



Indirect Forwarding Tunnel

- [Revision History](#), on page 1
- [Feature Description](#), on page 1
- [How It Works](#), on page 2
- [Configuring Indirect Forwarding Tunnel](#), on page 5
- [Monitoring and Troubleshooting](#), on page 6

Revision History



Note Revision history details are not provided for features introduced before release 21.24.

| Revision Details | Release |
|------------------|-----------|
| First introduced | Pre 21.24 |

Feature Description

SAEGW supports Indirect Forwarding Tunnel (IDFT) procedures for creation and deletion, which are applicable for Pure-S and Collapsed calls with multi-PDN and multi-bearers. This feature is applicable for IDFT support with and without S-GW relocation and collision scenarios.

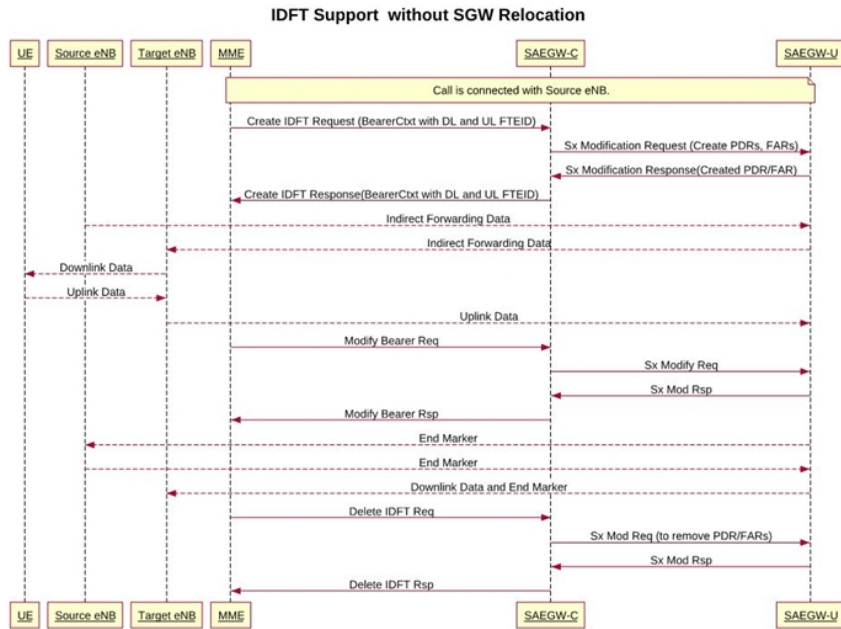


Note The IDFT in CUPS is a CLI-controlled feature. By default, the IDFT feature in CUPS is disabled.

How It Works

Call Flow

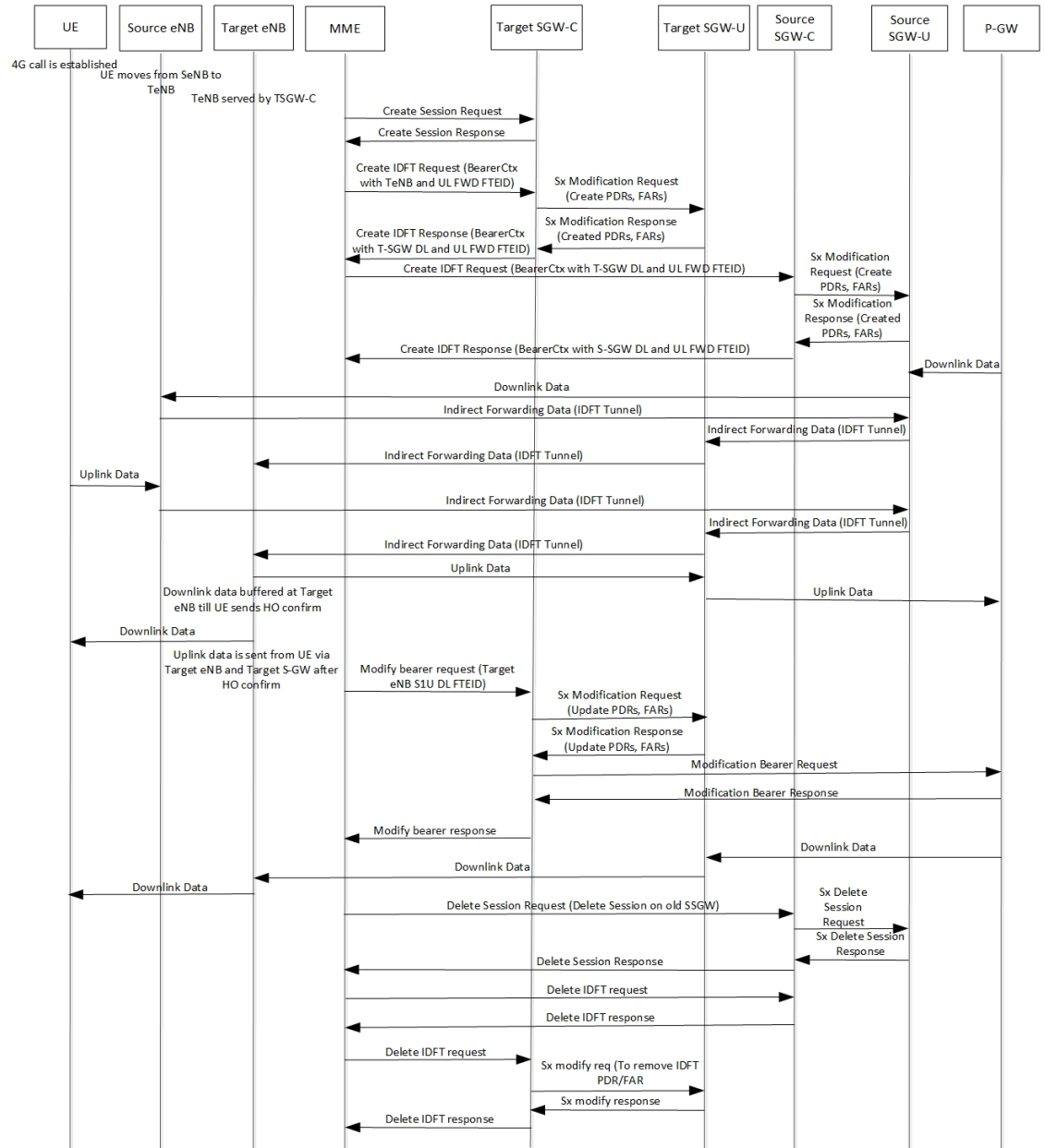
The following call flow illustrates, at a high-level, the IDFT support without S-GW Relocation.



437403

The following call flow illustrates the IDFT support with S-GW Relocation.

Figure 1: IDFT Support with S-GW Relocation



455686

The above call flow describes the IDFT tunnels establishment and deletion with S-GW relocation and without MME change.

If IDFT tunnels are not deleted by MME, then S-GW initiates the local delete of IDFT tunnels.

This feature supports the following scenarios for the Pure-S and Collapse calls:

- S-GW relocation with same MME
- S-GW relocation with same MME and different eNodeB
- S-GW relocation with different MME

- S1-based eNodeB Handoff
- EUTRAN to UTRAN Handoff



Note S4 interface is not supported in CUPS. Hence any EUTRAN to UTRAN handoffs (and vice-versa) involving S4 interface is also not supported.

- EUTRAN to UTRAN Handoff with S-GW relocation
- UTRAN to EUTRAN Handoff
- UTRAN to EUTRAN Handoff with S-GW relocation
- Sx transaction timeout during IDFT setup or removal
- Pending Sx transaction (event from PCRF or OCS) and IDFT request comes in
- Create Bearer Request (CBR) during Active IDFT
- Update Bearer Request (UBR) during Active IDFT
- Delete Bearer Request (DBR) during Active IDFT
- Modify Bearer Request (MBR) behavior on other PDN during Active IDFT
- Source MME path failure
- Target MME path failure
- MME path failure with NTSR enabled
- eGTP-C S5 path failure
- eGTP-C S5 path failure with P-GW restart notification enabled
- Sx path failure (clean IDFT and calls)
- Abort session (clear sub all, local abort, and so on.)
- CBR, UBR on other PDN during Active IDFT
- DBR on other PDN/bearer during Active IDFT
- S1-u path failure for target eNodeB
- S-GW path failure for target S-GW
- S1-u error indication on the default bearer while Active IDFT
- S1-u error indication on the dedicated bearer while Active IDFT
- S1-u error indication from the target S-GW to source S-GW bearer
- S1-u error indication from the target eNodeB to target S-GW bearer

Supported Functionality

The IDFT feature supports the following functionality:

- Create IDFT request for Collapsed, Pure-S, combination of Collapsed and Pure-S multi-PDN calls with multiple bearers.
- Data transfer on downlink and uplink IDFT bearers.
- Deletion of IDFT request from MME. Also, timer-based deletion of IDFT bearer after expiration of a default value of 100 seconds, if the MME does not send an IDFT request for deletion.
- Deletion of IDFT PDN, including Clear/Delete subscribers from MME/P-GW, when normal PDN goes down.
- Sx-Path failure handling in case of Pure-S and collapsed calls during IDFT Active/ IDFT Create Sx-Pending state.
- Message interaction and collision during IDFT PDN establishment or deletion with any other procedure.
- S11/S5 and Sx Path Failure Handling on non-IDFT PDN is now supported when IDFT PDN is Active.



Important

Transport GTP-U address capability is assumed to be same across eNodeB and S-GW.

Configuring Indirect Forwarding Tunnel

This section describes the CLI commands available in support of IDFT feature.

Enabling Indirect Forwarding Tunnel Feature

On Control Plane, use the following CLI commands to enable or disable the IDFT feature.

```
configure
  context context_name
    sgw-service service_name
      [ default | no ] egtp idft-support
    end
```

NOTES:

- **idft-support**: Enables/Disables the IDFT feature in CUPS.
- By default, the IDFT feature is disabled and this CLI command is applicable on run-time change.

Verifying the Indirect Forwarding Tunnel Feature

show sgw-service name <service_name>

On Control Plane, the output of this CLI command has been enhanced to display if the IDFT feature is enabled or disabled.

- IDFT-Feature Support for CUPS : Enabled/Disabled

Monitoring and Troubleshooting

This section provides information regarding the CLI commands available in support of monitoring and troubleshooting the feature.

Show Commands Input and/or Outputs

This section provides information regarding show commands and their outputs in support of the feature.

show subscribers saegw-only full all

On Control Plane, use this command to see the IDFT Local and Remote TEID data. The following is a sample output:

```

Indirect Fwding   : Active
DL fwd local  addr: 209.165.200.228           DL fwd remote  addr: 209.165.200.226

DL fwd local  teid: [0x80028004]             DL fwd remote  teid: [0x2002d2e5]
UL fwd local  addr: 209.165.200.228           UL fwd remote  addr: 209.165.200.226

UL fwd local  teid: [0x8002a004]             UL fwd remote  teid: [0x20042bca]

```

show subscribers user-plane-only callid <call-id> pdr all

On User Plane, use this command to see the PDR or FAR created for IDFT. The following is a sample output:



Important IDFT PDRs will have ACCESS as the source and destination interface type.

```

+-----Source Interface:      (C) - Core          (A) - Access
|-----Type                  (P) - CP-function   (.) - Unknown
|
|+-----Destination Interface: (C) - Core          (A) - Access
||-----Type                 (P) - CP-function   (.) - Unknown
||
||
||+----Rule-Type:             (S) - Static        (P) - Predefined
|||----Type                   (D) - Dynamic        (.) - Unknown
|||
|||
vvv  PDR-ID      Associated FAR-ID   Associated URR-ID(s)   Associated QER-ID(s)
---  -
ACS  0x0001     0x8001              n/a                    0x80000001
CAS  0x0002     0x8002              n/a                    0x80000001

```

```

ACD 0x0003      0x0003      0x00000007      0x00000002
                               n/a           0x80000003
CAD 0x0004      0x0004      0x00000007      0x00000002
                               n/a           0x80000003
CAD 0x0005      0x0005      0x00000000      n/a
ACD 0x0006      0x0006      0x00000000      n/a
CAD 0x0007      0x0007      0x00000000      n/a
ACD 0x0008      0x0008      0x00000000      n/a
AAD 0x0009      0x0009      0x00000000      n/a
AAD 0x000A      0x000A      0x00000000      n/a
AAD 0x000B      0x000B      0x00000000      n/a
AAD 0x000C      0x000C      0x00000000      n/a
    
```

Total subscribers matching specified criteria: 1

show subscribers user-plane-only full all



Important Data statistics on IDFT PDRs are captured in the same way as existing PDR statistics. However, it is captured with a limitation – Statistics for DL and UL IDFT will be incremented in Pkts-Down and Bytes-Down category.

The following is sample output:

```

Static & Predef Rule Match stats:
Rule Name      Pkts-Down  Bytes-Down  Pkts-Up    Bytes-Up    Hits      Match-Bypassed
FP-Down (Pkts/Bytes)  FP-Up (Pkts/Bytes)
-----
catchall      0           0           0           3           1368     3           0
              0/0         0/0
Dynamic Rule Match stats:
PDR Id      Pkts-Down  Bytes-Down  Pkts-Up    Bytes-Up    Hits      Match-Bypassed
FP-Down (Pkts/Bytes)  FP-Up (Pkts/Bytes)
-----
0x0004      2           856         0           0           2         0           0/0
              0/0
0x000b      2           856         0           0           2         0           0/0
              0/0
0x000c      2           168         0           0           2         0           0/0
              0/0
    
```

show subscribers user-plane-only full all