Serving Gateway Configuration

This chapter provides configuration information for the Serving Gateway (S-GW).

Important Information about all commands in this chapter can be found in the *Command Line Interface Reference*.

Because each wireless network is unique, the system is designed with a variety of parameters allowing it to perform in various wireless network environments. In this chapter, only the minimum set of parameters are provided to make the system operational. Optional configuration commands specific to the S-GW product are located in the *Command Line Interface Reference*.

This chapter includes the following topics:

Configuring the System as a Standalone eGTP S-GW

This section provides a high-level series of steps and the associated configuration file examples for configuring the system to perform as a eGTP S-GW in a test environment.

Information Required

The following sections describe the minimum amount of information required to configure and make the S-GW operational on the network. To make the process more efficient, you should have this information available prior to configuring the system.

There are additional configuration parameters that are not described in this section. These parameters deal mostly with fine-tuning the operation of the S-GW in the network. Information on these parameters can be found in the appropriate sections of the *Command Line Interface Reference*.

Required Local Context Configuration Information

The following table lists the information that is required to configure the local context on an eGTP S-GW.

Table 1: Required Information for Local Context Configuration

Required Information	Description
Management Interface Configuration	
Interface name	An identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured.
IP address and subnet	IPv4 addresses assigned to the interface. Multiple addresses and subnets are needed if multiple interfaces will be configured.
Physical port number	The physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides followed by the number of the physical connector on the card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Gateway IP address	Used when configuring static IP routes from the management interface(s) to a specific network.
Security administrator name	The name or names of the security administrator with full rights to the system.
Security administrator password	Open or encrypted passwords can be used.
Remote access type(s)	The type of remote access that will be used to access the system such as telnetd, sshd, and/or ftpd.

Required S-GW Ingress Context Configuration Information

The following table lists the information that is required to configure the S-GW ingress context on an eGTP S-GW.

Table 2: Required Information for S-GW Ingress Context Configuration

Required Information	Description
S-GW ingress context name	An identification string from 1 to 79 characters (alpha and/or numeric) by which the S-GW ingress context is recognized by the system.
Accounting policy name	An identification string from 1 to 63 characters (alpha and/or numeric) by which the accounting policy is recognized by the system. The accounting policy is used to set parameters for the Rf (off-line charging) interface.
S1-U/S11 Interface Configuration (To/from eNodeB/MME) **Note** The configuration provided in this guide assumes a shared S1-U/S11 interface. These interfaces can be separated to support a different network architecture. The information below applies to both.	

Required Information	Description
Interface name	An identification string between 1 and 79 characters (alpha and/or numeric) by which the interface is recognized by the system. Multiple names are needed if multiple interfaces will be configured.
IP address and subnet	IPv4 or IPv6 addresses assigned to the interface. Multiple addresses and subnets are needed if multiple interfaces will be configured.
Physical port number	The physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides followed by the number of the physical connector on the card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Gateway IP address	Used when configuring static IP routes from the interface(s) to a specific network.
Gateway IP address	Used when configuring static IP routes from the interface(s) to a specific network.
GTP-U Service Configuration	
GTP-U service name (for S1-U/S11 interface)	An identification string from 1 to 63 characters (alpha and/or numeric) by which the GTP-U service bound to the S1-U/S11 interface will be recognized by the system.
IP address	S1-U/S11 interface IPv4 or IPv6 address.
S-GW Service Configuration	
S-GW service name	An identification string from 1 to 63 characters (alpha and/or numeric) by which the S-GW service is recognized by the system. Multiple names are needed if multiple S-GW services will be used.
eGTP Ingress Service Configuration	
eGTP S1-U/S11 ingress service name	An identification string from 1 to 63 characters (alpha and/or numeric) by which the eGTP S1-U/S11 ingress service is recognized by the system.

Required S-GW Egress Context Configuration Information

The following table lists the information that is required to configure the S-GW egress context on an eGTP S-GW.

Table 3: Required Information for S-GW Egress Context Configuration

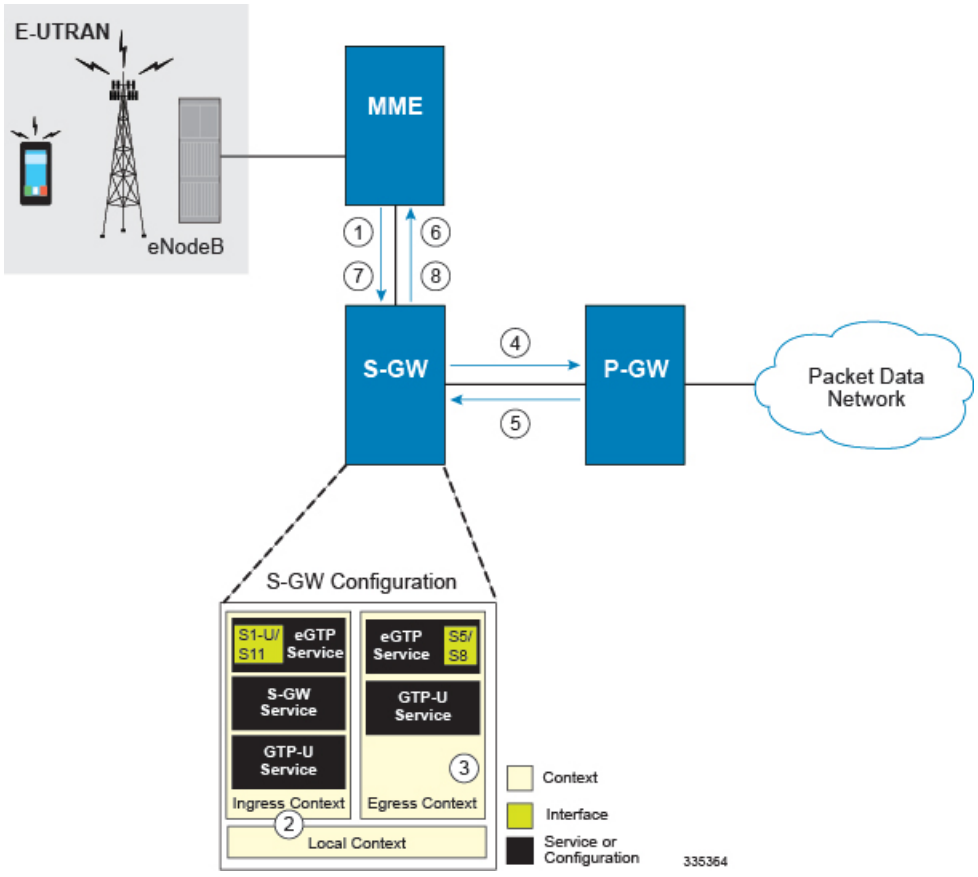
Required Information	Description
S-GW egress context name	An identification string from 1 to 79 characters (alpha and/or numeric) by which the S-GW egress context is recognized by the system.
S5/S8 Interface Configuration (To/from P-GW)	

Required Information	Description
Interface name	An identification string between 1 and 79 characters (alpha and/or numeric) by which the interface is recognized by the system. Multiple names are needed if multiple interfaces will be configured.
IP address and subnet	IPv4 or IPv6 addresses assigned to the interface. Multiple addresses and subnets are needed if multiple interfaces will be configured.
Physical port number	The physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides followed by the number of the physical connector on the card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Gateway IP address	Used when configuring static IP routes from the interface(s) to a specific network.
GTP-U Service Configuration	
GTP-U service name (for S5/S8 interface)	An identification string from 1 to 63 characters (alpha and/or numeric) by which the GTP-U service bound to the S5/S8 interface will be recognized by the system.
IP address	S5/S8 interface IPv4 or IPv6 address.
eGTP Egress Service Configuration	
eGTP Egress Service Name	An identification string from 1 to 63 characters (alpha and/or numeric) by which the eGTP egress service is recognized by the system.

How This Configuration Works

The following figure and supporting text describe how this configuration with a single ingress and egress context is used by the system to process a subscriber call.

Figure 1: eGTP S-GW Call Processing Using a Single Ingress and Egress Context

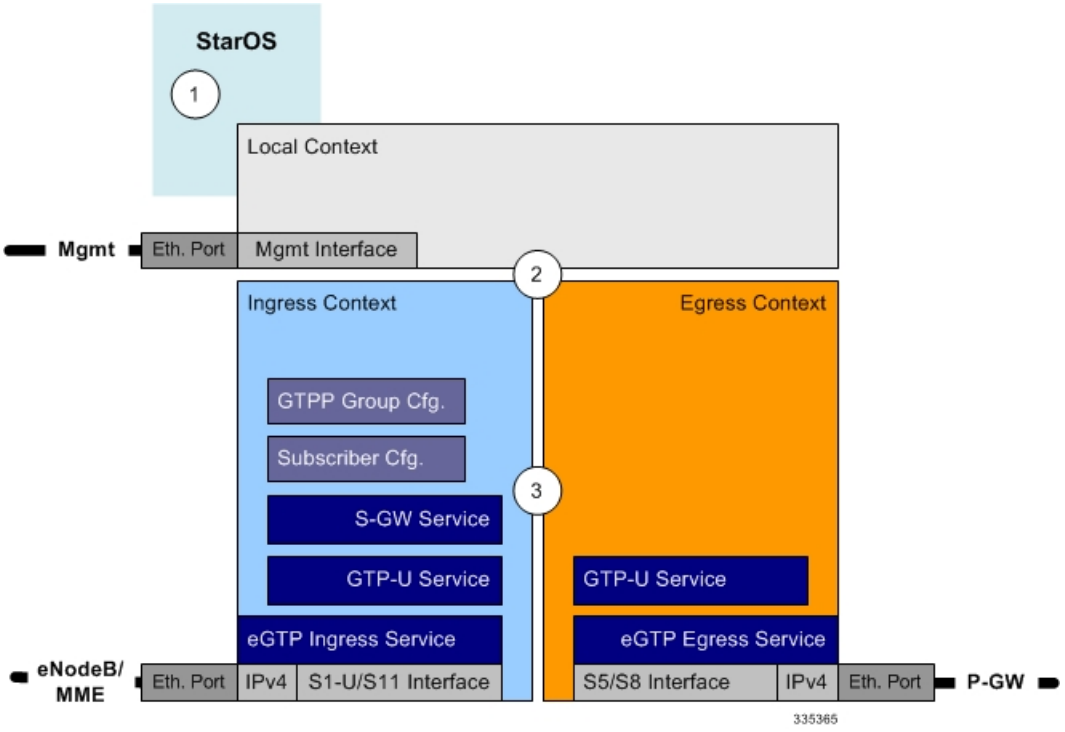


1. A subscriber session from the MME is received by the S-GW service over the S11 interface.
2. The S-GW service determines which context to use to access PDN services for the session. This process is described in the How the System Selects Contexts section located in the Understanding the System Operation and Configuration chapter of the System Administration Guide.
3. S-GW uses the configured egress context to determine the eGTP service to use for the outgoing S5/S8 connection.
4. The S-GW establishes the S5/S8 connection by sending a create session request message to the P-GW.
5. The P-GW responds with a Create Session Response message that includes the PGW S5/S8 Address for control plane and bearer information.
6. The S-GW conveys the control plane and bearer information to the MME in a Create Session Response message.
7. The MME responds with a Create Bearer Response and Modify Bearer Request message.
8. The S-GW sends a Modify Bearer Response message to the MME.

eGTP S-GW Configuration

To configure the system to perform as a standalone eGTP S-GW, review the following graphic and subsequent steps.

Figure 2: eGTP S-GW Configurable Components



Step 1 Set system configuration parameters such as activating PSCs by applying the example configurations found in the System Administration Guide.

Step 2 Set initial configuration parameters such as creating contexts and services by applying the example configurations found in theInitial Configuration, on page 6.

Step 3 Configure the system to perform as an eGTP S-GW and set basic S-GW parameters such as eGTP interfaces and an IP route by applying the example configurations presented in the eGTP Configuration, on page 9.

Step 4 Verify and save the configuration by following the instruction in the Verifying and Saving the Configuration, on page 10.

Initial Configuration

Step 1 Set local system management parameters by applying the example configuration in the Modifying the Local Context, on page 7.

Step 2 Create an ingress context where the S-GW and eGTP ingress service will reside by applying the example configuration in the Creating an S-GW Ingress Context, on page 7.

Step 3 Create an eGTP ingress service within the newly created ingress context by applying the example configuration in the Creating an eGTP Ingress Service, on page 8.

Step 4 Create an S-GW egress context where the eGTP egress services will reside by applying the example configuration in the Creating an S-GW Egress Context, on page 8.

Step 5 Create an eGTP egress service within the newly created egress context by applying the example configuration in the Creating an eGTP Egress Service, on page 8.

Step 6 Create a S-GW service within the newly created ingress context by applying the example configuration in the Creating an S-GW Service, on page 8.

Modifying the Local Context

Use the following example to set the default subscriber and configure remote access capability in the local context:

```
configure
  context local
    interface <lcl_cntxt_intrfc_name>
      ip address <ip_address> <ip_mask>
      exit
    server ftpd
      exit
    server telnetd
      exit
    subscriber default
      exit
    administrator <name> encrypted password <password> ftp
    ip route <ip_addr/ip_mask> <next_hop_addr> <lcl_cntxt_intrfc_name>
      exit
    port ethernet <slot/port>
      no shutdown
      bind interface <lcl_cntxt_intrfc_name> local
      end
```

Creating an S-GW Ingress Context

Use the following example to create an S-GW ingress context and Ethernet interfaces to an MME and eNodeB, and bind the interfaces to configured Ethernet ports.

```
configure
  context <ingress_context_name> -noconfirm
    subscriber default
    exit
  interface <s1u-s11_interface_name>
    ip address <ipv4_address_primary>
    ip address <ipv4_address_secondary>
    exit
  ip route 0.0.0.0 0.0.0.0 <next_hop_address> <sgw_interface_name>
    exit
  port ethernet <slot_number/port_number>
    no shutdown
```

```
   bind interface <s1u-s11_interface_name> <ingress_context_name>
   end
```

Notes:

- This example presents the S1-U/S11 connections as a shared interface. These interfaces can be separated to support a different network architecture.
- The S1-U/S11 interface IP address(es) can also be specified as IPv6 addresses using the **ipv6 address** command.

Creating an eGTP Ingress Service

Use the following configuration example to create an eGTP ingress service:

```
configure
  context <ingress_context_name>
    egtp-service <egtp_ingress_service_name> -noconfirm
    end
```

Creating an S-GW Egress Context

Use the following example to create an S-GW egress context and Ethernet interface to a P-GW and bind the interface to configured Ethernet ports.

```
configure
  context <egress_context_name> -noconfirm
    interface <s5s8_interface_name> tunnel
      ipv6 address <address>
        tunnel-mode ipv6ip
          source interface <name>
          destination address <ipv4 or ipv6 address>
          end
configure
  port ethernet <slot_number/port_number>
    no shutdown
    bind interface <s5s8_interface_name> <egress_context_name>
    end
```

Notes:

- The S5/S8 interface IP address can also be specified as an IPv4 address using the **ip address** command.

Creating an eGTP Egress Service

Use the following configuration example to create an eGTP egress service in the S-GW egress context:

```
configure
  context <egress_context_name>
    egtp-service <egtp_egress_service_name> -noconfirm
    end
```

Creating an S-GW Service

Use the following configuration example to create the S-GW service in the ingress context:

```
configure
   context <ingress_context_name>
      sgw-service <sgw_service_name> -noconfirm
      end
```

eGTP Configuration

Step 1 Set the system's role as an eGTP S-GW and configure eGTP service settings by applying the example configuration in the Setting the System's Role as an eGTP S-GW and Configuring GTP-U and eGTP Service Settings, on page 9.

Step 2 Configure the S-GW service by applying the example configuration in the Configuring the S-GW Service, on page 10.

Step 3 Specify an IP route to the eGTP Serving Gateway by applying the example configuration in the Configuring an IP Route, on page 10.

Setting the System's Role as an eGTP S-GW and Configuring GTP-U and eGTP Service Settings

Use the following configuration example to set the system to perform as an eGTP S-GW and configure the GTP-U and eGTP services:

```
configure
   context <sgw_ingress_context_name>
      gtpp group default
         exit
      gtpu-service <gtpu_ingress_service_name>
         bind ipv4-address <s1-u_s11_interface_ip_address>
            exit
         egtp-service <egtp_ingress_service_name>
            interface-type interface-sgw-ingress
            validation-mode default
            associate gtpu-service <gtpu_ingress_service_name>
            gtpc bind address <s1u-s11_interface_ip_address>
               exit
            exit
   context <sgw_egress_context_name>
      gtpu-service <gtpu_egress_service_name>
         bind ipv4-address <s5s8_interface_ip_address>
         exit
      egtp-service <egtp_egress_service_name>
         interface-type interface-sgw-egress
         validation-mode default
         associate gtpu-service <gtpu_egress_service_name>
         gtpc bind address <s5s8_interface_ip_address>
         end
```

Notes:

- The **bind** command in the GTP-U ingress and egress service configuration can also be specified as an IPv6 address using the **ipv6-address** command.

Configuring the S-GW Service

Use the following example to configure the S-GW service:

```
configure
   context <ingress_context_name>
      sgw-service <sgw_service_name> -noconfirm
         associate ingress egtp-service <egtp_ingress_service_name>
         associate egress-proto gtp egress-context <egress_context_name>
         qci-qos-mapping <map_name>
         end
```

Configuring an IP Route

Use the following example to configure an IP Route for control and user plane data communication with an eGTP PDN Gateway:

```
configure
   context <egress_context_name>
      ip route <pgw_ip_addr/mask> <sgw_next_hop_addr> <sgw_intrfc_name>
      end
```

Verifying and Saving the Configuration

Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Configuring Optional Features on the eGTP S-GW

The configuration examples in this section are optional and provided to cover the most common uses of the eGTP S-GW in a live network. The intent of these examples is to provide a base configuration for testing.

Configuring the GTP Echo Timer

The GTP echo timer on the ASR 5500 S-GW can be configured to support two different types of path management: default and dynamic. This timer can be configured on the GTP-C and/or the GTP-U channels.

Default GTP Echo Timer Configuration

The following examples describe the configuration of the default eGTP-C and GTP-U interface echo timers:

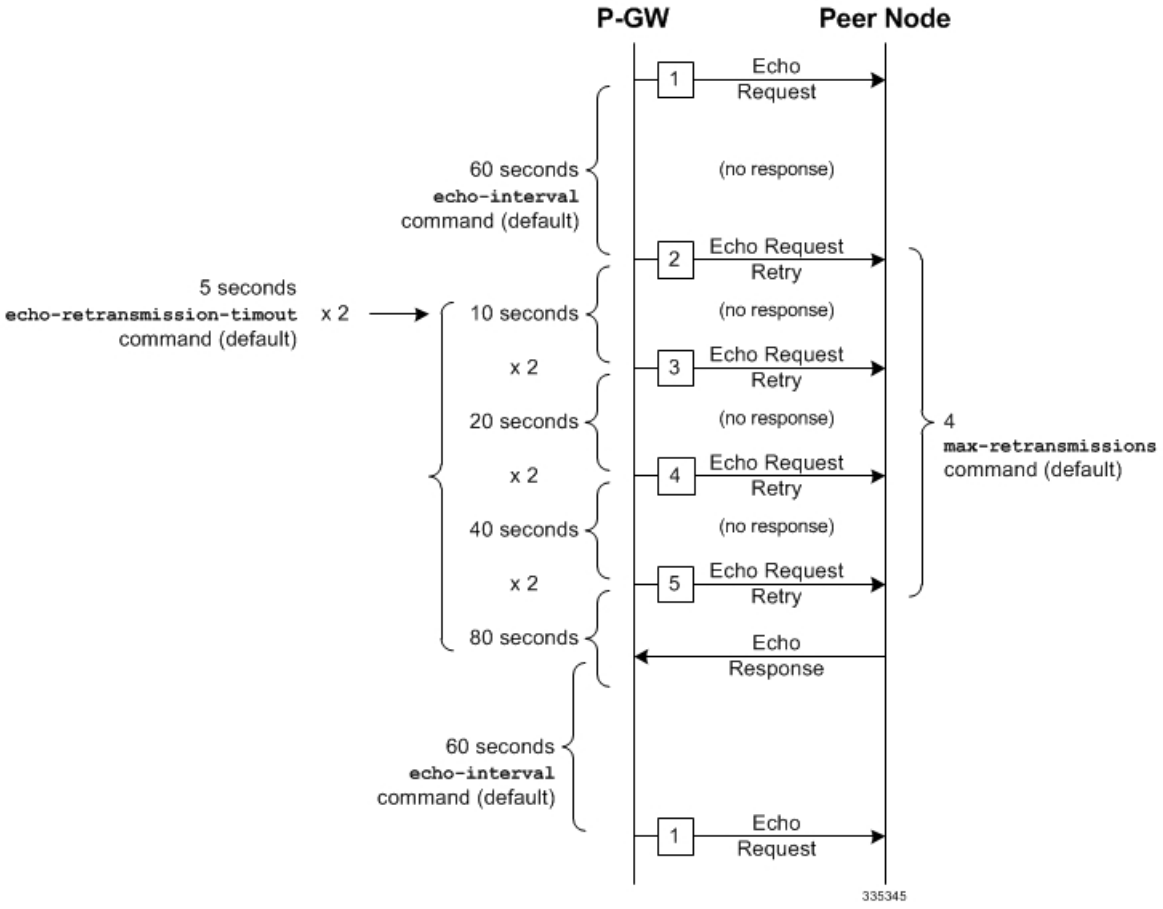
eGTP-C

```
configure
   context <context_name>
      egtp-service <egtp_service_name>
         gtpc echo-interval <seconds>
         gtpc echo-retransmission-timeout <seconds>
         gtpc max-retransmissions <num>
         end
```

Notes:

- This configuration can be used in either the ingress context supporting the S1-U and/or S11 interfaces with the eNodeB and MME respectively; and the egress context supporting the S5/S8 interface with the P-GW.
- The following diagram describes a failure and recovery scenario using default settings of the three **gtpc** commands in the example above:

Figure 3: Failure and Recovery Scenario: Example 1



- The multiplier (x2) is system-coded and cannot be configured.

GTP-U

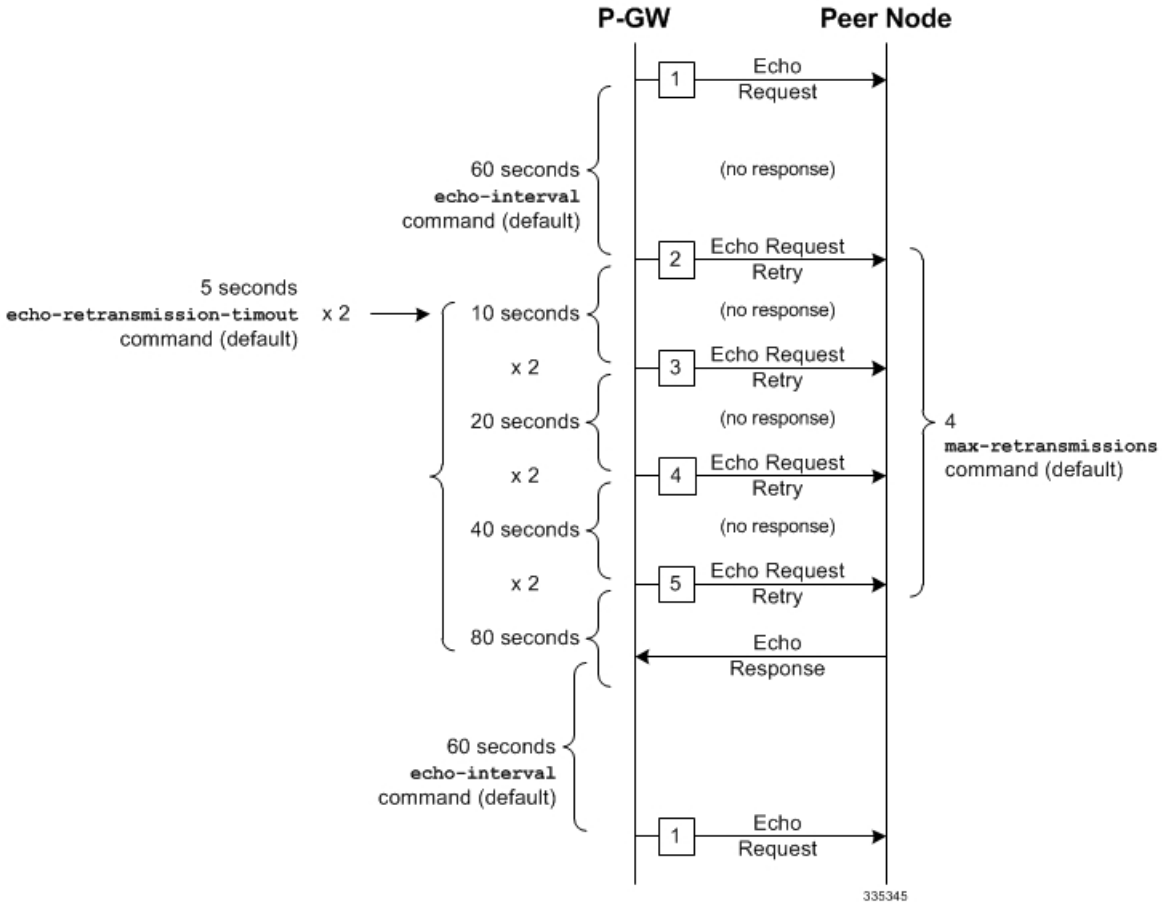
```
configure
   context <context_name>
      gtpu-service <gtpu_service_name>
         echo-interval <seconds>
         echo-retransmission-timeout <seconds>
         max-retransmissions <num>
         end
```

Notes:

- This configuration can be used in either the ingress context supporting the S1-U interfaces with the eNodeB and the egress context supporting the S5/S8 interface with the P-GW.

- The following diagram describes a failure and recovery scenario using default settings of the three GTP-U commands in the example above:

Figure 4: Failure and Recovery Scenario: Example 2



- The multiplier (x2) is system-coded and cannot be configured.

Dynamic GTP Echo Timer Configuration

The following examples describe the configuration of the dynamic eGTP-C and GTP-U interface echo timers:

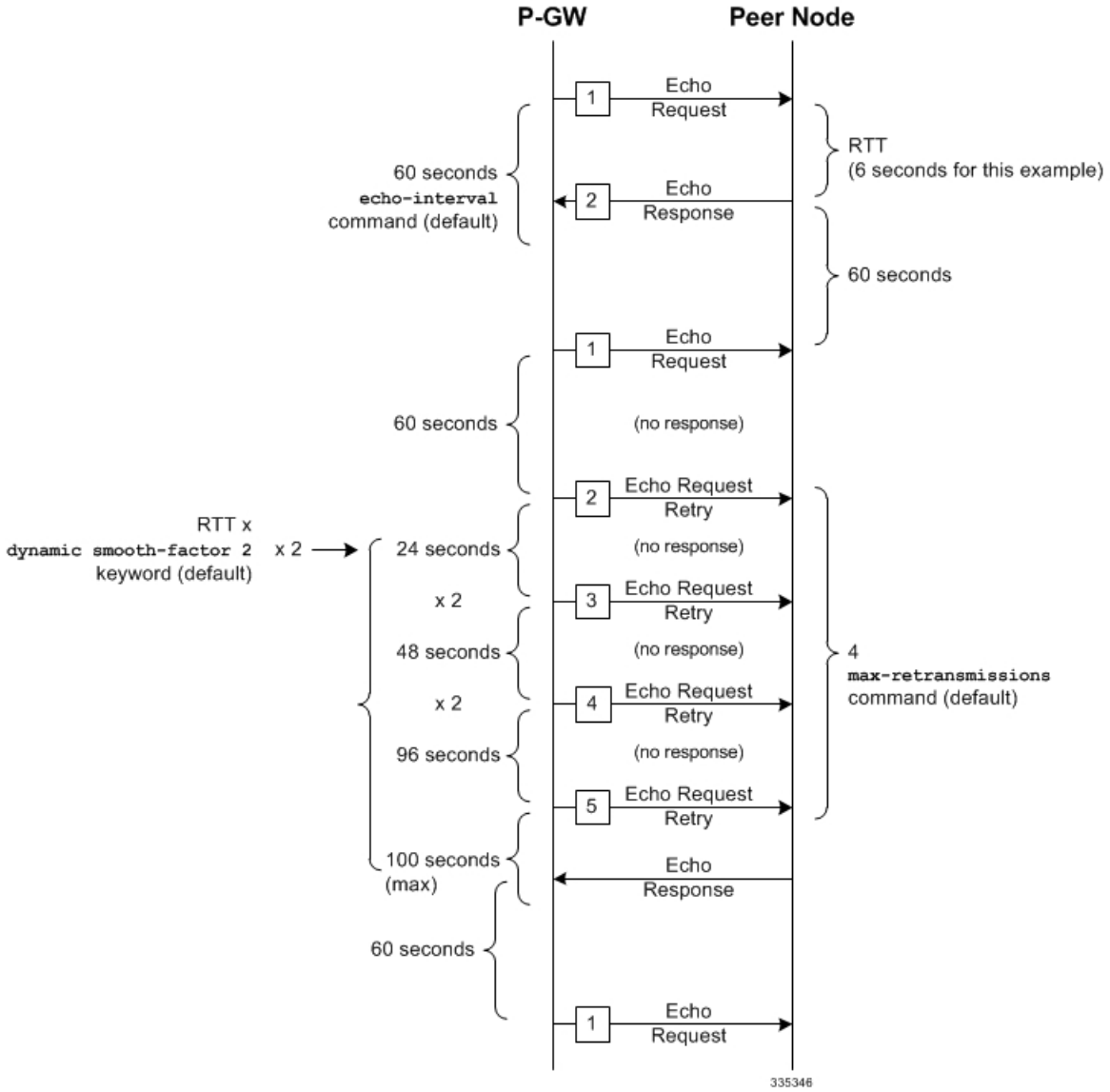
eGTP-C

```
configure
  context <context_name>
    egtp-service <egtp_service_name>
      gtpc echo-interval <seconds> dynamic smooth-factor <multiplier>
      gtpc echo-retransmission-timeout <seconds>
      gtpc max-retransmissions <num>
      end
```

Notes:

- This configuration can be used in either the ingress context supporting the S1-U and/or S11 interfaces with the eNodeB and MME respectively; and the egress context supporting the S5/S8 interface with the P-GW.
- The following diagram describes a failure and recovery scenario using default settings of the three **gtpc** commands in the example above and an example round trip timer (RTT) of six seconds:

Figure 5: Failure and Recovery Scenario: Example 3



- The multiplier (x2) and the 100 second maximum are system-coded and cannot be configured.

GTP-U

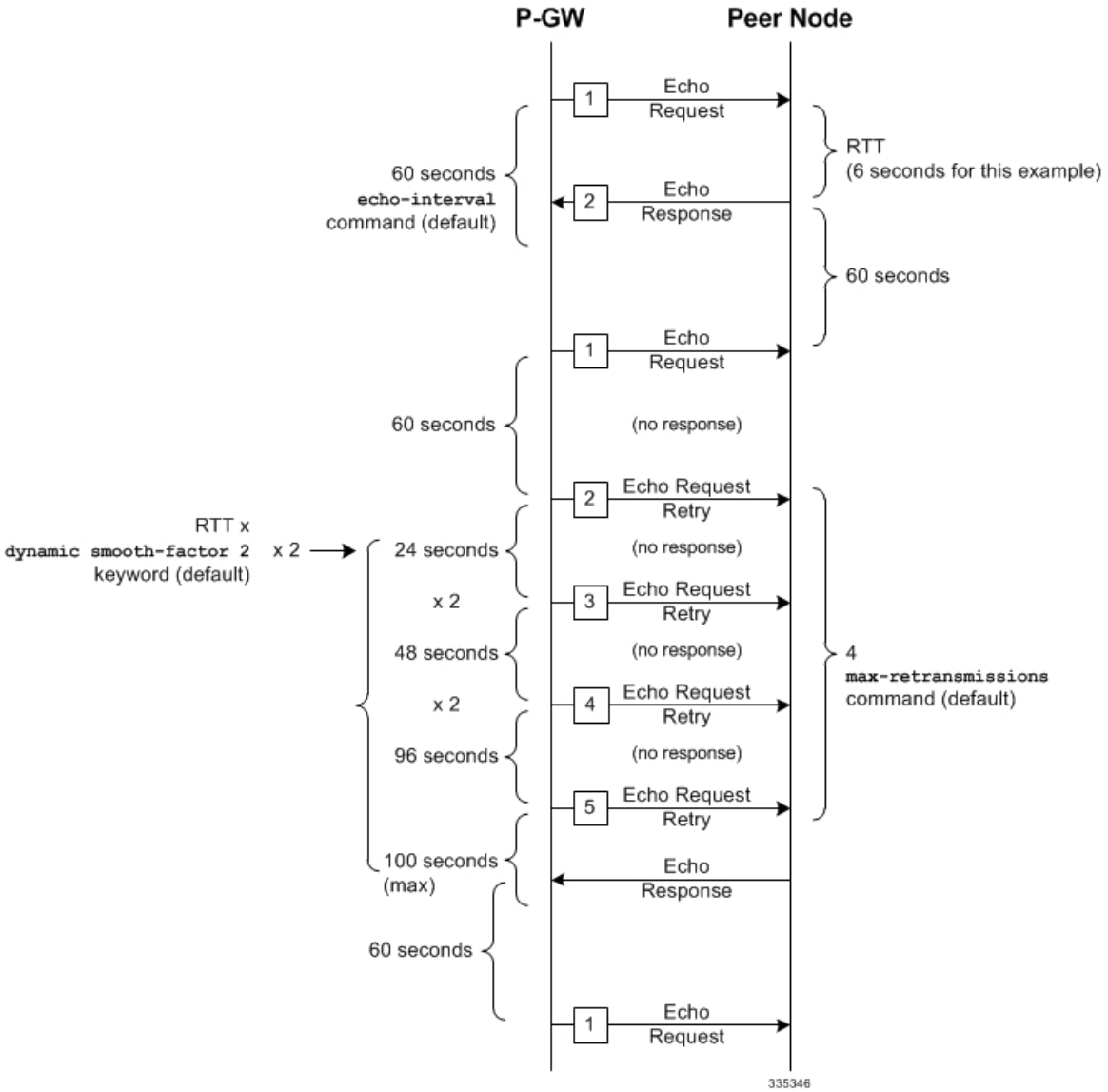
```
configure
  context <context_name>
    gtpu-service <gtpu_service_name>
      echo-interval <seconds> dynamic smooth-factor <multiplier>
```

```
echo-retransmission-timeout <seconds>
max-retransmissions <num>
end
```

Notes:

- This configuration can be used in either the ingress context supporting the S1-U interfaces with the eNodeB and the egress context supporting the S5/S8 interface with the P-GW.
- The following diagram describes a failure and recovery scenario using default settings of the three **gtpc** commands in the example above and an example round trip timer (RTT) of six seconds:

Figure 6: Failure and Recovery Scenario: Example 4



- The multiplier (x2) and the 100 second maximum are system-coded and cannot be configured.

Configuring GTPP Offline Accounting on the S-GW

By default the S-GW service supports GTPP accounting. To provide GTPP offline charging during, for example, scenarios where the foreign P-GW does not, configure the S-GW with the example parameters below:

```
configure
  gtpp single-source
    context <ingress_context_name>
      subscriber default
        accounting mode gtpp
        exit
      gtpp group default
        gtpp charging-agent address <gz_ipv4_address>
        gtpp echo-interval <seconds>
        gtpp attribute diagnostics
        gtpp attribute local-record-sequence-number
        gtpp attribute node-id-suffix <string>
        gtpp dictionary <name>
        gtpp server <ipv4_address> priority <num>
        gtpp server <ipv4_address> priority <num> node-alive enable
        exit
      policy accounting <gz_policy_name>
        accounting-level {type}
        operator-string <string>
        cc profile <index> buckets <num>
        cc profile <index> interval <seconds>
        cc profile <index> volume total <octets>
        exit
      sgw-service <sgw_service_name>
        accounting context <ingress_context_name> gtpp group default
        associate accounting-policy <gz_policy_name>
        exit
      exit
    context <ingress_context_name>
      interface <gz_interface_name>
        ip address <address>
      exit
    exit
  port ethernet <slot_number/port_number>
    no shutdown
    bind interface <gz_interface_name> <ingress_context_name>
    end
```

Notes:

- **gtpp single-source** is enabled to allow the system to generate requests to the accounting server using a single UDP port (by way of a AAA proxy function) rather than each AAA manager generating requests on unique UDP ports.
- **gtpp** is the default option for the **accounting mode** command.
- An accounting mode configured for the call-control profile will override this setting.

- **accounting-level** types are: flow, PDN, PDN-QCI, QCI, and subscriber. Refer to the Accounting Profile Configuration Mode Commands chapter in the *Command Line Interface Reference* for more information on this command.

Configuring Diameter Offline Accounting on the S-GW

By default the S-GW service supports GTPP accounting. You can enable accounting via RADIUS/Diameter (Rf) for the S-GW service. To provide Rf offline charging during, for example, scenarios where the foreign P-GW does not, configure the S-GW with the example parameters below:

Important Diameter Offline Accounting is not supported on the S-GW.

```
configure
  operator-policy name <policy_name>
    associate call-control-profile <call_cntrl_profile_name>
    exit
  call-control-profile <call_cntrl_profile_name>
    accounting mode radius-diameter
    exit
  lte-policy
    subscriber-map <map_name>
      precendence <number> match-criteria all operator-policy-name
<policy_name>
      exit
    exit
  context <ingress_context_name>
    policy accounting <rf_policy_name>
      accounting-level {type}
      operator-string <string>
      exit
    sgw-service <sgw_service_name>
      associate accounting-policy <rf_policy_name>
      associate subscriber-map <map_name>
      exit
    aaa group <rf-radius_group_name>
      radius attribute nas-identifier <id>
      radius accounting interim interval <seconds>
      radius dictionary <name>
      radius mediation-device accounting server <address> key <key>
      diameter authentication dictionary <name>
      diameter accounting dictionary <name>
      diameter accounting endpoint <rf_cfg_name>
      diameter accounting server <rf_cfg_name> priority <num>
      exit
    diameter endpoint <rf_cfg_name>
      use-proxy
      origin realm <realm_name>
      origin host <name> address <rf_ipv4_address>
      peer <rf_cfg_name> realm <name> address <ofcs_ipv4_or_ipv6_addr>
```

```
      route-entry peer <rf_cfg_name>
      exit
    exit
  context <ingress_context_name>
    interface <rf_interface_name>
      ip address <rf_ipv4_address>
      exit
    exit
  port ethernet <slot_number/port_number>
    no shutdown
    bind interface <rf_interface_name> <ingress_context_name>
    end
```

Notes:

- **accounting-level** types are: flow, PDN, PDN-QCI, QCI, and subscriber. Refer to the Accounting Profile Configuration Mode Commands chapter in the *Command Line Interface Reference* for more information on this command.
- The Rf interface IP address can also be specified as an IPv6 address using the **ipv6 address** command.

Configuring APN-level Traffic Policing on the S-GW

To enable traffic policing for scenarios where the foreign subscriber's P-GW doesn't enforce it, use the following configuration example:

```
configure
  apn-profile <apn_profile_name>
    qos rate-limit downlink non-gbr-qci committed-auto-readjust duration
 <seconds> exceed-action {action} violate-action {action}
    qos rate-limit uplink non-gbr-qci committed-auto-readjust duration
<seconds> exceed-action {action} violate-action {action}
    exit
  operator-policy name <policy_name>
    apn default-apn-profile <apn_profile_name>
    exit
  lte-policy
    subscriber-map <map_name>
      precendence <number> match-criteria all operator-policy-name
<policy_name>
      exit
    sgw-service <sgw_service_name>
      associate subscriber-map <map_name>
      end
```

Notes:

- For the **qos rate-limit** command, the actions supported for **violate-action** and **exceed-action** are: **drop**, **lower-ip-precedence**, and **transmit**.

Configuring X.509 Certificate-based Peer Authentication

The configuration example in this section enables X.509 certificate-based peer authentication, which can be used as the authentication method for IP Security on the S-GW.

Important Use of the IP Security feature requires that a valid license key be installed. Contact your local Sales or Support representative for information on how to obtain a license.

The following configuration example enables X.509 certificate-based peer authentication on the S-GW.

In Global Configuration Mode, specify the name of the X.509 certificate and CA certificate, as follows:

```
configure
   certificate name <cert_name> pem url <cert_pem_url> private-key pem url
<private_key_url>
   ca-certificate name <ca_cert_name> pem url <ca_cert_url>
   end
```

Notes:

- The **certificate name** and **ca-certificate list ca-cert-name** commands specify the X.509 certificate and CA certificate to be used.
- The PEM-formatted data for the certificate and CA certificate can be specified, or the information can be read from a file via a specified URL as shown in this example.

When creating the crypto template for IPSec in Context Configuration Mode, bind the X.509 certificate and CA certificate to the crypto template and enable X.509 certificate-based peer authentication for the local and remote nodes, as follows:

```
configure
   context <sgw_context_name>
      crypto template <crypto_template_name> ikev2-dynamic
         certificate name <cert_name>
         ca-certificate list ca-cert-name <ca_cert_name>
         authentication local certificate
         authentication remote certificate
         end
```

Notes:

- A maximum of sixteen certificates and sixteen CA certificates are supported per system. One certificate is supported per service, and a maximum of four CA certificates can be bound to one crypto template.
- The **certificate name** and **ca-certificate list ca-cert-name** commands bind the certificate and CA certificate to the crypto template.
- The **authentication local certificate** and **authentication remote certificate** commands enable X.509 certificate-based peer authentication for the local and remote nodes.

Configuring Dynamic Node-to-Node IP Security on the S1-U and S5 Interfaces

The configuration example in this section creates IPSec/IKEv2 dynamic node-to-node tunnel endpoints on the S1-U and S5 interfaces.

Important Use of the IP Security feature requires that a valid license key be installed. Contact your local Sales or Support representative for information on how to obtain a license.

Creating and Configuring an IPSec Transform Set

The following example configures an IPSec transform set, which is used to define the security association that determines the protocols used to protect the data on the interface:

```
configure
  context <sgw_context_name>
    ipsec transform-set <ipsec_transform-set_name>
      encryption aes-cbc-128
      group none
      hmac sha1-96
      mode tunnel
      end
```

Notes:

- The encryption algorithm, **aes-cbc-128**, or Advanced Encryption Standard Cipher Block Chaining, is the default algorithm for IPSec transform sets configured on the system.
- The **group none** command specifies that no crypto strength is included and that Perfect Forward Secrecy is disabled. This is the default setting for IPSec transform sets configured on the system.
- The **hmac** command configures the Encapsulating Security Payload (ESP) integrity algorithm. The **sha1-96** keyword uses a 160-bit secret key to produce a 160-bit authenticator value. This is the default setting for IPSec transform sets configured on the system.
- The **mode tunnel** command specifies that the entire packet is to be encapsulated by the IPSec header, including the IP header. This is the default setting for IPSec transform sets configured on the system.

Creating and Configuring an IKEv2 Transform Set

The following example configures an IKEv2 transform set:

```
configure
  context <sgw_context_name>
    ikev2-ikesa transform-set <ikev2_transform-set_name>
      encryption aes-cbc-128
      group 2
      hmac sha1-96
      lifetime <sec>
      prf sha1
      end
```

Notes:

- The encryption algorithm, **aes-cbc-128**, or Advanced Encryption Standard Cipher Block Chaining, is the default algorithm for IKEv2 transform sets configured on the system.
- The **group 2** command specifies the Diffie-Hellman algorithm as Group 2, indicating medium security. The Diffie-Hellman algorithm controls the strength of the crypto exponentials. This is the default setting for IKEv2 transform sets configured on the system.
- The **hmac** command configures the Encapsulating Security Payload (ESP) integrity algorithm. The **sha1-96** keyword uses a 160-bit secret key to produce a 160-bit authenticator value. This is the default setting for IKEv2 transform sets configured on the system.
- The **lifetime** command configures the time the security key is allowed to exist, in seconds.

- The **prf** command configures the IKE Pseudo-random Function, which produces a string of bits that cannot be distinguished from a random bit string without knowledge of the secret key. The **sha1** keyword uses a 160-bit secret key to produce a 160-bit authenticator value. This is the default setting for IKEv2 transform sets configured on the system.

Creating and Configuring a Crypto Template

The following example configures an IKEv2 crypto template:

```
configure
   context <sgw_context_name>
      crypto template <crypto_template_name> ikev2-dynamic
         ikev2-ikesa transform-set list <name1> . . . <name6>
         ikev2-ikesa rekey
         payload <name> match childsa match ipv4
            ipsec transform-set list <name1> . . . <name4>
            rekey
            end
```

Notes:

- The **ikev2-ikesa transform-set list** command specifies up to six IKEv2 transform sets.
- The **ipsec transform-set list** command specifies up to four IPSec transform sets.

Binding the S1-U and S5 IP Addresses to the Crypto Template

The following example configures the binding of the S1-U and S5 interfaces to the crypto template.

```
configure
   context <sgw_ingress_context_name>
      gtpu-service <gtpu_ingress_service_name>
         bind ipv4-address <s1-u_interface_ip_address> crypto-template
<enodeb_crypto_template>
         exit
      egtp-service <egtp_ingress_service_name>
         interface-type interface-sgw-ingress
         associate gtpu-service <gtpu_ingress_service_name>
         gtpc bind address <s1u_interface_ip_address>
         exit
      exit
   context <sgw_egress_context_name>
      gtpu-service <gtpu_egress_service_name>
         bind ipv4-address <s5_interface_ip_address> crypto-template
<enodeb_crypto_template>
         exit
      egtp-service <egtp_egress_service_name>
         interface-type interface-sgw-egress
         associate gtpu-service <gtpu_egress_service_name>
         gtpc bind address <s5_interface_ip_address>
         exit
      exit
   context <sgw_ingress_context_name>
      sgw-service <sgw_service_name> -noconfirm
```

```
egtp-service ingress service <egtp_ingress_service_name>
egtp-service egress context <sgw_egress_context_name>
end
```

Notes:

- The **bind** command in the GTP-U ingress and egress service configuration can also be specified as an IPv6 address using the **ipv6-address** command.

Configuring ACL-based Node-to-Node IP Security on the S1-U and S5 Interfaces

The configuration example in this section creates IKEv2/IPSec ACL-based node-to-node tunnel endpoints on the S1-U and S5 interfaces.

Important Use of the IP Security feature requires that a valid license key be installed. Contact your local Sales or Support representative for information on how to obtain a license.

Creating and Configuring a Crypto Access Control List

The following example configures a crypto ACL (Access Control List), which defines the matching criteria used for routing subscriber data packets over an IPSec tunnel:

```
configure
  context <sgw_context_name>
    ip access-list <acl_name>
      permit tcp host <source_host_address> host <dest_host_address>
      end
```

Notes:

- The **permit** command in this example routes IPv4 traffic from the server with the specified source host IPv4 address to the server with the specified destination host IPv4 address.

Creating and Configuring an IPSec Transform Set

The following example configures an IPSec transform set which is used to define the security association that determines the protocols used to protect the data on the interface:

```
configure
  context <sgw_context_name>
    ipsec transform-set <ipsec_transform-set_name>
      encryption aes-cbc-128
      group none
      hmac sha1-96
      mode tunnel
      end
```

Notes:

- The encryption algorithm, **aes-cbc-128**, or Advanced Encryption Standard Cipher Block Chaining, is the default algorithm for IPSec transform sets configured on the system.
- The **group none** command specifies that no crypto strength is included and that Perfect Forward Secrecy is disabled. This is the default setting for IPSec transform sets configured on the system.

- The **hmac** command configures the Encapsulating Security Payload (ESP) integrity algorithm. The **sha1-96** keyword uses a 160-bit secret key to produce a 160-bit authenticator value. This is the default setting for IPSec transform sets configured on the system.
- The **mode tunnel** command specifies that the entire packet is to be encapsulated by the IPSec header including the IP header. This is the default setting for IPSec transform sets configured on the system.

Creating and Configuring an IKEv2 Transform Set

The following example configures an IKEv2 transform set:

```
configure
  context <sgw_context_name>
    ikev2-ikesa transform-set <ikev2_transform-set_name>
      encryption aes-cbc-128
      group 2
      hmac sha1-96
      lifetime <sec>
      prf sha1
      end
```

Notes:

- The encryption algorithm, **aes-cbc-128**, or Advanced Encryption Standard Cipher Block Chaining, is the default algorithm for IKEv2 transform sets configured on the system.
- The **group 2** command specifies the Diffie-Hellman algorithm as Group 2, indicating medium security. The Diffie-Hellman algorithm controls the strength of the crypto exponentials. This is the default setting for IKEv2 transform sets configured on the system.
- The **hmac** command configures the Encapsulating Security Payload (ESP) integrity algorithm. The **sha1-96** keyword uses a 160-bit secret key to produce a 160-bit authenticator value. This is the default setting for IKEv2 transform sets configured on the system.
- The **lifetime** command configures the time the security key is allowed to exist, in seconds.
- The **prf** command configures the IKE Pseudo-random Function which produces a string of bits that cannot be distinguished from a random bit string without knowledge of the secret key. The **sha1** keyword uses a 160-bit secret key to produce a 160-bit authenticator value. This is the default setting for IKEv2 transform sets configured on the system.

Creating and Configuring a Crypto Map

The following example configures an IKEv2 crypto map and applies it to the S1-U interface:

```
configure
  context <sgw_ingress_context_name>
    crypto map <crypto_map_name> ikev2-ipv4
      match address <acl_name>
      peer <ipv4_address>
      authentication local pre-shared-key key <text>
      authentication remote pre-shared-key key <text>
      ikev2-ikesa transform-set list <name1> . . . <name6>
      payload <name> match ipv4
        lifetime <seconds>
```

```
        ipsec transform-set list <name1> . . . <name4>
        exit
      exit
    interface <s1-u_intf_name>
      ip address <ipv4_address>
      crypto-map <crypto_map_name>
      exit
    exit
  port ethernet <slot_number/port_number>
    no shutdown
    bind interface <s1_u_intf_name> <sgw_ingress_context_name>
    end
```

Notes:

- The type of crypto map used in this example is IKEv2-IPv4 for IPv4 addressing. An IKEv2-IPv6 crypto map can also be used for IPv6 addressing.
- The **ipsec transform-set list** command specifies up to four IPSec transform sets.

The following example configures an IKEv2 crypto map and applies it to the S5 interface:

```
configure
  context <sgw_egress_context_name>
    crypto map <crypto_map_name> ikev2-ipv4
      match address <acl_name>
      peer <ipv4_address>
      authentication local pre-shared-key key <text>
      authentication remote pre-shared-key key <text>
      payload <name> match ipv4
        lifetime <seconds>
        ipsec transform-set list <name1> . . . <name4>
        exit
      exit
    interface <s5_intf_name>
      ip address <ipv4_address>
      crypto map <crypto_map_name>
      exit
    exit
  port ethernet <slot_number/port_number>
      no shutdown
      bind interface <s5_intf_name> <sgw_egress_context_name>
      end
```

Notes:

- The type of crypto map used in this example is IKEv2-IPv4 for IPv4 addressing. An IKEv2-IPv6 crypto map can also be used for IPv6 addressing.
- The **ipsec transform-set list** command specifies up to four IPSec transform sets.

Configuring 3GPP Release 12 Load Control Support

3GPP R12 Load Control enables a GTP-C entity (for example, an S-GW/P-GW) to send its load information to a GTP-C peer (e.g. an MME/SGSN, ePDG, TWAN) to adaptively balance the session load across entities supporting the same function (for example, an S-GW cluster) according to their effective load. The load information reflects the operating status of the resources of the GTP-C entity.

Use the following example to configure this feature:

```
configure
  gtpc-load-control-profile profile_name
    inclusion-frequency advertisement-interval interval_in_seconds
   weightage system-cpu-utilization percentage system-memory-utilization
 percentage license-session-utilization percentage
    end
configure
  context context_name
    sgw-service sgw_service_name
      associate gtpc-load-control-profile profile_name
      end
```

Notes:

- The **inclusion-frequency** parameter determines how often the Load control information element is sent to the peer(s).
- The total of the three **weightage** parameters should not exceed 100.
- The **associate** command is used to associate the Load Control Profile with an existing S-GW service.

Configuring 3GPP Release 12 Overload Control Support

3GPP R12 Overload Control enables a GTP-C entity becoming or being overloaded to gracefully reduce its incoming signalling load by instructing its GTP-C peers to reduce sending traffic according to its available signaling capacity to successfully process the traffic. A GTP-C entity is in overload when it operates over its signaling capacity, which results in diminished performance (including impacts to handling of incoming and outgoing traffic).

Use the following example to configure this feature.

```
configure
  gtpc-overload-control-profile profile_name
    inclusion-frequency advertisement-interval interval_in_seconds
    weightage system-cpu-utilization percentage system-memory-utilization
percentage license-session-utilization percentage
    throttling-behavior emergency-events exclude
    tolerance threshold report-reduction-metric percentage
self-protection-limit percentage
    validity-period seconds
    end
configure
  context context_name
  sgw-service sgw_service_name
  associate gtpc-overload-control-profile profile_name
  end
```

Notes:

- The **inclusion-frequency** parameter determines how often the Overload control information element is sent to the peer(s).
- The total of the three **weightage** parameters should not exceed 100.
- **validity-period** configures how long the overload control information is valid. Valid entries are from 1 to 3600 seconds. The default is 600 seconds.
- The **associate** command is used to associate the Overload Control Profile with an existing S-GW service.

Configuring S4 SGSN Handover Capability

This configuration example configures an S4 interface supporting inter-RAT handovers between the S-GW and an S4 SGSN.

Use the following example to configure this feature:

```
configure
  context <ingress_context_name> -noconfirm
    interface <s4_interface_name>
      ip address <ipv4_address_primary>
      ip address <ipv4_address_secondary>
      exit
    exit
  port ethernet <slot_number/port_number>
    no shutdown
    bind interface <s4_interface_name> <ingress_context_name>
    exit
  context <ingress_context_name> -noconfirm
    gtpu-service <s4_gtpu_ingress_service_name>
      bind ipv4-address <s4_interface_ip_address>
      exit
    egtp-service <s4_egtp_ingress_service_name>
      interface-type interface-sgw-ingress
      validation-mode default
      associate gtpu-service <s4_gtpu_ingress_service_name>
      gtpc bind address <s4_interface_ip_address>
      exit
    sgw-service <sgw_service_name> -noconfirm
      associate ingress egtp-service <s4_egtp_ingress_service_name>
      end
```

Notes:

- The S4 interface IP address(es) can also be specified as IPv6 addresses using the **ipv6 address** command.