FreeACS Installation

Version 2018-V2.0.0

Table of Contents

Table of Contents

1 Document Introduction

1.1 Name of the system

The current name of the system is “FreeACS”. As this is a relatively new name, the old name “Fusion” is in frequent use and may continue to be for a very long time. An even older name “xAPS” is also in use.

1.2 Document Purpose

The purpose of the document is to explain how to install FreeACS (chapter 4).

1.3 Document Audience

The readers will be Administrators and System Operators.

1.4 Document History

Version	Editor	Date	Changes
2009-R1	Morten Simonsen	18-Feb-09	Initial public version.
2009-R1-U1	Fredrik Gratte	31-Mar-09	Updated platform requirements.
2009-R2	Morten Simonsen	02-Jul-09	Revised edition
2011R1	Morten Simonsen	21-Jan-11	Revised edition
2012R1	Morten Simonsen	28-Dec-11	Name change/upgrade from 2011R1 procedure. Added a chapter.
2013R1	Morten Simonsen	17-Jan-13	Updated to latest release
2014R1	Morten Simonsen	03-Feb-14	System is no longer a commercial product – is licensed under the MIT license for free usage. The differences between 2013R1 and 2014R1 are otherwise small.
2014R1	Morten Simonsen	07-Jul-14	Major overhaul. Has created an install-script to do most of the work automatically. Updated to run on Ubuntu 14.04. The installation procedure has been brought down to minimum 5-6 minutes.
2018-V2.0.0	Jarl André Hübenthal	04-Aug-18	Replacing references to tomcat with proper alternative configuration. Services are now no longer deployed in tomcat, but installed an run standalone. Changing name.

2 Quick Overview

A complete installation of a Default Setup is provided in chapter 4, you may skip chapter 2 and 3.

FreeACS can be run in several configurations depending on your needs. The following section will list all modules and comment on where there is a choice to be made.

2.1 Infrastructure:

This list can also be read as requirements from Ping Communication to the customer, as the customer needs to be knowledgeable about these infrastructure parts, or at the very least be able to acquire the necessary knowledge, to maintain all these components.

- FreeACS can run on one physical server if necessary. Several factors come into play to decide how many servers is optimal.
- Operating system which can run JRE 1.7 (see details and exception to this below in the 'OS' chapter).
- MySQL 5.7
- JRE 1.8 (latest update)

2.2 Modules

North-side modules (user interface modules):

- FreeACS Web (Standard web interface for management)
- FreeACS Shell (CLI, script automation, management)
- FreeACS Web Services (if system integration is needed)

Core modules:

- FreeACS DB (table definitions)
- FreeACS Core
- FreeACS Syslog Server
- FreeACS Monitor Server

South-side modules (CPE interface modules):

- FreeACS TR-069 Server (if you have TR-069 devices)
- FreeACS STUN (needed to support TR-111)

The following chapters will explain how to install these modules.

3 Customer requirements

3.1 Hardware

There are many ways one could organize the hardware to satisfy FreeACS. You could do with one physical server at start up. As the number of devices connected to FreeACS grows, you should probably split the processes/modules on several servers. This table should give you a quick overview of how Ping Communication thinks about this issue:

CPEs	Connects pr 24h	Devices w/syslog	FreeACS Web available on internet	Servers required	Comments
50K	1	Y	Y	1	The minimum requirement, see below for spec. for server.
500K	1	N	N	2	You should have server with the provisioning server (TR-069) in DMZ and the rest of the modules on another server within your intranet. This requirement is mainly due to security reasons.
500K	1	Y	N	3	Same as for the above, but syslog could generate a huge load, so it could be smart to have a separate server for the FreeACS Syslog Server *.
500K	24	N	N	5	You should have 3 provisioning servers, since the CPEs connect 24 times a day. The database would be put under some load here, so the database should also be place on its own server. The rest of the modules could be placed on one server.
12M	1	Y	Y	7	The same load as in the previous example, but syslog is turned on, so it will require some monitoring *. And don't put FreeACS Web on one of the provisioning server, just because they are both located in the DMZ, use a separate server for that module.

* Previously it was possible to use a different database for Syslog, but this is currently no longer possible. A consideration was done to keep the load on the database as low as possible in the other uses cases.

A server is expected to have a decent multi-core processor, minimum 8GB RAM, minimum 100 Mbit network interface and hard disk capacity of at least 500GB (this last requirement is only important for the database server). We expect the usage of fast HDD, since this is critical for the database. The minimum specification translates to some of the test servers we have used. Looking closely at these figures you should realize that this specification is a low-end system these days. A state-of-the-art system today would probably have more capacity. So if you think the number of servers will grow too rapidly with increasing numbers of connects pr

24h, keep in mind that in that situation you would probably use a state-of-the-art system, minimizing the number of servers required.

The tests we have done to come up with this list will of course not represent the absolute truth about how a potential customer will use the system. Particularly the number of parameters in the database, jobs activated, logging scheme, number of end users, number of interconnecting systems, will influence the performance. That said, we think these figures give a reasonable and reliable picture of the situation.

If you decide to run on multiple servers, the first split should be between FreeACS DB and the provisioning servers (TR-69), since these components are affected the most by an increase in devices. Another important point is that you can add provisioning servers to scale up the system, all of them connecting to the same FreeACS DB Server. There is another reason for this split as well, and that is that the provisioning servers must be reachable for all the devices, a requirement which you might not want for your database!

Another split would be to put all the interface modules (FreeACS Web, FreeACS Shell and FreeACS Web Services) on a separate server. A trigger for this move would be to secure these interfaces from direct access from the Internet.

Yet another split would be to put a syslog server on its own server, but that is something one does only if there’s a significant load on the syslog server. *

The bottleneck of this system will eventually be the database. However we believe that this bottleneck will not be hit before at least 10M CPEs are connected, possibly not before 30-50M CPEs are connected (it all depends on many factors). But this does not take into account that the database server may run in a cluster. We have not experimented with this, but we still believe this is an option, an option that no customer today is likely to reach without a very aggressive provisioning policy (e.g. many connects pr CPE every 24h).

3.2 OS

All modules in FreeACS are Java applications. In theory they can be installed on any OS that supports JRE 1.8. We have chosen to run on Linux, Ubuntu Server 16.04 64 bit, and we suggest that our customers do the same. If they do, it's easy to follow the installation procedure in chapter 5. We also support CentOS.

3.3 Database

FreeACS is advised to run on MySQL 5.7.

How to install MySQL is considered the responsibility of the customer. Furthermore it may be necessary to tweak the database somewhat as the load grows. This competence should be found within your company. That is to say that FreeACS is not a fool-proof system and will require some technical people to take part in the installation and operation.

That said we do have an installation procedure for a complete set up of a standard FreeACS Server which includes a reasonably good set up of MySQL 5.7. This was done to minimize our own support effort in the installation process, but also serves us well because the installation of FreeACS becomes more coherent across customers. Chapter 5 contains the detailed installation procedure of such a standard set up.

3.4 Java and Web container

As for the databases, you need to be able to install Java your system. FreeACS requires JRE 1.8, preferably the latest update. We previously deployed our services to Tomcat, but each services is now run indepently only depending on Java. They are firing up their own embedded web containers. This is good because if one services gets too much load the other services are unaffected.

4 Installation from scratch

You can have FreeACS up and running in 30 minutes, or possibly even in just 5 minutes (if you do this for the second time). Just read on.

The goal is to install a standard FreeACS Server (Default Setup), which requires installation of Ubuntu Server 16.04 64-bit and above, MySQL Server 5.7 (latest update) and JRE 1.8 (latest update). How to install Ubuntu 16.04 64-bit and above is beyond the scope of this simple document, but otherwise all other software installation is described. This is not to say that one cannot run on any other OS, but this is the standard/default FreeACS installation recommended for most users.

Do the following:

1. **Download install.sh from https://github.com/freeacs/freeacs/tree/master/scripts and run the script from your home folder on your ubuntu server. You must have root access. This covers 90% of the installation, and can be done in 1-5 minutes.**
2. **Go through the rest of the modifications described in this chapter. Should be possible to do in 5-30 minutes.**
3. **The server should be ready**

Important!! Yellow color indicates an optional step, but it's wise to read the comments before skipping.

4.1 /etc/mysql/my.cnf

Step	Command/Text	Comment
4.1.1	bind-address = 0.0.0.0	If you want your database to be accessible from outside localhost, set to 0.0.0.0. Else, the database will only be accessible for applications running on localhost (like FreeACS server).
4.1.2	max_allowed_packet = 32M	Should be at least 32M, to allow adding firmwares up until this size into the FreeACS database
4.1.3	innodb_buffer_pool_size=1024M	This is the most important memory setting, MySQL should have access to perhaps 50% av of total memory on server. If you set this setting to high MySQLs InnoDB engine may silently fail! Check in 2.7.
4.1.4	service mysql restart	Restart MySQL after changes

4.2 /opt/freeacs-monitor/config/application-config.properties

The monitor server itself is not critical for FreeACS, it's main job is to monitor the other servers in the FreeACS solution. Previously it reacted to triggers and sent mail, right now its primary usage is to check if modules are up. Email and trigger reactions is removed.

Step	Property	Comment
4.7.2	monitor.urlbase / monitor.url.[service]	Configure where to find FreeACS services, or configure where individual services are located.

4.3 /opt/freeacs-stun/config/application-config.properties

The STUN server is fairly important, since all server-side triggering of provisioning goes through this server. Thus, if you try to «kick» the CPE or press the «provisioning» button in the Web interface, the STUN server must have a correct configuration.

Step	Property	Comment
4.8.1	primary.ip	Set it to the IP address of your server. The server will try to bind to this IP on port 3478. If this fails, the server will not start unless you change the test.runwithstun
4.8.2	test.runwithstun	The server will start even if the STUN behaviour is not supported. In this case, the server can still be used to trigger/kick CPEs available on public ConnectionRequestURL addresses.

4.4 /opt/freeacs-web/config/application-config.properties

Step	Property	Comment
4.9.1	monitor.location	It should return a web-page (use wget to test). If not, change the url or check if the Monitor server is actually running.

4.5 Restart, firewalls and checks

Step	Command/Text	Comment
4.10.1	systemctl restart freeacs*	Check journalctl -u freeacs* -f to make sure FreeACS starts without errors.
4.10.2	wget localhost wget localhost/web wget localhost/tr069	If you have a firewall, open for TCP/80. You can check to see if tomcat is available by using the command. If everything went well you should get the FreeACS Web interface, with an user/password prompt. **Login using admin/freeacs as user/pass**. You may of course change the default password inside the web application. If the FreeACS Web interface does not appear, then try http://localhost/web. The TR-069 server should be available on http://localhost/tr069. The TR-069 clients will connect using HTTP POST, while the “browser” returns the response from HTTP GET.
4.10.3	freeacs-shell	This shell is providing a scripting environment to FreeACS. Previously shell could connect to multiple databases, but that is no more. It connects to a predefined database, usually localhost.
4.10.4	See chapter 5.3	Several port openings may be expected if a firewall is present
4.10.5	COMPLETE	The server is now ready!

4.6 Optional steps – SSL certificate on Nginx

Administrators is required to know how to install a certificate in nginx.

5 Technical reference and documentation

In this chapter you'll find important information of a installed FreeACS system; where to find log files, firewall settings, etc. By following the instructions in chapter 4, you'll end up with a Default Setup (DS), and for this setup we'll provide exact information.

5.1 Property files

Property files are found in **/opt/freeacs-[module]/config/application-config.properties** and also in **/var/lib/tomcat7/shell.**

application-config.properties: Contains all properties and control mechanism for the module.

Information about the various property files are found in the User Manuals of each module, but each property file is supposed to be self-documented.

Previously there was a log file also, but this is now internal to the service jar file. The logback.xml file can be extracted and modified and placed in the config folder. But then you would need to fix startup parameters to tell logback where to find the log configuration.

5.2 Log files

FreeACS logs

Log files are found in **/opt/freeacs-[module]/logs** The logs are named following this convention: **<modulename>(-<optionalname>).log**. Usually every module has a default/regular log: <modulename>.log, but some modules have multiple logs. See previous chapter on how to tune logging.

There is currently no log rotation.

5.3 Firewalls

The following holes in the firewall may/must be opened (for those modules placed behind the firewall):

Module	Port	Type	Comment
Monitor, TR069, Web, WS	80	TCP	In case you have setup the installation to run on port 80 (see chapter 4.5) To allow requests into TR-069 or HTTP for provisioning. Also access to monitor-server, Web and Web Services.
Monitor, TR069, Web, WS	8090 monitor 8081 web 8085 tr069 8088 ws	TCP	In case you run DS (skipped chapter 4.5). To allow requests into TR-069 or HTTP for provisioning. Also access to monitor-server, Web and Web Services.
Monitor, TR069, Web, WS	443	TCP	In case you have setup the installation to run on port 443 (see chapter 4.6). To allow requests

			into TR-069 or HTTP for provisioning. Also access to monitor-server, Web and Web Services.
DB	3306	TCP	Allows direct access to MySQL database (see chapter 4.1). This allows FreeACS Shell to run on a remote host accessing the DB directly. However its difficult to configure Shell atm for this.
STUN	3479 3480	UDP	To support TR-111 (the devices must also support this) and devices access this STUN-server.
Syslog	9116	UDP	To allow syslog messages to be sent to FreeACS Syslog server. Should always be open.

5.4 Documentation

All modules have a User Manual, to describe how to use the system. Some modules also have additional documentation. These documents are found in GitHub on the following locations.:

Server	**URL**	**Comment**
General	https://github.com/freeacs/readme	General documentation
Core	https://github.com/freeacs/freeacs/tree/master/core/docs	
Monitor	https://github.com/freeacs/freeacs/tree/master/monitor/docs	
Shell	https://github.com/freeacs/freeacs/tree/master/shell/docs	
STUN	https://github.com/freeacs/freeacs/tree/master/tr069/docs	Chapter 7
Syslog	https://github.com/freeacs/freeacs/tree/master/syslog/docs	
TR-069	https://github.com/freeacs/freeacs/tree/master/tr069/docs	
Web	https://github.com/freeacs/freeacs/tree/master/web/docs	
Web Services	Web Service module has been rewritten. Access new wsdl at http://[host]:[port]/ws/acs.wsdl. Access old docs at https://github.com/freeacs/freeacs/tree/master/webservice/docs	