Auto Discovery and Configuration of VigorACS

Version 1.0

Summary

This application note describes the auto discovery and configuration functions of VigorACS. This document applies for VigorACS 0.0.1.0 or later.

References

- VigorACS User Guide.
- VigorACS Quick Start Guide.
- TR-069 specification.
- TR-104 specification.

Revision History

Issue	Date	Description
1	September 19, 2008	Initial release for VigorACS 0.0.1.0 or later, by Boham Liu.

Table of Contents

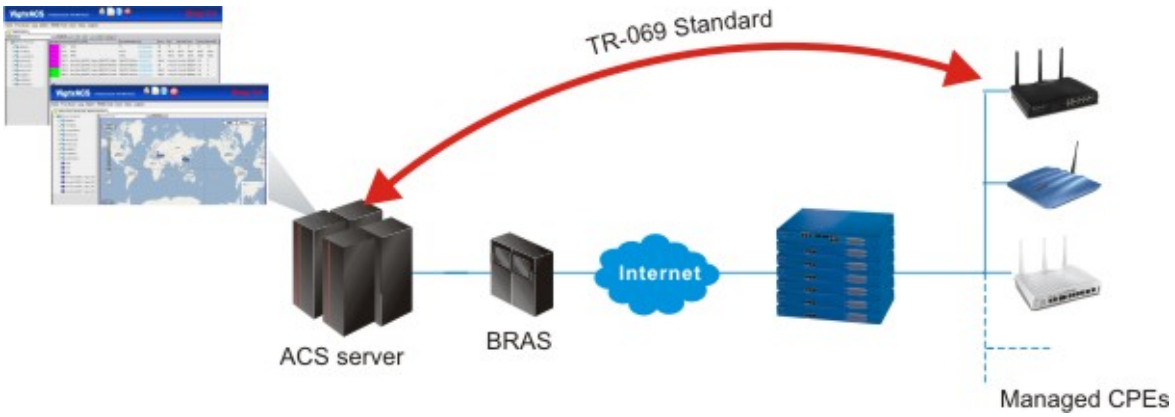
1. Introduction

For a long time, it is a difficult problem to manage all kinds of Internet access devices (CPEs). Since the broadband market boomed, Internet access probabilities grew as well, for example: modems, routers, gateways, Set-top box, VoIP-devices.

It is annoying to set each CPE one by one, especially when the configurations are complicated – too complicated for many users. And try to image this scenario: if you are an IT staff of a company that has 6000 CPEs, each CPE cost 1 minute to setup or configure. One day a new version firmware released, and you have to spend 6000 minutes for upgrading those devices! Thus a centralized management system is required.

VigorACS provides centralized devices management for TR-069 based CPEs, such as broadband gateway, VPN, xDSL router, VoIP gateway, and Wireless AP.

TR-069 is a DSL Forum technical specification entitled CPE WAN Management Protocol (CWMP). It defines an application layer protocol for remote management of end-user devices. The protocol allows VigorACS retrieve CPEs information, provision and manage CPEs. Most of CPE vendors support TR-069, including Draytek.



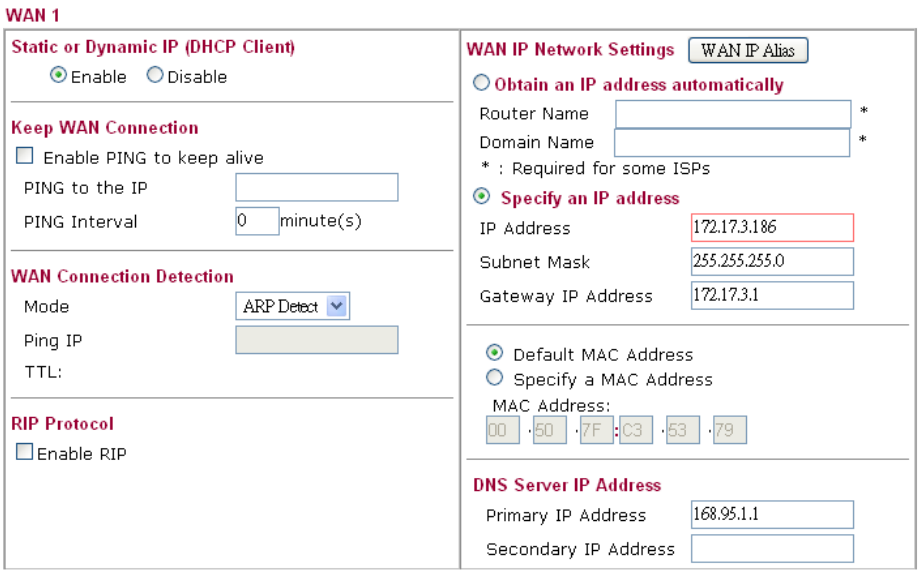
2. What is the auto discovery function of VigorACS

2.1 Auto discovery function

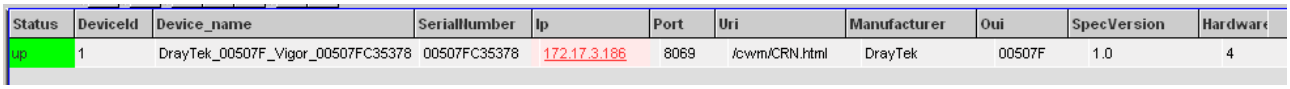
Some centralized management system need user to offer the information of the connected CPEs(ex. IP address) to connect to CPEs. VigorACS is capable of discovering and retrieve CPEs information automatically. Users don't have to offer any data of the new added CPE, once the CPE

set URL of VigorACS and initiate connection to VigorACS, its information would be set to VigorACS.

For example : A CPE was set its IP to 172.17.3.186

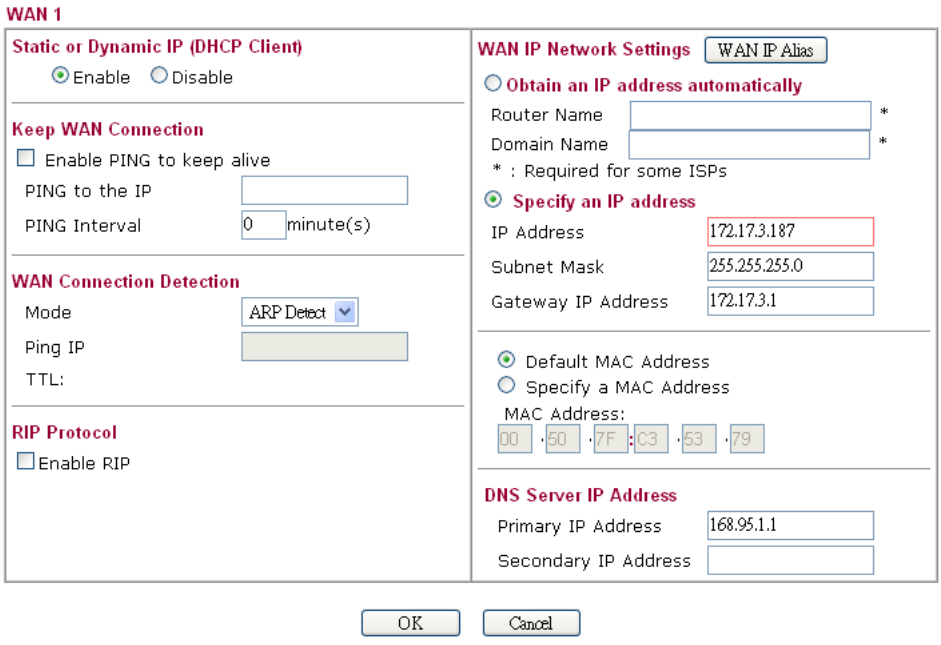


Check the information of the CPE on VigorACS .IP of the CPE was discovered.



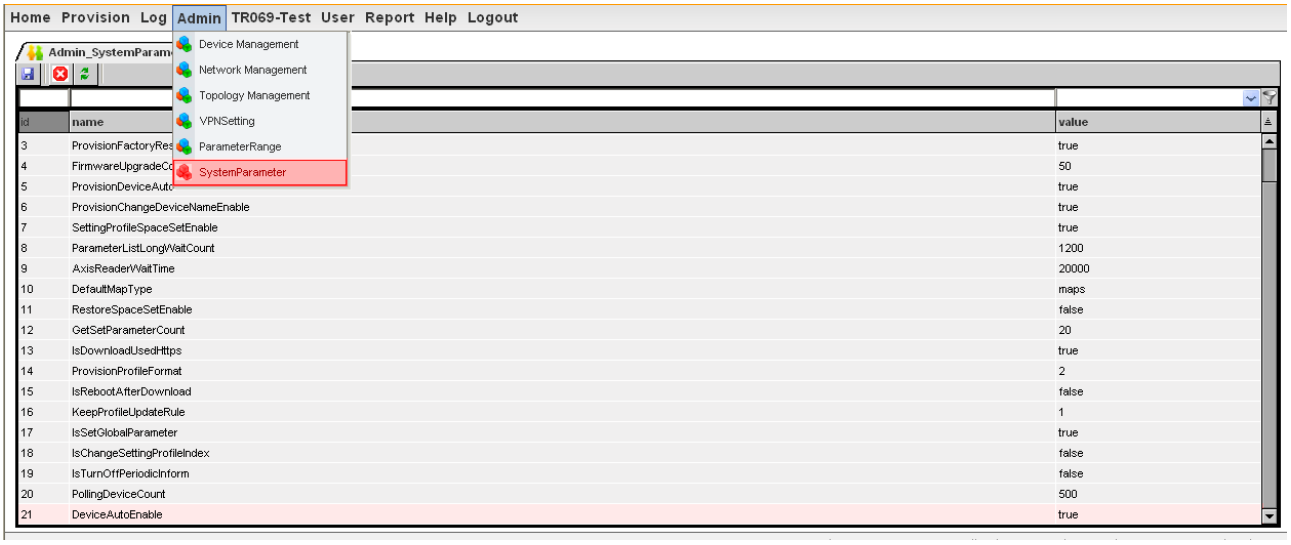
Status	DeviceId	Device_name	SerialNumber	Ip	Port	Uri	Manufacturer	Oui	SpecVersion	Hardware
up	1	DrayTek_00507F_Vigor_00507FC35378	00507FC35378	172.17.3.186	8069	/cwm/CRN.html	DrayTek	00507F	1.0	4

Now, change IP of the CPE to 172.17.3.187. After rebooting of the CPE, check information of the CPE on VigorACS, the newly assigned IP was discovered.



Status	DeviceId	Device_name	SerialNumber	Ip	Port	Uri	Manufacturer	Oui	SpecVersion	Hardware
up	1	DrayTek_00507F_Vigor_00507FC35378	00507FC35378	172.17.3.187	8069	/cwm/CRN.html	DrayTek	00507F	1.0	4

Normally, the discovered CPE would be showed in **Table View** (the default logging page, or click **Home→Table View** tab at up-left corner), while the System Parameter **DeviceAutoEnable** is set "**true**" (click **Admin→System Parameter** tab, and choose the 21th parameter).

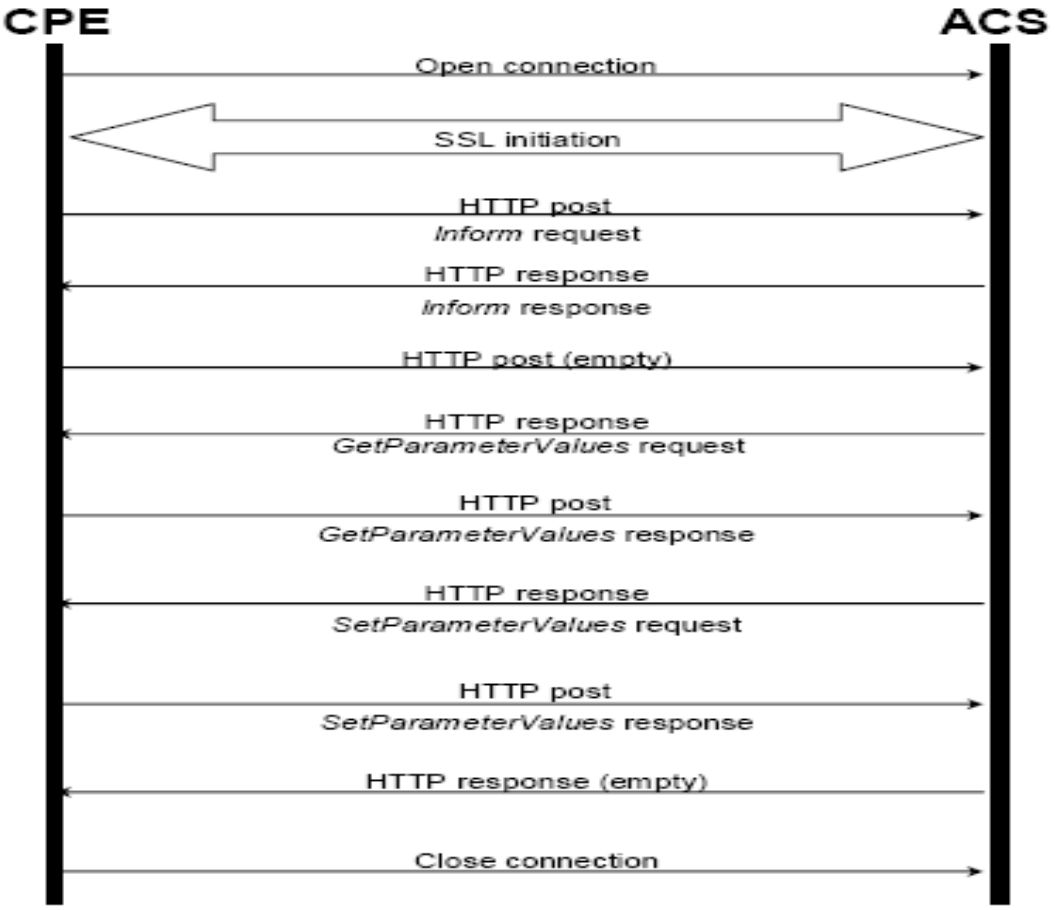


However, VigorACS allows users to define their own Table View、set CPE under another tree structure by configuring setting in **Network Management** and **Device Management** functions (click **Admin→ Device Management/Network Management**), or upload a pre-configured profile, those functions will be discuss later at section **3.2**.

2.2 Underlying concept of auto discovery and configuration

What makes VigorACS capable of those automatic functions? TR069 specification's connection model defined that CPE must initiatively connect to VigorACS, thus VigorACS could retrieve information like IP address, model name, OUI, etc. CPE would connect to VigorACS in following circumstances:

- At the first time the CPE establishes a connection to the access network
- On power-up or reset
- At every PeriodicInformInterval (for example, every 24-hours)
- When instructed by the optional ScheduleInform method
- Whenever the CPE receives a valid Connection Request from an ACS.
  Whenever the URL of the ACS changes



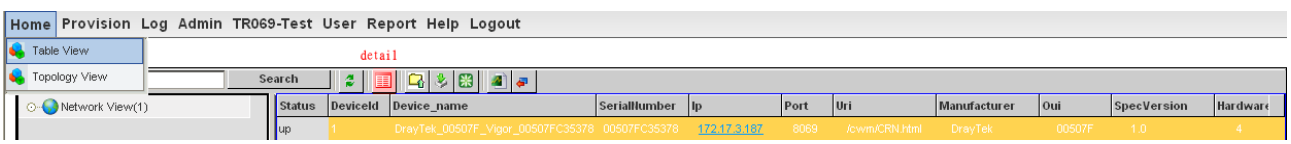
Once the connection initiated, VigorACS could provision and manage those CPEs.

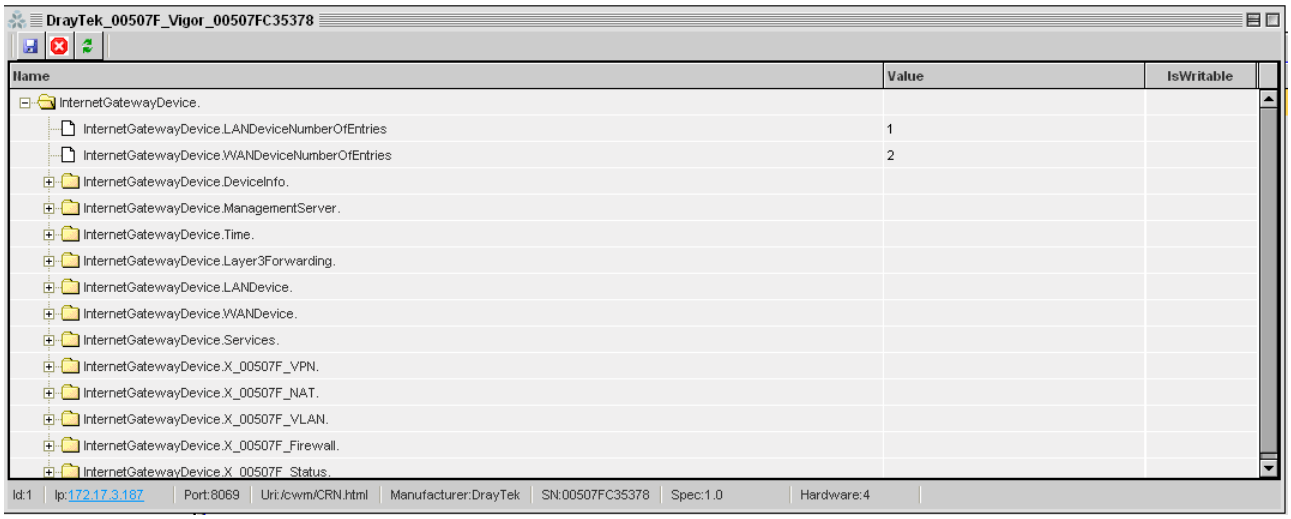
3. What is the auto configuration function of VigorACS

3.1 Auto configuration function

Centralized Management Systems doesn't comply with TR069, would have to be set management information base (MIB) to configure parameters of CPEs. If CPEs added new parameters, those systems need to be set new parameters. VigorACS is capable to get and configure parameter changes automatically, even for new parameters added on CPEs.

For example: Click on a device in **Table View** (the default page after logging, or click **Home→Table View**), then click **detail** button  . All parameters of the device would be retrieved by VigorACS and shown in a new window.





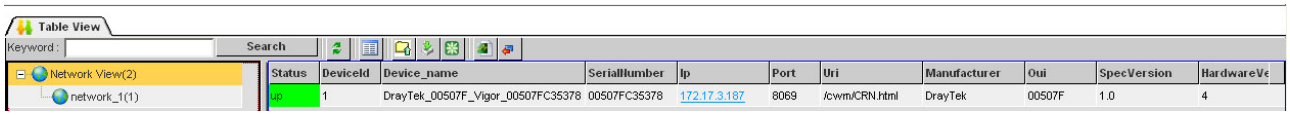
VigorACS retrieves and configures the parameters without any data defined or inputted by users, the process is totally automatic.

3.2 Application of auto configuration

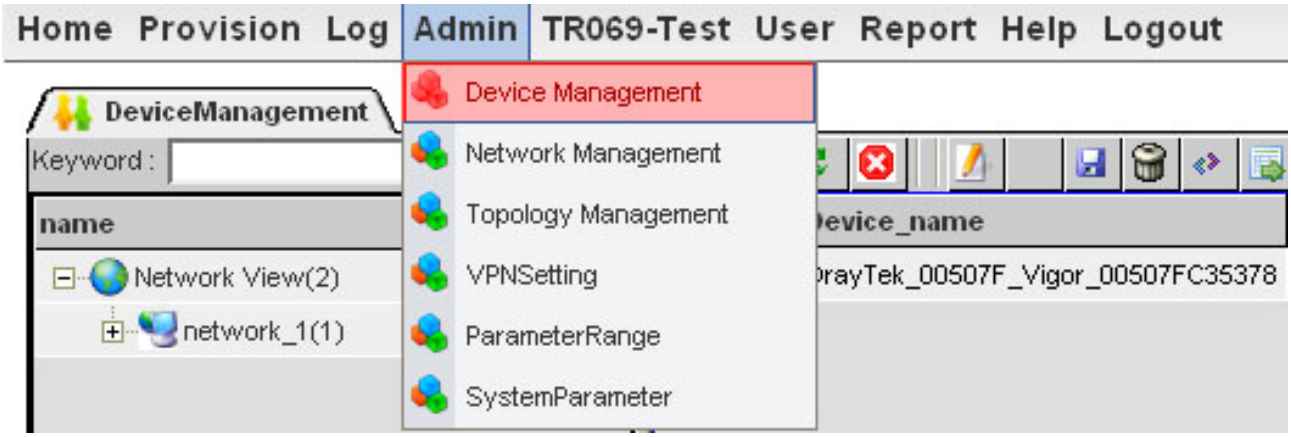
As mentioned in last paragraph of section 2.1, while the system parameter **DeviceAutoEnable** is set to "**true**", any newly discovered CPE would be added to the root network and shown in **Table View**. But what if users want to add the new CPE in another network? Even more, arrange numerous CPEs to specific network?

3.2.1 Change network

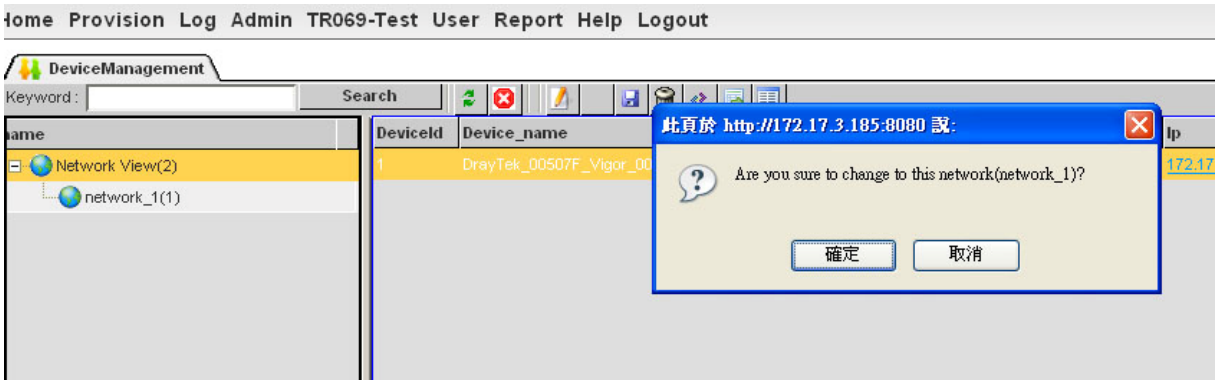
VigorACS allow user to change CPEs to defined network. For example: There are 2 networks in the **Table View**. Assume the device "DrayTek_00507F_Vigor_00507FC35378" under root network "**Network View**" need to be change to sub network "**network_1**".



Click **Admin→Device Management**, the device under the root network would be shown.

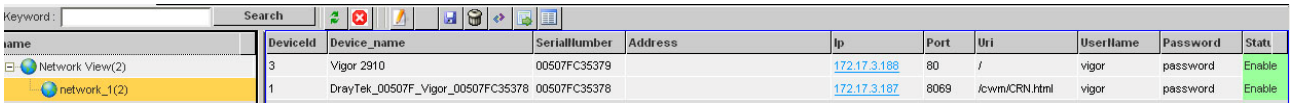


Drag “DrayTek_00507F_Vigor_00507FC35378” to “network_1”, the following dialog would pop, click “yes” to accept network change.



Check “network_1”, the device “DrayTek_00507F_Vigor_00507FC35378” was changed to it.

3.2.2 Configure xml profile



How do users change numerous CPEs to a certain network? It seems no effective to drag and change network one by one. Thus, VigorACS supports pre-configured profiles for this requirement.

The profile files are defined in text or xml format. For example, an xml file format is shown

AutoConfigurationExample.xml

```
1   <?xml version="1.0" encoding="UTF-8"?>
2   <tr069 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="tr069">
3    <items>
4     <!-- Parameters -->
5     <!-- SIP 1 -->
6     <item id="101" name="InternetGatewayDevice.Services.VoiceService.1.VoiceProfile.1.SIP.InboundAuthUsername"/>
7     <item id="102" name="InternetGatewayDevice.Services.VoiceService.1.VoiceProfile.1.Line.1.SIP.AuthUserName"/>
8     <item id="103" name="InternetGatewayDevice.Services.VoiceService.1.VoiceProfile.1.Line.1.SIP.AuthPassword"/>
9    </items>
10   <devices>
11    <!-- Device Test_B-->
12    <device serialnumber="00507FB888A8" name="Test_B" isreboot="true" network="MyNetwork1">
13     <parameter id="101" value="justin"/>
14     <parameter id="102" value="justin"/>
15     <parameter id="103" value="1001"/>
16    </device>
17    <!-- Device Test_D-->
18    <device serialnumber="00507FB88568" name="Test_D" isreboot="true" network="MyNetwork2">
19        <parameter id="101" value="kevin"/>
20        <parameter id="102" value="kevin"/>
21        <parameter id="103" value="1003"/>
22    </device>
23   </devices>
24  </tr069>
```

as below:

There are 2 main block in the xml file. One is the items block (line3 to line9), the other is the devices block (line10 to line23). Items block include the “item” element, which define valid TR069 parameters. The “id” is arbitrary unique number, and “name” is valid parameter name of TR069.

The devices block includes “device” elements, which define device attribute and parameters that reference to “item” (valid TR069 parameters). The attributes of device are **serialnumber, name, isreboot, value** and **network**.

“**serialnumber**” is the identification of the CPE. “**name**” is the CPE’s name, would automatically assigned with factory default. “**isreboot**” determine if the device needs to reboot after parameters have been set. “**value**” defines each parameter’s value. “**network**” defines which network would the CPE be set, if the network changed, the CPE would be set to different network.

3.2.3 Configure text profile

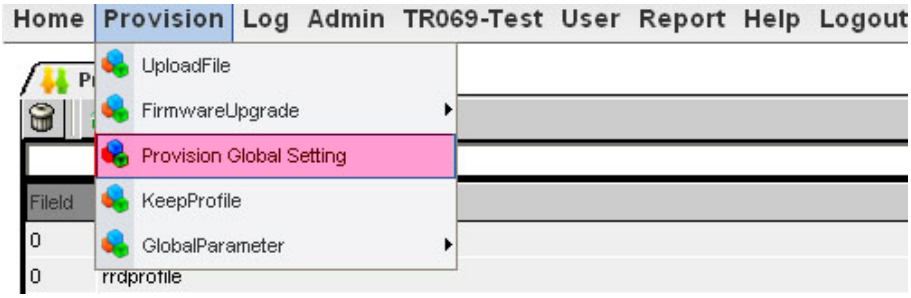
The profile can also be defined in text format:

AutoConfigurationExample.txt	
1	#This is comment
2	
3	[items]
4	#SIP
5	InternetGatewayDevice.Services.VoiceService.1.VoiceProfile.1.SIP.InboundAuthUsername
6	InternetGatewayDevice.Services.VoiceService.1.VoiceProfile.1.Line.1.SIP.AuthUserName
7	InternetGatewayDevice.Services.VoiceService.1.VoiceProfile.1.Line.1.SIP.AuthPassword
8	
9	[devices]
10	00507FB888A8,Test_B,ture,MyNetwork1,justin,justin,1001
11	00507FB88568,Test_D,true,MyNetwork2,kevin,Kevin,1003

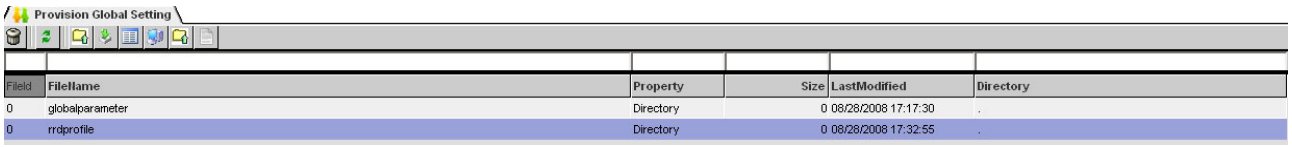
There are 2 main blocks, [items] and [devices]. As descriptions above, the “items” block (line3 to line7) defines parameter name of TR069 parameters. The “devices” block (line10 to line11) defines device and its attribute. The attributes are defined in sequence of **serialnumber, name, isreboot, network, value** (all elements listed after network in sequence). Each attribute is separated with “,”.

3.2.4 Set configured profile to VigorACS

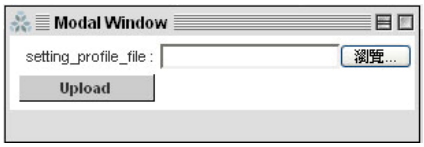
After defining profiles, click **Provision→Provision Global Setting**.



The following table would be displayed. There are 2 kinds of upload Button  . One prompted “Upload Text File”, for text format profile; the other prompted “Upload", for xml format profile. Click the button depend on what kind of profile you want upload.



The following upload window would be displayed after choosing the upload button.

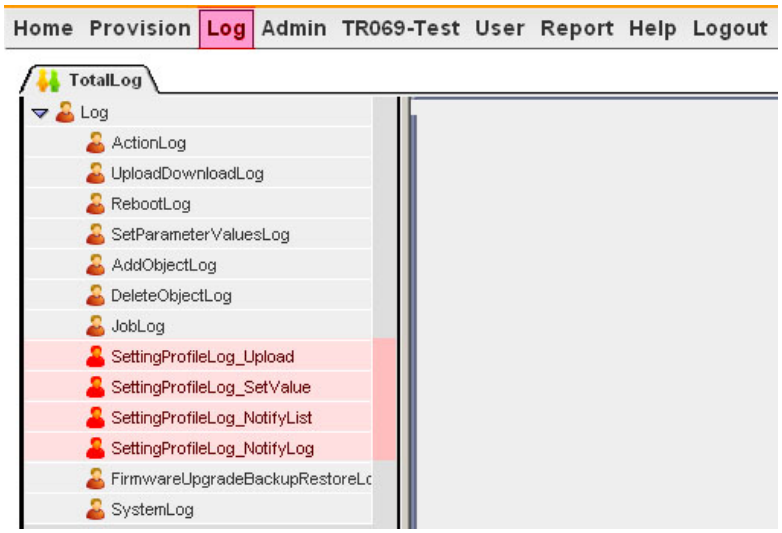

Modal Window
setting_profile_file :
瀏覽...
Upload

Click “browse” then choose the profile you want to upload. Click “Upload”, the configuration you defined in your profile file would be set to VigorACS.

3.2.5 Check setting profile log

How do users know if the setting profile process is successfully being executed? VigorACS support log function to summarize and record events.

Click **Log** to access the log functions.



Setting profile events are record in **SettingProfileLog_Upload**, **SettingProfileLog_SetValue**, **SettingProfileLog_NotifyList** and **SettingProfileLog_NotifyLog**.

After upload profile, you could check **SettingProfileLog_SetValue**, it tells if the setting profile process is executed correctly, the result shows as below: