



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Genesys Interaction Recording Solution Guide

Creating Folder Hierarchy for Recording Storage

4/2/2025

Creating Folder Hierarchy for Recording Storage

Contents

- [1 Creating Folder Hierarchy for Recording Storage](#)
 - [1.1 Prerequisites](#)
 - [1.2 Create and Use New Hierarchy](#)

WebDAV performance degrades over time and the file system becomes inoperable if all your recording files are saved in a single directory. If you are using Apache HTTP Server for your WebDAV server, Genesys recommends using the following example procedures to create a folder hierarchy.

Important

Genesys recommends that you create the storage directories before your contact center goes into production. Keep in mind, that if you use the following procedure to create the directories, you will need to extend the directories at the end of 2023.

Prerequisites

Before you can create recording storage with folder hierarchy, you must have the following items in place:

- The SIP Server's **recording-filename** option must be set to `$UUID$_$DATE$_$TIME$`
- The IVR Profile Recording Storage must be set to `http://<WebDAV server>/recordings`

Important

These instructions assume that `/recordings` is the base path of the WebDAV URI. If you use something else as the base path, change it accordingly in the rewrite rule.

- The IVR Profile **Recording Filename Template** option must be set to `id`

Important

The filename must be prefixed with `id`, to allow for additional parameters after this.

- For screen recording, these instructions assume that the storage path has a base path of `screens`. If you use something else as the base path, change it accordingly in the rewrite rule.

Create and Use New Hierarchy

For Linux

1. Create a file and copy and paste the content below into the file (for example, "createfolders.sh"). This will create many directories in yyyy/mm/dd/hh format. The example provided creates directories to the end of 2023.

```
#!/bin/sh
for year in 2019 2020 2021 2022 2023
do
for month in 01 02 03 04 05 06 07 08 09 10 11 12
do
for day in 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26
27 28 29 30 31
do
for hour in 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23
do
echo "mkdir -p $year/$month/$day/$hour"
mkdir -p $year/$month/$day/$hour
done
done
done
done
```

After saving the file, ensure the script has read and execute privileges and then run the script from /mnt/recordings.

2. Make sure the following Apache modules are loaded in the /etc/httpd/conf/httpd.conf file.
 - LoadModule rewrite_module modules/mod_rewrite.so
 - LoadModule proxy_module modules/mod_proxy.so
3. Depending on the Apache version, add the following rewrite rules in the /etc/httpd/conf/httpd.conf file to parse the file name into /recordings/{yyyy}/{mm}/{dd}/{hh}/{filename}:
 - Apache version 2.2 and lower

```
RewriteLog "logs/rewrite.log"
RewriteLogLevel 1
RewriteEngine on
RewriteRule ^/recordings/([A-Z0-9]*_(\d{4})-(\d\d)-(\d\d)_(\d\d).*)
/recordings/$2/$3/$4/$5/$1 [P,L]
RewriteRule ^/screens/([A-Z0-9]*_(\d{4})-(\d\d)-(\d\d)_(\d\d).*) /screens/$2/$3/$4/$5/$1
[P,L]
RewriteRule ^/screens/([A-Za-z0-9]*_(\d{4})_(\d\d)_(\d\d)_(\d\d).*)
/screens/$2/$3/$4/$5/$1 [P,L]
```

Tip

Test this rewrite rule in your environment to ensure that all the files are placed in the correct sub-directories.

- Apache version 2.4 and higher

```
LogLevel debug rewrite:trace1
RewriteEngine on
```

```
RewriteRule ^/recordings/([A-Z0-9]*_(\d{4})-(\d\d)-(\d\d)_(\d\d).*)  
/recordings/$2/$3/$4/$5/$1 [P,L]  
RewriteRule ^/screens/([A-Z0-9]*_(\d{4})-(\d\d)-(\d\d)_(\d\d).*) /screens/$2/$3/$4/$5/$1  
[P,L]  
RewriteRule ^/screens/([A-Za-z0-9]*_(\d{4})_(\d\d)_(\d\d)_(\d\d).*)  
/screens/$2/$3/$4/$5/$1 [P,L]
```

Important

For Apache version 2.4 and higher, rewrite logs are not written to a separate file (as in Apache 2.2) but to error_log file. To get the mod_rewrite specific log messages, you can use the following command:

```
tail -f error_log|fgrep '[rewrite:]'
```