Oracle® Communications Network Charging and Control

Messaging Firewall Technical Guide

Release 12.0.0

December 2017

ORACLE®

Copyright

Contents

About This Document

Scope

The scope of this document includes all the information required to administer the Messaging Firewall application.

Audience

This guide is written primarily for system administrators and other personnel who administer the Messaging Firewall application. However, the overview sections of the document are useful to anyone requiring an introduction to the application.

Prerequisites

Although it is not a prerequisite to using this guide, familiarity with the target platform would be an advantage.

A solid understanding of Unix and a familiarity with IN concepts are an essential prerequisite for safely using the information contained in this technical guide. Attempting to install, remove, configure or otherwise alter the described system without the appropriate background skills, could cause damage to the system; including temporary or permanent incorrect operation, loss of service, and may render your system beyond recovery.

This manual describes system tasks that should only be carried out by suitably trained operators.

Related Documents

The following documents are related to this document:

- *ACS User's Guide*
- *ACS Technical Guide*
- *MM Technical Guide*

Document Conventions

Typographical Conventions

The following terms and typographical conventions are used in the Oracle Communications Network Charging and Control (NCC) documentation.

Formatting Convention	Type of Information
Special Bold	Items you must select, such as names of tabs. Names of database tables and fields.
Italics	Name of a document, chapter, topic or other publication. Emphasis within text.
Button	The name of a button to click or a key to press. **Example:** To close the window, either click **Close**, or press **Esc**.
Key+Key	Key combinations for which the user must press and hold down one key and then press another. Example: **Ctrl+P** or **Alt+F4**.
`Monospace`	Examples of code or standard output.
`Monospace Bold`	Text that you must enter.
variable	Used to indicate variables or text that should be replaced with an actual value.
menu option > menu option >	Used to indicate the cascading menu option to be selected. Example: **Operator Functions > Report Functions**
hypertext link	Used to indicate a hypertext link.

Specialized terms and acronyms are defined in the glossary at the end of this guide.

Chapter 1

System Overview

Overview

Introduction

This chapter provides a high-level overview of the application. It explains the basic functionality of the system and lists the main components.

It is not intended to advise on any specific Oracle Communications Network Charging and Control (NCC) network or service implications of the product.

In this Chapter

This chapter contains the following topics.

What is Messaging Firewall?

Introduction

Messaging Firewall is a software-based IN solution that checks and handles native protocol messages within the defined protocol message flows. Messaging Firewall is protocol-aware, and checks and validates data to determine if the sender of a message is trusted or not.

Messaging Firewall performs the following functions, which are based on Visitor Location Register (VLR):

- VLR Steering
- VLR Velocity
- VLR Validation

About VLR Steering

VLR Steering prevents a subscriber's mobile device from connecting to a roaming operator who is not in the preferred network. However, you can enable the connection to a non-preferred network by configuring the number of times to attempt a connection before determining that the preferred network is not available.

Messaging Firewall performs VLR steering only when an UpdateLocation message is received.

VLR Steering performs the following tasks:

1. Decodes the msisdnReference from mapOpen.
2. If the configuration item no_of_vlr_steers is equal to 0, performs VLR Velocity.
3. If msisdnReference is not in cache, adds msisdnReference to the cache and tests the value of the no_of_vlr_steers configuration item.
4. If no_of_vlr_steers is set to a value greater than 0, increments the cache entry noOfSteers and rejects the UpdateLocation message.

5 If msisdnReference is in the local cache, tests the cache entry noOfSteers against the no_of_vlr_steer configuration item.
6 If noOfSteers is greater than the no_of_vlr_steer configuration item, allows the message to move on to the VLR Velocity check. Otherwise, it increments the noOfSteers entry value and rejects the message. The cache flushes the entry after Time to live (TTL) expires.

About VLR Velocity

VLR Velocity enables you to update the location that is geographically possible in the time since the last update. VLR Velocity compares the location of the message's global title calling party country code with the VMSC country code that is returned from the HLR. The HLR lookup is populated by the location message's IMSI.

Messaging Firewall performs VLR Velocity only when an UpdateLocation message is received.

VLR Velocity performs the following tasks:

1 Checks if msisdnReference is in the country code prefix.
2 Performs a RIMS lookup to retrieve the vmsc.
3 Checks if the vmsc is in the visiting country code prefix.
4 Sends UpdateLocation back on the network.

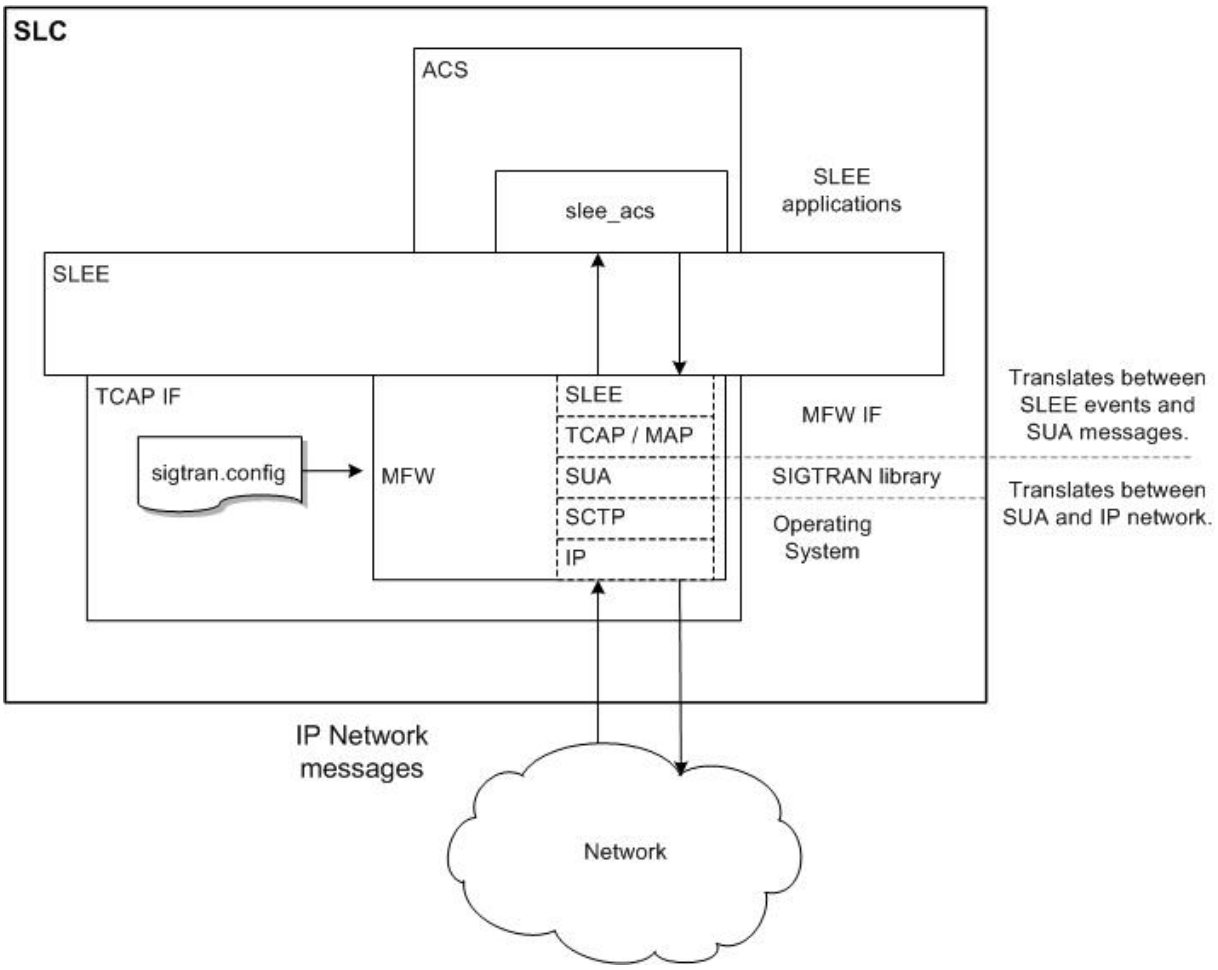
About VLR Validation

VLR Validation confirms if a record in the VLR is available in the HLR. If the record is not available in the HLR, VLR Validation rejects the MAP message. Messaging Firewall performs VLR Validation for all message types, except for the UpdateLocation message.

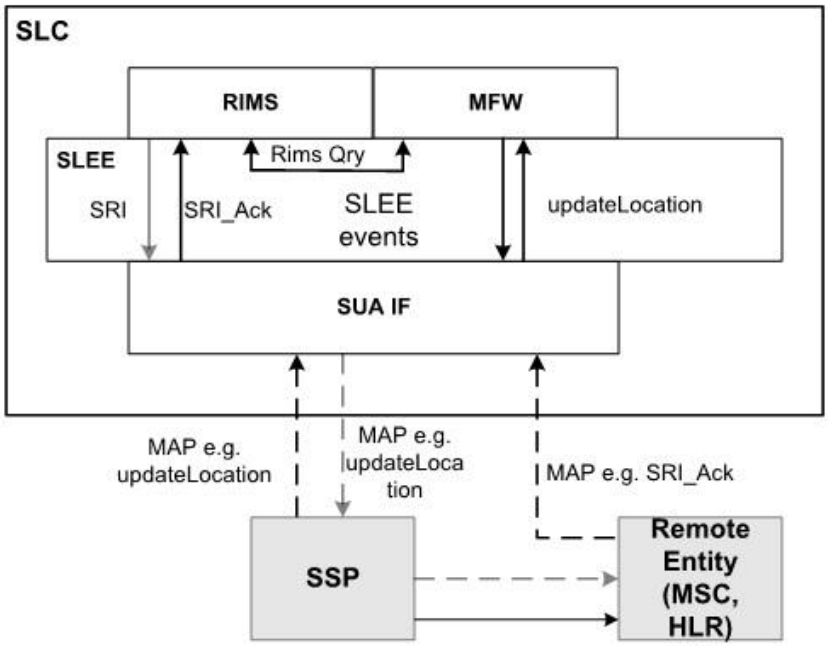
VLR Validation performs the following actions:

1 Decodes msisdnReference from mapOpen or from the actual message if no mapOpen is present.
2 Checks if msisdnReference is in the validation prefix code.
3 Performs a RIMS lookup to retrieve the vmsc.
4 If the RIMS vmsc and the MSISDN match, sends the message back on the network. If the RIMS vmsc and the MSISDN do not match, compares the MSISDN with "REJECT fail code". If a match is made, sends the REJECT message back on the network.
5 If no match is made with the "REJECT fail code" list, compares the MSISDN with the "DISCARD fail code". If a match is made, closes the dialog and returns an error message.
6 If no match is made with the "REJECT fail code" list or the "DISCARD fail code" list, closes the dialog and accepts the message.

The following diagram shows the SCCP SUA interface:



The following diagram illustrates the SLEE correlation ID:



Chapter 2

Configuration

Overview

Introduction

This chapter explains how to configure Messaging Firewall for the Oracle Communications Network Charging and Control (NCC) application.

In this chapter

This chapter contains the following topics.

Configuration Overview

Introduction

There are configuration options that are added to the configuration files that are not explained in this chapter. These configuration options are required by the system and should not be changed.

Configuration Components

The following components form the configuration components for Messaging Firewall:

Component	Location	Description	Further Information
mfw.cfg	SLC	Configures the Messaging Firewall application.	*mfw* (on page 7)
SLEE.cfg	SLC	The SLEE _ac interface is configured to include the Messaging Firewall application, the MTA and the cdrIF interface.	SLEE.cfg Configuration and *SLEE Technical Guide.*

Configuration Files Location

The Messaging Firewall configuration files are located in the **/IN/service_packages/MFW/etc** directory.

Editing Configuration Files

To ensure that you have a working copy, before making any changes to the configuration files, backup your current configuration.

To edit configuration files, open the configuration file using a standard file editor. Do not use file editors such as Microsoft Word that attach Microsoft DOS or Windows line termination characters (for example, ^M) at the end of each row, because this causes file errors when the application tries to read the configuration file.

Loading Configuration Changes

If you change a configuration file, then you must send a signal (SIGHUP) to the relevant process, or restart the SLEE to enable the new options to take effect.

SLEE.cfg Configuration

Introduction

You must configure the **SLEE.cfg** file to enable Messaging Firewall to work.

During the installation of the Messaging Firewall application, the **SLEE.cfg** file is updated automatically to set up the platform to use MFW.

For more information, see:

- *SLEE Technical Guide* for information on SLEE configuration.
- *XML TCAP Interface Technical Guide* for information on SERVICEKEY format.

MFW SLEE Configuration

After the installation of the Messaging Firewall application, verify that the following lines have been added to the **SLEE.cfg** file:

```
APPLICATION=mfwApplication mfw.sh /IN/service_packages/MFW/bin 1 1 # mfw
SERVICEKEY=INTEGER 0x1080000000a MFW #   (GSM RegsiterSS)
SERVICEKEY=INTEGER 0x1080000000b MFW #   (GSM EraseSS)
SERVICEKEY=INTEGER 0x1080000000c MFW #   (GSM ActivateSS)
SERVICEKEY=INTEGER 0x1080000000d MFW #   (GSM DeactivateSS)
SERVICEKEY=INTEGER 0x1080000000e MFW #   (GSM InterrogateSS)
SERVICEKEY=INTEGER 0x10800000011 MFW #   (GSM RegisterPassword)
SERVICEKEY=INTEGER 0x10800000002 MFW #   (GSM UpdateLocation)
SERVICEKEY=INTEGER 0x10800000004 MFW #   (GSM ProvideRoamingNumber)
SERVICEKEY=INTEGER 0x10800000043 MFW #   (GSM purgeMS)
SERVICEKEY=INTEGER 0x10800000039 MFW #   (GSM RestoreData)
SERVICEKEY=INTEGER 0x10800000036 MFW #   (GSM BeginSubscriberActivity)
SERVICEKEY=INTEGER 0x10800000013 MFW #   (GSM ProcessUSSData)
SERVICEKEY=INTEGER 0x10A0000002D MFW #   (GSM SRI_SM)
SERVICEKEY=INTEGER 0x10A00000016 MFW #   (GSM SRI)
SERVICEKEY=INTEGER 0x10A0000002E MFW #   (GSM MO SM)
SERVICEKEY=INTEGER 0x10A0000002C MFW #   (GSM MT SM)
SERVICE=MFW 1 mfwApplication MFW
```

During the installation, SERVICEKEY values are generated from the subsystem number and the specified base key and therefore may be different from those shown here.

Chapter 3

Background Processes

Overview

Introduction

This chapter explains the processes that are started automatically by Service Logic Execution Environment (SLEE).

Note: This chapter also includes some plug-ins to background processes which do not run independently.

In this chapter

This chapter contains the following topics.

mfw

Purpose

Messaging Firewall receives a MAP message from the TCAP Interface and performs VLR validation, VLR steering, or VLR roaming depending on the message type. An appropriate alarm is raised if there is a problem processing the message.

Startup

mfw is started automatically by SLEE. The following lines must be included in **SLEE.cfg** to start mfw:

```
APPLICATION=mfwApplication mfw.sh /IN/service_packages/MFW/bin 1 1
SERVICEKEY=INTEGER 0x10800000002 MFW
SERVICEKEY=INTEGER 0x10800000006 MFW
SERVICE=MFW 1 mfwApplication MFW
```

Note: The service key details are generated automatically during installation and therefore may be different from this example.

For more information, see SLEE.cfg Configuration.

Location

mfw is located on the SLC.

mfw.cfg configuration file

The configuration file for mfw is **mfw.cfg**.

The following text shows the default configuration for Messaging Firewall in the **mfw.cfg** file:

```
mfw = {

  # Cache tuning options
```

```
  cache = {
     # Maximum number of HLR responses to cache at any one time
     # default size = 1000000

     # Time to live
     # default = 3600 seconds
     # ttl = 3600
  }

  # OPTIONAL
  # VLR Steering - number of times to Roam/Steer UpdateLocation
  # messages, before accepting the message.
  # default is 3
  # no_of_vlr_steers = 3


  # OPTIONAL
  # VLR Velocity Visiting Country Code
  # The Visiting Country code and the minimum time allowed to travel there.
  # AF (93), SV (503)
  #
  # Example :
  # VisitingCCodes = [
  #                    { code="93"  }
  #                    { code="503" }
  #                 ]
  #
  VisitingCCodes = [
                    { code="93"  }
                    { code="503"  }
                  ]

  # OPTIONAL
  # VLR Velocity Country Codes
  # ie UK (44), FR (33)
  #
  # Where the VLR Velocity Visiting Country code prefixes
  # the VLR Velocity Country Codes.
  #
  # Example :
  #                     UK --> AF&SV
  # CountryCodes = [
  #                   { min_allowed_time="0030", visiting_and_country_code="9344"
} ie AFUK
  #                   { min_allowed_time="0120", visiting_and_country_code="50344"
} ie SVUK
  #
  #                    FR --> AF&SV
  #                   { min_allowed_time="0030", visiting_and_country_code="9333"
} ie AFFR
  #                   { min_allowed_time="0120", visiting_and_country_code="50333"
} ie SVFR
  #                 ]
  #
  CountryCodes = [
                    { min_allowed_time="0030", visiting_and_country_code="9344"  }
                    { min_allowed_time="0120", visiting_and_country_code="50344"
}

                    { min_allowed_time="0030", visiting_and_country_code="9333"  }
                    { min_allowed_time="0120", visiting_and_country_code="50333"
}
                  ]
  # OPTIONAL
```

```
# VLR Validation Prefix Codes
# Validates the initial message prefix
# e.g ActivateSS, DeactivateSS etc Global Title
# calling party number prefix.
#
# Example :
# ValidationPrefixCodes = [ { validation_prefix_code="441273"  }
#                           { validation_prefix_code="2711473" }
#                         ]
#
ValidationPrefixCodes = [ { validation_prefix_code="441273"  }
                          { validation_prefix_code="2711473"  }
                        ]

# OPTIONAL
# VLR Validation Reject Fail Codes
# If the VLR validation fails and the gt cgpn prefix
# is found within this prefix list, and having failed
# the VLR Validation check the resultant message sent
# will be a TCAP abort
#
# Example :
# RejectFailCodes = [ { reject_fail_code="441273"  }
#                     { reject_fail_code="2711473" }
#                   ]
#
RejectFailCodes = [ { reject_fail_code="441273"  }
                    { reject_fail_code="2711473"  }
                  ]

# OPTIONAL
# VLR Validation Discard Fail Codes
# If the VLR validation fails and the gt cgpn prefix
# is found within this prefix list, and having failed
# the VLR Validation check and the "Reject fail prefix" check
# the mfw program will generate a system error message.
#
# Note if the "Reject Fail prefix" check and the
# "Discard fail prefix" return no results then the
# message will be Accepted silently but not relayed
# back to the STP.
#
# Example :
# DiscardFailCodes = [ { discard_fail_code="441273"  }
#                      { discard_fail_code="2711473" }
#                    ]
#
DiscardFailCodes = [ { discard_fail_code="441273"  }
                     { discard_fail_code="2711473"  }
                   ]

# OPTIONAL
# rims_if
# The Rims interface handle
# rims_if = "rimsIf"

# OPTIONAL
# vmsc_if
# The VMSC interface handle
# vmsc_if = "vmscIf"

# OPTIONAL
# Specify the Point-Code standard used in configuration (ITU-TS or ANSI).
```

```
  # The default will be ITU-TS if not defined.
  # ITU-TS point codes will be encoded as a 14-bit structure consisting of
  # a 3-bit zone identification, and 8-bit area/network identification and
  # a 3-bit signaling point identification. ANSI point codes will be encoded
  # in a 24-bit structure consisting of an 8-bit network identification, an
  # 8-bit cluster identification and an 8-bit member identification.
  # Allowed values: "ITU-TS","ANSI"
  #
  # Example:
  # pc_format = "ITU-TS"

  # MANDATORY Not currently used.
  # Maps the SCCP calling party number in the original MAP message (if present)
  # to an IMSI value that will be present in any MAP responses sent by the MFW.
  #
  # prefix (MANDATORY)
  # The rule containing the longest prefix match will be fired. Prefix matching
is
  # performed on the SCCP calling party number in the original MAP message (if
present).
  # A default prefix of "*" will match any SCCP CgPN and will also match a
message
  # which does not have an SCCP CgPN defined. Only one default prefix can be
defined.
  # Example:
  # prefix = "44"
  #
  # imsi (MANDATORY)
  # The imsi value to use in any generated MAP responses.
  # Example:
  # imsi = "441473123456"
  #map_response_imsi = [ { prefix="44", imsi="441473123456" }
  #                      { prefix="*", imsi="441473000000" }
  #                    ]

  # OPTIONAL
  # Maps the abort code number returned by the Map Trigger Node to a
  # alarm message and severity.
  #
  # code (OPTIONAL)
  # The alarm code to match. If the code is not defined the rule will match any
alarm if
  # a match is not found in other rules.
  # Example:
  # code = 10
  #
  # message (MANDATORY)
  # The message to print in the alarm
  # Example:
  # message = "An Error Occurred"
  #
  # severity (MANDATORY)
  # The severity of the Alarm. Allowed values (in order of increasing severity)
are:
  # "NOTICE", "WARNING", "ERROR", "CRITICAL"
  #
  # abort_code_mapping = [  { code = 10, message = "A Critical Error Occurred",
severity = "CRITICAL" }
  #                         { code = 11, message = "An error occurred", severity
= "ERROR" }
  #                         { message = "A minor error occurred", severity =
"WARNING" }
  #                       ]
```

```
        # OPTIONAL
        # Set the default MAP version (1, 2 or 3). This is only relevant
        # if messages are received without an application context.
        map_version = 1
```

}

Parameters

mfw accepts the following parameters:

abort_code_mapping

Syntax: `abort_code_mapping = [ { code = "alarm_code", message = "alarm_message", severity = "severity_level" } ]`

Description: List of abort code number mappings (returned by the Map Trigger Node) to alarm messages and severity levels.

Type: Parameter array.

Optionality: Optional.

Allowed:

Default:

Notes:

- `code` - The alarm code to match. If the code is not defined, then the rule will match any alarm. A range of values can also be specified.
- `message` (mandatory) - The message to print in the alarm.
- `severity` (mandatory) - The severity of the alarm. Valid values are: NOTICE, WARNING, ERROR, and CRITICAL.

Example:

```
abort_code_mapping = [
{ code = "0-10", message = "A critical error occurred",
severity = "CRITICAL" }
]
```

VisitingCCodes

Syntax: `VisitingCCodes = The number of the visiting country code.`

Description: When processing a MAP message that requires Velocity checking, the prefix of the MSISDN must match with one of the VisitingCCodes prefix entries. For example, if the MSISDN has the number 5031273289900, in order to pass the VisitingCCode prefix check, 503 will need to be matched against the VisitingCCode prefixes.

Type: numeric.

Optionality: Optional.

Allowed: code = numeric

Default: code = "93" (AF), "503" (SV)

Notes: VisitingCCodes is used in conjunction with CountryCodes prefix

Example:

```
VisitingCCodes = [ {code="93"  }
                   {code="503"  }
                 ]
```

CountryCodes

Syntax: CountryCodes = A number of visitingCCode prefixes plus country code prefixes combined together and the minimum time allowed to move between countries (in minutes, a maximum of 4 digits must include leading zeros).

Description: When processing a RIMS query result message as part of velocity checking, the VMSC and the VMSC digits will have the Velocity VisitingCCode prefix pre-pended. The resultant number must then match with the country code prefix list in order to continue. For example, if the original MSISDN has the number 5031273289900, in order to pass the VisitingCCode prefix check, the first three digits (503) in the number will be matched against the VisitingCCode prefixes. In order to pass the Country Code prefix test, the VMSC number returned from the RIMS query (441273289900) will have 503 pre-pended (50344127328900) and then be matched with Country code prefix (50344).

Type: numeric.

Optionality: Optional.

Allowed:

Default: time is "0030", "0120" and visiting country code is "9344" AFUK "50344" SVUK

Notes: The VLR Velocity prefixes act as a parent list followed by a child list that is associated with the parent list. The association of the parent with the child is obtained by pre-pending the parent prefix to the child prefix. For example, if 503 is the parent prefix and 44 is the child prefix, then the County Code prefix must equal 50344.

Example: CountryCodes = [{ min_allowed_time="0030", visiting_and_country_code="50344" }]

RejectFailCodes

Syntax: ValidationPrefixCodes = A number of area prefix codes

Description: If the VLR validation fails and the gt cgpn prefix is found within this prefix list, the resultant message sent back will be a TCAP abort.

Type: numeric.

Optionality: Optional.

Allowed:

Default: 441273, 2711473

Notes:

Example:
```
RejectFailCodes = [ { reject_fail_code="441273"  }
                    { reject_fail_code="2711473" }
                  ]
```

DiscardFailCodes

Syntax: ValidationPrefixCodes = A number of area prefix codes

Description: If the VLR validation fails, the gt cgpn prefix is compared with the RejectFailCodes prefix list. If not found, it is then compared with the DiscardFailCodes prefix list. If found, a system error message (alarm) is generated.

Type: numeric.

Optionality: Optional.

Allowed:

Default: 441255,2711455

Notes: If the "Reject Fail codes" prefix check and the "Discard fail codes" prefix check

fail, the message is accepted but not relayed back to the STP.

Example:
```
DiscardFailCodes = [ { discard_fail_code="441255"  }
                     { discard_fail_code="2711455"  }
                   ]
```

map_version

Syntax: `map_version = map_version_number`

Description: Sets the MAP version to be used if a message is received with no application context.

Type: Integer

Optionality: Optional (default used if not set).

Allowed: 1, 2, or 3

Default: 1

Notes:

Example:
```
map_version = 1
```

pc_format

Syntax: `pc_format = "Point_Code_standard"`

Description: The point code standard used in the configuration

Type: String

Optionality: Optional (default used if not set).

Allowed: ITU-TS or ANSI

Default: ITU-TS

Notes: ITU-TS point codes are encoded as a 14-bit structure consisting of a:

- 3-bit zone identification, and 8-bit area/network identification
- 3-bit signaling point identification.

ANSI point codes are encoded in a 24-bit structure consisting of an:

- 8-bit network identification
- 8-bit cluster identification, and
- 8-bit member identification.

Example:
```
pc_format = "ITU-TS"
```

ValidationPrefixCodes

Syntax: `ValidationPrefixCodes = A number of area prefix codes`

Description: When processing VLR validation messages (for example, ActivateSS), the MSISDN digits must match the prefix within the ValidationPrefixCodes prefix list in order for VLR Validation to continue.

Type: numeric.

Optionality: Optional.

Allowed:

Default: 441273, 2711473

Notes:

Example:
```
ValidationPrefixCodes = [ { validation_prefx_code="441273"  }
                          { validation_prefx_code="2711473"  }
                        ]
```

size

Syntax: `size = [number of MSISDN requests to cache]`
Description: Maximum number of MSISDN requests to cache at any one time
Type: numeric.
Optionality: Optional.
Allowed:
Default: 1000000
Notes:
Example: size = 1000000

ttl

Syntax: `ttl = [number of seconds that MSISDN remains in the cache]`
Description: Number of seconds that an MSISDN remains in the cache.
Type: numeric.
Optionality: Optional.
Allowed:
Default: 3600
Notes:
Example: ttl = 3600

rims_if

Syntax: `rims_if = [Name of the messaging manager navigator interface]`
Description: Name of the configured Messaging Manager Navigator interface.
Type: string.
Optionality: Optional.
Allowed:
Default: rimsIf
Notes: This must match the interface name in /IN/service_packages/SLEE/etc/SLEE.cfg.
Example: rims_if = "rimsIf"

vmsc_if

Syntax: `vmsc_if = [Name of the sua_if interface]`
Description: Name of the configured sua_if interface to talk to the VMSC.
Type: string.
Optionality: Optional.
Allowed:
Default: vmscIf
Notes: This must match the interface name in /IN/service_packages/SLEE/etc/SLEE.cfg.
Example: vmsc_if = "vmscIf"

Failure

If mfw fails, alarms are raised to syslog.

Output

mfw writes error messages to the system messages file and writes additional output to **/IN/service_packages/MFW/tmp/mfw.log**.

Chapter 4

Tools and Utilities

Overview

Introduction

This chapter explains the tools and utilities that are available.

In this chapter

This chapter contains the following topics.

Statistics

Introduction

Messaging Firewall collects statistics using the standard SMS statistic mechanism and stores them to the SMF database. For more information about how the statistics are collected, see *SMS Technical Guide*.

VLR Steering Statistics

The following table describes the statistics that VLR Steering uses.

Statistic	Description
MFW_MAP2_UPDATE_LOCATION_TOTAL	Total UPDATE_LOCATION Map2 TOTAL messages. TOTAL equals the total number of updateLocation messages received.
MFW_MAP2_UPDATE_LOCATION_ALLOW	Total UPDATE_LOCATION Map2 ALLOW messages. ALLOW equals the total number of updateLocation messages successfully processed.
MFW_MAP2_UPDATE_LOCATION_REJECT	Total UPDATE_LOCATION Map2 REJECT messages. REJECT equals the total number of updateLocation messages steered away from the platform, sent a TCAP reject.
MFW_MAP2_UPDATE_LOCATION_DISCARD	Total UPDATE_LOCATION Map2 DISCARD messages. DISCARD equals the total number of updateLocation messages that failed the Visiting Country code / Country code prefix checks or there is a match but updateLocation has been sent within a time period shorter than the minimum time allowed.

VLR Validation Statistics

The following table describes the statistics that VLR Validation uses.

Statistic	Description
MFW_MAP2_REGISTER_SS_TOTAL	Total REGISTER_SS Map2 TOTAL messages
MFW_MAP2_REGISTER_SS_ALLOW	Total REGISTER_SS Map2 ALLOW messages
MFW_MAP2_REGISTER_SS_ACCEPT	Total REGISTER_SS Map2 ACCEPT messages
MFW_MAP2_REGISTER_SS_REJECT	Total REGISTER_SS Map2 REJECT messages
MFW_MAP2_REGISTER_SS_DISCARD	Total REGISTER_SS Map2 DISCARD messages
MFW_MAP2_ERASE_SS_TOTAL	Total ERASE_SS Map2 TOTAL messages
MFW_MAP2_ERASE_SS_ALLOW	Total ERASE_SS Map2 ALLOW messages
MFW_MAP2_ERASE_SS_ACCEPT	Total ERASE_SS Map2 ACCEPT messages
MFW_MAP2_ERASE_SS_REJECT	Total ERASE_SS Map2 REJECT messages
MFW_MAP2_ERASE_SS_DISCARD	Total ERASE_SS Map2 DISCARD messages
MFW_MAP2_ACTIVATE_SS_TOTAL	Total ACTIVATE_SS Map2 TOTAL messages
MFW_MAP2_ACTIVATE_SS_ALLOW	Total ACTIVATE_SS Map2 ALLOW messages
MFW_MAP2_ACTIVATE_SS_ACCEPT	Total ACTIVATE_SS Map2 ACCEPT messages
MFW_MAP2_ACTIVATE_SS_REJECT	Total ACTIVATE_SS Map2 REJECT messages
MFW_MAP2_ACTIVATE_SS_DISCARD	Total ACTIVATE_SS Map2 DISCARD messages
MFW_MAP2_DEACTIVATE_SS_TOTAL	Total DEACTIVATE_SS Map2 TOTAL messages
MFW_MAP2_DEACTIVATE_SS_ALLOW	Total DEACTIVATE_SS Map2 ALLOW messages
MFW_MAP2_DEACTIVATE_SS_ACCEPT	Total DEACTIVATE_SS Map2 ACCEPT messages
MFW_MAP2_DEACTIVATE_SS_REJECT	Total DEACTIVATE_SS Map2 REJECT messages
MFW_MAP2_DEACTIVATE_SS_DISCARD	Total DEACTIVATE_SS Map2 DISCARD messages
MFW_MAP2_INTERROGATE_SS_TOTAL	Total INTERROGATE_SS Map2 TOTAL messages
MFW_MAP2_INTERROGATE_SS_ALLOW	Total INTERROGATE_SS Map2 ALLOW messages
MFW_MAP2_INTERROGATE_SS_ACCEPT	Total INTERROGATE_SS Map2 ACCEPT messages
MFW_MAP2_INTERROGATE_SS_REJECT	Total INTERROGATE_SS Map2 REJECT messages
MFW_MAP2_INTERROGATE_SS_DISCARD	Total INTERROGATE_SS Map2 DISCARD messages
MFW_MAP2_REGISTER_PASSWORD_TOTAL	Total REGISTER_PASSWORD Map2 TOTAL messages
MFW_MAP2_REGISTER_PASSWORD_ALLOW	Total REGISTER_PASSWORD Map2 ALLOW messages
MFW_MAP2_REGISTER_PASSWORD_ACCEPT	Total REGISTER_PASSWORD Map2 ACCEPT messages
MFW_MAP2_REGISTER_PASSWORD_REJECT	Total REGISTER_PASSWORD Map2 REJECT messages
MFW_MAP2_REGISTER_PASSWORD_DISCARD	Total REGISTER_PASSWORD Map2 DISCARD messages
MFW_MAP2_PURGE_MS_TOTAL	Total PURGE_MS Map2 TOTAL messages
MFW_MAP2_PURGE_MS_ALLOW	Total PURGE_MS Map2 ALLOW messages
MFW_MAP2_PURGE_MS_ACCEPT	Total PURGE_MS Map2 ACCEPT messages
MFW_MAP2_PURGE_MS_REJECT	Total PURGE_MS Map2 REJECT messages
MFW_MAP2_PURGE_MS_DISCARD	Total PURGE_MS Map2 DISCARD messages

Statistic	Description
MFW_MAP1_BEG_SUB_ACTIVTY_TOTAL	Total BEG_SUB_ACTIVTY Map1 TOTAL messages
MFW_MAP1_BEG_SUB_ACTIVTY_ALLOW	Total BEG_SUB_ACTIVTY Map1 ALLOW messages
MFW_MAP1_BEG_SUB_ACTIVTY_ACCEPT	Total BEG_SUB_ACTIVTY Map1 ACCEPT messages
MFW_MAP1_BEG_SUB_ACTIVTY_REJECT	Total BEG_SUB_ACTIVTY Map1 REJECT messages
MFW_MAP1_BEG_SUB_ACTIVTY_DISCARD	Total BEG_SUB_ACTIVTY Map1 DISCARD messages
MFW_MAP1_PROCESS_USS_DATA_TOTAL	Total PROCESS_USS_DATA Map1 TOTAL messages
MFW_MAP1_PROCESS_USS_DATA_ALLOW	Total PROCESS_USS_DATA Map1 ALLOW messages
MFW_MAP1_PROCESS_USS_DATA_ACCEPT	Total PROCESS_USS_DATA Map1 ACCEPT messages
MFW_MAP1_PROCESS_USS_DATA_REJECT	Total PROCESS_USS_DATA Map1 REJECT messages
MFW_MAP1_PROCESS_USS_DATA_DISCARD	Total PROCESS_USS_DATA Map1 DISCARD messages
MFW_MAP2_RESTORE_DATA_TOTAL	Total RESTORE_DATA Map2 TOTAL messages
MFW_MAP2_RESTORE_DATA_ALLOW	Total RESTORE_DATA Map2 ALLOW messages
MFW_MAP2_RESTORE_DATA_ACCEPT	Total RESTORE_DATA Map2 ACCEPT messages
MFW_MAP2_RESTORE_DATA_REJECT	Total RESTORE_DATA Map1 REJECT messages
MFW_MAP2_RESTORE_DATA_DISCARD	Total RESTORE_DATA Map1 DISCARD messages
MFW_MAP1_SM_TOTAL	Total MAP1 Map 1 TOTAL messages
MFW_MAP1_SM_ALLOW	Total MAP1 Map 1 ALLOW messages
MFW_MAP1_SM_ACCEPT	Total SM Map 1 ACCEPT messages
MFW_MAP1_SM_REJECT	Total SM Map 1 REJECT messages
MFW_MAP1_SM_DISCARD	Total SM Map 1 DISCARD messages
MFW_MAP2_SM_MO_TOTAL	Total SM_MO Map 2 TOTAL messages
MFW_MAP2_SM_MO_ALLOW	Total SM_MO Map 2 ALLOW messages
MFW_MAP2_SM_MO_ACCEPT	Total SM_MO Map 2 ACCEPT messages
MFW_MAP2_SM_MO_REJECT	Total SM_MO Map 2 REJECT messages
MFW_MAP2_SM_MO_DISCARD	Total SM_MO Map 2 DISCARD messages
MFW_MAP2_SM_MT_TOTAL	Total SM_MT Map 2 TOTAL messages
MFW_MAP2_SM_MT_ALLOW	Total SM_MT Map 2 ALLOW messages
MFW_MAP2_SM_MT_ACCEPT	Total SM_MT Map 2 ACCEPT messages
MFW_MAP2_SM_MT_REJECT	Total SM_MT Map 2 REJECT messages
MFW_MAP2_SM_MT_DISCARD	Total SM_MT Map 2 DISCARD messages
MFW_MAP3_SM_MT_TOTAL	Total SM_MT Map 3 TOTAL messages
MFW_MAP3_SM_MT_ALLOW	Total SM_MT Map 3 ALLOW messages
MFW_MAP3_SM_MT_ACCEPT	Total SM_MT Map 3 ACCEPT messages
MFW_MAP3_SM_MT_REJECT	Total SM_MT Map 3 REJECT messages
MFW_MAP3_SM_MT_DISCARD	Total SM_MT Map 3 DISCARD messages
MFW_MAP_SRI_TOTAL	Total SRI TOTAL messages

Statistic	Description
MFW_MAP_SRI_ALLOW	Total SRI ALLOW messages
MFW_MAP_SRI_ACCEPT	Total SRI ACCEPT messages
MFW_MAP_SRI_REJECT	Total SRI REJECT messages
MFW_MAP_SRI_DISCARD	Total SRI DISCARD messages
MFW_MAP_SRI_SM_TOTAL	Total SRI SM TOTAL messages
MFW_MAP_SRI_SM_ALLOW	Total SRI SM ALLOW messages
MFW_MAP_SRI_SM_ACCEPT	Total SRI SM ACCEPT messages
MFW_MAP_SRI_SM_REJECT	Total SRI SM REJECT messages
MFW_MAP_SRI_SM_DISCARD	Total SRI SM DISCARD messages

where:

- TOTAL is the total number of messages received.
- ALLOW is the total number of successful messages.
- ACCEPT is the total number of messages that have been silently accepted after failing VLR validation.
- REJECT is the total number of messages that have been rejected after failing VLR validation.
- DISCARD is the total number of messages that have been discarded after failing VLR validation.

Glossary of Terms

ACS

Advanced Control Services configuration platform.

CC

Country Code. Prefix identifying the country for a numeric international address.

Connection

Transport level link between two peers, providing for multiple sessions.

FDA

First Delivery Attempt - the delivery of a short message directly to the SME rather than relaying it through the MC.

GPRS

General Packet Radio Service - employed to connect mobile cellular users to PDN (Public Data Network- for example the Internet).

GSM

Global System for Mobile communication.

It is a second generation cellular telecommunication system. Unlike first generation systems, GSM is digital and thus introduced greater enhancements such as security, capacity, quality and the ability to support integrated services.

HLR

The Home Location Register is a database within the HPLMN (Home Public Land Mobile Network). It provides routing information for MT calls and SMS. It is also responsible for the maintenance of user subscription information. This is distributed to the relevant VLR, or SGSN (Serving GPRS Support Node) through the attach process and mobility management procedures such as Location Area and Routing Area updates.

HPLMN

Home PLMN

HTML

HyperText Markup Language, a small application of SGML used on the World Wide Web.

It defines a very simple class of report-style documents, with section headings, paragraphs, lists, tables, and illustrations, with a few informational and presentational items, and some hypertext and multimedia.

IMSI

International Mobile Subscriber Identifier. A unique identifier allocated to each mobile subscriber in a GSM and UMTS network. It consists of a MCC (Mobile Country Code), a MNC (Mobile Network Code) and a MSIN (Mobile Station Identification Number).

The IMSI is returned by the HLR query (SRI-SM) when doing FDA. This tells the MSC exactly who the subscriber is that the message is to be sent to.

IN

Intelligent Network

ISDN

Integrated Services Digital Network - set of protocols for connecting ISDN stations.

ISUP

ISDN User Part - part of the SS7 protocol layer and used in the setting up, management, and release of trunks that carry voice and data between calling and called parties.

ITU

International Telecommunication Union

MAP

Mobile Application Part - a protocol which enables real time communication between nodes in a mobile cellular network. A typical usage of the protocol would be for the transfer of location information from the VLR to the HLR.

MC

Message Centre. Also known as SMSC.

MCC

Mobile Country Code. In the location information context, this is padded to three digits with leading zeros. Refer to ITU E.212 ("Land Mobile Numbering Plan") documentation for a list of codes.

Messaging Manager

The Messaging Manager service and the Short Message Service components of Oracle Communications Network Charging and Control product. Component acronym is MM (formerly MMX).

MM

Messaging Manager. Formerly MMX, see also *XMS* (on page 25) and *Messaging Manager* (on page 22).

MNC

Mobile Network Code. The part of an international address following the mobile country code (MCC), or at the start of a national format address. This specifies the mobile network code, that is, the operator owning the address. In the location information context, this is padded to two digits with a leading zero. Refer to ITU E.212 ("Land Mobile Numbering Plan") documentation for a list of codes.

MO

Mobile Originated

MS

Mobile Station

MSC

Mobile Switching Centre. Also known as a switch.

MSIN

Mobile Station Identification Number.

MSISDN

Mobile Station ISDN number. Uniquely defines the mobile station as an ISDN terminal. It consists of three parts; the country code (CC), the national destination code (NDC) and the subscriber number (SN).

MT

Mobile Terminated

MTP

Message Transfer Part (part of the SS7 protocol stack).

PLMN

Public Land Mobile Network

RIMS

Routing Information for Mobile Services. Used to cache HLR lookup information.

Note: Now known as "Messaging Manager Navigator".

SCCP

Signalling Connection Control Part (part of the SS7 protocol stack).

SGML

Standard Generalized Markup Language. The international standard for defining descriptions of the structure of different types of electronic document.

SGSN

Serving GPRS Support Node

SLC

Service Logic Controller (formerly UAS).

SLEE

Service Logic Execution Environment

SME

Short Message Entity - This is an entity which may send or receive short messages. It may be located in a fixed network, a mobile, or an SMSC.

SMS

Depending on context, can be:

- Service Management System hardware platform
- Short Message Service
- Service Management System platform
- NCC Service Management System application

SN

Service Number

SRI

Send Routing Information - This process is used on a GSM network to interrogate the HLR for subscriber routing information.

SS7

A Common Channel Signalling system is used in many modern telecoms networks that provides a suite of protocols which enables circuit and non-circuit related information to be routed about and between networks. The main protocols include MTP, SCCP and ISUP.

STP

Signalling Transfer Point. Telecom equipment routing SS7 signalling messages.

SUA

Signalling Connection Control Part User Adaptation Layer

TCAP

Transaction Capabilities Application Part – layer in protocol stack, message protocol.

VLR

Visitor Location Register - contains all subscriber data required for call handling and mobility management for mobile subscribers currently located in the area controlled by the VLR.

VMSC

Visited Mobile Switching Centre

XML

eXtensible Markup Language. It is designed to improve the functionality of the Web by providing more flexible and adaptable information identification.

It is called extensible because it is not a fixed format like HTML. XML is a `metalanguage' — a language for describing other languages—which lets you design your own customized markup languages for limitless different types of documents. XML can do this because it's written in SGML.

XMS

Three letter code used to designate some components and path locations used by the Oracle Communications Network Charging and Control *Messaging Manager* (on page 22) service and the Short Message Service. The published code is *MM* (on page 22) (formerly MMX).

Index

T

V

W

X