MRCP

PocketSphinx Plugin

Administrator Guide

Revision: 5
Distribution: Red Hat / Cent OS
Created: February 3, 2017
Last updated: August 30, 2017
Author: Arsen Chaloyan
Table of Contents

1 Overview........................................................................................................................................... 3
  1.1 Applicable Versions...................................................................................................................... 3
  1.2 Supported Distributions ............................................................................................................. 3
  1.3 Authentication ............................................................................................................................. 3
2 Installing RPMs Using YUM.............................................................................................................. 4
  2.1 Repository Configuration ............................................................................................................ 4
  2.2 Repository Verification ............................................................................................................... 4
  2.3 PocketSphinx Plugin Installation ............................................................................................... 5
3 Installing RPMs Manually ................................................................................................................ 6
  3.1 Package List ............................................................................................................................... 6
  3.2 Package Installation Order ......................................................................................................... 7
4 Obtaining License ........................................................................................................................... 8
  4.1 License Type ............................................................................................................................... 8
  4.2 Node Information ....................................................................................................................... 8
  4.3 License Installation ..................................................................................................................... 8
5 Configuring Server and Plugin ....................................................................................................... 9
  5.1 Plugin Factory ............................................................................................................................ 9
  5.2 Logger .......................................................................................................................................... 9
  5.3 PocketSphinx ............................................................................................................................ 9
6 Validating Setup ............................................................................................................................. 10
  6.1 Launching Server ....................................................................................................................... 10
  6.2 Launching Client ......................................................................................................................... 10
1 Overview

This guide describes how to obtain and install binary packages for the PocketSphinx plugin to the UniMRCP server on Red Hat-based Linux distributions. The document is intended for system administrators and developers.

1.1 Applicable Versions

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

- UniMRCP 1.5.0 and above
- UniMRCP PocketSphinx Plugin 1.0.0 and above

1.2 Supported Distributions

UniMRCP binary packages are currently available only for x86_64 (64-bit) architecture.

<table>
<thead>
<tr>
<th>Operating System</th>
<th>32-bit</th>
<th>64-bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red Hat / Cent OS 6</td>
<td></td>
<td>✔️</td>
</tr>
<tr>
<td>Red Hat / Cent OS 7</td>
<td></td>
<td>✔️</td>
</tr>
</tbody>
</table>

Note: packages for other distributions can be made available upon request. For more information, contact services@unimrcp.org.

1.3 Authentication

UniMRCP binary packages are available to authenticated users only. In order to register a free account with UniMRCP, please visit the following page.

- [https://www.unimrcp.org/profile-registration](https://www.unimrcp.org/profile-registration)

Note: a new account needs to be verified and activated prior further proceeding.
2 Installing RPMs Using YUM

Using the Yellowdog Updater, Modifier (yum), a command-line package management utility for Red Hat-based distributions, is recommended for installation of UniMRCP binary packages.

2.1 Repository Configuration

The content of a typical yum configuration file, to be placed in /etc/yum.repos.d/unimrcp.repo, is provided below.

```bash
[unimrcp]
name=UniMRCP Packages for Red Hat / Cent OS-$releasever $basearch
baseurl=https://username:password@unimrcp.org/repo/yum/main/rhel$releasever/$basearch/
enabled=1
sslverify=1
gpgcheck=1
gpgkey=https://unimrcp.org/keys/unimrcp-gpg-key.public

[unimrcp-noarch]
name=UniMRCP Packages for Red Hat / Cent OS-$releasever noarch
baseurl=https://username:password@unimrcp.org/repo/yum/main/rhel$releasever/noarch/
enabled=1
sslverify=1
gpgcheck=1
gpgkey=https://unimrcp.org/keys/unimrcp-gpg-key.public
```

The username and password fields included in the HTTPS URI must be replaced with the corresponding account credentials.

2.2 Repository Verification

In order to verify that yum can properly connect and access the UniMRCP repository, the following command can be used.

```
yum repolist unimrcp
yum repolist unimrcp-noarch
```

where `unimrcp` and `unimrcp-noarch` are names of the sections set in the yum configuration file above.

In order to retrieve a list of packages the UniMRCP repository provides, the following command can be used.

```
yum --disablerepo="*" --enablerepo="unimrcp" list available
```
2.3 PocketSphinx Plugin Installation

In order to install the PocketSphinx plugin package, including all the dependencies, use the following command.

```
yum install unimrcp-pocketsphinx
```

In order to install the en-US model package for PocketSphinx, use the following command.

```
yum install unimrcp-pocketsphinx-model-en-US
```

Note: this package provides acoustic data files required for the PocketSphinx engine to perform recognition for the en-US language. Packages for other languages can be made available upon request.

In order to install the additional data files for the sample client application umc, the following command can be used.

```
yum install umc-addons
```

Note: this package is optional and provides additional data which can be used for validation of basic setup.
3 Installing RPMs Manually

UniMRCP RPM packages can be installed manually using the `rpm` utility. Note, however, that the system administrator should take care of package dependencies and install all the packages in appropriate order.

The RPM packages have the following naming convention:

```
$packagename-$universion-$packageversion.el$rhelversion.$arch.rpm
```

where

- `packagename` is the name of a package
- `universion` is the UniMRCP version
- `packageversion` is the RPM release version
- `rhelversion` is the Red Hat version
- `arch` is the architecture (x86_64, i686, …)

3.1 Package List

The following is a list of UniMRCP RPM packages required for the installation of PocketSphinx plugin.

<table>
<thead>
<tr>
<th>Package Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>unimrcp-pocketsphinx</td>
<td>PocketSphinx plugin to the server.</td>
</tr>
<tr>
<td>umc-addons</td>
<td>Sample en-US data files used with umc. [Optional]</td>
</tr>
<tr>
<td>unilicnodegen</td>
<td>Node information retrieval tool, required for license deployment.</td>
</tr>
<tr>
<td>unimrcp-server</td>
<td>Shared library and application of the server.</td>
</tr>
<tr>
<td>unimrcp-client</td>
<td>Shared libraries and sample applications of the client. [Optional]</td>
</tr>
<tr>
<td>unimrcp-demo-plugins</td>
<td>Set of demo plugins to the server. [Optional]</td>
</tr>
<tr>
<td>unimrcp-common</td>
<td>Data common for the client and the server.</td>
</tr>
<tr>
<td>uniapr</td>
<td>UniMRCP edition of the Apache Portable Runtime (APR) library.</td>
</tr>
</tbody>
</table>
3.2 Package Installation Order

Note that all the RPM packages provided by UniMRCP are signed by a GNU Privacy Guard (GPG) key. Before starting the installation, you may need to import the public key in order to allow the `rpm` utility to verify the packages.

```
rpm --import https://unimrcp.org/keys/unimrcp-gpg-key.public
```

Packages for the APR, APR-Util and Sofia-SIP libraries must be installed first.

```
rpm -ivh uniapr-$aprversion-$packageversion.el$rhelversion.$arch.rpm
rpm -ivh uniapr-util-$apuversion-$packageversion.el$rhelversion.$arch.rpm
rpm -ivh unisofia-sip-$sofiaversion-$packageversion.el$rhelversion.$arch.rpm
```

Then, a package containing common data for the client and the server, and a package for the server should follow.

```
rpm -ivh unimrcp-common-$universion-$packageversion.el$rhelversion.$arch.rpm
rpm -ivh unimrcp-server-$universion-$packageversion.el$rhelversion.$arch.rpm
```

Next, a package containing the utility tool `unilicnodegen`, required for license deployment.

```
rpm -ivh unilicnodegen-$toolversion-$packageversion.el$rhelversion.$arch.rpm
```

Next, a package containing the PocketSphinx plugin should follow.

```
rpm -ivh unimrcp-pocketsphinx-$universion-$packageversion.el$rhelversion.$arch.rpm
```

Finally, one or more packages corresponding to a particular language should follow.

```
rpm -ivh unimrcp-pocketsphinx-model-en-US-$universion-$packageversion.el$rhelversion.noarch.rpm
```
4 Obtaining License

The PocketSphinx plugin to the UniMRCP server is a commercial product, which requires a license file to be installed.

4.1 License Type

The following license types are available:

- Trial
- Production
- Test and Development

4.2 Node Information

The license files are bound to a node the product is installed on. In order to obtain a license, the corresponding node information needs to be retrieved and submitted for generation of a license file.

Use the installed tool unilicnodegen to retrieve the node information.

```
/opt/unimrcp/bin/unilicnodegen
```

As a result, a text file uninode.info will be saved in the current directory. Submit the file uninode.info for license generation to services@unimrcp.org by mentioning the product name in the subject.

4.3 License Installation

The license file needs to be placed into the directory /opt/unimrcp/data.

```
cp umspocketsphinx_*.lic /opt/unimrcp/data
```
5 Configuring Server and Plugin

5.1 Plugin Factory

In order to load the PocketSphinx plugin into the UniMRCP server, open the file `unimrcpserver.xml`, located in the directory `/opt/unimrcp/conf`, and add the following entry under the XML element `<plugin-factory>`. Disable other recognition plugins, if available. The remaining demo plugins might also be disabled, if not installed.

```
<!-- Factory of plugins (MRCP engines) -->
<plugin-factory>
  <engine id="Demo-Synth-1" name="demosynth" enable="true"/>
  <engine id="Demo-Recog-1" name="demorecog" enable="false"/>
  <engine id="Demo-Verifier-1" name="demoverifier" enable="true"/>
  <engine id="Recorder-1" name="mrcprecorder" enable="true"/>
  <engine id="PS-Recog-1" name="umspocketsphinx" enable="true"/>
</plugin-factory>
```

5.2 Logger

In order to enable log output from the plugin and set filtering rules, open the configuration file `logger.xml`, located in the directory `/opt/unimrcp/conf`, and add the following entry under the element `<sources>`.

```
<source name="PS" priority="INFO" masking="NONE"/>
```

5.3 PocketSphinx

The configuration file of the plugin is located in `/opt/unimrcp/conf/umspocketsphinx.xml` and the data files are in the directory `/opt/unimrcp/data/pocketsphinx`.

Refer to the Usage Guide for more information.
6 Validating Setup

Validate your setup by using the sample UniMRCP client and server applications on the same host. The default configuration and data files should be sufficient for a basic test.

6.1 Launching Server

Launch the UniMRCP server application.

```
cd /opt/unimrcp/bin
./unimrcpserver
```

In the server log output, check whether the plugin is normally loaded.

```
[INFO] Load Plugin [PS-Recog-1] [/opt/unimrcp/plugin/umspocketsphinx.so]
```

Next, check for the license information.

```
[NOTICE] UniMRCP PocketSphinx License

-product name: umspocketsphinx
-product version: 1.0.0
-license owner: Name
-license type: trial
-issue date: 2017-01-28
-exp date: 2017-02-27
-channel count: 2
-feature set: 0
```

6.2 Launching Client

Note: the optional package umc-addons must be installed for this test to work.

Launch the sample UniMRCP client application umc.

```
cd /opt/unimrcp/bin
./umc
```

Run a typical speech recognition scenario by issuing the command run ps1 from the console of the umc client application.
run ps1

This command sends a RECOGNIZE request to the server and then starts streaming a sample audio input file callsteve.pcm to recognize.

Check for the NLSML results to be returned as expected.

```xml
<?xml version="1.0"?>
<result>
  <interpretation grammar="digit" confidence="1.00">
    <instance>call steve</instance>
    <input mode="speech">call steve</input>
  </interpretation>
</result>
```

Visually inspect the log output for any possible warnings or errors.

Note that utterances are stored in the var directory, if the corresponding parameter is enabled in the configuration file umspocketsphinx.xml and/or requested by the client.