

Configure StarOS MTU Interface, APN, and Local Subscribers

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Background Information](#)

[Configure](#)

[Interface MTU Size Configuration](#)

[APN Data Tunnel MTU Size Configuration](#)

[Subscriber Profile \(CDMA subs\)](#)

[Verify](#)

[Troubleshoot](#)

Introduction

This document describes how to configure StarOS MTU Interface, APN, and Local Subscribers.

StarOS has multiple CLIs to configure Maximum Transmission Unit (MTU) size for interfaces, APNs and local subscribers profile.

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

This document is not restricted to specific software and hardware versions.

Background Information

Maximum transmission unit (MTU) defines the largest size of packets that an interface can transmit without the need to fragment.

IP packets larger than the MTU must go through IP fragmentation procedures.

The MTU value is the frame size without Ethernet headers, VLAN tagging, or other overhead.

Configure

• Interface MTU Size Configuration

```
configure
context context_name interface interface_name ip mtu bytes end
```

Reference: [Command Line Interface Reference, Ethernet Interface Configuration Mode Commands, StarOS 21.23](#)

This config is enforced by the NPU/iftask/VPP of ASR5500/VPC.

This config works at a lower layer as compared to APN MTU config, meaning it works on ip/ipv6 pkts and is not tunnel aware.

To be more specific, it works only on outgoing IP pkts (data + ctrl) and fragments the packets to smaller sizes based on configured value.

IP MTU is supported for a normal interface and point-to-point interface (OLC ports).

The maximum MTU size allowed with an OLC port is 1600.

The maximum MTU size allowed with an Ethernet port is 2048. The default MTU size is 1500.

The maximum sizes for ethernet MTUs are:

- **Untagged traffic** (non-VLAN) : **ip MTU** *mtu-size* + ethernet header
- **VLAN traffic** : **ip MTU** *mtu-size* + ethernet header + vlan header

• APN Data Tunnel MTU Size Configuration

```
configure
context context_name
apn apn_name
pco-options link-mtu bytes
ppp mtu bytes
data-tunnel mtu bytes
policy ipv6 tunnel mtu exceed { fragment inner | notify-sender | fragment }
access-link ip-fragmentation { df-ignore | normal | df-fragment-and-icmp-notify }
end
```

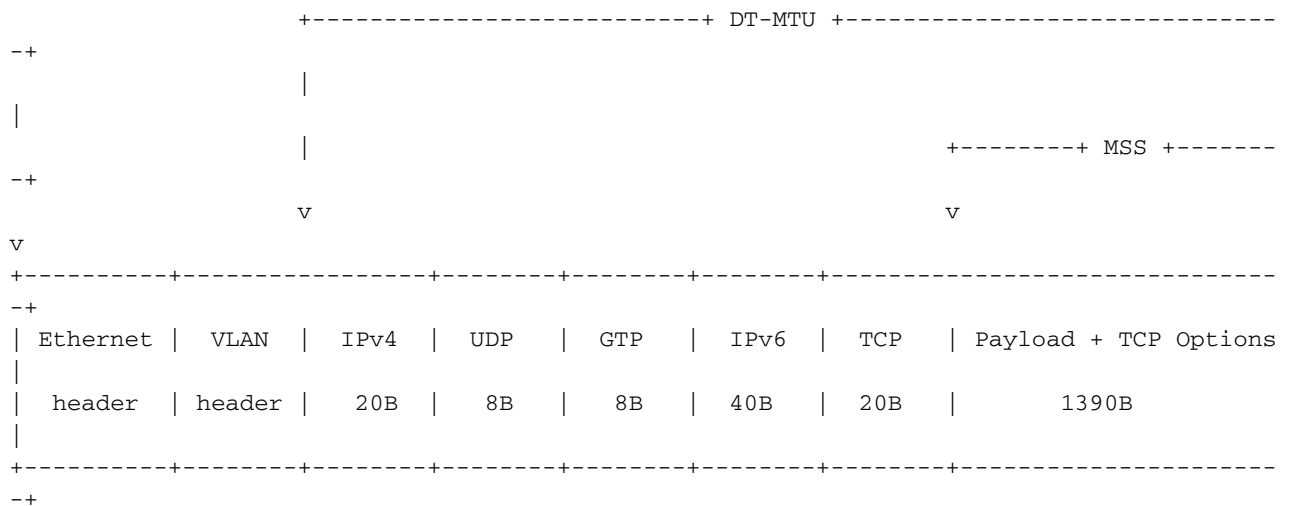
- **ppp mtu bytes** Reference: [Command Line Interface Reference, APN Configuration Mode Commands, StarOS 21.23](#)

This configuration controls the Downlink data tunnel MTU payload for IPv4 on StarOS Gateway. Packets exceeding this size are fragmented and then tunnel encapsulated. The value is an integer from 100 through 2000. Default: 1500 **Important:** The MTU refers to the PPP payload which excludes the two PPP octets. Therefore, an MTU of 1500 corresponds to the 3GPP standard MTU of 1502 for GTP packets with PPP payloads.

- **data-tunnel mtu bytes** Reference: [Command Line Interface Reference, APN Configuration Mode Commands, StarOS 21.23](#) As per RFC-4861 the P-GW supports the sending of the IPv6 MTU option in RAs for IPv6 and IPv4v6 PDN types towards the UE. The (Internet) can now send a downlink data packet and based on the configured

MTU, data fragmentation is performed at the source if required. This feature also reduces the number of ICMPv6 Packet Too Big Error messages in the customer's network. This configuration controls the Downlink data tunnel MTU payload for IPv6. Packet exceeding this size are dropped/fragmented based on "policy" configuration. The value is an integer between 1280 and 2000. Default: 1500

Caution: How this feature interacts with **TCP MSS Configuration**. RFC [6691](#) excerpt: "" 2. The Short Statement When calculating the value to put in the TCP MSS option, the MTU value SHOULD be decreased by only the size of the fixed IP and TCP headers and SHOULD NOT be decreased to account for any possible IP or TCP options; conversely, the sender MUST reduce the TCP data length to account for any IP or TCP options that it is including in the packets that it sends. The rest of this document just expounds on that statement, and the goal is to avoid IP-level fragmentation of TCP packets. "" This means that the actual payload is reduced by TCP Options bytes for a given TCP MSS. If we take as an example an MSS of 1390 and TCP options of 12 bytes, then the payload is 1378 bytes. PAYLOAD 1378 + TCP 32 + = 1410 (1378B [PAYLOAD] + 12B [TCP options] = 1390B MSS) IPv6 40 + = 1450 GTPU 8 + = 1462 UDP 8 + = 1470 IP 20 + = 1490 <-- `data-tunnel mtu` is compared here VLAN 4 + = 1494 ETH 14 = 1504



If configured data tunnel MTU is less than 1490 in the case above, the mtu exceed policy will be applied to the packet.

- **policy ipv6 tunnel mtu exceed { fragment inner | notify-sender | fragment }** Reference: [Command Line Interface Reference, APN Configuration Mode Commands, StarOS 21.23](#) If we consider the following packet seen on SGi interface:

```

| Frame 81: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits)
1514 | 14 | Ethernet II, Src: fa:16:3e:5e:0a:23 (fa:16:3e:5e:0a:23), Dst:
fa:16:3e:5d:f8:03 (fa:16:3e:5d:f8:03)
1500 | 4 | 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 70
1496 | 40 | Internet Protocol Version 6, Src: 2001:192:168:80:f816:3eff:fe60:93a, Dst:
2001:20::4c99:6101
1456 | 1456 | Internet Control Message Protocol v6

```

Where the first column represents the accumulated bytes and the second column - header length. - **fragment inner** The system will perform an inner IPv6 fragmentation at GTP tunnel initiator, which is us, if the subscriber packet exceeds MTU after encapsulation. The packet on S1-U will be fragmented as follows:

```

# Frag #1
| Frame 51: 1510 bytes on wire (12080 bits), 1510 bytes captured (12080 bits)
1510 | 14 | Ethernet II, Src: fa:16:3e:b5:f6:b2 (fa:16:3e:b5:f6:b2), Dst:

```

```

fa:16:3e:96:00:d9 (fa:16:3e:96:00:d9)
1496 | 4 | 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 40
1492 | 20 | Internet Protocol Version 4, Src: 10.1.40.1, Dst: 192.168.1.100
1472 | 8 | User Datagram Protocol, Src Port: 1, Dst Port: 2152
1464 | 8 | GPRS Tunneling Protocol
1456 | 48 | Internet Protocol Version 6, Src: 2001:192:168:80:f816:3eff:fe60:93a, Dst:
2001:20::4c99:6101
1408 | 1408 | Internet Control Message Protocol v6

```

Frag #2

```

| Frame 52: 150 bytes on wire (1200 bits), 150 bytes captured (1200 bits)
150 | 14 | Ethernet II, Src: fa:16:3e:b5:f6:b2 (fa:16:3e:b5:f6:b2), Dst:
fa:16:3e:96:00:d9 (fa:16:3e:96:00:d9)
136 | 4 | 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 40
132 | 20 | Internet Protocol Version 4, Src: 10.1.40.1, Dst: 192.168.1.100
112 | 8 | User Datagram Protocol, Src Port: 1408, Dst Port: 2152
104 | 8 | GPRS Tunneling Protocol
96 | 48 | Internet Protocol Version 6, Src: 2001:192:168:80:f816:3eff:fe60:93a, Dst:
2001:20::4c99:6101
48 | 48 | Data (48 bytes)

```

- **notify-sender** System is a PMTU discovery and send “ICMPv6 Packet Too Big” to the original sender if the subscriber packet exceeds MTU after encapsulation. - **fragment** The system will do an outer IPv6 fragmentation if the subscriber packet exceeds MTU after encapsulation. The packet on S1-U will be fragmented as follows:

Frag #1

```

| Frame 108: 1510 bytes on wire (12080 bits), 1510 bytes captured (12080 bits)
1510 | 14 | Ethernet II, Src: fa:16:3e:b5:f6:b2 (fa:16:3e:b5:f6:b2), Dst:
fa:16:3e:96:00:d9 (fa:16:3e:96:00:d9)
1496 | 4 | 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 40
1492 | 20 | Internet Protocol Version 4, Src: 10.1.40.1, Dst: 192.168.1.100
1472 | 8 | User Datagram Protocol, Src Port: 2152, Dst Port: 2152
1464 | 8 | GPRS Tunneling Protocol
1456 | 40 | Internet Protocol Version 6, Src: 2001:192:168:80:f816:3eff:fe60:93a, Dst:
2001:20::4c99:6201
1416 | 1416 | Internet Control Message Protocol v6

```

Frag #2

```

| Frame 109: 78 bytes on wire (624 bits), 78 bytes captured (624 bits)
78 | 14 | Ethernet II, Src: fa:16:3e:b5:f6:b2 (fa:16:3e:b5:f6:b2), Dst:
fa:16:3e:96:00:d9 (fa:16:3e:96:00:d9)
64 | 4 | 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 40
60 | 20 | Internet Protocol Version 4, Src: 10.1.40.1, Dst: 192.168.1.100
40 | 40 | Data (40 bytes)

```

- **access-link ip-fragmentation { df-ignore | normal | df-fragment-and-icmp-notify }** Reference: [Command Line Interface Reference, APN Configuration Mode Commands, StarOS 21.23](#)- **df-ignore** The default configuration, StarOS will always fragment packet irrespective of df bit.- **normal** Drops the packet and sends an ICMP unreachable message to the source of the packet.- **df-fragment-and-icmp-notify** Partially ignores the DF bit; fragments and forwards the packet, but also returns an ICMP error message to the source of the packet.
- **pco-options link-mtu <>** Reference: [Command Line Interface Reference, APN Configuration Mode Commands, StarOS 21.23](#) Configures APN to include link MTU in PCO IE, if it is requested by UE. When UE sends IPv4 Link MTU Size PCO request during Initial attach/ Standalone PDN connection, then the S-GW/SGSN/HSGW sends the same transparently in Create Session Request, Create/Update PDP Context Request, or PBU to P-GW, GGSN, or PMIP-PGW. Create Session Response, Create/ Update PDP Context Response/ PBA will be

sent with the latest configured MTU size PCO value in APN.If UE is in outbound roaming, then the default value (1500) will be provided in the MTU size PCO.

• Subscriber Profile (CDMA subs)

```
configure
 context context_name subscriber default ipv6 minimum-link-mtu bytes ppp mtu bytes
 mobile-ipv6 tunnel mtu bytes pco-options link-mtu bytes end
```

Reference: [Command Line Interface Reference, Subscriber Configuration Mode Commands, StarOS 21.23](#)

Verify

Use this section in order to confirm that your configuration works properly.

StarOS commands below to verify the configuration:

```
show configuration
show configuration verbose
show configuration apn apn_name
```

Troubleshoot

This section provides the information you can use in order to troubleshoot your configuration.

- monitor subscriber with minimum verbosity 3 - to see PCO options.
- External pcap - to see packet fragmentation