

*Powered by Universal Speech Solutions LLC*



# Yandex SR Plugin

## Administrator Guide

---

Revision: 3

Distribution: Red Hat / Cent OS

Created: December 22, 2018

Last updated: January 31, 2021

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 Yandex SR Plugin Installation.....	5
3 Installing RPMs Manually .....	6
3.1 Package List.....	6
3.2 Package Installation Order.....	7
4 Obtaining License .....	9
4.1 License Type.....	9
4.2 Node Information.....	9
4.3 License Installation .....	9
5 Obtaining Service Credentials .....	10
5.1 Service Subscription .....	<b>Error! Bookmark not defined.</b>
5.2 Installation of Credentials .....	<b>Error! Bookmark not defined.</b>
6 Configuring Server and Plugin .....	11
6.1 Plugin Factory Configuration .....	11
6.2 Logger Configuration .....	11
6.3 Yandex SR Plugin Configuration .....	11
7 Validating Setup.....	12
7.1 Setting up Folder ID .....	12
7.2 Launching Server.....	12
7.3 Launching Client.....	13

# 1 Overview

This guide describes how to obtain and install binary packages for the Yandex Speech Recognition (SR) plugin to 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 1.6.0 and above  
UniMRCP Yandex SR Plugin 1.0.0 and above

## 1.2 Supported Distributions


UniMRCP binary packages are currently available only for x86\_64 (64-bit) architecture.

Operating System	32-bit	64-bit
Red Hat / Cent OS 7		✓
Red Hat / Cent OS 8		✓

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

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.

```
[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
```

```
yum --disablerepo="*" --enablerepo="unimrcp-noarch" list available
```

## 2.3 Yandex SR Plugin Installation

In order to install the Yandex SR plugin, including all the dependencies, use the following command.

```
yum install unimrcp-yandex-sr
```

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 the Yandex SR plugin.

Package Name	Description
<b>unimrcp-yandex-sr</b>	Yandex SR plugin to the server.
<b>unigrpc</b>	UniMRCP edition of the gRPC library.
<b>unilibevent</b>	UniMRCP edition of the libevent library.
<b>umc-addons</b>	Sample en-US data files used with umc. [Optional]
<b>unilicnodegen</b>	Node information retrieval tool, required for license deployment.
<b>unimrcp-server</b>	Shared library and application of the server.
<b>unimrcp-client</b>	Shared libraries and sample applications of the client. [Optional]
<b>unimrcp-demo-plugins</b>	Set of demo plugins to the server. [Optional]
<b>unimrcp-common</b>	Data common for the client and the server.

<b>uniapr</b>	UniMRCP edition of the Apache Portable Runtime (APR) library.
<b>uniapr-util</b>	UniMRCP edition of the Apache Portable Runtime Utility (APR-Util) library.
<b>unisofia-sip</b>	UniMRCP edition of the Sofia SIP library.

## 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-$apuverion-$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, packages containing the gRPC and libevent libraries.

```
rpm -ivh unigrpc-$grpcversion-$packageversion.el$rhelversion.$arch.rpm
rpm -ivh unilibevent-$libeventversion-$packageversion.el$rhelversion.$arch.rpm
```

Finally, a package containing the Yandex SR plugin should follow.

```
rpm -ivh unimrcp-yandex-sr- $\$$ universion- $\$$ packageversion.el $\$$ rhelversion.noarch.rpm
```



# 4 Obtaining License

The Yandex SR 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](mailto: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 umsyandexsr_*.lic /opt/unimrcp/data
```

# 5 Obtaining Service Credentials

In order to utilize the Yandex SpeechKit API, corresponding service credentials need to be retrieved from the Yandex Cloud portal and further installed to the UniMRCP server.

## 5.1 Service Subscription

Subscribe to the Yandex Speech to Text API.

```
https://cloud.yandex.com/docs/speechkit/concepts/auth
```

Obtain your OAuth token.

1. Log in to Yandex using your Yandex Passport account.
2. Get an OAuth token from Yandex OAuth. To do this, follow the link, click **Allow** and copy the issued OAuth token.

## 5.2 Installation of Credentials

Create a text file *yandex.subscription.key* in the directory */opt/unimrcp/data*.

```
sudo nano /opt/unimrcp/data/yandex.subscription.key
```

Store your OAuth token in the text file.

```
*****
```

The provided OAuth token is used to obtain and periodically revalidate an IAM token.

# 6 Configuring Server and Plugin

## 6.1 Plugin Factory Configuration

In order to load the Yandex SR plugin into the UniMRCP server, open the file *unimrcpsvr.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="mrcpreorder" enable="true"/>
  <engine id="Yandex-SR-1" name="umsyandexsr" enable="true"/>
</plugin-factory>
```

## 6.2 Logger Configuration

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=" YANDEXSR-PLUGIN" priority="INFO" masking="NONE"/>
```

## 6.3 Yandex SR Plugin Configuration

The configuration file of the plugin is located in */opt/unimrcp/conf/umsyandexsr.xml*. Default settings should be sufficient for general use.

Refer to the *Usage Guide* for more information.

# 7 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.

## 7.1 Setting up Folder ID

The Yandex SpeechKit folder identifier must be specified in the configuration file of the plugin, located in `/opt/unimrcp/conf/umsyandexsr.xml`.

```
<streaming-recognition
  folder-id="*****"
  single-utterance="true"
  interim-results="true"
  start-of-input="service-originated"
  language="en-US"
  max-alternatives="1"
  alternatives-below-threshold="false"
  confidence-format="auto"
  results-indent="2"
  skip-unsupported-grammars="true"
  transcription-grammar="transcribe"
  auth-validation-period="3600"
/>
```

## 7.2 Launching Server

Start the UniMRCP server as a service.

```
systemctl start unimrcp
```

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

```
[INFO] Load Plugin [Yandex-SR-1] [/opt/unimrcp/plugin/umsyandexsr.so]
```

Next, check for the license information.

```
[NOTICE] UniMRCP YandexSR License

-product name:  umsyandexsr
-product version: 1.0.0
```

```
-license owner: -  
-license type: trial  
-issue date: 2018-12-22  
-exp date: 2019-01-21  
-channel count: 2  
-feature set: 0
```

## 7.3 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 gsr1* from the console of the *umc* client application.

```
run gsr2
```

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 version="1.0"?>  
<result>  
  <interpretation grammar="command" confidence="1">  
    <instance>Dial 5</instance>  
    <input mode="speech">Dial 5</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 *umsyandexsr.xml* and/or requested by the client.