|  |
| --- |
| TERAMATRIX TECHNOLOGIES |
| **R&D Document** |
| **TR-069** |
|  |

|  |
| --- |
|  |

# 

# Proprietary Information

The information contained herein is proprietary to Teramatrix Technologies Private Limited. Use or disclosure of this document or the information contained herein, for any purpose other than that for which it was furnished is not permitted or it shall not be disclosed or divulged to any third Party without the prior written consent of Teramatrix Technologies Private Limited.

# Copyright Notice

This document is copyright © Teramatrix Technologies Private Limited. All Rights Reserved. No part of this document, in whole or in part, may be used, reproduced, stored in a retrieval system or transmitted, in any form, or by any means, electronic or otherwise, including photocopying, reprinting, or recording, for any purpose, without the express written permission of Teramatrix Technologies Private Limited

# About Teramatrix

Teramatrix Technologies create E2E (end-to-end) Systems and Solutions to help clients address performance conflicts and process optimization by aggregation, analyses and visualization of the information in most simple and innovative manner.

Teramatrix create solutions to address some of the most challenging problems of the clients, spanning from management of large scale networks to analysis of millions of business events in real-time.

Teramatrix has successfully served sectors of Telecommunication, Security Surveillance and Manufacturing.

In a short span, company has created a significant knowledge base and experience in the domains of large scale Network Monitoring, Surveillance Systems, Business Intelligence and Product Engineering Services.

**Table of Contents**

[Proprietary Information 2](#_Toc432174845)

[Copyright Notice 2](#_Toc432174846)

[About Teramatrix 3](#_Toc432174847)

[Introduction 5](#_Toc432174848)

[Why use TR069? 5](#_Toc432174849)

[What devices use TR-069? 6](#_Toc432174850)

[TR-069 Functional Components: 6](#_Toc432174851)

[Protocol Components: 7](#_Toc432174852)

[Session Example: 8](#_Toc432174853)

[RPC List in CWMP: 9](#_Toc432174854)

[1.1 CPE Method: 9](#_Toc432174855)

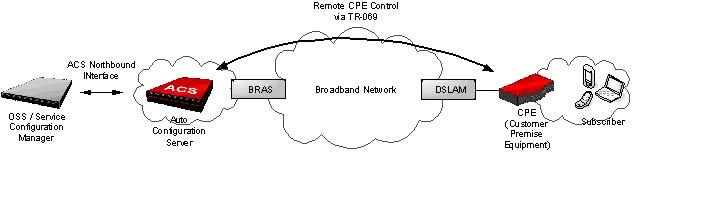
[1.2 ACS Method: 9](#_Toc432174856)

[Events: 13](#_Toc432174857)

[GenieACS Installation: 17](#_Toc432174858)

# Introduction

TR-069 stands for a technical report defined by the Broadband Forum that specifies the CWMP “CPE WAN Management Protocol”. It provides the communication between [customer-premises equipment](https://en.wikipedia.org/wiki/Customer-premises_equipment) (CPE) and Auto Configuration Servers (ACS). It includes both a safe auto configuration and the control of other CPE management functions within an integrated framework. The protocol addresses the growing number of different [Internet access](https://en.wikipedia.org/wiki/Internet_access) devices such as [modems](https://en.wikipedia.org/wiki/Modem), [routers](https://en.wikipedia.org/wiki/Router_(computing)), [gateways](https://en.wikipedia.org/wiki/Gateway_(computer_networking)), as well as end-user devices which connect to the Internet, such as [set-top boxes](https://en.wikipedia.org/wiki/Set-top_box), and [VoIP](https://en.wikipedia.org/wiki/VoIP)-[phones](https://en.wikipedia.org/wiki/Phone). The TR-069 standard was developed for automatic configuration and management of these devices by Auto Configuration Servers (ACS).  It commonly uses HTTP or HTTPS as transport for communication between CPE and the ACS. The message exchange is using SOAP (XML\_RPC) for configuration and management of the device.



# Why use TR069?

TR-069 is an application layer protocol, which has broad applicability and no access restriction. TR-69 standard allows the subscriber to manage all devices on a common platform regardless of its device type and manufacturer. Its specifications ensure that the device can be easily and securely configured, activated and managed from a console in the service provider's network. This allows the service provider to provide an efficient and cost effective deployment of services.

There are many benefits associated with using TR-069 to provision and manage end-user devices. TR-069 and its extensions allow to :

* Enable remote provisioning of CPE
* Better manage broadband networks with increased visibility and control of CPE
* Collect data for analytics on network usage and activity, home network characteristics, and service utilization
* Deliver new managed data services such as WiFi, content filtering and other parental controls, online backup, and home surveillance
* Offer subscribers a degree of self-service through web portals
* Expand service offerings and manage the connected home
* Improve your customer service with improved diagnostics, monitoring, and firmware management
* Reduce support calls and remove the burden of CPE configuration from subscribers

# What devices use TR-069?

TR-069 offers management capabilities for a wide range of devices including wireline and cable residential gateways, fiberoptical network terminals, IPTV set-top boxes, network attached storage, Home Plug adapters, IP phones, and more. More recently, the cable industry has seen the emergence of multimedia residential gateways that utilizes traditional DOCSIS provisioning along with TR-069 to manage value-added features such as gateway configuration, VoIP, WiFi, and IPTV set-top box services.

# **TR-069 Functional Components**:

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

• Status and performance monitoring

• Diagnostics

1. **Auto-Configuration and Dynamic Service Provisioning:**

The CPE WAN Management Protocol allows an ACS to provision a CPE or collection of CPE based on a variety of criteria. The provisioning mechanism allows CPE provisioning at the time of initial connection to the broadband access network, and the ability to re-provision or re-configure at any subsequent time.

2) **Software/firmware image management**:

The CPE WAN Management Protocol provides tools to manage downloading of CPE software/firmware image files. The protocol provides mechanisms for version identification, file download initiation (ACS initiated downloads and optional CPE initiated downloads), and notification of the ACS of the success or failure of a file download.

3). **Software Module Management**:

The CPE WAN Management Protocol enables an ACS to manage modular software and execution environments on a CPE. Capabilities provided include the ability to install, update, and uninstall software modules as well as notification to the ACS of success or failure of each action.

4). **Status and Performance Monitoring:**

The CPE WAN Management Protocol provides support for a CPE to make available information that the ACS may use to monitor the CPE’s status and performance statistics. It also defines a set of mechanisms that allow the CPE to actively notify the ACS of changes to its state.

5). **Diagnostics:**

The CPE WAN Management Protocol provides support for a CPE to make available information that the ACS may use to diagnose and resolve connectivity or service issues as well as the ability to execute defined diagnostic tests.

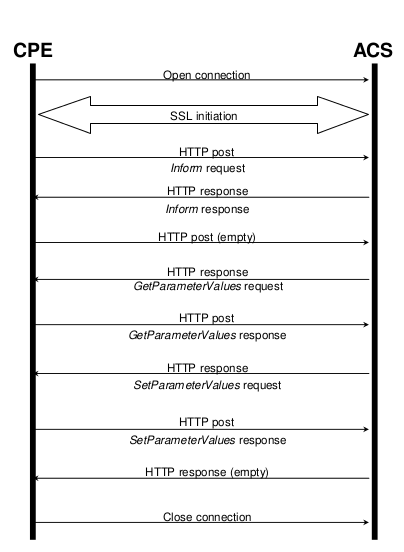
# Protocol Components:

The CPE WAN Management Protocol comprises several components that are unique to this protocol, and makes use of several standard protocols. The protocol stack defined by the CPE WAN Management Protocol.

**Protocol Stack**

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

# Session Example:

****

# RPC Method in CWMP:

There is following method for CWMP (ACS and CPE)

## CPE Method:

* GetRPCMethods
* SetParameterValues
* GetParameterValues
* GetParameterNames
* SetParameterAttributes
* GetParameterAttributes
* AddObject
* DeleteObject
* Reboot
* Download
* Upload
* FactoryReset
* GetQueuedTransfers
* GetAllQueuedTransfers
* ScheduleInform
* SetVouchers
* GetOptions

## ACS Method:

* GetRPCMethods
* Inform
* TransferComplete
* AutonomousTransferComplete
* RequestDownload
* Kicked

**GetRPCMethods**

This method may be used a CPE or Server to discover the set of methods supported by the Server or CPE it is in communication with. This list may include both standard methods (those defined in this specification or a subsequent version) and vender-specific method. The receiver of the response must ignore any unrecognized method.

**SetParameterValues**

This method may be used by Server to modify the value of one or more CPE parameters**.**

**GetParameterValues**

This method is used to obtain the value of one or more CPE parameters.

**GetParameterNames**

This method is used to discover the parameters accessible on a particular CPE.

**GetParameterAttributes**

This method is used to read the attributes associated with one or more CPE parameters.

**SetParameterAttributes**

This method is used to modify attributes associated with one or more CPE parameters.

**AddObject**

This method MAY be used by the ACS to create a new instance of a multi-instance

Object a collection of Parameters and/or other objects for which multiple instances are defined. The method call takes as an argument the path name of the collection of objects for which a new instance is to be created. For example:

Top.Group.Object.

This path name does not include an instance number for the object to be created. That instance number is assigned by the CPE and returned in the response. Once assigned the instance number of an object cannot be changed and persists until the object is deleted using the Delete Object method. After creation, Parameters or sub-objects within the object are referenced by the path name appended with the instance number. For example, if the Add Object method returned an instance number of 2, a Parameter within this instance can then be referred to by the path:

Top.Group.Object.2.Parameter

On creation of an object using this method, the Parameters contained within the object

MUST be set to their default values and the associated attributes MUST be set to the

following:

• Notification is set to zero (notification off) unless otherwise specified in the

appropriate data model definition

• Access List includes all defined entities

After doing this operation, it should the modified configuration into the flash

**DeleteObject**

This method is used to remove a particular instance of an object. This method call takes as an argument the path name of the object instance including the instance number. For example:

Top.Group.Object.2.

If this method call is successful, the specified instance of this object is subsequently

unavailable for access and the CPE must discard the state previously associated with all Parameters (values and attributes) and sub-objects contained within this instance.

When an object instance is deleted, the instance numbers associated with any other

instances of the same collection of objects remain unchanged. Thus, the instance

numbers of object instances in a collection might not be consecutive.

**Reboot**

This method causes the CPE to reboot.

**Download**

This method is used to cause the CPE to download a specified file from the designated location**.**

**Upload**

This method is used to cause the CPE to upload a specified file to the designated location.

**FactoryReset**

This method resets the CPE to its factory default state. This method should be used with extreme caution.

**GetQueuedTransfers**

This method may be used by an ACS to determine the status of previously requested

downloads or uploads.

**GetAllQueuedTransfers**

This method may be used by an ACS to determine the status of all queued downloads and uploads, including any that were not specifically requested by the ACS, i.e. autonomous transfers.

**ScheduleInform**

This method may be used by an ACS to request the CPE to schedule a one-time Inform

method call (separate from its periodic Inform method calls) sometime in the future.

**SetVouchers**

This method may be used by an ACS to set one or more option Vouchers in the CPE.

**GetOptions**

This method may be used by an ACS to obtain a list of the options currently set in a

CPE, and their associated state information.

**Inform**

A CPE must call the Inform method to initiate a transaction sequence whenever a

session with an ACS is established.

**TransferComplete**

This method informs the ACS of the completion (either successful or unsuccessful) of a

file transfer initiated by an earlier Download, Schedule Download or Upload method call.

This must be called only when the associated Download or upload response indicated that the transfer had not yet completed at that time.

When used this method should be called only the transfer has completed (or failed).The criteria used by a CPE to determine when a transfer is considered complete are specific to the implementation of the CPE.

**AutonomousTransferComplete**

This method informs the ACS of the completion (either successful or unsuccessful) of a

file transfer that was not specifically requested by the ACS. When used, this method

must be called only after the transfer has successfully completed, and in the case of a

download, the downloaded file has been successfully applied, or after the transfer or

apply has failed (e.g. a timeout expired). If this method fails, the CPE must not

regard the ACS as having been informed of the completion of the file transfer, and

must attempt to call the method again, either in the current session or in a new session.

**RequestDownload**

This method allows the CPE to request a file download from the ACS. On reception of

this request, the ACS MAY call the Download method to initiate the download.

**Kicked**

The CPE calls this method whenever the CPE is “kicked”.

# Events:

Event requires the CPE to notify the ACS via an Inform request

**Event Types:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Event Code** | **Cumulative**  **Behaviour** | **Explanation** | **ACS Response for**  **Successful Delivery** | **Retry/Discard**  **Policy** |
| "0BOOTSTRAP" | Single | Indicates that the session was established due to first-time CPE installation or a change to the ACS  URL. The specific conditions that MUST  result in the BOOTSTRAP  Event Code are:  • First time connection of the CPE  to the ACS from the factory.  • First time connection of the CPE  to the ACS after a factory reset.  • First time connection of the CPE  to the ACS after the ACS URL  has been modified in any way | Inform Response | The CPE must  not ever discard  an undelivered  BOOTSTRAP event. All other  Undelivered events must be discarded on BOOTSTRAP. |
| "1 BOOT" | Single | Indicates that the session was established due to the CPE being  powered up or reset. This includes initial system boot, as well as reboot due to any cause, including use of  the Reboot method. | Inform Response | The CPE must  retry delivery until it reboots before  discarding it. |
| "2 PERIODIC" | Single | Indicates that the session was established on a periodic Inform  interval. | Inform Response | The CPE must  not ever discard  an undelivered  PERIODIC event. |
| "3 SCHEDULED" | Single | Indicates that the session was established due to a Schedule Inform method call. This event code MUST only be used with the “M Schedule Inform” event  code | Inform Response | The CPE must  not ever discard  an undelivered  SCHEDULED  event. |
| "4 VALUE  CHANGE" | Single | Indicates that since the last successful Inform ,the value of one or more parameters with Passive or Active Notification enabled  has been modified  If this Event Code is included in the Event array, all such modified  parameters MUST be included in the Parameter List in this Inform. If this event is ever discarded then the list of modified parameters MUST  be discarded at the same time. | Inform Response | The CPE must  retry delivery until it reboots or the ACS URL is modified before discarding it. |
| "5 KICKED" | Single | Indicates that the session was established for the purpose of web  identity management  will be called one or  more times during this  session. | Kicked Response | The CPE may  retry delivery at its  discretion |
| “6  CONNECTION  REQUEST”” | Single | Indicates that the session was established due to a Connection Request from the ACS | Inform Response | The CPE must  not retry delivery |
| “7 TRANSFER  COMPLETE” | Single | indicates that the session was established to indicate the completion of a previously requested download or upload | Transfer Complete  Response | The CPE must  not ever discard  an undelivered  TRANSFER  COMPLETE event |
| "8  DIAGNOSTIC  COMPLETE" | Single | Used when re-establishing a  connection to the ACS after completing one or more diagnostic test initiated by the ACS. | Inform Response | The CPE must  retry delivery until it reboots before  discarding it. |
| “9 REQUEST  DOWNLOAD” | Single | Indicates that the session was established for the CPE to call the Request Download method | Request  Download  Response | The CPE MAY  retry delivery at its  discretion. |
| “10  AUTONOMOUS  TRANSFER  COMPLETE” | Single | Indicates that the session was  established to indicate the completion of a download or upload that was not specifically requested by the ACS and that the Autonomous Transfer Complete  method will be called one or more times during this session | Autonomous  Transfer-  Complete Response | The CPE MUST  NOT ever discard  an undelivered  AUTONOMOUS  TRANSFER  COMPLETE event |
| “M Reboot” | Multiple | The CPE rebooted upon request  from the ACS through the use of the Reboot RPC. Overlaps with one of the causes that can generate a “1 BOOT” event code. | Inform Response | The CPE MUST  NOT ever discard  an undelivered “M Reboot” event. |
| “M  Schedule Inform” | Multiple | The ACS requested a scheduled  Inform | Inform Response | The CPE MUST  NOT ever discard  an undelivered “M Schedule Inform”  event |
| “M Download” | Multiple | A content download previously  requested by the ACS using the  Download method | Transfer Complete  Response | The CPE MUST  NOT ever discard  an undelivered “M Download” event. |
| “M Upload” | Multiple | A content upload previously  requested by the ACS using the  Upload method | Transfer  Complete Response | The CPE MUST  NOT ever discard  an undelivered “M Upload” event. |
| "M " <vendor-  specific method> | Not  specified | Vendor-specific event. The OUI  after the “X“ and space is an  organizationally unique identifier  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 [9], and MUST be one that is assigned to the organization that defined this vendor-specific event. The value and interpretation of <event> is vendor-specific.  For example:  “X 012345 MyEvent” | Not specified | Not specified |
| “X “<OUI> ” ”  <event> | Not  specified | Vendor-specific event. The OUI  after the “X“ and space is an  organizationally unique identifier  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 [9], and MUST be  one that is assigned to the  organization that defined this  vendor-specific event. The value  and interpretation of <event> is  vendor-specific.  For example:  “X 012345 MyEvent” | Not specified | Not specified |

# GenieACS Installation:

Installation process for GenieACS use the centos 6.5 to OS .

First, install the Ruby execution environment and Radis. Install the required packages and YAML earlier

[Root @ genie ~] # yum -y install gcc openssl-devel zlib-devel readline-devel sqlite-devel   
  
 [Root @ genie ~] # wget http://pyyaml.org/download/libyaml/yaml-0.1.6.tar.gz   
 [Root @ genie ~] # tar xvzf yaml-0.1.6.tar.gz   
 [Root @ genie yaml-0.1.6] # ./configure   
 [Root @ genie yaml-0.1.6] # make && make install

Build a Ruby from source code.

[Root @ genie ~] # wget http://cache.ruby-lang.org/pub/ruby/2.1/ruby-2.1.2.tar.gz   
 [Root @ genie ~] # tar xvzf ruby-2.1.2.tar.gz   
 [Root @ genie ruby-2.1.2] # ./configure   
 [Root @ genie ruby-2.1.2] # make && make install   
  
 [Root @ genie ruby-2.1.2] # ruby -v   
 ruby 2.1.2p95 (2014-05-08 revision 45877)

It concludes After you install the Rails and bundle. By the way, Ruby, Rails is not necessary if you own to UI development in the UI application execution environment.

[Root @ genie ~] # gem install rails --no-ri --no-rdoc   
 [Root @ genie ~] # gem install bundle --no-ri --no-rdoc

Node.js runtime environment also build and install from source code.

[Root @ genie ~] # wget http://nodejs.org/dist/v0.10.28/node-v0.10.28.tar.gz   
 [Root @ genie ~] # tar xvzf node-v0.10.28.tar.gz   
  
 [Root @ genie node-v0.10.28] # yum -y install g ++   
 [Root @ genie node-v0.10.28] # ./configure   
 [Root @ genie node-v0.10.28] # make && make install   
  
 [Root @ genie node-v0.10.28] # node -v   
 v0.10.28

Followed by setting up of storage systems. First Redis to be used for cache.

[Root @ genie ~] # wget http://download.redis.io/releases/redis-2.8.9.tar.gz   
 [Root @ genie ~] # tar xvzf redis-2.8.9.tar.gz   
  
 [Root @ genie redis-2.8.9] # yum -y install tcl   
 [Root @ genie redis-2.8.9] # make   
 [Root @ genie redis-2.8.9] # make test   
  
 \ O / All tests passed without errors!   
  
 Cleanup: it may take some time ... OK   
  
 [Root @ genie redis-2.8.9] # make install   
  
 # Start redis server.   
 [Root @ genie redis-2.8.9] # redis-server

Database MongoDB. Such as registered CPE (CWMP client) of information and uploaded firmware file is saved in MongoDB.

First download mongodb-linux-x86\_64-rhel62-3.0.6.tgz tar file

[Root @ genie ~] # tar mongodb-linux-x86\_64-rhel62-3.0.6.tgz  
 [Root @ genie ~] # mkdir -p / data / db   
  
 # Start MongoDB server shutdown:. ./bin/mongod --shutdown   
 [Root @ genie mongodb-linux-x86\_64-rhel62-3.0.6] # ./bin/mongod --fork --syslog   
 Fri May 30 21: 15: 37.394   
 Fri May 30 21: 15: 37.395   
 about to fork child process, waiting until server is ready for connections.   
 forked process: 20880   
 using syslog ident: mongod.27017   
 child process started successfully, parent exiting

Install the GenieACS engine and Web UI in the next procedure.

# Install GenieACS Core (North Bound Interface, CWMP Interface, File Server)   
 [Root @ genie opt] # yum -y install git   
 [Root @ genie opt] # git clone https://github.com/zaidka/genieacs.git   
 [Root @ genie opt] # cd genieacs /   
 [Root @ genie genieacs] # npm install   
 [Root @ genie genieacs] # npm run configure   
 [Root @ genie genieacs] # npm run compile   
  
 # Install Web UI   
 [Root @ genie opt] # git clone https://github.com/zaidka/genieacs-gui   
 [Root @ genie opt] # cd genieacs-gui /   
 [Root @ genie genieacs-gui] # bundle

Initial operation setting, let's use it to rename the sample in the under / opt / genieacs-gui / config.

[Root @ genie config] # cp index\_parameters-sample.yml index\_parameters.yml   
 [Root @ genie config] # cp parameter\_renderers-sample.yml parameter\_renderers.yml   
 [Root @ genie config] # cp parameters\_edit-sample.yml parameters\_edit.yml   
 [Root @ genie config] # cp roles-sample.yml roles.yml   
 [Root @ genie config] # cp summary\_parameters-sample.yml summary\_parameters.yml   
 [Root @ genie config] # cp users-sample.yml users.yml   
 [Root @ genie config] # cp graphs-sample.json.erb graphs.json.erb

Finally, start the GenieACS service.

# Start North Bound / RESTful Interface service.   
 [Root @ genie ~] # / opt / genieacs / bin / genieacs-nbi   
 30 May 17:39:19 - Worker 26283 listening to 0.0.0.0:7557   
 30 May 17:39:19 - Worker 26284 listening to 0.0.0.0:7557   
  
 # Start ACS / CWMP service.   
 [Root @ genie ~] # / opt / genieacs / bin / genieacs-cwmp   
 30 May 17:36:49 - Worker 26233 listening to 0.0.0.0:7547 # HTTP   
 30 May 17:36:49 - Worker 26233 listening to 0.0.0.0:7548 # HTTPS   
  
 # Start HTTP / File streaming service.   
 [Root @ genie ~] # / opt / genieacs / bin / genieacs-fs   
 2 Jun 16:00:02 - Worker 13083 listening to 0.0.0.0:7567   
 2 Jun 16:00:02 - Worker 13084 listening to 0.0.0.0:7567   
  
 # Start GenieACS / Web UI   
 [Root @ genie ~] # cd / opt / genieacs-gui /   
 [Root @ genie genieacs-gui] # rails server   
 => Booting WEBrick   
 => Rails 4.0.0 application starting in development on http://0.0.0.0:3000   
 => Run `rails server -h` for more startup options   
 => Ctrl-C to shutdown server

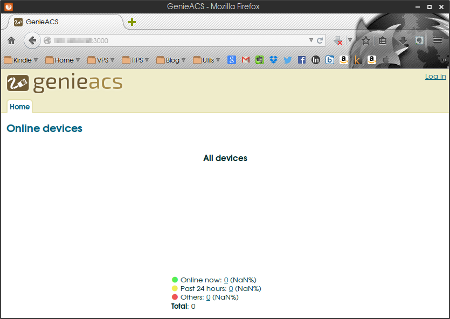
NOTE: -

Install GenieACS for another OS. Please refer the following link.

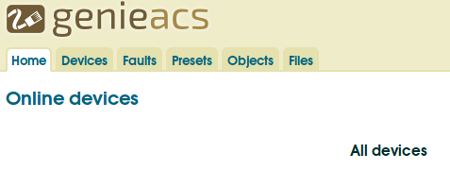
<https://genieacs.com/docs/>

Once you start all services then open any browser and put following url

http: // [Your GenieACS Host]: 3000

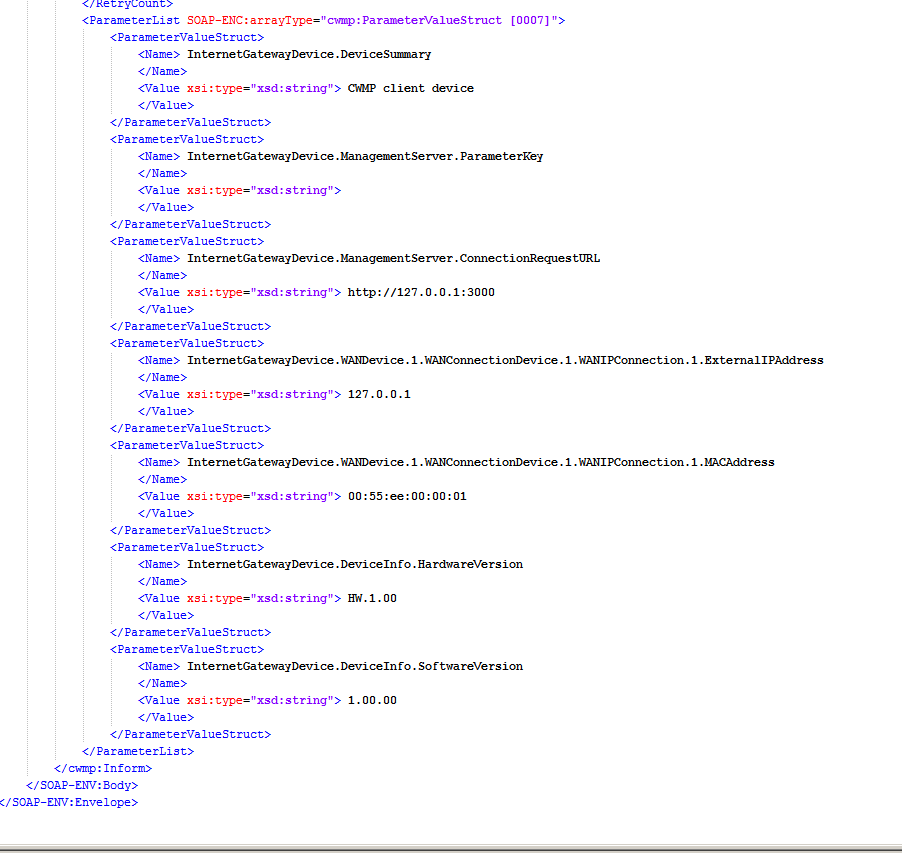


When you log in as admin / admin that is defined in the users.yml Home, Devices, appears Faults, Presets, Object, it is Files total of five menu.



After that write XML file for sends a CWMP message to the ACS, let's register the CPE. First we will prepare a SOAP message (XML file), such as the following.





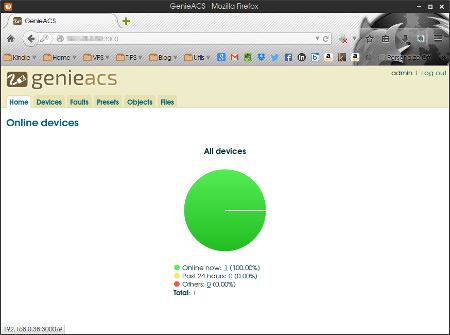
Save this file informBootsBoot.xml and let's HTTP POST to the ACS using the POST command.

(Messages sent from the CPE to the ACS might determine that use of the POST)

linux $ curl -i -X POST http://127.0.0.1:7547 -H "Content-Type: text/xml" --data-binary "@/opt/Bootsboot.xml"

200 OK   
 Connection: keep-alive   
 Date: Mon, 02 Jun 2014 07:12:20 GMT   
 Server: GenieACS / 0.9.9   
 Content-Length: 566   
 Content-Type: text / xml; charset = "utf-8"   
 Client-Date: Mon, 02 Jun 2014 07:12:09 GMT   
 Client-Peer: xxxx: 7547   
 Client-Response-Num: 1   
 Set-Cookie: ZGV2aWNlSWQ = MDA1NWVlLU41MDAwLXNuMTIzNDU2Nzg5MA   
 Set-Cookie: Y3dtcFZlcnNpb24 = MS4w   
 SOAPServer: GenieACS / 0.9.9   
  
 <? Xml version = "1.0" encoding = "UTF-8"?>   
 <Soap-env: Envelope   
  xmlns: soap-enc = "http://schemas.xmlsoap.org/soap/encoding/"   
  xmlns: soap-env = "http://schemas.xmlsoap.org/soap/envelope/"   
  xmlns: xsd = "http://www.w3.org/2001/XMLSchema"   
  xmlns: xsi = "http://www.w3.org/2001/XMLSchema-instance"   
  xmlns: cwmp = "urn: dslforum-org: cwmp-1-0">   
  <Soap-env: Header>   
    <Cwmp: ID soap-env: mustUnderstand = "1"> 0000000001 </ cwmp: ID>   
  </ Soap-env: Header>   
  <Soap-env: Body>   
    <Cwmp: InformResponse>   
      <MaxEnvelopes> 1 </ MaxEnvelopes>   
    </ Cwmp: InformResponse>   
  </ Soap-env: Body>   
 </ Soap-env: Envelope>

It has received a InformResponse from ACS.Now CPE Device has registered in GenieACS UI.



it has been registered. The profile registration of the device that transmitted when View Device information, also makes it possible to control the terminal from GenieACS.



About Teramatrix

Teramatrix Technologies creates Software Solutions to Aggregate, Analyse and Visualize Information in most Simple and innovative manner. ‘Infinite Possibilities’ is an inherent culture at Teramatrix. We believe in constant evolution and strive to deliver best in class solutions through cutting edge technologies.

[**http://www.teramatrix.in**](http://www.teramatrix.in)

**CONTACT US**

|  |  |
| --- | --- |
| B-401, Spaze i-Tech Park  Sec-49, Gurgaon-122018 |  |

**EMAIL:** info@teramatrix.in