Red Hat Enterprise Linux 8

Recording sessions

Using the Session Recording solution in Red Hat Enterprise Linux 8

Last Updated: 2025-05-20

Red Hat Enterprise Linux 8 Recording sessions

Using the Session Recording solution in Red Hat Enterprise Linux 8

Legal Notice

Abstract

This documentation collection provides introduction to using the Session Recording solution based on tlog with RHEL web console embedded player on Red Hat Enterprise Linux 8.

Table of Contents

PROVIDING FEEDBACK ON RED HAT DOCUMENTATION

We appreciate your feedback on our documentation. Let us know how we can improve it.

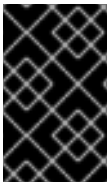
Submitting feedback through Jira (account required)

1. Log in to the Jira website.
2. Click **Create** in the top navigation bar.
3. Enter a descriptive title in the **Summary** field.
4. Enter your suggestion for improvement in the **Description** field. Include links to the relevant parts of the documentation.
5. Click **Create** at the bottom of the dialogue.

CHAPTER 1. GETTING STARTED WITH SESSION RECORDING ON RHEL

1.1. SESSION RECORDING IN RHEL

The Session Recording solution in Red Hat Enterprise Linux 8 is based on the **tlog** package. You can use the **tlog** package and its associated web console session player to record and play back user terminal sessions. You can configure the recording to take place per user or user group via the SSSD service. All terminal input and output is captured and stored in a text-based format in the system journal.



IMPORTANT

To not intercept raw passwords and other sensitive information, recording of the terminal input is disabled by default. Be aware that if you turn on recording of the terminal input, all entered passwords are captured in plaintext.

You can use this solution for auditing user sessions on security-sensitive systems or, in the event of a security breach, reviewing recorded sessions as part of forensic analysis. As an administrator, you can configure session recording locally on RHEL 8 systems. You can review the recorded sessions from the web console interface or in a terminal using the **tlog-play** command.

1.2. COMPONENTS OF SESSION RECORDING

There are three main components to the Session Recording solution: the **tlog** utility, the SSSD service and a web console embedded user interface.

tlog

The **tlog** utility is a terminal input/output (I/O) recording and playback program. It inserts the **tlog-rec-session** tool between the user terminal and the user shell, and logs everything that passes through as JSON messages.

SSSD

The System Security Services Daemon (SSSD) service provides a set of daemons to manage access to remote directories and authentication mechanisms. When configuring session recording, you can use SSSD to specify which users or user groups to record. You can configure these settings from the command line (CLI) or from the RHEL 8 web console interface.

The RHEL 8 web console embedded interface

The Session Recording page is part of the RHEL 8 web console interface and you can use it to manage recorded sessions.

IMPORTANT

You need administrator privileges to access the recorded sessions.

1.3. LIMITATIONS OF SESSION RECORDING

These are the most notable limitations of the Session Recording solution.

- Recordings of root user are not reliable, because the root user can circumvent the recording process.

- Session recording does not record the terminal in a **GNOME 3** graphical session. Recording terminals in graphical sessions is not supported because a graphical session has a single audit session ID for all terminals and **tlog** is unable to distinguish between the terminals and prevent repeated recordings.

- If session recording is configured to log to the **journal**, the recorded user will see the act of recording the results of viewing the system journal or **/var/log/messages**. Because viewing generates logs, which then print to the screen, this causes Session Recording to record this action, which generates more records, causing a loop of flooded output.
  You can use the following command to work around this problem:

  ```
  # journalctl -f | grep -v 'tlog-rec-session'
  ```

  You can also configure tlog to limit the output. For details, see **tlog-rec** or **tlog-rec-session** manual pages.

- To record users executing remote access commands, you must configure session recording for that user on the target host. For example, to record the following remote access command, you need to configure session recording for the **admin** user on the **client** host:

  ```
  ssh admin@client rm -f /some/file
  ```

- All recordings are lost on reboot because the **journal** is stored in-memory by default on RHEL 8. To export recordings see Exporting recorded sessions to a file .

CHAPTER 2. DEPLOYING SESSION RECORDING ON RHEL WEB CONSOLE

This section describes how to deploy the Session Recording solution on the Red Hat Enterprise Linux web console.

To be able to deploy the Session Recording solution you need to have the following packages installed:

- **tlog**
- SSSD
- **cockpit-session-recording**

2.1. INSTALLING TLOG

Install the **tlog** packages.

Procedure

- Use the following command:

```
# yum install tlog
```

2.2. INSTALLING COCKPIT-SESSION-RECORDING

The basic web console packages are a part of Red Hat Enterprise Linux 8 by default. To be able to use the Session Recording solution, you have to install the **cockpit-session-recording** packages and start or enable the web console on your system:

Procedure

1. Install **cockpit-session-recording**.

```
# yum install cockpit-session-recording
```

2. Start or enable the web console on your system:

```
# systemctl start cockpit.socket
# systemctl enable cockpit.socket
```

or

```
# systemctl enable cockpit.socket --now
```

2.3. ENABLING SESSION RECORDING FOR USERS AND GROUPS WITH SSSD FROM THE CLI

If you use SSSD for authentication, you can configure session recording for users and groups from the command line.

Procedure

1. Open the **sssd-session-recording.conf** configuration file:

```
# vi /etc/sssd/conf.d/sssd-session-recording.conf
```

NOTE

The **sssd-session-recording.conf** file is created automatically once you have opened the configuration page in the web console interface.

2. To specify the scope of session recording, enter one of the following values for the scope option:
   - **none** to record no sessions.
   - **some** to record only specified sessions.
   - **all** to record all sessions.
3. Optional: If you set the scope as **some** add the names of users and groups in comma-separated lists.
4. To enable the SSSD profile, run the following command:

```
# authselect select sssd with-files-domain
```

Example 2.1. SSSD configuration

In the following example users **example1** and **example2**, and group **examples** have session recording enabled.

```
[session_recording]
scope = some
users = example1, example2
groups = examples
```

2.4. ENABLING SESSION RECORDING FOR USERS AND GROUPS WITH SSSD FROM THE WEB UI

If you use SSSD for authentication, you can configure session recording for users and groups in the RHEL 8 web console.

Procedure

1. Connect to the RHEL 8 web console locally by entering **localhost:9090** or by entering your IP address **<IP_ADDRESS>:9090** into your browser.
2. Log in to the RHEL 8 web console.

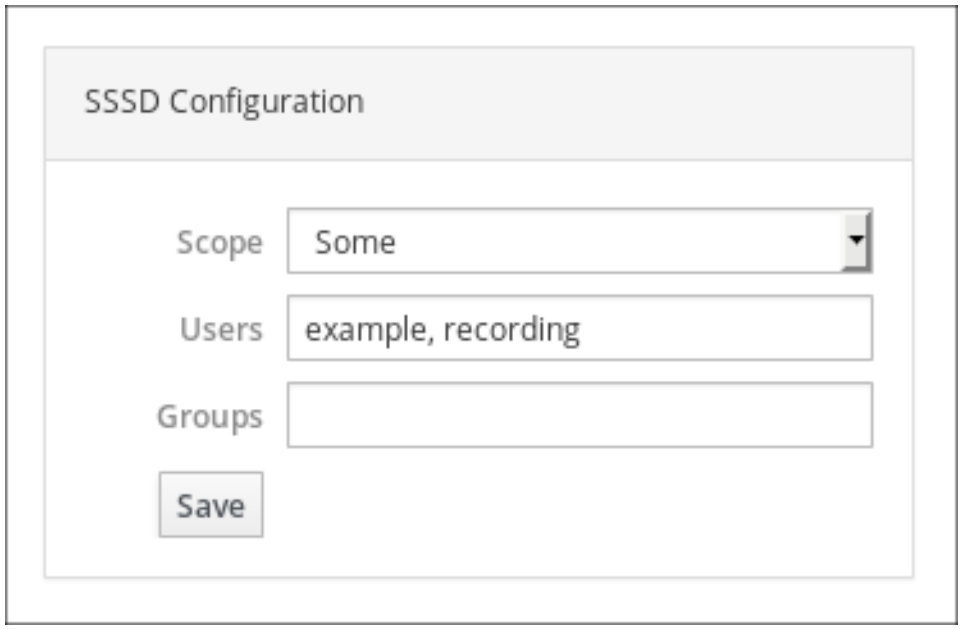
IMPORTANT

Your user has to have administrator privileges to be able to view recorded sessions.

3. Go to the Session Recording page in the menu on the left.
4. Click on the gear button in the right top corner.

5. Set your parameters in the SSSD Configuration table. Separate the lists of users and groups with commas.

Example 2.2. Configuration of recorded users with SSSD



2.5. ENABLING SESSION RECORDING FOR USERS WITHOUT SSSD

IMPORTANT

Red Hat recommends configuring your recorded users using SSSD, either from the command line or directly from the RHEL 8 web console.

Without SSSD, there is no centralized management for policies. Administrators must set the shell for each user on every system individually, which makes it difficult to scale across multiple systems. Group-based configurations or exclusions such as **exclude_users** and **exclude_groups** are also not available. Additionally, tools like Cockpit Session Recording are designed to work with SSSD and might not function as expected without it.

- To enable session recording without SSSD, change the shell of the user you want to record to **/usr/bin/tlog-rec-session**.

```
# sudo usermod -s /usr/bin/tlog-rec-session <user_name>
```

The system uses the configuration in the **tlog-rec-session.conf** file to determine the user's working shell.

2.6. EXPORTING RECORDED SESSIONS TO A FILE

You can export your recorded sessions and their logs and copy them.

The following procedure shows how to export recorded sessions on a local system.

Prerequisites

- Install the **systemd-journal-remote** package.

```
# yum install systemd-journal-remote
```

Procedure

1. Create a directory to store exported recording sessions, such as `/tmp/dir:

```
# mkdir /tmp/dir
```

2. Run the **journalctl -o export** command to export system journal entries related to tlog recordings:

```
# journalctl _COMM=tlog-rec _COMM=tlog-rec-sessio -o export | /usr/lib/systemd/systemd-journal-remote -o /tmp/dir/example.journal -
```

NOTE

The **COMM=tlog-rec-sessio** COMM name is shortened due to a 15 character limit.

CHAPTER 3. PLAYING BACK RECORDED SESSIONS

There are two methods for replaying recorded sessions:

- the **tlog-play** tool
- the RHEL 8 web console, also referred to as *Cockpit*.

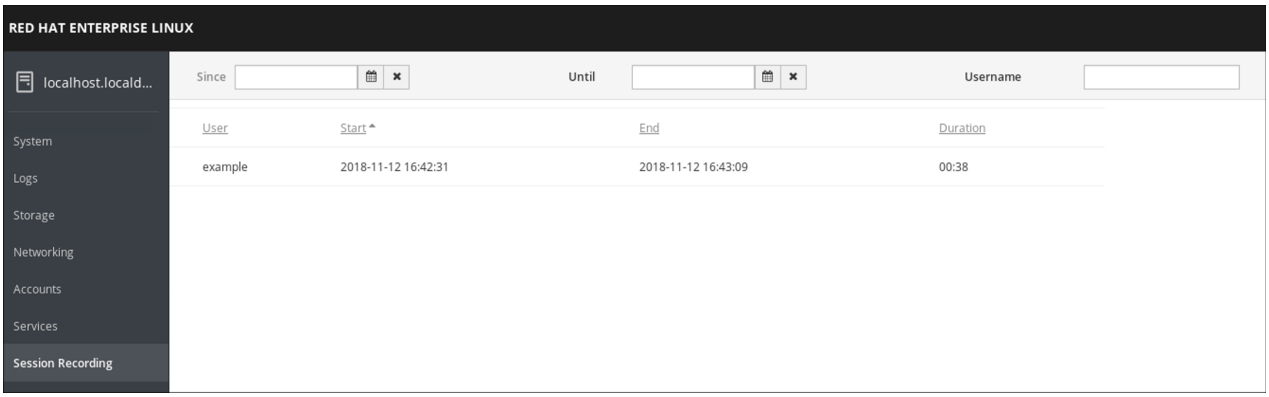
3.1. PLAYBACK WITH TLOG-PLAY

You can use the **tlog-play** tool to play back session recordings in a terminal. The **tlog-play** tool is a playback program for terminal input and output recorded with the **tlog-rec** tool. It reproduces the recording of the terminal it is under, but cannot change its size. For this reason the playback terminal needs to match the recorded terminal size for proper playback. The **tlog-play** tool loads its parameters from the **/etc/tlog/tlog-play.conf** configuration file. You can override those parameters with command line options described in the **tlog-play** manual pages.

3.2. PLAYBACK WITH THE WEB CONSOLE

The RHEL 8 web console has a whole interface for managing recorded sessions. You can choose the session you want to review directly from the Session Recording page, where the list of your recorded session is.

Example 3.1. Example list of recorded sessions



The web console player supports window resizing.

3.3. PLAYING BACK RECORDED SESSIONS WITH TLOG-PLAY

You can play back session recordings from exported log files or from the Systemd Journal.

Playing back from a file

You can play a session back from a file both during and after recording:

```
# tlog-play --reader=file --file-path=tlog.log
```

Playing back from the Journal

Generally, you can select Journal log entries for playback using Journal matches and timestamp limits, with the **-M** or **--journal-match**, **-S** or **--journal-since**, and **-U** or **--journal-until** options.

In practice however, playback from Journal is usually done with a single match against the **TLOG_REC** Journal field. The **TLOG_REC** field contains a copy of the **rec** field from the logged JSON data, which is a host-unique ID of the recording.

You can take the ID either from the **TLOG_REC** field value directly, or from the **MESSAGE** field from the JSON **rec** field. Both fields are part of log messages coming from the **tlog-rec-session** tool.

Procedure

1. You can play back the whole recording as follows:

```
# tlog-play -r journal -M TLOG_REC=<your-unique-host-id>
```

You can find further instructions and documentation in the **tlog-play** manual pages.

CHAPTER 4. CONFIGURING A SYSTEM FOR SESSION RECORDING BY USING RHEL SYSTEM ROLES

Use the **tlog** RHEL system role to record and monitor terminal session activities on your managed nodes in an automatic fashion. You can configure the recording to take place per user or user group by means of the **SSSD** service.

The session recording solution in the tlog RHEL system role consists of the following components:

- The **tlog** utility
- System Security Services Daemon (SSSD)
- Optional: The web console interface

4.1. CONFIGURING SESSION RECORDING FOR INDIVIDUAL USERS BY USING THE TLOG RHEL SYSTEM ROLE

Prepare and apply an Ansible playbook to configure a RHEL system to log session recording data to the **systemd** journal.

With that, you can enable recording the terminal output and input of a specific user during their sessions, when the user logs in on the console, or by SSH.

The playbook installs **tlog-rec-session**, a terminal session I/O logging program, that acts as the login shell for a user. The role creates an SSSD configuration drop file, and this file defines for which users and groups the login shell should be used. Additionally, if the **cockpit** package is installed on the system, the playbook also installs the **cockpit-session-recording** package, which is a **Cockpit** module that allows you to view and play recordings in the web console interface.

Prerequisites

- You have prepared the control node and the managed nodes
- You are logged in to the control node as a user who can run playbooks on the managed nodes.
- The account you use to connect to the managed nodes has **sudo** permissions on them.

Procedure

1. Create a playbook file, for example ~/**playbook.yml**, with the following content:

```
---
- name: Deploy session recording
  hosts: managed-node-01.example.com
  tasks:
    - name: Enable session recording for specific users
      ansible.builtin.include_role:
        name: redhat.rhel_system_roles.tlog
  vars:
    tlog_scope_sssd: some
    tlog_users_sssd:
      - <recorded_user>
```

tlog_scope_sssd: *<value>*

The **some** value specifies you want to record only certain users and groups, not **all** or **none**.

tlog_users_sssd: *<list_of_users>*

A YAML list of users you want to record a session from. Note that the role does not add users if they do not exist.

2. Validate the playbook syntax:

```
$ ansible-playbook --syntax-check ~/playbook.yml
```

Note that this command only validates the syntax and does not protect against a wrong but valid configuration.

3. Run the playbook:

```
$ ansible-playbook ~/playbook.yml
```

Verification

1. Check the SSSD drop-in file's content:

```
# cd /etc/sssd/conf.d/sssd-session-recording.conf
```

You can see that the file contains the parameters you set in the playbook.

2. Log in as a user whose session will be recorded, perform some actions, and log out.

3. As the **root** user:

   a. Display the list of recorded sessions:

```
# journalctl _COMM=tlog-rec-sessio
Nov 12 09:17:30 managed-node-01.example.com -tlog-rec-session[1546]:
{"ver":"2.3","host":"managed-node-
01.example.com","rec":"07418f2b0f334c1696c10cbe6f6f31a6-60a-e4a2","user":"demo-
user",...
...
```

You require the value of the **rec** (recording ID) field in the next step.

Note that the value of the **_COMM** field is shortened due to a 15 character limit.

   b. Play back a session:

```
# tlog-play -r journal -M TLOG_REC=<recording_id>
```

Additional resources

- **/usr/share/ansible/roles/rhel-system-roles.tlog/README.md** file
- **/usr/share/doc/rhel-system-roles/tlog/** directory

4.2. EXCLUDING CERTAIN USERS AND GROUPS FROM SESSION RECORDING BY USING THE TLOG RHEL SYSTEM ROLE

You can use the **tlog_exclude_users_sssd** and **tlog_exclude_groups_sssd** role variables from the **tlog** RHEL system role to exclude users or groups from having their sessions recorded and logged in the **systemd** journal.

The playbook installs **tlog-rec-session**, a terminal session I/O logging program, that acts as the login shell for a user. The role creates an SSSD configuration drop file, and this file defines for which users and groups the login shell should be used. Additionally, if the **cockpit** package is installed on the system, the playbook also installs the **cockpit-session-recording** package, which is a **Cockpit** module that allows you to view and play recordings in the web console interface.

Prerequisites

- You have prepared the control node and the managed nodes
- You are logged in to the control node as a user who can run playbooks on the managed nodes.
- The account you use to connect to the managed nodes has **sudo** permissions on them.

Procedure

1. Create a playbook file, for example ~/**playbook.yml**, with the following content:

```
---
- name: Deploy session recording excluding users and groups
  hosts: managed-node-01.example.com
  tasks:
    - name: Exclude users and groups
      ansible.builtin.include_role:
        name: redhat.rhel_system_roles.tlog
      vars:
        tlog_scope_sssd: all
        tlog_exclude_users_sssd:
          - jeff
          - james
        tlog_exclude_groups_sssd:
          - admins
```

tlog_scope_sssd: *<value>*

The value **all** specifies that you want to record all users and groups.

tlog_exclude_users_sssd: *<user_list>*

A YAML list of users user names you want to exclude from the session recording.

tlog_exclude_groups_sssd: *<group_list>*

A YAML list of groups you want to exclude from the session recording.

2. Validate the playbook syntax:

```
$ ansible-playbook --syntax-check ~/playbook.yml
```

Note that this command only validates the syntax and does not protect against a wrong but valid configuration.

3. Run the playbook:

```
$ ansible-playbook ~/playbook.yml
```

Verification

1. Check the SSSD drop-in file's content:

```
# cat /etc/sssd/conf.d/sssd-session-recording.conf
```

   You can see that the file contains the parameters you set in the playbook.

2. Log in as a user whose session will be recorded, perform some actions, and log out.

3. As the **root** user:

   a. Display the list of recorded sessions:

```
# journalctl _COMM=tlog-rec-sessio
Nov 12 09:17:30 managed-node-01.example.com -tlog-rec-session[1546]:
{"ver":"2.3","host":"managed-node-
01.example.com","rec":"07418f2b0f334c1696c10cbe6f6f31a6-60a-e4a2","user":"demo-
user",...
...
```

   You require the value of the **rec** (recording ID) field in the next step.

   Note that the value of the **_COMM** field is shortened due to a 15 character limit.

   b. Play back a session:

```
# tlog-play -r journal -M TLOG_REC=<recording_id>
```

Additional resources

- **/usr/share/ansible/roles/rhel-system-roles.tlog/README.md** file
- **/usr/share/doc/rhel-system-roles/tlog/** directory