Reference Guide

AudioCodes Multi-Service Business Routers (MSBR)

TR-069

CPE WAN Management Protocol (CWMP) for Remote Device Management

Version 7.2

CC audiocodes

Table of Contents

List of Figures

WEEE EU Directive

Pursuant to the WEEE EU Directive, electronic and electrical waste must not be disposed of with unsorted waste. Please contact your local recycling authority for disposal of this product.

Customer Support

Customer technical support and services are provided by AudioCodes or by an authorized AudioCodes Service Partner. For more information on how to buy technical support for AudioCodes products and for contact information, please visit our website at https://www.audiocodes.com/services-support/maintenance-and-support.

Stay in the Loop with AudioCodes

Abbreviations and Terminology

Each abbreviation, unless widely used, is spelled out in full when first used.

Document Revision Record

LTRT	Description
52340	Initial document release.
52341	TR-098 added.
52342	InternetGatewayDevice.UploadDiagnostics. parameters; InternetGatewayDevice.DownloadDiagnostics.parameters; InternetGatewayDevice.LANDevice.{i}.WLANConfiguration.{i}.IEEE11iAuthenticationMode ; InternetGatewayDevice.LANDevice.{i}.WLANConfiguration.{i}.IEEE11iEncryptionMode
52343	Updated to 7.2.
52344	Updated to patch version 7.2.154 (conf-change-notification and TR-181)
52345	New section - Switching between TR-181 and TR-098; updated with TR-181 Objects
52347	Device.Time. added to TR-181 data model; InternetGatewayDevice.Time. added to standard parameters for InternetGatewayDevice
52348	Device.GatewayInfo object added to TR-181 data model; ManageableDeviceNumberOfEntries parameter added to TR-098 data model; InternetGatewayDevice.ManagementServer.ManageableDevice.{i}. object added to TR-098 data model
52349	Updated to 7.2.256 (TR-069 over IPv6).
52355	Adding and deleting certificates through TR-098 data model (InternetGatewayDevice.ManagementServer.X_00908F_RootCertificate.{i}.); locking and unlocking device (InternetGatewayDevice.X_00908F_Admin.{i}.)
52356	XMPP
52357	New configuration command `ntp-dependency`.

Documentation Feedback

AudioCodes continually strives to produce high quality documentation. If you have any comments (suggestions or errors) regarding this document, please fill out the Documentation Feedback form on our website at https://online.audiocodes.com/documentation-feedback.

1 Introduction

This document describes the CPE WAN Management Protocol (TR-069), intended for communication between a CPE—**AudioCodes Mediant MSBR series**—and an Auto-Configuration Server (ACS). The CPE WAN Management Protocol defines a mechanism that encompasses secure auto-configuration of a CPE, and also incorporates other CPE management functions into a common framework.

The CPE WAN Management Protocol is intended to support a variety of functionalities to manage a collection of CPE, including the following primary capabilities:

■ Auto-configuration and dynamic service provisioning

■ Software/firmware image management

■ Software module management

■ Status and performance monitoring

■ Diagnostics

This page is intentionally left blank.

2   TR-069 Data Model

TR-069 is a specification published by Broadband Forum (http://www.broadband-forum.org/) entitled CPE WAN management protocol (CWMP). It defines an application layer protocol for remote management of end-user devices.

TR-069 uses a bi-directional SOAP/HTTP protocol for communication between the customer premises equipment (CPE) and the Auto Configuration Servers (ACS). For MSBR devices, the TR-069 connection to the ACS can be done on the LAN or WAN interface.

The protocol stack looks as follows:

Figure 2-1: TR-069 Protocol Stack

CPE/ACS Management Application
RPC Methods
SOAP
HTTP
SSL/TLS
TCP/IP

Communication is typically established by the CPE; hence, messages from CPE to ACS are typically carried in HTTP requests, and messages from ACS to CPE in HTTP responses.

Figure 2-2: TR-069 Session Example
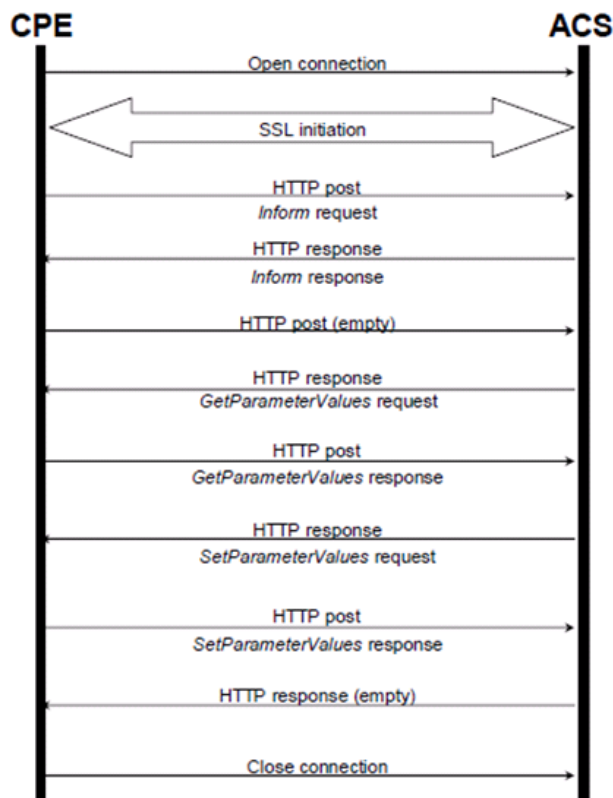
Communication between ACS and CPE is defined via Remote Procedure Call (RPC) methods. TR-069 defines a generic mechanism by which an ACS can read or write parameters to configure a CPE and monitor CPE status and statistics. It also defines the

mechanism for file transfer and firmware/software management. However, it does not define individual parameters; these are defined in separate documents, as described below.

Some of the RPC methods are Configuration File Download, Firmware upgrade, Get Parameter Value, Set Parameter Value, Reboot, and the upload and download files.

3   Configuring TR-069 through CLI

This section describes how to enable and configure TR-069 through the device's CLI.

➢ **To configure TR-069 through CLI:**

1. Establish serial communication with the device.
2. Log in to the CLI.
3. Access the System configuration mode:

```
# configure system
(config-system)#
```

4. At the prompt, access the CWMP command set by entering the following:

```
(config-system)# cwmp
(cwmp-tr069)#
```

The TR-069 parameters are described in the table below.

Table 3-1: TR-069 Parameters

Parameter	Description
`acs-password [STRING]` `acs-password [STRING] obscured`	Defines the password to access the ACS.
`acs-url-provisioning-mode` `[automatic | manual]`	Determines whether the ACS URL is acquired using DHCP Option 43 (automatic) or user-defined (manual). When using the automatic option, you also need to add parameters under the WAN interface.
`acs-url`	Defines the ACS URL. **Note:** This command is applicable only if **acs-url-provisioning-mode manual**.
`acs-user-name`	Defines the username to access the ACS.
`conf-change-notification` `{off|on}`	Enables the device to notify the TR-069 ACS of device configuration changes. The device sends this as a TR-069 Value Change Event when the management user logs out of the management interface (Web or CLI). This feature is available for Data Model InternetGatewayDevice (TR-098).
`connection-request-password`	Defines the CPE connection request password.
`connection-request-user-name`	Defines the CPE connection request username.
`cwmp-acl`	Binds an access list to the CWMP interface.
`data-model` `[InternetGatewayDevice|Device]`	Defines the TR-069 data model. **Note:** For the parameter to take effect, a device reset is required.
`default-inform-interval`	Defines the TR-069 default periodic inform interval (in seconds).
`period-inform-enable`	Enables the device to send periodic inform messages to the ACS.
`port`	Defines the TR-069 port of the device. **Note:** For the parameter to take effect, a device reset is required.
`send-connection-request`	The CPE issues a connection request event toward an ACS.
`service [on|off]`	Determines whether to enable the TR-069 service.

Parameter	Description
`tcp-fragment[on|off]`	Sets the "Don't fragment (DF)" flag for sending packets on the socket to 1 (off) and to 0 (on).
`tls-context`	Defines the TR-069 TLS Context index.
`ntp-dependency [on|off]`	When the device is configured to connect securely to the TR-069 Auto Configuration Server (ACS) over TLS and to verify the certificate, you can enable this command to connect only when the device is synchronized with the NTP server.
`tr069-cwmp-wait-interval`	Defines the TR-069 CWMP interval timeout (in seconds). This overwrites the TR's state-machine interval (Backoff algorithm).
`verify-certificate`	Enables certificate verifaction during TR-069 connection.
`verify-common-name`	Enables common name verification during TR-069 connection
`vrf-name`	TR-069 application's VRF name is set to 'main_vrf' for default vrf (main routing table).
WAN Configuration for acs-url- provisioning-mode command	
`ip address dhcp`	Defines the DHCP client for the interface.
`ip dhcp-client class-id [name of class id]`	Defines the Class-ID (Option 60).
`ip dhcp-client default-route`	Defines the received DHCP gateway address as the default route.
`ip dhcp-client acs-provisioning-sub-option 1`	TR-069 ACS provisioning sub-option.

The following are CWMP configuration examples:

■ **For Static ACS URL:**

```
configure system
   cwmp
    set acs-password $1$BHB3NjEx obscured
    set connection-request-password $1$kPbl8/Dn obscured
    set acs-url "https://friendly2:8443/dps-basic/TR069"
    set acs-user-name "tr069"
    set connection-request-user-name "ftacs"
    port 3050
    default-inform-interval 3600
    service on
    activate
   exit
```

■ **For acs-url- provisioning-mode automatic:**

```
configure system
   cwmp
    set acs-password $1$BHB3NjEx obscured
    set connection-request-password $1$kPbl8/Dn obscured
    acs-url-provisioning-mode automatic
    set acs-user-name "tr069"
    set connection-request-user-name "ftacs"
    port 3050
    service on
```

```
     activate
   exit

interface GigabitEthernet 0/0
    ip address dhcp
    ip dhcp-client class-id "xdslforum.org"
    ip dhcp-client default-route
    ip dhcp-client acs-provisioning-sub-option 1
    no shutdown
   exit
```

3.1 Switching between TR-181 and TR-098

The device can operate with TR-181 or TR-098 data models. To switch between models, you can use the CLI or ini file:

■ **CLI:**

```
# configure system
(config-system)# cwmp
(cwmp-tr069)# data-model {device|internetgatewaydevice}
```

where:

- **device:** TR-181 data model
- **internetgatewaydevice**: TR-098 data model

■ **ini File:** Load an ini file with the Tr069DataModel ini file parameter configured to one of the following enumerations:

- **0:** TR-181 data model
- **1:** TR-098 data model

Note: For the change to take effect, you need to reset the device.

4   TR-181 Data Model

> **Note:** The following notations are used in the tables below:
> - "**– (W)**": The attribute is "-" even though in the TR-104 standard it is defined as "Read-Write". This is because the specific attribute is not supported by our device.
> - "**-**": The attribute is Read Only.
> - "**W**": The attribute is Read-Write.

The following TR-181 objects are supported

Table 4-1: TR-181 Objects

Name	Type	Write	Description
Device.RootDataModel Version	string	-	Defines the root data model version, e.g., "2.4". For a vendor-defined root data model, this is the standard Broadband Forum model on which the vendor-defined model is based.
Device.DeviceInfo.	object	-	This object contains general device information.
Device.DeviceInfo.Man ufacturer	string	-	Defines the manufacturer of the CPE (human readable string).
Device.DeviceInfo.Man ufacturerOUI	string	-	Defines the organizationally unique identifier of the device manufacturer. Represented as a six hexadecimal-digit value using all upper-case letters and including any leading zeros. Possible patterns: [0-9A-F]{6} The value MUST be a valid OUI as defined in [OUI]. This value MUST remain fixed over the lifetime of the device, including across firmware updates. Any change would indicate that it's a new device and would therefore require a BOOTSTRAP Inform.
Device.DeviceInfo.Mod elName	string	-	Defines the model name of the CPE (human readable string).
Device.DeviceInfo.Prod uctClass	string	-	Defines the Identifier of the class of product for which the serial number applies. That is, for a given manufacturer, this parameter is used to identify the product or class of product over which the SerialNumber parameter is unique. This value MUST remain fixed over the lifetime of the device, including across firmware updates. Any change would indicate that it's a new device and would therefore require a BOOTSTRAP Inform.

Name	Type	Write	Description
Device.DeviceInfo.SerialNumber	string	-	Defines the Identifier of the particular device that is unique for the indicated class of product and manufacturer. This value MUST remain fixed over the lifetime of the device, including across firmware updates. Any change would indicate that it's a new device and would therefore require a BOOTSTRAP Inform.
Device.DeviceInfo.HardwareVersion	string	-	Defines a string identifying the particular CPE model and version.
Device.DeviceInfo.SoftwareVersion	string	-	Defines a string identifying the software version currently installed in the CPE (i.e., version of the overall CPE firmware). To allow version comparisons, this element SHOULD be in the form of dot-delimited integers, where each successive integer represents a more minor category of variation. For example, "3.0.21" where the components mean: "Major.Minor.Build".
Device.DeviceInfo.ProvisioningCode	string	W	Defines the Identifier of the primary service provider and other provisioning information, which MAY be used by the ACS to determine service provider-specific customization and provisioning parameters.
Device.DeviceInfo.UpTime	unsignedInt	-	Defines the time in seconds since the CPE was last restarted.
Device.DeviceInfo.FirstUseDate	dateTime	-	Defines the Date and time in UTC that the CPE first both successfully established an IP-layer network connection and acquired an absolute time reference using NTP or equivalent over that network connection. The CPE may reset this date after a factory reset. If NTP or equivalent is not available, this parameter, if present, should be set to the Unknown Time value.
Device.DeviceInfo. X_00908F_DeviceLog	string	-	This parameter contains the Systemlog that is displayed for informational purposes in the User Interface. The content must be preserved across a device reboot.
Device.DeviceInfo. X_00908F_ConfigFile	string	W	Defines a dump of the currently running configuration on the CPE.
Device.ManagementServer.	object	-	This object contains parameters relating to the CPE's association with an ACS.
Device.ManagementServer.EnableCWMP	boolean	W	Enables and disables the CPE's support for CWMP.

Name	Type	Write	Description
			False means that CWMP support in the CPE is disabled, in which case the device MUST NOT send any Inform messages to the ACS or accept any Connection Request notifications from the ACS. True means that CWMP support on the CPE is enabled. The subscriber can re-enable the CPE's CWMP support either by performing a factory reset or by using a LAN-side protocol to change the value of this parameter back to true. The factory default value MUST be true.
Device.ManagementServer.URL	string	W	Defines the URL, as defined in RFC 3986, for the CPE to connect to the ACS using the CPE WAN Management Protocol. This parameter MUST be in the form of a valid HTTP or HTTPS URL. The host portion of this URL is used by the CPE for validating the ACS certificate when using SSL or TLS. Note that on a factory reset of the CPE, the value of this parameter might be reset to its factory value. If an ACS modifies the value of this parameter, it SHOULD be prepared to accommodate the situation that the original value is restored as the result of a factory reset.
Device.ManagementServer.Username	string	W	Defines the username used to authenticate the CPE when making a connection to the ACS using the CPE WAN Management Protocol. This username is used only for HTTP-based authentication of the CPE. Note that on a factory reset of the CPE, the value of this parameter might be reset to its factory value. If an ACS modifies the value of this parameter, it SHOULD be prepared to accommodate the situation that the original value is restored as the result of a factory reset.
Device.ManagementServer.Password	string	W	Defines the Password used to authenticate the CPE when making a connection to the ACS using the CPE WAN Management Protocol. This password is used only for HTTP-based authentication of the CPE. Note that on a factory reset of the CPE, the value of this parameter might be reset to its factory value. If an ACS modifies the value of this parameter, it SHOULD

Name	Type	Write	Description
			be prepared to accommodate the situation that the original value is restored as the result of a factory reset.
Device.ManagementServer.PeriodicInformEnable	boolean	W	Defines whether or not the CPE must periodically send CPE information to the ACS using the Inform method call.
Device.ManagementServer.PeriodicInformInterval	integer	W	Defines the duration in seconds of the interval for which the CPE MUST attempt to connect with the ACS and call the Inform method if PeriodicInformEnable is true.
Device.ManagementServer.PeriodicInformTime	dateTime	W	Defines an absolute time reference in UTC to determine when the CPE will initiate the periodic Inform method calls. Each Inform call MUST occur at this reference time plus or minus an integer multiple of the PeriodicInformInterval. PeriodicInformTime is used only to set the phase of the periodic Informs. The actual value of PeriodicInformTime can be arbitrarily far into the past or future. For example, if PeriodicInformInterval is 86400 (a day) and if PeriodicInformTime is set to UTC midnight on some day (in the past, present, or future) then periodic Informs will occur every day at UTC midnight. These MUST begin on the very next midnight, even if PeriodicInformTime refers to a day in the future. The Unknown Time value defined in [Section 3.2/TR-106a2] indicates that no particular time reference is specified. That is, the CPE MAY locally choose the time reference, and needs only to adhere to the specified PeriodicInformInterval. If absolute time is not available to the CPE, its periodic Inform behavior MUST be the same as if the PeriodicInformTime parameter was set to the Unknown Time value.
Device.ManagementServer.ParameterKey	string	W	The ParameterKey provides the ACS a reliable and extensible means to track changes made by the ACS. The value of ParameterKey MUST be equal to the value of the ParameterKey argument from the most recent successful SetParameterValues, AddObject, or DeleteObject method call from the ACS. The CPE MUST set ParameterKey to the value specified in the corresponding method arguments if and only if the method completes successfully and no fault response is generated. If a method call does not complete successfully

Name	Type	Write	Description
			(implying that the changes requested in the method did not take effect), the value of ParameterKey MUST NOT be modified. The CPE MUST only modify the value of ParameterKey as a result of SetParameterValues, AddObject, DeleteObject, or due to a factory reset. On factory reset, the value of ParameterKey MUST be set to an empty string.
Device.ManagementServer.ConnectionRequestURL	string	-	Defines the HTTP URL, as defined in RFC 3986, for an ACS to make a Connection Request notification to the CPE. In the form: http://host:port/path The host portion of the URL MAY be the IP address for the management interface of the CPE in lieu of a host name. Note: If the host portion of the URL is a literal IPv6 address then it MUST be enclosed in square brackets (see Section 3.2.2 RFC 3986).
Device.ManagementServer.ConnectionRequestUsername	string	W	Defines the Username used to authenticate an ACS making a Connection Request to the CPE.
Device.ManagementServer.ConnectionRequestPassword	string	W	Defines the Password used to authenticate an ACS making a Connection Request to the CPE.
Device.ManagementServer.CWMPRetryMinimumWaitInterval	unsignedInt	W	Configures the first session retry wait interval, in seconds, as specified in Section 3.2.1.1 of TR-069a2. A value of 5 corresponds to the default behavior that is described in [TR-069a2]. The device MUST use a random value between CWMPRetryMinimumWaitInterval and (CWMPRetryMinimumWaitInterval * CWMPRetryIntervalMultiplier / 1000) as the first retry wait interval. Other values in the retry pattern MUST be calculated using this value as a starting point.
Device.ManagementServer.CWMPRetryIntervalMultiplier	integer	W	Configures the retry interval multiplier as specified in Section 3.2.1.1 of TR-069a2. This value is expressed in units of 0.001. Hence the values of the multiplier range between 1.000 and 65.535.

Name	Type	Write	Description
			A value of 2000 corresponds to the default behavior that is described in [TR-069a2].
			The device MUST use a random value between CWMPRetryMinimumWaitInterval and (CWMPRetryMinimumWaitInterval * CWMPRetryIntervalMultiplier / 1000) as the first retry wait interval. Other values in the retry pattern MUST be calculated using this value as a starting point.
Device.ManagementServer.AliasBasedAddressing	boolean	-	Indicates whether or not the Alias-Based Addressing Mechanism is supported. A true value indicates that the CPE supports the Alias-Based Addressing Mechanism, as defined in Section 3.6.1 of TR-069a4 and described in Appendix II of TR-069a4.
Device.ManagementServer.X_00908F_ACSSignalCounter	unsignedInt	W	Defines the remaining seconds to signal ACS activity to the customer with a blinking LED, a running light or similar means. The concrete way of signaling is specified in the device specification. The ACS will set this parameter in different use cases and then the signaling must occur for the period set. If read, the parameter returns the number of seconds remaining in the signaling period. If the value is set to 0 or the countdown reaches 0, the signaling must stop immediately. The value must persist across a reboot of the device. That means, if the device signals ACS activity and the value reached for instance 15 seconds and the device has to reboot, the device has to start signaling ACS activity immediately after reboot and continue the countdown from 15 seconds downwards. It is acceptable that the time between Start reboot and Finish reboot will not reduce the residual time of the counter.
Device.ManagementServer.ManageableDeviceNumberOfEntries	unsignedInt	-	The number of entries in the ManageableDevice table.
Device.ManagementServer.ManageableDevice.{i}.	object	-	Each entry in this table corresponds to a distinct LAN Device that supports Device-Gateway Association according to [Annex F/TR-069a2] as indicated by the presence of the DHCP option specified in that Annex.
Device.ManagementServer.ManageableDevice.{i}.ManufacturerOUI	string	-	Organizationally unique identifier of the Device manufacturer as provided to the Gateway by the Device. Represented as

Name	Type	Write	Description
			a six hexadecimal-digit value using all upper-case letters and including any leading zeros. Possible patterns: [0-9A-F]{6} The value MUST be a valid OUI as defined in [OUI].
Device.ManagementServer.ManageableDevice.{i}.SerialNumber	string(64)	-	Serial number of the Device as provided to the Gateway by the Device.
Device.ManagementServer.ManageableDevice.{i}.ProductClass	string(64)	-	Identifier of the class of product for which the Device's serial number applies as provided to the Gateway by the Device. If the Device does not provide a Product Class, then this parameter MUST be an empty string.
Device.ManagementServer.ManageableDevice.{i}.Host	String(1024)	-	Comma-separated list (maximum list length 1024) of strings. Each list item MUST be the Path Name of a row in the Hosts.Host table. If the referenced object is deleted, the corresponding item MUST be removed from the list. References all the Hosts.Host table entries, whether active or inactive, that correspond to this physical LAN device. There can be multiple such entries if the device has more than one network interface.
Device.ManagementServer.X_00908F_RootCertificateNumberOfEntries	unsignedInt	-	The number of entries in the X_00908F_RootCertificate table.
Device.ManagementServer.X_00908F_RootCertificate.{i}.	object	-	This object is a superset of the Device.Security.Certificate.{i} object defined in [Device:2.4], which has been extended to add and delete certificates.
Device.ManagementServer.X_00908F_RootCertificate.{i}.Enabled	boolean	-	Displays if this root certificate is enabled for validity checking.
Device.ManagementServer.X_00908F_RootCertificate.{i}.Certificate	base64(4096)	W	The DER encoded certificate.
Device.ManagementServer.X_00908F_RootCertificate.{i}.LastModif	dateTime	-	The last modification time of this certificate.
Device.ManagementServer.X_00908F_RootCertificate.{i}.SerialNumber	string(64)	-	The Serial Number field in an X.509 certificate (see RFC 5280).

Name	Type	Write	Description
Device.ManagementServer.X_00908F_RootCertificate.{i}.Issuer	string(256)	-	The Issuer field in an X.509 certificate (see RFC 5280). In other words, the Distinguished Name (DN) of the entity who has signed the certificate.
Device.ManagementServer.X_00908F_RootCertificate.{i}.NotBefore	dateTime	-	The beginning of the certificate validity period. In other words, the Not Before field in an X.509 certificate (see RFC 5280).
Device.ManagementServer.X_00908F_RootCertificate.{i}.NotAfter	dateTime	-	The end of the certificate validity period. In other words, the Not After field in an X.509 certificate (see RFC 5280).
Device.ManagementServer.X_00908F_RootCertificate.{i}.Subject	string(256)	-	The Distinguished Name (DN) of the entity associated with the Public Key. In other words, the Subject field in an X.509 certificate (see RFC 5280).
Device.ManagementServer.X_00908F_RootCertificate.{i}.SubjectAlt	string(256)	-	Comma-separated list (maximum length 256) of strings. Each item is a DNS Name. The Subject Alternative Names extension field in an X.509 certificate (see RFC 5280).
Device.ManagementServer.X_00908F_RootCertificate.{i}.SignatureAlgorithm	string(256)	-	The algorithm used in signing the certificate. In other words, the Signature Algorithm field in an X.509 certificate (see RFC 5280).
Device.GatewayInfo.	object	-	This object contains information associated with a connected Internet Gateway Device.
ManufacturerOUI	string(6)	-	Organizationally unique identifier of the associated Internet Gateway Device. Possible patterns: ▪ <Empty> (an empty string) ▪ [0-9A-F]{6} An empty string indicates that there is no associated Internet Gateway Device that has been detected.
ProductClass	string(64)	-	Identifier of the product class of the associated Internet Gateway Device. An empty string indicates either that there is no associated Internet Gateway Device that has been detected, or the Internet Gateway Device does not support the use of the product-class parameter.
SerialNumber	string(64)	-	Serial number of the associated Internet Gateway Device. An empty string indicates that there is no associated Internet Gateway Device that has been detected.

Name	Type	Write	Description
Device.Time.	object	-	This object contains parameters relating to an NTP or SNTP time client in the CPE.
Enable	boolean	W	Enables or disables the NTP or SNTP time client.
Status	string	-	Status of Time support on the CPE. Enumeration: ▪ Disabled ▪ Unsynchronized: indicates that the CPE's absolute time has not yet been set. ▪ Synchronized: indicates that the CPE has acquired accurate absolute time; its current time is accurate. ▪ Error_FailedToSynchronize: indicates that the CPE failed to acquire accurate absolute time; its current time is not accurate. ▪ Error (optional): The value MAY be used by the CPE to indicate a locally defined error condition.
NTPServer1	string(64)	W	First NTP timeserver. Either a host name or IP address.
NTPServer2	string(64)	W	Second NTP timeserver. Either a host name or IP address.
CurrentLocalTime	dateTime	-	The current date and time in the CPE's local time zone.
LocalTimeZone	string(256)	W	The local time zone definition, encoded according to IEEE 1003.1 (POSIX). For example: EST+5 EDT,M4.1.0/2,M10.5.0/2
Device.UserInterface.	object	-	This object contains parameters relating to the user interface of the CPE.
Device.UserInterface.PasswordReset	boolean	W	Present only if the CPE provides a password-protected LAN-side user interface and supports LAN-side Auto-Configuration. When set to true, resets LANConfigSecurity.ConfigPassword to its factory value. When read, this parameter returns false, regardless of the actual value. The value of this parameter is not part of the device configuration and is always false when read.
Device.InterfaceStack.{i}.	object	-	This table contains information about the relationships between the multiple layers of interface objects ([Section 4.3/TR-181i2]). In particular, it contains

Name	Type	Write	Description
			information on which interfaces run on top of which other interfaces. This table is auto-generated by the CPE based on the LowerLayers parameters on individual interface objects. Each table row represents a "link" between two interface objects, a higher-layer interface object (referenced by HigherLayer) and a lower-layer interface object (referenced by LowerLayer). Consequently, if a referenced interface object is deleted, the CPE MUST delete the corresponding InterfaceStack row(s) that had referenced it. At most one entry in this table can exist with the same values for HigherLayer and LowerLayer, or with a given value for Alias.
Device.InterfaceStack.{i }.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in [Section 3.6.1/TR-069a4] and described in [Appendix II/TR-069a4], the following mandatory constraints MUST be enforced: Its value MUST NOT be empty. Its value MUST start with a letter. If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. The CPE MUST NOT change the parameter value.
Device.InterfaceStack.{i }.HigherLayer	string	-	Defines a reference to the interface object corresponding to the higher layer of the relationship, i.e., the interface which runs on top of the interface identified by the corresponding instance of LowerLayer.
Device.InterfaceStack.{i }.LowerLayer	string	-	Defines a reference to the interface object corresponding to the lower layer of the relationship, i.e. the interface which runs below the interface identified by the corresponding instance of HigherLayer. When the referenced lower layer interface is deleted, the CPE MUST delete the table row.
Device.InterfaceStack.{i }.HigherAlias	string	-	Defines the value of the Alias parameter for the interface object referenced by HigherLayer.

Name	Type	Write	Description
Device.InterfaceStack.{i}.LowerAlias	string	-	Defines the value of the Alias parameter for the interface object referenced by LowerLayer.
Device.DSL.	object	-	This object models DSL lines, DSL channels, DSL bonding, and DSL diagnostics. The specific interface objects defined here are Line, Channel, and BondingGroup. Each Line models a layer 1 DSL Line interface, and each Channel models a layer 1 DSL Channel interface where multiple channels can run over a DSL line. In the case where bonding is configured, it is expected that BondingGroup is stacked above the Channel instances within its group.
Device.DSL.LineNumberOfEntries	unsignedInt	-	Defines the number of entries in the Line table.
Device.DSL.ChannelNumberOfEntries	unsignedInt	-	Defines the number of entries in the Line table.
Device.DSL.Line.{i}.	object	-	Defines a DSL Line table (a stackable interface object as described in [Section 4.2/TR-181i2]). This table models physical DSL lines. At most one entry in this table can exist with a given value for Alias, or with a given value for Name.
Device.DSL.Line.{i}.Enable	boolean	W	Enables or disables the DSL line. This parameter is based on ifAdminStatus from RFC 2863.
Device.DSL.Line.{i}.EnableDataGathering	boolean	W	Enables or disables data gathering on the DSL line.
Device.DSL.Line.{i}.Status	string	-	Defines the current operational state of the DSL line (see Section 4.2.2 of TR-181i2). Possible values: ▪ Up ▪ Down ▪ Unknown ▪ Dormant ▪ NotPresent ▪ LowerLayerDown ▪ Error (OPTIONAL) When Enable is false then Status SHOULD normally be Down (or NotPresent or Error if there is a fault condition on the interface). When Enable is changed to true then Status SHOULD change to Up if and only if the interface is able to transmit and

Name	Type	Write	Description
			receive network traffic; it SHOULD change to Dormant if and only if the interface is operable but is waiting for external actions before it can transmit and receive network traffic (and subsequently change to Up if still operable when the expected actions have completed); it SHOULD change to LowerLayerDown if and only if the interface is prevented from entering the Up state because one or more of the interfaces beneath it is down; it SHOULD remain in the Error state if there is an error or other fault condition detected on the interface; it SHOULD remain in the NotPresent state if the interface has missing (typically hardware) components; it SHOULD change to Unknown if the state of the interface can not be determined for some reason. This parameter is based on ifOperStatus from RFC 2863.
Device.DSL.Line.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in [Section 3.6.1/TR-069a4] and described in [Appendix II/TR-069a4], the following mandatory constraints MUST be enforced: Its value MUST NOT be empty. Its value MUST start with a letter. If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. The CPE MUST NOT change the parameter value.
Device.DSL.Line.{i}.Name	string	-	Defines the textual name of the DSL line as assigned by the CPE.
Device.DSL.Line.{i}.LastChange	string	-	Defines the accumulated time in seconds since the DSL line entered its current operational state.
Device.DSL.Line.{i}.LowerLayers	string	W	Each list item MUST be the path name of an interface object that is stacked immediately below this interface object. If the referenced object is deleted, the corresponding item MUST be removed from the list. Comma-separated list (maximum list length 1024) of strings. See Section 4.2.1 of TR-181i2.

Name	Type	Write	Description
			Note: Since Line is a layer 1 interface, it is expected that LowerLayers will not be used.
Device.DSL.Line.{i}.Upstream	boolean	-	Defines whether the interface points towards the Internet (true) or towards End Devices (false). For example: For an Internet Gateway Device, Upstream will be true for all WAN interfaces and false for all LAN interfaces. For a standalone WiFi Access Point that is connected via Ethernet to an Internet Gateway Device, Upstream will be true for the Ethernet interface and false for the WiFi Radio interface. For an End Device, Upstream will be true for all interfaces.
Device.DSL.Line.{i}.FirmwareVersion	string	-	Defines a string identifying the version of the modem firmware currently installed for this interface. This is applicable only when the modem firmware is separable from the overall CPE software.
Device.DSL.Line.{i}.StandardsSupported	string	-	Defines a List of items that indicate which DSL standards and recommendations are supported by the Line instance. Comma-separated list of strings. Possible values: ▪ G.992.1_Annex_A ▪ G.992.1_Annex_B ▪ G.992.3_Annex_A ▪ G.992.3_Annex_B ▪ G.992.3_Annex_I ▪ G.992.3_Annex_J ▪ G.992.3_Annex_L ▪ G.992.3_Annex_M ▪ G.992.5_Annex_A ▪ G.992.5_Annex_B ▪ G.992.5_Annex_I ▪ G.992.5_Annex_J ▪ G.992.5_Annex_M ▪ 993.2_Annex_A ▪ 993.2_Annex_B ▪ 993.2_Annex_C **Note:** In G.997.1, this parameter is called "xDSL Transmission system capabilities". See ITU-T Recommendation [G.997.1]. This parameter is DEPRECATED because its entries are out-of-date; XTSE points to a current list. Therefore its value

Name	Type	Write	Description
			MAY be an empty string if (and only if) XTSE is supported.
Device.DSL.Line.{i}.StandardUsed	string	-	Defines the value MUST be a member of the list reported by the Standards Supported parameter. Indicates the standard that the Line instance is using for the connection.
			Note: In G.997.1, this parameter is called "xDSL Transmission system". See ITU-T Recommendation [G.997.1].
			This parameter is DEPRECATED because its entries are out-of-date; XTSE points to a current list. Therefore its value MAY be an empty string if (and only if) XTSUsed is supported.
Device.DSL.Line.{i}.AllowedProfiles	string	-	Comma-separated list of strings. List items indicate which VDSL2 profiles are allowed on the line. Each list item is an enumeration of:
			▪ 8a
			▪ 8b
			▪ 8c
			▪ 8d
			▪ 12a
			▪ 12b
			▪ 17a
			▪ 17b
			▪ 30a
			Note: In G.997.1, this parameter is called PROFILES. See ITU-T Recommendation G.997.1.
			Note: This parameter is OPTIONAL at the G and S/T interfaces in G.997.1 Amendment 1. If the parameter is implemented but no value is available, its value MUST be an empty string.
Device.DSL.Line.{i}.CurrentProfile	string	-	Defines which VDSL2 profile is currently in use on the line.The value MUST be a member of the list reported by the AllowedProfiles parameter, or else be an empty string.
			Note: This parameter is OPTIONAL at the G and S/T interfaces in G.997.1 Amendment 1. If the parameter is implemented but no value is available, its value MUST be an empty string.
Device.DSL.Line.{i}.UpstreamMaxBitRate	unsignedInt	-	Defines the current maximum attainable data rate upstream (expressed in Kbps).
			Note: This parameter is related to the G.997.1 parameter ATTNDRus, which is measured in bits/s. See ITU-T Recommendation [G.997.1].

Name	Type	Write	Description
Device.DSL.Line.{i}.DownstreamMaxBitRate	unsignedInt	-	Defines the current maximum attainable data rate downstream (expressed in Kbps). **Note:** This parameter is related to the G.997.1 parameter ATTNDRds, which is measured in bits/s. See ITU-T Recommendation G.997.1.
Device.DSL.Line.{i}.UpstreamNoiseMargin	integer	-	The current signal-to-noise ratio margin (expressed in 0.1dB) in the upstream direction. **Note:** In G.997.1, this parameter is called SNRMus. See ITU-T Recommendation G.997.1.
Device.DSL.Line.{i}.DownstreamNoiseMargin	integer	-	Defines the current signal-to-noise ratio margin (expressed in 0.1dB) in the downstream direction. **Note:** In G.997.1, this parameter is called SNRMds. See ITU-T Recommendation G.997.1.
Device.DSL.Line.{i}.UpstreamAttenuation	integer	-	Defines the current upstream signal loss (expressed in 0.1dB).
Device.DSL.Line.{i}.DownstreamAttenuation	integer	-	Defines the current downstream signal loss (expressed in 0.1dB).
Device.DSL.Line.{i}.UpstreamPower	integer	-	Defines the current output power at the CPE's DSL line (expressed in 0.1dBmV).
Device.DSL.Line.{i}.DownstreamPower	integer	-	Defines the current received power at the CPE's DSL line (expressed in 0.1dBmV).
Device.DSL.Line.{i}.Stats.	object	-	Defines throughput statistics for this interface. The CPE MUST reset the interface's Stats parameters (unless otherwise stated in individual object or parameter descriptions) either when the interface becomes operationally down due to a previous administrative down (i.e. the interface's Status parameter transitions to a down state after the interface is disabled) or when the interface becomes administratively up (i.e. the interface's Enable parameter transitions from false to true). Administrative and operational interface status is discussed in Section 4.2.2 in TR-181i2.
Device.DSL.Line.{i}.Stats.BytesSent	unsignedLong	-	Defines the total number of bytes transmitted out of the interface, including framing characters.
Device.DSL.Line.{i}.Stats.BytesReceived	unsignedLong	-	Defines the total number of bytes received on the interface, including framing characters.

Name	Type	Write	Description
Device.DSL.Line.{i}.Stats.PacketsSent	unsignedLong	-	Defines the total number of packets transmitted out of the interface.
Device.DSL.Line.{i}.Stats.PacketsReceived	unsignedLong	-	Defines the total number of packets received on the interface.
Device.DSL.Line.{i}.Stats.ErrorsSent	unsignedInt	-	Defines the total number of outbound packets that could not be transmitted because of errors.
Device.DSL.Line.{i}.Stats.ErrorsReceived	unsignedInt	-	Defines the total number of inbound packets that contained errors preventing them from being delivered to a higher-layer protocol.
Device.DSL.Line.{i}.Stats.DiscardPacketsSent	unsignedInt	-	Defines the total number of outbound packets which were chosen to be discarded even though no errors had been detected to prevent their being transmitted. One possible reason for discarding such a packet could be to free up buffer space.
Device.DSL.Line.{i}.Stats.DiscardPacketsReceived	unsignedInt	-	Defines the total number of inbound packets which were chosen to be discarded even though no errors had been detected to prevent their being delivered. One possible reason for discarding such a packet could be to free up buffer space.
Device.DSL.Line.{i}.Stats.TotalStart	unsignedInt	-	Defines the Number of seconds since the beginning of the period used for collection of Total statistics. This is a DSL-specific statistic. Statistics SHOULD continue to be accumulated across CPE reboots, though this might not always be possible. **Note:** TotalStart SHOULD NOT be reset when the interface statistics are reset via an interface disable / enable cycle.
Device.DSL.Line.{i}.Stats.ShowtimeStart	unsignedInt	-	Defines the Number of seconds since the most recent DSL Showtime - the beginning of the period used for collection of Showtime statistics. This is a DSL-specific statistic. Showtime is defined as successful completion of the DSL link establishment process. The Showtime statistics are those collected since the most recent establishment of the DSL link. **Note:** ShowtimeStart SHOULD NOT be reset when the interface statistics are reset via an interface disable / enable cycle.

Name	Type	Write	Description
Device.DSL.Line.{i}.Stats.LastShowtimeStart	unsignedInt	-	Defines the number of seconds since the second most recent DSL Showtime-the beginning of the period used for collection of LastShowtime statistics. This is a DSL-specific statistic. If the CPE has not retained information about the second most recent Showtime (e.g., on reboot), the start of LastShowtime statistics MAY temporarily coincide with the start of Showtime statistics. **Note:** LastShowtimeStart SHOULD NOT be reset when the interface statistics are reset via an interface disable / enable cycle.
Device.DSL.Line.{i}.Stats.CurrentDayStart	unsignedInt	-	Defines the number of seconds since the beginning of the period used for collection of CurrentDay statistics. This is a DSL-specific statistic. The CPE may align the beginning of each CurrentDay interval with days in the UTC time zone, but is not required to do so. Statistics should continue to be accumulated across CPE reboots, though this might not always be possible. **Note:** CurrentDayStart SHOULD NOT be reset when the interface statistics are reset via an interface disable / enable cycle.
Device.DSL.Line.{i}.Stats.QuarterHourStart	unsignedInt	-	Defines the number of seconds since the beginning of the period used for collection of QuarterHour statistics. This is a DSL-specific statistic. The CPE MAY align the beginning of each QuarterHour interval with real-time quarter-hour intervals, but is not required to do so. Statistics SHOULD continue to be accumulated across CPE reboots, though this might not always be possible. **Note:** QuarterHourStart SHOULD NOT be reset when the interface statistics are reset via an interface disable / enable cycle.
Device.DSL.Line.{i}.Stats.Showtime.	object	-	This object contains DSL line statistics accumulated since the most recent DSL Showtime. See [Chapter 7.2.6/G.997.1]. **Note:** The Showtime parameters SHOULD NOT be reset when the interface statistics are reset via an interface disable / enable cycle.

Name	Type	Write	Description
Device.DSL.Line.{i}.Stats.Showtime.ErroredSecs	unsignedInt	-	Defines the number of errored seconds since the most recent DSL Showtime (ES-L as defined in ITU-T Rec. G.997.1). **Note:** This parameter is OPTIONAL at the G and S/T interfaces in G.997.1 Amendment 1. If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Line.{i}.Stats.Showtime.SeverelyErroredSecs	unsignedInt	-	Defines the number of severely errored seconds since the most recent DSL Showtime (SES-L as defined in ITU-T Rec. [G.997.1]). **Note:** This parameter is OPTIONAL at the G and S/T interfaces in G.997.1 Amendment 1. If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Line.{i}.Stats.LastShowtime.	object	-	This object contains DSL line statistics accumulated since the second most recent DSL Showtime. See [Chapter 7.2.6/G.997.1]. **Note:** The LastShowtime parameters SHOULD NOT be reset when the interface statistics are reset via an interface disable / enable cycle.
Device.DSL.Line.{i}.Stats.LastShowtime.ErroredSecs	unsignedInt	-	Defines the number of errored seconds since the second most recent DSL Showtime (ES-L as defined in ITU-T Rec. G.997.1]). **Note:** This parameter is OPTIONAL at the G and S/T interfaces in G.997.1 Amendment 1. If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Line.{i}.Stats.LastShowtime.SeverelyErroredSecs	unsignedInt	-	Defines the number of severely errored seconds since the second most recent DSL Showtime (SES-L as defined in ITU-T Rec. G.997.1). **Note:** This parameter is OPTIONAL at the G and S/T interfaces in G.997.1 Amendment 1. If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Line.{i}.Stats.CurrentDay.	object	-	Defines the number of severely errored seconds since the second most recent DSL Showtime (SES-L as defined in ITU-T Rec. G.997.1). **Note:** This parameter is OPTIONAL at the G and S/T interfaces in G.997.1

Name	Type	Write	Description
			Amendment 1. If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Line.{i}.Stats.CurrentDay.ErroredSecs	unsignedInt	-	Defines the number of errored seconds since the second most recent DSL Showtime (ES-L as defined in ITU-T Rec. G.997.1). **Note:** This parameter is OPTIONAL at the G and S/T interfaces in G.997.1 Amendment 1. If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Line.{i}.Stats.CurrentDay.SeverelyErroredSecs	unsignedInt	-	Defines the number of severely errored seconds since the second most recent DSL Showtime (SES-L as defined in ITU-T Rec. G.997.1). **Note:** This parameter is OPTIONAL at the G and S/T interfaces in G.997.1 Amendment 1. If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Line.{i}.Stats.QuarterHour.	object	-	This object contains DSL line statistics accumulated during the current quarter hour. See Chapter 7.2.6 of G.997.1. **Note:** The QuarterHour parameters SHOULD NOT be reset when the interface statistics are reset via an interface disable / enable cycle.
Device.DSL.Line.{i}.Stats.QuarterHour.ErroredSecs	unsignedInt	-	Defines the number of errored seconds since the second most recent DSL Showtime (ES-L as defined in ITU-T Rec. G.997.1). **Note:** This parameter is optional at the G and S/T interfaces in G.997.1 Amendment 1. If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Line.{i}.Stats.QuarterHour.SeverelyErroredSecs	unsignedInt	-	Defines the number of severely errored seconds since the second most recent DSL Showtime (SES-L as defined in ITU-T Rec. G.997.1). **Note:** This parameter is OPTIONAL at the G and S/T interfaces in G.997.1 Amendment 1. If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).

Name	Type	Write	Description
			mechanism for an ACS to label this instance for future reference.
			If the CPE supports the Alias-based Addressing feature as defined in [Section 3.6.1/TR-069a4] and described in [Appendix II/TR-069a4], the following mandatory constraints MUST be enforced:
			▪ Its value MUST NOT be empty.
			▪ Its value MUST start with a letter.
			▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix.
			▪ The CPE MUST NOT change the parameter value.
Device.DSL.Channel.{i}.Name	string	-	Defines the textual name of the channel as assigned by the CPE.
Device.DSL.Channel.{i}.LastChange	unsignedInt	-	Defines the accumulated time in seconds since the channel entered its current operational state.
Device.DSL.Channel.{i}.LowerLayers	string	-	Each list item MUST be the path name of an interface object that is stacked immediately below this interface object. This is a comma-separated list (maximum list length 1024) of strings. If the referenced object is deleted, the corresponding item MUST be removed from the list. See Section 4.2.1 of TR-181i2.
Device.DSL.Channel.{i}.LinkEncapsulationSupported	string	-	List items indicate which link encapsulation standards and recommendations are supported by the Channel instance. This is a comma-separated list of strings. Each list item is an enumeration of: ▪ G.992.3_Annex_K_ATM ▪ G.992.3_Annex_K_PTM ▪ G.993.2_Annex_K_ATM ▪ G.993.2_Annex_K_PTM G.994.1 (Auto)
Device.DSL.Channel.{i}.LinkEncapsulationUsed	string	-	Indicates the link encapsulation standard that the Channel instance is using for the connection. Possible values: ▪ G.992.3_Annex_K_ATM ▪ G.992.3_Annex_K_PTM ▪ G.993.2_Annex_K_ATM ▪ G.993.2_Annex_K_PTM

Name	Type	Write	Description
			When ATM encapsulation is identified then an upper-layer ATM.Link interface MUST be used. When PTM encapsulation is identified then an upper-layer PTM.Link interface MUST be used.
Device.DSL.Channel.{i}.UpstreamCurrRate	unsignedInt	-	Defines the current physical layer aggregate data rate (expressed in Kbps) of the upstream DSL connection. **Note:** If the parameter is implemented but no value is available, it MUST have the value 4294967295 (the maximum for its data type).
Device.DSL.Channel.{i}.DownstreamCurrRate	unsignedInt	-	Defines the current physical layer aggregate data rate (expressed in Kbps) of the downstream DSL connection. **Note:** If the parameter is implemented but no value is available, it MUST have the value 4294967295 (the maximum for its data type).
Device.DSL.Channel.{i}.ACTNDR	unsignedInt	-	Defines the actual net data rate expressed in Kbps. Independent whether retransmission is used or not in a given transmit direction: ▪ In L0 state, this parameter reports the net data rate (as specified in [ITU T G.992.3], [ITU T G.992.5] or [ITU T G.993.2]) at which the bearer channel is operating. ▪ In L2 state, the parameter contains the net data rate (as specified in [ITU T G.992.3], [ITU T G.992.5] or [ITU T G.993.2]) in the previous L0 state. ▪ **Note:** See ITU-T Recommendation G.997.1.
Device.DSL.Channel.{i}.Stats.	object	-	Defines throughput statistics for this interface. The CPE MUST reset the interface's Stats parameters (unless otherwise stated in individual object or parameter descriptions) either when the interface becomes operationally down due to a previous administrative down (i.e. the interface's Status parameter transitions to a down state after the interface is disabled) or when the interface becomes administratively up (i.e. the interface's Enable parameter transitions from false to true). Administrative and operational interface status is discussed in [Section 4.2.2 of TR-181i2].

Name	Type	Write	Description
Device.DSL.Channel.{i}.Stats.BytesSent	unsignedLong	-	Defines the total number of bytes transmitted out of the interface, including framing characters.
Device.DSL.Channel.{i}.Stats.BytesReceived	unsignedLong	-	Defines the total number of bytes received on the interface, including framing characters.
Device.DSL.Channel.{i}.Stats.PacketsSent	unsignedLong	-	Defines the total number of packets transmitted out of the interface.
Device.DSL.Channel.{i}.Stats.PacketsReceived	unsignedLong	-	Defines the total number of packets received on the interface.
Device.DSL.Channel.{i}.Stats.ErrorsSent	unsignedInt	-	Defines the total number of outbound packets that could not be transmitted because of errors.
Device.DSL.Channel.{i}.Stats.ErrorsReceived	unsignedInt	-	Defines the total number of inbound packets that contained errors preventing them from being delivered to a higher-layer protocol.
Device.DSL.Channel.{i}.Stats.DiscardPacketsSent	unsignedInt	-	Defines the total number of outbound packets which were chosen to be discarded even though no errors had been detected to prevent their being transmitted. One possible reason for discarding such a packet could be to free up buffer space.
Device.DSL.Channel.{i}.Stats.DiscardPacketsReceived	unsignedInt	-	Defines the total number of inbound packets which were chosen to be discarded even though no errors had been detected to prevent their being delivered. One possible reason for discarding such a packet could be to free up buffer space.
Device.DSL.Channel.{i}.Stats.TotalStart	unsignedInt	-	Defines the Number of seconds since the beginning of the period used for collection of Total statistics. This is a DSL-specific statistic. Statistics should continue to be accumulated across CPE reboots, though this might not always be possible. **Note:** TotalStart SHOULD NOT be reset when the interface statistics are reset via an interface disable / enable cycle.
Device.DSL.Channel.{i}.Stats.ShowtimeStart	unsignedInt	-	Defines the Number of seconds since the most recent DSL Showtime - the beginning of the period used for collection of Showtime statistics. Showtime is defined as successful completion of the DSL link establishment process. The Showtime statistics are

Name	Type	Write	Description
			those collected since the most recent establishment of the DSL link. This is a DSL-specific statistic. **Note:** ShowtimeStart SHOULD NOT be reset when the interface statistics are reset via an interface disable / enable cycle.
Device.DSL.Channel.{i}.Stats.Showtime.	object	-	This object contains DSL channel statistics accumulated since the most recent DSL Showtime G.997.1. **Note:** The Showtime parameters SHOULD NOT be reset when the interface statistics are reset via an interface disable / enable cycle.
Device.DSL.Channel.{i}.Stats.Showtime.XTURFECErrors	unsignedInt	-	Defines the number of FEC errors detected since the most recent DSL Showtime (FEC-C as defined in ITU-T Rec. G.997.1). **Note:** If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Channel.{i}.Stats.Showtime.XTUCFECErrors	unsignedInt	-	Defines the number of FEC errors detected by the ATU-C since the most recent DSL Showtime (FEC-CFE as defined in ITU-T Rec. G.997.1). **Note:** If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Channel.{i}.Stats.Showtime.XTURHECErrors	unsignedInt	-	Defines the number of HEC errors detected since the most recent DSL Showtime (HEC-P as defined in ITU-T Rec. [G.997.1]). **Note:** If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Channel.{i}.Stats.Showtime.XTUCHECErrors	unsignedInt	-	Defines the number of HEC errors detected by the ATU-C since the most recent DSL Showtime (HEC-PFE as defined in ITU-T Rec. G.997.1). **Note:** If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Channel.{i}.Stats.Showtime.XTURCRCErrors	unsignedInt	-	Defines the number of CRC errors detected since the most recent DSL Showtime (CV-C as defined in ITU-T Rec. G.997.1). **Note:** If the parameter is implemented but no value is available, its value MUST

Name	Type	Write	Description
			be 4294967295 (the maximum for its data type).
Device.DSL.Channel.{i}.Stats.Showtime.XTUCCRCErrors	unsignedInt	-	Defines the number of CRC errors detected by the ATU-C since the most recent DSL Showtime (CV-CFE as defined in ITU-T Rec. G.997.1). **Note:** If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Channel.{i}.Stats.CurrentDay.	object	-	This object contains DSL channel statistics accumulated during the current day G.997.1. **Note:** The CurrentDay parameters SHOULD NOT be reset when the interface statistics are reset via an interface disable / enable cycle.
Device.DSL.Channel.{i}.Stats.CurrentDay.XTURFECErrors	unsignedInt	-	Defines the number of FEC errors detected since the second most recent DSL Showtime (FEC-C as defined in ITU-T Rec. G.997.1). **Note:** If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Channel.{i}.Stats.CurrentDay.XTUCFECErrors	unsignedInt	-	Defines the number of FEC errors detected by the ATU-C since the second most recent DSL Showtime (FEC-CFE as defined in ITU-T Rec. G.997.1). **Note:** If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Channel.{i}.Stats.CurrentDay.XTURHECErrors	unsignedInt	-	Defines the number of HEC errors detected since the second most recent DSL Showtime (HEC-P as defined in ITU-T Rec. G.997.1). **Note:** If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Channel.{i}.Stats.CurrentDay.XTUCHECErrors	unsignedInt	-	Defines the number of HEC errors detected by the ATU-C since the second most recent DSL Showtime (HEC-PFE as defined in ITU-T Rec. G.997.1). **Note:** If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).

Name	Type	Write	Description
Device.DSL.Channel.{i}.Stats.CurrentDay.XTURCRCErrors	unsignedInt	-	Defines the number of CRC errors detected since the second most recent DSL Showtime (CV-C as defined in ITU-T Rec. G.997.1). **Note:** If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Channel.{i}.Stats.CurrentDay.XTUCCRCErrors	unsignedInt	-	Defines the number of CRC errors detected by the ATU-C since the second most recent DSL Showtime (CV-CFE as defined in ITU-T Rec. G.997.1). **Note:** If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Channel.{i}.Stats.QuarterHour.	object	-	This object contains DSL channel statistics accumulated during the current quarter hour G.997.1. **Note:** The QuarterHour parameters SHOULD NOT be reset when the interface statistics are reset via an interface disable / enable cycle.
Device.DSL.Channel.{i}.Stats.QuarterHour.XTURFECErrors	unsignedInt	-	Defines the number of FEC errors detected since the second most recent DSL Showtime (FEC-C as defined in ITU-T Rec. G.997.1). **Note:** If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Channel.{i}.Stats.QuarterHour.XTUCFECErrors	unsignedInt	-	Defines the number of FEC errors detected by the ATU-C since the second most recent DSL Showtime (FEC-CFE as defined in ITU-T Rec. [G.997.1]). **Note:** If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Channel.{i}.Stats.QuarterHour.XTURHECErrors	unsignedInt	-	Defines the number of HEC errors detected since the second most recent DSL Showtime (HEC-P as defined in ITU-T Rec. G.997.1]). **Note:** If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Channel.{i}.Stats.QuarterHour.XTUCHECErrors	unsignedInt	-	Defines the number of HEC errors detected by the ATU-C since the second most recent DSL Showtime (HEC-PFE as defined in ITU-T Rec. G.997.1).

Name	Type	Write	Description
			Note: If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Channel.{i}.Stats.QuarterHour.XTURCRCErrors	unsignedInt	-	Defines the number of CRC errors detected since the second most recent DSL Showtime (CV-C as defined in ITU-T Rec. G.997.1). **Note:** If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.DSL.Channel.{i}.Stats.QuarterHour.XTUCCRCErrors	unsignedInt	-	Defines the number of CRC errors detected by the ATU-C since the second most recent DSL Showtime (CV-CFE as defined in ITU-T Rec. G.997.1). **Note:** If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
Device.ATM.	object	-	Defines the Asynchronous Transfer Mode (ATM) object that contains the Link interface and Diagnostics.F5Loopback diagnostics.
Device.ATM.LinkNumberOfEntries	unsignedInt	-	Defines the number of entries in the Link table.
Device.ATM.Link.{i}.	object	-	Defines the ATM link-layer table (a stackable interface object as described in Section 4.2 of TR-181i2). Models an ATM PVC virtual circuit and the ATM Adaption Layer (AAL). An ATM Link entry is typically stacked on top of either a DSL.Channel. or a DSL.BondingGroup. object. When an ATM Link interface is used, a lower-layer DSL.Channel interface MUST be configured with ATM encapsulation (see DSL.Channel.{i}.LinkEncapsulationUsed). At most one entry in this table (regardless of whether or not it is enabled) can exist with a given value for Alias, or with a given value for Name. On creation of a new table entry, the CPE MUST choose initial values for Alias and Name such that the new entry does not conflict with any existing entries.
Device.ATM.Link.{i}.Enable	boolean	W	Enables or disables the link. This parameter is based on ifAdminStatus from RFC 2863.

Name	Type	Write	Description
Device.ATM.Link.{i}.Status	string	-	Defines the current operational state of the link (see Section 4.2.2 of TR-181i2). Possible values: • Up • Down • Unknown • Dormant • NotPresent • LowerLayerDown • Error (OPTIONAL) When Enable is "false" then Status should normally be Down (or NotPresent or Error if there is a fault condition on the interface). When Enable is changed to" true" then Status should change to Up if and only if the interface is able to transmit and receive network traffic; it should change to Dormant if and only if the interface is operable but is waiting for external actions before it can transmit and receive network traffic (and subsequently change to Up if still operable when the expected actions have completed); it should change to LowerLayerDown if and only if the interface is prevented from entering the Up state because one or more of the interfaces beneath it is down; it should remain in the Error state if there is an error or other fault condition detected on the interface; it should remain in the NotPresent state if the interface has missing (typically hardware) components; it should change to Unknown if the state of the interface can not be determined for some reason. This parameter is based on ifOperStatus from [RFC 2863].
Device.ATM.Link.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in [Section 3.6.1/TR-069a4] and described in [Appendix II/TR-069a4], the following mandatory constraints MUST be enforced:: • Its value MUST NOT be empty. • Its value MUST start with a letter. • If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix.

Name	Type	Write	Description
			▪ The CPE MUST NOT change the parameter value.
Device.ATM.Link.{i}.Name	string	-	Defines the textual name of the link as assigned by the CPE.
Device.ATM.Link.{i}.LastChange	unsignedInt	-	The accumulated time in seconds since the link entered its current operational state.
Device.ATM.Link.{i}.LowerLayers	string	W	Each list item MUST be the path name of an interface object that is stacked immediately below this interface object. If the referenced object is deleted, the corresponding item MUST be removed from the list. This is a comma-separated list (maximum list length 1024) of strings. See Section 4.2.1 of TR-181i2.
Device.ATM.Link.{i}.LinkType	string	W	Indicates the type of connection and refers to the complete stack of protocol used for this connection.
Device.ATM.Link.{i}.AutoConfig	boolean	-	Indicates if the CPE is currently using some auto configuration mechanisms for this connection. If this variable is true, all writable variables in this connection instance become -. Any attempt to change one of these variables SHOULD fail and an error SHOULD be returned.
Device.ATM.Link.{i}.DestinationAddress	string	W	Defines the destination address of this link, in the form "VPI/VCI" (e.g. "8/23" or "0/35").
Device.ATM.Link.{i}.Encapsulation	string	W	Defines the Identifies the connection encapsulation that will be used.
Device.ATM.Link.{i}.AAL	string	-	Describes the ATM Adaptation Layer (AAL) currently in use on the PVC.
Device.ATM.Link.{i}.Stats.	object	-	Defines throughput statistics for this interface. The CPE MUST reset the interface's Stats parameters (unless otherwise stated in individual object or parameter descriptions) either when the interface becomes operationally down due to a previous administrative down (i.e. the interface's Status parameter transitions to a down state after the interface is disabled) or when the interface becomes administratively up (i.e. the interface's Enable parameter transitions from false to true). Administrative and operational interface status is discussed in Section 4.2.2 of TR-181i2.

Name	Type	Write	Description
Device.ATM.Link.{i}.Stats.BytesSent	unsignedLong	-	Defines the total number of bytes transmitted out of the interface, including framing characters.
Device.ATM.Link.{i}.Stats.BytesReceived	unsignedLong	-	Defines the total number of bytes received on the interface, including framing characters.
Device.ATM.Link.{i}.Stats.PacketsSent	unsignedLong	-	Defines the total number of packets transmitted out of the interface.
Device.ATM.Link.{i}.Stats.PacketsReceived	unsignedLong	-	Defines the total number of packets received on the interface.
Device.ATM.Link.{i}.Stats.ErrorsSent	unsignedInt	-	Defines the total number of outbound packets that could not be transmitted because of errors.
Device.ATM.Link.{i}.Stats.ErrorsReceived	unsignedInt	-	Defines the total number of inbound packets that contained errors preventing them from being delivered to a higher-layer protocol.
Device.ATM.Link.{i}.Stats.DiscardPacketsSent	unsignedInt	-	Defines the total number of outbound packets which were chosen to be discarded even though no errors had been detected to prevent their being transmitted. One possible reason for discarding such a packet could be to free up buffer space.
Device.ATM.Link.{i}.Stats.DiscardPacketsReceived	unsignedInt	-	The total number of inbound packets which were chosen to be discarded even though no errors had been detected to prevent their being delivered. One possible reason for discarding such a packet could be to free up buffer space.
Device.ATM.Link.{i}.Stats.CRCErrors	unsignedInt	-	Defines the count of the ATM layer cyclic redundancy check (CRC) errors. This refers to CRC errors at the ATM adaptation layer (AAL). The AAL in use is indicated by the AAL parameter. The value of the CRCErrors parameter MUST be 0 for AAL types that have no CRCs.
Device.ATM.Link.{i}.Stats.HECErrors	unsignedInt	-	Defines the count of the number of Header Error Check related errors at the ATM layer.
Device.ATM.Link.{i}.QoS.	object	-	Defines the ATM Link QoS object.
Device.ATM.Link.{i}.QoS.QoSClass	string	W	Describes the ATM Quality Of Service (QoS) being used on the VC.
Device.ATM.Link.{i}.QoS.PeakCellRate	unsignedInt	W	Specifies the upstream peak cell rate in cells per second.
Device.ATM.Link.{i}.QoS.MaximumBurstSize	unsignedInt	W	Specifies the upstream maximum burst size in cells.

Name	Type	Write	Description
Device.ATM.Link.{i}.QoS.SustainableCellRate	unsignedInt	W	Specifies the upstream sustainable cell rate, in cells per second.
Device.Ethernet.	object	-	Defines the Ethernet object. This object models several Ethernet interface objects, each representing a different stack layer, including: Interface, Link, and VLANTermination. Interface is media-specific and models a port, the PHY layer, and the MAC layer. Link is media-independent and models the Logical Link Control (LLC) layer. An "outer" VLANTermination, when present, is expected to be stacked on top of Link objects to receive and send frames with a configured VLANID.
Device.Ethernet.InterfaceNumberOfEntries	unsignedInt	-	Defines the number of entries in the Interface table.
Device.Ethernet.LinkNumberOfEntries	unsignedInt	-	Defines the number of entries in the Link table.
Device.Ethernet.VLANTerminationNumberOfEntries	unsignedInt	-	Defines the number of entries in the VLANTermination table.
Device.Ethernet.RMONStatsNumberOfEntries	unsignedInt	-	Defines the number of entries in the RMONStats table.
Device.Ethernet.Interface.{i}.	object	-	Defines the Ethernet interface table (a stackable interface object as described in Section 4.2 of TR-181i2). This table models physical Ethernet ports, but in terms of the interface stack it also models the PHY and MAC level Ethernet interface. At most one entry in this table can exist with a given value for Alias, or with a given value for Name.
Device.Ethernet.Interface.{i}.Enable	boolean	W	Enables or disables the interface. This parameter is based on ifAdminStatus from [RFC 2863].
Device.Ethernet.Interface.{i}.Status	string	-	Defines the current operational state of the interface (see Section 4.2.2 of TR-181i2). Possible values: ▪ Up ▪ Down ▪ Unknown ▪ Dormant ▪ NotPresent ▪ LowerLayerDown ▪ Error (Optional) When Enable is "false" then Status SHOULD normally be Down (or

Name	Type	Write	Description
			NotPresent or Error if there is a fault condition on the interface).
			When Enable is changed to "true" then Status should change to Up if and only if the interface is able to transmit and receive network traffic; it should change to Dormant if and only if the interface is operable but is waiting for external actions before it can transmit and receive network traffic (and subsequently change to Up if still operable when the expected actions have completed); it should change to LowerLayerDown if and only if the interface is prevented from entering the Up state because one or more of the interfaces beneath it is down; it should remain in the Error state if there is an error or other fault condition detected on the interface; it should remain in the NotPresent state if the interface has missing (typically hardware) components; it should change to Unknown if the state of the interface can not be determined for some reason.
			This parameter is based on ifOperStatus from [RFC 2863].
Device.Ethernet.Interface.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference.
			If the CPE supports the Alias-based Addressing feature as defined in [Section 3.6.1/TR-069a4] and described in [Appendix II/TR-069a4], the following mandatory constraints MUST be enforced:
			▪ Its value MUST NOT be empty.
			▪ Its value MUST start with a letter.
			▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix.
			▪ The CPE MUST NOT change the parameter value.
Device.Ethernet.Interface.{i}.Name	string	-	Defines the textual name of the interface as assigned by the CPE.
Device.Ethernet.Interface.{i}.LastChange	unsignedInt	-	Defines the accumulated time in seconds since the interface entered its current operational state.
Device.Ethernet.Interface.{i}.LowerLayers	string	W	Each list item MUST be the path name of an interface object that is stacked immediately below this interface object. If the referenced object is deleted, the corresponding item MUST be removed from the list.

Name	Type	Write	Description
			This is a comma-separated list (maximum list length 1024) of strings. See Section 4.2.1 of TR-181i2. **Note:** Since Interface is a layer 1 interface, it is expected that LowerLayers will not be used.
Device.Ethernet.Interface.{i}.Upstream	boolean	-	Defines whether the interface points towards the Internet (true) or towards End Devices (false). For example: ▪ For an Internet Gateway Device, Upstream will be true for all WAN interfaces and false for all LAN interfaces. ▪ For a standalone WiFi Access Point that is connected via Ethernet to an Internet Gateway Device, Upstream will be true for the Ethernet interface and false for the WiFi Radio interface. ▪ For an End Device, Upstream will be true for all interfaces.
Device.Ethernet.Interface.{i}.MACAddress	string	-	Defines the MAC Address of the interface. **Note:** This is not necessarily the same as the Ethernet header source or destination MAC address, which is associated with the IP interface and is modeled via the Ethernet.Link.{i}.MACAddress parameter.
Device.Ethernet.Interface.{i}.MaxBitRate	integer	W	Defines the maximum upstream and downstream PHY bit rate supported by this interface (expressed in Mbps). A value of -1 indicates automatic selection of the maximum bit rate.
Device.Ethernet.Interface.{i}.CurrentBitRate	unsignedInt	-	Defines the current upstream and downstream PHY bit rate on this interface (expressed in Mbps). A value of 0 indicates that the current bit rate is unknown.
Device.Ethernet.Interface.{i}.DuplexMode	string	W	Defines the duplex mode available to this connection.
Device.Ethernet.Interface.{i}.Stats.	object	-	Defines the throughput statistics for this interface. The CPE MUST reset the interface's Stats parameters (unless otherwise stated in individual object or parameter descriptions) either when the interface becomes operationally down due to a previous administrative down (i.e. the interface's Status parameter transitions to a down state after the interface is

Name	Type	Write	Description
			to a multicast address at this layer, including those that were discarded or not sent.
Device.Ethernet.Interface.{i}.Stats.MulticastPacketsReceived	unsignedLong	-	Defines the The total number of received packets, delivered by this layer to a higher layer, which were addressed to a multicast address at this layer.
Device.Ethernet.Interface.{i}.Stats.BroadcastPacketsSent	unsignedLong	-	Defines the total number of packets that higher-level protocols requested for transmission and which were addressed to a broadcast address at this layer, including those that were discarded or not sent.
Device.Ethernet.Interface.{i}.Stats.BroadcastPacketsReceived	unsignedLong	-	Defines the total number of received packets, delivered by this layer to a higher layer, which were addressed to a broadcast address at this layer.
Device.Ethernet.Link.{i}.	object	-	Defines an Ethernet link layer table (a stackable interface object as described in Section 4.2 of TR-181i2). Table entries model the Logical Link Control (LLC) layer. It is expected that an Ethernet Link interface can be stacked above any lower-layer interface object capable of carrying Ethernet frames. At most one entry in this table (regardless of whether or not it is enabled) can exist with a given value for Alias, or with a given value for Name. On creation of a new table entry, the CPE MUST choose initial values for Alias and Name such that the new entry does not conflict with any existing entries. At most one enabled entry in this table can exist with a given value for MACAddress.
Device.Ethernet.Link.{i}.Enable	boolean	-	Enables or disables the link. This parameter is based on ifAdminStatus from [RFC 2863].
Device.Ethernet.Link.{i}.Status	string	-	Defines the current operational state of the link (see Section 4.2.2 of TR-181i2). Possible values: ▪ Up ▪ Down ▪ Unknown ▪ Dormant ▪ NotPresent ▪ LowerLayerDown ▪ Error (Optional)

Name	Type	Write	Description
			When Enable is "false" then Status SHOULD normally be Down (or NotPresent or Error if there is a fault condition on the interface). When Enable is changed to "true" then Status should change to Up if and only if the interface is able to transmit and receive network traffic; it should change to Dormant if and only if the interface is operable but is waiting for external actions before it can transmit and receive network traffic (and subsequently change to Up if still operable when the expected actions have completed); it should change to LowerLayerDown if and only if the interface is prevented from entering the Up state because one or more of the interfaces beneath it is down; it should remain in the Error state if there is an error or other fault condition detected on the interface; it should remain in the NotPresent state if the interface has missing (typically hardware) components; it should change to Unknown if the state of the interface can not be determined for some reason. This parameter is based on ifOperStatus from RFC 2863.
Device.Ethernet.Link.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in [Section 3.6.1/TR-069a4] and described in [Appendix II/TR-069a4], the following mandatory constraints must be enforced: ▪ Its value MUST NOT be empty. ▪ Its value MUST start with a letter. ▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. ▪ The CPE MUST NOT change the parameter value.
Device.Ethernet.Link.{i}.Name	string	-	Defines the textual name of the link as assigned by the CPE.
Device.Ethernet.Link.{i}.LastChange	unsignedInt	-	Defines the accumulated time in seconds since the link entered its current operational state.
Device.Ethernet.Link.{i}.LowerLayers	string	W	Each list item MUST be the path name of an interface object that is stacked immediately below this interface object. This is a comma-separated list (maximum list length 1024) of strings.

Name	Type	Write	Description
			If the referenced object is deleted, the corresponding item MUST be removed from the list. See Section 4.2.1 of TR-181i2.
Device.Ethernet.Link.{i}.MACAddress	string	-	Defines the MAC address used for packets sent via this interface. Provides the source MAC address for outgoing traffic and the destination MAC address for incoming traffic.
Device.Ethernet.Link.{i}.Stats.	object	-	Defines throughput statistics for this interface. The CPE MUST reset the interface's Stats parameters (unless otherwise stated in individual object or parameter descriptions) either when the interface becomes operationally down due to a previous administrative down (i.e. the interface's Status parameter transitions to a down state after the interface is disabled) or when the interface becomes administratively up (i.e. the interface's Enable parameter transitions from false to true). Administrative and operational interface status is discussed in Section 4.2.2 of TR-181i2.
Device.Ethernet.Link.{i}.Stats.BytesSent	unsignedLong	-	Defines the total number of bytes transmitted out of the interface, including framing characters.
Device.Ethernet.Link.{i}.Stats.BytesReceived	unsignedLong	-	Defines the total number of bytes received on the interface, including framing characters.
Device.Ethernet.Link.{i}.Stats.PacketsSent	unsignedLong	-	Defines the total number of packets transmitted out of the interface.
Device.Ethernet.Link.{i}.Stats.PacketsReceived	unsignedLong	-	Defines the total number of packets received on the interface.
Device.Ethernet.Link.{i}.Stats.ErrorsSent	unsignedInt	-	Defines the total number of outbound packets that could not be transmitted because of errors.
Device.Ethernet.Link.{i}.Stats.ErrorsReceived	unsignedInt	-	Defines the total number of inbound packets that contained errors preventing them from being delivered to a higher-layer protocol.
Device.Ethernet.Link.{i}.Stats.UnicastPacketsSent	unsignedLong	-	Defines the total number of packets requested for transmission which were not addressed to a multicast or broadcast address at this layer, including those that were discarded or not sent.
Device.Ethernet.Link.{i}.Stats.UnicastPacketsReceived	unsignedLong	-	Defines the total number of received packets, delivered by this layer to a higher layer, which were not addressed

Name	Type	Write	Description
			to a multicast or broadcast address at this layer.
Device.Ethernet.Link.{i}.Stats.DiscardPacketsSent	unsignedInt	-	Defines the total number of outbound packets which were chosen to be discarded even though no errors had been detected to prevent their being transmitted. One possible reason for discarding such a packet could be to free up buffer space.
Device.Ethernet.Link.{i}.Stats.DiscardPacketsReceived	unsignedInt	-	Defines the total number of inbound packets which were chosen to be discarded even though no errors had been detected to prevent their being delivered. One possible reason for discarding such a packet could be to free up buffer space.
Device.Ethernet.Link.{i}.Stats.MulticastPacketsSent	unsignedLong	-	Defines the total number of packets that higher-level protocols requested for transmission and which were addressed to a multicast address at this layer, including those that were discarded or not sent.
Device.Ethernet.Link.{i}.Stats.MulticastPacketsReceived	unsignedLong	-	Defines the total number of received packets, delivered by this layer to a higher layer, which were addressed to a multicast address at this layer.
Device.Ethernet.Link.{i}.Stats.BroadcastPacketsSent	unsignedLong	-	Defines the total number of packets that higher-level protocols requested for transmission and which were addressed to a broadcast address at this layer, including those that were discarded or not sent.
Device.Ethernet.Link.{i}.Stats.BroadcastPacketsReceived	unsignedLong	-	Defines the total number of received packets, delivered by this layer to a higher layer, which were addressed to a broadcast address at this layer.
Device.Ethernet.VLANTermination.{i}.	object	-	Defines a VLAN Termination table (a stackable interface object as described in [Section 4.2/TR-181i2]). A VLAN Termination entry is typically stacked on top of a Link object to receive and send frames with the configured VLANID. At most one entry in this table (regardless of whether or not it is enabled) can exist with a given value for Alias, or with a given value for Name. On creation of a new table entry, the CPE MUST choose initial values for Alias and Name such that the new entry does not conflict with any existing entries.
Device.Ethernet.VLANTermination.{i}.Enable	boolean	W	Enables or disables the VLANTermination entry.

Name	Type	Write	Description
			This parameter is based on ifAdminStatus from RFC 2863.
Device.Ethernet.VLANT ermination.{i}.Status	string	-	Defines the current operational state of the VLANTermination entry (see Section 4.2.2 of TR-181i2). Possible values: ▪ Up ▪ Down ▪ Unknown ▪ Dormant ▪ NotPresent ▪ LowerLayerDown ▪ Error (Optional) When Enable is "false" then Status SHOULD normally be Down (or NotPresent or Error if there is a fault condition on the interface). When Enable is changed to "true" then Status should change to Up if and only if the interface is able to transmit and receive network traffic; it should change to Dormant if and only if the interface is operable but is waiting for external actions before it can transmit and receive network traffic (and subsequently change to Up if still operable when the expected actions have completed); it should change to LowerLayerDown if and only if the interface is prevented from entering the Up state because one or more of the interfaces beneath it is down; it should remain in the Error state if there is an error or other fault condition detected on the interface; it should remain in the NotPresent state if the interface has missing (typically hardware) components; it should change to Unknown if the state of the interface can not be determined for some reason. This parameter is based on ifOperStatus from RFC 2863.
Device.Ethernet.VLANT ermination.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in [Section 3.6.1/TR-069a4] and described in [Appendix II/TR-069a4], the following mandatory constraints MUST be enforced: ▪ Its value MUST NOT be empty.

Name	Type	Write	Description
			▪ Its value MUST start with a letter. ▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. ▪ The CPE MUST NOT change the parameter value.
Device.Ethernet.VLANTermination.{i}.Name	string	-	Defines the textual name of the VLANTermination entry as assigned by the CPE.
Device.Ethernet.VLANTermination.{i}.LastChange	unsignedInt	-	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in Section 3.6.1 of TR-069a4] and described in Appendix II of TR-069a4, the following mandatory constraints MUST be enforced: ▪ Its value MUST NOT be empty. ▪ Its value MUST start with a letter. ▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. ▪ The CPE MUST NOT change the parameter value.
Device.Ethernet.VLANTermination.{i}.LowerLayers	string	W	Each list item MUST be the path name of an interface object that is stacked immediately below this interface object. This is a comma-separated list (maximum list length 1024) of strings. If the referenced object is deleted, the corresponding item MUST be removed from the list. See Section 4.2.1 of TR-181i2.
Device.Ethernet.VLANTermination.{i}.VLANID	unsignedInt	W	Defines the VLAN ID for this VLANTermination entry (as defined in [802.1Q-2011]). Only ingress frames with this VLAN ID will be passed to higher protocol layers; frames sent from higher protocol layers will be tagged with this VLAN ID.
Device.Ethernet.VLANTermination.{i}.Stats.	object	-	Defines throughput statistics for this interface. The CPE MUST reset the interface's Stats parameters (unless otherwise stated in individual object or parameter descriptions) either when the interface becomes operationally down due to a previous administrative down (i.e. the interface's Status parameter transitions to a down state after the interface is disabled) or when the interface becomes administratively up (i.e. the interface's

Name	Type	Write	Description
			Enable parameter transitions from false to true). Administrative and operational interface status is discussed in Section 4.2.2 of TR-181i2.
Device.Ethernet.VLANTermination.{i}.Stats.BytesSent	unsignedLong	-	Defines the total number of bytes transmitted out of the interface, including framing characters.
Device.Ethernet.VLANTermination.{i}.Stats.BytesReceived	unsignedLong	-	Defines the total number of bytes received on the interface, including framing characters.
Device.Ethernet.VLANTermination.{i}.Stats.PacketsSent	unsignedLong	-	Defines the total number of packets transmitted out of the interface.
Device.Ethernet.VLANTermination.{i}.Stats.PacketsReceived	unsignedLong	-	Defines the total number of packets received on the interface.
Device.Ethernet.VLANTermination.{i}.Stats.ErrorsSent	unsignedInt	-	Defines the total number of outbound packets that could not be transmitted because of errors.
Device.Ethernet.VLANTermination.{i}.Stats.ErrorsReceived	unsignedInt	-	Defines the total number of inbound packets that contained errors preventing them from being delivered to a higher-layer protocol.
Device.Ethernet.VLANTermination.{i}.Stats.UnicastPacketsSent	unsignedLong	-	Defines the total number of packets requested for transmission which were not addressed to a multicast or broadcast address at this layer, including those that were discarded or not sent.
Device.Ethernet.VLANTermination.{i}.Stats.UnicastPacketsReceived	unsignedLong	-	Defines the total number of received packets, delivered by this layer to a higher layer, which were not addressed to a multicast or broadcast address at this layer.
Device.Ethernet.VLANTermination.{i}.Stats.DiscardPacketsSent	unsignedInt	-	Defines the total number of outbound packets which were chosen to be discarded even though no errors had been detected to prevent their being transmitted. One possible reason for discarding such a packet could be to free up buffer space.
Device.Ethernet.VLANTermination.{i}.Stats.DiscardPacketsReceived	unsignedInt	-	Defines the total number of inbound packets which were chosen to be discarded even though no errors had been detected to prevent their being delivered. One possible reason for discarding such a packet could be to free up buffer space.
Device.Ethernet.VLANTermination.{i}.Stats.MulticastPacketsSent	unsignedLong	-	Defines the total number of packets that higher-level protocols requested for transmission and which were addressed

Name	Type	Write	Description
			to a multicast address at this layer, including those that were discarded or not sent.
Device.Ethernet.VLANTermination.{i}.Stats.MulticastPacketsReceived	unsignedLong	-	Defines the total number of received packets, delivered by this layer to a higher layer, which were addressed to a multicast address at this layer.
Device.Ethernet.VLANTermination.{i}.Stats.BroadcastPacketsSent	unsignedLong	-	Defines the total number of packets that higher-level protocols requested for transmission and which were addressed to a broadcast address at this layer, including those that were discarded or not sent.
Device.Ethernet.VLANTermination.{i}.Stats.BroadcastPacketsReceived	unsignedLong	-	Defines the total number of received packets, delivered by this layer to a higher layer, which were addressed to a broadcast address at this layer.
Device.Bridging.	object	-	Defines the Layer 2 bridging configuration. Specifies bridges between different layer 2 interfaces. Bridges can be defined to include layer 2 filter criteria to selectively bridge traffic between interfaces. This object can be used to configure both 802.1D [802.1D-2004] and 802.1Q [802.1Q-2011] bridges. Not all 802.1D and 802.1Q features are modeled, and some additional features not present in either 802.1D or 802.1Q are modeled. 802.1Q [802.1Q-2011] bridges incorporate 802.1Q [802.1Q-2005] customer and 802.1ad [802.1ad-2005] provider bridges.
Device.Bridging.MaxBridgeEntries	unsignedInt	-	Defines the maximum number of entries available in the Bridging.Bridge table.
Device.Bridging.MaxDBridgeEntries	unsignedInt	-	Defines the maximum number of 802.1D [802.1D-2004] entries available in the Bridging.Bridge table. A positive value for this parameter implies support for 802.1D. There is no guarantee that this many 802.1D Bridges can be configured. For example, the CPE might not be able simultaneously to support both 802.1D and 802.1Q Bridges.
Device.Bridging.MaxQBridgeEntries	unsignedInt	-	Defines the maximum number of 802.1Q [802.1Q-2011] entries available in the Bridging.Bridge table. A non-zero value for this parameter implies support for 802.1Q.

Name	Type	Write	Description
			There is no guarantee that this many 802.1Q Bridges can be configured. For example, the CPE might not be able simultaneously to support both 802.1D and 802.1Q Bridges.
Device.Bridging.MaxVLANEntries	unsignedInt	-	Defines the maximum number of 802.1Q [802.1Q-2011] VLANs supported per Bridging.Bridge table entry.
Device.Bridging.MaxProviderBridgeEntries	unsignedInt	-	Defines the maximum number of entries available in the Bridging.ProviderBridge table. A non-zero value for this parameter implies support for 802.1Q Provider Bridges.
Device.Bridging.ProviderBridgeNumberOfEntries	unsignedInt	-	Defines the number of entries in the ProviderBridge table.
Device.Bridging.MaxFilterEntries	unsignedInt	-	Defines the maximum number of entries available in the Filter table.
Device.Bridging.BridgeNumberOfEntries	unsignedInt	-	Defines the number of entries in the Bridge table.
Device.Bridging.FilterNumberOfEntries	unsignedInt	-	Defines the number of entries in the Filter table.
Device.Bridging.Bridge.{i}.	object	-	Defines the Bridge table.
Device.Bridging.Bridge.{i}.Enable	boolean	W	Enables or disables this Bridge.
Device.Bridging.Bridge.{i}.Status	string	-	Defines the status of this Bridge. Possible values: ▪ Disabled ▪ Enabled ▪ Error (Optional) The Error value may be used by the CPE to indicate a locally defined error condition.
Device.Bridging.Bridge.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in [Section 3.6.1/TR-069a4] and described in [Appendix II/TR-069a4], the following mandatory constraints MUST be enforced: ▪ Its value MUST NOT be empty. ▪ Its value MUST start with a letter. ▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix.

Name	Type	Write	Description
			▪ The CPE MUST NOT change the parameter value.
Device.Bridging.Bridge.{i}.PortNumberOfEntries	unsignedInt	-	Defines the number of entries in the Port table.
Device.Bridging.Bridge.{i}.VLANNumberOfEntries	unsignedInt	-	Defines the number of entries in the VLAN table.
Device.Bridging.Bridge.{i}.VLANPortNumberOfEntries	unsignedInt	-	Defines the number of entries in the VLANPort table.
Device.Bridging.Bridge.{i}.Port.{i}.	object	-	Defines the Bridge Port table, which MUST contain an entry for each bridge port (a stackable interface object as described in Section 4.2 of TR-181i2). There are two types of bridge ports: management (upward facing) and non-management (downward facing). This is determined by configuring the Boolean ManagementPort parameter. The CPE will automatically configure each management bridge port to appear in the interface stack above all non-management bridge ports that share the same Bridge instance. At most one entry in this table (regardless of whether or not it is enabled) can exist with a given value for Alias, or with a given value for Name. On creation of a new table entry, the CPE MUST choose initial values for Alias and Name such that the new entry does not conflict with any existing entries.
Device.Bridging.Bridge.{i}.Port.{i}.Enable	boolean	W	Enables or disables the bridge port. This parameter is based on ifAdminStatus from RFC 2863.
Device.Bridging.Bridge.{i}.Port.{i}.Status	string	-	The current operational state of the bridge port (see Section 4.2.2 of TR-181i2). Possible values: ▪ Up ▪ Down ▪ Unknown ▪ Dormant ▪ NotPresent ▪ LowerLayerDown ▪ Error (Optional) When Enable is "false", then this parameter should normally be Down (or NotPresent or Error if there is a fault condition on the interface).

Name	Type	Write	Description
			When Enable is changed to "true" then this parameter should change to Up if and only if the interface is able to transmit and receive network traffic; it should change to Dormant if and only if the interface is operable but is waiting for external actions before it can transmit and receive network traffic (and subsequently change to Up if still operable when the expected actions have completed); it should change to LowerLayerDown if and only if the interface is prevented from entering the Up state because one or more of the interfaces beneath it is down; it should remain in the Error state if there is an error or other fault condition detected on the interface; it should remain in the NotPresent state if the interface has missing (typically hardware) components; it should change to Unknown if the state of the interface can not be determined for some reason. This parameter is based on ifOperStatus from RFC 2863.
Device.Bridging.Bridge.{i}.Port.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in Section 3.6.1/TR-069a4 and described in Appendix II/TR-069a4, the following mandatory constraints MUST be enforced: ▪ Its value MUST NOT be empty. ▪ Its value MUST start with a letter. ▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. ▪ The CPE MUST NOT change the parameter value.
Device.Bridging.Bridge.{i}.Port.{i}.Name	string	-	Defines the textual name of the bridge port as assigned by the CPE.
Device.Bridging.Bridge.{i}.Port.{i}.LastChange	unsignedInt	-	Defines the accumulated time in seconds since the bridge port entered its current operational state.
Device.Bridging.Bridge.{i}.Port.{i}.LowerLayers	string	W	Each list item MUST be the path name of an interface object that is stacked immediately below this interface object. If the referenced object is deleted, the corresponding item MUST be removed from the list.

Name	Type	Write	Description
			This is a comma-separated list (maximum list length 1024) of strings. See Section 4.2.1/TR-181i2. When ManagementPort is set to "true", the CPE MUST set LowerLayers to reference all non-management bridge ports that are within the same Bridge instance (and update LowerLayers when subsequent non-management bridge ports are added or deleted on that Bridge). The ACS SHOULD NOT set LowerLayers in this case.
Device.Bridging.Bridge.{i}.Port.{i}.ManagementPort	boolean	-	If "true", then the entry is a management (upward facing) bridge port rather than a non-management (downward facing) bridge port. For a given Bridge instance, each management bridge port appears in the interface stack above all non-management bridge ports. The concept of Management Port is discussed in Chapter 8/802.1Q-2005.
Device.Bridging.Bridge.{i}.Port.{i}.PortState	string	-	Defines the Bridge Port state as defined in 802.1D [802.1D-2004] and 802.1Q [802.1Q-2011]. Possible values: ▪ Disabled ▪ Blocking ▪ Listening ▪ Learning ▪ Forwarding ▪ Broken
Device.Bridging.Bridge.{i}.Port.{i}.Stats.	object	-	Throughput statistics for this interface. The CPE MUST reset the interface's Stats parameters (unless otherwise stated in individual object or parameter descriptions) either when the interface becomes operationally down due to a previous administrative down (i.e. the interface's Status parameter transitions to a down state after the interface is disabled) or when the interface becomes administratively up (i.e. the interface's Enable parameter transitions from false to true). Administrative and operational interface status is discussed in Section 4.2.2/TR-181i2.
Device.Bridging.Bridge.{i}.Port.{i}.Stats.BytesSent	unsignedLong	-	Defines the total number of bytes transmitted out of the interface, including framing characters.
Device.Bridging.Bridge.{i}.Port.{i}.Stats.BytesReceived	unsignedLong	-	Defines the total number of bytes received on the interface, including framing characters.

Name	Type	Write	Description
Device.Bridging.Bridge.{i}.Port.{i}.Stats.PacketsSent	unsignedLong	-	Defines the total number of packets transmitted out of the interface.
Device.Bridging.Bridge.{i}.Port.{i}.Stats.PacketsReceived	unsignedLong	-	Defines the total number of packets received on the interface.
Device.Bridging.Bridge.{i}.Port.{i}.Stats.ErrorsSent	unsignedInt	-	Defines the total number of outbound packets that could not be transmitted because of errors.
Device.Bridging.Bridge.{i}.Port.{i}.Stats.ErrorsReceived	unsignedInt	-	Defines the total number of inbound packets that contained errors preventing them from being delivered to a higher-layer protocol.
Device.Bridging.Bridge.{i}.Port.{i}.Stats.UnicastPacketsSent	unsignedLong	-	Defines the total number of packets requested for transmission which were not addressed to a multicast or broadcast address at this layer, including those that were discarded or not sent.
Device.Bridging.Bridge.{i}.Port.{i}.Stats.UnicastPacketsReceived	unsignedLong	-	Defines the total number of received packets, delivered by this layer to a higher layer, which were not addressed to a multicast or broadcast address at this layer.
Device.Bridging.Bridge.{i}.Port.{i}.Stats.DiscardPacketsSent	unsignedInt	-	Defines the total number of outbound packets which were chosen to be discarded even though no errors had been detected to prevent their being transmitted. One possible reason for discarding such a packet could be to free up buffer space.
Device.Bridging.Bridge.{i}.Port.{i}.Stats.DiscardPacketsReceived	unsignedInt	-	Defines the total number of inbound packets which were chosen to be discarded even though no errors had been detected to prevent their being delivered. One possible reason for discarding such a packet could be to free up buffer space.
Device.Bridging.Bridge.{i}.Port.{i}.Stats.MulticastPacketsSent	unsignedLong	-	Defines the total number of packets that higher-level protocols requested for transmission and which were addressed to a multicast address at this layer, including those that were discarded or not sent.
Device.Bridging.Bridge.{i}.Port.{i}.Stats.MulticastPacketsReceived	unsignedLong	-	Defines the total number of received packets, delivered by this layer to a higher layer, which were addressed to a multicast address at this layer.
Device.Bridging.Bridge.{i}.Port.{i}.Stats.BroadcastPacketsSent	unsignedLong	-	Defines the total number of packets that higher-level protocols requested for transmission and which were addressed

Name	Type	Write	Description
			to a broadcast address at this layer, including those that were discarded or not sent.
Device.Bridging.Bridge.{i}.Port.{i}.Stats.BroadcastPacketsReceived	unsignedLong	-	Defines the total number of received packets, delivered by this layer to a higher layer, which were addressed to a broadcast address at this layer.
Device.PPP.	object	-	Defines the Point-to-Point Protocol RFC 1661. This object contains the Interface table.
Device.PPP.InterfaceNumberOfEntries	integer	-	Defines the number of entries in the Interface table.
Device.PPP.SupportedNCPs	integer	-	Defines the Network Control Protocols (NCPs) that are supported by the device. This is a comma-separated list of strings. Possible values: ▪ ATCP (AppleTalk Control Protocol [RFC1378]) IPCP ([RFC1332]) ▪ IPXCP ([RFC1552]) ▪ NBFCP ([RFC2097]) ▪ IPv6CP ([RFC5072]) Note that IPv6CP is an IPv6 capability.
Device.PPP.Interface.{i}.	object	-	Defines the PPP interface table (a stackable interface object as described in Section 4.2/TR-181i2). At most one entry in this table (regardless of whether or not it is enabled) can exist with a given value for Alias, or with a given value for Name. On creation of a new table entry, the CPE MUST choose initial values for Alias and Name such that the new entry does not conflict with any existing entries.
Device.PPP.Interface.{i}.Enable	boolean	W	Enables or disables the interface. This parameter is based on ifAdminStatus from RFC 2863.
Device.PPP.Interface.{i}.Status	string	-	The current operational state of the interface (see Section 4.2.2/TR-181i2). Possible values: ▪ Up ▪ Down ▪ Unknown ▪ Dormant ▪ NotPresent ▪ LowerLayerDown ▪ Error (Optional) When Enable is "false", then Status SHOULD normally be Down (or

Name	Type	Write	Description
			NotPresent or Error if there is a fault condition on the interface).
			When Enable is changed to "true", then Status should change to Up if and only if the interface is able to transmit and receive network traffic; it should change to Dormant if and only if the interface is operable but is waiting for external actions before it can transmit and receive network traffic (and subsequently change to Up if still operable when the expected actions have completed); it should change to LowerLayerDown if and only if the interface is prevented from entering the Up state because one or more of the interfaces beneath it is down; it should remain in the Error state if there is an error or other fault condition detected on the interface; it should remain in the NotPresent state if the interface has missing (typically hardware) components; it should change to Unknown if the state of the interface can not be determined for some reason.
			This parameter is based on ifOperStatus from RFC 2863.
Device.PPP.Interface.{i}.Alias	string	W	Alias provides a mechanism for an ACS to label this instance for future reference.
			This is a non-volatile handle used to reference this instance.
			If the CPE supports the Alias-based Addressing feature as defined in [Section 3.6.1/TR-069a4] and described in [Appendix II/TR-069a4], the following mandatory constraints MUST be enforced:
			▪ Its value MUST NOT be empty.
			▪ Its value MUST start with a letter.
			▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix.
			▪ The CPE MUST NOT change the parameter value.
Device.PPP.Interface.{i}.Name	string	-	Defines the textual name of the interface as assigned by the CPE.
Device.PPP.Interface.{i}.LastChange	unsignedInt	-	Defines the accumulated time in seconds since the interface entered its current operational state.
Device.PPP.Interface.{i}.LowerLayers	string	W	Each list item MUST be the path name of an interface object that is stacked immediately below this interface object. If the referenced object is deleted, the

Name	Type	Write	Description
			corresponding item MUST be removed from the list.
			This is a comma-separated list (maximum list length 1024) of strings.
			See Section 4.2.1/TR-181i2.
Device.PPP.Interface.{i}.Reset	boolean	W	When set to true, the device MUST tear down the existing PPP connection represented by this object and establish a new one.
			The device MUST initiate the reset after completion of the current CWMP session. The device MAY delay resetting the connection in order to avoid interruption of a user service such as an ongoing voice call.
			Reset on a disabled interface is a no-op (not an error).
			When read, this parameter returns false, regardless of the actual value.
			The value of this parameter is not part of the device configuration and is always false when read.
Device.PPP.Interface.{i}.ConnectionStatus	string	-	Defines the current status of the connection.
Device.PPP.Interface.{i}.AutoDisconnectTime	unsignedInt	W	Defines the time in seconds since the establishment of the connection after which connection termination is automatically initiated by the CPE. This occurs irrespective of whether the connection is being used or not. A value of 0 (zero) indicates that the connection is not to be shut down automatically.
Device.PPP.Interface.{i}.IdleDisconnectTime	unsignedInt	W	Defines the time in seconds, that if the connection remains idle, the CPE automatically terminates the connection. A value of 0 (zero) indicates that the connection is not to be shut down automatically.
Device.PPP.Interface.{i}.WarnDisconnectDelay	unsignedInt	W	Time in seconds the ConnectionStatus remains in the PendingDisconnect state before transitioning to disconnecting state to drop the connection.
Device.PPP.Interface.{i}.Username	string	W	Defines the Username to be used for authentication.
Device.PPP.Interface.{i}.Password	string	W	Defines the Password to be used for authentication.
Device.PPP.Interface.{i}.EncryptionProtocol	string	-	Describes the PPP encryption protocol.
Device.PPP.Interface.{i}.CompressionProtocol	string	-	Describes the PPP compression protocol.

Name	Type	Write	Description
Device.PPP.Interface.{i}.AuthenticationProtocol	string	-	Describes the PPP authentication protocol.
Device.PPP.Interface.{i}.MaxMRUSize	unsignedInt	W	Defines the maximum allowed size of frames sent from the remote peer.
Device.PPP.Interface.{i}.CurrentMRUSize	unsignedInt	-	Defines the current MRU in use over this connection.
Device.PPP.Interface.{i}.ConnectionTrigger	string	W	Defines the Trigger used to establish the PPP connection. Possible values: ▪ OnDemand (If this PPP connection is disconnected for any reason, it is to remain disconnected until the CPE has one or more packets to communicate over this connection, at which time the CPE automatically attempts to reestablish the connection) ▪ AlwaysOn (If this PPP connection is disconnected for any reason, the CPE automatically attempts to reestablish the connection (and continues to attempt to reestablish the connection as long it remains disconnected)) ▪ Manual (If this PPP connection is disconnected for any reason, it is to remain disconnected until the user of the CPE explicitly instructs the CPE to reestablish the connection) **Note**: The reason for a PPP connection becoming disconnected to begin with might be either external to the CPE, such as termination by the BRAS or momentary disconnection of the physical interface, or internal to the CPE, such as use of the IdleDisconnectTime and/or AutoDisconnectTime parameters in this object.
Device.PPP.Interface.{i}.LCPEcho	unsignedInt	-	Defines the PPP LCP Echo period in seconds.
Device.PPP.Interface.{i}.LCPEchoRetry	unsignedInt	-	Defines the number of PPP LCP Echo retries within an echo period.
Device.PPP.Interface.{i}.IPCPEnable	boolean	W	Enables or disables IPCP (RFC 1332) on this interface. If this parameter is present, IPCP MUST be included in SupportedNCPs.
Device.PPP.Interface.{i}.IPv6CPEnable	boolean	W	Enables or disables IPv6CP ([RFC 5072]) on this interface. If this parameter is present, IPv6CP MUST be included in SupportedNCPs.

Name	Type	Write	Description
Device.PPP.Interface.{i}.X_00908F_NATEnabled	boolean	W	Indicates if Network Address Translation (NAT) is enabled for this connection.
Device.PPP.Interface.{i}.X_00908F_FirewallEnable	boolean	W	Indicates if the Firewall is enabled for this connection.
Device.PPP.Interface.{i}PPPOE.	object	-	PPPoE object that functionally groups PPPoE related parameters.
Device.PPP.Interface.{i}PPPOE. ACName	string(256)		PPPoE Access Concentrator.
Device.PPP.Interface.{i}PPPOE.ServiceName	string(256)		PPPoE Service name.
Device.PPP.Interface.{i}.IPCP.	object	-	Defines the IP Control Protocol (IPCP) client object for this PPP interface RFC 1332. IPCP only applies to IPv4.
Device.PPP.Interface.{i}.IPCP.LocalIPAddress	boolean	-	Defines the local IPv4 address for this connection received via IPCP.
Device.PPP.Interface.{i}.IPCP.RemoteIPAddress	string	-	Defines the remote IPv4 address for this connection received via IPCP.
Device.PPP.Interface.{i}.IPCP.DNSServers	string	-	Defines IPv4Address. This is a comma-separated list (up to 2 items) of IPv4Addresses. Items represent DNS Server IPv4 address(es) received via IPCP RFC 1877.
Device.PPP.Interface.{i}.Stats.	object	-	Defines throughput statistics for this interface. The CPE MUST reset the interface's Stats parameters (unless otherwise stated in individual object or parameter descriptions) either when the interface becomes operationally down due to a previous administrative down (i.e., the interface's Status parameter transitions to a down state after the interface is disabled) or when the interface becomes administratively up (i.e. the interface's Enable parameter transitions from false to true). Administrative and operational interface status is discussed in Section 4.2.2/TR-181i2.
Device.PPP.Interface.{i}.Stats.BytesSent	unsignedLong	-	Defines the total number of bytes transmitted out of the interface, including framing characters.
Device.PPP.Interface.{i}.Stats.BytesReceived	unsignedLong	-	Defines the total number of bytes received on the interface, including framing characters.

Name	Type	Write	Description
Device.PPP.Interface.{i}.Stats.PacketsSent	unsignedLong	-	Defines the total number of packets transmitted out of the interface.
Device.PPP.Interface.{i}.Stats.PacketsReceived	unsignedLong	-	Defines the total number of packets received on the interface.
Device.PPP.Interface.{i}.Stats.ErrorsSent	unsignedInt	-	Defines the total number of outbound packets that could not be transmitted because of errors.
Device.PPP.Interface.{i}.Stats.ErrorsReceived	unsignedInt	-	Defines the total number of inbound packets that contained errors preventing them from being delivered to a higher-layer protocol.
Device.PPP.Interface.{i}.Stats.UnicastPacketsSent	unsignedLong	-	Defines the total number of packets requested for transmission which were not addressed to a multicast or broadcast address at this layer, including those that were discarded or not sent.
Device.PPP.Interface.{i}.Stats.UnicastPacketsReceived	unsignedLong	-	Defines the total number of received packets, delivered by this layer to a higher layer, which were not addressed to a multicast or broadcast address at this layer.
Device.PPP.Interface.{i}.Stats.DiscardPacketsSent	unsignedInt	-	Defines the total number of outbound packets which were chosen to be discarded even though no errors had been detected to prevent their being transmitted. One possible reason for discarding such a packet could be to free up buffer space.
Device.PPP.Interface.{i}.Stats.DiscardPacketsReceived	unsignedInt	-	Defines the total number of inbound packets which were chosen to be discarded even though no errors had been detected to prevent their being delivered. One possible reason for discarding such a packet could be to free up buffer space.
Device.PPP.Interface.{i}.Stats.MulticastPacketsSent	unsignedLong	-	Defines the total number of packets that higher-level protocols requested for transmission and which were addressed to a multicast address at this layer, including those that were discarded or not sent.
Device.PPP.Interface.{i}.Stats.MulticastPacketsReceived	unsignedLong	-	Defines the total number of received packets, delivered by this layer to a higher layer, which were addressed to a multicast address at this layer.
Device.PPP.Interface.{i}.Stats.BroadcastPacketsSent	unsignedLong	-	Defines the total number of packets that higher-level protocols requested for transmission and which were addressed to a broadcast address at this layer,

Name	Type	Write	Description
			including those that were discarded or not sent.
Device.PPP.Interface.{i}.Stats.BroadcastPacketsReceived	unsignedLong	-	Defines the total number of received packets, delivered by this layer to a higher layer, which were addressed to a broadcast address at this layer.
Device.IP.	object	-	IP object that contains the Interface, ActivePort, and Diagnostics objects.
Device.IP.IPv4Capable	boolean	-	Indicates whether the device is IPv4 capable.
Device.IP.IPv4Enable	boolean	-	Enables or disables the IPv4 stack, and so the use of IPv4 on the device. This affects only layer 3 and above. When "false", IP interfaces that had been operationally up and passing IPv4 packets will now no longer be able to do so, and will be operationally down (unless also attached to an enabled IPv6 stack).
Device.IP.IPv4Status	string	-	Indicates the status of the IPv4 stack. Possible values: ▪ Disabled ▪ Enabled ▪ Error (Optional) The Error value may be used by the CPE to indicate a locally defined error condition.
Device.IP.IPv6Capable	boolean	-	Indicates whether the device is IPv6 capable. **Note:** If "false", it is expected that IPv6-related parameters, enumeration values, etc will not be implemented by the device.
Device.IP.IPv6Enable	boolean	W	Enables or disables the IPv6 stack, and so the use of IPv6 on the device. This affects only Layer 3 and above. When "false", IP interfaces that had been operationally up and passing IPv6 packets will now no longer be able to do so, and will be operationally down (unless also attached to an enabled IPv4 stack).
Device.IP.IPv6Status	string	-	Indicates the status of the IPv6 stack. Possible values: ▪ Disabled ▪ Enabled ▪ Error (Optional) The Error value may be used by the CPE to indicate a locally defined error condition.

Name	Type	Write	Description
Device.IP.InterfaceNumberOfEntries	unsignedInt	-	The number of entries in the Interface table.
Device.IP.Interface.{i}.	object	-	Defines IP interface table (a stackable interface object as described in [Section 4.2/TR-181i2]). This table models the layer 3 IP interface.
			Each IP interface can be attached to the IPv4 and/or IPv6 stack. The interface's IP addresses and prefixes are listed in the IPv4Address, IPv6Address and IPv6Prefix tables.
			Note that support for manipulating Loopback interfaces is OPTIONAL, so the implementation MAY choose not to create (or allow the ACS to create) Interface instances of type Loopback.
			When the ACS administratively disables the interface, i.e. sets Enable to false, the interface's automatically-assigned IP addresses and prefixes may be retained. When the ACS administratively enables the interface, i.e. sets Enable to true, these IP addresses and prefixes MUST be refreshed. It's up to the implementation to decide exactly what this means: it should take all reasonable steps to refresh everything but if it is unable, for example, to refresh a prefix that still has a significant lifetime, it might well choose to retain rather than discard it.
			Any Tunneled IP interface instances instantiated by the CPE MUST NOT have any statistics, writable parameters, IP addresses or IPv6 prefixes. Any - parameters, e.g. Status, MUST return the same information as for the corresponding Tunnel interface. The reason for these rules is that Tunneled IP interfaces exist only in order to be the targets of references (within the data model) and do not model any concepts over and above those already modeled by the Tunnel IP interfaces.
			Note that Tunnel and Tunneled IP interfaces are part of a legacy mechanism that is only used for IPv6rd, DSLite and IPsec tunnels and MUST NOT be used in any other context. For all other tunneling mechanisms Normal IP interfaces are stacked above technology-specific Tunnel Interfaces, e.g. above GRE.Tunnel.{i}.Interface or MAP.Domain.{i}.Interface objects.

Name	Type	Write	Description
			At most one entry in this table (regardless of whether or not it is enabled) can exist with a given value for Alias, or with a given value for Name. On creation of a new table entry, the CPE MUST choose initial values for Alias and Name such that the new entry does not conflict with any existing entries.
Device.IP.Interface.{i}.Enable	boolean	W	Enables or disables the interface (regardless of IPv4Enable and IPv6Enable). This parameter is based on ifAdminStatus from RFC 2863.
Device.IP.Interface.{i}.IPv4Enable	boolean	-	If set to "true", attaches this interface to the IPv4 stack. If set to "false", detaches this interface from the IPv4 stack. Once detached from the IPv4 stack, the interface will now no longer be able to pass IPv4 packets, and will be operationally down (unless also attached to an enabled IPv6 stack). For an IPv4 capable device, if IPv4Enable is not present this interface SHOULD be permanently attached to the IPv4 stack. **Note:** IPv4Enable is independent of Enable, and that to administratively enable an interface for IPv4 it is necessary for both Enable and IPv4Enable to be true.
Device.IP.Interface.{i}.IPv6Enable	boolean	W	If set to "true", attaches this interface to the IPv6 stack. If set to false, detaches this interface from the IPv6 stack. Once detached from the IPv6 stack, the interface will now no longer be able to pass IPv6 packets, and will be operationally down (unless also attached to an enabled IPv4 stack). For an IPv6 capable device, if IPv6Enable is not present this interface should be permanently attached to the IPv6 stack. **Note:** IPv6Enable is independent of Enable, and that to administratively enable an interface for IPv6 it is necessary for both Enable and IPv6Enable to be true.
Device.IP.Interface.{i}.Status	string	-	Defines the current operational state of the interface (see Section 4.2.2/TR-181i2). Possible values: Up:

Name	Type	Write	Description
			Down
			Unknown
			Dormant
			NotPresent
			LowerLayerDown
			Error (Optional)
			When Enable is "false", then Status should normally be Down (or NotPresent or Error if there is a fault condition on the interface).
			When Enable is changed to "true", then Status should change to Up if and only if the interface is able to transmit and receive network traffic; it should change to Dormant if and only if the interface is operable but is waiting for external actions before it can transmit and receive network traffic (and subsequently change to Up if still operable when the expected actions have completed); it should change to LowerLayerDown if and only if the interface is prevented from entering the Up state because one or more of the interfaces beneath it is down; it should remain in the Error state if there is an error or other fault condition detected on the interface; it should remain in the NotPresent state if the interface has missing (typically hardware) components; it should change to Unknown if the state of the interface can not be determined for some reason.
			This parameter is based on ifOperStatus from RFC-2863.
Device.IP.Interface.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference.
			If the CPE supports the Alias-based Addressing feature as defined in [Section 3.6.1/TR-069a4] and described in [Appendix II/TR-069a4], the following mandatory constraints MUST be enforced:
			▪ Its value MUST NOT be empty.
			▪ Its value MUST start with a letter.
			▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix.
			▪ The CPE MUST NOT change the parameter value.
Device.IP.Interface.{i}.Name	string	-	Defines the textual name of the interface as assigned by the CPE.

Name	Type	Write	Description
Device.IP.Interface.{i}.LastChange	unsignedInt	-	The accumulated time in seconds since the interface entered its current operational state.
Device.IP.Interface.{i}.LowerLayers	string	W	Each list item MUST be the path name of an interface object that is stacked immediately below this interface object. If the referenced object is deleted, the corresponding item MUST be removed from the list. See Section 4.2.1/TR-181i2. This is a comma-separated list (maximum list length 1024) of strings. LowerLayers MUST be an empty string and - when Type is Loopback, Tunnel, or Tunneled.
Device.IP.Interface.{i}.Router	string	W	The value MUST be the path name of a row in the Routing.Router table. If the referenced object is deleted, the parameter value MUST be set to an empty string. The Router instance that is associated with this IP Interface entry.
Device.IP.Interface.{i}.Reset	boolean	W	When set to "true", the device MUST tear down the existing IP connection represented by this object and establish a new one. The device MUST initiate the reset after completion of the current CWMP session. The device MAY delay resetting the connection in order to avoid interruption of a user service such as an ongoing voice call. Reset on a disabled interface is a no-op (not an error). When read, this parameter returns false, regardless of the actual value. The value of this parameter is not part of the device configuration and is always false when read.
Device.IP.Interface.{i}.MaxMTUSize	unsignedInt	W	Defines the maximum transmission unit (MTU); the largest allowed size of an IP packet (including IP headers, but excluding lower layer headers such as Ethernet, PPP, or PPPoE headers) that is allowed to be transmitted by or through this device.
Device.IP.Interface.{i}.Type	string	-	Defines the IP interface type. Possible values: ▪ Normal ▪ Loopback ▪ Tunnel (Only used with legacy (Tunnel,Tunneled) IP interface pairs)

Name	Type	Write	Description
			• Tunneled (Only used with legacy (Tunnel,Tunneled) IP interface pairs) For Loopback, Tunnel, and Tunneled IP interface objects, the LowerLayers parameter MUST be an empty string.
Device.IP.Interface.{i}.Loopback	boolean	W	When set to "true", the IP interface becomes a loopback interface and the CPE MUST set Type to Loopback. In this case, the CPE MUST also set LowerLayers to an empty string and fail subsequent attempts at setting LowerLayers until the interface is no longer a loopback. Support for manipulating loopback interfaces is Optional.
Device.IP.Interface.{i}.IPv4AddressNumberOfEntries	unsignedInt	-	Defines the number of entries in the IPv4Address table.
Device.IP.Interface.{i}.IPv6AddressNumberOfEntries	unsignedInt	-	Defines the number of entries in the IPv6Address table.
Device.IP.Interface.{i}.IPv4Address.{i}.	object	-	Defines the IPv4 address table. Entries are auto-created and auto-deleted as IP addresses are added and deleted via DHCP, auto-IP, or IPCP. Static entries are created and configured by the ACS.
Device.IP.Interface.{i}.IPv4Address.{i}.Enable	boolean	W	Enables or disables this IPv4 address.
Device.IP.Interface.{i}.IPv4Address.{i}.Status	string	-	The status of this IPv4Address table entry. Possible values: • Disabled • Enabled • Error_Misconfigured • Error (OPTIONAL) The Error_Misconfigured value indicates that a necessary configuration value is undefined or invalid. The Error value may be used by the CPE to indicate a locally defined error condition.
Device.IP.Interface.{i}.IPv4Address.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in [Section 3.6.1/TR-069a4] and described in [Appendix II/TR-069a4], the following mandatory constraints MUST be enforced:

Name	Type	Write	Description
			▪ Its value MUST NOT be empty. ▪ Its value MUST start with a letter. ▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. ▪ The CPE MUST NOT change the parameter value. ▪ This parameter can only be modified if AddressingType is Static.
Device.IP.Interface.{i}.IPv4Address.{i}.IPAddress	string	W	Defines the IPv4Address - IPv4 address. This parameter can only be modified if the AddressingType is Static.
Device.IP.Interface.{i}.IPv4Address.{i}.SubnetMask	boolean	W	Defines the IPv4Address Subnet mask. This parameter can only be modified if the AddressingType is Static.
Device.IP.Interface.{i}.IPv4Address.{i}.AddressingType	string	W	Defines the addressing method used to assign the IP address. Possible values: ▪ DHCP ▪ IKEv2 (Assigned by IKEv2 [RFC5996]) ▪ AutoIP ▪ IPCP ▪ Static
Device.IP.Interface.{i}. IPv6Address.{i}.	object	W	This table contains the IP interface's IPv6 unicast addresses. There MUST be an entry for each such address, including anycast addresses. There are several ways in which entries can be added to and deleted from this table, including: ▪ Automatically via SLAAC [RFC4862], which covers generation of link-local addresses (for all types of device) and global addresses (for non-router devices). ▪ Automatically via DHCPv6 [RFC3315], which covers generation of any type of address (subject to the configured DHCP server policy). ▪ Manually via a GUI or some other local management interface. ▪ Manually via factory default configuration. ▪ By the ACS. This table MUST NOT include entries for the Subnet-Router anycast address [Section 2.6.1/RFC4291]. Such entries would be identical to others but with a zero interface identifier, and would add no value.

Name	Type	Write	Description
			A loopback interface will always have address ::1 [Section 2.5.3/RFC4291] and MAY also have link-local address fe80::1. This object is based on ipAddressTable from [RFC4293]. At most one entry in this table (regardless of whether or not it is enabled) can exist with a given value for Alias. On creation of a new table entry, the Agent MUST choose an initial value for Alias such that the new entry does not conflict with any existing entries. At most one enabled entry in this table can exist with a given value for IPAddress.
Device.IP.Interface.{i}.IPv6Address.{i}.IPAddress	String (45)	W	[IPv6Address] IPv6 address. This parameter can only be modified if the Origin is Static. This parameter is based on ipAddressAddr from [RFC4293].
Device.IP.Interface.{i}.Stats.	object	-	Defines the hroughput statistics for this interface. The CPE MUST reset the interface's Stats parameters (unless otherwise stated in individual object or parameter descriptions) either when the interface becomes operationally down due to a previous administrative down (i.e. the interface's Status parameter transitions to a down state after the interface is disabled) or when the interface becomes administratively up (i.e. the interface's Enable parameter transitions from false to true). Administrative and operational interface status is discussed in Section 4.2.2 / TR-181i2.
Device.IP.Interface.{i}.Stats.BytesSent	unsignedLong	-	Defines the total number of bytes transmitted out of the interface, including framing characters.
Device.IP.Interface.{i}.Stats.BytesReceived	unsignedLong	-	Defines the total number of bytes received on the interface, including framing characters.
Device.IP.Interface.{i}.Stats.PacketsSent	unsignedLong	-	Defines the total number of packets transmitted out of the interface.
Device.IP.Interface.{i}.Stats.PacketsReceived	unsignedLong	-	Defines the total number of packets received on the interface.
Device.IP.Interface.{i}.Stats.ErrorsSent	unsignedInt	-	Defines the total number of outbound packets that could not be transmitted because of errors.

Name	Type	Write	Description
Device.IP.Interface.{i}.Stats.ErrorsReceived	unsignedInt	-	Defines the total number of inbound packets that contained errors preventing them from being delivered to a higher-layer protocol.
Device.IP.Interface.{i}.Stats.UnicastPacketsSent	unsignedLong	-	Defines the total number of packets requested for transmission which were not addressed to a multicast or broadcast address at this layer, including those that were discarded or not sent.
Device.IP.Interface.{i}.Stats.UnicastPacketsReceived	unsignedLong	-	Defines the total number of received packets, delivered by this layer to a higher layer, which were not addressed to a multicast or broadcast address at this layer.
Device.IP.Interface.{i}.Stats.DiscardPacketsSent	unsignedInt	-	Defines the total number of outbound packets which were chosen to be discarded even though no errors had been detected to prevent their being transmitted. One possible reason for discarding such a packet could be to free up buffer space.
Device.IP.Interface.{i}.Stats.DiscardPacketsReceived	unsignedInt	-	Defines the total number of inbound packets which were chosen to be discarded even though no errors had been detected to prevent their being delivered. One possible reason for discarding such a packet could be to free up buffer space.
Device.IP.Interface.{i}.Stats.MulticastPacketsSent	unsignedLong	-	Defines the total number of packets that higher-level protocols requested for transmission and which were addressed to a multicast address at this layer, including those that were discarded or not sent.
Device.IP.Interface.{i}.Stats.MulticastPacketsReceived	unsignedLong	-	Defines the total number of received packets, delivered by this layer to a higher layer, which were addressed to a multicast address at this layer.
Device.IP.Interface.{i}.Stats.BroadcastPacketsSent	unsignedLong	-	Defines the total number of packets that higher-level protocols requested for transmission and which were addressed to a broadcast address at this layer, including those that were discarded or not sent. **Note:** IPv6 does not define broadcast addresses, so IPv6 packets will never cause this counter to increment.
Device.IP.Interface.{i}.Stats.BroadcastPacketsReceived	unsignedLong	-	Defines the total number of received packets, delivered by this layer to a higher layer, which were addressed to a broadcast address at this layer.

Name	Type	Write	Description
			Note: IPv6 does not define broadcast addresses, so IPv6 packets will never cause this counter to increment.
Device.IP.Diagnostics	object	-	The IP Diagnostics objects.
Device.IP.Diagnostics.IPv4PingSupported	boolean	-	Indicates that ping over IPv4 is supported.
Device.IP.Diagnostics.IPv6PingSupported	boolean	-	Indicates that ping over IPv6 is supported.
Device.IP.Diagnostics.IPv4TraceRouteSupported	boolean	-	Indicates that TraceRoute over IPv4 is supported.
Device.IP.Diagnostics.IPv6TraceRouteSupported	boolean	-	Indicates that TraceRoute over IPv6 is supported.
Device.IP.Diagnostics.IPv4DownloadDiagnosticsSupported	boolean	-	Indicates that DownloadDiagnostics over IPv4 is supported.
Device.IP.Diagnostics.IPv6DownloadDiagnosticsSupported	boolean	-	Indicates that DownloadDiagnostics over IPv6 is supported.
Device.IP.Diagnostics.IPv4UploadDiagnosticsSupported	boolean	-	Indicates that UploadDiagnostics over IPv4 is supported.
Device.IP.Diagnostics.IPv6UploadDiagnosticsSupported	boolean	-	Indicates that UploadDiagnostics over IPv6 is supported.
Device.IP.Diagnostics.IPv4UDPEchoDiagnosticsSupported	boolean	-	Indicates that UDP Echo Diagnostics over IPv4 is supported.
Device.IP.Diagnostics.IPv6UDPEchoDiagnosticsSupported	boolean	-	Indicates that UD Echo Diagnostics over IPv6 is supported.
Device.IP.Diagnostics.IPv4ServerSelectionDiagnosticsSupported	boolean	-	Indicates that ServerSelection Diagnostics over IPv4 is supported.
Device.IP.Diagnostics.IPv6ServerSelectionDiagnosticsSupported	boolean	-	Indicates that ServerSelection Diagnostics over IPv6 is supported.
Device.IP.Diagnostics.DownloadDiagnostics.	object	-	This object defines the diagnostics configuration for an HTTP and FTP DownloadDiagnostics Test. Files received in the DownloadDiagnostics do not require file storage on the CPE device.
Device.IP.Diagnostics.DownloadDiagnostics.DiagnosticsState	string	W	Indicate the availability of diagnostic data. Enumeration of: ▪ None (READONLY)

Name	Type	Write	Description
			▪ Requested ▪ Completed (READONLY) ▪ Error_CannotResolveHostName (READONLY) ▪ Error_NoRouteToHost (READONLY) ▪ Error_InitConnectionFailed (READONLY) ▪ Error_NoResponse (READONLY) ▪ Error_TransferFailed (READONLY) ▪ Error_PasswordRequestFailed (READONLY) ▪ Error_LoginFailed (READONLY) ▪ Error_NoTransferMode (READONLY) ▪ Error_NoPASV (READONLY) ▪ Error_IncorrectSize (READONLY) ▪ Error_Timeout (READONLY) ▪ Error_Internal (READONLY) ▪ Error_Other (READONLY) If the ACS sets the value of this parameter to Requested, the CPE MUST initiate the corresponding diagnostic test. When writing, the only allowed value is Requested. To ensure the use of the proper test parameters (the writable parameters in this object), the test parameters MUST be set either prior to or at the same time as (in the same SetParameterValues) setting the DiagnosticsState to Requested. When requested, the CPE SHOULD wait until after completion of the communication session with the ACS before starting the diagnostic. When the test is completed, the value of this parameter MUST be either Completed (if the test completed successfully), or one of the Error values listed above. If the value of this parameter is anything other than Completed, the values of the results parameters for this test are indeterminate. When the diagnostic initiated by the ACS is completed (successfully or not), the CPE MUST establish a new connection to the ACS to allow the ACS to view the results, indicating the Event code 8 DIAGNOSTICS COMPLETE in the Inform message. After the diagnostic is complete, the value of all result parameters (all read-only parameters in this object) MUST be retained by the CPE until either this

Name	Type	Write	Description
			diagnostic is run again, or the CPE reboots. After a reboot, if the CPE has not retained the result parameters from the most recent test, it MUST set the value of this parameter to None and remove all object instances from PerConnectionResult and IncrementalResult.
			Modifying any of the writable parameters in this object except for this one MUST result in the value of this parameter being set to None and remove all object instances fromPerConnectionResult and IncrementalResult.
			While the test is in progress, modifying any of the writable parameters in this object except for this one MUST result in the test being terminated and the value of this parameter being set to None and remove all object instances from PerConnectionResult and IncrementalResult.
			While the test is in progress, setting this parameter to Requested (and possibly modifying other writable parameters in this object) MUST result in the test being terminated and then restarted using the current values of the test parameters.
Device.IP.Diagnostics. DownloadDiagnostics.Diagnostics.Interface	string(256)	-	The value MUST be the Path Name of a table row. The IP-layer interface over which the test is to be performed. Example: Device.IP.Interface.1 If an empty string is specified, the CPE MUST use the interface as directed by its routing policy (Forwarding table entries) to determine the appropriate interface.
Device.IP.Diagnostics. DownloadDiagnostics.Diagnostics.DownloadURL	string(256)		The URL, as defined in [RFC3986], for the CPE to perform the download on. This parameter MUST be in the form of a valid HTTP [RFC2616] or FTP [RFC959] URL. When using FTP transport, FTP binary transfer MUST be used. When using HTTP transport, persistent connections MUST be used and pipelining MUST NOT be used. When using HTTP transport the HTTP Authentication MUST NOT be used.
Device.IP.Diagnostics. DownloadDiagnostics.Diagnostics.Transport	string	-	Comma-separated list of strings. Supported DownloadDiagnostics transport protocols for a CPE device. Each list item is an enumeration of:

Name	Type	Write	Description
			• HTTP • FTP (optional)
Device.IP.Diagnostics.DownloadDiagnostics.Diagnostics.DSCP	unsignedInt	-	The DiffServ code point for marking packets transmitted in the test. The default value SHOULD be zero.
Device.IP.Diagnostics.DownloadDiagnostics.Diagnostics.EthernetPriority	unsignedint		Ethernet priority code for marking packets transmitted in the test (if applicable). The default value SHOULD be zero.
Device.IP.Diagnostics.DownloadDiagnostics.Diagnostics.ROMTime	dateTime	-	Request time in UTC, which MUST be specified to microsecond precision. For example: 2008-04-09T15:01:05.123456Z • For HTTP, this is the time at which the client sends the GET command. • For FTP, this is the time at which the client sends the RTRV command. If multiple connections are used, then ROMTime is set to the earliest ROMTime across all connections.
Device.IP.Diagnostics.DownloadDiagnostics.Diagnostics.BOMTime	dateTime	-	Begin of transmission t time in UTC, which MUST be specified to microsecond precision. For example: 2008-04-09T15:01:05.123456Z • For HTTP, this is the time at which the client sends the GET command. • For FTP, this is the time at which the client sends the RTRV command. If multiple connections are used, then BOMTime is set to the earliest BOMTime across all connections.
Device.IP.Diagnostics.DownloadDiagnostics.Diagnostics.EOMTime	dateTime	-	Begin of transmission time in UTC, which MUST be specified to microsecond precision For example: 2008-04-09T15:01:05.123456Z • For HTTP, this is the time at which the first data packet is received. • For FTP, this is the time at which the client receives the first data packet on the data connection. If multiple connections are used, then BOMTime is set to the earliest BOMTime across all connections.
Device.IP.Diagnostics.DownloadDiagnostics.Diagnostics.TCPOpenRequestTime	dateTime	-	Request time in UTC, which MUST be specified to microsecond precision. For example: 2008-04-09T15:01:05.123456Z

Name	Type	Write	Description
			▪ For HTTP, this is the time at which the TCP socket open (SYN) was sent for the HTTP connection. ▪ For FTP, this is the time at which the TCP socket open (SYN) was sent for the data connection. **Note:** Interval of 1 microsecond SHOULD be supported. If multiple connections are used, then TCPOpenRequestTime is set to the latest TCPOpenRequestTime across all connections.
Device.IP.Diagnostics. DownloadDiagnostics.D iagnostics.TCPOpenRe sponseTime	dateTime	-	Response time in UTC, which MUST be specified to microsecond precision. For example: 2008-04-09T15:01:05.123456Z ▪ For HTTP, this is the time at which the TCP ACK to the socket opening the HTTP connection was received. ▪ For FTP, this is the time at which the TCP ACK to the socket opening the data connection was received. **Note:** Interval of 1 microsecond SHOULD be supported. If multiple connections are used, then TCPOpenResponseTime is set to the latest TCPOpenResponseTime across all connections.
Device.IP.Diagnostics. DownloadDiagnostics.D iagnostics.TimeBasedT estDuration	unsignedInt		Controls time based testing [Section 4.3/TR-143a1]. When TimeBasedTestDuration > 0, TimeBasedTestDuration is the duration in seconds of a time based test. If TimeBasedTestDuration is 0, the test is not based on time, but on the size of the file to be downloaded. The default value SHOULD be 0.
Device.IP.Diagnostics. DownloadDiagnostics.D iagnostics.TimeBasedT estMeasurementInterva l	unsignedInt	-	The measurement interval duration in seconds for objects in IncrementalResult for a time based FTP/HTTP download test (when TimeBasedTestDuration > 0). The default value SHOULD be 0, which implies IncrementalResult collection is disabled. For example if $TimeBasedTestDuration$ is 90 seconds and TimeBasedTestMeasurementInterval is 10 seconds, there will be 9 results in $IncrementalResult$, each with a 10 seconds duration.
Device.IP.Diagnostics. DownloadDiagnostics.D	unsignedInt	-	This TimeBasedTestMeasurementOffset works in conjunction with

Name	Type	Write	Description
iagnostics.TimeBasedTestMeasurementOffset			TimeBasedTestMeasurementInterval to allow the interval measurement to start a number of seconds after BOMTime. The test measurement interval in IncrementalResult starts at time BOMTime + TimeBasedTestMeasurementOffset to allow for slow start window removal of file transfers. This TimeBasedTestMeasurementOffset is in seconds. The default value SHOULD be 0
Device.IP.Diagnostics.DownloadDiagnostics.Diagnostics.ProtocolVersion	string	-	Indicates the IP protocol version to be used. The default value SHOULD be Any. Enumeration of: • Any (Use either IPv4 or IPv6 depending on the system preference) • IPv4 (Use IPv4 for the requests) • IPv6 (Use IPv6 for the requests)
Device.IP.Diagnostics.DownloadDiagnostics.Diagnostics.TestBytesReceived	unsignedInt	-	The number of bytes received during the FTP/HTTP transaction including FTP/HTTP headers, between BOMTime and EOMTime across all connections.
Device.IP.Diagnostics.DownloadDiagnostics.Diagnostics.X_00908F_Speed	string	-	Textual result of download speed test. Example: 8.1 Mbps.
Device.IP.Diagnostics.DownloadDiagnostics.Diagnostics.X_00908F_TestDuration	unsignedInt	-	Time took to run the download Speed Test.
Device.IP.Diagnostics.UploadDiagnostics.	object	-	This object defines the diagnostics configuration for an HTTP or FTP UploadDiagnostics test. Files sent by the UploadDiagnostics do not require file storage on the CPE device, and MAY be an arbitrary stream of bytes.
Device.IP.Diagnostics.UploadDiagnostics.DiagnosticsState	string	W	Indicates the availability of diagnostic data. Enumeration of: • None (READONLY) • Requested • Completed (READONLY) • Error_CannotResolveHostName (READONLY) • Error_NoRouteToHost (READONLY) • Error_InitConnectionFailed (READONLY) • Error_NoResponse (READONLY) • Error_PasswordRequestFailed (READONLY)

Name	Type	Write	Description
			▪ Error_LoginFailed (READONLY)
			▪ Error_NoTransferMode (READONLY)
			▪ Error_NoPASV (READONLY)
			▪ Error_NoCWD (READONLY)
			▪ Error_NoSTOR (READONLY)
			▪ Error_NoTransferComplete (READONLY)
			▪ Error_Timeout (READONLY)
			▪ Error_Internal (READONLY)
			▪ Error_Other (READONLY)
			If the ACS sets the value of this parameter to Requested, the CPE MUST initiate the corresponding diagnostic test. When writing, the only allowed value is Requested. To ensure the use of the proper test parameters (the writable parameters in this object), the test parameters MUST be set either prior to or at the same time as (in the same SetParameterValues) setting the DiagnosticsState to Requested.
			When requested, the CPE SHOULD wait until after completion of the communication session with the ACS before starting the diagnostic.
			When the test is completed, the value of this parameter MUST be either Completed (if the test completed successfully), or one of the Error values listed above.
			If the value of this parameter is anything other than Completed, the values of the results parameters for this test are indeterminate.
			When the diagnostic initiated by the ACS is completed (successfully or not), the CPE MUST establish a new connection to the ACS to allow the ACS to view the results, indicating the Event code 8 DIAGNOSTICS COMPLETE in the Inform message.
			After the diagnostic is complete, the value of all result parameters (all read-only parameters in this object) MUST be retained by the CPE until either this diagnostic is run again, or the CPE reboots. After a reboot, if the CPE has not retained the result parameters from the most recent test, it MUST set the value of this parameter to None and remove all object instances from UploadDiagnostics and IncrementalResult.

Name	Type	Write	Description
			Modifying any of the writable parameters in this object except for this one MUST result in the value of this parameter being set to None and remove all object instances from PerConnectionResult and IncrementalResult. While the test is in progress, modifying any of the writable parameters in this object except for this one MUST result in the test being terminated and the value of this parameter being set to None and remove all object instances from PerConnectionResult and IncrementalResult. While the test is in progress, setting this parameter to Requested (and possibly modifying other writable parameters in this object) MUST result in the test being terminated and then restarted using the current values of the test parameters.
Device.IP.Diagnostics. UploadDiagnostics.Interface	string(256)	-	The value MUST be the Path Name of a table row. The IP-layer interface over which the test is to be performed. Example: Device.IP.Interface.1 If an empty string is specified, the CPE MUST use the interface as directed by its routing policy (Forwarding table entries) to determine the appropriate interface.
Device.IP.Diagnostics. UploadDiagnostics.UploadURL	string(256)	W	The URL, as defined in [RFC3986], for the CPE to Upload to. This parameter MUST be in the form of a valid HTTP [RFC2616] or FTP [RFC959] URL. ▪ When using FTP transport, FTP binary transfer MUST be used. ▪ When using HTTP transport, persistent connections MUST be used and pipelining MUST NOT be used. ▪ When using HTTP transport the HTTP Authentication MUST NOT be used.
Device.IP.Diagnostics. UploadDiagnostics.UploadTransports	string	-	Comma-separated list of strings. Supported UploadDiagnostics transport protocols for a CPE device. Each list item is an enumeration of: ▪ HTTP ▪ FTP (optional)
Device.IP.Diagnostics. UploadDiagnostics.DSCP	unsignedInt	-	DiffServ code point for marking packets transmitted in the test. The default value SHOULD be zero.
Device.IP.Diagnostics. UploadDiagnostics.EthernetPriority	unsignedInt	-	Ethernet priority code for marking packets transmitted in the test (if applicable).

Name	Type	Write	Description
			The default value SHOULD be zero.
Device.IP.Diagnostics. UploadDiagnostics.Test FileLength	unsignedInt	w	The size of the file (in bytes) to be uploaded to the server. The CPE MUST ensure the appropriate number of bytes are sent.
Device.IP.Diagnostics. UploadDiagnostics.Tim eBasedTestDuration	unsignedInt	-	Controls time-based testing [Section 4.3/TR-143a1]. When TimeBasedTestDuration > 0, TimeBasedTestDuration is the duration in seconds of a time based test. If TimeBasedTestDuration is 0, the test is not based on time, but on the size of the file to be uploaded. The default value SHOULD be 0.
Device.IP.Diagnostics. UploadDiagnostics.Tim eBasedTestMeasureme ntInterval	unsignedInt	-	The measurement interval duration in seconds for objects in IncrementalResult for a time based FTP/HTTP upload test (when TimeBasedTestDuration > 0). The default value SHOULD be 0, which implies IncrementalResult collection is disabled. For example if TimeBasedTestDuration is 90 seconds and TimeBasedTestMeasurementInterval is 10 seconds, there will be 9 results in IncrementalResult, each with a 10 seconds duration.
Device.IP.Diagnostics. UploadDiagnostics.Tim eBasedTestMeasureme ntOffset	unsignedInt	-	This TimeBasedTestMeasurementOffset works in conjunction with TimeBasedTestMeasurementInterval and allows the interval measurement to start a number of seconds after BOMTime. The test measurement interval in IncrementalResult starts at time BOMTime + TimeBasedTestMeasurementOffset to allow for slow start window removal of file transfers. This TimeBasedTestMeasurementOffset is in seconds. The default value SHOULD be 0.
Device.IP.Diagnostics. UploadDiagnostics.Prot ocolVersion	string	-	Indicates the IP protocol version to be used. The default value SHOULD be Any. Enumeration of: ▪ Any (Use either IPv4 or IPv6 depending on the system preference) ▪ IPv4 (Use IPv4 for the requests) ▪ IPv6 (Use IPv6 for the requests)
Device.IP.Diagnostics. UploadDiagnostics.RO MTime	dateTime	-	Request time in UTC, which MUST be specified to microsecond precision.

Name	Type	Write	Description
			For example: 2008-04-09T15:01:05.123456Z ▪ For HTTP, this is the time at which the client sends the GET command. ▪ For FTP, this is the time at which the client sends the RTRV command. If multiple connections are used, then ROMTime is set to the earliest ROMTime across all connections.
Device.IP.Diagnostics.UploadDiagnostics.BOMTime	dateTime	-	Start of transmission time in UTC, which MUST be specified to microsecond precision. For example: 2008-04-09T15:01:05.123456Z For HTTP, this is the time at which the first data packet is received. For FTP, this is the time at which the client receives the first data packet on the data connection. If multiple connections are used, then BOMTime is set to the earliest BOMTime across all connections.
Device.IP.Diagnostics.UploadDiagnostics.EOMTime	dateTime	-	End of transmission in UTC, which MUST be specified to microsecond precision. For example: 2008-04-09T15:01:05.123456Z ▪ For HTTP, this is the time at which the last data packet is received. ▪ For FTP, this is the time at which the client receives the last packet on the data connection. If multiple connections are used, then EOMTime is set to the latest EOMTime across all connections.
Device.IP.Diagnostics.UploadDiagnostics.TCPOpenRequestTime	dateTime	-	Request time in UTC, which MUST be specified to microsecond precision. For example: 2008-04-09T15:01:05.123456Z ▪ For HTTP, this is the time at which the TCP socket open (SYN) was sent for the HTTP connection. ▪ For FTP, this is the time at which the TCP socket open (SYN) was sent for the data connection. **Note:** Interval of 1 microsecond SHOULD be supported. If multiple connections are used, then TCPOpenRequestTime is set to the latest TCPOpenRequestTime across all connections.

Name	Type	Write	Description
Device.IP.Diagnostics. UploadDiagnostics.TCP OpenResponseTime	dateTime	-	Response time in UTC, which MUST be specified to microsecond precision. For example: 2008-04-09T15:01:05.123456Z ▪ For HTTP, this is the time at which the TCP ACK to the socket opening the HTTP connection was received. ▪ For FTP, this is the time at which the TCP ACK to the socket opening the data connection was received. **Note:** Interval of 1 microsecond SHOULD be supported. If multiple connections are used, then TCPOpenResponseTime is set to the latest TCPOpenResponseTime across all connections.
Device.IP.Diagnostics. UploadDiagnostics.X_0 0908F_Speed	String	-	Textual result of download speed test. Example: 8.1 Mbps.
Device.IP.Diagnostics. UploadDiagnostics X_00908F_TestDura-tion	unsignedInt	-	Time took to run the download Speed Test.
Device.Routing.	object	-	Defines the routing object that contains the Router table and RIP protocol object.
Device.Routing.Router NumberOfEntries	unsignedInt	-	Defines the number of entries in the Router table.
Device.Routing.Route r.{i}.	object	-	This object allows the handling of the routing and forwarding configuration of the device.
Device.Routing.Router.{ i}.Enable	boolean	-	Enables or disables this "Router" entry.
Device.Routing.Router.{ i}.Status	string	-	Defines the status of this Router entry. Possible values: ▪ Disabled ▪ Enabled ▪ Error (Optional) The Error value MAY be used by the CPE to indicate a locally defined error condition
Device.Routing.Router.{ i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in Section 3.6.1/TR-069a4 and described in Appendix II/TR-069a4, the following

Name	Type	Write	Description
			mandatory constraints MUST be enforced: • Its value MUST NOT be empty. • Its value MUST start with a letter. • If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. • The CPE MUST NOT change the parameter value.
Device.Routing.Router.{i}.IPv4ForwardingNumberOfEntries	unsignedInt	-	Defines the number of entries in the IPv4Forwarding table.
Device.Routing.Router.{i}.IPv6ForwardingNumberOfEntries	unsignedInt	-	Defines the number of entries in the IPv6Forwarding table.
Device.Routing.Router.{i}.IPv4Forwarding.{i}.	object	-	Defines the Layer 3 IPv4 forwarding table. In addition to statically configured routes, this table MUST include dynamic routes learned through layer 3 routing protocols, including RIP (i.e. RIP version 2), OSPF, DHCPv4, and IPCP. The CPE MAY reject attempts to delete or modify a dynamic route entry. For each incoming packet, the layer 3 forwarding decision is conceptually made as follows: Only enabled table entries with a matching ForwardingPolicy are considered, i.e. those that either do not specify a ForwardingPolicy, or else specify a ForwardingPolicy that matches that of the incoming packet. Next, table entries that also have a matching destination address/mask are considered, and the matching entry with the longest prefix is applied to the packet (i.e. the entry with the most specific network). An unspecified destination address is a wild-card and always matches, but with a prefix length of zero. For enabled table entries, if Interface is not a valid reference to an IPv4-capable interface (that is attached to the IPv4 stack), then the table entry is inoperable and the CPE MUST set Status to Error_Misconfigured. **Note:** The IPv4Forwarding table includes a unique key parameter that is a strong reference. If a strongly referenced object is deleted, the CPE will set the referencing parameter to an empty string. However, doing so under these circumstances might cause the updated

Name	Type	Write	Description
			IPv4Forwarding row to then violate the table's unique key constraint; if this occurs, the CPE MUST disable the offending IPv4Forwarding row. At most one entry in this table (regardless of whether or not it is enabled) can exist with a given value for Alias. On creation of a new table entry, the CPE MUST choose an initial value for Alias such that the new entry does not conflict with any existing entries. At most, one enabled entry in this table can exist with all the same values for DestIPAddress, DestSubnetMask, ForwardingPolicy, GatewayIPAddress, Interface and ForwardingMetric.
Device.Routing.Router.{i}.IPv4Forwarding.{i}.Enable	boolean	-	Enables or disables the forwarding entry. On creation, an entry is disabled by default.
Device.Routing.Router.{i}.IPv4Forwarding.{i}.Status	string	-	Indicates the status of the forwarding entry. Possible values: ▪ Disabled ▪ Enabled ▪ Error_Misconfigured ▪ Error (OPTIONAL) The Error value MAY be used by the CPE to indicate a locally defined error condition. The Error_Misconfigured value indicates that a necessary configuration value is undefined or invalid.
Device.Routing.Router.{i}.IPv4Forwarding.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in Section 3.6.1/TR-069a4 and described in Appendix II/TR-069a4, the following mandatory constraints MUST be enforced: ▪ Its value MUST NOT be empty. ▪ Its value MUST start with a letter. ▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. ▪ The CPE MUST NOT change the parameter value.
Device.Routing.Router.{i}.IPv4Forwarding.{i}.StaticRoute	boolean	-	If "true", this route is a Static route.

Name	Type	Write	Description
Device.Routing.Router.{i}.IPv4Forwarding.{i}.DestIPAddress	string	W	Defines the Destination IPv4 address. An empty string indicates no destination address is specified. A Forwarding table entry for which DestIPAddress and DestSubnetMask are both an empty string is a default route.
Device.Routing.Router.{i}.IPv4Forwarding.{i}.DestSubnetMask	string	W	Defines IPv4Address - Destination subnet mask. An empty string indicates no destination subnet mask is specified. If a destination subnet mask is specified, the DestSubnetMask is ANDed with the destination address before comparing with the DestIPAddress. Otherwise, the full destination address is used as is. A Forwarding table entry for which DestIPAddress and DestSubnetMask are both an empty string is a default route.
Device.Routing.Router.{i}.IPv4Forwarding.{i}.GatewayIPAddress	string	W	Defines the IPv4 address of the gateway. Only one of GatewayIPAddress and Interface should be configured for a route. If both are configured, GatewayIPAddress and Interface MUST be consistent with each other.
Device.Routing.Router.{i}.IPv4Forwarding.{i}.Interface	string	W	The value MUST be the path name of a row in the IP.Interface table. If the referenced object is deleted, the parameter value MUST be set to an empty string. Specifies the egress layer 3 interface associated with this entry. Example: Device.IP.Interface.1. Only one of GatewayIPAddress and Interface SHOULD be configured for a route. If both are configured, GatewayIPAddress and Interface MUST be consistent with each other. For a route that was configured by setting GatewayIPAddress but not Interface, read access to Interface MUST return the full hierarchical parameter name for the routes egress interface.
Device.Routing.Router.{i}.IPv4Forwarding.{i}.ForwardingMetric	integer	W	Defines Forwarding metric. A value of -1 indicates this metric is not used.
Device.DNS.	object	-	Defines properties for Domain Name Service (DNS).
Device.DNS.SupportedRecordTypes	string	-	Defines the DNS record types that are supported by the device.

Name	Type	Write	Description
			This is a comma-separated list of strings.Each list item is an enumeration of: • A ([RFC1035] • AAAA ([RFC3596]) • SRV ([RFC2782]) • PTR ([RFC1035])
Device.DNS.Client.	object	-	Defines client properties for Domain Name Service (DNS). The DNS client resolves FQDN on behalf of device internal (client) applications.
Device.DNS.Client.Enable	boolean	-	Enables or disables the DNS client.
Device.DNS.Client.Status	string	-	Defines the status of the DNS client. Possible values: • Disabled • Enabled • Error (Optional) The Error value MAY be used by The CPE to indicate a locally defined Error condition.
Device.DNS.Client.ServerNumberOfEntries	unsignedInt	-	Defines the number of entries in the Server table.
Device.DNS.Client.Server.{i}.	object	-	This table contains the DNS Server IP address to be used by the DHCP Client (it does ''not'' model a DNS Server). Entries are either automatically created as result of DHCP (v4 or v6), IPCP, or RA received DNS server information, or are statically configured by the ACS.
Device.DNS.Client.Server.{i}.Enable	boolean	W	Enables or disables this entry.
Device.DNS.Client.Server.{i}.Status	string	-	Defines the status of this entry. Possible values: • Disabled • Enabled • Error (Optional) The Error value MAY be used by the CPE to indicate a locally defined error condition.
Device.DNS.Client.Server.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in [Section 3.6.1/TR-069a4] and described in [Appendix II/TR-069a4], the following

Name	Type	Write	Description
			mandatory constraints MUST be enforced: ▪ Its value MUST NOT be empty. ▪ Its value MUST start with a letter. ▪ f its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. ▪ The CPE MUST NOT change the parameter value.
Device.DNS.Client.Server.{i}.DNSServer	string	W	Defines the DNS server IP addresses. **Note:** DNSServer is only writable when Type is Static; otherwise, DNSServer is automatically configured as result of DHCP, IPCP, or RA received DNS server information.
Device.DNS.Client.Server.{i}.Interface	string	W	The value MUST be the path name of a row in the IP.Interface table. If the referenced object is deleted, the parameter value MUST be set to an empty string. This parameter specifies the IP interface over which the DNS query is sent. If an empty string is specified, the CPE MUST use its routing policy (Forwarding table entries), if necessary, to determine the appropriate interface. **Note:** Interface is only writable when Type is Static; otherwise, Interface is automatically configured as result of DHCP, IPCP, or RA received DNS server information.
Device.DNS.Client.Server.{i}.Type	string	W	Defines the method used to assign the DNSServer address. Possible values: ▪ DHCP (Replaced by DHCPv4, DEPRECATED) ▪ DHCPv4 ▪ DHCPv6 ▪ RouterAdvertisement ▪ IPCP ▪ Static Table entries that are automatically created as result of DHCP, IPCP, or RA received DNS server information will have Type set to DHCPv4, DHCPv6, IPCP, or RouterAdvertisement, as the case may be. Manually created table entires will have their Type set to Static.
Device.DNS.Relay.	object	-	Defines the DNS Relay object. The DNS proxy (or relay) function allows the forwarding of local network DNS queries to local or external DNS server(s) RFC 5625.

Name	Type	Write	Description
Device.DNS.Relay.Enable	boolean	W	Enables or disables the DNS Relay function.
Device.DNS.Relay.Status	string	-	Defines the status of the DNS relay. Possible values: • Disabled • Enabled • Error (Optional) The Error value MAY be used by the CPE to indicate a locally defined error condition.
Device.DNS.Relay.ForwardNumberOfEntries	unsignedInt	-	Defines the number of entries in the Forwarding table.
Device.DNS.Relay.Forwarding.{i}.	object	-	Defines the DNS Server forwarding policy to be used by the DNS Relay. Entries are either automatically created as result of DHCP (v4 or v6), IPCP, or RA received DNS server information, or are statically configured by the ACS. **Note:** Management of re-directing queries to the device embedded DNS server is not defined in this version of the specification.
Device.DNS.Relay.Forwarding.{i}.Enable	boolean	-	Enables or disables this entry.
Device.DNS.Relay.Forwarding.{i}.Status	string	-	Defines the status of this entry. Possible values: • Disabled • Enabled • Error (Optional) The Error value MAY be used by the CPE to indicate a locally defined error condition.
Device.DNS.Relay.Forwarding.{i}.Alias	string	-	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in Section 3.6.1/TR-069a4 and described in [Appendix II/TR-069a4], the following mandatory constraints MUST be enforced: • Its value MUST NOT be empty. • Its value MUST start with a letter. • If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. • The CPE MUST NOT change the parameter value.

Name	Type	Write	Description
Device.DNS.Relay.Forwarding.{i}.DNSServer	string	-	Defines the DNS server IP addresses. **Note:** DNSServer is only writable when Type is Static; otherwise, DNSServer is automatically configured as result of DHCP, IPCP, or RA received DNS server information.
Device.DNS.Relay.Forwarding.{i}.Interface	string	-	The value MUST be the path name of a row in the IP.Interface table. If the referenced object is deleted, the parameter value MUST be set to an empty string. Specifies the IP interface over which the DNS query is sent. If an empty string is specified, the CPE MUST use its routing policy (IP Forwarding table entries), if necessary, to determine the appropriate interface. **Note:** Interface is only writable when Type is Static; otherwise, Interface is automatically configured as result of DHCP, IPCP, or RA received DNS server information.
Device.DNS.Relay.Forwarding.{i}.Type	string	-	Defines the method used to assign the DNSServer address. Possible values: ▪ DHCP (Replaced by DHCPv4, DEPRECATED) ▪ DHCPv4 ▪ DHCPv6 ▪ RouterAdvertisement ▪ IPCP ▪ Static Table entries that are automatically created as result of DHCP, IPCP, or RA received DNS server information will have Type set to DHCPv4, DHCPv6, IPCP, or RouterAdvertisement, as the case may be. Manually created table entires will have their Type set to Static.
Device.NAT.	object	-	Defines the properties for Network Address Translation (NAT). The entire NAT object only applies to IPv4.
Device.NAT.InterfaceSettingNumberOfEntries	unsignedInt	-	Defines the number of entries in the InterfaceSetting table.
Device.NAT.PortMappingNumberOfEntries	unsignedInt	-	Defines the number of entries in the PortMapping table.
Device.NAT.InterfaceSetting.{i}.	object	-	Defines NAT settings for an associated IP Interface on which NAT is enabled. For enabled table entries, if Interface is not a valid reference then the table entry

Name	Type	Write	Description
			is inoperable and the CPE MUST set Status to Error_Misconfigured.
			Note: The InterfaceSetting table includes a unique key parameter that is a strong reference. If a strongly referenced object is deleted, the CPE will set the referencing parameter to an empty string. However, doing so under these circumstances might cause the updated InterfaceSetting row to then violate the table's unique key constraint; if this occurs, the CPE MUST set Status to Error_Misconfigured and disable the offending InterfaceSetting row.
			At most one entry in this table (regardless of whether or not it is enabled) can exist with a given value for Alias. On creation of a new table entry, the CPE MUST choose an initial value for Alias such that the new entry does not conflict with any existing entries.
			At most one enabled entry in this table can exist with a given value for Interface.
Device.NAT.InterfaceSetting.{i}.Enable	boolean	W	Enables or disables the InterfaceSetting entry, indicating if NAT is enabled for the referenced IP Interface instance. On creation, an InterfaceSetting entry is disabled by default.
Device.NAT.InterfaceSetting.{i}.Status	string	-	Defines the status of this entry. Possible values: • Disabled • Enabled • Enabled_NATForcedDisabled (NAT enabled but forced by a third party to be operationally disabled, e.g. because a MAP.Domain is enabled but there is no Basic Mapping Rule [RFC 7597]) • Enabled_PortMappingDisabled (NAT enabled but port mapping has been operationally disabled by a third party, e.g., because this is required by the current Firewall level) • Error_Misconfigured • Error The Error_Misconfigured value indicates that a necessary configuration value is undefined or invalid. The Error value may be used by the CPE to indicate a locally defined error condition.

Name	Type	Write	Description
Device.NAT.InterfaceSetting.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in Section 3.6.1/TR-069a4 and described in Appendix II/TR-069a4, the following mandatory constraints MUST be enforced: ▪ Its value MUST NOT be empty. ▪ Its value MUST start with a letter. ▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. ▪ The CPE MUST NOT change the parameter value.
Device.NAT.InterfaceSetting.{i}.Interface	string	W	The value MUST be the path name of a row in the IP.Interface table. If the referenced object is deleted, the parameter value MUST be set to an empty string. The associated IP interface on which NAT is to be enabled.
Device.NAT.PortMapping.{i}.	object	-	Defines the Port mapping table. This table MUST contain all NAT port mappings associated with this connection, including static and dynamic port mappings programmatically created via local control protocol, such as UPnP. This table MUST NOT contain dynamic NAT binding entries associated with the normal operation of NAT. If the CPE hosts a firewall, it is assumed that it will appropriately configure the firewall for the port mapping. For enabled table entries, if InternalClient is an empty string, or if Interface is not a valid reference and AllInterfaces is false, then the table entry is inoperable and the CPE MUST set Status to Error_Misconfigured. At most one entry in this table (regardless of whether or not it is enabled) can exist with a given value for Alias. On creation of a new table entry, the CPE MUST choose an initial value for Alias such that the new entry does not conflict with any existing entries. At most, one enabled entry in this table can exist with all the same values for RemoteHost, ExternalPort and Protocol.

Name	Type	Write	Description
Device.NAT.PortMapping.{i}.Enable	boolean	W	Enables or disables the port mapping instance. On creation, an entry is disabled by default.
Device.NAT.PortMapping.{i}.Status	string	-	Defines the status of this entry. Possible values: ▪ Disabled ▪ Enabled ▪ Error_Misconfigured ▪ Error (Optional) The Error_Misconfigured value indicates that a necessary configuration value is undefined or invalid. The Error value MAY be used by the CPE to indicate a locally defined error condition.
Device.NAT.PortMapping.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in Section 3.6.1/TR-069a4 and described in Appendix II/TR-069a4, the following mandatory constraints MUST be enforced: ▪ Its value MUST NOT be empty. ▪ Its value MUST start with a letter. ▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. ▪ The CPE MUST NOT change the parameter value. **Note:** There is no guarantee that the Alias value on automatically-created PortMapping instances will be retained. This is because automatically-created PortMapping instances can be transitory.
Device.NAT.PortMapping.{i}.Interface	string	W	The value MUST be the path name of a row in the IP.Interface table. If the referenced object is deleted, the parameter value MUST be set to an empty string. Specifies the IP interface to which this port mapping applies.
Device.NAT.PortMapping.{i}.AllInterfaces	boolean	W	Indicates whether this port mapping applies to all IP interfaces that support port mappings. If true, the value of Interface is ignored since all supported IP interfaces are indicated.
Device.NAT.PortMapping.{i}.LeaseDuration	unsignedInt	W	Determines the time to live, in seconds, of a port mapping lease, where "time to

Name	Type	Write	Description
			live" means the number of seconds before the port mapping expires. A value of 0 means the port mapping is static. Support for dynamic (non-static) port mappings is OPTIONAL. That is, the only value for LeaseDuration that MUST be supported is 0. For a dynamic (non-static) port mapping, when this parameter is read, the value represents the time remaining on the port mapping lease. That is, for a dynamic port mapping, the value counts down toward 0. When a dynamic port mapping lease expires, the CPE MUST automatically terminate that port mapping, and MUST automatically delete the corresponding PortMapping table entry.
Device.NAT.PortMapping.{i}.RemoteHost	string	W	This parameter is the IP address of the source of inbound packets. An empty string indicates a "wildcard", i.e. any IP address (this will be an empty string in most cases). CPE are REQUIRED only to support an empty string. When RemoteHost is an empty string, all traffic sent to the ExternalPort on the WAN interface of the gateway is forwarded to the IP.Interface associated with the InternalClient on the InternalPort. When RemoteHost is specified as one external IP address, the NAT will only forward inbound packets from this RemoteHost to the InternalClient, all other packets will be dropped. If a CPE supports non-empty values for RemoteHost, it MAY additionally support the ability to have more than one port mapping with the same ExternalPort and Protocol, but with differing values of RemoteHost. When wildcard values are used for RemoteHost and/or ExternalPort, the following precedence order applies (with the highest precedence listed first): ▪ Explicit RemoteHost, explicit ExternalPort ▪ Explicit RemoteHost, zero ExternalPort ▪ Empty RemoteHost, explicit ExternalPort ▪ Empty RemoteHost, zero ExternalPort If an incoming packet matches the criteria associated with more than one entry in this table, the CPE MUST apply the port

Name	Type	Write	Description
			mapping associated with the highest precedence entry.
Device.NAT.PortMapping.{i}.ExternalPort	unsignedInt	W	The external port (or the first port of a range of external ports) that the NAT gateway would listen on for traffic to a corresponding InternalPort. Inbound packets to this external port on the WAN interface SHOULD be forwarded to the IP.Interface associated with the InternalClient on the InternalPort. A value of zero (0) represents a "wildcard", i.e. any port number. If this value is 0, traffic on all external ports (that are not otherwise mapped) will be forwarded to InternalClient, and the value(s) of InternalPort on InternalClient are ignored. When wildcard values are used for RemoteHost and/or ExternalPort, the following precedence order applies (with the highest precedence listed first): ▪ Explicit RemoteHost, explicit ExternalPort ▪ Explicit RemoteHost, zero ExternalPort ▪ Empty RemoteHost, explicit ExternalPort ▪ Empty RemoteHost, zero ExternalPort If an incoming packet matches the criteria associated with more than one entry in this table, the CPE MUST apply the port mapping associated with the highest precedence entry.
Device.NAT.PortMapping.{i}.ExternalPortEndRange	unsignedInt	W	Indicates the last port of the external port range that starts with ExternalPort. If an external port range is specified, then the behavior described for ExternalPort applies to all ports within the range. A value of zero (0) indicates that no external port range is specified, i.e. that the range consists only of ExternalPort. If ExternalPort is zero (wildcard), the value of this parameter MUST be ignored. If specified, the value of this parameter MUST be greater than or equal to the value of ExternalPort.
Device.NAT.PortMapping.{i}.InternalPort	unsignedInt	W	Defines the port on InternalClient that the gateway should forward traffic to.
Device.NAT.PortMapping.{i}.Protocol	string	W	Defines the protocol of the port mapping. Possible values:

Name	Type	Write	Description
			▪ TCP ▪ UDP
Device.NAT.PortMapping.{i}.InternalClient	string	W	The IP address or DNS host name of an internal client (on the LAN). Support for an IP address is mandatory. If InternalClient is specified as an IP address and the LAN device's IP address subsequently changes, the port mapping MUST remain associated with the original IP address. Support for DNS host names is OPTIONAL. If InternalClient is specified as a DNS host name and the LAN device's IP address subsequently changes, the port mapping MUST remain associated with this LAN device. In this case, it is the responsibility of the CPE to maintain the name-to-address mapping in the event of IP address changes. This can be accomplished, for example, by assigning the DNS host name via use of DHCP option 12 (Host Name) or option 81 (FQDN). Note that the ACS can learn the host name associated with a given LAN device via the Hosts.Host. table. Read access to this parameter MUST always return the exact value that was last set by the ACS. For example, if the internal client is set to a DNS host name, it MUST read back as a DNS host name and not as an IP address. It MUST be possible to set the InternalClient to the broadcast IP address 255.255.255.255 for UDP mappings. This is to enable multiple NAT clients to use the same well-known port simultaneously.
Device.NAT.PortMapping.{i}.Description	string	W	Defines a user-readable description of this port mapping.
Device.DHCPv4.	object	-	Defines the Dynamic Host Configuration Protocol (DHCP) IPv4 object [RFC 2131]. This entire object applies to IPv4 only. It contains the Client, Server, and Relay objects.
Device.DHCPv4.ClientNumberOfEntries	unsignedInt	-	Defines the number of entries in the Client table.
Device.DHCPv4.Client.{i}.	object	-	This object contains DHCP client settings for an associated IP Interface indicated by Interface. For enabled table entries, if Interface is not a valid reference then the table entry is inoperable and the CPE MUST set Status to Error_Misconfigured.

Name	Type	Write	Description
			Note: The Client table includes a unique key parameter that is a strong reference. If a strongly referenced object is deleted, the CPE will set the referencing parameter to an empty string. However, doing so under these circumstances might cause the updated Client row to then violate the table's unique key constraint; if this occurs, the CPE MUST set Status to Error_Misconfigured and disable the offending Client row. This table MUST contain at least 1 entry. At most, one entry in this table (regardless of whether or not it is enabled) can exist with a given value for Alias. On creation of a new table entry, the CPE MUST choose an initial value for Alias such that the new entry does not conflict with any existing entries. At most, one enabled entry in this table can exist with a given value for Interface.
Device.DHCPv4.Client.{ i}.Enable	boolean	-	Enables or disables the DHCP Client entry.
Device.DHCPv4.Client.{ i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in Section 3.6.1/TR-069a4 and described in Appendix II/TR-069a4, the following mandatory constraints MUST be enforced: ▪ Its value MUST NOT be empty. ▪ Its value MUST start with a letter. ▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. ▪ The CPE MUST NOT change the parameter value.
Device.DHCPv4.Client.{ i}.Interface	string	-	The value MUST be the path name of a row in the IP.Interface table. If the referenced object is deleted, the parameter value MUST be set to an empty string. The IP Interface associated with the Client entry.
Device.DHCPv4.Client.{ i}.Status	string	-	Defines the status of this table entry. Possible values: ▪ Disabled ▪ Enabled ▪ Error_Misconfigured ▪ Error (Optional)

Name	Type	Write	Description
			The Error_Misconfigured value indicates that a necessary configuration value is undefined or invalid. The Error value MAY be used by the CPE to indicate a locally defined error condition.
Device.DHCPv4.Client.{i}.IPAddress	string	W	Defines the IPv4 Address option received from the DHCP Server. An empty string when Status is not equal to Bound.
Device.DHCPv4.Client.{i}.SubnetMask	string	W	Defines the IPv4Address Subnet mask option received from the DHCP Server. An empty string when Status is not equal to Bound. Value is information received via DHCP Option 1.
Device.DHCPv4.Client.{i}.LeaseTimeRemaining	integer	W	Defines DHCP lease time remaining in seconds. A value of -1 indicates an infinite lease.
Device.DHCPv4.Client.{i}.SentOptionNumberOfEntries	unsignedInt	-	Defines the number of entries in the SentOption table.
Device.DHCPv4.Client.{i}.ReqOptionNumberOfEntries	unsignedInt	-	Defines the number of entries in the ReqOption table.
Device.DHCPv4.Client.{i}.SentOption.{i}.	object	-	Defines each instance of this object represents a DHCP option that MUST, if enabled, be sent in DHCP client requests. All sent DHCP options MUST be listed.
Device.DHCPv4.Client.{i}.SentOption.{i}.Enable	boolean	-	Enables or disables this SentOption table entry.
Device.DHCPv4.Client.{i}.SentOption.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in [Section 3.6.1/TR-069a4] and described in [Appendix II/TR-069a4], the following mandatory constraints MUST be enforced: ▪ Its value MUST NOT be empty. ▪ Its value MUST start with a letter. ▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. ▪ The CPE MUST NOT change the parameter value.
Device.DHCPv4.Client.{i}.SentOption.{i}.Tag	unsignedInt	W	Defines the option tag as defined in RFC 2132.

Name	Type	Write	Description
Device.DHCPv4.Client.{i}.SentOption.{i}.Value	hexBinary	W	Defines a hexbinary encoded option value.
Device.DHCPv4.Client.{i}.ReqOption.{i}.	object	-	Each instance of this object represents a DHCP option that MUST, if enabled, be requested in DHCP client requests. All requested DHCP options MUST be listed.
Device.DHCPv4.Client.{i}.ReqOption.{i}.Enable	boolean	-	Enables or disables this ReqOption table entry.
Device.DHCPv4.Client.{i}.ReqOption.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in Section 3.6.1/TR-069a4 and described in Appendix II/TR-069a4, the following mandatory constraints MUST be enforced: ▪ Its value MUST NOT be empty. ▪ Its value MUST start with a letter. ▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. ▪ The CPE MUST NOT change the parameter value.
Device.DHCPv4.Client.{i}.ReqOption.{i}.Tag	unsignedInt	W	Defines the option tag as defined in RFC 2132.
Device.DHCPv4.Client.{i}.ReqOption.{i}.Value	hexBinary	W	Defines a hexbinary encoded, most recently received DHCP option value. If no option value has been received, then the value MUST represent {{empty}}. Received DHCP options values may, but need not, persist across CPE reboots.
Device.DHCPv4.Server.	object	-	Defines the DHCP server configuration.
Device.DHCPv4.Server.Enable	boolean	W	Enables or disables the DHCP server.
Device.DHCPv4.Server.PoolNumberOfEntries	unsignedInt	-	Defines the number of entries in the Pool table.
Device.DHCPv4.Server.Pool.{i}.	object	-	Defines the DHCP conditional serving pool table. Each instance of this object defines a DHCP conditional serving pool. Client requests are associated with pools based on criteria such as source interface, supplied DHCP options, and MAC address.

Name	Type	Write	Description
			Overlapping pool ranges MUST be supported.
			For enabled table entries, if Interface is not a valid reference, or MinAddress, MaxAddress, or SubnetMask is not a valid value, then the table entry is inoperable and the CPE MUST set Status to Error_Misconfigured.
			At most one entry in this table (regardless of whether or not it is enabled) can exist with a given value for Alias. On creation of a new table entry, the CPE MUST choose an initial value for Alias such that the new entry does not conflict with any existing entries.
Device.DHCPv4.Server .Pool.{i}.Enable	boolean	W	Enables or disables the Pool entry.
Device.DHCPv4.Server .Pool.{i}.Status	string	-	Defines the status of this entry. Possible values: ▪ Disabled ▪ Enabled ▪ Error_Misconfigured ▪ Error (Optional) The Error_Misconfigured value indicates that a necessary configuration value is undefined or invalid. The Error value MAY be used by the CPE to indicate a locally defined error condition.
Device.DHCPv4.Server .Pool.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in [Section 3.6.1/TR-069a4] and described in [Appendix II/TR-069a4], the following mandatory constraints MUST be enforced: ▪ Its value MUST NOT be empty. ▪ Its value MUST start with a letter. ▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. ▪ The CPE MUST NOT change the parameter value.
Device.DHCPv4.Server .Pool.{i}.Interface	string	-	The value MUST be the path name of a row in the IP.Interface table. If the referenced object is deleted, the parameter value MUST be set to an empty string. The IP Interface associated with the Pool entry.

Name	Type	Write	Description
Device.DHCPv4.Server .Pool.{i}.MinAddress	string	W	Defines the first IPv4 address in the pool to be assigned by the DHCP server on the LAN interface. The parameter value can be overwritten by dynamic values retrieved via a DHCP client with Client.{i}.PassthroughEnable or a PPP interface with PPP.Interface.{i}.IPCP.PassthroughEnable equal to true.
Device.DHCPv4.Server .Pool.{i}.MaxAddress	string	W	Defines the last IPv4 address in the pool to be assigned by the DHCP server on the LAN interface. The parameter value can be overwritten by dynamic values retrieved via a DHCP client with Client.{i}.PassthroughEnable or a PPP interface with PPP.Interface.{i}.IPCP.PassthroughEnable equal to true.
Device.DHCPv4.Server .Pool.{i}.SubnetMask	string	W	Defines the client's network subnet mask. (IPv4Address). The parameter value can be overwritten by dynamic values retrieved via a DHCP client with Client.{i}.PassthroughEnable or a PPP interface with PPP.Interface.{i}.IPCP.PassthroughEnable equal to true.
Device.DHCPv4.Server .Pool.{i}.DNSServers	string	W	Defines a comma-separated list (up to 4 items) of IPv4Addresses. List items represent DNS servers offered to DHCP clients. Support for more than three DNS Servers is optional. The parameter value can be overwritten by dynamic values retrieved via a DHCP client with Client.{i}.PassthroughEnable or a PPP interface with PPP.Interface.{i}.IPCP.PassthroughEnable equal to true.
Device.DHCPv4.Server .Pool.{i}.DomainName	string	W	Sets the domain name to provide to clients on the LAN interface.
Device.DHCPv4.Server .Pool.{i}.IPRouters	string	W	Defines a comma-separated list (up to 4 items) of IPv4Addresses. List items represent addresses of routers on this subnet. Also known as default gateway. Support for more than one Router address is optional. The parameter value can be overwritten by dynamic values retrieved via a DHCP client with Client.{i}.PassthroughEnable or a PPP interface with PPP.Interface.{i}.IPCP.PassthroughEnable equal to true.

Name	Type	Write	Description
Device.DHCPv4.Server.Pool.{i}.LeaseTime	integer	W	Specifies the lease time in seconds of client assigned addresses. A value of -1 indicates an infinite lease.
Device.DHCPv4.Server.Pool.{i}.StaticAddressNumberOfEntries	unsignedInt	-	Defines the number of entries in the StaticAddress table.
Device.DHCPv4.Server.Pool.{i}.OptionNumberOfEntries	unsignedInt	-	Defines the number of entries in the Option table.
Device.DHCPv4.Server.Pool.{i}.StaticAddress.{i}.	object	-	Defines the DHCP static address table. Entries in this table correspond to what [RFC2131] calls "manual allocation", where a client's IP address is assigned by the network administrator, and DHCP is used simply to convey the assigned address to the client. Each instance of this object specifies a hardware address (MAC address) and an IP address within the pool. When serving from this pool, this IP address MUST, if available, be assigned to the DHCP client with this hardware address, and MUST NOT be assigned to any other client. **Note:** It is possible that an IP address in this table is present in one or more of the other conditional serving pools, in which case it is possible that such an address will be assigned to a different client. At most one entry in this table (regardless of whether or not it is enabled) can exist with a given value for Alias. On creation of a new table entry, the CPE MUST choose an initial value for Alias such that the new entry does not conflict with any existing entries. At most one enabled entry in this table can exist with a given value for Chaddr.
Device.DHCPv4.Server.Pool.{i}.StaticAddress.{i}.Enable	boolean	-	Enables or disables the StaticAddress table entry. Disabling an entry does not return the IP address to the pool.
Device.DHCPv4.Server.Pool.{i}.StaticAddress.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in [Section 3.6.1/TR-069a4] and described in [Appendix II/TR-069a4], the following mandatory constraints MUST be enforced: ▪ Its value MUST NOT be empty.

Name	Type	Write	Description
			▪ Its value MUST start with a letter. ▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. ▪ The CPE MUST NOT change the parameter value.
Device.DHCPv4.Server.Pool.{i}.StaticAddress.{i}.Chaddr	string	W	Defines the Hardware address (MAC address) of the physical interface of the DHCP client.
Device.DHCPv4.Server.Pool.{i}.StaticAddress.{i}.Yiaddr	string	W	Defines the IPv4 address to be assigned by the DHCP server to the DHCP client with the specified hardware address (MAC address).
Device.DHCPv4.Server.Pool.{i}.Option.{i}.	object	-	This object specifies the DHCP options that MUST, if enabled, be returned to clients whose DHCP requests are associated with this pool.
Device.DHCPv4.Server.Pool.{i}.Option.{i}.Enable	boolean	-	Enables or disables this Option table entry.
Device.DHCPv4.Server.Pool.{i}.Option.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in [Section 3.6.1/TR-069a4] and described in [Appendix II/TR-069a4], the following mandatory constraints MUST be enforced: ▪ Its value MUST NOT be empty. ▪ Its value MUST start with a letter. ▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. ▪ The CPE MUST NOT change the parameter value.
Device.DHCPv4.Server.Pool.{i}.Option.{i}.Tag	unsignedInt	W	Defines the Option tag as defined in RFC 2132.
Device.DHCPv4.Server.Pool.{i}.Option.{i}.Value	hexBinary	W	Defines a hexbinary encoded option value.
Device.DHCPv4.Server.Pool.{i}.Client.{i}.	object	-	Defines a DHCPv4 client table. This table lists details of DHCPv4 clients that matched the filter criteria of this Pool entry. At most one entry in this table can exist with a given value for Alias, or with a given value for Chaddr.
Device.DHCPv4.Server.Pool.{i}.Client.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a

Name	Type	Write	Description
			mechanism for an ACS to label this instance for future reference.
			If the CPE supports the Alias-based Addressing feature as defined in [Section 3.6.1/TR-069a4] and described in [Appendix II/TR-069a4], the following mandatory constraints MUST be enforced:
			▪ Its value MUST NOT be empty.
			▪ Its value MUST start with a letter.
			▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix.
			▪ The CPE MUST NOT change the parameter value.
			Note: There is no guarantee that the Alias value on automatically-created Client instances will be retained. This is because automatically-created Client instances can be transitory.
Device.DHCPv4.Server.Pool.{i}.Client.{i}.Chaddr	string	-	Defines the MAC address of the DHCPv4 client.
Device.DHCPv4.Server.Pool.{i}.Client.{i}.Active	boolean	-	Defines whether or not the DHCPv4 client is currently present on the LAN. The method of presence detection is a local matter to the CPE.
			The ability to list inactive DHCPv4 clients is Optional. If the CPE includes inactive DHCPv4 clients in this table, Active MUST be set to "false" for each inactive DHCPv4 client. The length of time an inactive DHCPv4 client remains listed in this table is a local matter to the CPE.
Device.DHCPv4.Server.Pool.{i}.Client.{i}.IPv4AddressNumberOfEntries	unsignedInt	-	Defines the number of entries in the IPv4Address table.
Device.DHCPv4.Server.Pool.{i}.Client.{i}.OptionNumberOfEntries	unsignedInt	-	Defines the number of entries in the Option table.
Device.DHCPv4.Server.Pool.{i}.Client.{i}.IPv4Address.{i}.	object	-	Defines the IPv4 addresses assigned to this client.
Device.DHCPv4.Server.Pool.{i}.Client.{i}.IPv4Address.{i}.IPAddress	string	-	Defines the IPv4 address.
Device.DHCPv4.Server.Pool.{i}.Client.{i}.IPv4Address.{i}.LeaseTimeRemaining	dateTime	-	Defines the time at which the DHCP lease will expire or 0001-01-01T00:00:00Z if not known. For an infinite lease, the parameter value MUST be 9999-12-31T23:59:59Z.

Name	Type	Write	Description
Device.Firewall.	object	-	Defines the Firewall configuration object. The Config parameter enables and disables the Firewall, and can select either a predefined configuration (High or Low) or an explicitly-defined Advanced configuration. For an Advanced configuration, AdvancedLevel controls the currently active Firewall Level, and the Firewall Levels are defined in the Level, Chain and Chain.{i}.Rule tables. The Firewall rules modeled by this object operate only on the forwarding path. This means that they affect only routed traffic, and do not affect traffic that is destined for or generated by the CPE. Note that any NAT processing on the ingress packet occurs before Firewall rules are applied so, for example, the Firewall rules will see the translated destination IP address and port in a downstream packet that has passed through the NAT. See Appendix VIII/TR-181i2a2 for an example Advanced configuration.
Device.Firewall.Enable	boolean	-	Enables or disables the Firewall. Firewalls often implement additional Denial of Service and other vulnerability protections, such as those described in [ICSA-Baseline]. If a Stateful Firewall is enabled, then it is assumed that all implemented stateful protections are enabled, unless they are overridden by some other part of the data model.
Device.Firewall.Config	string	-	Defines ow this firewall is configured. Possible values: ▪ High (The firewall implements the "Traffic Denied Inbound" and "Minimally Permit Common Services Outbound" components of the ICSA residential certification's Required Services Security Policy [ICSA-Residential]. If DoS and vulnerability protections are implemented [ICSA-Baseline], these are enabled) ▪ Low (All Outbound traffic and pinhole-defined Inbound traffic is allowed. If DoS and vulnerability protections are implemented [ICSA-Baseline], these are enabled) ▪ Off (All Inbound and Outbound traffic is allowed, and the CPE is only protected by NAT settings (if supported and enabled). If DoS and

Name	Type	Write	Description
			vulnerability protections are implemented [ICSA-Baseline], these are disabled. This value is DEPRECATED because it is the same as setting Enable to false, DEPRECATED) ▪ Advanced (Advanced firewall configuration applies, as specified by AdvancedLevel, OPTIONAL) ▪ Vendors can extend the enumerated values with vendor specific extensions, in which case the rules outlined in Section 3.3/TR-106a2 MUST be adhered to.
Device.Firewall.AdvancedLevel	string	-	The value MUST be the path name of a row in the Firewall.Level table. If the referenced object is deleted, the parameter value MUST be set to an empty string. Selects the currently active Firewall Level. AdvancedLevel only applies when Config is Advanced.
Device.Firewall.Type	string	-	Defines the Firewall Type. Possible values: ▪ Stateless (The Firewall only implements stateless packet inspection) ▪ Stateful (The Firewall implements stateful packet inspection) A Stateless Firewall treats each packet individually and thus has no concept of sessions. Therefore a Stateless Firewall cannot distinguish between unsolicited downstream traffic and downstream traffic in response to outbound messages. This means that it has to accept all incoming downstream traffic. Furthermore, because a Stateless Firewall treats each packet individually, it does not provide any DoS or vulnerability protections. A Stateful Firewall maintains state about previous packets and thus supports sessions. Therefore a Stateful Firewall can distinguish between solicited and unsolicited downstream traffic. In a Stateful Firewall, explicitly configured rules only apply to unsolicited traffic, and can not cause packets received in response to an upstream request to be dropped. Furthermore, because a Stateful Firewall maintains state, it can provide DoS and vulnerability protections.

Name	Type	Write	Description
			A device that has a Stateless Firewall depends on the NAT to provide protection against unsolicited downstream IPv4 traffic. This is because, as noted above, a Stateless Firewall has to be configured to accept all incoming downstream traffic. For IPv6 there is no NAT, so a Stateless Firewall can not provide simple security protections against unsolicited downstream IPv6 traffic.
Device.Firewall.Version	string	-	Defines a string identifying the firewall settings version currently used in the CPE, or an empty string if the firewall settings are not associated with a version.
Device.Firewall.LastChange	dateTime	-	Defines the time at which the firewall settings most recently changed.
Device.Firewall.LevelNumberOfEntries	unsignedInt	-	Defines the number of entries in the Level table.
Device.Firewall.ChainNumberOfEntries	unsignedInt	-	Defines the number of entries in the Chain table.
Device.Firewall.Level.{i}.	object	-	Defines the Firewall Level table. When an Advanced configuration is selected, AdvancedLevel selects the currently active entry in this table. Each Level table entry references the Chain that contains the rules for this level. At most one entry in this table can exist with a given value for Alias, or with a given value for Name. On creation of a new table entry, the CPE MUST choose initial values for Alias and Name such that the new entry does not conflict with any existing entries.
Device.Firewall.Level.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in Section 3.6.1/TR-069a4 and described in Appendix II/TR-069a4, the following mandatory constraints MUST be enforced: ▪ Its value MUST NOT be empty. ▪ Its value MUST start with a letter. ▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. ▪ The CPE MUST NOT change the parameter value.

Name	Type	Write	Description
Device.Firewall.Level.{i}.Name	string	W	Defines a human-readable name associated with this Level entry.
Device.Firewall.Level.{i}.Description	string	W	Defines a human-readable description associated with this Level entry.
Device.Firewall.Level.{i}.Order	unsignedInt	W	Defines the position of the Level entry for user interface display. Levels can be presented according to an increasing or decreasing level of security. When this value is modified, if the value matches that of an existing entry, the Order value for the existing entry and all greater-valued Order entries is incremented to ensure uniqueness of this value. A deletion causes Order values to be compacted. When a value is changed, incrementing occurs before compaction. The value of Order on creation of a Level table entry MUST be one greater than the largest current value.
Device.Firewall.Level.{i}.Chain	unsignedInt	-	The value MUST be the path name of a row in the Firewall.Chain table. If the referenced object is deleted, the parameter value MUST be set to an empty string. The Chain containing Firewall Level Rules associated with this Level entry. On creation of a new Level entry, the device will automatically create a new Chain table entry that this Chain parameter will reference.
Device.Firewall.Level.{i}.PortMappingEnabled	boolean	W	Indicates whether NAT port mapping is enabled or disabled when this is the active Level. For a Stateless Firewall this can be set to false to force any port mappings to be operationally disabled (for a Stateful Firewall this is not necessary because the same effect can be achieved via Firewall rules). This parameter affects all the interfaces on which NAT is enabled. It operationally enables or disables port mapping functionality and therefore does not affect the individual NAT.PortMapping.{i}.Enable settings. Note that the current NAT status and NAT port mapping status are indicated by the NAT.InterfaceSetting.{i}.Status parameter.
Device.Firewall.Level.{i}.DefaultPolicy	string	W	Defines the default action for packets not matching any of the level rules. Possible values:

Name	Type	Write	Description
			- Drop (The firewall discards packets matching this rule) - Accept (The firewall forwards packets matching this rule) - Reject (The firewall discards packets matching this rule, and sends an ICMP message to the originating host, Optional)
Device.Firewall.Level.{i} .DefaultLogPolicy	boolean	W	Enables or disables logging, in a DeviceInfo.VendorLogFile, of packets not matching any of the level rules.
Device.Firewall.Chain. {i}.	object	-	Defines the value MUST be the path name of a row in the Firewall.Chain table. If the referenced object is deleted, the parameter value MUST be set to an empty string. The Chain containing Firewall Level Rules associated with this Level entry. On creation of a new Level entry, the device will automatically create a new Chain table entry that this Chain parameter will reference.
Device.Firewall.Chain.{i }.Enable	boolean	-	Enables or disables this Chain entry.
Device.Firewall.Chain.{i }.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in [Section 3.6.1/TR-069a4] and described in [Appendix II/TR-069a4], the following mandatory constraints MUST be enforced: - Its value MUST NOT be empty. - Its value MUST start with a letter. - If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. - The CPE MUST NOT change the parameter value.
Device.Firewall.Chain.{i }.Name	string	W	Defines a Human-readable name associated with this Chain entry.
Device.Firewall.Chain.{i }.Creator	string	-	Defines the creator of this Chain entry and of its Rules. Possible values - Defaults (The Chain entry is present in the factory default configuration) - PortMapping (The Chain entry was created as a side-effect of a [UPnP-IGD:1] or [TR-064] port mapping)

Name	Type	Write	Description
			▪ WANIPv6FirewallControl (The Chain entry was created by UPnP-IGD:2 WANIPv6FirewallControl) ▪ ACS (The Chain entry was created by the Auto Configuration Server) ▪ UserInterface (The Chain entry was created by device user interface or command line interface) ▪ Other (The Chain entry was created by another entity) **Note:** This is the entity that originally created the Chain entry. The value of this parameter does not change if the Chain entry or one of its rules is subsequently changed by another entity.
Device.Firewall.Chain.{i}.RuleNumberOfEntries	unsignedInt	-	Defines the number of entries in the Rule table.
Device.Firewall.Chain.{i}.Rule.{i}.	object	-	Defines the Firewall Rule table. Each entry defines a Firewall packet selection rule. The Target parameter defines the action to perform for traffic matching this rule: the packet can be dropped, accepted, rejected or passed to another Chain. This table MUST NOT contain dynamic Firewall rules associated with Stateful Firewall sessions. All entries are created by the creator of the parent Chain, as indicated by its Creator parameter. Rule entries in a Chain with a Creator of Defaults, ACS, UserInterface or (maybe) Other are referred to as Static Rules. Whether or not a Rule in a Chain with Creator Other is regarded as Static is a local matter to the CPE. Some of this object's parameter descriptions refer to whether a Rule is Static when specifying whether or not the parameter value can be modified. For enabled table entries, if SourceInterface is not a valid reference and SourceAllInterfaces is false, or if DestInterface is not a valid reference and DestAllInterfaces is false, then the table entry is inoperable and the CPE MUST set Status to Error_Misconfigured. At most one entry in this table (regardless of whether or not it is enabled) can exist with a given value for Alias. On creation of a new table entry, the CPE MUST choose an initial value for Alias such that the new entry does not conflict with any existing entries.

Name	Type	Write	Description
Device.Firewall.Chain.{i}.Rule.{i}.Enable	boolean	W	Enables or disables this Rule entry.
Device.Firewall.Chain.{i}.Rule.{i}.Status	string	-	Defines the status of this Rule entry. Possible values: ▪ Disabled ▪ Enabled ▪ Error_Misconfigured ▪ Error (Optional) The Error_Misconfigured value indicates that a necessary configuration value is undefined or invalid. The Error value MAY be used by the CPE to indicate a locally defined error condition.
Device.Firewall.Chain.{i}.Rule.{i}.Order	unsignedInt	W	Defines the osition of the Rule entry in the order of precedence. A value of 1 indicates the first entry considered (highest precedence). For each packet, the highest ordered entry that matches the rule criteria is applied. All lower order entries are ignored. When this value is modified, if the value matches that of an existing entry, the Order value for the existing entry and all lower Order entries is incremented (lowered in precedence) to ensure uniqueness of this value. A deletion causes Order values to be compacted. When a value is changed, incrementing occurs before compaction. The value of Order on creation of a Rule table entry MUST be one greater than the largest current value (initially assigned the lowest precedence).
Device.Firewall.Chain.{i}.Rule.{i}.Alias	string	W	Defines a non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature as defined in [Section 3.6.1/TR-069a4] and described in [Appendix II/TR-069a4], the following mandatory constraints MUST be enforced: ▪ Its value MUST NOT be empty. ▪ Its value MUST start with a letter. ▪ If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. ▪ The CPE MUST NOT change the parameter value.

Name	Type	Write	Description
Device.Firewall.Chain.{i}.Rule.{i}.Description	string	W	Defines a Human-readable description associated with this Rule entry.
Device.Firewall.Chain.{i}.Rule.{i}.Target	string	W	Defines an action to perform for traffic matching this Rule entry. Possible values: ▪ Drop (The firewall discards packets matching this rule) ▪ Accept (The firewall forwards packets matching this rule) ▪ Reject (The firewall discards packets matching this rule, and sends an ICMP message to the originating host, OPTIONAL) ▪ Return (The firewall doesn't consider the remaining rules (if any) in the current chain, OPTIONAL) ▪ TargetChain (The rules in the chain referenced by the TargetChain parameter are matched, OPTIONAL) This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.TargetChain	string	W	The value MUST be the path name of a row in the Firewall.Chain table. If the referenced object is deleted, the parameter value MUST be set to an empty string. Specifies the chain to process when Target equals TargetChain. If there are no matching rules in the referenced chain, processing continues with the next rule in this chain (if any). In other words, TargetChain behaves like a subroutine call. An empty string indicates no TargetChain is specified. This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.Log	boolean	W	Enables or disables logging, in a DeviceInfo.VendorLogFile, of packets matching this Rule. If the Rule is not Static (as explained in the object description), whether changes to this parameter persist across re-boot is a local matter to the CPE.
Device.Firewall.Chain.{i}.Rule.{i}.CreationDate	dateTime	-	Defines the Date and time when this Rule entry was created.
Device.Firewall.Chain.{i}.Rule.{i}.ExpiryDate	dateTime	W	Defines the the time at which this Rule entry will expire, or 0001-01-01T00:00:00Z if not known. For an infinite lifetime, the parameter value MUST be 9999-12-31T23:59:59Z.

Name	Type	Write	Description
			The only value that MUST be supported is 9999-12-31T23:59:59Z (i.e., support for rules that expire is Optional).
			When a rule expires, the CPE MUST automatically terminate that rule and MUST automatically delete the corresponding Rule table entry.
			This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.SourceInterface	string	W	Defines Rule criterion. The value MUST be the path name of a table row. If the referenced object is deleted, the parameter value MUST be set to an empty string. This specifies the ingress interface associated with the entry. It MAY be a layer 1, 2 or 3 interface, however, the types of interfaces for which Rules can be instantiated is a local matter to the CPE.
			This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.SourceInterfaceExclude	boolean	W	If "false", the rule matches only those packets that match the SourceInterface entry, if specified.
			If "true", the rule matches all packets except those that match the SourceInterface entry, if specified.
			This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.SourceAllInterfaces	boolean	W	Defines Rule criterion. This specifies that all ingress interfaces are associated with the entry. If true, the values of SourceInterface and SourceInterfaceExclude are ignored since all ingress interfaces are indicated. This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.DestInterface	string	W	Defines Rule criterion. The value MUST be the path name of a table row. If the referenced object is deleted, the parameter value MUST be set to an empty string. This specifies the egress interface associated with the entry. It MAY be a layer 1, 2 or 3 interface, however, the types of interfaces for which Rules can be instantiated is a local matter to the CPE.

Name	Type	Write	Description
			This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.DestInterfaceExclude	boolean	W	If "false", the rule matches only those packets that match the DestInterface entry, if specified. If "true", the rule matches all packets except those that match the DestInterface entry, if specified. This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.DestAllInterfaces	boolean	W	Defines Rule criterion. This specifies that all egress interfaces are associated with the entry. If true, the values of DestInterface and DestInterfaceExclude are ignored since all ingress interfaces are indicated. This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.IPVersion	integer	W	Defines Rule criterion. IP Protocol Version (e.g., 4 for IPv4 and 6 for IPv6). A value of -1 indicates this criterion is not used for matching. This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.DestIP	string	W	Defines Rule criterion. Destination IP address. An empty string indicates this criterion is not used for matching. **Note:** Firewall rules are applied after any NAT processing, so if NAT is enabled on the source interface this is always the translated address. This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.DestMask	string	W	Defines the Destination IP address mask, represented as an IP routing prefix using CIDR notation RFC 4632. The IP address part MUST be an empty string (and, if specified, MUST be ignored).
Device.Firewall.Chain.{i}.Rule.{i}.DestIPExclude	boolean	W	If "false", the rule matches only those packets that match the (masked) DestIP entry, if specified. If "true", the rule matches all packets except those that match the (masked) DestIP entry, if specified.

Name	Type	Write	Description
			This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.SourceIP	string	W	Defines Rule criterion. Source IP address. An empty string indicates this criterion is not used for matching. This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.SourceMask	string	W	Defines the Source IP address mask, represented as an IP routing prefix using CIDR notation RFC 4632. The IP address part MUST be an empty string (and, if specified, MUST be ignored).
Device.Firewall.Chain.{i}.Rule.{i}.SourceIPExclude	boolean	W	If "false", the rule matches only those packets that match the (masked) SourceIP entry, if specified. If "true", the rule matches all packets except those that match the (masked) SourceIP entry, if specified. This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.Protocol	integer	W	Rule criterion. Defines the Protocol number. A value of -1 indicates this criterion is not used for matching. This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.ProtocolExclude	boolean	W	If "false", the rule matches only those packets that match the Protocol entry, if specified. If "true", the rule matches all packets except those that match the Protocol entry, if specified. This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.DestPort	integer	W	Rule criterion. Defines the Destination port number. A value of -1 indicates this criterion is not used for matching. Note that Firewall rules are applied after any NAT processing, so if NAT is enabled on the source interface this is always the translated port number. This parameter can only be modified if the Rule is Static (as explained in the object description).

Name	Type	Write	Description
Device.Firewall.Chain.{i}.Rule.{i}.DestPortRangeMax	integer	W	Rule criterion. If specified, indicates the Rule criterion is to include the port range from DestPort through DestPortRangeMax (inclusive). If specified, DestPortRangeMax MUST be greater than or equal to DestPort. A value of -1 indicates that no port range is specified. **Note:** Firewall rules are applied after any NAT processing, so if NAT is enabled on the source interface this is always the translated port number. This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.DestPortExclude	boolean	W	If "false", the rule matches only those packets that match the DestPort entry (or port range), if specified. If "true", the rule matches all packets except those that match the DestPort entry (or port range), if specified. This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.SourcePort	integer	W	Rule criterion. Defines the Source port number. A value of -1 indicates this criterion is not used for matching. This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.SourcePortRangeMax	integer	W	Rule criterion. If specified, indicates the Rule criterion is to include the port range from SourcePort through SourcePortRangeMax (inclusive). If specified, SourcePortRangeMax MUST be greater than or equal to SourcePort. A value of -1 indicates that no port range is specified. This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.SourcePortExclude	boolean	W	If "false", the rule matches only those packets that match the SourcePort entry (or port range), if specified. If "true", the rule matches all packets except those that match the SourcePort entry (or port range), if specified. This parameter can only be modified if the Rule is Static (as explained in the object description).

Name	Type	Write	Description
Device.Firewall.Chain.{i}.Rule.{i}.DSCP	integer	W	Rule criterion. Defines the DiffServ codepoint (defined in RFC 2474). If set to a Class Selector Codepoint (defined in [RFC2474]), all DSCP values that match the first 3 bits will be considered a valid match. A value of -1 indicates this criterion is not used for matching. This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.Firewall.Chain.{i}.Rule.{i}.DSCPExclude	boolean	W	If "false", the rule matches only those packets that match the DSCP entry, if specified. If "true", the rule matchess all packets except those that match the DSCP entry, if specified. This parameter can only be modified if the Rule is Static (as explained in the object description).
Device.X_00908F.	object	-	This object contains vendor-defined parameters.
Device.X_00908F.DeviceInfo.	object	-	This object contains vendor-defined parameters.
Device.X_00908F.DeviceInfo.IncrementalConfig	string	W	Get – displays the current configuration (show run). Set – loads incremental CLI script file.
Device.X_00908F.Security.{i}.	object	W	This object defines Terminal Access Controller Access Control System (TACACS+) server addresses. A maximum of two servers can be defined (must have different addresses).
TACACSServer	string	W	Defines the IP address of the TACACS server.
Device.X_00908F.RemoteAccess	object	-	This object defines remote access parameters.
Device.X_00908F.RemoteAccess.SSH.Enable	boolean	W	Enables remote WAN access to the device through SSH.
Device.X_00908F.RemoteAccess.HTTPS.Enable	boolean	W	Enables remote WAN access to the device through HTTPS.
Device.X_00908F.RemoteAccess.SNMP.Enable	boolean	W	Enables remote WAN access to the device through SNMP.
Device.X_00908F.RemoteAccess.Telnet.Enable	boolean	W	Enables remote WAN access to the device through Telnet.

4.1 XMPP Connection Request

Extensible Messaging and Presence Protocol (XMPP) allows an ACS to initiate a session with any device, including devices that cannot be contacted directly by the ACS. This provides the equivalent functionality of the HTTP Connection Request, but makes use of a different mechanism, based on XMPP.

Table 4-2: XMPP Objects

Name	Type	Write	Description
ManagementServer.	Object	-	-
Device.ManagementServer.ConnReqXMPPConnection	String	W	The value MUST be the Path Name of the XMPP.Connection instance that is used for connection requests to this ManagementServer.
Device.ManagementServer.SupportedConnReqMethods	String	R	Comma-separated list of strings. The connection request methods supported by this ManagementServer. Each list item is an enumeration of: HTTP (The connection request method, REQUIRED) STUN (The connection request method, OPTIONAL) XMPP (The connection request method, OPTIONAL)
Device.ManagementServer.ConnReqAllowedJabberIDs	String	W	When an XMPP Connection Request is sent to a CPE the XMPP IQ Stanza will contian a "from" address that contains the Jabber ID of the initiating entity. This parameter is a comma-separated list (up to 32 items) of strings (maximum item length 256). Each entry represents a Jabber ID, or address, that is allowed to initiate an XMPP Connection Request. Each Jabber ID is allowed to be either a "Full JID" (containing a local-part, domain-part, and resource-part in the following format: "local-part@domain-part/resource-part") or a "Bare JID" (containing a local-part and a domain-part in the following format: "local-part@domain-part"). "Full JID" entries require an exact match whereas "Bare JID" entries are a wildcard match and will match any Jabber ID with the same local-part and domain-part independent of the resource-part. For example, if ConnReqAllowedJabberIDs contained ACS1@tr069.example.com, ACS2@tr-069.example.com/resource1 then the following incoming "from" addresses would be allowed: ACS1@tr069.example.com/resource1 ACS1@tr069.example.com/resource2

Name	Type	Write	Description
			ACS2@tr069.example.com/resource1
			And the following incoming "from" addresses would not be allowed:
			ACS2@tr069.example.com/resource2
			ACS@tr069.example.com/resource
			If this Parameter is an empty string then all Jabber IDs are allowed and an XMPP Connection Request can not be deemed invalid due to the "from" address.
Device.ManagementServer.ConnReqJabberID	String	R	The value MUST be the value of the XMPP.Connection.{i}.JabberID? contained in the XMPP.Connection? instance referenced by the ConnReqXMPPConnection parameter.
			If the ConnReqXMPPConnection parameter is empty or the XMPP.Connection? instance being reference is disabled, then this value will be empty.
			If the XMPP.Connection? instance being referenced by the ConnReqXMPPConnection parameter is lost (XMPP.Connection.{i}.Status? parameter changes from Enabled to some other value), then the value of this parameter MUST remain unchanged until the connection can be reestablished and a new XMPP.Connection.{i}.JabberID? can be established. If the new XMPP.Connection.{i}.JabberID? is identical to the existing value, then the value of this parameter MUST NOT be altered and a VALUE CHANGE event MUST NOT be generated.
Device.XMPP.ConnectionNumberOfEntries	unsignedInt	R	The number of entries in the Connection table.
Device.XMPP.Connection.{i}.	Object	W	The Connection represents a XMPP connection between the device and a server. The Username, Domain and Resource comprise the full identity (JabberID) of this Connection for this device.
			At most one entry in this table (regardless of whether or not it is enabled) can exist with a given value for Alias. On creation of a new table entry, the Agent MUST choose an initial value for Alias such that the new entry does not conflict with any existing entries.

Name	Type	Write	Description
			At most one enabled entry in this table can exist with all the same values for Username, Domain and Resource.
Device.XMPP.Connection.{i}.Enable	Boolean	W	Enables or disables this Connection. This parameter is based on ifAdminStatus from [RFC2863].
Device.XMPP.Connection.{i}.Alias	string(64)	W	[Alias] A non-volatile handle used to reference this instance. Alias provides a mechanism for an ACS to label this instance for future reference. If the CPE supports the Alias-based Addressing feature, the following mandatory constraints MUST be enforced: Its value MUST NOT be empty. Its value MUST start with a letter. If its value is not assigned by the ACS, it MUST start with a "cpe-" prefix. The CPE MUST NOT change the parameter value.
Device.XMPP.Connection.{i}.Username	string(256)	W	The proposed local-part of the Jabber ID of this Connection, and the value to be used to authenticate this Connection when making a connection to the Server.
Device.XMPP.Connection.{i}.Password	string(256)	W	Password used to authenticate this Connection when making a connection to the Server. Note that on a factory reset of the CPE, the value of this parameter might be reset to its factory value. If an ACS modifies the value of this parameter, it SHOULD be prepared to accommodate the situation that the original value is restored as the result of a factory reset. When read, this parameter returns an empty string, regardless of the actual value.
Device.XMPP.Connection.{i}.Domain	string(64)	W	The proposed domain-part of the Jabber ID of this Connection.
Device.XMPP.Connection.{i}.Resource	string(64)	W	The proposed resource-part of the Jabber ID of this Connection.
Device.XMPP.Connection.{i}.JabberID	string	R	The complete Jabber ID as determined by the first-hop XMPP server at time of connection establishment. This Jabber ID will usually be the standard concatenation of the local-part (Username), domain-part (Domain), and resource-part (Resource) as defined in [Section 2.1/RFC6120] (local-part@domain-part/resource-part), but since the Jabber ID is owned by the first-

Name	Type	Write	Description
			hop XMPP Server there are cases where the value will be different.
Device.XMPP.Connection.{i}.Status	string	R	The current operational state of this Connection. Enumeration of: Enabled Disabled Unknown Dormant Connecting ServerNotPresent Error_Misconfigured Error_AuthenticationFailure Error (OPTIONAL) When Enable is false then Status SHOULD normally be Disabled or Error if there is a fault condition on the interface). When Enable is changed to true then Status SHOULD change to Enabled if and only if the interface is able to transmit and receive PDUs; it SHOULD change to Dormant if and only if the interface is operable but is waiting for external actions before it can transmit and receive network traffic (and subsequently change to Enabled if still operable when the expected actions have completed); it SHOULD change to Unknown if the state of the interface can not be determined for some reason.
Device.XMPP.Connection.{i}. LastChangeDate	dateTime	R	The date and time at which this Connection entered into its current operational state.

This page is intentionally left blank.

5   TR-098 Data Model

TR-098 specifies the InternetGatewayDevice (IGD) data model for TR-069 enabled devices. This data model is a set of parameters, modelled in a tree structure, which can be managed using CWMP.

5.1   InternetGatewayDevice - Proprietary Parmeters

The name of AudioCodes-specific parameters and objects has the following form:

X_00908F_<vendor specific parameter or object name>

Table 5-1: Proprietary InternetGatewayDevice Table

Name	Type	Write	Description
InternetGatewayDevice.X_00908F_Access List.{i}.	object	W	Access to the Access List table.
X_00908F_AccessList_Action	string	W	Defines the Access List action: ▪ permit ▪ deny ▪ permit stateless
X_00908F_AccessList_Dest	IP address (x.x.x.x)	W	Defines the Access List destination (IP address).
X_00908F_AccessList_DestMask	IP address (x.x.x.x)	W	Defines the Access List destination mask.
X_00908F_AccessList_DestPort	unsignedInt	W	Defines the start destination port (0-65535).
X_00908F_AccessList_DestPortEnd	unsignedInt	W	Defines the end destination port (0-65535). For one port, define the same as the source port start. For any port, define start and end to 0.
X_00908F_AccessList_DSCPMask	IP address (x.x.x.x)	W	Defines the Access List DiffServ codepoint mask.
X_00908F_AccessList_DSCPValue	unsignedInt	W	Defines the Access List DiffServ codepoint (0-63; default is 0).
X_00908F_AccessList_Log	boolean	W	Logs the Access List actions.
X_00908F_AccessList_Name	string	W	Defines the Access List name (up to 25 characters).
X_00908F_AccessList_Protocol	unsignedInt	W	Defines the Access List protocol (0-255).
X_00908F_AccessList_Source	IP address (x.x.x.x)	W	Defines the Access List source address.
X_00908F_AccessList_SourceMask	IP address (x.x.x.x)	W	Defines the Access List source mask.

Name	Type	Write	Description
X_00908F_AccessList_SourcePort	unsignedInt	W	Defines the start source port (0-65535).
X_00908F_AccessList_SourcePortEnd	unsignedInt	W	Defines the end source port (0-65535). For one port, define the same as the source port start. For any port, define the start and end to 0.
InternetGatewayDevice.DeviceConfig.	object	W	This object contains general configuration parameters.
X_00908F_Trusted_IP_1_IP_Address	string	W	Defines a Trusted IP address for remote management of the device.
X_00908F_Trusted_IP_1_IP_Mask	string	W	Defines a Trusted IP mask for remote management of the device.
X_00908F_UserRestrictionEnable	unsignedInt	W	Defines administrative lease limit.
X_00908F_CellularMode	string	W	Defines the Cellular mode: ▪ cdc_ether ▪ ppp
X_00908F_RemoteManagementHTTPEnable	boolean	W	Allows acces to HTTP from the WAN.
X_00908F_RemoteManagementHTTPSEnable	boolean	W	Allows acces to HTTPS from the WAN.
X_00908F_RemoteManagementSNMPEnable	boolean	W	Allows acces to SNMP from the WAN.
X_00908F_RemoteManagementSNMPEnable	boolean	W	Allows acces to SNMP from the WAN.
X_00908F_RemoteManagementSSHEnable	boolean	W	Allows acces to SSH from the WAN.
X_00908F_RemoteManagementTelnetEnable	boolean	W	Allows acces to Telnet from the WAN.
InternetGatewayDevice.DeviceInfo.	object	-	This object contains information about the INI file.
X_00908F_ConfigurationVersion	unsignedInt	W	Contains the INI file version number that is reported in the acEV_BOARD_STARTED event.
InternetGatewayDevice.X_00908F_Admin. {i}.	object	-	Performs administration operations on the device.
X_00908F_Admin.State	string	W	Set - defines the action: ▪ Lock – locks the device ▪ Unlocked – unlocks the device Get - displays the device state: ▪ "Lock" ▪ "Unlocked" ▪ "ShuttingDown"
X_00908F_Admin.GracePeriod	Unsigned Integer	W	Defines the graceful period (in seconds) after which the device is locked. For more

Name	Type	Write	Description
			information, refer to the device's *User's Manual*.
X_00908F_Admin.Timer	Unsigned Integer	R	Displays the remaining time until the graceful period ends.
InternetGatewayDevice.DownloadDiagnostics.	object	-	This object defines the diagnostics configuration for a HTTP and FTP DownloadDiagnostics Test. Files received in the DownloadDiagnostics do not require file storage on the CPE device.
X_00908F_Speed	string	-	Textual result of download speed test. Example: 8.1 Mbps.
X_00908F_TestDuration	Unsigned Integer	-	Time took to run the download Speed Test.
InternetGatewayDevice.UploadDiagnostics.	object	-	This object defines the diagnostics configuration for a HTTP or FTP UploadDiagnostics test. Files sent by the UploadDiagnostics do not require file storage on the CPE device, and MAY be an arbitrary stream of bytes.
X_00908F_Speed	string	-	Textual result of upload speed test. Example: 8.1 Mbps.
X_00908F_TestDuration	Unsigned Integer	-	Time took to run the upload Speed Test.
InternetGatewayDevice.WANDevice.{i}	Object	-	Each instance contains all objects associated with a particular physical WAN interface.
X_00908F_IGMPSnoopingEnable	Boolean		
InternetGatewayDevice.WANDevice.{i}.WANConnectionDevice.{i}.WANPPPConnection.{i}.	object	W	This object enables configuration of PPP connections on the WAN interface of a CPE. This object is required for all WANConnectionDevices that employ PPP addressing, and must not be active for WANConnectionDevices that do not employ PPP addressing.
X_00908F_CellularPincode	char[5]	W	Defines the cellular pincode.
X_00908F_CellularModemError	string	-	Displays the last cellular modem status error: ▪ SIM PIN required ▪ SIM PUK required ▪ SIM failure ▪ SIM busy ▪ SIM wrong ▪ incorrect password ▪ SIM PIN2 required ▪ SIM PUK2 required
X_00908F_CellularOperator	string	-	Displays the cellular operator.

Name	Type	Write	Description
X_00908F_CellularSignalStrength	string	-	Displays the cellular signal strength (RSSI) in dBm.
X_00908F_CellularRoamStatus	string	-	Displays whether the cellular modem is at home or in roaming.
X_00908F_AccessList_Name_In	string	W	Defines the AccessList name to filter the incomming traffic on this interface.
X_00908F_FirewallEnable	boolean	W	Activates the firewall on this interface.
X_00908F_MaxMTUSize	unsignedInt	W	Defines the maximum MTU in bytes on this interface. For automatic, set it to 0.
X_00908F_WANPPP_AccessList_Name_Out	string	W	Defines the AccessList name to filter the outgoing traffic on this interface.
InternetGatewayDevice.WANDevice.{i}.WANConnectionDevice.{i}.WANIPConnection.{i}.	object	W	This object representing the WAN IP connection device.
X_00908F_AccessList_Name_In	string	W	Defines the AccessList name to filter the incomming traffic on this interface.
X_00908F_AccessList_Name_Out	string	W	Defines the AccessList name to filter the outgoing traffic on this interface.
X_00908F_DHCP_DefaultRoute	boolean	W	Sets the received DHCP gateway address as the default route.
X_00908F_FirewallEnable	boolean	W	Activates the firewall on this interface.
InternetGatewayDevice.LANDevice.{i}.WLANConfiguration.{i}.	object	-	This object contains information about the WLAN channel.
X_00908F_WLANConfiguration_ChannelRescan	boolean	W	Defines the WLAN Channel Auto-rescan. Applied only if channel was chosen automatically. When there is no traffic, periodic rescan occurs to locate the better channel.
InternetGatewayDevice.LANDevice.{i}.	object	-	
X_0090F8_Name	string	-	Defines the LAN interface name.
InternetGatewayDevice.LANDevice.{i}.WLANConfiguration.{i}.	object	-	This object models an 802.11 LAN connection on a CPE device. This object MUST be implemented for CPE that contain an 802.11 interface on the LAN side.
X_00908F_ChannelRescan	boolean	W	If this option is turned on, the Wi-Fi will periodically rescan the spectrum for a less congested band (unless already connected to a client).
InternetGatewayDevice.UserInterface.	object	-	This object contains information about user accounts.
X_00908F_AdminUser_Password	string	W	Defines the password for the "Admin" user account (Web and CLI).
X_00908F_AdminUser_UserName	string	W	Defines the username for the "Admin" user account (Web and CLI).

Name	Type	Write	Description
X_00908F_User_Password	string	W	Defines the password for the "User" account (Web and CLI)
X_00908F_User_UserName	string	W	Defines the username for the "User" account (Web and CLI).
Services.X_00908F_Firewall.{i}.	object	W	Network access list (internal firewall).
X_00908F_ByteBurst	integer	W	Defines the allowed traffic burst in bytes (0 is unused).
X_00908F_ByteRate	integer	W	Defines the allowed traffic in bytes per second (0 is unused)
X_00908F_EndPort	unsignedInt	W	Defines the port range end.
X_00908F_InterfaceID	string	W	Defines the interface name (none for all).
X_00908F_PacketSize	integer	W	Defines the maximum packet size (0 is unused).
X_00908F_PrefixLen	unsignedInt	W	Defines the prefix length of the source IP address (defining a subnet).
X_00908F_Protocol	string	W	Defines the IP user-level protocol (TCP, UDP, ICMP, ESP, SIP, MGCP, TPNCP, ANY or numeric value)
X_00908F_SourceIP	unsignedInt	W	Defines the source IP address for the Access rule.
X_00908F_SourcePort	unsignedInt	W	Defines the source port (0 for ALL).
X_00908F_StartPort	unsignedInt	W	Defines the port range start.
X_00908F_UseSpecificInterface	boolean	W	Defines the specific interface or for all the interfaces on which the rule is applied.
X_00908Fl_AllowType	string	W	Allows or blocks traffic matching the rule.
Services.VoiceService.{i}.VoiceProfile.{i}.SIP.	object	-	Voice line parameters that are specific to SIP call signaling.
X_00908F_ProxyName	string	W	
InternetGatewayDevice.ManagementServer.X_00908F_RootCertificate.{i}.	object	-	This object adds and deletes certificates.
Device.ManagementServer.X_00908F_RootCertificate.{i}.Enabled	boolean	-	Displays if this root certificate is enabled for validity checking.
Device.ManagementServer.X_00908F_RootCertificate.{i}.Certificate	base64(4096)	W	The DER encoded certificate.
Device.ManagementServer.X_00908F_RootCertificate.{i}.LastModif	dateTime	-	The last modification time of this certificate.
Device.ManagementServer.X_00908F_RootCertificate.{i}.SerialNumber	string(64)	-	The Serial Number field in an X.509 certificate (see RFC 5280).
Device.ManagementServer.X_00908F_RootCertificate.{i}.Issuer	string(256)	-	The Issuer field in an X.509 certificate (see RFC 5280). In other words, the Distinguished

Name	Type	Write	Description
			Name (DN) of the entity who has signed the certificate.
Device.ManagementServer.X_00908F _RootCertificate.{i}.NotBefore	dateTime	-	The beginning of the certificate validity period. In other words, the Not Before field in an X.509 certificate (see RFC 5280).
Device.ManagementServer.X_00908F _RootCertificate.{i}.NotAfter	dateTime	-	The end of the certificate validity period. In other words, the Not After field in an X.509 certificate (see RFC 5280).
Device.ManagementServer.X_00908F _RootCertificate.{i}.Subject	string(256)	-	The Distinguished Name (DN) of the entity associated with the Public Key. In other words, the Subject field in an X.509 certificate (see RFC 5280).
Device.ManagementServer.X_00908F _RootCertificate.{i}.SubjectAlt	string(256)	-	Comma-separated list (maximum length 256) of strings. Each item is a DNS Name. The Subject Alternative Names extension field in an X.509 certificate (see RFC 5280).
Device.ManagementServer.X_00908F _RootCertificate.{i}.SignatureAlgorithm	string(256)	-	The algorithm used in signing the certificate. In other words, the Signature Algorithm field in an X.509 certificate (see RFC 5280).

5.2   InternetGatewayDevice - Standard Parameters

The following table lists the standard InternetGatewayDevice objects and parameters.

Table 5-2: Standard InternetGatewayDevice Table

Name	Type	Write	Description
InternetGatewayDevice.	object	-	
LANDeviceNumberOfEntries	unsigned Int	-	Number of instances of LANDevice.
WANDeviceNumberOfEntries	unsigned Int	-	Number of instances of WANDevice.
InternetGatewayDevice.DeviceSummary	string	-	As defined in [TR-106a1]. This parameter is DEPRECATED because DeviceInfo.SupportedDataModel and associated Device Type XML documents (DT Instances) provide a more granular and scalable way of describing the device's data model. Therefore its value MAY be an empty string if (and only if) DeviceInfo.SupportedDataModel is supported.
InternetGatewayDevice.CaptivePortal.	object	-	This object contains parameters relating to the captive portal configuration on the CPE. The captive portal configuration defines the CPE's WAN-destined HTTP (port 80) traffic redirect behavior. When the captive portal is disabled, WAN-destined HTTP (port 80) traffic MUST be permitted to all destinations. When the captive portal is enabled, WAN-destined HTTP (port 80) traffic MUST be permitted only to destinations listed in the AllowedList; traffic to all other destinations MUST be redirected to the CaptivePortalURL.
Enable	string	W	Enables or disables the captive portal.
InternetGatewayDevice.DeviceConfig.	object	-	This object contains general configuration parameters.
InternetGatewayDevice.DeviceInfo.	object	-	This object contains general device information.
AdditionalHardwareVersion	string	-	Any additional hardware version information the vendor might wish to supply.
AdditionalSoftwareVersion	string	-	Any additional software version information the vendor might wish to supply.
Description	string	-	A full description of the CPE device (human readable string).
DeviceLog	string	-	Vendor-specific log(s).

Name	Type	Write	Description
EnabledOptions	string	-	This parameter is DEPRECATED because the "voucher mechanism", as defined in [Annex C/TR-069a3] is deprecated in favor of the "Software Module Management mechanism" as described in [Appendix II/TR-157a3]. Comma-separated list (maximum list length 1024) of strings. The OptionName of each Option that is currently enabled in the CPE. The OptionName of each is identical to the OptionName element of the OptionStruct described in [TR-069a3]. Only those options are listed whose State indicates the option is enabled.
FirstUseDate	dateTime	-	Date and time in UTC that the CPE first both successfully established an IP-layer network connection and acquired an absolute time reference using NTP or equivalent over that network connection. The CPE MAY reset this date after a factory reset. If NTP or equivalent is not available, this parameter, if present, SHOULD be set to the Unknown Time value.
HardwareVersion	string	-	A string identifying the particular CPE model and version.
Manufacturer	string	-	The manufacturer of the CPE (human readable string).
ManufacturerOUI	string	-	Organizationally unique identifier of the device manufacturer. Represented as a six hexadecimal-digit value using all upper-case letters and including any leading zeros. Possible patterns: ▪ [0-9A-F]{6} ▪ The value MUST be a valid OUI as defined in [OUI]. ▪ This value MUST remain fixed over the lifetime of the device, including across firmware updates. Any change would indicate that it's a new device and would therefore require a BOOTSTRAP Inform.
ModelName	string	-	Model name of the CPE (human readable string).
ModemFirmwareVersion	string	-	A string identifying the version of the modem firmware currently installed in the CPE. This is applicable only when the modem firmware is separable from the overall CPE software.
ProductClass	string	-	Identifier of the class of product for which the serial number applies. That is, for a given manufacturer, this parameter is used to identify the product or class of product

Name	Type	Write	Description
			over which the SerialNumber parameter is unique. This value MUST remain fixed over the lifetime of the device, including across firmware updates. Any change would indicate that it's a new device and would therefore require a BOOTSTRAP Inform.
ProvisioningCode	string	W	Identifier of the primary service provider and other provisioning information, which MAY be used by the ACS to determine service provider-specific customization and provisioning parameters. If not empty, this argument SHOULD be in the form of a hierarchical descriptor with one or more nodes specified. Each node in the hierarchy is represented as a 4-character sub-string, containing only numerals or upper-case letters. If there is more than one node indicated, each node is separated by a "." (dot). Examples: "TLCO" or "TLCO.GRP2".
SerialNumber	string	-	Identifier of the particular device that is unique for the indicated class of product and manufacturer. This value MUST remain fixed over the lifetime of the device, including across firmware updates. Any change would indicate that it's a new device and would therefore require a BOOTSTRAP Inform.
SoftwareVersion	string	-	A string identifying the software version currently installed in the CPE. To allow version comparisons, this element SHOULD be in the form of dot-delimited integers, where each successive integer represents a more minor category of variation. For example, ''3.0.21'' where the components mean: ''Major.Minor.Build''.
SpecVersion	string	-	Represents the version of the specification implemented by the device. Currently 1.0 is the only available version. The value of this parameter MUST equal "1.0". This parameter is DEPRECATED because its value is fixed and it therefore serves no purpose. However, it is a Forced Inform parameter and therefore cannot be OBSOLETED.
UpTime	string	-	Time in seconds since the CPE was last restarted.
InternetGatewayDevice.DownloadDiagnostics.	object	-	This object defines the diagnostics configuration for a HTTP and FTP DownloadDiagnostics Test.

Name	Type	Write	Description
			Files received in the DownloadDiagnostics do not require file storage on the CPE device.
DiagnosticsState	string	W	Indicate the availability of diagnostic data: ▪ None (READONLY) ▪ Requested ▪ Completed (READONLY) ▪ Error_InitConnectionFailed (READONLY) ▪ Error_NoResponse (READONLY) ▪ Error_TransferFailed (READONLY) ▪ Error_PasswordRequestFailed (READONLY) ▪ Error_LoginFailed (READONLY) ▪ Error_NoTransferMode (READONLY) ▪ Error_NoPASV (READONLY) ▪ Error_IncorrectSize (READONLY) ▪ Error_Timeout (READONLY) If the ACS sets the value of this parameter to Requested, the CPE MUST initiate the corresponding diagnostic test. When writing, the only allowed value is Requested. To ensure the use of the proper test parameters (the writable parameters in this object), the test parameters MUST be set either prior to or at the same time as (in the same SetParameterValues) setting the DiagnosticsState to Requested. When requested, the CPE SHOULD wait until after completion of the communication session with the ACS before starting the diagnostic. When the test is completed, the value of this parameter MUST be either Completed (if the test completed successfully), or one of the Error values listed above. If the value of this parameter is anything other than Completed, the values of the results parameters for this test are indeterminate. When the diagnostic initiated by the ACS is completed (successfully or not), the CPE MUST establish a new connection to the ACS to allow the ACS to view the results, indicating the Event code 8 DIAGNOSTICS COMPLETE in the Inform message. After the diagnostic is complete, the value of all result parameters (all read-only parameters in this object) MUST be retained by the CPE until either this diagnostic is run again, or the CPE reboots. After a reboot, if the CPE has not retained the result parameters from the most recent

Name	Type	Write	Description
			test, it MUST set the value of this parameter to None.
			Modifying any of the writable parameters in this object except for this one MUST result in the value of this parameter being set to None.
			While the test is in progress, modifying any of the writable parameters in this object except for this one MUST result in the test being terminated and the value of this parameter being set to None.
			While the test is in progress, setting this parameter to Requested (and possibly modifying other writable parameters in this object) MUST result in the test being terminated and then restarted using the current values of the test parameters.
Interface	string	-	The value MUST be the path name of the IP-layer interface over which the test is to be performed.
			The value of this parameter MUST be either a valid interface or an empty string. An attempt to set this parameter to a different value MUST be rejected as an invalid parameter value.
			If an empty string is specified, the CPE MUST use the default routing interface.
DownloadURL	string	W	The URL, as defined in [RFC3986], for the CPE to perform the download on. This parameter MUST be in the form of a valid HTTP [RFC2616] or FTP [RFC862] URL.
			When using FTP transport, FTP binary transfer MUST be used.
			When using HTTP transport, persistent connections MUST be used and pipelining MUST NOT be used.
			When using HTTP transport the HTTP Authentication MUST NOT be used.
DSCP	Unsigned Integer	-	The DiffServ code point for marking packets transmitted in the test.
			The default value SHOULD be zero.
EthernetPriority	Unsigned Integer	-	Ethernet priority code for marking packets transmitted in the test (if applicable).
			The default value SHOULD be zero.
ROMTime	dateTime	-	Request time in UTC, which MUST be specified to microsecond precision.
			For example: 2008-04-09T15:01:05.123456
			For HTTP this is the time at which the client sends the GET command.

Name	Type	Write	Description
			For FTP this is the time at which the client sends the RTRV command.
BOMTime	dateTime	-	Begin of transmission time in UTC, which MUST be specified to microsecond precision. For example: 2008-04-09T15:01:05.123456 For HTTP this is the time at which the first data packet is received. For FTP this is the time at which the client receives the first data packet on the data connection.
EOMTime	dateTime	-	End of transmission in UTC, which MUST be specified to microsecond precision. For example: 2008-04-09T15:01:05.123456 For HTTP this is the time at which the last data packet is received.For FTP this is the time at which the client receives the last packet on the data connection.
TestBytesReceived	Unsigned Integer	-	The test traffic received in bytes during the FTP/HTTP transaction including FTP/HTTP headers, between BOMTime and EOMTime.
TCPOpenRequestTime	dateTime	-	Request time in UTC, which MUST be specified to microsecond precision. For example: 2008-04-09T15:01:05.123456 ▪ For HTTP, this is the time at which the TCP socket open (SYN) was sent for the HTTP connection. ▪ For FTP, this is the time at which the TCP socket open (SYN) was sent for the data connection. Note: Interval of 1 microsecond SHOULD be supported.
TCPOpenResponseTime	dateTime	-	Response time in UTC, which MUST be specified to microsecond precision. For example: 2008-04-09T15:01:05.123456 ▪ For HTTP, this is the time at which the TCP ACK to the socket opening the HTTP connection was received. ▪ For FTP, this is the time at which the TCP ACK to the socket opening the data connection was received. Note: Interval of 1 microsecond SHOULD be supported.
InternetGatewayDevice.UploadDiagnostics	object	-	This object defines the diagnostics configuration for a HTTP or FTP UploadDiagnostics test. Files sent by the UploadDiagnostics do not require file storage on the CPE device, and MAY be an arbitrary stream of bytes.

Name	Type	Write	Description
DiagnosticsState	String	-	Indicate the availability of diagnostic data: ▪ None (READONLY) ▪ Requested ▪ Completed (READONLY) ▪ Error_InitConnectionFailed (READONLY) ▪ Error_NoResponse (READONLY) ▪ Error_PasswordRequestFailed (READONLY) ▪ Error_LoginFailed (READONLY) ▪ Error_NoTransferMode (READONLY) ▪ Error_NoPASV (READONLY) ▪ Error_NoCWD (READONLY) ▪ Error_NoSTOR (READONLY) ▪ Error_NoTransferComplete (READONLY) If the ACS sets the value of this parameter to Requested, the CPE MUST initiate the corresponding diagnostic test. When writing, the only allowed value is Requested. To ensure the use of the proper test parameters (the writable parameters in this object), the test parameters MUST be set either prior to or at the same time as (in the same SetParameterValues) setting the DiagnosticsState to Requested. When requested, the CPE SHOULD wait until after completion of the communication session with the ACS before starting the diagnostic. When the test is completed, the value of this parameter MUST be either Completed (if the test completed successfully), or one of the Error values listed above. If the value of this parameter is anything other than Completed, the values of the results parameters for this test are indeterminate. When the diagnostic initiated by the ACS is completed (successfully or not), the CPE MUST establish a new connection to the ACS to allow the ACS to view the results, indicating the Event code 8 DIAGNOSTICS COMPLETE in the Inform message. After the diagnostic is complete, the value of all result parameters (all read-only parameters in this object) MUST be retained by the CPE until either this diagnostic is run again, or the CPE reboots. After a reboot, if the CPE has not retained the result parameters from the most recent

Name	Type	Write	Description
			test, it MUST set the value of this parameter to None.
			Modifying any of the writable parameters in this object except for this one MUST result in the value of this parameter being set to None.
			While the test is in progress, modifying any of the writable parameters in this object except for this one MUST result in the test being terminated and the value of this parameter being set to None.
			While the test is in progress, setting this parameter to Requested (and possibly modifying other writable parameters in this object) MUST result in the test being terminated and then restarted using the current values of the test parameters.
Interface	String	-	The value MUST be the path name of the IP-layer interface over which the test is to be performed.
			The value of this parameter MUST be either a valid interface or an empty string. An attempt to set this parameter to a different value MUST be rejected as an invalid parameter value.
			If an empty string is specified, the CPE MUST use the default routing interface.
UploadURL	String	W	The URL, as defined in [RFC3986], for the CPE to Upload to. This parameter MUST be in the form of a valid HTTP [RFC2616] or FTP [RFC862] URL.
			When using FTP transport, FTP binary transfer MUST be used.
			When using HTTP transport, persistent connections MUST be used and pipelining MUST NOT be used.
			When using HTTP transport the HTTP Authentication MUST NOT be used.
DSCP	String	-	DiffServ code point for marking packets transmitted in the test.
			The default value SHOULD be zero.
EthernetPriority	Unsigned Integer	-	Ethernet priority code for marking packets transmitted in the test (if applicable).
			The default value SHOULD be zero.
TestFileLength	String	W	The size of the file (in bytes) to be uploaded to the server.
			The CPE MUST insure the appropriate number of bytes are sent.
ROMTime	dateTime	-	Request time in UTC, which MUST be specified to microsecond precision.

Name	Type	Write	Description
			For example: 2008-04-09T15:01:05.123456
			For HTTP this is the time at which the client sends the PUT command.
			For FTP this is the time at which the STOR command is sent.
BOMTime	dateTime	-	Begin of transmission time in UTC, which MUST be specified to microsecond precision.
			For example: 2008-04-09T15:01:05.123456
			For HTTP this is the time at which the first data packet is sent.
			For FTP this is the time at which the client receives the ready for transfer notification.
EOMTime	dateTime	-	End of transmission in UTC, which MUST be specified to microsecond precision.
			For example: 2008-04-09T15:01:05.123456
			For HTTP this is the time when the HTTP successful response code is received.
			For FTP this is the time when the client receives a transfer complete.
TCPOpenRequestTime	dateTime	-	Request time in UTC, which MUST be specified to microsecond precision.
			For example: 2008-04-09T15:01:05.123456
			For HTTP this is the time at which the TCP socket open (SYN) was sent for the HTTP connection.
			For FTP this is the time at which the TCP socket open (SYN) was sent for the data connection
			Note: Interval of 1 microsecond SHOULD be supported.
TCPOpenResponseTime	dateTime	-	Response time in UTC, which MUST be specified to microsecond precision.
			For example: 2008-04-09T15:01:05.123456
			For HTTP this is the Time at which the TCP ACK to the socket opening the HTTP connection was received.
			For FTP this is the Time at which the TCP ACK to the socket opening the Data connection was received.
			Note: Interval of 1 microsecond SHOULD be supported.
InternetGatewayDevice.ManagementServer.	object	-	This object contains parameters relating to the CPE's association with an ACS.
ConnectionRequestPassword	string	W	Password used to authenticate an ACS making a Connection Request to the CPE.

Name	Type	Write	Description
ConnectionRequestURL	string(256)	-	HTTP URL, as defined in {{bibref|RFC3986}}, for an ACS to make a Connection Request notification to the CPE. In the form: **http://host:port/path** The "host" portion of the URL MAY be the IP address for the management interface of the CPE in lieu of a host name.
ConnectionRequestUsername	string	W	Username used to authenticate an ACS making a Connection Request to the CPE.
ParameterKey	string(32)	W	ParameterKey provides the ACS a reliable and extensible means to track changes made by the ACS. The value of ParameterKey MUST be equal to the value of the ParameterKey argument from the most recent successful SetParameterValues, Addobject, or Deleteobject method call from the ACS. The CPE MUST set ParameterKey to the value specified in the corresponding method arguments if and only if the method completes successfully and no fault response is generated. If a method call does not complete successfully (implying that the changes requested in the method did not take effect), the value of ParameterKey MUST NOT be modified. The CPE MUST only modify the value of ParameterKey as a result of SetParameterValues, Addobject, Deleteobject, or due to a factory reset. On factory reset, the value of ParameterKey MUST be set to an empty string.
Password	string(256)	W	Password used to authenticate the CPE when making a connection to the ACS using the CPE WAN Management Protocol. his password is used only for HTTP-based authentication of the CPE. Note that on a factory reset of the CPE, the value of this parameter might be reset to its factory value. If an ACS modifies the value of this parameter, it SHOULD be prepared to accommodate the situation that the original value is restored as the result of a factory reset.
PeriodicInformEnable	boolean	W	Whether or not the CPE MUST periodically send CPE information to the ACS using the Inform method call.
PeriodicInformInterval	unsignedInt	W	The duration in seconds of the interval for which the CPE MUST attempt to connect with the ACS and call the Inform method ifPeriodicInformEnable is true.

Name	Type	Write	Description
PeriodicInformTime	dateTime	W	An absolute time reference in UTC to determine when the CPE will initiate the periodic Inform method calls. Each Inform call MUST occur at this reference time plus or minus an integer multiple of the PeriodicInformInterval. PeriodicInformTime is used only to set the "phase" of the periodic Informs. The actual value of PeriodicInformTime can be arbitrarily far into the past or future. For example, if PeriodicInformInterval is 86400 (a day) and if PeriodicInformTime is set to UTC midnight on some day (in the past, present, or future) then periodic Informs will occur every day at UTC midnight. These MUST begin on the very next midnight, even if PeriodicInformTime refers to a day in the future. The Unknown Time value defined in [Section 2.2/TR-098a2] indicates that no particular time reference is specified. That is, the CPE MAY locally choose the time reference, and needs only to adhere to the specified PeriodicInformInterval. If absolute time is not available to the CPE, its periodic Inform behavior MUST be the same as if the PeriodicInformTime parameter was set to the Unknown Time value.
RetryIntervalMultiplier	integer	W	Configures the retry interval multiplier as specified in [Section 3.2.1.1/TR-069a2]. This value is expressed in units of 0.001. Hence the values of the multiplier range between 1.000 and 65.535. A value of 2000 corresponds to the default behavior that is described in [TR-069a2]. The device MUST use a random value between CWMPRetryMinimumWaitInterval and (CWMPRetryMinimumWaitInterval * CWMPRetryIntervalMultiplier / 1000) as the first retry wait interval. Other values in the retry pattern MUST be calculated using this value as a starting point.
RetryMinimumWaitInterval	integer	W	Configures the first session retry wait interval, in seconds, as specified in [Section 3.2.1.1/TR-069a2]. A value of 5 corresponds to the default behavior that is described in [TR-069a2]. The device MUST use a random value between CWMPRetryMinimumWaitInterval and (CWMPRetryMinimumWaitInterval * CWMPRetryIntervalMultiplier / 1000) as the first retry wait interval. Other values in the

Name	Type	Write	Description
			retry pattern MUST be calculated using this value as a starting point.
UpgradesManaged	boolean	W	Indicates whether or not the ACS will manage upgrades for the CPE. If true, the CPE SHOULD NOT use other means other than the ACS to seek out available upgrades. If false, the CPE MAY use other means for this purpose. Note that an autonomous upgrade (reported via an "10 AUTONOMOUS TRANSFER COMPLETE" Inform Event code) SHOULD be regarded as a managed upgade if it is performed according to ACS-specified policy.
URL	string(256)	W	URL, as defined in RFC3986, for the CPE to connect to the ACS using the CPE WAN Management Protocol. This parameter MUST be in the form of a valid HTTP or HTTPS URL RFC2616. The "host" portion of this URL is used by the CPE for validating the ACS certificate when using SSL or TLS. Note that on a factory reset of the CPE, the value of this parameter might be reset to its factory value. If an ACS modifies the value of this parameter, it SHOULD be prepared to accommodate the situation that the original value is restored as the result of a factory reset.
Username	string(256)	W	Username used to authenticate the CPE when making a connection to the ACS using the CPE WAN Management Protocol. This username is used only for HTTP-based authentication of the CPE. Note that on a factory reset of the CPE, the value of this parameter might be reset to its factory value. If an ACS modifies the value of this parameter, it SHOULD be prepared to accommodate the situation that the original value is restored as the result of a factory reset.
ManageableDeviceNumberOfEntries	unsignedInt		Number of entries in the ManageableDevice table
InternetGatewayDevice.ManagementServer.ManageableDevice.{i}.	object	-	Each entry in this table corresponds to a distinct LAN Device that supports Device-Gateway Association according to [Annex F/TR-069a2] as indicated by the presence of the DHCP option specified in that Annex. At most one entry in this table can exist with all the same values for ManufacturerOUI, SerialNumber and ProductClass.

Name	Type	Write	Description
ManufacturerOUI	string(6)	-	Organizationally unique identifier of the Device manufacturer as provided to the Gateway by the Device. Represented as a six hexadecimal-digit value using all upper-case letters and including any leading zeros. The value MUST be a valid OUI as defined in [OUI].
SerialNumber	string(64)	-	Serial number of the Device as provided to the Gateway by the Device.
ProductClass	string(64)	-	Identifier of the class of product for which the Device's serial number applies as provided to the Gateway by the Device. If the Device does not provide a Product Class, then this parameter MUST be an empty string.
Host	string(1024)	-	Comma-separated list (maximum list length 1024) of strings. Each list item MUST be the Path Name of a Host table row. If the referenced object is deleted, the corresponding item MUST be removed from the list. This list MUST include an item for each Host table entry, whether active or inactive, that corresponds to this physical LAN Device. As such entries are added to or removed from the Host tables, the value of this parameter MUST be updated accordingly. For example: InternetGatewayDevice.LANDevice.1.Hosts.Host.1,InternetGatewayDevice.LANDevice.1.Hosts.Host.5
InternetGatewayDevice.Time.	object	-	This object contains parameters relating an NTP or SNTP time client in the CPE.
NTPServer1	string(64)	W	First NTP timeserver. Either a host name or IP address.
NTPServer2	string(64)	W	Second NTP timeserver. Either a host name or IP address.
CurrentLocalTime	dateTime	-	The current date and time in the CPE's local time zone.
LocalTimeZone	string(6)	W	The local time offset from UTC in the form: +hh:mm -hh:mm
LocalTimeZoneName	string(64)	W	Name of the local time zone (human readable string). The name SHOULD be encoded according to IEEE 1003.1 (POSIX). The following is an example value: EST+5EDT,M4.1.0/2,M10.5.0/2
Enable	boolean	W	Enables or disables the NTP or SNTP time client.
Status	string	-	Status of Time support on the CPE.

Name	Type	Write	Description
			Enumeration: ▪ Disabled ▪ Unsynchronized: indicates that the CPE's absolute time has not yet been set. ▪ Synchronized: indicates that the CPE has acquired accurate absolute time; its current time is accurate. ▪ Error_FailedToSynchronize: indicates that the CPE failed to acquire accurate absolute time; its current time is not accurate. ▪ Error (OPTIONAL): The value MAY be used by the CPE to indicate a locally defined error condition.
InternetGatewayDevice.Services.	object	-	This object contains general services information.
InternetGatewayDevice.UserInterface.	object	-	This object contains parameters relating to the user interface of the CPE.
InternetGatewayDevice.Layer3Forwarding.	object	-	This object allows the handling of the routing and forwarding configuration of the device.
ForwardNumberOfEntries	unsignedInt	-	Number of forwarding instances.
InternetGatewayDevice.Layer3Forwarding.Forwarding.{i}.	object	-	Layer 3 forwarding table. In addition to statically configured routes, this table MUST include dynamic routes learned through layer 3 routing protocols, including RIP, OSPF, DHCP, and IPCP. The CPE MAY reject attempts to delete or modify a dynamic route entry. For each incoming packet, the layer 3 forwarding decision is conceptually made as follows: Only table entries with a matching ForwardingPolicy are considered, i.e. those that either do not specify a ForwardingPolicy, or else specify a ForwardingPolicy that matches that of the incoming packet. For the remaining table entries, those for which the source address/mask matches are sorted by longest prefix, i.e. with the most specific networks first (an unspecified source address is a wild-card and always matches, with a prefix length of zero). For the remaining table entries, those for which the destination address/mask matches are sorted by longest prefix, i.e. with the most specific networks first (an unspecified destination address is a wild-

Name	Type	Write	Description
			card and always matches, with a prefix length of zero).
			The first of the remaining table entries is applied to the packet.
			At most one entry in this table (regardless of whether or not it is enabled) can exist with a given value for Alias. On creation of a new table entry, the CPE MUST choose an initial value for Alias such that the new entry does not conflict with any existing entries.
			At most one enabled entry in this table can exist with all the same values for DestIPAddress, DestSubnetMask, SourceIPAddress and SourceSubnetMask.
DestIPAddress	string	W	[IPAddress] Destination address. An empty string or a value of "0.0.0.0" indicates no destination address is specified.
			A Forwarding table entry for which DestIPAddress and DestSubnetMask are both an empty string or "0.0.0.0" is a default route.
DestSubnetMask	string	W	IPAddress Destination subnet mask. An empty string or a value of "0.0.0.0" indicates no destination subnet mask is specified.
			If a destination subnet mask is specified, the DestSubnetMask is ANDed with the destination address before comparing with the DestIPAddress. Otherwise, the full destination address is used as is.
			A Forwarding table entry for which DestIPAddress and DestSubnetMask are both an empty string or "0.0.0.0" is a default route
ForwardingMetric	unsignedInt	W	Forwarding metric. A value of -1 indicates this metric is not used.
GatewayIPAddress	string	W	IP address of the gateway.
			Only one of GatewayIPAddress and Interface SHOULD be configured for a route.
			If both are configured, GatewayIPAddress and Interface MUST be consistent with each other.
Interface	string	W	Specifies the egress interface associated with this entry. the layer 3 connection object Example: ''InternetGatewayDevice.WANDevice.1.WANConnectionDevice.2.WANPPPConnection.1'.
			Only one of GatewayIPAddress and Interface SHOULD be configured for a route.

Name	Type	Write	Description
			If both are configured, GatewayIPAddress and Interface MUST be consistent with each other. For a route that was configured by setting GatewayIPAddress but not Interface, read access to Interface MUST return the full hierarchical parameter name for the route's egress interface.
SourceIPAddress	string	W	Source address. empty or a value of "0.0.0.0" indicates no source address is specified.
SourceSubnetMask	string	W	[IPAddress] Source subnet mask. An empty string or a value of "0.0.0.0" indicates no source subnet mask is specified. If a source subnet mask is specified, the SourceSubnetMask is ANDed with the source address before comparing with the SourceIPAddress. Otherwise, the full source address is used as is.
StaticRoute	boolean	-	If true, this route is a Static route.
Status	string	-	Indicates the status of the forwarding entry. Enumeration of: ▪ Disabled ▪ Enabled ▪ Error (OPTIONAL) The Error value MAY be used by the CPE to indicate a locally defined error condition.
InternetGatewayDevice.LANDevice. {i}.	object	-	Each instance models a LAN side layer 3 IP interface. Each instance has children that correspond to the layer 2 interfaces that are connected to the Gateway's IP router via the modeled IP interface. If a LANDevice instance is deleted, the objects modeling those layer 2 interfaces that are as a result no longer connected to the Gateway's IP router will move to the InternetGatewayDevice.LANInterfaces object. If the "Layer2Bridging" object is implemented, the view that it provides of the CPE's underlying bridging configuration MUST be consistent with the view provided by any "LANDevice" and "WAN**Connection" objects. The implications of this are explained in Annex A.6.
LANEthernetInterfaceNumberOfEntries	unsignedInt	-	Number of instances of LANEthernetInterfaceConfig in this "LANDevice".

Name	Type	Write	Description
LANUSBInterfaceNumberOfEntries	unsignedInt	-	Number of instances of LANUSBInterfaceConfig in this "LANDevice".
LANWLANConfigurationNumberOfEntries	unsignedInt	-	Number of instances of WLANConfiguration in this "LANDevice".
InternetGatewayDevice.LANDevice.{i}.Hosts.	object	-	This object provides information about each of the hosts on the LAN, including those whose IP address was allocated by the CPE using DHCP as well as hosts with statically allocated IP addresses.
HostNumberOfEntries	unsignedInt	-	Number of entries in the Host table.
InternetGatewayDevice.LANDevice.{i}.Hosts.Host.{i}.	object	-	Host table.
Active	boolean	-	Whether or not the host is currently present on the LAN. The method of presence detection is a local matter to the CPE. The ability to list inactive hosts is OPTIONAL. If the CPE includes inactive hosts in this table, this variable MUST be set to false for each inactive host. The length of time an inactive host remains listed in this table is a local matter to the CPE.
AddressSource	string	-	Indicates whether the IP address of the host was allocated by the CPE using DHCP, was assigned to the host statically, or was assigned using automatic IP address allocation.
HostName	string(64)	-	The device's host name or empty if unknown.
InterfaceType	string	-	Type of physical interface through which this host is connected to the CPE.
IPAddress	string	-	Current IP Address of the host.
LeaseTimeRemaining	int	-	DHCP lease time remaining in seconds. A value of -1 indicates an infinite lease. The value MUST be 0 (zero) if the AddressSource is not DHCP.
MACAddress	string	-	MAC address of the host.
Layer2Interface	string(256)		This parameter is the full hierarchical parameter name of a particular LAN**InterfaceConfig object or a WLANConfiguration object. For example: InternetGatewayDevice.LANDevice.1.LANEthernetInterfaceConfig.2. In the case of an embedded Ethernet switch, the Layer2Interface parameter references the LANEthernetInterfaceConfig object that corresponds to the switch port the device is connected to (a

Name	Type	Write	Description
			The value of this counter MAY be reset to zero when the CPE is rebooted.
BytesReceived	unsignedInt	-	The total number of bytes received on the interface, including framing characters. The value of this counter MAY be reset to zero when the CPE is rebooted.
BytesSent	unsignedInt	-	The total number of bytes transmitted out of the interface, including framing characters. The value of this counter MAY be reset to zero when the CPE is rebooted.
DiscardPacketsReceived	unsignedInt	-	The total number of inbound packets which were chosen to be discarded even though no errors had been detected to prevent their being deliverable. One possible reason for discarding such a packet could be to free up buffer space. The value of this counter MAY be reset to zero when the CPE is rebooted.
DiscardPacketsSent	unsignedInt	-	The total number of outbound packets which were chosen to be discarded even though no errors had been detected to prevent their being transmitted. One possible reason for discarding such a packet could be to free up buffer space. The value of this counter MAY be reset to zero when the CPE is rebooted.
ErrorsReceived	unsignedInt	-	The total number of inbound packets that contained errors preventing them from being deliverable. The value of this counter MAY be reset to zero when the CPE is rebooted.
ErrorsSent	unsignedInt	-	The total number of outbound packets that could not be transmitted because of errors. The value of this counter MAY be reset to zero when the CPE is rebooted.
MulticastPacketsReceived	unsignedInt	-	The total number of received packets which were addressed to a multicast address. The value of this counter MAY be reset to zero when the CPE is rebooted.
MulticastPacketsSent	unsignedInt	-	The total number of packets requested for transmission which were addressed to a multicast address, including those that were discarded or not sent. The value of this counter MAY be reset to zero when the CPE is rebooted.
PacketsReceived	unsignedInt	-	The total number of packets which were received on this interface.

Name	Type	Write	Description
			The value of this counter MAY be reset to zero when the CPE is rebooted.
PacketsSent	unsignedInt	-	The total number of packets transmitted out of the interface. The value of this counter MAY be reset to zero when the CPE is rebooted.
UnicastPacketsReceived	unsignedInt	-	The total number of received packets which were not addressed to a multicast or broadcast address. The value of this counter MAY be reset to zero when the CPE is rebooted.
UnicastPacketsSent	unsignedInt	-	The total number of packets requested for transmission which were not addressed to a multicast or broadcast address, including those that were discarded or not sent. The value of this counter MAY be reset to zero when the CPE is rebooted.
InternetGatewayDevice.LANDevice.{i}.LANHostConfigManagement.	object	-	This object enables reporting of LAN-related device information and setting and configuring LAN IP addressing. The DHCP parameters in this object define the behavior of the default DHCP server, i.e. the behavior for DHCP requests that do not match any of the DHCP conditional serving pool entries.
DHCPLeaseTime	int	W	Specifies the lease time in seconds of client assigned addresses. A value of -1 indicates an infinite lease.
DHCPRelay	string	-	Indicates if the DHCP server performs the role of a server (false) or a relay (true) on the LAN interface. This parameter is DEPRECATED because the functionality that it describes is not well-defined. The CPE MAY set it to the value that it thinks most appropriate, based on its configuration.
DHCPServerConfigurable	boolean	W	Enables the configuration of the DHCP server on the LAN interface. If this variable is set to false, the CPE SHOULD restore its default DHCP server settings.
DHCPServerEnable	boolean	W	Enables or disables the DHCP server on the LAN interface.
DNSServers	string	W	DNS servers offered to DHCP clients. Support for more than three DNS Servers is OPTIONAL.
DomainName	string	W	Sets the domain name to provide to clients on the LAN interface.
IPInterfaceNumberOfEntries	unsignedInt	-	The number of entries in the IPInterface table.

Name	Type	Write	Description
IPRouters	string	W	IP addresses of routers on this subnet. Also known as default gateway. Support for more than one Router address is OPTIONAL.
MaxAddress	string	W	Specifies last address in the pool to be assigned by the DHCP server on the LAN interface. This parameter MUST have a valid value before the DHCP server can be enabled.
MinAddress	string	W	Specifies first address in the pool to be assigned by the DHCP server on the LAN interface. This parameter MUST have a valid value before the DHCP server can be enabled.
SubnetMask	string	W	Specifies the client's network subnet mask. This parameter MUST have a valid value before the DHCP server can be enabled.
InternetGatewayDevice.LANDevice. {i}.LANHostConfigManagement.IPI nterface.{i}.	object	-	IP address table with each object representing an IP address on the LANDevice IP interface. Support for more than one interface instance is OPTIONAL.
Enable	boolean	W	Enables or disables this entry. On creation, an entry is disabled by default.
IPInterfaceAddressingType	string	W	Represents the addressing method used to assign the LAN-side IP address of the CPE on this interface.
IPInterfaceIPAddress	string	W	IP address of the LAN-side interface of the CPE.
IPInterfaceSubnetMask	string	W	Subnet mask of the LAN-side interface of the IGD.
InternetGatewayDevice.LANDevice. {i}.WLANConfiguration.{i}.	object	-	This object models an 802.11 LAN connection on a CPE device. This object MUST be implemented for CPE that contain an 802.11 interface on the LAN side.
AutoChannelEnable	boolean	W	Enable or disable automatic channel selection. Set to false to disable the automatic channel selection procedure, in which case the currently selected channel remains selected. Set to true to enable the automatic channel selection procedure. This procedure MUST automatically select the channel, and MAY also change it subsequently. AutoChannelEnable MUST automatically change to false whenever the channel is

Name	Type	Write	Description
			manually selected, i.e. whenever the Channel parameter is written. Whenever AutoChannelEnable is true, the value of the Channel parameter MUST be the channel selected by the automatic channel selection procedure.
AutoRateFallBackEnabled	boolean	-	Indicates whether the access point can automatically reduce the data rate in the event of undue noise or contention.
BasicAuthenticationMode	string	W	Authentication modes that are available when basic 802.11 is enabled. Enumeration of: ▪ None (Open authentication) ▪ EAPAuthentication (OPTIONAL) ▪ SharedAuthentication (OPTIONAL) If this WLANConfiguration instance does not support basic 802.11 then this parameter MUST NOT be present in this instance of the WLANConfiguration object.
BasicEncryptionModes	string	W	Encryption modes that are available when basic 802.11 is enabled. WEPEncryption implies that all wireless clients can use WEP for data encryption. Enumeration of: ▪ None ▪ WEPEncryption If this WLANConfiguration instance does not support basic 802.11 then this parameter MUST NOT be present in this instance of the WLANConfiguration object
BeaconType	string	W	The capabilities that are currently enabled on the access point (and that are announced via beacons if BeaconAdvertisementEnabled is true). Write access to this parameter enables and disables such capabilities. An attempt to set this parameter to one of the REQUIRED (mandatory) values MAY be rejected if (and only if) the requested capability is not available on this WLANConfiguration instance but is available on another WLANConfiguration instance within this Internet Gateway Device. For example, only basic 802.11 might be supported by one virtual AP, and only WPA might be supported by another virtual AP. A value of None means that no capabilities are currently enabled on the access point and that no stations will be able to associate with it. Enumeration of: ▪ None ▪ Basic

Name	Type	Write	Description
			▪ WPA ▪ 11i (OPTIONAL) ▪ BasicandWPA (OBSOLETED) ▪ Basicand11i (OBSOLETED) ▪ WPAand11i (OPTIONAL) ▪ BasicandWPAand11i (OBSOLETED) 11i SHOULD be taken to refer to both the 802.11i specification and to the WPA2 specification (any WPA2-certified device will implement all mandatory parts of the 802.11i standard). The OBSOLETED values are those for Basic + WPA/WPA2 mixed modes, which are not permitted by the WPA specifications.
BSSID	string	-	The MAC address of the interface.
Channel	unsignedInt	W	The current radio channel used by the connection. To request automatic channel selection, set AutoChannelEnable to true. Whenever AutoChannelEnable is true, the value of the Channel parameter MUST be the channel selected by the automatic channel selection procedure.
Enable	boolean	W	Enables or disables this interface. When there are multiple "WLANConfiguration" instances, e.g. each instance supports a different 802.11 standard or has a different security configuration, this parameter can be used to control which of the instances are currently enabled.
KeyPassphrase	string	-	A passphrase from which the WEP keys are to be generated. This parameter is the same as the parameter "InternetGatewayDevice.LANDevice.{i}.WLANConfiguration.{i}.PreSharedKey.1.KeyPassphrase" for the same instance of "WLANConfiguration". When either parameter is changed, the value of the other is changed as well. IfKeyPassphrase is written, all four WEP keys are immediately generated. The ACS SHOULD NOT set the passphrase and also set the WEP keys directly (the result of doing this is undefined). This MUST either be a valid key length divided by 8, in which case each byte contributes 8 bits to the key, or else MUST consist of Hex digits and be a valid key

Name	Type	Write	Description
			length divided by 4, in which case each byte contributes 4 bits to the key. Note: If a passphrase is used, all four WEP keys will be the same.
MaxBitRate	string	-	The maximum upstream and downstream bit rate available to this connection in Mbps. Either "Auto", or the largest of the param|OperationalDataTransmitRates values.
Name	string	-	The name of this layer 2 interface, chosen by the vendor, e.g. "wlan0".
RadioEnabled	boolean	W	Indicates whether or not the access point radio is enabled.
SSID	string(32)	W	The current service set identifier in use by the connection. The SSID is an identifier that is attached to packets sent over the wireless LAN that functions as a "password" for joining a particular radio network (BSS). Note: If an access point wishes to be known by more than one SSID, it MUST provide a "WLANConfiguration" instance for each SSID.
SSIDAdvertisementEnabled	boolean	W	Indicates whether or not beacons include the SSID name. This parameter has an effect only if BeaconAdvertisementEnabled is true.
Standard	string	W	Indicates which IEEE 802.11 standard this WLANConfiguration instance is configured for. Enumeration of: ▪ a ▪ b ▪ g (b and g clients supported) ▪ g-only (only g clients supported) ▪ n Where each value indicates support for only the indicated standard. If the device is configured simultaneously for more than one standard, a separate WLANConfiguration instance MUST be used for each supported standard.
Status	string	-	Indicates the status of this interface. Enumeration of: ▪ Up ▪ Error (OPTIONAL) ▪ Disabled The Error value MAY be used by the CPE to indicate a locally defined error condition.
TotalAssociations	unsignedInt	-	The number of devices currently associated with the access point. This corresponds to

Name	Type	Write	Description
			the number of entries in the AssociatedDevice table.
TotalBytesReceived	unsignedInt	-	The total number of bytes received on the interface, including framing characters. The value of this counter MAY be reset to zero when the CPE is rebooted.
TotalBytesSent	unsignedInt	-	The total number of bytes transmitted out of the interface, including framing characters. The value of this counter MAY be reset to zero when the CPE is rebooted.
TotalPacketsReceived	unsignedInt	-	The total number of packets which were received on this interface. The value of this counter MAY be reset to zero when the CPE is rebooted.
TotalPacketsSent	unsignedInt	-	The total number of packets transmitted out of the interface. The value of this counter MAY be reset to zero when the CPE is rebooted.
TransmitPower	unsignedInt	W	Indicates the current transmit power level as a percentage of full power. The value MUST be one of the values reported by the param|TransmitPowerSupported parameter.
TransmitPowerSupported	string	-	Supported transmit power levels as percentages of full power. For example, "0,25,50,75,100".
WEPEncryptionLevel	string	W	A passphrase from which the WEP keys are to be generated. This parameter is the same as the parameter InternetGatewayDevice.LANDevice.{i}.WLANConfiguration.{i}.PreSharedKey.1.KeyPassphrase for the same instance of WLANConfiguration. When either parameter is changed, the value of the other is changed as well. If KeyPassphrase is written, all four WEP keys are immediately generated. The ACS SHOULD NOT set the passphrase and also set the WEP keys directly (the result of doing this is undefined). This MUST either be a valid key length divided by 8, in which case each byte contributes 8 bits to the key, or else MUST consist of Hex digits and be a valid key length divided by 4, in which case each byte contributes 4 bits to the key. Note: If a passphrase is used, all four WEP keys will be the same.

Name	Type	Write	Description
			When read, this parameter returns an empty string, regardless of the actual value.
WEPKeyIndex	unsignedInt	W	The index of the default WEP key.
WPAAuthenticationMode	string	W	Authentication modes that are available when WPA is enabled. Enumeration of: ▪ PSKAuthentication ▪ EAPAuthentication (OPTIONAL) If this WLANConfiguration instance does not support WPA then this parameter MUST NOT be present in this instance of the WLANConfiguration object.
WPAEncryptionModes	string	W	Encryption modes that are available when WPA is enabled. Enumeration of: ▪ WEPEncryption (DEPRECATED) ▪ TKIPEncryption ▪ WEPandTKIPEncryption (DEPRECATED) ▪ AESEncryption (OPTIONAL) ▪ WEPandAESEncryption (DEPRECATED) ▪ TKIPandAESEncryption (OPTIONAL) ▪ WEPandTKIPandAESEncryption (DEPRECATED) If this WLANConfiguration instance does not support WPA then this parameter MUST NOT be present in this instance of the WLANConfiguration object. The DEPRECATED values are those that combine WEP with TKIP and/or AES, which is not permitted by the WPA specifications.
IEEE11iAuthenticationMode	string	W	Authentication modes that are available when 802.11i is enabled: ▪ PSKAuthentication ▪ EAPAuthentication (OPTIONAL) ▪ EAPandPSKAuthentication (OPTIONAL) If this WLANConfiguration instance does not support 802.11i, this parameter MUST NOT be present in this instance of the WLANConfiguration object. IEEE11i SHOULD be taken to refer to both the 802.11i specification and to the WPA2 specification (any WPA2-certified device will implement all mandatory parts of the 802.11i standard).
IEEE11iEncryptionMode	string	W	Encryption modes that are available when 802.11i is enabled: ▪ WEPEncryption (DEPRECATED) ▪ TKIPEncryption (OPTIONAL)

Name	Type	Write	Description
			▪ WEPandTKIPEncryption (DEPRECATED) ▪ AESEncryption ▪ WEPandAESEncryption (DEPRECATED) ▪ TKIPandAESEncryption (OPTIONAL) ▪ WEPandTKIPandAESEncryption (DEPRECATED) If this WLANConfiguration instance does not support 802.11i then this parameter MUST NOT be present in this instance of the WLANConfiguration object. IEEE11i SHOULD be taken to refer to both the 802.11i specification and to the WPA2 specification (any WPA2-certified device will implement all mandatory parts of the 802.11i standard). The DEPRECATED values are those that combine WEP with TKIP and/or AES, which is not permitted by the WPA2 specifications.
InternetGatewayDevice.LANDevice. {i}.WLANConfiguration.{i}.Associat edDevice.{i}.	object	-	A table of the devices currently associated with the access point. The size of this table is given by InternetGatewayDevice.LANDevice.{i}.WLA NConfiguration.{i}.TotalAssociations. This object MUST be implemented for CPE that contain an 802.11 interface on the LAN side.
AssociatedDeviceAuthenticationState	boolean	-	Whether an associated device has authenticated (true) or not (false).
AssociatedDeviceIPAddress	string(64)	-	The IP address or DNS name of an associated device.
AssociatedDeviceMACAddress	string	-	The MAC address of an associated device.
InternetGatewayDevice.LANDevice. {i}.WLANConfiguration.{i}.PreShare dKey.{i}.	object	-	This is a table of preshared keys. The size of this table is fixed with exactly 10 entries (with instance numbers 1 through 10). This object MUST be implemented for CPE that contain an 802.11 interface on the LAN side.
AssociatedDeviceMACAddress	string	W	The MAC address associated with a preshared key, or empty if no MAC address is associated with the key.
KeyPassphrase	string	W	A passphrase from which the PSK is to be generated. The first table entry is the same as the parameter InternetGatewayDevice.LANDevice.{i}.WLA NConfiguration.{i}.KeyPassphrase for the same instance of WLANConfiguration.

Name	Type	Write	Description
			When either parameter is changed, the value of the other is changed as well.
			If KeyPassphrase is written, the PSK is immediately generated. The ACS SHOULD NOT set the passphrase and also set the PSK directly (the result of doing this is undefined).
			The key is generated as specified by WPA, which uses PBKDF2 from PKCS #5: Password-based Cryptography Specification Version 2.0 (RFC 2898 RFC2898).
PreSharedKey	string	W	A literal WPA PSK expressed as a hexadecimal string.
			The first table entry contains the defaultPreSharedKey} ("InternetGatewayDevice.LANDevice.{i}.WLANConfiguration.{i}.PreSharedKey.1.PreSharedKey").
			If param|KeyPassphrase is written, the PSK is immediately generated. The ACS SHOULD NOT set the passphrase and also set the PSK directly (the result of doing this is undefined).
InternetGatewayDevice.LANDevice.{i}.WLANConfiguration.{i}.WEPKey.{i}.	object	-	This is a table of WEP keys. The size of this table is fixed with exactly 4 entries (with instance numbers 1 through 4). This object MUST be implemented for CPE that contain an 802.11 interface on the LAN side.
WEPKey	string(128)	W	A WEP key expressed as a hexadecimal string.
			The WEP encryption level for a given key is inferred from the key length, e.g. 10 characters for 40-bit encryption, or 26 characters for 104-bit encryption (keys do not all have to be of the same length, although they will be if the CPE uses InternetGatewayDevice.LANDevice.{i}.WLANConfiguration.{i}.KeyPassphrase to generate them).
			If InternetGatewayDevice.LANDevice.{i}.WLANConfiguration.{i}.KeyPassphrase is written, all four WEP keys are immediately generated. The ACS SHOULD NOT set the passphrase and also set the WEP keys directly (the result of doing this is undefined).
InternetGatewayDevice.WANDevice.{i}.	Object	-	Each instance contains all objects associated with a particular physical WAN interfac
WANConnectionNumberOfEntries	unsignedInt	-	Number of instances of WANConnectionDevice in this WANDevice.

Name	Type	Write	Description
InternetGatewayDevice.WANDevice.{i}.WANCommonInterfaceConfig.	object	-	This object models WAN interface properties common across all connection instances.
EnabledForInternet	boolean	W	Used to enable or disable access to and from the Internet across all connection instances.
Layer1DownstreamMaxBitRate	unsignedInt	-	Specifies the maximum downstream theoretical bit rate for the WAN device in bits per second. This describes the maximum possible rate given the type of interface assuming the best-case operating environment, regardless of the current operating rate. For example, if the physical interface is 100BaseT, this value would be 100000000, regardless of the current operating rate.
Layer1UpstreamMaxBitRate	unsignedInt	-	Specifies the maximum upstream theoretical bit rate for the WAN device in bits per second. This describes the maximum possible rate given the type of interface assuming the best-case operating environment, regardless of the current operating rate. For example, if the physical interface is 100BaseT, this value would be 100000000, regardless of the current operating rate.
PhysicalLinkStatus	string	-	Indicates the state of the physical connection (link) from WANDevice to a connected entity.
TotalBytesReceived	unsignedInt	-	The cumulative counter for total number of bytes received downstream across all connection service instances on the WAN device.
TotalBytesSent	unsignedInt	-	The cumulative counter for total number of bytes sent upstream across all connection service instances on the WAN device.
TotalPacketsReceived	unsignedInt	-	The cumulative counter for total number of packets (IP or PPP) received downstream across all connection service instances on the WAN device.
TotalPacketsSent	unsignedInt	-	The cumulative counter for total number of packets (IP or PPP) sent upstream across all connection service instances on the WAN device.
WANAccessType	string	-	Specifies the WAN access (modem) type.
InternetGatewayDevice.WANDevice.{i}.WANConnectionDevice.{i}.	object	-	Each instance contains objects associated with a given WAN link. In the case of DSL, each instance corresponds to either a single ATM VC or a PTM Ethernet link. On creation of a "WANConnectionDevice"

Name	Type	Write	Description
			instance, there are initially no connection objects contained within. In the case of Ethernet (interface or link), only one ''WANConnectionDevice'' instance is supported.
WANIPConnectionNumberOfEntries	unsignedInt	-	Number of instances of WANIPConnection in this ''WANConnectionDevice''.
WANPPPConnectionNumberOfEntries	unsignedInt	-	Number of instances of WANPPPConnection in this ''WANConnectionDevice''.
InternetGatewayDevice.WANDevice.{i}.WANConnectionDevice.{i}.WANDSLLinkConfig.	object	-	This object models the ATM layer properties specific to a single physical connection of a DSL modem used for Internet access on a CPE. This object is intended for a CPE with a DSL modem WAN interface, and is exclusive of any other ''WAN*LinkConfig'' object within a given ''WANConnectionDevice'' instance.
DestinationAddress	string(256)	W	Destination address of this link. One of: ▪ * ''PVC: VPI/VCI'' ▪ * ''SVC: ATM connection name'' ▪ * ''SVC: ATM address'' The "PVC:" or "SVC:" prefix is part of the parameter value and MUST be followed by 0 or 1 space characters. For example, possible values for this parameter are "PVC:8/23" or "PVC: 0/35".
Enable	boolean	W	Enables or disables the link. On creation of a ''WANConnectionDevice'', this object is disabled by default.
LinkStatus	string	W	Status of the link.
LinkType	string	-	Indicates the type of DSL connection and refers to the complete stack of protocol used for this connection. Enumeration of: ▪ EoA (RFC 2684 [RFC2684] bridged Ethernet over ATM) ▪ IPoA (RFC 2684 [RFC2684] routed IP over ATM) ▪ PPPoA (RFC 2364 [RFC2634] PPP over ATM) ▪ PPPoE (RFC 2516 [RFC2516] PPP over Ethernet on RFC 2684 [RFC2684] bridged Ethernet over ATM, DEPRECATED) ▪ CIP (RFC 2225 [RFC2225] Classical IP over ATM) ▪ Unconfigured The value PPPoE has always been DEPRECATED and EoA SHOULD be used instead (see [Annex B/TR-098a2]). The ACS MUST NOT set this parameter to

Name	Type	Write	Description
			PPPoE and the CPE MUST reject attempts to do so.
InternetGatewayDevice.WANDevice.{i}.WANConnectionDevice.{i}.WANEthernetLinkConfig.	object	-	This object models the Ethernet link layer properties specific to a single physical connection used for Internet access on a CPE. This object is intended for a CPE with an Ethernet WAN interface, and is exclusive of any other "WAN*LinkConfig" object within a given "WANConnectionDevice" instance. Note that this object is not related to the Ethernet protocol layer sometimes used in associated with a DSL connection.
EthernetLinkStatus	string	-	Status of the Ethernet link.
InternetGatewayDevice.WANDevice.{i}.WANConnectionDevice.{i}.WANIPConnection.{i}.	object	-	This object enables configuration of IP connections on the WAN interface of a CPE. If the "Layer2Bridging" object is implemented, the view that it provides of the CPE's underlying bridging configuration MUST be consistent with the view provided by any "LANDevice" and "WAN**Connection" objects. The implications of this are explained in TR-098a2-Annex A.6.
AddressingType	string	-	The method used to assign an address to the WAN side interface of the CPE for this connection.
AutoDisconnectTime	unsignedInt	-	The time in seconds since the establishment of the connection after which connection termination is automatically initiated by the CPE. This occurs irrespective of whether the connection is being used or not. A value of 0 (zero) indicates that the connection is not to be shut down automatically.
ConnectionStatus	boolean	-	Current status of the connection. Enumeration of: • Unconfigured • Connecting • Connected • PendingDisconnect • Disconneting (DEPRECATED) • Disconnecting • Disconnected The value Disconneting is DEPRECATED because it is a typo. The ACS MUST treat Disconneting and Disconnecting the same.
ConnectionTrigger	string	W	Trigger used to establish the IP connection. Enumeration of: • OnDemand

Name	Type	Write	Description
			▪ AlwaysOn ▪ Manual The above values are defined as follows: ▪ OnDemand: If this IP connection is disconnected for any reason, it is to remain disconnected until the CPE has one or more packets to communicate over this connection, at which time the CPE automatically attempts to reestablish the connection. ▪ AlwaysOn: If this IP connection is disconnected for any reason, the CPE automatically attempts to reestablish the connection (and continues to attempt to reestablish the connection as long it remains disconnected). ▪ Manual: If this IP connection is disconnected for any reason, it is to remain disconnected until the user of the CPE explicitly instructs the CPE to reestablish the connection. Note that the reason for an IP connection becoming disconnected to begin with might be either external to the CPE, such as non-renewal of a DHCP lease or momentary disconnection of the physical interface, or internal to the CPE, such as use of the IdleDisconnectTime and/or AutoDisconnectTime parameters in this object. Note also that the means by which a CPE would keep an IP connection disconnected (while waiting for the designated trigger) if it is otherwise physically connected and has an IP address is a local matter specific to the implementation of the CPE.
DefaultGateway	string	-	The IP address of the default gateway for this connection. This parameter is configurable only if the AddressingType is Static.
DNSEnabled	boolean	-	Whether or not the device SHOULD attempt to query a DNS server across this connection.
DNSServers	string(64)	-	DNS server IP addresses for this connection. Support for more than three DNS Servers is OPTIONAL.
Enable	boolean	W	Enables or disables the connection instance. On creation of a "WANIPConnection" instance, it is initially disabled.
ExternalIPAddress	string	-	This is the external IP address used by NAT for this connection. This parameter is

Name	Type	Write	Description
			configurable only if the AddressingType is Static. If Active Notification is enabled, the CPE MUST initiate an Inform whenever either the value of this parameter changes or the default WAN connection changes to a different connection.
IdleDisconnectTime	unsignedInt	-	The time in seconds that if the connection remains idle, the CPE automatically terminates the connection. A value of 0 (zero) indicates that the connection is not to be shut down automatically.
MACAddress	string	-	The physical address of the "WANIPConnection" if applicable. Configurable only if MACAddressOverride is present and true.
MaxMTUSize	unsignedInt	W	The maximum allowed size of an Ethernet frame from LAN-side devices.
Name	string(256)	W	User-readable name of this connection.
NATEnabled	boolean	W	Indicates if Network Address Translation (NAT) is enabled for this connection.
PortMappingNumberOfEntries	unsignedInt	-	Total number of port mapping entries.
PossibleConnectionTypes	string	-	Types of connections possible for this connection instance.
SubnetMask	string	-	Subnet mask of the WAN interface. This parameter is configurable only if the AddressingType isStatic.
Uptime	unsignedInt	-	The time in seconds that this connection has been up.
WarnDisconnectDelay	unsignedInt	-	Time in seconds the Status remains in the pending disconnect state before transitioning to disconnecting state to drop the connection.
InternetGatewayDevice.WANDevice.{i}.WANConnectionDevice.{i}.WANIPConnection.{i}.PortMapping.{i}.	object	-	Port mapping table. This table MUST contain all NAT port mappings associated with this connection, including static and dynamic port mappings programmatically created via local control protocol, such as UPnP. This table MUST NOT contain dynamic NAT binding entries associated with the normal operation of NAT. At most one entry in an instance of this table can exist with all of the same values for RemoteHost, ExternalPort, and PortMappingProtocol. If the ACS attempts to set the parameters of an existing entry

Name	Type	Write	Description
			such that this requirement would be violated, the CPE MUST reject the request. In this case, the SetParameterValues response MUST include a SetParameterValuesFault element for each parameter in the corresponding request whose modification would have resulted in such a violation. On creation of a new table entry, the CPE MUST choose default values for ExternalPort and PortMappingProtocol such that the new entry does not conflict with any existing entry.
			At most one entry in this table (regardless of whether or not it is enabled) can exist with a given value for Alias. On creation of a new table entry, the CPE MUST choose an initial value for Alias such that the new entry does not conflict with any existing entries.
			At most one enabled entry in this table can exist with all the same values for RemoteHost, ExternalPort and PortMappingProtocol.
ExternalPort	integer	W	The external port (or the first port of a range of external ports) that the NAT gateway would listen on for connection requests to a corresponding InternalPort. Inbound packets to this external port on the WAN interface SHOULD be forwarded to InternalClient on the InternalPort.
			A value of zero (0) represents a "wildcard". If this value is a wildcard, connection requests on all external ports (that are not otherwise mapped) will be forwarded to InternalClient. In the wildcard case, the value(s) of InternalPort on InternalClient are ignored.
			When wildcard values are used for RemoteHost and/or ExternalPort, the following precedence order applies (with the highest precedence listed first): ▪ Explicit RemoteHost, explicit ExternalPort ▪ Explicit RemoteHost, wildcard ExternalPort ▪ Wildcard RemoteHost, explicit ExternalPort ▪ Wildcard RemoteHost, wildcard ExternalPort
			If an incoming packet matches the criteria associated with more than one entry in this table, the CPE MUST apply the port mapping associated with the highest precedence entry.

Name	Type	Write	Description
			At most one entry in this table can exist with all of the same values for RemoteHost, ExternalPort, and PortMappingProtocol.
ExternalPortEndRange	integer	W	Indicates the last port of the external port range that starts with ExternalPort. If an external port range is specified, then the behavior described for ExternalPort applies to all ports within the range. A value of zero (0) indicates that no external port range is specified, i.e. that the range consists only of ExternalPort. If ExternalPort is zero (wildcard), the value of this parameter MUST be ignored. If specified, the value of this parameter MUST be greater than or equal to the value of ExternalPort.
InternalClient	string	W	The IP address or DNS host name of an internal client (on the LAN). Support for an IP address is mandatory. If InternalClient is specified as an IP address and the LAN device's IP address subsequently changes, the port mapping MUST remain associated with the original IP address. Support for DNS host names is OPTIONAL. If InternalClient is specified as a DNS host name and the LAN device's IP address subsequently changes, the port mapping MUST remain associated with this LAN device. In this case, it is the responsibility of the CPE to maintain the name-to-address mapping in the event of IP address changes. This can be accomplished, for example, by assigning the DNS host name via use of DHCP option 12 (Host Name) or option 81 (FQDN). Note that the ACS can learn the host name associated with a given LAN device via the Hosts table (InternetGatewayDevice.LANDevice.{i}.Hosts). Read access to this parameter MUST always return the exact value that was last set by the ACS. For example, if the internal client is set to a DNS host name, it MUST read back as a DNS host name and not as an IP address. An empty string indicates an unconfigured InternalClient. If this parameter is unconfigured, this port mapping MUST NOT be operational. It MUST be possible to set the InternalClient to the broadcast IP address 255.255.255.255 for UDP mappings. This is

Name	Type	Write	Description
			to enable multiple NAT clients to use the same well-known port simultaneously.
InternalPort	integer	W	The port on InternalClient that the gateway SHOULD forward connection requests to. A value of zero (0) is not allowed.
PortMappingDescription	string	W	User-readable description of this port mapping.
PortMappingEnabled	boolean	W	Enables or disables the port mapping instance. On creation, an entry is disabled by default.
PortMappingLeaseDuration	integer	W	Determines the time to live, in seconds, of a port-mapping lease, where "time to live" means the number of seconds before the port mapping expires. A value of 0 means the port mapping is static. Support for dynamic (non-static) port mappings is OPTIONAL. That is, the only value for PortMappingLeaseDuration that MUST be supported is 0. For a dynamic (non-static) port mapping, when this parameter is read, the value represents the time remaining on the port-mapping lease. That is, for a dynamic port mapping, the value counts down toward 0. When a dynamic port-mapping lease expires, the CPE MUST automatically terminate that port mapping, and MUST automatically delete the corresponding PortMapping table entry.
PortMappingProtocol	string	W	The protocol of the port mapping. Enumeration of: • TCP • UDP At most one entry in this table can exist with all of the same values for RemoteHost, ExternalPort, and PortMappingProtocol.
RemoteHost	string	W	This parameter is the IP address of the source of inbound packets. An empty string indicates a "wildcard" (this will be a wildcard in most cases). CPE are REQUIRED only to support wildcards. When RemoteHost is a wildcard, all traffic sent to the ExternalPort on the WAN interface of the gateway is forwarded to the InternalClient on the InternalPort. When RemoteHost is specified as one external IP address, the NAT will only forward inbound packets from this RemoteHost to the InternalClient, all other packets will be dropped. If a CPE supports non-wildcard values for RemoteHost, it MAY additionally support

Name	Type	Write	Description
			the ability to have more than one port mapping with the same ExternalPort and PortMappingProtocol, but with differing values of RemoteHost.
			When wildcard values are used for RemoteHost and/or ExternalPort, the following precedence order applies (with the highest precedence listed first):
			▪ Explicit RemoteHost, explicit ExternalPort
			▪ Explicit RemoteHost, wildcard ExternalPort
			▪ Wildcard RemoteHost, explicit ExternalPort
			▪ Wildcard RemoteHost, wildcard ExternalPort
			If an incoming packet matches the criteria associated with more than one entry in this table, the CPE MUST apply the port mapping associated with the highest precedence entry.
			At most one entry in this table can exist with all of the same values for RemoteHost, ExternalPort, and PortMappingProtocol.
InternetGatewayDevice.WANDevice.{i}.WANConnectionDevice.{i}.WANIPConnection.{i}.Stats.	object	-	This object contains statistics for all connections within the same "WANConnectionDevice" that share a common MAC address. The contents of this object SHOULD be identical for each such connection.
			This object is intended only for "WANConnectionDevice"s that can support an Ethernet-layer on this interface (e.g., "PPPoE", "IPoE").
EthernetBytesReceived	unsignedInt	-	The total number of bytes received, including framing characters, over all connections within the same "WANConnectionDevice" that share a common MAC address.
			The value of this counter MAY be reset to zero when the CPE is rebooted.
EthernetBytesSent	unsignedInt	-	The total number of bytes transmitted, including framing characters, over all connections within the same "WANConnectionDevice" that share a common MAC address.
			The value of this counter MAY be reset to zero when the CPE is rebooted.
EthernetDiscardPacketsReceived	unsignedInt	-	The total number of inbound packets which were chosen to be discarded even though no errors had been detected to prevent their being deliverable, over all connections

Name	Type	Write	Description
			within the same "WANConnectionDevice" that share a common MAC address. One possible reason for discarding such a packet could be to free up buffer space. The value of this counter MAY be reset to zero when the CPE is rebooted.
EthernetDiscardPacketsSent	unsignedInt	-	The total number of outbound packets which were chosen to be discarded even though no errors had been detected to prevent their being transmitted, over all connections within the same "WANConnectionDevice" that share a common MAC address. One possible reason for discarding such a packet could be to free up buffer space. The value of this counter MAY be reset to zero when the CPE is rebooted.
EthernetErrorsReceived	unsignedInt	-	The total number of inbound packets that contained errors preventing them from being deliverable, over all connections within the same "WANConnectionDevice" that share a common MAC address. The value of this counter MAY be reset to zero when the CPE is rebooted.
EthernetErrorsSent	unsignedInt	-	The total number of outbound packets that could not be transmitted because of errors, over all connections within the same "WANConnectionDevice" that share a common MAC address. The value of this counter MAY be reset to zero when the CPE is rebooted.
EthernetPacketsReceived	unsignedInt	-	The total number of packets which were received over all connections within the same "WANConnectionDevice" that share a common MAC address. The value of this counter MAY be reset to zero when the CPE is rebooted.
EthernetPacketsSent	unsignedInt	-	The total number of packets transmitted over all connections within the same "WANConnectionDevice" that share a common MAC address. The value of this counter MAY be reset to zero when the CPE is rebooted.
InternetGatewayDevice.WANDevice.{i}.WANConnectionDevice.{i}.WANPPPConnection.{i}.	object	-	This object enables configuration of PPP connections on the WAN interface of a CPE. If the "Layer2Bridging" object is implemented, the view that it provides of the CPE's underlying bridging configuration MUST be consistent with the view provided by any "LANDevice" and "WAN**Connection" objects. The

Name	Type	Write	Description
			implications of this are explained in TR-098a2-Annex A.
ConnectionStatus	string	-	Current status of the connection.
ConnectionTrigger	string	W	Trigger used to establish the PPP connection. Enumeration of: ▪ OnDemand ▪ AlwaysOn ▪ Manual The above values are defined as follows: ▪ OnDemand: If this PPP connection is disconnected for any reason, it is to remain disconnected until the CPE has one or more packets to communicate over this connection, at which time the CPE automatically attempts to reestablish the connection. ▪ AlwaysOn: If this PPP connection is disconnected for any reason, the CPE automatically attempts to reestablish the connection (and continues to attempt to reestablish the connection as long it remains disconnected). ▪ Manual: If this PPP connection is disconnected for any reason, it is to remain disconnected until the user of the CPE explicitly instructs the CPE to reestablish the connection. Note that the reason for a PPP connection becoming disconnected to begin with might be either external to the CPE, such as termination by the BRAS or momentary disconnection of the physical interface, or internal to the CPE, such as use of the IdleDisconnectTime and/or AutoDisconnectTime parameters in this object.
ConnectionType	string	-	Specifies the connection type of the connection instance.
DNSEnabled	boolean	-	Whether or not the device SHOULD attempt to query a DNS server across this connection.
DNSServers	string(64)	-	DNS server IP addresses for this connection. Support for more than three DNS Servers is OPTIONAL.
Enable	boolean	W	Enables or disables the connection instance. On creation of a "WANPPPConnection" instance, it is initially disabled.
ExternalIPAddress	string	-	This is the external IP address used by NAT for this connection.

Name	Type	Write	Description
			If Active Notification is enabled, the CPE MUST initiate an Inform whenever either the value of this parameter changes or the default WAN connection changes to a different connection.
MACAddress	string	-	The physical address of the "WANPPPConnection" if applicable. Configurable only if MACAddressOverride is present and true. If TransportType is PPPoA-TransportType, the value of this parameter is irrelevant and MUST be empty.
Name	string(256)	W	User-readable name of this connection.
NATEnabled	boolean	W	Indicates if Network Address Translation (NAT) is enabled for this connection.
Password	string(64)	W	Password to be usef for authentication.
PortMappingNumberOfEntries	unsignedInt	-	Total number of port mapping entries.
PossibleConnectionTypes	string	-	Types of connections possible for this connection instance.
PPPLCPEcho	unsignedInt	W	PPP LCP Echo period in seconds.
PPPLCPEchoRetry	unsignedInt	W	Number of PPP LCP Echo retries within an echo period.
PPPoEACName	string(256)	W	PPPoE Access Concentrator.
PPPoEServiceName	string(256)	W	PPPoE Service Name.
TransportType	string	-	PPP transport type of the connection.
Uptime	unsignedInt	-	The time in seconds that this connection has been up.
Username	string(64)	W	Username to be used for authentication.
InternetGatewayDevice.WANDevice.{i}.WANConnectionDevice.{i}.WANPPPConnection.{i}.PortMapping.{i}.	object	-	Port mapping table. This table MUST contain all NAT port mappings associated with this connection, including static and dynamic port mappings programmatically created via local control protocol, such as UPnP. This table MUST NOT contain dynamic NAT binding entries associated with the normal operation of NAT. At most one entry in an instance of this table can exist with all of the same values for RemoteHost, ExternalPort, and PortMappingProtocol. If the ACS attempts to set the parameters of an existing entry such that this requirement would be

Name	Type	Write	Description
			violated, the CPE MUST reject the request. In this case, the SetParameterValues response MUST include a SetParameterValuesFault element for each parameter in the corresponding request whose modification would have resulted in such a violation. On creation of a new table entry, the CPE MUST choose default values for param and PortMappingProtocol such that the new entry does not conflict with any existing entry.
ExternalPort	Unsigned integer	W	The external port (or the first port of a range of external ports) that the NAT gateway would listen on for connection requests to a corresponding InternalPort. Inbound packets to this external port on the WAN interface SHOULD be forwarded to InternalClient on the InternalPort. A value of zero (0) represents a "wildcard". If this value is a wildcard, connection request on all external ports (that are not otherwise mapped) will be forwarded to InternalClient. In the wildcard case, the value(s) of InternalPort on InternalClient are ignored. When wildcard values are used for RemoteHost and/or ExternalPort, the following precedence order applies (with the highest precedence listed first): • Explicit RemoteHost, explicit ExternalPort • Explicit RemoteHost, wildcard ExternalPort • Wildcard RemoteHost, explicit ExternalPort • Wildcard RemoteHost, wildcard ExternalPort If an incoming packet matches the criteria associated with more than one entry in this table, the CPE MUST apply the port mapping associated with the highest precedence entry. At most one entry in this table can exist with all of the same values for RemoteHost, ExternalPort, and PortMappingProtocol.
ExternalPortEndRange	Unsigned integer	W	Indicates the last port of the external port range that starts with ExternalPort. If an external port range is specified, then the behavior described for ExternalPort applies to all ports within the range. A value of zero (0) indicates that no external port range is specified, i.e. that the range consists only of ExternalPort.

Name	Type	Write	Description
			If ExternalPort is zero (wildcard), the value of this parameter MUST be ignored. If specified, the value of this parameter MUST be greater than or equal to the value of ExternalPort.
InternalClient	string	W	The IP address or DNS host name of an internal client (on the LAN). Support for an IP address is mandatory. If InternalClient is specified as an IP address and the LAN device's IP address subsequently changes, the port mapping MUST remain associated with the original IP address. Support for DNS host names is OPTIONAL. If InternalClient is specified as a DNS host name and the LAN device's IP address subsequently changes, the port mapping MUST remain associated with this LAN device. In this case, it is the responsibility of the CPE to maintain the name-to-address mapping in the event of IP address changes. This can be accomplished, for example, by assigning the DNS host name via use of DHCP option 12 (Host Name) or option 81 (FQDN). Note that the ACS can learn the host name associated with a given LAN device via the Hosts table (InternetGatewayDevice.LANDevice.{i}.Hosts). Read access to this parameter MUST always return the exact value that was last set by the ACS. For example, if the internal client is set to a DNS host name, it MUST read back as a DNS host name and not as an IP address. An empty string indicates an unconfigured InternalClient. If this parameter is unconfigured, this port mapping MUST NOT be operational. It MUST be possible to set the InternalClient to the broadcast IP address 255.255.255.255 for UDP mappings. This is to enable multiple NAT clients to use the same well-known port simultaneously.
InternalPort	Unsigned integer	W	The port on InternalClient that the gateway SHOULD forward connection requests to. A value of zero (0) is not allowed.
PortMappingDescription	string	W	User-readable description of this port mapping.
PortMappingEnabled	boolean	W	Enables or disables the port mapping instance. On creation, an entry is disabled by default.

Name	Type	Write	Description
PortMappingLeaseDuration	integer	W	Determines the time to live, in seconds, of a port-mapping lease, where "time to live" means the number of seconds before the port mapping expires. A value of 0 means the port mapping is static. Support for dynamic (non-static) port mappings is OPTIONAL. That is, the only value for PortMappingLeaseDuration that MUST be supported is 0. For a dynamic (non-static) port mapping, when this parameter is read, the value represents the time remaining on the port-mapping lease. That is, for a dynamic port mapping, the value counts down toward 0. When a dynamic port-mapping lease expires, the CPE MUST automatically terminate that port mapping, and MUST automatically delete the corresponding PortMapping table entry.
PortMappingProtocol	string	W	The protocol of the port mapping. Enumeration of: ▪ TCP ▪ UDP At most one entry in this table can exist with all of the same values for RemoteHost, ExternalPort, and PortMappingProtocol.
RemoteHost	string	W	This parameter is the IP address of the source of inbound packets. empty indicates a "wildcard" (this will be a wildcard in most cases). CPE are REQUIRED only to support wildcards. WhenRemoteHost is a wildcard, all traffic sent to the ExternalPort on the WAN interface of the gateway is forwarded to the InternalClient on the InternalPort. When RemoteHost is specified as one external IP address, the NAT will only forward inbound packets from this RemoteHost to the InternalClient, all other packets will be dropped. If a CPE supports non-wildcard values for RemoteHost, it MAY additionally support the ability to have more than one port mapping with the same ExternalPort and PortMappingProtocol, but with differing values of RemoteHost. When wildcard values are used forRemoteHost and/or ExternalPort, the following precedence order applies (with the highest precedence listed first): ▪ # Explicit RemoteHost, explicit ExternalPort

Name	Type	Write	Description
			▪ # Explicit RemoteHost, wildcard ExternalPort ▪ # Wildcard RemoteHost, explicit ExternalPort ▪ # Wildcard RemoteHost, wildcard ExternalPort If an incoming packet matches the criteria associated with more than one entry in this table, the CPE MUST apply the port mapping associated with the highest precedence entry. At most one entry in this table can exist with all of the same values for RemoteHost, ExternalPort, and PortMappingProtocol.
InternetGatewayDevice.WANDevice.{i}.WANDSLInterfaceConfig.	object	-	This object models physical layer properties specific to a single physical connection of a DSL modem used for Internet access on a CPE. This object is required for a CPE with a DSL modem WAN interface, and is exclusive of any other "WAN*InterfaceConfig" object within a given "WANDevice" instance.
AllowedProfiles	string	-	Comma-separated list of strings. Indicates which VDSL2 profiles are allowed on the line. Each list item is an enumeration of: ▪ 8a ▪ 8b ▪ 8c ▪ 8d ▪ 12a ▪ 12b ▪ 17a ▪ 17b ▪ 30a Note: In G.997.1, this parameter is called PROFILES. See ITU-T Recommendation G.997.1. Note: This parameter is OPTIONAL at the G and S/T interfaces in G.997.1 Amendment 1. If the parameter is implemented but no value is available, its value MUST be an empty string.
CurrentProfile	string	-	Indicates which VDSL2 profile is currently in use on the line. Note: This parameter is OPTIONAL at the G and S/T interfaces in G.997.1 Amendment 1. If the parameter is implemented but no value is available, its value MUST be empty.
DownstreamAttenuation	int	-	The current downstream signal loss (expressed in 0.1 dB).

Name	Type	Write	Description
DownstreamCurrRate	unsignedInt	-	The current physical layer aggregate data rate (expressed in Kbps) of the downstream DSL connection. Note: This parameter is OPTIONAL at the G and S/T interfaces in G.997.1 Amendment 1. If the parameter is implemented but no value is available, it MUST have the value 4294967295 (the maximum for its data type).
DownstreamMaxRate	unsignedInt	-	The current attainable rate (expressed in Kbps) of the downstream DSL channel. Note: This parameter is related to the G.997.1 parameter ATTNDRds, which is measured in bits/s. See ITU-T Recommendation G.997.1.
DownstreamNoiseMargin	int	-	The current signal-to-noise ratio margin (expressed in 0.1 dB) in the downstream direction. Note: In G.997.1, this parameter is called SNRMds. See ITU-T Recommendation G.997.1.
DownstreamPower	int	-	The current received power at the CPE's DSL interface (expressed in 0.1 dBmV).
Enable	boolean	W	Enables or disables the link.
LinkEncapsulationSupported	string	-	Indicates which link encapsulation standards and recommendations are supported by the B-NT.
LinkEncapsulationUsed	string	-	Indicates the link encapsulation standard that the B-NT is using for the connection. Enumeration of: • G.992.3_Annex_K_ATM • G.992.3_Annex_K_PTM • G.993.2_Annex_K_ATM • G.993.2_Annex_K_PTM When the standard identifies ATM encapsulation then the InternetGatewayDevice.WANDevice.{i}.WANConnectionDevice.{i}.WANDSLLinkConfig object MUST be used. When the standard identifies PTM encapsulation then the InternetGatewayDevice.WANDevice.{i}.WANConnectionDevice.{i}.WANPTMLinkConfig object MUST be used.
ShowtimeStart	unsignedInt	-	Number of seconds since the most recent DSL Showtime - the beginning of the period used for collection of "Showtime" statistics. Showtime is defined as successful completion of the DSL link establishment process. The "Showtime" statistics are

Name	Type	Write	Description
			those collected since the most recent establishment of the DSL link.
StandardsSupported	string	-	Indicates which DSL standards and recommendations are supported by the B-NT.
StandardUsed	string	-	Indicates the standard that the B-NT is using for the connection.
Status	string	-	Status of the DSL physical link. Enumeration of: ▪ Up ▪ Initializing ▪ EstablishingLink ▪ NoSignal ▪ Error (OPTIONAL) ▪ Disabled The Error value MAY be used by the CPE to indicate a locally defined error condition.
TotalStart	unsignedInt	-	Number of seconds since the beginning of the period used for collection of "Total" statistics. Statistics SHOULD continue to be accumulated across CPE reboots, though this might not always be possible.
UpstreamAttenuation	int	-	The current upstream signal loss (expressed in 0.1 dB).
UpstreamCurrRate	unsignedInt	-	The current physical layer aggregate data rate (expressed in Kbps) of the upstream DSL connection. Note: This parameter is OPTIONAL at the G and S/T interfaces in G.997.1 Amendment 1. If the parameter is implemented but no value is available, it MUST have the value 4294967295 (the maximum for its data type).
UpstreamMaxRate	unsignedInt	-	The current attainable rate (expressed in Kbps) of the upstream DSL channel. Note: This parameter is related to the G.997.1 parameter ATTNDRus, which is measured in bits/s. See ITU-T Recommendation G.997.1.
UpstreamNoiseMargin	int	-	The current signal-to-noise ratio margin (expressed in 0.1 dB) in the upstream direction. Note: In G.997.1, this parameter is called SNRMus. See ITU-T Recommendation G.997.1.
UpstreamPower	int	-	The current output power at the CPE's DSL interface (expressed in 0.1 dBmV).

Name	Type	Write	Description
InternetGatewayDevice.WANDevice .{i}.WANDSLInterfaceConfig.Stats.	object	-	This object contains statistics for a WAN DSL physical interface.
InternetGatewayDevice.WANDevice .{i}.WANDSLInterfaceConfig.Stats. Showtime.	object	-	This object contains DSL statistics accumulated since the most recent DSL Showtime.
CRCErrors	unsignedInt	-	Number of CRC errors detected since the most recent DSL Showtime (CV-C as defined in ITU-T Rec. G.997.1). Note: This parameter is OPTIONAL at the G and S/T interfaces in G.997.1 Amendment 1. If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
InternetGatewayDevice.WANDevice .{i}.WANDSLInterfaceConfig.Stats. Total.	object	-	This object contains DSL total statistics.
CRCErrors	unsignedInt	-	Total number of CRC errors detected (CV-C as defined in ITU-T Rec. G.997.1). Note: This parameter is OPTIONAL at the G and S/T interfaces in G.997.1 Amendment 1. If the parameter is implemented but no value is available, its value MUST be 4294967295 (the maximum for its data type).
InternetGatewayDevice.WANDevice .{i}.WANEthernetInterfaceConfig.	object	-	This object models physical layer properties specific to a single Ethernet physical connection used for Internet access on a CPE. This object is required for a CPE with an Ethernet WAN interface, and is exclusive of any other "WAN*InterfaceConfig" object within a given "WANDevice" instance. Note that this object is not related to the Ethernet protocol layer sometimes used in associated with a DSL connection.
DuplexMode	string	-	The duplex mode available to this connection.
Enable	boolean	W	Enables or disables this interface.
MACAddress	string	-	The physical address of the interface.
MaxBitRate	string	-	The maximum upstream and downstream bit rate available to this connection.
Status	string	-	Indicates the status of this interface. Enumeration of: ▪ Up ▪ NoLink ▪ Error (OPTIONAL) ▪ Disabled

Name	Type	Write	Description
			The Error value MAY be used by the CPE to indicate a locally defined error condition.
InternetGatewayDevice.WANDevice.{i}.WANEthernetInterfaceConfig.Stats.	object	-	This object contains statistics for an Ethernet WAN interface on a CPE device.
BytesReceived	unsignedInt	-	Total number of bytes received over the interface since the CPE was last reset.
BytesSent	unsignedInt	-	Total number of bytes sent over the interface since the CPE was last reset.
PacketsReceived	unsignedInt	-	Total number of packets received over the interface since the CPE was last reset.
PacketsSent	unsignedInt	-	Total number of packets sent over the interface since the CPE was last reset.

6 TR-104 Data Model

TR-104 defines the data model for the provisioning of a voice-over-IP (VoIP) CPE device by an Auto-Configuration Server (ACS) using the mechanism defined in TR-069.

The following profiles are implemented by MSBR products:
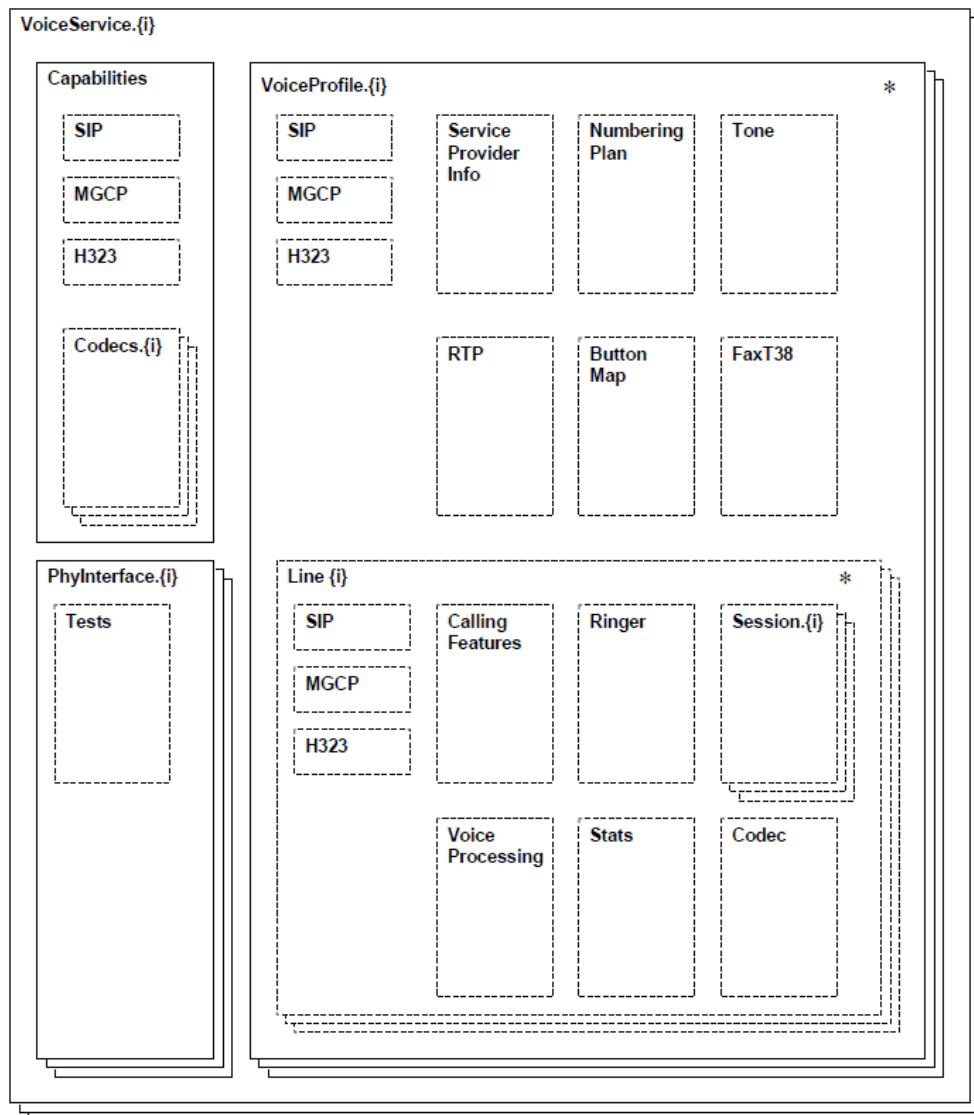
- Endpoint Profile
- SIP Endpoint Profile

> **Note:** TR-104 is supported only for FXS interfaces.

6.1 Major Elements

The following diagram provides an overview of all objects defined in TR-104:

Figure 6-1: Objects Defined in TR-104

TR-104 describes functionality of a VoIP device that has one or more FXS lines.

The MSBR configuration model differs considerably from the model described in TR-104. For example, instead of the *VoiceProfile* object that corresponds to a group of phone lines, a separate *TelProfile* and/or IP Profile objects are used. TR-104 uses an implicit routing model while MSBR supports explicit routing tables that may be used to implement much more complicated call routings.

To "bridge" between these two very different configuration models, the following is assumed regarding the way MSBR is configured:

- Default Tel Profile (0) is used for all Hunt Groups (FXS Lines).
- "IP to Tel" Routing for phone lines (FXS ports) is implemented via the *PSTNPrefix* table. For each phone line (FXS port), a separate and unique entry in this table is created.
- "Tel to IP" routing is performed by the default *ProxySet (0)* and when the *IsProxySet* parameter is set to "1" (enabled). The first IP address entry in the Proxy Set configuration is used.
- Default IP Profile (0) is used for all routing rules.
- Outbound Proxy (if needed) is implemented via a single line in the *PSTNPrefix* table with '*' wildcard for source and destination prefix.

6.2   VoiceService

VoiceService is a container "service" object as defined in TR-106.

MSBR implements a single instance of this object – VoiceService.1 only if it is equipped with the FXS ports.

6.2.1   VoiceService.{i}.Capabilities.Codecs

Codecs table describes the set of supported codecs. The table is read-only. Line.{i}.Codec.List table is used instead to customize list and parameters of coders assigned to the specific phone line.

MSBR will populate this table with *all supported* coders (as per CoderName_Type and CoderName_Rate ENUMs, taking into account the current DSP template). Each entry will contain "default" parameters (e.g. p-time) that correspond to the specific coder.

6.2.2   VoiceService.{i}.VoiceProfile

VoiceProfile corresponds to one or more phone lines (FXS ports) that share the same basic configuration.

MSBR implements a single instance of the VoiceProfile.1 object.

6.2.3   VoiceService.{i}.VoiceProfile.{i}.Line

The Line object corresponds to a single phone line (FXS port).

MSBR implements an instance of this object for each phone line (FXS port) configured in Hunt Group (TrunkGroup) table. Add/remove operations will be supported to allow configuration/removal of specific FXS port (See details in the table below).

6.2.4   VoiceService.{i}.VoiceProfile.{i}.Line.{i}.Codec

The Codec object describes voice coder currently used by the specific phone line. In addition to that it provides a list of coders (Codec.List.{i}) enabled on the specific line and the ability to modify this configuration.

MSBR implements a global Coder table that is shared by all phone lines (FXS ports). It allows "per-line" customization of coders via Coder Group table. However, use of this functionality

requires use of different Tel Profiles for different Hunt Groups (FXS port) and number of coder groups that may be configured is limited to 4.

Map Codec.List.{i} table directly to Coders (CoderName) table. This essentially means that all lines share the same configuration and configuration change for one line immediately affects all other lines.

6.2.5   VoiceService.{i}.VoiceProfile.{i}.Line.{i}.SIP

The Line.{i}.SIP object contains username/password assigned to a specific phone line (FXS port).

MSBR will map this object to corresponding entry in Authentication Table.

6.2.6   VoiceService.{i}. X_0090F8_SwitchOverToProxy

The X_0090F8_SwitchOverToProxy parameter is an AudioCodes proprietary parameter, which disables automatic dialing between FXS and FXO endpoints.

6.3   Endpoint Profile

MSBR complies with Version 1 of Endpoint profile (Endpoint:1) as defined in TR-104 Section 4.2.

Table 6-1: Endpoint Profile Table

Name	Type	Write	Description	Comment
Services.VoiceService.{i}.	object	–	The top-level object.	A single instance of VoiceService object will be created.
VoiceProfileNumberOfEntries	unsignedInt	–	Defines the number of instances of VoiceProfile.	1
.VoiceService.{i}.Capabilities.	object	–	Defines the overall capabilities of the VoIP CPE.	
MaxProfileCount	unsignedInt	–	Defines the maximum total number of distinct voice profiles supported.	1
MaxLineCount	unsignedInt	–	Defines the maximum total number of lines supported across all profiles. This parameter is applicable only for a VoIP endpoint.	Total number of FXS interfaces (e.g. 4).
MaxSessionsPerLine	unsignedInt	–	Defines the maximum number of voice sessions supported for any given line across all profiles. A value greater than one indicates support	1

Name	Type	Write	Description	Comment
			for CPE provided conference calling. This parameter is applicable only for a VoIP endpoint.	
MaxSessionCount	unsignedInt	–	Defines the maximum total number of voice sessions supported across all lines and profiles. (This might differ from MaxLineCount if each line can support more than one session for CPE provided conference calling. This value MAY be less than the product of MaxLineCount and MaxSessionsPerLine.)	same value as MaxLineCount
SignalingProtocols	string(256)	–	Defines the comma-separated list of signaling protocols supported. Each item is an enumeration of: "SIP" "MGCP" "MGCP-NCS" "H.248" "H.323" Each entry MAY be appended with a version indicator in the form "/X.Y". For example: "SIP/2.0" The list MAY include vendor-specific protocols, which MUST be in the format defined in [3]. For example: "X_EXAMPLE-COM_MyProt"	"SIP"
Regions	string(256)	–	Defines the comma-separated list of geographic regions supported by the CPE. Each item is the list MUST be an alpha-2 (two-character alphabetic) country code as	<empty>

Name	Type	Write	Description	Comment
			specified by ISO 3166. An empty list indicates that the CPE does not support region-based customization via the Region parameter in the VoiceService.{i}.Voice Profile.{i} object.	
RTCP	boolean	–	Defines support for RTCP. A true value indicates support for the object VoiceService.{i}.Voice Profile.{i}.RTP.RTCP. This parameter is applicable only for a VoIP endpoint.	True
SRTP	boolean	–	Defines support for SRTP. A true value indicates support for the object VoiceService.{i}.Voice Profile.{i}.RTP.SRTP. A true value also indicates that the SRTPKeyingMethods and SRTPEncryptionKeyS izes parameters in this object are present. This parameter is applicable only for a VoIP endpoint.	True
SRTPKeyingMethods	string(256)	–	Defines comma-separated list of keying protocols supported by this endpoint for SRTP. Each item is an enumeration of: "Null" "Static" "SDP" "IKE" This list MAY include vendor-specific keying methods, which MUST use the format defined in [3].	"Static"

Name	Type	Write	Description	Comment
			This parameter is applicable only if the SRTP parameter in this object is equal to true. **Note:** This parameter is NOT part of *EndpointProfile*, but it's mandatory when SRTP is supported.	
SRTPEncryptionKeySizes	string(256)	–	Defines comma-separated list of unsigned integers, each represented a supported SRTP encryption key size. This parameter is applicable only if the SRTP parameter in this object is equal to true.	according to SRTPofferedSuites parameter possible values, the only supported value is "128"
RTPRedundancy	boolean	–	Defines support for RTP payload redundancy as defined in RFC 2198. A true value indicates support for VoiceService.{i}.VoiceProfile.{i}.RTP.Redundancy. This parameter is applicable only for a VoIP endpoint.	True
DSCPCoupled	boolean	–	Defines a true value that indicates that the CPE is constrained such that transmitted call control packets use the same DSCP marking as transmitted RTP packets. If the value is true, the CPE MUST NOT support the DSCPMark parameter for call control. This parameter is applicable only for a VoIP endpoint.	False
EthernetTaggingCoupled	boolean	–	Defines a true value that indicates that the CPE is constrained such that transmitted	False

Name	Type	Write	Description	Comment
			call control packets use the same Ethernet tagging (VLAN ID Ethernet Priority) as transmitted RTP packets. If the value is true, the CPE MUST NOT support the VLANIDMark or EthernetPriorityMark parameters within a call control object (e.g., SIP, MGCP, or H323). This parameter is applicable only for a VoIP endpoint.	
PSTNSoftSwitchOver	boolean	–	Defines a true value that indicates the CPE is capable of supporting the PSO_Activate Facility Action, which allows a call to be switched to a PSTN FXO (Foreign eXchange Office) line. This parameter is applicable only for a VoIP endpoint.	False
FaxT38	boolean	–	Defines support for T.38 fax. A true value indicates support for the object VoiceService.{i}.Voice Profile.{i}.FaxT38. This parameter is applicable only for a VoIP endpoint.	True
FaxPassThrough	boolean	–	Defines support for fax pass-through. A true value indicates support for the parameter VoiceService.{i}.Voice Profile.{i}.FaxPassThr ough. This parameter is applicable only for a VoIP endpoint.	True
ModemPassThrough	boolean	–	Defines support for modem pass-through.	True

Name	Type	Write	Description	Comment
			A true value indicates support for the parameter VoiceService.{i}.Voice Profile.{i}.ModemPass Through. This parameter is applicable only for a VoIP endpoint.	
ToneGeneration	boolean	–	Defines support for tone generation. A true value indicates support for the object VoiceService.{i}.Voice Profile.{i}.Tone. A true value also indicates that the ToneDescriptionsEdit able, PatternBasedToneGe neration, and FileBasedToneGener ation parameters in this object are present. This parameter is applicable only for a VoIP endpoint.	False
RingGeneration	boolean	–	Defines support for ring generation. A true value indicates support for control of ring generation via the VoiceService.{i}.Voice Profile.{i}.Line.{i}.Ring er object. A true value also indicates that the RingDescriptionsEdita ble, PatternBasedRingGe neration, and FileBasedRingGenera tion parameters in this object are present. This parameter is applicable only for a VoIP endpoint.	False
NumberingPlan	boolean	–	Defines support for a configurable numbering plan. A true value indicates support for a configurable	False

Name	Type	Write	Description	Comment
			numbering plan via the VoiceService.{i}.Voice Profile.{i}.NumberingP lan object. This parameter is applicable only for a VoIP endpoint.	
ButtonMap	boolean	–	Defines support for a configurable button map. A true value indicates support for a configurable button map via the VoiceService.{i}.Voice Profile.{i}.ButtonMap object. This parameter is applicable only for a VoIP endpoint.	False
VoicePortTests	boolean	–	Defines support for remotely accessible voice-port tests. A true value indicates support for the VoiceService.{i}.PhyIn terface.{i}.Tests object. This parameter is applicable only for a VoIP endpoint.	False
.VoiceService.{i}.Capab ilities.Codecs.{i}.	object	–	Table to describe the set of supported codecs. Each entry refers to a distinct combination of codec and bit rate. The table MUST include a distinct entry for each supported combination of these parameters. Applicable only for a VoIP endpoint.	This table will be populated with entries for each supported coder (as per CoderName_Type enum and applicable bit-rates). If possible we should limit list of supported coders to those supported in current DSP template only. For each entry, default coder parameters (e.g. p-time) will be specified. **Note:** This table is NOT mapped to Coders (CoderName) table. The latter is mapped to Line.{i}.Codec.List table.

Name	Type	Write	Description	Comment
EntryID	unsignedInt [1:]	–	Defines a unique identifier for each entry in this table.	
Codec	string(64)	–	Defines an identifier of the type of codec. Enumeration of: ▪ "G.711MuLaw" ▪ "G.711ALaw" ▪ "G.726" ▪ "G.729" ▪ "G.729a" ▪ "G.729e" ▪ "G.728" ▪ "G.723.1" ▪ "G.722" ▪ "G.722.1" ▪ "G.722.2" ▪ "GSM-FR" ▪ "GSM-HR" ▪ "GSM-EFR" ▪ "iLBC" The parameter MAY instead be a vendor-specific codec, which MUST be in the format defined in [3]. For example: "X_EXAMPLE-COM_MyCodec"	
BitRate	unsignedInt	–	Defines a Bit Rate in bits per second. The value MUST be among the values appropriate for the specified codec.	
PacketizationPeriod	string(64)	–	Defines a comma-separated list of supported packetization periods, in milliseconds, or continuous ranges of packetization periods. Ranges are indicated as a hyphen-separated pair of unsigned integers. Examples: "20" indicates a single discrete value. "10, 20, 30" indicates a set of discrete	

Name	Type	Write	Description	Comment
			values. "5-40" indicates a continuous inclusive range. "5-10, 20, 30" indicates a continuous range in addition to a set of discrete values A range MUST only be indicated if all values within the range are supported.	
SilenceSuppression	boolean	–	Indicates support for silence suppression for this codec.	
.VoiceService.{i}.Voice Profile.{i}.	object	C	Support for creation and deletion of Profiles is REQUIRED only if more than one Profile is supported as indicated by VoiceService.{i}.Capabilities.MaxProfileCount.	A single instance of VoiceProfile object will be created. Add/remove operations will NOT be supported.
Enable	string	W	Enables or disables all lines in this profile, or places it into a quiescent state. Enumeration of: "Disabled" "Quiescent" "Enabled" On creation, a profile MUST be in the Disabled state. In the Quiescent state, in-progress sessions remain intact, but no new sessions are allowed. Support for the Quiescent state in a CPE is optional. If this parameter is set to "Quiescent" in a CPE that does not support the Quiescent state, it MUST treat it the same as the Disabled state.	Only 'Enabled' state will be supported.

Name	Type	Write	Description	Comment
Reset	boolean	W	When written as true, forces all lines in this profile to be reset, causing it to re-initialize and perform all start-up actions such as registration. Always false when read.	
NumberOfLines	unsignedInt	–	Defines the number of instances of Line within this VoiceProfile. Applicable only for a VoIP endpoint.	Number of entries in Hunt Group (TrunkGroups) table of type FXS.
Name	string(64)	W	Defines a human-readable string to identify the profile instance.	'Default Profile' Write operation will not be supported.
SignalingProtocol	string(64)	W	Defines the protocol to be used for this profile. A single protocol selected from among the available protocols indicated in VoiceService.{i}.Capabilities.SignalingProtocols.	"SIP"
MaxSessions	unsignedInt	W	Defines the limit on the number of simultaneous voice sessions across all lines in this profile. Must be less than or equal to VoiceService.{i}.Capabilities.-MaxSessionCount. (This MAY be greater than the number of lines if each line can support more than one session, for example for CPE provided conference calling.)	We will assume that a single session is supported per line – hence we will use the same value as for NumberOfLines.
DTMFMethod	string(64)	W	Defines the method by which DTMF digits MUST be passed. Enumeration of: "InBand" "RFC2833"	RxDTMFOption==0 && TxDTMFOption==1 → "SIPInfoNortel" RxDTMFOption==0 && TxDTMFOption==3 → "SIPInfo"

Name	Type	Write	Description	Comment
			"SIPInfo" If the parameter DTMFMethodG711 is non-empty, then this parameter applies only when the current codec is not G.711. The value "SIPInfo" is applicable only if the SignalingProtocol is SIP. This parameter is applicable only for a VoIP endpoint.	RxDTMFOption==0 && TxDTMFOption==2 → "Notify" RxDTMFOption==3 && TxDTMFOption==4 → "RFC2833" RxDTMFOption==0 && TxDTMFOption==0 && DTMFTransportType== 2 → "InBand" Any other combination → "Other" (read-only) **Note:** These are _global_ parameters. They may be overwritten by setting corresponding parameter in IPProfile. If we manage to "link" TelProfile and IPProfile together (see …) , we'll use parameters from IPProfile; otherwise global parameters will be used.
DTMFMethodG711	string(64)	W	Defines the method by which DTMF digits MUST be passed if the current codec is G.711. Enumeration of: "InBand" "RFC2833" "SIPInfo" An empty value for this parameter indicates that the value of the DTMFMethod parameter is to apply whether or not the current codec is G.711. The value "SIPInfo" is applicable only if the SignalingProtocol is SIP. This parameter is applicable only for a VoIP endpoint.	We support "empty" values only.
.VoiceService.{i}.Voice Profile.{i}.RTP.	object	–	Voice profile parameters related to	

Name	Type	Write	Description	Comment
			the voice stream sent via RTP. Applicable only for a VoIP endpoint.	
LocalPortMin	unsignedInt [0:65535]	W	Defines the base of port range to be used for incoming RTP streams for this profile.	BaseUDPport (or use Media Realm's Port Range Start, if Media Realm is defined)
DSCPMark	unsignedInt [0:63]	W		
TelephoneEventPayloadType	unsignedInt [0:128]	W	Defines the payload type to be used for RTP telephone events. This parameter indicates the payload type to be used for DTMF events if RFC 2833 transmission of DTMF information is in use.	According to RFC 2833 PayloadType.
.VoiceService.{i}.VoiceProfile.{i}.Line.{i}.	object	C	Defines the support for creation and deletion of Lines is REQUIRED only if more than one Line is supported as indicated by VoiceService.{i}.Capabilities.MaxLineCount.	An instance of this object is created for each Hunt Group / TrunkGroup entry of type FXS. Add operation will be supported only if some "unused" FXS line exists. It will create a new entry in Hunt Group / TrunkGroup table. Remove operation will delete corresponding entry from Hunt Group table.
Enable	string	W	Enables or disables this line, or places it into a quiescent state. Enumeration of: "Disabled" "Quiescent" "Enabled" On creation, a line MUST be in the Disabled state. In the Quiescent state, in-progress sessions remain	Quiescent state is not supported. Disabled state is implemented "at TR069 level only" (the entry will NOT exist in Hunt Group table).

Reference Guide   194   Document #: LTRT-52357

Name	Type	Write	Description	Comment
			intact, but no new sessions are allowed. Support for the Quiescent state in a CPE is optional. If this parameter is set to "Quiescent" in a CPE that does not support the Quiescent state, it MUST treat it the same as the Disabled state (and indicate Disabled in the Status parameter).	
Status	string	–	Indicates the status of this line. Enumeration of: "Up" "Initializing" "Registering" "Unregistering" "Error" "Testing" "Quiescent" "Disabled"	The following statuses are supported: Up Disabled
CallState	string	–	Indicates the call state for this line. Enumeration of: "Idle" "Calling" "Ringing" "Connecting" "InCall" "Hold" "Disconnecting"	Need to query SIP database to get this info.
.VoiceService.{i}.Voice Profile.{i}.Line.{i}.Code c.	object	–	This object indicates the state of the transmit and receive codec for this voice line instance.	**Note:** This object shows the data about currently established media session (i.e. the data should be taken from MediaEngine).
TransmitCodec	string(64)	–	Defines the codec currently in use for the outgoing voice stream. Enumeration from the list of available codecs as given in the VoiceService.{i}.Capa bilities.Codecs table.	

Name	Type	Write	Description	Comment
ReceiveCodec	string(64)	–	Defines the codec currently in use for the incoming voice stream. Enumeration from the list of available codecs as given in the VoiceService.{i}.Capabilities.Codecs table.	
TransmitBitRate	unsignedInt	–	Defines the codec bit rate in bits per second for the codec currently in use for the outgoing voice stream.	
ReceiveBitRate	unsignedInt	–	Defines the codec bit rate in bits per second for the codec currently in use for the incoming voice stream.	
TransmitSilenceSuppression	boolean	–	Defines whether or not silence suppression is in use for the outgoing voice stream.	
ReceiveSilenceSuppression	boolean	–	Defines whether or not silence suppression is in use for the incoming voice stream.	
TransmitPacketizationPeriod	unsignedInt	–	Defines the current outgoing packetization period in milliseconds.	
.VoiceService.{i}.VoiceProfile.{i}.Line.{i}.Codec.List.{i}.	object	–	Table to describe the set of codecs enabled for use with this line. Each entry in this table refers to a distinct combination of codec and bit rate. When a Line is created, this object MUST be populated with the set of supported codecs matching the VoiceService.{i}.Capabilities.Codecs table. The ACS MAY restrict and/or prioritize the codec support for this profile using this object.	The table will be populated with all supported coders (similar to Capabilities.Codecs table). Each "Enabled" object will be mapped to the entry in Coders (CoderName) table. Index in Coders table will be determined according to the object's Priority parameter. "Disabled" objects will exist "at TR-069 level only".

Name	Type	Write	Description	Comment
			Applicable only for a VoIP endpoint.	
EntryID	unsignedInt [1:]	–	Defines the unique identifier for each entry in this table. The value MUST match that of the corresponding entry in the VoiceService.{i}.Capabilities.Codecs table.	Similar to Capabilities.Codecs
Codec	string(64)	–	Defines the Identifier of the codec type. The value MUST match that of the corresponding entry in the VoiceService.{i}.Capabilities.Codecs table.	Initial value - similar to Capabilities.Codecs; when "Enabled" – CoderName_Type
BitRate	unsignedInt	–	Defines the Bit rate in bits per second. The value MUST match that of the corresponding entry in the VoiceService.{i}.Capabilities.Codecs table.	Initial value - similar to Capabilities.Codecs; ; when "Enabled" – CoderName_Rate
PacketizationPeriod	string(64)	W	Defines the comma-separated list of supported packetization periods, in milliseconds, or continuous ranges of packetization periods as defined in VoiceService.{i}.Capabilities.Codecs.PacketizationPeriod. The set of packetization periods may be restricted by modifying the value of this parameter to a more restricted set of values than is listed in VoiceService.{i}.Capabilities.Codecs.PacketizationPeriod. The CPE MUST ignore any values or portions of ranges outside of those specified in VoiceService.{i}.Capabil	Initial value – similar to Capabilities.Codecs; when "Enabled" – CoderName_PacketInterval Can be modified by user.

Name	Type	Write	Description	Comment
			ities.Codecs.Packetizati on.Period.	
SilenceSuppression	boolean	W	Indicates support for silence suppression for this codec. If silence suppression is supported, it can be disabled for this codec/bit-rate by setting this parameter to false	Initial value – similar to Capabilities.Codecs; when "Enabled" – CoderName_SCE Can be modified by user.
Enable	boolean	W	This parameter is REQUIRED to be writable only if there is more than one entry in this table.	When set to 'true' corresponding entry in Coders (CoderName) table is created (with index that corresponds to Priority parameter). *__When set to 'false', corresponding entry is removed from the Coders table__*. **Note:** in DR add 2 options: Either not able option to put DISABLE to this param, or add new field of Admin-State to Coder Tables
Priority	unsignedI nt [1:]	W	This parameter is REQUIRED to be writable only if there is more than one entry in this table.	Will be used to determine index of entry in Coders (CoderName) table. When value is changed and Enable=='true' existing Coders entry will be removed and a new entry will be created instead (with a new index).
.VoiceService.{i}.Voice Profile.{i}.Line.{i}.Sessi on.{i}.	object	–	Information on each active session associated with this voice line instance.	Information about currently active call – should be taken from SIP DB.
SessionStartTime	date time	–	Defines the time that the session started, in UTC.	
SessionDuration	Unsignedi nt	–	Defines the duration time of the current session, in seconds.	
FarEndIPAddress	string	–	Defines the IP address of far end VoIP device.	

Name	Type	Write	Description	Comment
FarEndUDPPort	unsignedInt [0:65535]	–	Defines the UDP port used for current RTP session in the far end device.	
LocalUDPPort	unsignedInt [0:65535]	–	Defines the local UDP port used for current RTP session.	
.VoiceService.{i}.Voice Profile.{i}.Line.{i}.Stats.	object	–	Statistics for this voice line instance.	Will be mapped to corresponding PMs (if such exist), SIP or VE counters. We may need to create new PMs to support all needed counters.
PacketsLost	unsignedInt	–	Defines the total number of RTP packets that have been lost for this line.	

6.3.1   Endpoint Profile Extensions

The following parts of TR-104 are not part of Endpoint profile (Endpoint:1) however are applicable to MSBR due to one of the following:

■   are implied from the way we implement VoiceService.1.Capabilities object.

■   were explicitly requested by potential customer

Table 6-2: Endpoint Profile Extensions Table

Name	Type	Write	Description	Comment
.VoiceService.{i}. VoiceProfile.{i}.	object	–		
FaxPassThrough	string	W	Specifies the behavior of the CPE for pass-through of fax data. Enumeration of: "Disable" "Auto" "Force" The value "Disable" prevents the CPE from switching to a fax pass-through mode. The value "Auto" allows the CPE to automatically detect fax data to determine whether or not to switch to a fax pass-through mode. The value "Force" forces the CPE to switch to a fax pass-through mode regardless of	According to IsFaxUsed and FaxTransportMode parameters: **Read:** (IsFaxUsed==2) || ((IsFaxUsed==0) && (FaxTransportMode== 2)) →"Auto" Otherwise → "Disabled" **Write:** "Auto" → IsFaxUsed=2 "Disable"→ IsFaxUsed=0, FaxTransportMode=0 (unless

Name	Type	Write	Description	Comment
			whether fax signaling is detected. If this parameter is supported, the capability VoiceService.{i}.Capabilities. FaxPassThrough MUST be equal to true. This parameter is appropriate only for a VoIP endpoint.	FaxT38.Enable == true) **Note**: the same values may be taken from "default" IPProfile 0.
ModemPassThrough	string	W	Specifies the behavior of the CPE for passthrough of modem data. Enumeration of: "Disable" "Auto" "Force" The value "Disable" prevents the CPE from switching to a modem pass-through mode. The value "Auto" allows the CPE to automatically detect modem data to determine whether or not to switch to a modem pass-through mode. The value "Force" forces the CPE to switch to a modem pass-through mode regardless of whether modem signaling is detected. If this parameter is supported, the capability VoiceService.{i}.Capabilities. ModemPassThrough MUST be equal to true. This parameter is appropriate only for a VoIP endpoint.	According to V21ModemTransport Type, V22ModemTransport Type, V23ModemTransport Type, V32ModemTransport Type, V34ModemTransport Type and BellModemTransportT ype parameters: **Read:** If all VxxModemTransportT ype == 2 → "Auto" otherwise → "False" **Write:** "Auto" → all VxxModemTransportT ype = 2 "Disable" → all VxxModemTransportT ype = 0
.VoiceService.{i}. VoiceProfile.{i}.R TP.RTCP.	object	–	Voice profile parameters related to RTCP. If this object is supported, the capability VoiceService.{i}.Capabilities. RTCP MUST be equal to true. Applicable only for a VoIP endpoint.	Neither of the below parameters are configurable via WEB/CLI/INI file.
Enable	boolean	W	Enables or disables RTCP.	Internal parameter: RtcpInterval==0

Name	Type	Write	Description	Comment
TxRepeatInterval	unsignedInt [1:]	W	Defines the Transmission repeat interval, in milliseconds	Internal parameter: RtcpInterval
VoiceService.{i}.VoiceProfile.{i}.RTP.SRTP.	object	–	Defines the Voice profile parameters for secure voice transmission via SRTP. If this object is supported, the capability VoiceService.{i}.Capabilities.SRTP MUST be equal to true. Applicable only for a VoIP endpoint.	
Enable	boolean	W	Enables or disables the use of SRTP. If RTCP is enabled, a true value of this parameter also implies the use of SRTCP.	EnableMediaSecurity
KeyingMethods	string(256)	W	Defines a comma-separated list of keying methods that may be used. The value MUST be a subset of those listed in the parameter VoiceService.{i}.Capabilities.SRTPKeyingMethods.	"Static" Write operation will not be supported.
EncryptionKeySizes	string(256)	W	Defines a comma-separated list of encryption key sizes that may be used. The value MUST be a subset of those listed in the parameter VoiceService.{i}.Capabilities.SRTPEncryptionKeySizes.	According to SRTPofferedSuites parameter. But since we support only "128" – this is the only possible value.
VoiceService.{i}.VoiceProfile.{i}.RTP.Redundancy	object	–	Defines Voice profile parameters for RTP payload redundancy as defined by RFC 2198. If this object is supported, the capability VoiceService.{i}.Capabilities.Redundancy MUST be equal to true. Applicable only for a VoIP endpoint.	
Enable	boolean	W	Enables or disables the use of RTP payload redundancy as defined by RFC 2198.	RTPRedundancyDepth
PayloadType	unsignedInt [0:127]	W	Defines the Payload Type of RTP packet using RFC 2198. Values should be	RFC2198PayloadType

Name	Type	Write	Description	Comment
			within the range of dynamic Payload Types (96-127).	
FaxAndModemRedundancy	int[-1:5]	W	Specifies the redundancy number for fax and modem pass-through data transmissions. A non-negative value indicates that RFC 2198 is to be used for fax and modem pass-through data. The value indicates the number of redundant copies to be transmitted (the total number transmitted is one plus this value). A value of -1 indicates RFC 2198 is not to be used for fax and modem pass-through data. If the optional parameter ModemRedudancy is present, then FaxAndModemRedudancy applies only to fax transmissions, but not to modem transmissions.	FaxRelayRedundancy Depth
VoiceService.{i}.VoiceProfile.{i}.NumberingPlan	object	-	This object contains information related the numbering plan. This object is applicable only if the device supports a dialing mechanism for which a number plan is needed (for example, a device with an explicit Dial button may not need to be aware of the dialing plan) and if the device does not already support a numbering plan mechanism for this profile (e.g., in-band via MGCP). If supported, the value of .Capabilities.NumberingPlan MUST be true. Applicable only for a VoIP endpoint.	
InterDigitTimerStd	unsignedInt	W	This timer is the maximum allowable time (expressed in milliseconds) between the dialing of digits. This timer is restarted every time a digit is dialed. Expiration of this timer indicates "End of Dialing".	

Name	Type	Write	Description	Comment
VoiceService.{i}.VoiceProfile.{i}.FaxT38	object	–	Defines T.38 Fax information for devices that support T.38 relay. If this object is supported, the capability VoiceService.{i}.Capabilities.FaxT38 MUST be equal to true. Applicable only to a VoIP endpoint.	
Enable	boolean	W	Enables or disables the use of T.38.	According to IsFaxUsed and FaxTransportMode parameters: **Read**: ((IsFaxUsed==1) || (IsFaxUsed==3) || ((IsFaxUsed==0) && (FaxTransportMode==1)) → true otherwise → false **Write:** true → IsFaxUsed=1 false → IsFaxUsed=0, FaxTransportMode=0 (unless FaxPassThrough == "Auto") **Note:** The same values may be taken from "default" IPProfile 0.
BitRate	unsignedInt	W	Defines the maximum data rate for fax. Enumeration of the following values: ▪ 2400 ▪ 4800 ▪ 7200 ▪ 9600 ▪ 12000 ▪ 14400 ▪ 33600	mapped to FaxRelayMaxRate
HighSpeedPacketRate	unsignedInt	W	Defines the rate at which high speed data will be sent across the network, in milliseconds. Enumeration of the following values: ▪ 10 ▪ 20 ▪ 30	Mapped to FaxModemBypassBasicRTPPacketInterval

Name	Type	Write	Description	Comment
			▪ 40	
HighSpeedRedundancy	unsignedInt [0:3]	W	Specifies the packet-level redundancy for high-speed data transmissions (i.e., T.4 image data). The value MUST be in the range 0 through 3.	FaxRelayRedundancyDepth
LowSpeedRedundancy	unsignedInt [0:5]	W	Specifies the packet-level redundancy for low-speed data transmissions (i.e., T.30 handshaking information). The value MUST be in the range 0 through 5.	FaxRelayEnhancedRedundancyDepth
.VoiceService.{i}.VoiceProfile.{i}.Line.{i}.	object	C	Defines support for creation and deletion of Lines is REQUIRED only if more than one Line is supported as indicated by VoiceService.{i}.Capabilities.MaxLineCount.	
DirectoryNumber	string(32)	W	Defines the directory number associated with this line. May be used to identify the line to the user. In case of H.323 signaling, this MUST be an E.164 number.	TrunkGroup_FirstPhoneNumber (from corresponding entry in Hun Group table)
.VoiceService.{i}.VoiceProfile.{i}.Line.{i}.CallingFeatures.	object	–	Defines Voice line parameters related to optional endpoint based calling features.	
CallerIDNameEnable	boolean	W	Enables or disables the transmission of caller ID information on outgoing calls.	This and the next parameters will be mapped to Caller Display Information table (CallerDisplayInfo) CallerDisplayInfo_IsCidRestricted
CallerIDName	string(256)	W	Defines a string used to identify the caller.	CallerDisplayInfo_Displaystring
.VoiceService.{i}.VoiceProfile.{i}.Line.{i}.VoiceProcessing	object	-	Voice line parameters related to voice processing capabilities.	
TransmitGain	int	W	Gain in 01. dB to apply to the transmitted voice signal prior to encoding. This gain is a modifier of the default transmit-gain, which is unspecified.	
ReceiveGain	int	W	Gain in 0.1 dB to apply to the received voice signal after decoding. This gain is a modifier of	

Name	Type	Write	Description	Comment
			the default receive-gain, which is unspecified.	
EchoCancellationEnable	boolean	W	Enables or disables echo cancellation for this line.	
EchoCancellationInUse	boolean	-	Indication of whether or not echo cancellation is currently in use for this line.	
EchoCancellationTail	unsignedInt	-	Tail length in milliseconds of the echo canceller associated with this line (whether or not it is currently in use).	

6.4   SIP Endpoint Profile

MSBR complies with Version 1 of SIP Endpoint profile (SIPEndpoint:1) as defined in TR-104 section 4.3.

Table 6-3: SIP Endpoint Profile Table

Name	Type	Write	Description	Comment
.VoiceService.{i}.Capabilities.SIP.	object	–	Defines SIP-specific capabilities. Applicable only if SIP is among the list of supported protocols.	
Role	string	–	Defines the role of this VoIP CPE. Enumeration of: "UserAgent" "BackToBackUserAgents" "OutboundProxy" A single VoiceService instance MUST have only one role. If a device includes the capabilities for more than one role, each role MUST be represented as separate VoiceService instances.	Always set to "UserAgent"
Extensions	string(256)	–	Defines a comma-separated list of SIP extension methods supported. SIP extension methods MUST be in the form of the method name in upper case. The list MAY include vendor-specific extensions, which MUST use the format defined in [3]. Examples: ▪ "REFER" ▪ "INFO" ▪ "X_EXAMPLE-COM_MyExt"	"REFER, INFO" SUBSCRIBE , etc..
Transports	string(256)	–	Defines a comma-separated list of SIP transport protocols supported. Each entry is an enumeration of: ▪ "UDP" ▪ "TCP" ▪ "TLS" ▪ "SCTP" The list MAY include vendor-specific transports, which MUST use the format defined in [3].	According to SIPTransportType UDP/TCP/TLS

Name	Type	Write	Description	Comment
URISchemes	string(256)	–	Defines a comma-separated list of URI schemes supported beyond the URI schemes required by the SIP specification. Each URI scheme is given by the URI prefix, without the colon separator. Example: ▪ "tel, fax"	"" (empty)
EventSubscription	boolean	–	Defines support for SIP event subscription. A true value indicates support for the VoiceService.{i}.VoiceProfile.{i}.SIP.EventSubscribe and VoiceService.{i}.VoiceProfile.{i}.Line.{i}.SIP.EventSubscribe.{i} objects.	False
ResponseMap	boolean	–	Defines support for SIP response map. A true value indicates support for the VoiceService.{i}.VoiceProfile.{i}.SIP.ResponseMap object. This parameter is applicable only for a VoIP endpoint.	False
.VoiceService.{i}.VoiceProfile.{i}.SIP.	object	–	Defines Voice profile parameters that are specific to SIP user agents.	
ProxyServer	string(256)	W	Defines a Host name or IP address of the SIP proxy server. All SIP signaling traffic MUST be sent to the host indicated by this parameter and the port indicated by the ProxyServerPort parameter unless OutboundProxy parameter is non-empty or a different route was discovered during normal operations SIP routing operation. Regardless of which host the traffic gets sent to (the ProxyServer or the OutboundProxy), the value of this parameter MUST be used to derive the URI placed into the SIP Route header field of all requests originated by this end-point unless a different proxy host	ProxyIp_IpAddress from the 1st entry in "default" Proxy Set (0); note that ProxyIp_IpAddress contains _both_ hostname and port – hence some manipulation is needed. During "write" operation ensure that IsProxySet parameter is set to 1). During "read" operation, if IsProxySet parameter is set to 0, return empty string.

Name	Type	Write	Description	Comment
			was discovered dynamically during normal SIP routing operations.	
ProxyServerPort	unsignedI nt [0:65535]	W	Defines the Destination Port to be used in connecting to the SIP server.	ProxyIp_IpAddress from the 1st entry in "default" Proxy Set (0); note that ProxyIp_IpAddress contains _both_ hostname and port – hence some manipulation is needed.
ProxyServerTranspo rt	string	W	Defines the Transport protocol to be used in connecting to the SIP server. Must be chosen from among the transports supported, as indicated by VoiceService.{i}.Capabilities. SIP.Transports. Enumeration of: ▪ "UDP" ▪ "TCP" ▪ "TLS" ▪ "SCTP"	According to ProxyIp_Transport Type from the 1st entry in "default" Proxy Set (0
RegistrarServer	string(256)	W	Defines a Host name or IP address of the SIP registrar server. If this parameter is empty, the CPE MUST obtain all of the registrar server configuration information, including host name or IP address, port, and transport protocol, from the corresponding ProxyServer parameters (ProxyServer, ProxyServerPort, and ProxyServerTransport), ignoring all of the registrar server parameters (RegistrarServer, RegistrarServerPort and RegistrarServerTransport).	RegistrarIP **Note:** RegistrarIP may contain both hostbname and port – hence some manipulation is needed.
RegistrarServerPort	unsignedI nt [0:65535]	W	Defines the Destination port to be used in connecting to the SIP registrar server. If the RegistrarServer parameter is empty, the CPE MUST obtain all of the registrar server configuration information, including host name or IP address, port,	RegistrarIP **Note:** RegistrarIP may contain both hostbname and port – hence some manipulation is needed

Name	Type	Write	Description	Comment
			and transport protocol, from the corresponding ProxyServer parameters (ProxyServer, ProxyServerPort, and ProxyServerTransport), ignoring all of the registrar server parameters (RegistrarServer, RegistrarServerPort and RegistrarServerTransport).	
RegistrarServerTransport	string	W	Defines the Transport protocol to be used in connecting to the registrar server. Must be chosen from among the transports supported, as indicated by VoiceService.{i}.Capabilities.SIP.Transports. Enumeration of: "UDP" "TCP" "TLS" "SCTP" If the RegistrarServer parameter is empty, the CPE MUST obtain all of the registrar server configuration information, including host name or IP address, port, and transport protocol, from the corresponding ProxyServer parameters (ProxyServer, ProxyServerPort, and ProxyServerTransport), ignoring all of the registrar server parameters (RegistrarServer, RegistrarServerPort and RegistrarServerTransport).	According to RegistrarTransportType
UserAgentDomain	string(256)	W	Defines a CPE domain string. If empty, the CPE SHOULD use its IP address as the domain.	ProxyName
UserAgentPort	unsignedInt [0:65535]	W	Defines a port used for incoming call control signaling.	LocalSIPPort or TCPLocalSIPPort or TLSLocalSIPPort – according to SIPTransportType
UserAgentTransport	string	W	Defines the Transport protocol to be used for	SIPTransportType

Name	Type	Write	Description	Comment
			incoming call control signaling. Must be chosen from among the transports supported, as indicated by VoiceService.{i}.Capabilities. SIP.Transports. Enumeration of: ▪ "UDP" ▪ "TCP" ▪ "TLS" ▪ "SCTP"	
OutboundProxy	string(256)	W	Defines the Host name or IP address of the outbound proxy. If a non-empty value is specified, the SIP endpoint MUST send all SIP traffic (requests and responses) to the host indicated by this parameter and the port indicated by the OutboundProxyPort parameter. This MUST be done regardless of the routes discovered using normal SIP operations, including use of Route headers initialized from Service-Route and Record-Route headers previously received. The OutboundProxy value is NOT used to generate the URI placed into the Route header of any requests.	Outbound Proxy will be mapped to the 1st entry in PREFIX table that (if exists) must look as follows: [PREFIX] FORMAT PREFIX_Index = PREFIX_DestinationPrefix, PREFIX_DestAddress, PREFIX_SourcePrefix, PREFIX_ProfileId, PREFIX_MeteringCode, PREFIX_DestPort, PREFIX_SrcIPGroupID, PREFIX_DestHostPrefix, PREFIX_DestIPGroupID, PREFIX_SrcHostPrefix, PREFIX_TransportType, PREFIX_SrcTrunkGroupID; PREFIX 0 = *, 10.8.211.180, *, 0, 255, 0, -1, , -1, , -1, -1; [\PREFIX]
OutboundProxyPort	unsignedInt [0:65535]	W	Defines the Destination port to be used in connecting to the outbound proxy. This parameter MUST be ignored unless the value of the OutboundProxy parameter in this object is non-empty.	PREFIX_DestPort (see above)

Name	Type	Write	Description	Comment
RegistrationPeriod	unsignedInt [1:]	W	Defines the Period over which the user agent must periodically register, in seconds.	RegistrationTime
RegisterExpires	unsignedInt [1:]	W	Defines the Register Request Expires header value, in seconds.	RegistrationTime (the same value as RegistrationPeriod)
UseCodecPriorityIn SDPResponse	Boolean	W	When true, in the SDP included in an OK response to an Invite, the first listed codec MUST be the highest priority codec among those offered in the Invite, based on the priorities specified in VoiceProfile.{i}.Line.{i}.Codec.List.{i}. The list of codecs in the SDP MAY also include other lower priority codecs. When false, there is no specific requirement for choosing the codecs listed in the SDP included in an OK response.	
.**VoiceService.{i}.VoiceProfile.{i}.Line.{i}.SIP.**	object	–	Defines the Voice line parameters that are specific to SIP call signaling.	
AuthUserName	string(128)	W	Defines the Username used to authenticate the connection to the server.	This and the next parameter will be mapped to the corresponding entry in Authentication table. Authentication_UserId On write, we will ensure that AuthenticationMode is set to 3 (PerFXS). On read, if AuthenticationMode is not set to 3 (PerFXS), empty string will be returned.
AuthPassword	string(128)	W	Defines the Password used to authenticate the connection to the server. When read, this parameter returns an empty string,	Authentication_UserPassword On write, we will ensure that AuthenticationMod

Name	Type	Write	Description	Comment
			regardless of the actual value.	e is set to 3 (PerFXS). On read, if AuthenticationMode is not set to 3 (PerFXS), empty string will be returned.

This page is intentionally left blank.

International Headquarters
1 Hayarden Street,
Airport City
Lod 7019900, Israel
Tel: +972-3-976-4000
Fax: +972-3-976-4040

AudioCodes Inc.
80 Kingsbridge Rd
Piscataway, NJ 08854, USA
Tel: +1-732-469-0880
Fax: +1-732-469-2298

Contact us: https://www.audiocodes.com/corporate/offices-worldwide
Website: https://www.audiocodes.com

Document #: LTRT-52357