Fusion System Parameter Reference

Version 2012R1

Table of Contents

1 Document Introduction

1.1 Document Purpose

The purpose of this document is to provide a detailed description the system parameters of Fusion.

1.2 Document Audience

Fusion Operators and Adiministratos

1.3 Document History

Version	Editor	Date	Changes
2012R1	M. Simonsen	9-Dec-11	Initial version

1.4 Acronyms and Abbreviations

Acronym	Explanation
ACS	Auto Configuration Server.
APS	Automatic Provisioning System.
Fusion	Ping Communication's eXtensible APS with advanced features such as Service Windows, Job Control and Smart Groups.
Fusion Module	The Fusion consists of several independently running Modules. The Modules may run on separate hosts in the network.
North Side	Common term describing the part of the Fusion which includes all Fusion Modules that provide management interfaces.
South Side	Common term describing the part of the Fusion which includes all Fusion Modules that provide CPE communication protocols.
Core	Common term describing the part of the Fusion which includes all Fusion Modules that provide neither management interfaces nor CPE communication protocols.
CPE	Customer Premises Equipment. Used in this document to refer to a single physical device. Same as the term “Device”.
Parameter	Each individual configuration setting is represented in the Fusion Data Model as a Parameter. A Parameter consists of a name and usually (but not always) a value.
Unit	A dataset in the Fusion database consisting of Parameter values relating to a single CPE. This dataset may extend beyond the Parameter values actually sent to the CPE, as some Parameter values may only be useful or needed by the Fusion itself. Also, the dataset may represent only a subset of all the configurable settings in the CPE. For these reasons, it is important to distinguish the term “Unit” from the terms “CPE” and “Device”.
Profile	Dataset stored in the Fusion containing Parameter values shared by multiple Units of the same Unit Type. A Unit is always

	assigned to a single profile. Multiple Profiles may be created for a Unit Type.
Unit Type	Units that represent CPEs of the same model share a common definition of that CPE model named Unit Type. The Unit Type definition is a list of Parameter names only, as the Unit Type never contains any Parameter values (values are stored in the Unit and/or Profile).
Group	A set of matching criteria used to search for Units. Commonly referred to as Smart Group.
Job	Automates and controls changes to Units within a Group. Partitions the changes over time according to rules to limit network load.
Job Chain	Multiple Jobs being automatically executed in a designated sequence.
Periodic Mode	Provisioning Mode where the Fusion automatically configures all CPEs based on their combined Unit and Profile parameter values.
Inspection Mode	Provisioning Mode where an Fusion Operator manually inspects and configures a single CPE through the Fusion Web Interface.
Staging	Fusion functionality used for optimizations in manufacturing, logistics and time-to-market for CPEs.
TR-069	Industry standard provisioning protocol used by the Fusion to read and write configurations from and to the CPEs, in addition to handle upgrades.
SPP	Simple Protocol Provisioning, covers HTTP(S)/TFTP

2 Introduction

The system parameters are of vital importance for Fusion. These parameters controls the behaviour of the provisioning. Some are modified using Web interface/tools/wizards, but all of them can be manipulated directly if you know what to do. By reading this document you will gain a deeper understanding on how to modify and control Fusion.

The system parameters are stored the same way as regular parameters from the device. They can be found in the UnitType/Profile/Group/Job/Unit and work the same way in most cases. The major difference is that Fusion will use the information is these parameters from time to time and they will not ever be provisioned to the device.

2.1 Parameter Naming Convention

All parameter has the prefix "System.X_OWERA-COM.". Owera refers to a company reponsible for the predecessor of Fusion, and otherwise the naming convention follows the TR-069 parameter naming convention.

3 Parameter Reference

All parameters can be used as Group parameter, so for simplicity's sake we only mentioned where to use the parameters otherwise (Job, Unit or Profile).

3.1 Debug

Read-write Unit/Profile/Job parameter
Values allowed: 0 or 1
Protocols: N/A

Setting it to 1 triggers server-side debug-mode, which means that loglevels will be overridden for the Units affected and set to Debug. The setting will not affect the loglevels on the device itself. Be careful about setting this on profile level, it has some performance impact.

3.2 DesiredScriptVersion

Read-write Unit/Profile/Job parameter
Values allowed: Version number on a File of SCRIPT type in Fusion (same Unit Type)
Protocols: TR-069

Setting it will trigger the server to issue a Download-request for this particular script-file. The device will then run the script. The script MUST update a version-number within the device which changes the parameter `(InternetGateway)Device.DeviceInfo.VendorConfigFile.Version`, otherwise the Download-request will be issued upon every connect from the device to the server (a loop situation). If the version numbers match (in DesiredScriptVersion and in VendorConfiguFile.Version) no Download-request will be issued.

3.3 DesiredSoftwareVersion

Read-write Unit/Profile/Job parameter
Values allowed: Version number on a File of SOFTWARE type in Fusion (same Unit Type)
Protocols: TR-069, HTTP(S), TFTP

The behaviour is the same as for DesiredScriptVersion, except that the compare parameter in the device is `(InternetGateway)Device.DeviceInfo.SoftwareVersion`. It is important that the new software update this parameter to match the DesiredSoftwareVersion.

3.4 Device.MAC

Read-only Unit parameter
Protocols: TR-069, HTTP(S), TFTP

The MAC address of the device is populated when the device connects to the server. Sometimes the value may be a Serialnumber instead.

3.5 Device.PeriodicInterval

Read-only Unit parameter
Protocols: TR-069, HTTP(S), TFTP

PeriodicInterval is decided by the server by the usage of ServiceWindow parameters. This parameter is a copy whatever periodic interval is computed and sent to the device.

3.6 Device.PublicIPAddress

Read-only Unit parameter
Protocols: TR-069, HTTP(S), TFTP

Contains the IP address of the device upon connect to the server. This is the public IP address and does not tell if the device is behind NAT or not.

3.7 Device.SoftwareVersion

Read-only Unit parameter
Protocols: TR-069, HTTP(S), TFTP

Contains the Softwareversion of the device upon connect to the server.

3.8 FirstConnectTms

Read-only Unit parameter
Protocols: TR-069, HTTP(S), TFTP

Contains the timestamp of the first connect from the device to the server.

3.9 IM.Message

Read-only Unit parameter
Protocols: TR-069

Message about response from Unit sent from Fusion STUN server to Web/Shell when performing Inspection/Kick.

3.10 Job.Current

Read-only Unit parameter
Protocols: TR-069, HTTP(S), TFTP

Tells the job Id (internal job id in Fusion DB) of a job that is on-going on this Unit. This is important to be able to perform both Execution and Verification of a Job (two-step process).

3.10.1 Job.Disruptive

Read-only Unit parameter
Protocols: TR-069, HTTP(S), TFTP

If set to 1 a previous job in a currently ongoing job chain has been a disruptive job. To avoid a situation where job processing stops due to short Disruptive Service Window, this parameter is set and will tell all the rest of the jobs in this job chain to disregard the Service Window and complete the tasks.

3.11 Job.History

Read-only Unit parameter, possible to modify but at some risk
Protocols: TR-069, HTTP(S), TFTP

To keep track of which jobs has been performed, every Unit populates a "Job History". The history is a series of job-history-elements of this format:

<job-id>:<repeat-count>:<executed-tms>

The elements are separated by commas, and extra commas are placed at the front and the end of the history string. The job ids are seen on the Job page in Web. New jobs are inserted in the history at the beginning of the parameter value. The repeat count and executed tms is only interesting for repeatable jobs. An example of a job history parameter:

```
,150:0:1304432665457,108:29:1294136844783,
```

You may restart a job by manipulating job history, but at some risk. If the job history gets longer than a Unit parameter value can possibly be (512 characters), problems will occur and a cleanup of old jobs and the history parameter should be performed. If you remember completely stop (FINISH) old and unused jobs, Fusion Core will delete those automatically (check xaps-core.properties). The job history Unit parameter will then be cleared of non-existing jobs by TR-069 or SPP server upon connect with the device.

3.12 LastConnectTms

Read-only Unit parameter
Protocols: TR-069, HTTP(S), TFTP

Contains the timestamp of the last connect from the device to the server.

3.13 ProvisioningMode

Read-only Unit parameter, can be changed is special cases
Protocols: TR-069

The modes can be REGULAR (default) or READALL. If in REGULAR mode, a normal provisioning cycle will be executed. This includes things like discovery of a device, regular provisioning of a configuration, rebooting, resetting and download of software/script.

However, if you want to read all the parameters from a device, you can change the mode into READALL. This mode will automatically revert to REGULAR within approximately 15 minutes, but if the device initiates a provisioning cycle before those 15 minutes have passed, then the ACS will request ALL parameters from the device. All these parameters are stored in a separate cache on the server, not overwriting or disturbing the data read from the REGULAR mode. You will see these parameters as an extra column appearing in the Web/Shell. The idea is that using READALL is something you'll do to "debug" the device.

3.14 Reset

Read-write Unit/Profile/Job parameter
Values allowed: 0 or 1
Protocols: TR-069

If set to 1 a FactoryReset command will be issued to the device upon connect. The TR-069 server will automatically reset the value to 0 after issuing the command.

3.15 Restart

Read-write Unit/Profile/Job parameter
Values allowed: 0 or 1
Protocols: TR-069, HTTP(S), TFTP

If set to 1 a Reboot command will be issued to the device upon connect. The TR-069 server will automatically reset the value to 0 after issuing the command. Not all devices running HTTP(S)/TFTP will be able to perform a reboot.

3.16 ScriptURL

Read-write Unit/Profile/Job parameter
Values allowed: A URL pointing to a script file
Protocols: TR-069

This parameter is only used if you're not satisfied with Fusion handing out the files. One can imagine a situation where a large upgrade will create too much stress on the TR-069 server (usually handling the File-request from the devices). In that case this URL will override the default setting and you may specify a URL to any server to offload TR-069 Server.

3.17 Secret

Read-write Unit/Profile parameter
Values allowed: The value of the ACS-password found in the corresponding CPE
Protocols: TR-069, HTTP(S), TFTP

The parameter should be set using a lotfile/taiwanfile. Another option is to use Discovery mode in TR-069 server to detect the password (it will be stored in this parameter automatically). Some vendors produce devices all with the same ACS-password, and in that case it is appropriate to specify the Secret on the Profile level.

3.18 SecretScheme

Read-write Unit/Profile parameter
Values allowed: Anything
Protocols: HTTP(S)/TFTP

If value is specified at all, it will be interpreted as the response from the server should be encrypted. Not all devices can support this, so this is highly device-dependent.

3.19 ServiceWindow.Disruptive

Read-write Unit/Profile/Job parameter
Values allowed: Service Window format
Protocols: TR-069, HTTP(S), TFTP

A Service Window describes a time window where provisioning is allowed. In this particular Service Window, we describe a time window for disruptive changes on the device (in general that is SOFTWARE changes, RESET and RESTART changes). The format is

<mo|tu|we|th|fr|sa|su>-<mo|tu|we|th|fr|sa|su>:<hhmm>-<hhmm>

Examples are:

mo-su:0000-0000 (default window, all day - all week)
mo-fr:0800-1600 (a window open only during working hours)
mo-fr:0100-0500 (a window open only during night before work)
su-we:2300-0500 (window open 6 hours each night from Sunday Wednesday)

This parameter should for the most part be set on the profile, so all Units will inherit the settings. If a job chooses to run under a Disruptive Service Window, it is to say that a Unit executing a Job will find the ServiceWindow either on the Unit or in the Profile. Setting a Service Window in a job will not affect the time when such a job is executed, but it will change the Unit setting of that device and thus change the time window for changes the next time a device connects to the server.

3.20 ServiceWindow.Enable

Read-write Unit/Profile/Job parameter
Values allowed: 0 or 1
Protocols: TR-069, HTTP(S), TFTP

By default Service Windows are enabled. Disabling it will let the servers ignore the Service Window and Spread settings. However, the Frequency will still be honored.

3.21 ServiceWindow.Frequency

Read-write Unit/Profile/Job parameter
Values allowed: An integer from 1 to 20160. Higher numbers will not lead to increased frequency.
Protocols: TR-069, HTTP(S), TFTP

The frequency specifies the number of time to perform a periodic provisioning during one week. Default value is 7. If Service Window is enabled, the frequency will be calculated according to the open hours of the Service Window specifed. If a SW is only open 4 hours from monday to friday (total of 20 hours), and frequency is set to 20, the device will provision every hour in the "opening hours" of the Service Window.

3.22 ServiceWindow.Regular

Read-write Unit/Profile/Job parameter
Values allowed: Service Window format
Protocols: TR-069, HTTP(S), TFTP

This Service Window is the default window to be used, unless there is SOFTWARE, RESET or RESTART change. However, a job can override the setting, so that a CONFIG job that actually makes the device reboot can be specified to run under a Disruptive Service Window. Apart from that Regular Service Window behaves the same as Disruptive Service Window.

3.23 ServiceWindow.Spread

Read-write Unit/Profile/Job parameter
Allowed values: An integer from 0 to 100
Protocols: TR-069, HTTP(S), TFTP

Default value is 50, which means 50%. With this setting the provisioning intervals is spread by a factor of 50%. If the frequency is set to 7 and service windows are open all week, all day, then a provisioning interval can be from 12 to 36 hours. Set it 0 to turn off spread. The idea of spread is to avoid synchronization of device connects (which can happen in a large power outage scenario, key servers breakdown, etc).

3.24 SoftwareURL

Read-write Unit/Profile/Job parameter
Values allowed: A URL pointing to a software file

Protocols: TR-069

This parameter is only used if you're not satisfied with Fusion handing out the files. One can imagine a situation where a large upgrade will create too much stress on the TR-069 server (usually handling the File-request from the devices). In that case this URL will override the default setting and you may specify a URL to any server to offload TR-069 Server.

3.25 Telnet.DesiredScriptVersion

Read-write Unit/Profile/Job parameter
Values allowed: Version number on a File of SCRIPT type in Fusion (same Unit Type)
Protocols: Telnet

The parameter will be read and used if you start a Telnet Job. Then this script will be loaded from Fusion DB and sent to the device.

3.26 Telnet.IPAddress

Read-write Unit/Profile/Job parameter
Values allowed: Public IP address of the device. If device is behind NAT, port forwarding must be setup on the gateway.
Protocols: Telnet

The parameter will be read and used if you start a Telnet Job. Then the script will send the script to this IP address.

3.27 Telnet.Password

Read-write Unit/Profile/Job parameter
Values allowed: The password (if required) by the device upon Telnet session initiation.
Protocols: Telnet

The parameter will be read and used if you start a Telnet Job. The password will be used if necessary. Currently the device must request the string (case-insensitive) "password" in the login screen, otherwise the Telnet-client will not understand where to respond with the password.

3.28 Telnet.Port

Values allowed: Public telnet port of he device. If device is behind NAT, port forwarding must be setup on the gateway.
Protocols: Telnet

The parameter will be read and used if you start a Telnet Job. Then the script will send the script to this port.

3.29 Telnet.Username

Read-write Unit/Profile/Job parameter
Values allowed: The username (if required) by the device upon Telnet session initiation.
Protocols: Telnet

The parameter will be read and used if you start a Telnet Job. The username will be used if necessary. Currently the device must request the string (case-insensitive) "login" or "username" in the login screen, otherwise the Telnet-client will not understand where to respond with the username.