

Powered by Universal Speech Solutions LLC



Asterisk Modules

Developer Guide

Revision: 53

Last Updated: April 4, 2020

Author: Arsen Chaloyan

Table of Contents

1	Overview.....	3
1.1	Applicable Versions.....	3
1.2	Supported Platforms	3
2	Obtaining the Source Code	4
2.1	Prerequisites.....	4
2.2	Downloading the Release Package	4
2.3	Retrieving from the Repository	4
3	Build the Project	6
3.1	Procedure	6
3.2	Configure Options.....	6
4	Test the Installation.....	8

1 Overview

This guide describes how to obtain and build from source UniMRCP module for Asterisk. The document is intended for developers familiar with software programming and integrated development environments.

1.1 Applicable Versions

Instructions provided in this guide are applicable to the following versions.

 UniMRCP modules for Asterisk 1.3.0 and above

1.2 Supported Platforms

The following Linux distributions are primarily supported.

Operating System
Red Hat / Cent OS 6
Red Hat / Cent OS 7
Red Hat / Cent OS 8
Ubuntu 14.04 LTS (trusty)
Ubuntu 16.04 LTS (xenial)
Ubuntu 18.04 LTS (bionic)

Other Linux distributions should also be supported with no or minimal changes in the installation procedure.

2 Obtaining the Source Code

2.1 Prerequisites

The development tools such as *autoconf*, *libtool*, *gcc*, and others must be installed on the system. The following command can be used to install all the prerequisites on a Red Hat-based distribution:

```
yum groupinstall "Development Tools"
```

Asterisk as well as UniMRCP need to be installed prior to installation of the UniMRCP modules for Asterisk. The following versions are supported:

```
Asterisk 1.6, 1.8, 10, 11, 12, 13, 14, 15, 16, 17  
UniMRCP 1.1.0 and above
```

Note: this guide does not cover installation of Asterisk.

UniMRCP can be installed either from source or using binary development packages from the corresponding YUM or Apt repositories.

```
yum install unimrcp-client-devel
```

or

```
sudo apt-get install unimrcp-client-dev
```

Refer to UniMRCP installation manuals for detailed installation procedure.

2.2 Downloading the Release Package

The release source packages have the following naming convention *asterisk-unimrcp-major.minor.patch*. The packages are available in *tar.gz* format, and can be downloaded from the following location:

```
http://www.unimrcp.org/downloads/asterisk
```

Note, that starting from 1.3.0 version, the release packages are no longer bundled with any specific version of Asterisk or UniMRCP.

2.3 Retrieving from the Repository

The source repository is hosted on [GitHub](#). The code can be retrieved using a git client.

```
git clone https://github.com/unispeech/asterisk-unimrcp.git
```

3 Build the Project

3.1 Procedure

If the source has been checked out from repository, the *bootstrap* script must be executed first in order to generate the *configure* script and other required files.

```
./bootstrap
```

The usual sequence of commands *configure*, *make* and *make install* should follow in order to build and install the project from source.

```
./configure  
make  
make install
```

As a result, the modules *res_speech_unimrcp.so* and *app_unimrcp.so* will be installed in the modules directory of Asterisk such as */usr/lib/asterisk/modules* by default. Similarly, the configuration files *res-speech-unimrcp.conf* and *mrcp.conf* will be placed in */etc/asterisk* by default.

3.2 Configure Options

There are a number of options which can be additionally configured.

To explicitly specify where to look for Asterisk, use the option *--with-asterisk*. For example, if Asterisk is installed in */usr/local/asterisk*, use:

```
./configure --with-asterisk=/usr/local/asterisk
```

To explicitly specify where the Asterisk configuration files are located, use the option *--with-asterisk-conf*. For example:

```
./configure --with-asterisk-conf=/usr/local/asterisk/conf
```

To explicitly specify the Asterisk version, use the option *--with-asterisk-version*. For example:

```
./configure --with-asterisk-version=11.2.1
```

To explicitly specify where to look for UniMRCP, use the option *--with-unimrcp*. For example, if UniMRCP is installed using the binary packages in */opt/unimrcp*, use:

```
./configure --with-unimrcp=/opt/unimrcp
```

To exclude the module *res_speech_unimrcp.so* from build, use:

```
./configure --disable-res-speech-unimrcp
```

To exclude the module *app_unimrcp.so* from build, use:

```
./configure --disable-app-unimrcp
```

4 Test the Installation

Run Asterisk and check if the UniMRCP modules are loaded normally.

```
*CLI> module show like res_speech_unimrcp.so
```

Module	Description	Use Count
res_speech_unimrcp.so	UniMRCP Speech Engine	0

1 modules loaded

```
*CLI> module show like app_unimrcp.so
```

Module	Description	Use Count
app_unimrcp.so	MRCP suite of applications	0

1 modules loaded