configuration:

```
configure
  active-charging service acs_service_name
    xheader-format xheader_format_name
      insert xheader_field_name variable { bearer subscriber-ip-address } [
 encrypt ] [ delete-existing ]
      exit
```

NOTES:

- **delete-existing**: This command enables detection of spoofing in an X-header file.
- This feature is disabled by default.
- A maximum number of 10 X-header fields can be configured with the **delete-existing** CLI keyword, which also includes fields configured only for X-header insertion.

OAM Support

This section provides information regarding show commands and their outputs for the anti-spoofing feature.

show user-plane-service statistics charging-action [name | all]

The output of this command displays the user plane statistics for charging action.

- X-header information:
  - XHeader Bytes Injected
  - XHeader Pkts Injected
  - XHeader Bytes Removed
  - XHeader Pkts Removed

show user-plane-service statistics rulebase [name | all]

The output of this command displays user plane statistics for rulebase.

- Header Enrichment stats:
  - HTTP header buffering limit reached