Oracle® Communications Convergent Charging Controller

VPN Provisioning Interface Commands

Release 12.0.0

December 2017

ORACLE®

Copyright

Contents

About This Document

Scope

The scope of this document includes all the information required to configure the VPN PI commands.

Audience

The audience for this document includes system administrators responsible for the monitoring, maintenance, and configuration of the Oracle Communications Convergent Charging Controller IN applications.

Prerequisites

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.

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

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:

- *Provisioning Interface User's and Technical Guide*
- *Virtual Private Network User's Guide*

Document Conventions

Typographical Conventions

The following terms and typographical conventions are used in the Oracle Communications Convergent Charging Controller 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

PI Commands Overview

Overview

Introduction

The provisioning interface (PI) uses TCP/IP-based UNIX sockets to receive provisioning commands and parameters. These are translated into SQL commands that update prepaid application tables of the SMF and E2BE Oracle databases. This chapter defines the rules and packages required to translate the provisioning commands into SQL commands.

In this chapter

This chapter contains the following topics.

Command List

Command list

The following table lists the available PI commands for provisioning VPN information on the SMS. To use the commands they must have been installed with the `piVpnSms` package.

Function	Command
Add a VPN wide default tariff	VPNCU1=ADD
Add a VPN network	VPNNW1=ADD
Modify a VPN network	VPNNW1=CHG
Delete a VPN network	VPNNW1=DEL
Query a VPN network	VPNNW1=QRY
Add a white list number	VPNNW2=ADD
Delete a white list number	VPNNW2=DEL
Query a white list number	VPNNW2=QRY
Add a network speed dial number	VPNNW3=ADD
Change a network speed dial number	VPNNW3=CHG
Delete a network speed dial number	VPNNW3=DEL
Query a network speed dial number	VPNNW3=QRY
Add a VPN station	VPNST1=ADD
Change a VPN station	VPNST1=CHG
Delete a VPN station	VPNST1=DEL
Query a VPN station	VPNST1=QRY
Add a VPN station white list	VPNST2=ADD

Function	Command
Change a VPN station white list	VPNST2=CHG
Delete a VPN station white list	VPNST2=DEL
Query a VPN station white list	VPNST2=QRY
Add a VPN station hunting list	VPNST5=ADD
Change a VPN station hunting list	VPNST5=CHG
Delete a VPN station hunting list	VPNST5=DEL
Query a VPN station hunting list	VPNST5=QRY
Add a VPN station hunting list plan	VPNST6=ADD
Change a VPN station hunting list plan	VPNST6=CHG
Delete a VPN station hunting list plan	VPNST6=DEL
Query a VPN station hunting list plan	VPNST6=QRY
Change the tariff for all VPN stations on a network	VPNST7=CHG

About Service Provider Restrictions

For security reasons, the data that a PI user can query or modify is restricted by service provider. This means that you can run PI commands to query or modify the data only for a service provider that is associated with your PI user.

The system administrator specifies which service providers to associate with your PI user on the **Users** tab in the Administration screen in the PI UI. For more information, see *PI User's and Technical Guide*.

Attempts to run PI commands for a service provider that is not associated with your PI user will result in a negative acknowledgement (NACK) message being returned.

Chapter 2

VPN Network

Overview

Introduction

This chapter explains the VPN PI commands for provisioning VPN Networks.

In this chapter

This chapter contains the following topics.

Add a Network

Name

VPNNW1=ADD

Description

Adds a new VPN Network definition.

Required Parameters

Here are the required parameters for this command.

NAME

Syntax: NAME=*name*
Description: Name of VPN Network.
Format: String
Example: NAME="VPNNW1"

PROVIDER

Syntax: PROVIDER=*sp*
Description: Name of ACS service provider.
Format: String
Example: PROVIDER="Boss"

ORIGINATING

Syntax: ORIGINATING=*origcp*
Description: Originating ACS Call Plan.

Format: String
Constraints: Should exist in acs_call_plan and start “VPN_”
Example: `ORIGINATING="VPN_Originating"`

TERMINATING

Syntax: `TERMINATING=termcp`
Description: Terminating ACS Call Plan.
Format: String
Constraints: Should exist in acs_call_plan and start “VPN_”
Example: `TERMINATING="VPN_Terminating"`

MANAGEMENT

Syntax: `MANAGEMENT=mancp`
Description: Management ACS Call Plan.
Format: String
Constraints: Should exist in acs_call_plan and start “VPN_”
Example: `MANAGEMENT="VPN_Management"`

SITE_CODE

Syntax: `SITE_CODE=scode`
Description: A site code.
Format: 10 character string containing only the characters 0123456789ABCD*#
Note:
* == B and is stored as B in the database
# == C and is stored as C in the database
Constraints: Cannot be a sub or super string of an existing site code.
Example: `SITE_CODE="6449398461"`

Constraint required parameters

This command accepts the following constraint required parameter.

EXTLENGTH

Syntax: `EXTLENGTH=len`
Description: Extension Length for a station extension number.
Format: Number
Default: 4
Example: `EXTLENGTH=4`

Optional parameters

This command accepts the following optional parameters.

PHYSRANGE

Syntax: `PHYSRANGE=Y|N`
Description: Physical range needed?
Format: Y or N.
Default: N

Example: PHYSRANGE=N

COMMENTS

Syntax: COMMENTS=*text*
Description: Comment field.
Format: String
Example: COMMENTS="This is a comment"

PRESENTONNETADDR

Syntax: PRESENTONNETADDR=*Y|N*
Description: Present the on-net address?
Format: Y or N
Default: N
Example: PRESENTONNETADDR=N

RESTRICTCLI

Syntax: RESTRICTCLI=*Y|N*
Description: Restrict CLI?
Format: Y or N
Default: N
Example: RESTRICTCLI=N

ALLOWSHORTEXT

Syntax: ALOWSHORTEXT=*Y|N*
Description: Allow station IDs with less than EXTLENGTH digits?
Format: Y or N
Constraints: If ALLOWSHORTEXT is specified, then EXTLENGTH must also be specified.
Default: Y
Example: ALLOWSHORTEXT=Y

TARIFFNAME

Syntax: TARIFFNAME=*tariff name*
Description: Default ACS tariff name to use.
Format: String
Note: Matches ACS_TARIFF_CODE
Example: TARIFFNAME="TR1"

TAG

Syntax: TAG=*tag*
Description: Profile tag(s) to alter.
Format: Profile tag name(s).
Allowed: Pre-defined name corresponding to PROFILE_TAG_NAME in the ACS_PROFILE_DETAILS table.
Notes: Multiple tags should be separated by the pipe symbol '|'.
Example: TAG=LANGUAGE

VALUE

Syntax: VALUE=*val*

Description: The value of the tag(s).

Format: Profile tag value(s).

The VALUE should match the type of the TAG specified, as defined in the ACS_PROFILE_DETAILS table.

If..	VALUE format is...
DATE	"YYYYMMDDHHMMSS"
BOOLEAN	"T" (true) or "F" (false)
INTEGER	a decimal integer, maximum 4 bytes.
BYTE	a signed single byte as a decimal integer (-128 to 127).
PREFIX or OPREFIX,	the prefix tree data should be specified separated by the colon character. To specify a colon in the data, prefix it with a backslash (\:). To specify a backslash, use two backslashes together (\\). A single backslash will result in a badly formatted parameter error code 68. The maximum number of characters or digits for a single value is 255. If OPREFIX, the value data should be in the desired order.
STRING	free-form text.
NSTRING	limited to the digits 0 to 9

For all types, the separator characters comma and pipe cannot be used.

Notes: Multiple values should be separated by the pipe symbol '|' and be in the same order as for TAG.

Example: VALUE=8

Logic and constraints

- When creating a new VPN Network: NAME, PROVIDER, ORIGINATING, TERMINATING, and MANAGEMENT must be specified.
- An empty Blacklist will be created by default (i.e. no numbers barred)
- To specify more than one tag to update, the TAG parameter should contain the names of the tags separated by pipe symbols. The VALUE parameter should contain the values separated by pipe symbols in the same order as for TAG
- The TAG and VALUE parameters must have the same number of items.

Success return

VPNNW1=ADD:ACK[:NETID=*vpn_network.id*];

Error codes

Error Codes: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 19, 20, and 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Change a VPN Network Details

Name

VPNNW1=CHG

Description

Changes the details associated with a VPN network.

Required Parameter

Here is the required parameter for this command.

NAME

Syntax: NAME=*name*
Description: Name of VPN Network.
Format: String
Example: NAME="VPNNW1"

Constraint required parameters

This command accepts the following constraint required parameter.

NEW_NAME

Syntax: NEW_NAME=*name*
Description: The new NAME for VPN Network.
Format: String
Constraints: New name should not already exist.
Example: NEW_NAME="VPNNW2"

PROVIDER

Syntax: PROVIDER=*sp*
Description: Name of ACS service provider.
Format: String
Example: PROVIDER="Boss"

ORIGINATING

Syntax: ORIGINATING=*origcp*
Description: Originating ACS Call Plan.
Format: String
Constraints: Should exist in acs_call_plan and start “VPN_”
Example: ORIGINATING="VPN_Originating"

TERMINATING

Syntax: TERMINATING=*termcp*
Description: Terminating ACS Call Plan.
Format: String

Constraints: Should exist in acs_call_plan and start “VPN_”
Example: `TERMINATING="VPN_Terminating"`

MANAGEMENT

Syntax: `MANAGEMENT=mancp`
Description: Management ACS Call Plan.
Format: String
Constraints: Should exist in acs_call_plan and start “VPN_”
Example: `MANAGEMENT="VPN_Management"`

PHYSRANGE

Syntax: `PHYSRANGE=Y|N`
Description: Physical range needed?
Format: Y or N.
Default: N
Example: `PHYSRANGE=N`

COMMENTS

Syntax: `COMMENTS=text`
Description: Comment field.
Format: String
Example: `COMMENTS="This is a comment"`

PRESENTONNETADDR

Syntax: `PRESENTONNETADDR=Y|N`
Description: Present the on-net address?
Format: Y or N
Default: N
Example: `PRESENTONNETADDR=N`

RESTRICTCLI

Syntax: `RESTRICTCLI=Y|N`
Description: Restrict CLI?
Format: Y or N
Default: N
Example: `RESTRICTCLI=N`

TARIFFNAME

Syntax: `TARIFFNAME=tariff name`
Description: Default ACS tariff name to use.
Format: String
Note: Matches ACS_TARIFF_CODE
Example: `TARIFFNAME="TR1"`

SITE_CODE

Syntax: SITE_CODE=*scode*

Description: A site code.

Format: 10 character string containing only the characters 0123456789ABCD*#

Note:
* == B and is stored as B in the database
# == C and is stored as C in the database

Constraints: Cannot be a sub or super string of an existing site code.

Example: SITE_CODE="6449398461"

Optional parameters

This command accepts the following optional parameters.

TAG

Syntax: TAG=*tag*

Description: Profile tag(s) to alter.

Format: Profile tag name(s).

Allowed: Pre-defined name corresponding to PROFILE_TAG_NAME in the ACS_PROFILE_DETAILS table.

Notes: Multiple tags should be separated by the pipe symbol '|'.

Example: TAG=LANGUAGE

VALUE

Syntax: VALUE=*val*

Description: The value of the tag(s).

Format: Profile tag value(s).

The VALUE should match the type of the TAG specified, as defined in the ACS_PROFILE_DETAILS table.

If..	VALUE format is...
DATE	“YYYYMMDDHHMMSS”
BOOLEAN	“T” (true) or “F” (false)
INTEGER	a decimal integer, maximum 4 bytes.
BYTE	a signed single byte as a decimal integer (-128 to 127).
PREFIX or OPREFIX,	the prefix tree data should be specified separated by the colon character. To specify a colon in the data, prefix it with a backslash (\:). To specify a backslash, use two backslashes together (\\). A single backslash will result in a badly formatted parameter error code 68. The maximum number of characters or digits for a single value is 255. If OPREFIX, the value data should be in the desired order.
STRING	free-form text.
NSTRING	limited to the digits 0 to 9

For all types, the separator characters comma and pipe cannot be used.

Notes: Multiple values should be separated by the pipe symbol ‘|’ and be in the same order as for TAG.

Example: `VALUE=8`

Logic and constraints

- NAME must exist
- At least one optional parameter should be specified.
- NEW_NAME should not already exist
- PROVIDER should exist in acs_customer.name
- The TAG and VALUE parameters must have the same number of items
- To delete a TAG value, the TAG and VALUE should be specified with an empty VALUE.
  Examples:
  - "TAG=to_delete,VALUE=", or
  - "TAG=set_1|to_delete|set_2,VALUE=value_1||value_2"

Success return

```
VPNNW1=CHG:ACK;
```

Error codes

Error codes: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 18, 19, 20, 21, 22, 23.

See *PI Command Errors* (on page 70) for a description of error codes.

Delete a VPN Network

Name

VPNNW1=DEL

Description

Delete a VPN Network and *all stations* defined for that network in vpn_station.

Required Parameter

Here is the required parameter for this command.

NAME

Syntax: NAME=*name*
Description: Name of VPN Network.
Format: String
Example: NAME="VPNNW1"

Constraint required parameters

There are no constraint required parameters.

Optional parameters

There are no optional parameters.

Logic and constraints

- NAME must exist.

Success return

VPNNW1=DEL:ACK; if the network has stations defined.

VPNNW1=DEL:ACK:0; if no stations defined for network.

Error codes

Error codes: 1, 19, and 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Query a VPN Network

Name

VPNNW1=QRY

Description

Query a VPN network's details.

Required parameters

This command has no required parameters.

Constraint required parameters

This command accepts the following constraint required parameter.

NAME

Syntax: NAME=*name*
Description: Name of VPN Network.
Format: String
Example: NAME="VPNNW1"

LISTTYPE

Syntax: LISTTYPE=*type*
Description: The type of list to return.
Allowed: Valid values:

Value	Returns ..
SHORT	a list of network names only.
LONG	a list of network name (vpn_network.name), white list for each network name, speed dial numbers for each network name.
DEFTARIFF	the default tariff for that network
TARIFF	a list of ACS Tariffs used by the Network NAME. This is found by listing a distinct list of Tariffs from every station in the network, the network default, and the customer default.
STATION	a list of all stations on this network.

Default: SHORT
Example: LISTTYPE=SHORT

Optional parameter

This command accepts the following optional parameter.

TAG

Syntax: TAG=*tag*
Description: Profile tag(s) to alter.
Format: Profile tag name(s).
Allowed: Pre-defined name corresponding to PROFILE_TAG_NAME in the ACS_PROFILE_DETAILS table.
Notes: Multiple tags should be separated by the pipe symbol '|'.
Example: TAG=LANGUAGE

Logic and constraints

- At least one optional parameter must be specified.
- NAME can be specified to retrieve a sub set of Network(s) information. If it is not specified then information for all networks are returned.

- To specify more than one tag to query, the tag parameter should contain the names of the tags separated by pipe symbols
  - DATE values are returned as YYYYMMDDHHMMSS
  - BOOLEAN values are returned as “T” (true) or “F” (false)
  - INTEGER values are returned as the decimal integer unaltered
  - BYTE values are returned as a decimal integer
  - PREFIX and OPREFIX values are returned colon separated. Colons in the data are prefixed with a backslash (\:). Backslashes are returned as two backslashes together (\\)
  - STRING and NSTRING values are returned unaltered

Success return

`VPNNW1=QRY:QRYNet:ACK[:][result 1|...];`

result n for LISTTYPE=

- SHORT:
  `NETWORK[number]name:name[,tag 1=value 1,...,tag n=value n]`
- LONG:
  `NETWORK[number]name:name, whitelist numbers:number:..., speed dial numbers:number:...[,tag 1=value 1,...,tag n=value n]`
  If no white list numbers are defined, "`none defined`" appears in place of the first number.
- DEFTARIFF:
  `NETWORK[number]name:name, default tariff:tariff code[,tag 1=value 1,...,tag n=value n]`
- TARIFF:
  `NETWORK[number]name:name, tariff(s) defined:tariff code:...[,tag 1=value 1,...,tag n=value n]`
  If no tariffs are defined, "`none defined`" appears in place of the first tariff code.
- STATION:
  `NETWORK[number]stations:station_name_1/physical_address_1:...:station_name_n/physical_address_1[,tag 1=value 1,...,tag n=value n]`
  If no stations are defined, "`none defined`" appears in place of the first station.

Error codes

Error codes: 1, 2 3, 4, 19, 20, and 21

See *PI Command Errors* (on page 70) for a description of error codes.

Chapter 3

Network White List Numbers

Overview

Introduction

This chapter explains the VPN PI commands for provisioning VPN Network White List Numbers.

In this chapter

This chapter contains the following topics.

Add Network White List Numbers

Name

VPNNW2=ADD

Description

Adds new white list numbers.

Required Parameters

Here are the required parameters for this command.

NAME

Syntax: NAME=*name*
Description: Name of VPN Network.
Format: String
Example: NAME="VPNNW1"

NUMBER

Syntax: NUMBER=*num*
Description: Number(s) to add to, or delete from, list.
Format: Single number or colon separated list of numbers.
Example: NUMBER=1234567

Constraint required parameters

There are no constraint required parameters.

Optional parameters

There are no optional parameters.

Logic and constraints

- When adding a white list number: NAME must exist, and NUMBER must be specified.
- If whitelist is not active for Network NAME, then blacklist will be deactivated, numbers in the blacklist deleted, and the whitelist made active with the number(s) supplied in NUMBER.

Success return

`VPNNW2=ADD:ACK[:`*`number of whitelist numbers added`*`];`

Error codes

Error codes: 1, 2, 19, and 20

See *PI Command Errors* (on page 70) for a description of error codes.

Delete Network White List Numbers

Name

VPNNW2=DEL

Description

Deletes white list numbers from a specified VPN Network.

Required Parameters

Here are the required parameters for this command.

NAME

Syntax: `NAME=`*`name`*
Description: Name of VPN Network.
Format: String
Example: `NAME="VPNNW1"`

NUMBER

Syntax: `NUMBER=`*`num`*
Description: Number(s) to add to, or delete from, list.
Format: Single number or colon separated list of numbers.
Example: `NUMBER=1234567`

Constraint required parameters

There are no constraint required parameters.

Optional parameters

There are no optional parameters.

Logic and constraints

- When removing white list numbers, NAME and NUMBER must exist.
- When deleting all numbers in white list, an empty blacklist will become active. This makes no numbers barred.
- If delete from white list is used when black list is active, the NUMBER does not exist for NAME error will be used.
- The NUMBER data must consist of 1 or more values, separated by :
- It is possible to specify ALL as the list of numbers which will indicate delete all numbers.

Success return

```
VPNNW2=DEL:ACK[:number of whitelist numbers deleted];
```

Error codes

Error codes: 1, 2, 3, 19, 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Query Network White List Numbers

Name

VPNNW2=QRY

Description

Queries white list numbersfor a specified VPN Network.

Required Parameters

Here are the required parameters for this command.

NAME

Syntax: NAME=*name*
Description: Name of VPN Network.
Format: String
Example: NAME="VPNNW1"

Constraint required parameters

There are no constraint required parameters.

Optional parameters

There are no optional parameters.

Logic and constraints

- When querying a white list number: NAME must exist.

Success return

VPNNW2=QRY:ACK:NUMBER=*number 1*:*number 2*:...;

- The list of white list numbers will consist of a NUMBER tag with the data format of each number separated with a :
- It is possible for the NUMBER tag data to be zero length.

Error codes

Error codes: 1, 2, 19, 20

See *PI Command Errors* (on page 70) for a description of error codes.

Chapter 4

Network Speed Dial Number

Overview

Introduction

This chapter explains the VPN PI commands for provisioning VPN Network Speed Dial Numbers.

In this chapter

This chapter contains the following topics.

Add a Network Speed Dial Number

Name

VPNNW3=ADD

Description

Adds a new speed dial number to a specified network.

Required Parameters

Here are the required parameters for this command.

NAME

Syntax: NAME=*name*
Description: Name of VPN Network.
Format: String
Example: NAME="VPNNW1"

NUMBER

Syntax: NUMBER=*num*
Description: Number to add to the list.
Format: *xxx*:*yyyyy*:*n*
where:
- *xxx* is the speed dial number
- *yyyyy* is the physical address for termination, and
- *n* is an On-net/Off-net indicator, where Y=On-net and N=Off-net.

Example: NUMBER=1.23456.Y

Constraint required parameters

There are no constraint required parameters.

Optional parameters

There are no optional parameters.

Logic and constraints

- When adding a speed dial number: NAME must exist, and NUMBER must be specified.

Success return

```
VPNNW3=ADD:ACK;
```

Error codes

Error codes: 1, 2, 19, 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Change a Network Speed Dial Number

Name

VPNNW3=CHG

Description

Change a speed dial number for a specified network.

Required Parameters

Here are the required parameters for this command.

NAME

Syntax: NAME=*name*
Description: Name of VPN Network.
Format: String
Example: NAME="VPNNW1"

NUMBER

Syntax: NUMBER=*num*
Description: Number to change from speed dial list.
Format: *xxx*
Example: NUMBER=1

NEW_NUMBER

Syntax: NEW_NUMBER=*num*
Description: New Number.

Format: *xxx*:*yyyyy*:*n*

where:

- *xxx* is the speed dial number
- *yyyyy* is the physical address for termination, and
- *n* is an On-net/Off-net indicator, where Y=On-net and N=Off-net.

Example: `NEW_NUMBER=2.23456.Y`

Constraint required parameters

There are no constraint required parameters.

Optional parameters

There are no optional parameters.

Logic and constraints

- When changing a speed dial number: NAME, NUMBER must exist, and NEW_NUMBER must not already exist.
- NUMBER should be a single existing speed dial number.

Success return

```
VPNNW3=CHG:ACK;
```

Error codes

Error codes: 1, 2, 3, 18, 19, 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Delete a Network Speed Dial Number

Name

VPNNW3=DEL

Description

Delete a speed dial number from a specified network.

Required Parameters

Here are the required parameters for this command.

NAME

Syntax: `NAME=name`

Description: Name of VPN Network.

Format: String

Example: `NAME="VPNNW1"`

NUMBER

Syntax: NUMBER=*num*
Description: Number(s) to delete from speed dial list.
Format: Colon separated list of numbers.
Example: NUMBER=1:2:3

Constraint required parameters

There are no constraint required parameters.

Optional parameters

There are no optional parameters.

Logic and constraints

- NUMBER may be a colon separated list of speed dial numbers.

Success return

VPNNW3=DEL:ACK[:*number of speed dials deleted*];

Error codes

Error codes: 1, 2, 19, 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Query Network Speed Dial Numbers

Name

VPNNW3=QRY

Description

Queries speed dial numbers for a specified network.

Required Parameters

Here are the required parameters for this command.

NAME

Syntax: NAME=*name*
Description: Name of VPN Network.
Format: String
Example: NAME="VPNNW1"

Constraint required parameters

There are no constraint required parameters.

Optional parameters

There are no optional parameters.

Logic and constraints

- NAME must be specified.

Success return

```
VPNNW3=QRY:ACK[:list of speed dials];
```

- The list of speed dial numbers will consist of NUMBER tags with the data format *xxx*:*yyyyy*:*n* where *xxx* is the speed dial number, *yyyyy* is the physical address for termination and *n* is an On-net/Off-net indicator where Y=On-net and N=Off-net.
- Format: `NUMBER=<number 1>|<number 2>|...`
- An ACK may be returned with no NUMBER tags if no Speed dial numbers are defined for this network.

Error codes

Error codes: 1, 19, 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Chapter 5

VPN Station

Overview

Introduction

This chapter explains the VPN PI commands for provisioning VPN Stations.

In this chapter

This chapter contains the following topics.

Add a VPN Station

Name

VPNST1=ADD

Description

Adds a new VPN Station definition.

Required Parameters

Here are the required parameters for this command.

NETNAME

Syntax: NETNAME=*network name*
Description: The name of the VPN Network.
Format: String
Example: NETNAME="Net1"

STNNAME

Syntax: STNNAME=*station name*
Description: The name of VPN Station. (Extension number field in VPN Screen).
Format: Extension number.
Example: STNNAME=123

PHYSADDR

Syntax: PHYSADDR=*msisdn*
Description: The Physical Address, usually real MSISDN.

Format: MSISDN number.
Example: `PHYSADDR=6494111111`

Constraint required parameters

There are no constraint required parameters.

Optional parameters

This command accepts the following optional parameters.

GVNSADDR

Syntax: `GVNSADDR=gvns address`
Description: The GVNS Address name.
Format: String
Example: `GVNSADDR="6449393404"`

VDDIADDR

Syntax: `VDDIADDR=vddi address`
Description: The VDDI Address.
Format: String
Example: `VDDIADDR="6449393404"`

COMMENTS

Syntax: `COMMENTS=text`
Description: Comment field.
Format: String
Example: `COMMENTS="This is a comment"`

OFFNET

Syntax: `OFFNET=Y|N`
Description: Allow or deny off net calls.
Format: Y, or N.
Default: Y
Example: `OFFNET=Y`

INOFFNET

Syntax: `INOFFNET=Y|N`
Description: Allow or deny all incoming calls from off net.
Format: Y, or N.
Example: `INOFFNET=N`

TARIFFNAME

Syntax: `TARIFFNAME=tariff name`
Description: Default ACS tariff name to use.
Format: String
Note: Matches ACS_TARIFF_CODE

Example: TARIFFNAME="TR1"

WHITELIST

Syntax: WHITELIST=*whitelist*

Description: The numbers to add to, or delete from, the white list.

Format: Single number, or colon separated list of numbers.

Example: WHITELIST=6449393404:6449393764

TAG

Syntax: TAG=*tag*

Description: Profile tag(s) to alter.

Format: Profile tag name(s).

Allowed: Pre-defined name corresponding to PROFILE_TAG_NAME in the ACS_PROFILE_DETAILS table.

Notes: Multiple tags should be separated by the pipe symbol '|'.

Example: TAG=LANGUAGE

VALUE

Syntax: VALUE=*val*

Description: The value of the tag(s).

Format: Profile tag value(s).

The VALUE should match the type of the TAG specified, as defined in the ACS_PROFILE_DETAILS table.

If..	VALUE format is...
DATE	"YYYYMMDDHHMMSS"
BOOLEAN	"T" (true) or "F" (false)
INTEGER	a decimal integer, maximum 4 bytes.
BYTE	a signed single byte as a decimal integer (-128 to 127).
PREFIX or OPREFIX,	the prefix tree data should be specified separated by the colon character. To specify a colon in the data, prefix it with a backslash (\:). To specify a backslash, use two backslashes together (\\). A single backslash will result in a badly formatted parameter error code 68. The maximum number of characters or digits for a single value is 255. If OPREFIX, the value data should be in the desired order.
STRING	free-form text.
NSTRING	limited to the digits 0 to 9

For all types, the separator characters comma and pipe cannot be used.

Notes: Multiple values should be separated by the pipe symbol '|' and be in the same order as for TAG.

Example: VALUE=8

Logic and constraints

- The number of digits in the STNNAME must equal EXTLENGTH in the add network command (VPNNW1=ADD) if ALLOWSHORTEXT = N
- The number of digits in the STNNAME must be less than or equal to the EXTLENGTH defined in the add network command (VPNNW1=ADD) if ALLOWSHORTEXT = Y
- If no COMMENTS are supplied, PI will insert an automatic comment to show PI was used.
- If OFFNET is not specified it will be set to Y (default).
- If INOFFNET is not specified it will be set to N (default).
- If WHITELIST is not specified, an empty blacklist will be created and activated. This means no numbers are barred.
- To specify more than one tag to update, the TAG parameter should contain the names of the tags separated by pipe symbols. The VALUE parameter should contain the values separated by pipe symbols in the same order as for TAG.
- The TAG and VALUE parameters must have the same number of items.

Success return

`VPNST1=ADD:ACK[:STNID=`*`vpn_station.id`*`];`

Error codes

Error codes: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 17, 19, 20, 21, 22, 23.

See *PI Command Errors* (on page 70) for a description of error codes.

Change a VPN Station Details

Name

VPNST1=CHG

Description

Changes the details associated with a VPN station.

Required parameters

This command has no required parameters.

Constraint required parameters

This command accepts the following constraint required parameter.

PHYSADDR

Syntax: `PHYSADDR=`*`msisdn`*

Description: The Physical Address, usually real MSISDN.

Format: MSISDN number.

Example: `PHYSADDR=6494111111`

STNNAME

Syntax: `STNNAME=`*`station name`*

Description: The name of VPN Station. (Extension number field in VPN Screen).

Format: Extension number.
Example: `STNNAME=123`

NEW_NAME

Syntax: `NEW_NAME=name`
Description: The new NETNAME for VPN station.
Format: String
Constraints: New name should already exist.
Example: `NEW_NAME="Net2"`

NEW_STNNAME

Syntax: `NEW_STNNAME=exnum`
Description: The new STNNAME for VPN station.
Format: Extension number.
Example: `NEW_STNNAME=124`

NEW_PHYSADDR

Syntax: `NEW_PHYSADDR=addr`
Description: The new PHYSADDR for the VPN station.
Format: MSISDN number.
Example: `NEW_PHYSADDR=6494111112`

GVNSADDR

Syntax: `GVNSADDR=gvns address`
Description: The GVNS Address name.
Format: String
Example: `GVNSADDR="6449393404"`

VDDIADDR

Syntax: `VDDIADDR=vddi address`
Description: The VDDI Address.
Format: String
Example: `VDDIADDR="6449393404"`

COMMENTS

Syntax: `COMMENTS=text`
Description: Comment field.
Format: String
Example: `COMMENTS="This is a comment"`

TARIFFNAME

Syntax: `TARIFFNAME=tariff name`
Description: Default ACS tariff name to use.
Format: String

Note: Matches ACS_TARIFF_CODE

Example: `TARIFFNAME="TR1"`

Optional parameters

This command accepts the following optional parameters.

TAG

Syntax: `TAG=tag`

Description: Profile tag(s) to alter.

Format: Profile tag name(s).

Allowed: Pre-defined name corresponding to PROFILE_TAG_NAME in the ACS_PROFILE_DETAILS table.

Notes: Multiple tags should be separated by the pipe symbol '|'.

Example: `TAG=LANGUAGE`

VALUE

Syntax: `VALUE=val`

Description: The value of the tag(s).

Format: Profile tag value(s).

The VALUE should match the type of the TAG specified, as defined in the ACS_PROFILE_DETAILS table.

If..	VALUE format is...
DATE	"YYYYMMDDHHMMSS"
BOOLEAN	"T" (true) or "F" (false)
INTEGER	a decimal integer, maximum 4 bytes.
BYTE	a signed single byte as a decimal integer (-128 to 127).
PREFIX or OPREFIX,	the prefix tree data should be specified separated by the colon character. To specify a colon in the data, prefix it with a backslash (\:). To specify a backslash, use two backslashes together (\\). A single backslash will result in a badly formatted parameter error code 68. The maximum number of characters or digits for a single value is 255. If OPREFIX, the value data should be in the desired order.
STRING	free-form text.
NSTRING	limited to the digits 0 to 9

For all types, the separator characters comma and pipe cannot be used.

Notes: Multiple values should be separated by the pipe symbol '|' and be in the same order as for TAG.

Example: `VALUE=8`

Logic and constraints

- At least one optional parameter should be specified.
- If specified, NETNAME must exist. It is used in conjunction with STNNAME and PHYSADDR to determine a unique station.
- Either or both STNNAME and PHYSADDR must be specified, exist, and define a unique station
- NETNAME should exist in vpn_network.name
- PHYSADDR must exist in vpn_network_physical_range table.
- GVNSADDR must exist in vpn_network_gvns_range table.
- VDDIADDR must exist in vpn_network_vddi_range table.
- TARIFFNAME should exist in the acs_tariff table.
- TAG is a pre-defined name corresponding to PROFILE_TAG_NAME in the ACS_PROFILE_DETAILS table.
- To specify more than one tag to update, the TAG parameter should contain the names of the tags separated by pipe symbols. The VALUE parameter should contain the values separated by pipe symbols in the same order as for TAG
- The TAG and VALUE parameters must have the same number of items
- To delete a TAG value, the TAG and VALUE should be specified with an empty VALUE. e.g. "TAG=to_delete,VALUE=", or "TAG=set_1|to_delete|set_2,VALUE=value_1||value_2"

Success return

```
VPNST1=CHG:ACK
```

Error codes

Error codes: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 17, 18, 19, 20, 21, 22, 23.

See *PI Command Errors* (on page 70) for a description of error codes.

Delete a VPN Station

Name

VPNST1=DEL

Description

Deletes a VPN station.

Required parameters

This command has no required parameters.

Constraint required parameters

This command accepts the following constraint required parameter.

PHYSADDR

Syntax: PHYSADDR=*msisdn*

Description: The Physical Address, usually real MSISDN.

Format: MSISDN number.

Example: PHYSADDR=6494111111

STNNAME

Syntax: STNNAME=*station name*
Description: The name of VPN Station. (Extension number field in VPN Screen).
Format: Extension number.
Example: STNNAME=123

Optional parameter

This command accepts the following optional parameter.

NETNAME

Syntax: NETNAME=*network name*
Description: The name of the VPN Network.
Format: String
Example: NETNAME="Net1"

Logic and constraints

- If specified, NETNAME must exist. It is used in conjunction with STNNAME and PHYSADDR to determine a unique station.
- Either or both STNNAME and PHYSADDR must be specified, exist, and define a unique station for a single station to be deleted.
- No optional parameters specified will delete all stations for the specified Network in NETNAME.

Success return

```
VPNST1=DEL:ACK
```

Error codes

Error codes: 1, 17, 19, 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Query a VPN Station

Name

VPNST1=QRY

Description

Queries a VPN station's details.

Required parameters

This command has no required parameters.

Constraint required parameters

This command accepts the following constraint required parameter.

STNNAME

Syntax: STNNAME=*station name*
Description: The name of VPN Station. (Extension number field in VPN Screen).
Format: Extension number.
Example: STNNAME=123

PHYSADDR

Syntax: PHYSADDR=*msisdn*
Description: The Physical Address, usually real MSISDN.
Format: MSISDN number.
Example: PHYSADDR=6494111111

Optional parameters

This command accepts the following optional parameters.

NETNAME

Syntax: NETNAME=*network name*
Description: The name of the VPN Network.
Format: String
Example: NETNAME="Net1"

LISTTYPE

Syntax: LISTTYPE=*type*
Description: The type of list to return.
Allowed: Valid values:
- GROUP
- BASIC
- ALL

Default: GROUP
Example: LISTTYPE=GROUP

PHYSADDR

Syntax: PHYSADDRMASK=*pref*
Description: The Prefix to search for MSISDNs (PHYSADDR list).
Format: MSISDN number.
Example: PHYSADDRMASK=64941

TAG

Syntax: TAG=*tag*
Description: Profile tag(s) to alter.
Format: Profile tag name(s).

Allowed: Pre-defined name corresponding to PROFILE_TAG_NAME in the ACS_PROFILE_DETAILS table.

Notes: Multiple tags should be separated by the pipe symbol '|'.

Example: `TAG=LANGUAGE`

Logic and constraints

- One or both of STNNAME or PHYSADDR must be specified.
- If PHYSADDRMASK is specified:
  - STNNAME must also be specified.
  - LISTTYPE must be specified as BASIC.
- LISTTYPE = BASIC can only be used if PHYSADDRMASK is specified.
- TAG is a pre-defined name corresponding to PROFILE_TAG_NAME in the ACS_PROFILE_DETAILS table.
- To specify more than one tag to query, the tag parameter should contain the names of the tags separated by pipe symbols.
  - DATE values are returned as YYYYMMDDHHMMSS
  - BOOLEAN values are returned as "T" (true) or "F" (false)
  - INTEGER values are returned as the decimal integer unaltered
  - BYTE values are returned as a decimal integer
  - PREFIX and OPREFIX values are returned colon separated. Colons in the data are prefixed with a backslash (\:). Backslashes are returned as two backslashes together (\\)
  - STRING and NSTRING values are returned unaltered

Success return

`VPNST1=QRY:ACK[:`*`result of query returned`*`;`

The result may contain one or more of the following comma separated values. There may be more than one of each tag. Records always start with STNNAME:

For LISTTYPE=

- BASIC:
  - STNNAME=*station name*
  - NETNAME=*network name*
  - *tag*=*value*
- GROUP:
  - STNNAME=*station name*
  - NETNAME=*network name*
  - PHYSADDR=*physical address (MSISDN)*
  - *tag*=*value*
- ALL:
  - STNNAME=*station name*
  - NETNAME=*network name*
  - PHYSADDR=*physical address (MSISDN)*
  - VDDIADDR=*vddi address*
  - GVNSADDR=*gvns address*
  - COMMENTS=*comment*
  - TARIFFNAME=*tariff name*
  - OFFNET=*Y* or *N*
  - INOFFNET=*Y* or *N*
  - WHITELIST=*whitelist*
  - *tag*=*value*

Error codes

Error codes: 1, 2, 3, 4, 5, 19, 20, 21.

See *PI Command Errors* (on page 70) for a description of error codes.

Chapter 6

VPN Station White List

Overview

Introduction

This chapter explains the VPN PI commands for provisioning VPN Station White Lists.

In this chapter

This chapter contains the following topics.

Add a VPN Station White List

Name

VPNST2=ADD

Description

Adds a new VPN Station whitelist definition.

Required parameters

This command has no required parameters.

Constraint required parameters

This command accepts the following constraint required parameter.

STNNAME

Syntax: STNNAME=*station name*
Description: The name of VPN Station. (Extension number field in VPN Screen).
Format: Extension number.
Example: STNNAME=123

PHYSADDR

Syntax: PHYSADDR=*msisdn*
Description: The Physical Address, usually real MSISDN.
Format: MSISDN number.
Example: PHYSADDR=6494111111

Optional parameters

This command accepts the following optional parameters.

NETNAME

Syntax: NETNAME=*network name*

Description: The name of the VPN Network.

Format: String

Example: NETNAME="Net1"

WHITELIST

Syntax: WHITELIST=*whitelist*

Description: The numbers to add to, or delete from, the white list.

Format: Single number, or colon separated list of numbers.

Example: WHITELIST=6449393404:6449393764

Logic and constraints

- If specified, NETNAME must exist. It is used in conjunction with STNNAME and PHYSADDR to determine a unique station.
- Either or both STNNAME and PHYSADDR must be specified, exist, and define a unique station.
- WHITELIST number must not already exist for the defined STNNAME and PHYSADDR pair.
- WHITELIST should be a single number, or a colon separated list of valid white list numbers.
- If white list is not active, then black list will be deleted (including numbers), and the white list made active with the number(s) specified by WHITELIST.

Success return

VPNST2=ADD:ACK:*number of whitelist numbers added;*

Error codes

Error codes: 1, 3, 7, 17, 19, 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Change a VPN Station Restriction Details

Name

VPNST2=CHG

Description

Changes the details associated with a VPN station restrictions.

Required parameters

This command has no required parameters.

Constraint required parameters

This command accepts the following constraint required parameter.

STNNAME

Syntax: STNNAME=*station name*
Description: The name of VPN Station. (Extension number field in VPN Screen).
Format: Extension number.
Example: STNNAME=123

PHYSADDR

Syntax: PHYSADDR=*msisdn*
Description: The Physical Address, usually real MSISDN.
Format: MSISDN number.
Example: PHYSADDR=6494111111

OFFNET

Syntax: OFFNET=*Y|N*
Description: Allow or deny off net calls.
Format: Y, or N.
Default: Y
Example: OFFNET=Y

INOFFNET

Syntax: INOFFNET=*Y|N*
Description: Allow or deny all incoming calls from off net.
Format: Y, or N.
Example: INOFFNET=N

WHITELIST

Syntax: WHITELIST=*whitelist*
Description: The existing value in the white list.
Format: Single number.
Example: WHITELIST=6449393404

NEW_OFFNET

Syntax: NEW_OFFNET=*Y|N*
Description: New value for allow/deny off net calls.
Format: Y, or N.
Example: NEW_OFFNET=Y

NEW_INOFFNET

Syntax: INOFFNET=Y|N
Description: New value for Allow or deny all incoming calls from off net.
Format: Y, or N.
Example: INOFFNET=N

NEW_WHITELIST

Syntax: NEW_WHITELIST=*tname*
Description: New value for the white list, replacing old value.
Format: Single number.
Example: NEW_WHITELIST=6449393403

Optional parameters

This command accepts the following optional parameters.

NETNAME

Syntax: NETNAME=*network name*
Description: The name of the VPN Network.
Format: String
Example: NETNAME="Net1"

Logic and constraints

- If specified, NETNAME must exist. It is used in conjuction with STNNAME and PHYSADDR to determine a unique station.
- Either or both STNNAME and PHYSADDR must be specified, exist, and define a unique station
- At least one of OFFNET, INOFFNET or WHITELIST should be specified.
- If OFFNET is specified, NEW_OFFNET must also be specified.
- If INOFFNET is specified, NEW_INOFFNET must also be specified.
- If WHITELIST is specified, NEW_WHITELIST must also be specified.

Success return

```
VPNST2=CHG:ACK
```

Error codes

Error codes: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 17, 18, 19, 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Delete a VPN Station White List

Name

VPNST2=DEL

Description

Deletes a VPN station white list number.

Required parameters

This command has no required parameters.

Constraint required parameters

This command accepts the following constraint required parameter.

STNNAME

Syntax: STNNAME=*station name*
Description: The name of VPN Station. (Extension number field in VPN Screen).
Format: Extension number.
Example: STNNAME=123

PHYSADDR

Syntax: PHYSADDR=*msisdn*
Description: The Physical Address, usually real MSISDN.
Format: MSISDN number.
Example: PHYSADDR=6494111111

Optional parameters

This command accepts the following optional parameters.

NETNAME

Syntax: NETNAME=*network name*
Description: The name of the VPN Network.
Format: String
Example: NETNAME="Net1"

WHITELIST

Syntax: WHITELIST=*whitelist*
Description: The numbers to add to, or delete from, the white list.
Format: Single number, or colon separated list of numbers.
Example: WHITELIST=6449393404:6449393764

Logic and constraints

- If specified, NETNAME must exist. It is used in conjuction with STNNAME and PHYSADDR to determine a unique station.
- Either or both STNNAME and PHYSADDR must be specified, exist, and define a unique station
- Number(s) in WHITELIST must exist in a PHYSADDR range for the Network in NETNAME.
- Either a single number or colon seperated list of numbers must be specified for WHITELIST.

Success return

VPNST2=DEL:ACK[:*number of whitelist numbers deleted*];

Error codes

Error codes: 1, 3, 7, 17, 19, 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Query a Station White List

Name

VPNST2=QRY

Description

Query a white list number for a specified VPN Station.

Required parameters

This command has no required parameters.

Constraint required parameters

This command accepts the following constraint required parameter.

STNNAME

Syntax: STNNAME=*station name*
Description: The name of VPN Station. (Extension number field in VPN Screen).
Format: Extension number.
Example: STNNAME=123

PHYSADDR

Syntax: PHYSADDR=*msisdn*
Description: The Physical Address, usually real MSISDN.
Format: MSISDN number.
Example: PHYSADDR=6494111111

Optional parameter

This command accepts the following optional parameter.

NETNAME

Syntax: NETNAME=*network name*
Description: The name of the VPN Network.
Format: String
Example: NETNAME="Net1"

Logic and constraints

- If specified, NETNAME must exist. It is used in conjuction with STNNAME and PHYSADDR to determine a unique station.
- Either or both STNNAME and PHYSADDR must be specified, exist, and define a unique station

Success return

VPNST2=QRY:ACK[:*result list*];

- *result list* contains one or more of the following:

- OFFNET=*Y* or *N*
- INOFFNET=*Y* or *N*
- WHITELIST=*number1*:*number2*:...

Error codes

Error codes: 1, 7, 17, 19, 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Chapter 7

VPN Station Hunting List

Overview

Introduction

This chapter explains the VPN PI commands for provisioning VPN Station Hunting Lists.

In this chapter

This chapter contains the following topics.

Add a VPN Station Hunting List

Name

VPNST5=ADD

Description

Adds a new VPN Station hunting list.

Required Parameters

Here are the required parameters for this command.

HUNTLIST

Syntax: HUNTLIST=*name*
Description: The name of the Hunt list.
Format: String
Example: HUNTLIST="HL1"

RANK

Syntax: RANK=*order*
Description: The order to apply entries, in ascending order.
Format: Number
Example: RANK=1

TERMNUM

Syntax: TERMNUM=*num*
Description: The terminating number.

Format: Terminating number.
Example: `TERMNUM=649393377`

TIMEOUT

Syntax: `TIMEOUT=num`
Description: The timeout.
Format: Number of seconds. Integer value between 1 and 99.
Example: `TIMEOUT=5`

Constraint required parameters

This command accepts the following constraint required parameter.

STNNAME

Syntax: `STNNAME=station name`
Description: The name of VPN Station. (Extension number field in VPN Screen).
Format: Extension number.
Example: `STNNAME=123`

PHYSADDR

Syntax: `PHYSADDR=msisdn`
Description: The Physical Address, usually real MSISDN.
Format: MSISDN number.
Example: `PHYSADDR=6494111111`

Optional parameters

This command accepts the following optional parameters.

NETNAME

Syntax: `NETNAME=network name`
Description: The name of the VPN Network.
Format: String
Example: `NETNAME="Net1"`

ONNET

Syntax: `ONNET=Y|N`
Description: Is the terminating number an on-net number?
Format: Y, or N
Default: N
Example: `ONNET=Y`

Logic and constraints

- If specified, NETNAME must exist. It is used in conjuction with STNNAME and PHYSADDR to determine a unique station.
- Either or both STNNAME and PHYSADDR must be specified, exist, and define a unique station.

- If HUNTLIST does:
  - not already exist, it will be created.
  - exist, the specified information will be added to that HUNTLIST.
- RANK must either exist or be number of items +1 (the last entry in the list).
- If RANK exists the new value will be inserted into the list.
- If ONNET=Y then TERMNUM must be a valid station address for the network.

Success return

```
VPNST5=ADD:ACK;
```

Error codes

Error codes: 1, 3, 4, 5, 6, 9, 10, 17, 19, 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Change a VPN Station Hunting List

Name

VPNST5=CHG

Description

Changes the details associated with a VPN Station Hunting List.

Required Parameter

Here is the required parameter for this command.

HUNTLIST

Syntax: HUNTLIST=*name*
Description: The name of the Hunt list.
Format: String
Example: HUNTLIST="HL1"

Constraint required parameters

This command accepts the following constraint required parameter.

STNNAME

Syntax: STNNAME=*station name*
Description: The name of VPN Station. (Extension number field in VPN Screen).
Format: Extension number.
Example: STNNAME=123

PHYSADDR

Syntax: PHYSADDR=*msisdn*
Description: The Physical Address, usually real MSISDN.

Format: MSISDN number.
Example: `PHYSADDR=6494111111`

RANK

Syntax: `RANK=order`
Description: The order to apply entries, in ascending order.
Format: Number
Example: `RANK=1`

TIMEOUT

Syntax: `TIMEOUT=num`
Description: The timeout.
Format: Number of seconds. Integer value between 1 and 99.
Example: `TIMEOUT=5`

ONNET

Syntax: `ONNET=Y|N`
Description: Is the terminating number an on-net number?
Format: Y, or N
Default: N
Example: `ONNET=Y`

Optional parameters

This command accepts the following optional parameters.

NETNAME

Syntax: `NETNAME=network name`
Description: The name of the VPN Network.
Format: String
Example: `NETNAME="Net1"`

RANK

Syntax: `NEW_RANK=order`
Description: The order to apply entries, in ascending order.
Format: Number
Example: `NEW_RANK=3`

TIMEOUT

Syntax: `NEW_TIMEOUT=num`
Description: The timeout.
Format: Number of seconds. Integer value between 1 and 99.
Example: `NEW_TIMEOUT=4`

ONNET

Syntax: NEW_ONNET=*Y|N*
Description: Is the terminating number an on-net number?
Format: Y, or N
Default: N
Example: NEW_ONNET=N

Logic and constraints

- The HUNTLIST must already exist.
- If specified, NETNAME must exist. It is used in conjunction with STNNAME and PHYSADDR to determine a unique station.
- Either or both STNNAME and PHYSADDR must be specified, exist, and define a unique station
- If NEW_RANK is specified, then RANK must be specified and exist
- If NEW_TIMEOUT is specified, then TIMEOUT must be specified and exist
- If NEW_ONNET is specified, then ONNET must be specified and exist.
- NEW_ONNET must be Y or N.
- If NEW_ONNET is Y, all defined termination numbers must be VPN numbers for this network.

Success return

```
VPNST5=CHG:ACK;
```

Error codes

Error codes: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 17, 18, 19, 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Delete a VPN Station Hunting List

Name

VPNST5=DEL

Description

Deletes the details associated with a VPN Station Hunting List.

Required Parameter

Here is the required parameter for this command.

HUNTLIST

Syntax: HUNTLIST=*name*
Description: The name of the Hunt list.
Format: String
Example: HUNTLIST="HL1"

Constraint required parameters

This command accepts the following constraint required parameter.

STNNAME

Syntax: STNNAME=*station name*
Description: The name of VPN Station. (Extension number field in VPN Screen).
Format: Extension number.
Example: STNNAME=123

PHYSADDR

Syntax: PHYSADDR=*msisdn*
Description: The Physical Address, usually real MSISDN.
Format: MSISDN number.
Example: PHYSADDR=6494111111

TERMNUM

Syntax: TERMNUM=*num*
Description: The terminating number.
Format: Terminating number.
Example: TERMNUM=649393377

Optional parameter

This command accepts the following optional parameter.

NETNAME

Syntax: NETNAME=*network name*
Description: The name of the VPN Network.
Format: String
Example: NETNAME="Net1"

Logic and constraints

- The HUNTLIST must already exist.
- If specified, NETNAME must exist. It is used in conjuction with STNNAME and PHYSADDR to determine a unique station.
- Either or both STNNAME and PHYSADDR must be specified, exist, and define a unique station
- If deleting a Hunt List entry then STNNAME/PHYSADDR, HUNTLIST and TERMNUM must be specified.

Success return

```
VPNST5=DEL:ACK;
```

Error codes

Error codes: 1, 2, 3, 4, 10, 17, 19, 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Query a VPN Station Hunting List

Name

VPNST5=QRY

Description

Query a VPN station's Hunting List details.

Required Parameter

Here is the required parameter for this command.

HUNTLIST

Syntax: HUNTLIST=*name*
Description: The name of the Hunt list.
Format: String
Example: HUNTLIST="HL1"

Constraint required parameters

This command accepts the following constraint required parameter.

STNNAME

Syntax: STNNAME=*station name*
Description: The name of VPN Station. (Extension number field in VPN Screen).
Format: Extension number.
Example: STNNAME=123

PHYSADDR

Syntax: PHYSADDR=*msisdn*
Description: The Physical Address, usually real MSISDN.
Format: MSISDN number.
Example: PHYSADDR=6494111111

Optional parameter

This command accepts the following optional parameter.

NETNAME

Syntax: NETNAME=*network name*
Description: The name of the VPN Network.
Format: String
Example: NETNAME="Net1"

Logic and constraints

- The HUNTLIST must already exist.

- If specified, NETNAME must exist. It is used in conjunction with STNNAME and PHYSADDR to determine a unique station.
- Either or both STNNAME and PHYSADDR must be specified, exist, and define a unique station.

Success return

```
VPNST5=QRY:ACK:RANK=rank 1:...,TERMNUM=terminating number 1:...,TIMEOUT=timeout
1:...,ONNET=Y/N:...;
```

Each parameter can have a colon separated list of values.

Error codes

Error codes: 1, 2, 10, 17, 19, 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Chapter 8

VPN Hunting List Plan

Overview

Introduction

This chapter explains the VPN PI commands for provisioning VPN Station Hunting List Plans.

In this chapter

This chapter contains the following topics.

Add a VPN Station Hunting List Plan

Name

VPNST6=ADD

Description

Adds a new VPN Station hunt list plan to an existing hunt list definition.

Required Parameters

Here are the required parameters for this command.

HUNTLIST

Syntax: HUNTLIST=*name*
Description: The name of the Hunt list.
Format: String
Example: HUNTLIST="HL1"

TIMERANGE

Syntax: TIMERANGE=*type*
Description: The time range type.
Format: Type is:
- TOD (Time of day)
- DOW (Day of week)
- DOY (Day of year)

Example: TIMERANGE=TOD

STARTTIME

Syntax: STARTTIME=*time*

Description: The start time to use HUNTLIST.

Format: If TIMERANGE=

- TOD: HH:MM
- DOW: DDD:HH:MM
- DOY: MON:DD:HH:MM

HH = hours 00 to 23

MM = minutes 00 to 59

DDD = day prefix SUN,MON,TUE,WED,THU,FRI,SAT

DD = day of month 01 to 31

MON = month prefix JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC

Example: STARTTIME=23:11

ENDTIME

Syntax: ENDTIME=*time*

Description: The end time to use HUNTLIST.

Format: If TIMERANGE=

- TOD: HH:MM
- DOW: DDD:HH:MM
- DOY: MON:DD:HH:MM

HH = hours 00 to 23

MM = minutes 00 to 59

DDD = day prefix SUN,MON,TUE,WED,THU,FRI,SAT

DD = day of month 01 to 31

MON = month prefix JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC

Example: ENDTIME=08:00

Constraint required parameters

This command accepts the following constraint required parameter.

STNNAME

Syntax: STNNAME=*station name*

Description: The name of VPN Station. (Extension number field in VPN Screen).

Format: Extension number.

Example: STNNAME=123

PHYSADDR

Syntax: PHYSADDR=*msisdn*

Description: The Physical Address, usually real MSISDN.

Format: MSISDN number.

Example: PHYSADDR=6494111111

Optional parameters

This command accepts the following optional parameters.

NETNAME

Syntax: NETNAME=*network name*

Description: The name of the VPN Network.

Format: String

Example: NETNAME="Net1"

LOCATION

Syntax: LOCATION=*loc*

Description: Location of Calling number to apply hunt list plan.

Format: *MccMncLacCellid*

where:

- Mcc: A 3-digit country code
- Mnc: A 2 or 3-digit network code (starting with 0)
- Lac: A 5-digit Location code with decimal value (starting with 0), and
- Cellid: A 5-digit Cell ID with decimal value (starting with 0).

Example: LOCATION="530020012304567"

CLI

Syntax: CLI=*msisdn*

Description: CLI, or range of CLIs to apply hunt list plan.

Format: MSISDN number, or colon separated list of MSISDN numbers.

Example: CLI=649049393461

Logic and constraints

- The HUNTLIST must already exist.
- If specified, NETNAME must exist. It is used in conjuction with STNNAME and PHYSADDR to determine a unique station.
- Either or both STNNAME and PHYSADDR must be specified, exist, and define a unique station.
- When the first VPNST6=ADD command is used the specified HUNTLIST will become the default for the station.
- The default of 'Hunt Unconditionally' will be set.
- TIMERANGE must be one of TOD, DOW, DOY
- If TOD is specified STARTTIME and ENDTIME must be in the format HH:MM.
- If DOW is specified STARTTIME and ENDTIME must be in the format DDD:HH:MM where DDD is MON, TUE etc.
- If DOY is specified STARTTIME and ENDTIME must be in the format DD:MON:HH:MM where DD is 01-31, MON is JAN, FEB etc.

Success return

```
VPNST6=ADD:ACK;
```

Error codes

Error codes: 1, 2, 4, 5, 6, 7, 8, 17, 19, 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Change a VPN Station Hunting List Plan

Name

VPNST6=CHG

Description

Changes the details associated with a VPN station Hunt Plan.

Required parameters

This command has no required parameters.

Constraint required parameters

This command accepts the following constraint required parameter.

STNNAME

Syntax: STNNAME=*station name*
Description: The name of VPN Station. (Extension number field in VPN Screen).
Format: Extension number.
Example: STNNAME=123

PHYSADDR

Syntax: PHYSADDR=*msisdn*
Description: The Physical Address, usually real MSISDN.
Format: MSISDN number.
Example: PHYSADDR=6494111111

LOCATION

Syntax: LOCATION=*loc*
Description: Location of Calling number to apply hunt list plan.
Format: *MccMncLacCellid*
where:
- Mcc: A 3-digit country code
- Mnc: A 2 or 3-digit network code (starting with 0)
- Lac: A 5-digit Location code with decimal value (starting with 0), and
- Cellid: A 5-digit Cell ID with decimal value (starting with 0).

Example: LOCATION="530020012304567"

CLI

Syntax: CLI=*msisdn*
Description: CLI, or range of CLIs to apply hunt list plan.

Format: MSISDN number, or colon separated list of MSISDN numbers.
Example: `CLI=649049393461`

HUNTLIST

Syntax: `HUNTLIST=`*`name`*
Description: The name of the Hunt list.
Format: String
Example: `HUNTLIST="HL1"`

TIMERANGE

Syntax: `TIMERANGE=`*`type`*
Description: The time range type.
Format: Type is:

- TOD (Time of day)
- DOW (Day of week)
- DOY (Day of year)

Example: `TIMERANGE=TOD`

STARTTIME

Syntax: `STARTTIME=`*`time`*
Description: The start time to use HUNTLIST.
Format: If TIMERANGE=

- TOD: HH:MM
- DOW: DDD:HH:MM
- DOY: MON:DD:HH:MM

HH =	hours 00 to 23
MM =	minutes 00 to 59
DDD =	day prefix SUN,MON,TUE,WED,THU,FRI,SAT
DD =	day of month 01 to 31
MON =	month prefix JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC

Example: `STARTTIME=23:11`

ENDTIME

Syntax: `ENDTIME=`*`time`*
Description: The end time to use HUNTLIST.

Format: If TIMERANGE=

- TOD: HH:MM
- DOW: DDD:HH:MM
- DOY: MON:DD:HH:MM

HH = hours 00 to 23
MM = minutes 00 to 59
DDD = day prefix SUN,MON,TUE,WED,THU,FRI,SAT
DD = day of month 01 to 31
MON = month prefix JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC

Example: `ENDTIME=08:00`

Optional parameters

This command accepts the following optional parameters.

NETNAME

Syntax: `NETNAME=network name`
Description: The name of the VPN Network.
Format: String
Example: `NETNAME="Net1"`

NEW_LOCATION

Syntax: `NEW_LOCATION="loc"`
Description: Location of Calling number to apply hunt list plan.
Format: string
Example: `NEW_LOCATION="Ipswich office"`

NEW_CLI

Syntax: `NEW_CLI=msisdn`
Description: CLI, or range of CLIs to apply hunt list plan.
Format: MSISDN number, or colon separated list of MSISDN numbers.
Example: `NEW_CLI=`

NEW_HUNTLIST

Syntax: `NEW_HUNTLIST=name`
Description: The name of the Hunt list.
Format: String
Example: `NEW_HUNTLIST="HL2"`

NEW_TIMERANGE

Syntax: `NEW_TIMERANGE=type`
Description: The time range type.

Format: Type is:
- TOD (Time of day)
- DOW (Day of week)
- DOY (Day of year)

Example: `NEW_TIMERANGE=TOD`

`NEW_STARTTIME`

Syntax: `NEW_STARTTIME=time`

Description: The start time to use HUNTLIST.

Format: If TIMERANGE=
- TOD: HH:MM
- DOW: DDD:HH:MM
- DOY: MON:DD:HH:MM

HH = hours 00 to 23

MM = minutes 00 to 59

DDD = day prefix SUN,MON,TUE,WED,THU,FRI,SAT

DD = day of month 01 to 31

MON = month prefix JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC

Example: `NEW_STARTTIME=22:00`

`NEW_ENDTIME`

90Syntax: `NEW_ENDTIME=time`

Description: The end time to use HUNTLIST.

Format: If TIMERANGE=
- TOD: HH:MM
- DOW: DDD:HH:MM
- DOY: MON:DD:HH:MM

HH = hours 00 to 23

MM = minutes 00 to 59

DDD = day prefix SUN,MON,TUE,WED,THU,FRI,SAT

DD = day of month 01 to 31

MON = month prefix JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC

Example: `NEW_ENDTIME=08:00`

Logic and constraints

- The HUNTLIST must already exist.
- If specified, NETNAME must exist. It is used in conjunction with STNNAME and PHYSADDR to determine a unique station.
- Either or both STNNAME and PHYSADDR must be specified, exist, and define a unique station
- TIMERANGE and NEW_TIMERANGE must be one of TOD, DOW, DOY
- If TOD is specified STARTTIME, ENDTIME NEW_STARTTIME, NEW_ENDTIME must be in the format HH:MM

- If DOW is specified STARTTIME, ENDTIME NEW_STARTTIME, NEW_ENDTIME must be in the format DDD:HH:MM where DDD is MON, TUE etc
- If DOY is specified STARTTIME, ENDTIME NEW_STARTTIME, NEW_ENDTIME must be in the format DD:MON:HH:MM where DD is 01-31, MON is JAN, FEB etc.
- If NEW_LOCATION is specified then LOCATION must be specified and exist
- If NEW_CLI is specified then CLI must be specified
- If NEW_HUNTLIST is specified then HUNTLIST must be specified
- If NEW_STARTTIME is specified then STARTTIME must be specified
- If NEW_ENDTIME is specified then ENDTIME must be specified
- The resulting combination of LOCATION, CLI, HUNTLIST, STARTTIME and ENDTIME exists and must be a unique match

Success return

```
VPNST6=CHG:ACK;
```

Error codes

Error codes: 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 19, 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Delete a VPN Station Hunting List Plan

Name

VPNST6=DEL

Description

Delete details of a VPN Station hunting plan.

Required parameters

This command has no required parameters.

Constraint required parameters

This command accepts the following constraint required parameter.

STNNAME

Syntax: STNNAME=*station name*
Description: The name of VPN Station. (Extension number field in VPN Screen).
Format: Extension number.
Example: STNNAME=123

PHYSADDR

Syntax: PHYSADDR=*msisdn*
Description: The Physical Address, usually real MSISDN.
Format: MSISDN number.
Example: PHYSADDR=6494111111

LOCATION

Syntax: LOCATION=*loc*

Description: Location of Calling number to apply hunt list plan.

Format: *MccMncLacCellid*

where:

- Mcc: A 3-digit country code
- Mnc: A 2 or 3-digit network code (starting with 0)
- Lac: A 5-digit Location code with decimal value (starting with 0), and
- Cellid: A 5-digit Cell ID with decimal value (starting with 0).

Example: LOCATION="530020012304567"

CLI

Syntax: CLI=*msisdn*

Description: CLI, or range of CLIs to apply hunt list plan.

Format: MSISDN number, or colon separated list of MSISDN numbers.

Example: CLI=649049393461

TIMERANGE

Syntax: TIMERANGE=*type*

Description: The time range type.

Format: Type is:

- TOD (Time of day)
- DOW (Day of week)
- DOY (Day of year)

Example: TIMERANGE=TOD

STARTTIME

Syntax: STARTTIME=*time*

Description: The start time to use HUNTLIST.

Format: If TIMERANGE=

- TOD: HH:MM
- DOW: DDD:HH:MM
- DOY: MON:DD:HH:MM

HH =	hours 00 to 23
MM =	minutes 00 to 59
DDD =	day prefix SUN,MON,TUE,WED,THU,FRI,SAT
DD =	day of month 01 to 31
MON =	month prefix JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC

Example: STARTTIME=23:11

ENDTIME

Syntax: ENDTIME=*time*

Description: The end time to use HUNTLIST.

Format: If TIMERANGE=

- TOD: HH:MM
- DOW: DDD:HH:MM
- DOY: MON:DD:HH:MM

HH = hours 00 to 23

MM = minutes 00 to 59

DDD = day prefix SUN,MON,TUE,WED,THU,FRI,SAT

DD = day of month 01 to 31

MON = month prefix JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC

Example: `ENDTIME=08:00`

Optional parameter

This command accepts the following optional parameter.

NETNAME

Syntax: `NETNAME=`*`network name`*

Description: The name of the VPN Network.

Format: String

Example: `NETNAME="Net1"`

Logic and constraints

- If specified, NETNAME must exist. It is used in conjunction with STNNAME and PHYSADDR to determine a unique station.
- Either or both STNNAME and PHYSADDR must be specified, exist, and define a unique station
- If deleting a complete Hunt Plan then just the STNNAME and/or PHYSADDR are required
- If deleting a Hunt Plan entry then at least one of the rest of optional parameters must be specified.
- If TOD is specified STARTTIME and ENDTIME must be in the format HH:MM
- If DOW is specified STARTTIME and ENDTIME must be in the format DDD:HH:MM where DDD is MON, TUE etc
- If DOY is specified STARTTIME and ENDTIME must be in the format DD:MON:HH:MM where DD is 01-31, MON is JAN, FEB etc.

Success return

```
VPNST6=DEL:ACK;
```

Error codes

Error codes: 1, 4, 5, 6, 7, 13, 15, 16, 17, 19, 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Query a VPN Station Hunting List Plan

Name

VPNST6=QRY

Description

Query a VPN Station's Hunting Plan details.

Required parameters

This command has no required parameters.

Constraint required parameters

This command accepts the following constraint required parameter.

STNNAME

Syntax: STNNAME=*station name*
Description: The name of VPN Station. (Extension number field in VPN Screen).
Format: Extension number.
Example: STNNAME=123

PHYSADDR

Syntax: PHYSADDR=*msisdn*
Description: The Physical Address, usually real MSISDN.
Format: MSISDN number.
Example: PHYSADDR=6494111111

Optional parameter

This command accepts the following optional parameter.

NETNAME

Syntax: NETNAME=*network name*
Description: The name of the VPN Network.
Format: String
Example: NETNAME="Net1"

Logic and constraints

- If specified, NETNAME must exist. It is used in conjuction with STNNAME and PHYSADDR to determine a unique station.
- Either or both STNNAME and PHYSADDR must be specified, exist, and define a unique station

Success return

VPNST6=QRY:ACK:*hunting plan1|hunting plan 2|...;*

For each hunting plan, the following is returned, comma separated:

- LOCATION=*location*
- CLI=*cli*
- TIMERANGE=*timerange type (TOD, DOW, DOY)*
- STARTTIME=*start time*

- ENDTIME=*end time*
- HUNTLIST=*name*

STARTTIME, ENDTIME are formatted for TIMERANGE as:

- TOD: Format HH:MM
- DOW: Format DDD:HH:MM
- DOY: Format DD:MON:HH:MM

where:

- DD = day of month 01 to 31
- DDD = day of week (SUN, MON, TUE, WED, THU, FRI, SAT)
- MON = short month (JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC)
- HH = 24 hour clock hours 00 to 23
- MM = minute 00 to 59

Error codes

Error codes: 1, 7, 15, 16, 17, 19, 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Chapter 9

Tariff

Overview

Introduction

This chapter explains the VPN PI commands for provisioning VPN tariffs.

In this chapter

This chapter contains the following topics.

Add a VPN Wide Default Tariff

Name

VPNCU1=ADD

Description

Adds a new VPN wide default ACS Tariff to use if there is no Network or Station default defined.

Required Parameter

Here is the required parameter for this command.

TARIFFNAME

Syntax: TARIFFNAME=*tariff name*
Description: Default ACS tariff name to use.
Format: String
Note: Matches ACS_TARIFF_CODE
Example: TARIFFNAME="TR1"

Constraint required parameters

There are no constraint required parameters.

Optional parameter

This command accepts the following optional parameter.

PROVIDER

Syntax: PROVIDER=*sp*
Description: Name of ACS service provider.

Format: String
Example: PROVIDER="Boss"

Logic and constraints

The following rules apply when using the VPNCU1=ADD command:

- TARIFFNAME must exist.
- If PROVIDER is defined a single Service Provider will be updated, if it is not specified ALL service providers will be updated.
- If a VPN wide default tariff is already defined this will over write the existing setting.

Success return

VPNCU1=ADD:ACK:TARIFFCODE=*acs_tariff.tariff_code*;

Error codes

Error Codes: 1, 2, 19, 20

See *PI Command Errors* (on page 70) for a description of error codes.

Change the Tariff for all VPN Stations in a Network

Name

VPNST7=CHG

Description

Changes the ACS tariff for all VPN stations associated with a specified VPN network.

Required Parameters

Here are the required parameters for this command.

NAME

Syntax: NAME=*name*
Description: Name of VPN Network.
Format: String
Example: NAME="VPNNW1"

NEW_TARIFF

Syntax: NEW_TARIFF=*tname*
Description: Name of new ACS Tariff for all stations in a network.
Format: String
Example: NEW_TARIFF="TR1"

Constraint required parameters

There are no constraint required parameters.

Optional parameters

There are no optional parameters.

Logic and constraints

- NAME and NEW_TARIFF must exist.

Success return

```
VPNST7=CHG:ACK[:number of stations changed];
```

Error codes

Error codes: 1, 2, 18, 19, 20.

See *PI Command Errors* (on page 70) for a description of error codes.

Chapter 10

Error Code Lists

Overview

Introduction

This chapter explains the error codes for Oracle Communications Convergent Charging Controller provisioning interface (PI) commands.

In this chapter

This chapter contains the following topics.

PI Chassis Errors

Format

Command:NACK:*error code-error* m*essage*;

Example: `VPNST7=CHG:NACK:91-TIMEOUT`

Error List

This table describes the PI Chassis error codes.

Code	Message	Description
70	`TOO MANY SESSIONS`	All PI sessions are in use.
71	`LOGON SYNTAX ERROR`	The login string was incorrectly formatted.
72	`INVALID LOGON - username, password`	Invalid username and/or password
73	`INVALID LOGON - user not allowed on this port`	The user attempted to log in to the wrong PI port.
74	`INVALID LOGON - host`	The PI client is unknown.
75	`UNKNOWN COMMAND`	Client sent an unknown command.
76	`USER DOES NOT HAVE SUFFICIENT SECURITY`	The user's security level is less than the command's security level.
77	`SYNSTAMP NOT FOUND`	Synstamps are turned on, but the client did not send one.
78	`SYNSTAMP NOT VALID`	Synstamps are turned on, but the synstamp sent by the client is invalid.

Code	Message	Description
79	INVALID OR MISSING CHECKSUM	Checksums are turned on, but the client is one of the following: • Did not send one • It was invalid
80	UNKNOWN PARAMETER FOR COMMAND	A parameter was sent that was not valid for this command.
81	MISSING PARAMETERS FROM COMMAND	A required parameter is missing.
82		Undefined
83	DUPLICATE PARAMETER	The client sent two identically named parameters.
84	ERROR RUNNING PROCEDURE	An internal error occurred running the command.
85	USER SESSION TERMINATED	The user's session has been terminated by an administrator.
86	COMMAND TOO BIG	The command sent is too long. Indicates an incorrectly formatted command.
87	COMMAND SYNTAX ERROR	The command sent is incorrectly formatted.
88	PARAMETER NAME TOO BIG	A parameter name is too long. Indicates the command was incorrectly formatted.
89	PARAMETER VALUE TOO BIG	A parameter value is too long. Indicates the command was incorrectly formatted.
90	SYNSTAMP OUT OF PLACE	The synstamp is not at the end of the command, but before the checksum.
91	TIMEOUT	The command took too long to run.

PI Command Errors

Format

Command:NACK:*error code-error message*;

Example: `VPNST6=DEL:NACK:17-Station selection is not unique`

Error list

This table describes the PI Command error codes.

Code	Message	Commands
1	TARIFFNAME *name* does not exist	VPNCU1=ADD
1	NAME *name* already exists in vpn_network table	VPNNW1=ADD
1	NAME *name* does not exist	VPNNW1=CHG VPNST7=CHG
1	NAME *name* does not exist in vpn_network table	VPNNW1=DEL
1	No Network exists with prefix of NAME *name* in vpn_network table	VPNNW1=QRY

Code	Message	Commands
1	NAME *name* is not valid	VPNNW2=ADD VPNNW2=DEL VPNNW2=QRY VPNNW3=ADD VPNNW3=CHG VPNNW3=DEL VPNNW3=QRY
1	Station defined by NETNAME *name* STNNAME *name* PHYSADDR *address* already exists	VPNST1=ADD
1	Station defined by [NETNAME *name*][STNNAME *name*][PHYSADDR *address*] does not exist (dependent on the combination of NETNAME, STNNAME and PHYSADDR specified).	VPNST1=CHG VPNST1=DEL VPNST1=QRY VPNST2=ADD VPNST2=CHG VPNST2=DEL VPNST2=QRY VPNST5=ADD VPNST5=CHG VPNST5=DEL VPNST5=QRY VPNST6=ADD VPNST6=CHG VPNST6=DEL VPNST6=QRY
2	PROVIDER *name* does not exist	VPNCU1=ADD
2	PROVIDER *name* is not valid	VPNNW1=ADD
2	NEW_NAME *name* already exists	VPNNW1=CHG
2	LISTTYPE *name* is not SHORT, LONG, DEFTARIFF, TARIFF or STATION	VPNNW1=QRY
2	NUMBER *number* already exists for NAME *name*	VPNNW2=ADD VPNNW3=ADD
2	NUMBER [*number*]does not exist for NAME *name*	VPNNW2=DEL VPNNW3=DEL
2	Blacklist is active	VPNNW2=QRY
2	NUMBER *number* does not exist	VPNNW2=CHG
2	NETNAME *name* is not valid	VPNST1=ADD
2	NEW_STNNAME *name* already exists in NEW_NAME *name* network	VPNST1=CHG
2	LISTTYPE *name* is not one of GROUP, BASIC, ALL	VPNST1=QRY
2	OFFNET is invalid (not Y or N)	VPNST2=CHG
2	HUNTLIST *name* does not exist	VPNST5=CHG VPNST5=DEL VPNST5=QRY VPNST6=ADD VPNST6=CHG
2	NEW_TARIFF *name* does not exist	VPNST7=CHG
3	ORIGINATING Call Plan *name* does not exist in acs_call_plan table	VPNNW1=ADD

Code	Message	Commands
3	PROVIDER *name* does not exist	VPNNW1=CHG
3	No default network tariff defined	VPNNW1=QRY
3	Blacklist is active	VPNNW2=DEL
3	NEW_NUMBER *number* already exists	VPNNW3=CHG
3	STNNAME *name* length is invalid (not Y or N)	VPNST1=ADD
3	NEW_NAME *name* does not exist	VPNST1=CHG
3	STNNAME not specified with PHYSADDRMASK	VPNST1=QRY
3	WHITELIST already exists	VPNST2=ADD
3	NEW_OFFNET is invalid (not Y or N)	VPNST2=CHG
3	WHITELIST does not exist for STNNAME *name* and PHYSADDR *name* pair	VPNST2=DEL
3	RANK *rank* out of range	VPNST5=ADD
3	RANK *rank* is not in a valid format	VPNST5=CHG
3	TERMNUM *number* does not exist	VPNST5=DEL
4	TERMINATING Call Plan *name* does not exist in acs_call_plan table	VPNNW1=ADD
4	ORIGINATING Call Plan *name* does not exist	VPNNW1=CHG
4	One of NAME or LISTTYPE must be specified	VPNNW1=QRY
4	OFFNET is invalid (not Y or N)	VPNST1=ADD
4	GVNSADDR *name* does not exist	VPNST1=CHG
4	LISTTYPE specified with PHYSADDRMASK but LISTTYPE value is not BASIC	VPNST1=QRY
4	INOFFNET is invalid (not Y or N)	VPNST2=CHG
4	TERMNUM *number* is invalid	VPNST5=ADD
4	TIMEOUT *timeout* is not in a valid format	VPNST5=CHG
4	HUNTLIST *name* in use	VPNST5=DEL
4	TIMERANGE *name* is not valid (not one of TOD, DOW, DOY)	VPNST6=ADD VPNST6=CHG VPNST6=DEL
5	MANAGEMENT Call Plan *name* does not exist in acs_call_plan table	VPNNW1=ADD
5	TERMINATING Call Plan *name* does not exist	VPNNW1=CHG
5	NEW_INOFFNET is invalid (not Y or N)	VPNST1=ADD
5	PHYSADDR *name* does not exist	VPNST1=CHG
5	LISTTYPE specified as BASIC with no PHYSADDRMASK	VPNST1=QRY
5	NEW_INOFFNET is invalid (not Y or N)	VPNST2=CHG
5	TIMEOUT *value* is invalid	VPNST5=ADD
5	ONNET is not Y or N	VPNST5=CHG
5	STARTIME *time* is not in a valid format	VPNST6=ADD
5	STARTIME *time* is not valid	VPNST6=CHG VPNST6=DEL
6	PHYSRANGE value not Y or N	VPNNW1=ADD

Code	Message	Commands
6	PHYSADDR *name* does not exist in a vpn_network_physical_range:start_number to end_number range	VPNST1=ADD
6	MANAGEMENT Call Plan *name* does not exist	VPNNW1=CHG
6	VDDIADDR *name* does not exist	VPNST1=CHG
6	WHITELIST number does not exist for STNNAME *name* and PHYSADDR *name* pair	VPNST2=CHG
6	ONNET is not Y or N	VPNST5=ADD
6	NEW_RANK *rank* is not valid	VPNST5=CHG
6	ENDTIME *time* is not in a valid format	VPNST6=ADD
6	ENDTIME *time* is not valid	VPNST6=CHG VPNST6=DEL
7	GNVSADDR *name* does not exist in a vpn_network_physical_range:start_number to end_number range	VPNST1=ADD
7	PRESENTONNETADDR value not Y or N	VPNNW1=ADD
7	PHYSRANGE value not Y or N	VPNNW1=CHG
7	TARIFFNAME *name* does not exist in ACS_TARIFF table	VPNST1=CHG
7	STNNAME and PHYSADDR both not specified	VPNST2=ADD VPNST2=CHG VPNST2=DEL VPNST2=QRY VPNST6=ADD VPNST6=DEL VPNST6=QRY
7	NEW_TIMEOUT *timeout* is not valid	VPNST5=CHG
8	RESTRICTCLI value not Y or N	VPNNW1=ADD
8	PRESENTONNETADDR value not Y or N	VPNNW1=CHG
8	VDDIADDR *name* does not exist in a vpn_network_physical_range:start_number to end_number range	VPNST1=ADD
8	NEW_PHYSADDR *name* does not exist in a vpn_network_physical_range:start_number to end_number range or is not unique or is invalid	VPNST1=CHG
8	OFFNET and NEW_OFFNET pair not provided	VPNST2=CHG
8	NEW_ONNET is not Y or N	VPNST5=CHG
8	STARTTIME-ENDTIME *time-time* overlaps an existing plan range	VPNST6=ADD VPNST6=CHG
9	EXTLENGTH not valid - *reason*	VPNNW1=ADD
9	RESTRICTCLI value not Y or N	VPNNW1=CHG
9	TARIFFNAME *name* not valid	VPNST1=ADD
9	NEW_STNNAME *name* is invalid	VPNST1=CHG
9	INOFFNET and NEW_INOFFNET pair not provided	VPNST2=CHG
9	TERMNUM *number* is not valid station number	VPNST5=ADD

Code	Message	Commands
9	Cannot change ONNET to Y: A termination number in the list is not a VPN number	VPNST5=CHG
9	NEW_HUNTLIST *name* does not exist	VPNST6=CHG
10	ALLOWSHORTEXT value not Y or N	VPNNW1=ADD
10	TARIFFNAME *name* does not exist	VPNNW1=CHG
10	PHYSADDR *address* is already in use	VPNST1=ADD
10	STNNAME *name* does not exist	VPNST1=CHG
10	WHITELIST and NEW_WHITELIST pair not provided	VPNST2=CHG
10	STNNAME and PHYSADDR both not specified	VPNST5=ADD VPNST5=CHG VPNST5=DEL VPNST5=QRY
10	NEW_TIMERANGE *name* is not valid (not one of TOD, DOW, DOY)	VPNST6=CHG
11	TARIFFNAME *name* does not exist	VPNNW1=ADD
11	TARIFFNAME *name* not valid for PROVIDER *name*	VPNNW1=CHG
11	WHITELIST does not exist for STNNAME *name* and PHYSADDR *name* pair	VPNST1=ADD
11	NEW_RANK specified without RANK	VPNST5=CHG
11	NEW_STARTTIME *time* is not valid	VPNST6=CHG
12	TARIFFNAME *name* not valid for PROVIDER *name*	VPNNW1=ADD
12	NEW_TIMEOUT specified without TIMEOUT	VPNST5=CHG
12	NEW_ENDTIME *time* is not valid	VPNST6=CHG
13	SITE_CODE cannot be a sub or super string of an existing SITE_CODE	VPNNW1=ADD VPNNW1=CHG
13	NEW_ONNET specified without ONNET	VPNST5=CHG
14	The resulting combination of LOCATION, CLI, STARTTIME and ENDTIME is not a unique match	VPNST6=CHG
15	The selected station does not have an associated hunting plan	VPNST6=CHG VPNST6=DEL VPNST6=QRY
16	The selected station does not have an associated hunting plan table	VPNST6=CHG VPNST6=DEL VPNST6=QRY
17	Station selection is not unique	VPNST1=ADD VPNST1=CHG VPNST1=DEL VPNST2=ADD VPNST2=CHG VPNST2=DEL VPNST2=QRY VPNST5=ADD VPNST5=CHG VPNST5=DEL VPNST5=QRY VPNST6=ADD VPNST6=CHG VPNST6=DEL

Code	Message	Commands
		VPNST6=QRY
18	Nothing to change	VPNNW1=CHG VPNNW3=CHG VPNST1=CHG VPNST2=CHG
19	Badly formatted parameter *parameter*	All commands
20	*General error*	All commands
21	TAG *tag name* does not exist	VPNNW1=ADD VPNNW1=CHG VPNNW1=QRY VPNST1=ADD VPNST1=CHG VPNST1=QRY
22	VALUE *value* is not valid	VPNNW1=ADD VPNNW1=CHG VPNST1=ADD VPNST1=CHG
23	Wrong number of values for the number of tags	VPNNW1=ADD VPNNW1=CHG VPNST1=ADD VPNST1=CHG

Glossary of Terms

ACS

Advanced Control Services configuration platform.

ANI

Automatic Number Identification - Term used in the USA by long-distance carriers for CLI.

CC

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

CLI

Calling Line Identification - the telephone number of the caller. Also referred to as ANI.

Convergent

Also “convergent billing”. Describes the scenario where post-paid and pre-paid calls are handed by the same service platform and the same billing system. Under strict converged billing, post-paid subscribers are essentially treated as “limited credit pre-paid”.

cron

Unix utility for scheduling tasks.

DTMF

Dual Tone Multi-Frequency - system used by touch tone telephones where one high and one low frequency, or tone, is assigned to each touch tone button on the phone.

GUI

Graphical User Interface

GVNS

Global Virtual Numbering Scheme - When multiple VPNs are in use by a customer, the capability to route calls between these VPNs requires a numbering scheme that uses destination addresses based on a customer id and extension number. These GVNS addresses can then be interpreted to provide inter VPN operation.

Hunting

A terminating call feature where a subscriber may request a list of alternate destination addresses. If their mobile station is not attached, or does not answer a call, then the service logic should attempt to reach the supplied alternate destinations in sequence.

IN

Intelligent Network

IP

1) Internet Protocol

2) Intelligent Peripheral - This is a node in an Intelligent Network containing a Specialized Resource Function (SRF).

ISDN

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

Messaging Manager

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

MM

Messaging Manager. Formerly MMX, see also *XMS* (on page 79) and *Messaging Manager* (on page 78).

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).

PI

Provisioning Interface - used for bulk database updates/configuration instead of GUI based configuration.

Service Provider

See Telco.

SLC

Service Logic Controller (formerly UAS).

SMS

Depending on context, can be:

- Service Management System hardware platform
- Short Message Service
- Service Management System platform
- Convergent Charging Controller Service Management System application

SN

Service Number

SQL

Structured Query Language is a database query language.

SRF

Specialized Resource Function – This is a node on an IN which can connect to both the SSP and the SLC and delivers additional special resources into the call, mostly related to voice data, for example play voice announcements or collect DTMF tones from the user. Can be present on an SSP or an Intelligent Peripheral (IP).

SSP

Service Switching Point

TCP

Transmission Control Protocol. This is a reliable octet streaming protocol used by the majority of applications on the Internet. It provides a connection-oriented, full-duplex, point to point service between hosts.

Telco

Telecommunications Provider. This is the company that provides the telephone service to customers.

Telecommunications Provider

See Telco.

VDDI

Virtual Direct Dial In

VPN

The Virtual Private Network product is an enhanced services capability enabling private network facilities across a public telephony network.

XMS

Three letter code used to designate some components and path locations used by the Oracle Communications Convergent Charging Controller *Messaging Manager* (on page 78) service and the Short Message Service. The published code is *MM* (on page 78) (formerly MMX).

Index

O

P

Q

R

S

T

V

W

X