Application Awareness Capability on ZBFW

This document describes a solution based on the NBAR classification engine. In this case, you can define Zone-Based FW policy (ZBFW) based on the applications that NBAR can detect and identify the ZBFW application.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About Application Awareness capability on ZBFW

GPRS Overview

Current zone based firewall uses policies based on network layers L3/L4, for example, class map are based on ACL and L4 protocolsTCP/UDP/ICMP or L7 protocolsFTP and SIP. Even where the policy can be defined using the L7 protocol, those policies utilize the protocol's destination port to classify the packet. ZBF lacks application visibility, it supports FTP inspection through the FTP ALG, but only identifies the protocol based on the well-known port, usually port 21. If an FTP control flow is opened on some random port, ZBF has no way to identify the application.

GTP Overview

General Packet Radio Service (GPRS) Tunneling Protocol (GTP) allows multiprotocol packets to be tunneled through the GPRS backbone between GPRS Support Nodes (GSN). Three GTP versions are available. The GPRS Tunneling Support feature supports two GTP versions: GTP Version 0 (GTPv0) and GTP Version 1 (GTPv1).

In GTPv0, a GPRS Mobile Station (MS) is connected to a Serving GPRS Support Node (SGSN) without being aware of the protocol. A Packet Data Protocol (PDP) context is identified by the Tunnel Identifier (TID), which is a combination of the International Mobile Subscriber Identity (IMSI) and the Network Service Access Point Identifier (NSAPI). Each MS can have up to 15 NSAPIs. This allows an MS to create multiple PDP contexts with different NSAPIs, based on the application requirements for various quality of service (QoS) levels. The TID is carried in the GTPv0 header.

An IMSI has the following three parts:

- Mobile Country Code (MCC) that consists of three digits. The MCC uniquely identifies the country of domicile of a mobile subscriber.
- Mobile Network Code (MNC) that consists of two or three digits for GSM applications. The MNC identifies the home GSM Public Land Mobile Network (PLMN) of the mobile subscriber. The length of the MNC depends on the value of the MCC.

  Note A combination of two- and three-digit MNC codes within a single MCC area is not recommended.

- Mobile Subscriber Identification Number (MSIN) that identifies a mobile subscriber within a GSM PLMN. The National Mobile Subscriber Identity (NMSI) consists of the MNC and the MSIN.

GTPv1 introduces the concept of primary and secondary contexts for an MS. A primary context is associated with an IP address and indicates other parameters like the Access Point Name (APN) to be attached to the receiving GSN. Secondary contexts created for this primary PDP context share the IP address and other parameters that are already associated with the primary context. This allows an MS to initiate another context with a different quality of service (QoS) requirement and also share the IP address already obtained for the primary context. Primary and secondary contexts share the Tunnel Endpoint ID (TEID) on the control plane and have different TEID values in the data plane. Since all primary and associated secondary contexts share the IP address, Traffic Flow Templates (TFT) are used to classify traffic in the downlink direction towards the MS. TFTs are exchanged during context creation.

Only the create PDP context request for the primary PDP contains an IMSI. The IMSI and NSAPI together uniquely identify a PDP context. A secondary PDP context activation contains a Linked NSAPI (LNSAPI) indicating the NSAPI that is assigned to any one of the already activated PDP contexts for this PDP address and APN.

Note UDP is the only supported, defined path protocol for signaling messages for GTPv0 and GTPv1.

GGSN Pooling Support

A Gateway GPRS Support Node (GGSN) supports firewall load balancing by using the Server Load Balancing (SLB) feature. SLB balances traffic flows across a group of firewalls called a firewall farm. In this cluster or pool, clients can connect to the IP address of a virtual server. When a client initiates a connection to the virtual server, SLB chooses a real server for the connection, based on a configured load-balancing algorithm.

While configuring GTP load balancing, a pool of GGSNs is configured as a GGSN farm in SLB. You can use these GGSNs to load balance GPRS Tunneling Protocol (GTP) sessions. A virtual server instance is configured in the SLB to load balance GTP sessions across the GGSN farm.

To support GGSN pooling, a device must allow load balancing GSNs to respond to Packet Data Protocol (PDP) requests, even if the GSN is different from the one specified as SLB IP address in the GTP packet.

In GGSN pooling, when a subscriber who is using a roaming connection sends a PDP request from a Serving GPRS Support Node (SGSN) to the GGSN that resides behind the SLB, the firewall should accept the PDP request. As pinholes are not created for unknown GGSNs, the firewall drops PDP responses. To avoid PDP responses from getting dropped by the firewall, you need to configure an access control list (ACL). A firewall pinhole is a port that is opened through the firewall to allow an application to gain controlled access to a protected network.

A global session database records all pending PDP request contexts when a PDP request is received from an SGSN. When a PDP request is received from an SGSN, a session lookup is performed to match the context, and the response is dropped if no match is found. Packet data is transferred by establishing a PDP context, which is effectively a data session.

GTP Traffic Through Firewall

The main General Packet Radio Service (GPRS) Tunneling Protocol (GTP) traffic that a device inspects is the roaming traffic. Roaming traffic is caused when a Mobile Station (MS) moves from its Home Public Land Mobile Network (HPLMN) to a Visited PLMN (VPLMN).

The GTP traffic through the firewall includes the following messages:

- Serving GPRS Support Node (SGSN) to Gateway GPRS Support Node (GGSN) GTP messages
- GGSN-to-SGSN GTP messages
- SGSN-to-SGSN GTP messages

How to Configure GGSN Pooling Support for Firewalls

Configuring Access Control Lists and Class Maps for GGSN Pooling

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **access-list** *access-list-number* **permit** *protocol source source-wildcard* **any**
4. **access-list** *access-list-number* **permit** *protocol* **any** *destination destination-wildcard*

5. **access-list** *access-list-number* **permit** *protocol source source-wildcard* **any**
6. **class-map type inspect gtpv1 match-any** *class-map-name*
7. **match mcc** *country-code* **mnc** *network-code*
8. **match mcc** *country-code* **mnc** *network-code*
9. **exit**
10. **class-map type inspect gtpv1 match-any** *class-map-name*
11. **match mcc** *country-code* **mnc** *network-code*
12. **match mcc** *country-code* **mnc** *network-code*
13. **exit**
14. **class-map type inspect gtpv1 match-all** *class-map-name*
15. **match protocol** *protocol-name*
16. **match access-group** *access-list-number*
17. **exit**
18. **class-map type inspect gtpv1 match-all** *class-map-name*
19. **match protocol** *protocol-name*
20. **match access-group** *access-list-number*
21. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**access-list** *access-list-number* **permit** *protocol source source-wildcard* **any** **Example:** `Device(config)# access-list 101 permit ip 10.2.2.0 255.255.255.0 any`	Defines an extended IP access list. • Access list 101 that is configured in this example permits traffic from a GGSN or SGSN to any destination.
Step 4	**access-list** *access-list-number* **permit** *protocol* **any** *destination destination-wildcard* **Example:** `Device(config)# access-list 102 permit ip any 10.2.2.0 255.255.255.0`	Defines an extended IP access list. • Access list 102 that is configured in this example permits traffic from any source to a GGSN or SGSN.
Step 5	**access-list** *access-list-number* **permit** *protocol source source-wildcard* **any** **Example:** `Device(config)# access-list 103 permit ip 10.2.2.0 255.255.255.0 any`	Defines an extended IP access list. • Access list 103 that is configured in this example permits traffic from a GGSN or SGSN to any destination.

	Command or Action	Purpose
Step 6	**class-map type inspect gtpv1 match-any** *class-map-name* **Example:** `Device(config)# class-map type inspect gtpv1 match-any gtp-cl7-rev`	Creates an application-specific inspect type class map and specifies that packets must meet any one of the specified match criteria to be considered a member of the class, and enters QoS class-map configuration mode.
Step 7	**match mcc** *country-code* **mnc** *network-code* **Example:** `Device(config-cmap)# match mcc 1 mnc 1`	Configures filtering for a valid Mobile Country Code (MCC) and a Mobile Network Code (MNC). • In this example you configure filtering of roaming connections to a foreign MCC and MNC.
Step 8	**match mcc** *country-code* **mnc** *network-code* **Example:** `Device(config-cmap)# match mcc 2 mnc 1`	Configures filtering for a valid MCC and an MNC. • In this example you configure filtering of roaming connections to a local MCC and MNC.
Step 9	**exit** **Example:** `Device(config-cmap)# exit`	Exits QoS class-map configuration mode and enters global configuration mode.
Step 10	**class-map type inspect gtpv1 match-any** *class-map-name* **Example:** `Device(config)# class-map type inspect gtpv1 match-any gtp-cl7`	Creates an application-specific inspect type class map and specifies that packets must meet any one of the specified match criteria to be considered a member of the class, and enters QoS class-map configuration mode.
Step 11	**match mcc** *country-code* **mnc** *network-code* **Example:** `Device(config-cmap)# match mcc 2 mnc 1`	Configures filtering for a valid MCC and an MNC.
Step 12	**match mcc** *country-code* **mnc** *network-code* **Example:** `Device(config-cmap)# match mcc 1 mnc 1`	Configures filtering for a valid MCC and an MNC.
Step 13	**exit** **Example:** `Device(config-cmap)# exit`	Exits QoS class-map configuration mode and enters global configuration mode.
Step 14	**class-map type inspect gtpv1 match-all** *class-map-name* **Example:** `Device(config)# class-map type inspect gtpv1 match-all gtp-l4c`	Creates an application-specific inspect type class map and specifies that packets must meet all specified match criteria to be considered a member of the class, and enters QoS class-map configuration mode.
Step 15	**match protocol** *protocol-name* **Example:** `Device(config-cmap)# match protocol gtpv1`	Configures a match criterion for a class map on the basis of the specified protocol.

	Command or Action	Purpose
Step 16	**match access-group** *access-list-number* **Example:** `Device(config-cmap)# match access-group 101`	Configures a match criterion for a class map on the basis of the specified ACL.
Step 17	**exit** **Example:** `Device(config-cmap)# exit`	Exits QoS class-map configuration mode and enters global configuration mode.
Step 18	**class-map type inspect gtpv1 match-all** *class-map-name* **Example:** `Device(config)# class-map type inspect gtpv1 match-all gtp-l4c-rev`	Creates an application-specific inspect type class map and specifies that packets must meet all specified match criteria to be considered a member of the class, and enters QoS class-map configuration mode.
Step 19	**match protocol** *protocol-name* **Example:** `Device(config-cmap)# match protocol gtpv1`	Configures a match criterion for a class map on the basis of the specified protocol.
Step 20	**match access-group** *access-list-number* **Example:** `Device(config-cmap)# match access-group 102`	Configures a match criterion for a class map on the basis of the specified ACL.
Step 21	**end** **Example:** `Device(config-cmap)# end`	Exits QoS class-map configuration mode and enters privileged EXEC mode.

Configuring Policy Maps for GGSN Pooling

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map type inspect gtpv1** *gtpv1-policy*
4. **class type inspect gtpv1** *class-map-name*
5. **log**
6. **exit**
7. **class class-default**
8. **exit**
9. **policy-map type inspect gtpv1** *gtpv1-policy*
10. **class type inspect gtpv1** *class-map-name*
11. **log**
12. **exit**
13. **class class-default**
14. **exit**

15. **policy-map type inspect gtpv1** *gtpv1-policy*
16. **class type inspect gtpv1** *class-map-name*
17. **inspect**
18. **service-policy** *policy-map-name*
19. **exit**
20. **class class-default**
21. **exit**
22. **policy-map type inspect gtpv1** *gtpv1-policy*
23. **class type inspect gtpv1** *class-map-name*
24. **inspect**
25. **service-policy** *policy-map-name*
26. **exit**
27. **class class-default**
28. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**policy-map type inspect gtpv1** *gtpv1-policy* **Example:** `Device(config)# policy-map type inspect gtpv1 gtp-l7p-rev`	Creates a protocol-specific inspect-type policy map and enters QoS policy-map configuration mode.
Step 4	**class type inspect gtpv1** *class-map-name* **Example:** `Device(config-pmap)# class type inspect gtpv1 gtp-cl7-rev`	Specifies a traffic class on which an action is to be performed and enters QoS policy-map class configuration mode.
Step 5	**log** **Example:** `Device(config-pmap-c)# log`	Generates a log of messages.
Step 6	**exit** **Example:** `Device(config-pmap-c)# exit`	Exits QoS policy-map class configuration mode and enters QoS policy-map configuration mode.
Step 7	**class class-default** **Example:**	Specifies the default class so that you can configure or modify its policy.

	Command or Action	Purpose
	`Device(config-pmap)# class class-default`	
Step 8	**exit** **Example:** `Device(config-pmap)# exit`	Exits QoS policy-map configuration mode and enters global configuration mode.
Step 9	**policy-map type inspect gtpv1** *gtpv1-policy* **Example:** `Device(config)# policy-map type inspect gtpv1 gtp-l7p`	Creates a protocol-specific inspect-type policy map and enters QoS policy-map configuration mode.
Step 10	**class type inspect gtpv1** *class-map-name* **Example:** `Device(config-pmap)# class type inspect gtpv1 gtp-cl7`	Specifies a traffic class on which an action is to be performed and enters QoS policy-map class configuration mode.
Step 11	**log** **Example:** `Device(config-pmap-c)# log`	Generates a log of messages.
Step 12	**exit** **Example:** `Device(config-pmap-c)# exit`	Exits QoS policy-map class configuration mode and enters QoS policy-map configuration mode.
Step 13	**class class-default** **Example:** `Device(config-pmap)# class class-default`	Specifies the default class so that you can configure or modify its policy.
Step 14	**exit** **Example:** `Device(config-pmap)# exit`	Exits QoS policy-map configuration mode and enters global configuration mode.
Step 15	**policy-map type inspect gtpv1** *gtpv1-policy* **Example:** `Device(config)# policy-map type inspect gtpv1 gtp-l4p-rev`	Creates a protocol-specific inspect-type policy map and enters QoS policy-map configuration mode.
Step 16	**class type inspect gtpv1** *class-map-name* **Example:** `Device(config-pmap)# class type inspect gtpv1 gtp-l4c-rev`	Specifies a traffic class on which an action is to be performed and enters QoS policy-map class configuration mode.
Step 17	**inspect** **Example:** `Device(config-pmap-c)# inspect`	Enables stateful packet inspection.

	Command or Action	Purpose
Step 18	**service-policy** *policy-map-name* **Example:** `Device(config-pmap-c)# service-policy gtp-l7p-rev`	Uses a service policy as a QoS policy within a policy map (called a hierarchical service policy).
Step 19	**exit** **Example:** `Device(config-pmap-c)# exit`	Exits QoS policy-map class configuration mode and enters QoS policy-map configuration mode.
Step 20	**class class-default** **Example:** `Device(config-pmap)# class class-default`	Specifies the default class so that you can configure or modify its policy.
Step 21	**exit** **Example:** `Device(config-pmap)# exit`	Exits QoS policy-map configuration mode and enters global configuration mode.
Step 22	**policy-map type inspect gtpv1** *gtpv1-policy* **Example:** `Device(config)# policy-map type inspect gtpv1 gtp-l4p`	Creates a protocol-specific inspect-type policy map and enters QoS policy-map configuration mode.
Step 23	**class type inspect gtpv1** *class-map-name* **Example:** `Device(config-pmap)# class type inspect gtpv1 gtp-l4c`	Specifies a traffic class on which an action is to be performed and enters QoS policy-map class configuration mode.
Step 24	**inspect** **Example:** `Device(config-pmap-c)# inspect`	Enables stateful packet inspection.
Step 25	**service-policy** *policy-map-name* **Example:** `Device(config-pmap-c)# service-policy gtp-l7p`	Uses a service policy as a QoS policy within a policy map (called a hierarchical service policy).
Step 26	**exit** **Example:** `Device(config-pmap)# exit`	Exits QoS policy-map class configuration mode and enters QoS policy-map configuration mode.
Step 27	**class class-default** **Example:** `Device(config-pmap)# class class-default`	Specifies the default class so that you can configure or modify its policy.
Step 28	**end** **Example:** `Device(config-pmap-c)# end`	Exit QoS policy-map class configuration mode and enters privileged EXEC mode.

Configuring Zones and Zone Pairs for GGSN Pooling Support

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **zone security** *security-zone*
4. **exit**
5. **zone security** *security-zone*
6. **exit**
7. **zone-pair security** *zone-pair-name* **source** *source-zone* **destination** *destination-zone*
8. **service-policy type inspect** *policy-map-name*
9. **exit**
10. **zone-pair security** *zone-pair-name* **source** *source-zone* **destination** *destination-zone*
11. **service-policy type inspect** *policy-map-name*
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device(config)# configure terminal`	Enters global configuration mode.
Step 3	**zone security** *security-zone* **Example:** `Device(config)# zone security roam-in`	Creates a security zone to which interfaces can be assigned and enters security zone configuration mode. • Your configuration must have two security zones to create a zone pair: a source and a destination zone. • In a zone pair, you can use the default zone as either the source or the destination zone.
Step 4	**exit** **Example:** `Device(config-sec-zone)# exit`	Exits security zone configuration mode and enters global configuration mode.
Step 5	**zone security** *security-zone* **Example:** `Device(config-sec-zone)# zone security roam-out`	Creates a security zone to which interfaces can be assigned and enters security zone configuration mode.
Step 6	**exit** **Example:**	Exits security zone configuration mode and enters global configuration mode.

	Command or Action	Purpose
	`Device(config-sec-zone)# exit`	
Step 7	**zone-pair security** *zone-pair-name* **source** *source-zone* **destination** *destination-zone* **Example:** `Device(config)# zone-pair security in2out source roam-in destination roam-out`	Creates a pair of security zones and enters security zone-pair configuration mode. • To apply a policy, you must configure a zone pair.
Step 8	**service-policy type inspect** *policy-map-name* **Example:** `Device(config-sec-zone-pair)# service-policy type inspect gtp-l4p`	Attaches a firewall policy map to the destination zone pair.
Step 9	**exit** **Example:** `Device(config-sec-zone-pair)# exit`	Exits security zone-pair configuration mode and enters global configuration mode.
Step 10	**zone-pair security** *zone-pair-name* **source** *source-zone* **destination** *destination-zone* **Example:** `Device(config)# zone-pair security out2in source roam-out destination roam-in`	Creates a pair of security zones and enters security zone-pair configuration mode.
Step 11	**service-policy type inspect** *policy-map-name* **Example:** `Device(config-sec-zone-pair)# service-policy type inspect gtp-l4p-rev`	Attaches a firewall policy map to the destination zone pair.
Step 12	**end** **Example:** `Device(config-sec-zone)# end`	Exits security zone-pair configuration mode and enters privileged EXEC mode.

Configuration Examples for GGSN Pooling Support for Firewalls

Example: Configuring Access Control Lists and Class Maps for GGSN Pooling

```
Device# configure terminal
Device(config)# access-list 101 permit ip 10.2.2.0 255.255.255.0 any
Device(config)# access-list 102 permit ip any 10.2.2.0 255.255.255.0
Device(config)# access-list 103 permit ip 10.2.2.0 255.255.255.0 any
Device(config)# class-map type inspect gtpv1 match-any gtp-cl7-rev
Device(config-cmap)# match mcc 1 mnc 1
Device(config-cmap)# match mcc 2 mnc 1
Device(config-cmap)# exit
Device(config)# class-map type inspect gtpv1 match-any gtp-cl7
Device(config-cmap)# match mcc 2 mnc 1
```

```
Device(config-cmap)# match mcc 1 mnc 1
Device(config-cmap)# exit
Device(config)# class-map type inspect gtpv1 match-all gtp-l4c
Device(config-cmap)# match protocol gtpv1
Device(config-cmap)# match access-group 101
Device(config-cmap)# exit
Device(config)# class-map type inspect gtpv1 match-all gtp-l4c-rev
Device(config-cmap)# match protocol gtpv1
Device(config-cmap)# match access-group 102
Device(config-cmap)# end
```

Example: Configuring Policy Maps for GGSN Pooling

```
Device# configure terminal
Device(config)# policy-map type inspect gtpv1 gtp-l7p-rev
Device(config-pmap)# class type inspect gtpv1 gtp-cl7-rev
Device(config-pmap-c)# log
Device(config-pmap-c)# exit
Device(config-pmap)# class class-default
Device(config-pmap)# exit
Device(config)# policy-map type inspect gtpv1 gtp-l7p
Device(config-pmap)# class type inspect gtpv1 gtp-cl7
Device(config-pmap-c)# log
Device(config-pmap-c)# exit
Device(config-pmap)# class class-default
Device(config-pmap)# exit
Device(config)# policy-map type inspect gtpv1 gtp-l4p-rev
Device(config-pmap)# class type inspect gtpv1 gtp-l4c-rev
Device(config-pmap-c)# inspect
Device(config-pmap-c)# service-policy gtp-l7p-rev
Device(config-pmap-c)# exit
Device(config-pmap)# class class-default
Device(config-pmap)# exit
Device(config)# policy-map type inspect gtpv1 gtp-l4p
Device(config-pmap)# class type inspect gtpv1 gtp-l4c
Device(config-pmap-c)# inspect
Device(config-pmap-c)# service-policy gtp-l7p
Device(config-pmap)# exit
Device(config-pmap)# class class-default
Device(config-pmap-c)# end
```

Example: Configuring Zones and Zone Pairs for GGSN Pooling

```
Device(config)# configure terminal
Device(config)# zone security roam-in
Device(config-sec-zone)# exit
Device(config-sec-zone)# zone security roam-out
Device(config-sec-zone)# exit
Device(config)# zone-pair security in2out source roam-in destination roam-out
Device(config-sec-zone-pair)# service-policy type inspect gtp-l4p
Device(config-sec-zone-pair)# exit
Device(config)# zone-pair security out2in source roam-out destination roam-in
Device(config-sec-zone-pair)# service-policy type inspect gtp-l4p-rev
Device(config)# end
```

Additional References for Firewall Stateful Interchassis Redundancy

Related Documents

Related Topic	Document Title
Cisco IOS commands	Master Command List, All Releases
Security commands	• Security Command Reference: Commands A to C • Security Command Reference: Commands D to L • Security Command Reference: Commands M to R • Security Command Reference: Commands S to Z

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for GGSN Pooling Support for Firewalls

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1: Feature Information for GGSN Pooling Support for Firewalls

Feature Name	Releases	Feature Information
GGSN Pooling Support for Firewalls	Cisco IOS XE Release 3.7S	The GGSN Pooling Support for Firewalls feature enhances the GPRS Tunneling Protocol (GTP) feature by adding load balancing support. GTP supports the inspection of control traffic that is designated to a single GGSN. To provide efficiency and scalability to GSM networks, load balancing is added to the topology. The load balancer dispatches requests from the SGSN to various GGSNs in the pool.