

# Oracle® Communications Convergent Charging Controller VPN Provisioning Interface Commands



Release 15.0.1

June 2024

The Oracle logo, consisting of the word "ORACLE" in white, uppercase, sans-serif font, centered within a solid red square.

# Copyright

Copyright © 2024, Oracle and/or its affiliates.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software" or "commercial computer software documentation" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

# Contents

|                                                |           |
|------------------------------------------------|-----------|
| About This Document .....                      | v         |
| Document Conventions .....                     | vi        |
| <b>Chapter 1</b>                               |           |
| <b>PI Commands Overview .....</b>              | <b>1</b>  |
| Overview .....                                 | 1         |
| Command List .....                             | 1         |
| About Service Provider Restrictions .....      | 2         |
| <b>Chapter 2</b>                               |           |
| <b>VPN Network.....</b>                        | <b>3</b>  |
| Overview .....                                 | 3         |
| Add a Network .....                            | 3         |
| Change a VPN Network Details .....             | 7         |
| Delete a VPN Network.....                      | 11        |
| Query a VPN Network .....                      | 11        |
| <b>Chapter 3</b>                               |           |
| <b>Network White List Numbers.....</b>         | <b>15</b> |
| Overview .....                                 | 15        |
| Add Network White List Numbers .....           | 15        |
| Delete Network White List Numbers .....        | 16        |
| Query Network White List Numbers .....         | 17        |
| <b>Chapter 4</b>                               |           |
| <b>Network Speed Dial Number .....</b>         | <b>19</b> |
| Overview .....                                 | 19        |
| Add a Network Speed Dial Number .....          | 19        |
| Change a Network Speed Dial Number .....       | 20        |
| Delete a Network Speed Dial Number .....       | 21        |
| Query Network Speed Dial Numbers .....         | 22        |
| <b>Chapter 5</b>                               |           |
| <b>VPN Station .....</b>                       | <b>25</b> |
| Overview .....                                 | 25        |
| Add a VPN Station.....                         | 25        |
| Change a VPN Station Details .....             | 28        |
| Delete a VPN Station.....                      | 31        |
| Query a VPN Station .....                      | 32        |
| <b>Chapter 6</b>                               |           |
| <b>VPN Station White List.....</b>             | <b>37</b> |
| Overview .....                                 | 37        |
| Add a VPN Station White List.....              | 37        |
| Change a VPN Station Restriction Details ..... | 38        |

|                                      |    |
|--------------------------------------|----|
| Delete a VPN Station White List..... | 40 |
| Query a Station White List.....      | 42 |

## Chapter 7

### **VPN Station Hunting List..... 45**

|                                         |    |
|-----------------------------------------|----|
| Overview.....                           | 45 |
| Add a VPN Station Hunting List .....    | 45 |
| Change a VPN Station Hunting List ..... | 47 |
| Delete a VPN Station Hunting List ..... | 49 |
| Query a VPN Station Hunting List .....  | 51 |

## Chapter 8

### **VPN Hunting List Plan ..... 53**

|                                              |    |
|----------------------------------------------|----|
| Overview.....                                | 53 |
| Add a VPN Station Hunting List Plan .....    | 53 |
| Change a VPN Station Hunting List Plan ..... | 56 |
| Delete a VPN Station Hunting List Plan ..... | 60 |
| Query a VPN Station Hunting List Plan.....   | 62 |

## Chapter 9

### **Tariff ..... 65**

|                                                           |    |
|-----------------------------------------------------------|----|
| Overview.....                                             | 65 |
| Add a VPN Wide Default Tariff.....                        | 65 |
| Change the Tariff for all VPN Stations in a Network ..... | 66 |

## Chapter 10

### **Error Code Lists..... 69**

|                        |    |
|------------------------|----|
| Overview.....          | 69 |
| PI Chassis Errors..... | 69 |
| PI Command Errors..... | 70 |

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

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

# 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 .....                        | 1 |
| About Service Provider Restrictions ..... | 2 |

## 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.

# 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 .....                | 3  |
| Change a VPN Network Details ..... | 7  |
| Delete a VPN Network .....         | 11 |
| Query a VPN Network .....          | 11 |

## Add a Network

### Name

VPNNW1=ADD

### Description

Adds a new VPN Network definition.

### Required Parameters

Here are the required parameters for this command.

NAME

|                     |                      |
|---------------------|----------------------|
| <b>Syntax:</b>      | NAME= <i>name</i>    |
| <b>Description:</b> | Name of VPN Network. |
| <b>Format:</b>      | String               |
| <b>Example:</b>     | NAME="VPNNW1"        |

PROVIDER

|                     |                               |
|---------------------|-------------------------------|
| <b>Syntax:</b>      | PROVIDER= <i>sp</i>           |
| <b>Description:</b> | Name of ACS service provider. |
| <b>Format:</b>      | String                        |
| <b>Example:</b>     | PROVIDER="Boss"               |

ORIGINATING

|                     |                            |
|---------------------|----------------------------|
| <b>Syntax:</b>      | ORIGINATING= <i>origcp</i> |
| <b>Description:</b> | 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:** ALLOWSHORTEXT=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.<br>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.<br>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"

## Chapter 2

### 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.

#### PTMODE

**Syntax:** PTMODE=*ptupdatemode*

**Description:** Specifies the type of operation to be performed on an existing prefix tree.

**Type:** String

**Optionality:** Optional

**Allowed:** ADD: Adds profile values to an existing prefix tree. Duplicate entries are ignored.  
DEL: Removes the profile values from an existing prefix tree. Missing entries are ignored.

**Notes:** Updates the prefix tree specified in the TAG field with the profile values specified in the VALUE field.

**Example:** PTMODE=ADD

#### 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<br>OPREFIX, | the prefix tree data should be specified separated by the colon character.<br><br>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.<br><br>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

VPN1=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

|                     |                      |
|---------------------|----------------------|
| <b>Syntax:</b>      | NAME= <i>name</i>    |
| <b>Description:</b> | Name of VPN Network. |
| <b>Format:</b>      | String               |
| <b>Example:</b>     | NAME="VPN1 "         |

### Constraint required parameters

There are no constraint required parameters.

### Optional parameters

There are no optional parameters.

### Logic and constraints

- NAME must exist.

### Success return

VPN1=DEL:ACK; if the network has stations defined.

VPN1=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

VPN1=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

OFFSET

**Syntax:** OFFSET=*value*  
**Description:** OFFSET can be used to switch the pages which prints 50 records of network-station details at a time. When OFFSET value is incremented, next set of 50 records are printed and so on.  
**Format:** Integer  
**Optionality:** Optional (default used if not set).  
**Default:** 0  
**Notes:** OFFSET can be used with or without NETWORK name in the request parameter.  
**Example:** OFFSET=0

## Optional parameter

This command accepts the following optional parameter.

## TAG

|                     |                                                                                      |
|---------------------|--------------------------------------------------------------------------------------|
| <b>Syntax:</b>      | <code>TAG=tag</code>                                                                 |
| <b>Description:</b> | Profile tag(s) to alter.                                                             |
| <b>Format:</b>      | Profile tag name(s).                                                                 |
| <b>Allowed:</b>     | Pre-defined name corresponding to PROFILE_TAG_NAME in the ACS_PROFILE_DETAILS table. |
| <b>Notes:</b>       | Multiple tags should be separated by the pipe symbol ' '.<br>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.
- OFFSET can be used to print 50 records of network-station details at a time. For network without station, only network names will be printed. If station exists for the network, details for each station will be printed irrespective of whether it belongs to same network or not until it reaches the count. To indicate which station belongs to which network, network name is also printed along with station.
  - If the offset value is set to negative, "VPN offset must be positive" message is returned.
  - If the offset value crosses the number of available records, "OFFSET out of bound" message is 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

```
VPNW1=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.

# 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 .....    | 15 |
| Delete Network White List Numbers ..... | 16 |
| Query Network White List Numbers .....  | 17 |

## 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

|                     |                      |
|---------------------|----------------------|
| <b>Syntax:</b>      | NAME= <i>name</i>    |
| <b>Description:</b> | Name of VPN Network. |
| <b>Format:</b>      | String               |
| <b>Example:</b>     | NAME="VPNNW1 "       |

NUMBER

|                     |                                                   |
|---------------------|---------------------------------------------------|
| <b>Syntax:</b>      | NUMBER= <i>num</i>                                |
| <b>Description:</b> | Number(s) to add to, or delete from, list.        |
| <b>Format:</b>      | Single number or colon separated list of numbers. |
| <b>Example:</b>     | 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

```
VPN2=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

VPN2=DEL

### Description

Deletes white list numbers from a specified VPN Network.

### Required Parameters

Here are the required parameters for this command.

NAME

|                     |                      |
|---------------------|----------------------|
| <b>Syntax:</b>      | NAME= <i>name</i>    |
| <b>Description:</b> | Name of VPN Network. |
| <b>Format:</b>      | String               |
| <b>Example:</b>     | NAME="VPN1 "         |

NUMBER

|                     |                                                   |
|---------------------|---------------------------------------------------|
| <b>Syntax:</b>      | NUMBER= <i>num</i>                                |
| <b>Description:</b> | Number(s) to add to, or delete from, list.        |
| <b>Format:</b>      | Single number or colon separated list of numbers. |
| <b>Example:</b>     | 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 numbers for a specified VPN Network.

## Required Parameters

Here are the required parameters for this command.

NAME

|                     |                      |
|---------------------|----------------------|
| <b>Syntax:</b>      | NAME= <i>name</i>    |
| <b>Description:</b> | Name of VPN Network. |
| <b>Format:</b>      | String               |
| <b>Example:</b>     | 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.

# 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 .....    | 19 |
| Change a Network Speed Dial Number ..... | 20 |
| Delete a Network Speed Dial Number ..... | 21 |
| Query Network Speed Dial Numbers .....   | 22 |

## 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

|                     |                      |
|---------------------|----------------------|
| <b>Syntax:</b>      | NAME= <i>name</i>    |
| <b>Description:</b> | Name of VPN Network. |
| <b>Format:</b>      | String               |
| <b>Example:</b>     | NAME="VPNNW1"        |

NUMBER

|                     |                                                                                                                                                                                                                                                                             |
|---------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Syntax:</b>      | NUMBER= <i>num</i>                                                                                                                                                                                                                                                          |
| <b>Description:</b> | Number to add to the list.                                                                                                                                                                                                                                                  |
| <b>Format:</b>      | <i>xxx:yyyyy:n</i><br>where: <ul style="list-style-type: none"> <li>• <i>xxx</i> is the speed dial number</li> <li>• <i>yyyyy</i> is the physical address for termination, and</li> <li>• <i>n</i> is an On-net/Off-net indicator, where Y=On-net and N=Off-net.</li> </ul> |
| <b>Example:</b>     | 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

|                     |                      |
|---------------------|----------------------|
| <b>Syntax:</b>      | NAME= <i>name</i>    |
| <b>Description:</b> | Name of VPN Network. |
| <b>Format:</b>      | String               |
| <b>Example:</b>     | NAME="VPNNW1 "       |

NUMBER

|                     |                                        |
|---------------------|----------------------------------------|
| <b>Syntax:</b>      | NUMBER= <i>num</i>                     |
| <b>Description:</b> | Number to change from speed dial list. |
| <b>Format:</b>      | xxx                                    |
| <b>Example:</b>     | NUMBER=1                               |

NEW\_NUMBER

|                     |                        |
|---------------------|------------------------|
| <b>Syntax:</b>      | NEW_NUMBER= <i>num</i> |
| <b>Description:</b> | 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

|                     |                                           |
|---------------------|-------------------------------------------|
| <b>Syntax:</b>      | NUMBER= <i>num</i>                        |
| <b>Description:</b> | Number(s) to delete from speed dial list. |
| <b>Format:</b>      | Colon separated list of numbers.          |
| <b>Example:</b>     | 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

|                     |                      |
|---------------------|----------------------|
| <b>Syntax:</b>      | NAME= <i>name</i>    |
| <b>Description:</b> | Name of VPN Network. |
| <b>Format:</b>      | String               |
| <b>Example:</b>     | 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:yyyy:n* where *xxx* is the speed dial number, *yyyy* 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.



# 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.....             | 25 |
| Change a VPN Station Details ..... | 28 |
| Delete a VPN Station .....         | 31 |
| Query a VPN Station .....          | 32 |

## 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

|                     |                              |
|---------------------|------------------------------|
| <b>Syntax:</b>      | NETNAME= <i>network name</i> |
| <b>Description:</b> | The name of the VPN Network. |
| <b>Format:</b>      | String                       |
| <b>Example:</b>     | NETNAME="Net1"               |

STNNAME

|                     |                                                                  |
|---------------------|------------------------------------------------------------------|
| <b>Syntax:</b>      | STNNAME= <i>station name</i>                                     |
| <b>Description:</b> | The name of VPN Station. (Extension number field in VPN Screen). |
| <b>Format:</b>      | Extension number.                                                |
| <b>Example:</b>     | STNNAME=123                                                      |

PHYSADDR

|                     |                                            |
|---------------------|--------------------------------------------|
| <b>Syntax:</b>      | PHYSADDR= <i>msisdn</i>                    |
| <b>Description:</b> | 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.<br><br>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.<br><br>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 (VPN1=ADD) if ALLOWSHORTTEXT = N
- The number of digits in the STNNAME must be less than or equal to the EXTLENGTH defined in the add network command (VPN1=ADD) if ALLOWSHORTTEXT = 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
```

|                     |                                            |
|---------------------|--------------------------------------------|
| <b>Syntax:</b>      | PHYSADDR= <i>msisdn</i>                    |
| <b>Description:</b> | The Physical Address, usually real MSISDN. |
| <b>Format:</b>      | MSISDN number.                             |
| <b>Example:</b>     | PHYSADDR=6494111111                        |

```
STNNAME
```

|                     |                                                                  |
|---------------------|------------------------------------------------------------------|
| <b>Syntax:</b>      | STNNAME= <i>station name</i>                                     |
| <b>Description:</b> | 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.<br><br>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.<br><br>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

|                     |                                            |
|---------------------|--------------------------------------------|
| <b>Syntax:</b>      | PHYSADDR= <i>msisdn</i>                    |
| <b>Description:</b> | The Physical Address, usually real MSISDN. |
| <b>Format:</b>      | MSISDN number.                             |
| <b>Example:</b>     | PHYSADDR=6494111111                        |

### STNNNAME

**Syntax:** STNNNAME=*station name*  
**Description:** The name of VPN Station. (Extension number field in VPN Screen).  
**Format:** Extension number.  
**Example:** STNNNAME=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 STNNNAME and PHYSADDR to determine a unique station.
- Either or both STNNNAME 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.

## STNNNAME

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



# 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.....              | 37 |
| Change a VPN Station Restriction Details ..... | 38 |
| Delete a VPN Station White List.....           | 40 |
| Query a Station White List.....                | 42 |

## 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

|                     |                                                                  |
|---------------------|------------------------------------------------------------------|
| <b>Syntax:</b>      | STNNAME= <i>station name</i>                                     |
| <b>Description:</b> | The name of VPN Station. (Extension number field in VPN Screen). |
| <b>Format:</b>      | Extension number.                                                |
| <b>Example:</b>     | STNNAME=123                                                      |

PHYSADDR

|                     |                                            |
|---------------------|--------------------------------------------|
| <b>Syntax:</b>      | PHYSADDR= <i>msisdn</i>                    |
| <b>Description:</b> | The Physical Address, usually real MSISDN. |
| <b>Format:</b>      | MSISDN number.                             |
| <b>Example:</b>     | PHYSADDR=6494111111                        |

## Optional parameters

This command accepts the following optional parameters.

### NETNAME

|                     |                                          |
|---------------------|------------------------------------------|
| <b>Syntax:</b>      | <code>NETNAME=<i>network name</i></code> |
| <b>Description:</b> | The name of the VPN Network.             |
| <b>Format:</b>      | String                                   |
| <b>Example:</b>     | <code>NETNAME="Net1"</code>              |

### WHITELIST

|                     |                                                        |
|---------------------|--------------------------------------------------------|
| <b>Syntax:</b>      | <code>WHITELIST=<i>whitelist</i></code>                |
| <b>Description:</b> | The numbers to add to, or delete from, the white list. |
| <b>Format:</b>      | Single number, or colon separated list of numbers.     |
| <b>Example:</b>     | <code>WHITELIST=6449393404:6449393764</code>           |

## 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.

### STNNNAME

**Syntax:** STNNNAME=*station name*  
**Description:** The name of VPN Station. (Extension number field in VPN Screen).  
**Format:** Extension number.  
**Example:** STNNNAME=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:** NEW\_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 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
- 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.

### STNNNAME

|                     |                                                                  |
|---------------------|------------------------------------------------------------------|
| <b>Syntax:</b>      | STNNNAME= <i>station name</i>                                    |
| <b>Description:</b> | The name of VPN Station. (Extension number field in VPN Screen). |
| <b>Format:</b>      | Extension number.                                                |
| <b>Example:</b>     | STNNNAME=123                                                     |

### PHYSADDR

|                     |                                            |
|---------------------|--------------------------------------------|
| <b>Syntax:</b>      | PHYSADDR= <i>msisdn</i>                    |
| <b>Description:</b> | The Physical Address, usually real MSISDN. |
| <b>Format:</b>      | MSISDN number.                             |
| <b>Example:</b>     | PHYSADDR=6494111111                        |

## Optional parameters

This command accepts the following optional parameters.

### NETNAME

|                     |                              |
|---------------------|------------------------------|
| <b>Syntax:</b>      | NETNAME= <i>network name</i> |
| <b>Description:</b> | The name of the VPN Network. |
| <b>Format:</b>      | String                       |
| <b>Example:</b>     | NETNAME="Net1"               |

### WHITELIST

|                     |                                                        |
|---------------------|--------------------------------------------------------|
| <b>Syntax:</b>      | WHITELIST= <i>whitelist</i>                            |
| <b>Description:</b> | The numbers to add to, or delete from, the white list. |
| <b>Format:</b>      | Single number, or colon separated list of numbers.     |
| <b>Example:</b>     | WHITELIST=6449393404:6449393764                        |

## Logic and constraints

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

|                     |                                                                  |
|---------------------|------------------------------------------------------------------|
| <b>Syntax:</b>      | <code>STNNAME=station name</code>                                |
| <b>Description:</b> | The name of VPN Station. (Extension number field in VPN Screen). |
| <b>Format:</b>      | Extension number.                                                |
| <b>Example:</b>     | <code>STNNAME=123</code>                                         |

PHYSADDR

|                     |                                            |
|---------------------|--------------------------------------------|
| <b>Syntax:</b>      | <code>PHYSADDR=msisdn</code>               |
| <b>Description:</b> | The Physical Address, usually real MSISDN. |
| <b>Format:</b>      | MSISDN number.                             |
| <b>Example:</b>     | <code>PHYSADDR=6494111111</code>           |

### Optional parameter

This command accepts the following optional parameter.

NETNAME

|                     |                                   |
|---------------------|-----------------------------------|
| <b>Syntax:</b>      | <code>NETNAME=network name</code> |
| <b>Description:</b> | The name of the VPN Network.      |
| <b>Format:</b>      | String                            |
| <b>Example:</b>     | <code>NETNAME="Net1"</code>       |

### 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

### 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.



# 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 .....    | 45 |
| Change a VPN Station Hunting List ..... | 47 |
| Delete a VPN Station Hunting List ..... | 49 |
| Query a VPN Station Hunting List .....  | 51 |

## Add a VPN Station Hunting List

### Name

VPNST5=ADD

### Description

Adds a new VPN Station hunting list. Use the argument **ADDENTRIES** to add multiple entries in a single hunting list.

### Required Parameters

Here are the required parameters for this command.

HUNTLIST

|                     |                            |
|---------------------|----------------------------|
| <b>Syntax:</b>      | HUNTLIST= <i>name</i>      |
| <b>Description:</b> | The name of the Hunt list. |
| <b>Format:</b>      | String                     |
| <b>Example:</b>     | HUNTLIST="HL1"             |

### Constraint required parameters

This command accepts the following constraint required parameter.

STNNAME

|                     |                                                                  |
|---------------------|------------------------------------------------------------------|
| <b>Syntax:</b>      | STNNAME= <i>station name</i>                                     |
| <b>Description:</b> | The name of VPN Station. (Extension number field in VPN Screen). |
| <b>Format:</b>      | 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.

### ADDENTRIES

**Syntax:** ADDENTRIES=RANK1:TERMNUM1:TIMEOUT1:ONNET1 |  
RANK2:TERMNUM2:TIMEOUT2:ONNET2 |.....

**Description:** Adds multiple entries in a single hunting list. You can include a maximum of 10 entries.

**Format:** String

**Example:** ADDENTRIES=4:321100005:5:N|5:321100006:4:N

### 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

### 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`

**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 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.
- When **ADDENTRIES** argument is used, RANK, TERMNUM and TIMEOUT are not required.

**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.

### STNNNAME

|                     |                                                                  |
|---------------------|------------------------------------------------------------------|
| <b>Syntax:</b>      | <code>STNNNAME=station name</code>                               |
| <b>Description:</b> | The name of VPN Station. (Extension number field in VPN Screen). |
| <b>Format:</b>      | Extension number.                                                |
| <b>Example:</b>     | <code>STNNNAME=123</code>                                        |

### PHYSADDR

|                     |                                            |
|---------------------|--------------------------------------------|
| <b>Syntax:</b>      | <code>PHYSADDR=msisdn</code>               |
| <b>Description:</b> | The Physical Address, usually real MSISDN. |
| <b>Format:</b>      | MSISDN number.                             |
| <b>Example:</b>     | <code>PHYSADDR=6494111111</code>           |

### RANK

|                     |                                                 |
|---------------------|-------------------------------------------------|
| <b>Syntax:</b>      | <code>RANK=order</code>                         |
| <b>Description:</b> | The order to apply entries, in ascending order. |
| <b>Format:</b>      | Number                                          |
| <b>Example:</b>     | <code>RANK=1</code>                             |

### TIMEOUT

|                     |                                                    |
|---------------------|----------------------------------------------------|
| <b>Syntax:</b>      | <code>TIMEOUT=num</code>                           |
| <b>Description:</b> | The timeout.                                       |
| <b>Format:</b>      | Number of seconds. Integer value between 1 and 99. |
| <b>Example:</b>     | <code>TIMEOUT=5</code>                             |

### ONNET

|                     |                                             |
|---------------------|---------------------------------------------|
| <b>Syntax:</b>      | <code>ONNET=Y   N</code>                    |
| <b>Description:</b> | Is the terminating number an on-net number? |
| <b>Format:</b>      | Y, or N                                     |
| <b>Default:</b>     | N                                           |
| <b>Example:</b>     | <code>ONNET=Y</code>                        |

## Optional parameters

This command accepts the following optional parameters.

### NETNAME

|                     |                                   |
|---------------------|-----------------------------------|
| <b>Syntax:</b>      | <code>NETNAME=network name</code> |
| <b>Description:</b> | The name of the VPN Network.      |
| <b>Format:</b>      | String                            |
| <b>Example:</b>     | <code>NETNAME="Net1"</code>       |

**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. Use the argument **DELENTRIES** to delete multiple entries from a single 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.

DELENTRIES

**Syntax:** DELENTRIES=*TERMNUM1 | TERMNUM2 | .....*  
**Description:** Deletes multiple entries from a single hunting list. You can include a maximum of 10 entries.  
**Format:** String  
**Example:** DELENTRIES=3211000004 | 321100005

**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
- 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.

# 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 .....    | 53 |
| Change a VPN Station Hunting List Plan ..... | 56 |
| Delete a VPN Station Hunting List Plan ..... | 60 |
| Query a VPN Station Hunting List Plan .....  | 62 |

## 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

|                     |                            |
|---------------------|----------------------------|
| <b>Syntax:</b>      | HUNTLIST= <i>name</i>      |
| <b>Description:</b> | The name of the Hunt list. |
| <b>Format:</b>      | String                     |
| <b>Example:</b>     | HUNTLIST="HL1"             |

TIMERANGE

|                     |                                                                                                                                          |
|---------------------|------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Syntax:</b>      | TIMERANGE= <i>type</i>                                                                                                                   |
| <b>Description:</b> | The time range type.                                                                                                                     |
| <b>Format:</b>      | Type is: <ul style="list-style-type: none"> <li>• TOD (Time of day)</li> <li>• DOW (Day of week)</li> <li>• DOY (Day of year)</li> </ul> |
| <b>Example:</b>     | 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 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.
- 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

**Syntax:** 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

|                     |                                                                  |
|---------------------|------------------------------------------------------------------|
| <b>Syntax:</b>      | STNNAME= <i>station name</i>                                     |
| <b>Description:</b> | The name of VPN Station. (Extension number field in VPN Screen). |
| <b>Format:</b>      | Extension number.                                                |
| <b>Example:</b>     | STNNAME=123                                                      |

PHYSADDR

|                     |                                            |
|---------------------|--------------------------------------------|
| <b>Syntax:</b>      | PHYSADDR= <i>msisdn</i>                    |
| <b>Description:</b> | The Physical Address, usually real MSISDN. |
| <b>Format:</b>      | MSISDN number.                             |
| <b>Example:</b>     | 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.

STNNNAME

**Syntax:** STNNNAME=*station name*  
**Description:** The name of VPN Station. (Extension number field in VPN Screen).  
**Format:** Extension number.  
**Example:** STNNNAME=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 conjunction with STNNNAME and PHYSADDR to determine a unique station.
- Either or both STNNNAME 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*

## Chapter 8

- 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.

## 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.....                        | 65 |
| Change the Tariff for all VPN Stations in a Network ..... | 66 |

## 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

|                     |                                 |
|---------------------|---------------------------------|
| <b>Syntax:</b>      | TARIFFNAME= <i>tariff name</i>  |
| <b>Description:</b> | Default ACS tariff name to use. |
| <b>Format:</b>      | String                          |
| <b>Note:</b>        | Matches ACS_TARIFF_CODE         |
| <b>Example:</b>     | TARIFFNAME="TR1"                |

### Constraint required parameters

There are no constraint required parameters.

### Optional parameter

This command accepts the following optional parameter.

PROVIDER

|                     |                               |
|---------------------|-------------------------------|
| <b>Syntax:</b>      | PROVIDER= <i>sp</i>           |
| <b>Description:</b> | 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.
- TARIFFNAME should be present in acs\_tariff table.
- TARIFFNAME should be present in Services > ACS Service > ACS Resources > Tariff Codes screen which is created manually in UI.

### 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.
- NEW\_TARIFF name should be present in acs\_tariff table.
- NEW\_TARIFF name should be present in Services > ACS Service > ACS Resources > Tariff Codes screen which is created manually in UI.

### 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.



# 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..... | 69 |
| PI Command Errors..... | 70 |

## PI Chassis Errors

### Format

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

**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,<br>password            | Invalid username and/or password                                         |
| 73   | INVALID LOGON - user not<br>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<br>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: <ul style="list-style-type: none"> <li>• Did not send one</li> <li>• It was invalid</li> </ul> |
| 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 <i>name</i> does not exist                                  | VPNCU1=ADD               |
| 1    | NAME <i>name</i> already exists in vpn_network table                   | VPNNW1=ADD               |
| 1    | NAME <i>name</i> does not exist                                        | VPNNW1=CHG<br>VPNST7=CHG |
| 1    | NAME <i>name</i> does not exist in vpn_network table                   | VPNNW1=DEL               |
| 1    | No Network exists with prefix of NAME <i>name</i> in vpn_network table | VPNNW1=QRY               |

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

| Code | Message                                                                        | Commands                               |
|------|--------------------------------------------------------------------------------|----------------------------------------|
| 3    | PROVIDER <i>name</i> does not exist                                            | VPNFW1=CHG                             |
| 3    | No default network tariff defined                                              | VPNFW1=QRY                             |
| 3    | Blacklist is active                                                            | VPNFW2=DEL                             |
| 3    | NEW_NUMBER <i>number</i> already exists                                        | VPNFW3=CHG                             |
| 3    | STNNAME <i>name</i> length is invalid (not Y or N)                             | VPNST1=ADD                             |
| 3    | NEW_NAME <i>name</i> 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 <i>name</i> and PHYSADDR <i>name</i> pair | VPNST2=DEL                             |
| 3    | RANK <i>rank</i> out of range                                                  | VPNST5=ADD                             |
| 3    | RANK <i>rank</i> is not in a valid format                                      | VPNST5=CHG                             |
| 3    | TERMNUM <i>number</i> does not exist                                           | VPNST5=DEL                             |
| 4    | TERMINATING Call Plan <i>name</i> does not exist in acs_call_plan table        | VPNFW1=ADD                             |
| 4    | ORIGINATING Call Plan <i>name</i> does not exist                               | VPNFW1=CHG                             |
| 4    | One of NAME or LISTTYPE must be specified                                      | VPNFW1=QRY                             |
| 4    | OFFNET is invalid (not Y or N)                                                 | VPNST1=ADD                             |
| 4    | GVNSADDR <i>name</i> 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 <i>number</i> is invalid                                               | VPNST5=ADD                             |
| 4    | TIMEOUT <i>timeout</i> is not in a valid format                                | VPNST5=CHG                             |
| 4    | HUNTLIST <i>name</i> in use                                                    | VPNST5=DEL                             |
| 4    | TIMERANGE <i>name</i> is not valid (not one of TOD, DOW, DOY)                  | VPNST6=ADD<br>VPNST6=CHG<br>VPNST6=DEL |
| 5    | MANAGEMENT Call Plan <i>name</i> does not exist in acs_call_plan table         | VPNFW1=ADD                             |
| 5    | TERMINATING Call Plan <i>name</i> does not exist                               | VPNFW1=CHG                             |
| 5    | NEW_INOFFNET is invalid (not Y or N)                                           | VPNST1=ADD                             |
| 5    | PHYSADDR <i>name</i> 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 <i>value</i> is invalid                                                | VPNST5=ADD                             |
| 5    | ONNET is not Y or N                                                            | VPNST5=CHG                             |
| 5    | STARTTIME <i>time</i> is not in a valid format                                 | VPNST6=ADD                             |
| 5    | STARTTIME <i>time</i> is not valid                                             | VPNST6=CHG<br>VPNST6=DEL               |
| 6    | PHYSRANGE <i>value</i> not Y or N                                              | VPNFW1=ADD                             |

| Code | Message                                                                                                                                       | Commands                                                                                       |
|------|-----------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------|
| 6    | PHYSADDR <i>name</i> does not exist in a<br>vpn_network_physical_range:start_number to end_number<br>range                                    | VPNST1=ADD                                                                                     |
| 6    | MANAGEMENT Call Plan <i>name</i> does not exist                                                                                               | VPNNW1=CHG                                                                                     |
| 6    | VDDIADDR <i>name</i> does not exist                                                                                                           | VPNST1=CHG                                                                                     |
| 6    | WHITELIST number does not exist for STNNAME <i>name</i> and<br>PHYSADDR <i>name</i> pair                                                      | VPNST2=CHG                                                                                     |
| 6    | ONNET is not Y or N                                                                                                                           | VPNST5=ADD                                                                                     |
| 6    | NEW_RANK <i>rank</i> is not valid                                                                                                             | VPNST5=CHG                                                                                     |
| 6    | ENDTIME <i>time</i> is not in a valid format                                                                                                  | VPNST6=ADD                                                                                     |
| 6    | ENDTIME <i>time</i> is not valid                                                                                                              | VPNST6=CHG<br>VPNST6=DEL                                                                       |
| 7    | GNVSADDR <i>name</i> does not exist in a<br>vpn_network_physical_range:start_number to end_number<br>range                                    | VPNST1=ADD                                                                                     |
| 7    | PRESENTONNETADDR value not Y or N                                                                                                             | VPNNW1=ADD                                                                                     |
| 7    | PHYSRANGE value not Y or N                                                                                                                    | VPNNW1=CHG                                                                                     |
| 7    | TARIFFNAME <i>name</i> does not exist in ACS_TARIFF table                                                                                     | VPNST1=CHG                                                                                     |
| 7    | STNNAME and PHYSADDR both not specified                                                                                                       | VPNST2=ADD<br>VPNST2=CHG<br>VPNST2=DEL<br>VPNST2=QRY<br>VPNST6=ADD<br>VPNST6=DEL<br>VPNST6=QRY |
| 7    | NEW_TIMEOUT <i>timeout</i> 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 <i>name</i> does not exist in a<br>vpn_network_physical_range:start_number to end_number<br>range                                    | VPNST1=ADD                                                                                     |
| 8    | NEW_PHYSADDR <i>name</i> does not exist in a<br>vpn_network_physical_range:start_number to end_number<br>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 <i>time-time</i> overlaps an existing plan<br>range                                                                         | VPNST6=ADD<br>VPNST6=CHG                                                                       |
| 9    | EXTLENGTH not valid - <i>reason</i>                                                                                                           | VPNNW1=ADD                                                                                     |
| 9    | RESTRICTCLI value not Y or N                                                                                                                  | VPNNW1=CHG                                                                                     |
| 9    | TARIFFNAME <i>name</i> not valid                                                                                                              | VPNST1=ADD                                                                                     |
| 9    | NEW_STNNAME <i>name</i> is invalid                                                                                                            | VPNST1=CHG                                                                                     |
| 9    | INOFFNET and NEW_INOFFNET pair not provided                                                                                                   | VPNST2=CHG                                                                                     |
| 9    | TERMNUM <i>number</i> 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 <i>name</i> does not exist                                                 | VPNST6=CHG                                           |
| 10   | ALLOWSHORTEXT value not Y or N                                                          | VPNST1=ADD                                           |
| 10   | TARIFFNAME <i>name</i> does not exist                                                   | VPNST1=CHG                                           |
| 10   | PHYSADDR <i>address</i> is already in use                                               | VPNST1=ADD                                           |
| 10   | STNNAME <i>name</i> does not exist                                                      | VPNST1=CHG                                           |
| 10   | WHITELIST and NEW WHITELIST pair not provided                                           | VPNST2=CHG                                           |
| 10   | STNNAME and PHYSADDR both not specified                                                 | VPNST5=ADD<br>VPNST5=CHG<br>VPNST5=DEL<br>VPNST5=QRY |
| 10   | NEW_TIMERANGE <i>name</i> is not valid (not one of TOD, DOW, DOY)                       | VPNST6=CHG                                           |
| 11   | TARIFFNAME <i>name</i> does not exist                                                   | VPNST1=ADD                                           |
| 11   | TARIFFNAME <i>name</i> not valid for PROVIDER <i>name</i>                               | VPNST1=CHG                                           |
| 11   | WHITELIST does not exist for STNNAME <i>name</i> and PHYSADDR <i>name</i> pair          | VPNST1=ADD                                           |
| 11   | NEW_RANK specified without RANK                                                         | VPNST5=CHG                                           |
| 11   | NEW_STARTTIME <i>time</i> is not valid                                                  | VPNST6=CHG                                           |
| 12   | TARIFFNAME <i>name</i> not valid for PROVIDER <i>name</i>                               | VPNST1=ADD                                           |
| 12   | NEW_TIMEOUT specified without TIMEOUT                                                   | VPNST5=CHG                                           |
| 12   | NEW_ENDTIME <i>time</i> is not valid                                                    | VPNST6=CHG                                           |
| 13   | SITE_CODE cannot be a sub or super string of an existing SITE_CODE                      | VPNST1=ADD<br>VPNST1=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<br>VPNST6=DEL<br>VPNST6=QRY               |
| 16   | The selected station does not have an associated hunting plan table                     | VPNST6=CHG<br>VPNST6=DEL<br>VPNST6=QRY               |

| Code | Message                                       | Commands                                                                                                                                                                                                       |
|------|-----------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 17   | Station selection is not unique               | VPNST1=ADD<br>VPNST1=CHG<br>VPNST1=DEL<br>VPNST2=ADD<br>VPNST2=CHG<br>VPNST2=DEL<br>VPNST2=QRY<br>VPNST5=ADD<br>VPNST5=CHG<br>VPNST5=DEL<br>VPNST5=QRY<br>VPNST6=ADD<br>VPNST6=CHG<br>VPNST6=DEL<br>VPNST6=QRY |
| 18   | Nothing to change                             | VPNNW1=CHG<br>VPNNW3=CHG<br>VPNST1=CHG<br>VPNST2=CHG                                                                                                                                                           |
| 19   | Badly formatted parameter <i>parameter</i>    | All commands                                                                                                                                                                                                   |
| 20   | <i>General error</i>                          | All commands                                                                                                                                                                                                   |
| 21   | TAG <i>tag name</i> does not exist            | VPNNW1=ADD<br>VPNNW1=CHG<br>VPNNW1=QRY<br>VPNST1=ADD<br>VPNST1=CHG<br>VPNST1=QRY                                                                                                                               |
| 22   | VALUE <i>value</i> is not valid               | VPNNW1=ADD<br>VPNNW1=CHG<br>VPNST1=ADD<br>VPNST1=CHG                                                                                                                                                           |
| 23   | Wrong number of values for the number of tags | VPNNW1=ADD<br>VPNNW1=CHG<br>VPNST1=ADD<br>VPNST1=CHG                                                                                                                                                           |