

Sonarr System

System information, logs, scheduled tasks, and status monitoring for Sonarr administration and troubleshooting

Table of Contents

- ▶ [Table of Contents](#)
- ▶ [Status](#)
- ▶ [Health](#)
 - ▶ [System Warnings](#)
 - ▶ [Currently installed .NET Framework is old and unsupported](#)
 - ▶ [Currently installed .NET Framework is supported but upgrading is recommended](#)
 - ▶ [Currently installed mono version is old and unsupported](#)
 - ▶ [Package Maintainer Message](#)
 - ▶ [New update is available](#)
 - ▶ [Cannot install update because startup folder and/or UI folder are not writable by the user](#)
 - ▶ [Cannot install update because startup folder is in an App Translocation folder](#)
 - ▶ [Updating will not be possible to prevent deleting AppData on Update](#)
 - ▶ [Failed to resolve the IP Address for the Configured Proxy Host](#)
 - ▶ [Proxy Failed Test](#)
 - ▶ [System Time is off by more than 1 day](#)
 - ▶ [MediaInfo Library Could not be Loaded](#)
 - ▶ [Mono Legacy TLS enabled](#)
 - ▶ [Download Clients](#)
 - ▶ [No download client is available](#)
 - ▶ [Unable to communicate with download client](#)
 - ▶ [Download clients are unavailable due to failure](#)
 - ▶ [Enable Completed Download Handling](#)
 - ▶ [Downloading into Root Folder](#)
 - ▶ [Completed Download Handling is disabled](#)
 - ▶ [Download Client Removes Completed Downloads](#)
 - ▶ [Indexers](#)

- ▶ [No indexers available with automatic search enabled, Sonarr will not provide any automatic search results](#)
- ▶ [No indexers available with RSS sync enabled, Sonarr will not grab new releases automatically](#)
- ▶ [No indexers are enabled](#)
- ▶ [Enabled indexers do not support searching](#)
- ▶ [No indexers available with Interactive Search Enabled](#)
- ▶ [Indexers are unavailable due to failures](#)
- ▶ [Jackett All Endpoint Used](#)
 - ▶ [Solutions](#)
- ▶ [Media & Lists](#)
 - ▶ [Series Removed from TheTVDB](#)
 - ▶ [Lists are unavailable due to failures](#)
 - ▶ [Import List Missing Root Folder](#)
 - ▶ [Missing Root Folder](#)
 - ▶ [Missing root folder](#)
 - ▶ [Series Path Mount is Read Only](#)
- ▶ [Disk Space](#)
- ▶ [About](#)
- ▶ [More Info](#)
- ▶ [Tasks](#)
 - ▶ [Scheduled](#)
 - ▶ [Queue](#)
- ▶ [Backup](#)
- ▶ [Updates](#)
- ▶ [Events](#)
- ▶ [Log Files](#)

Status

Health

- ▶ This page contains a list of health checks errors. These health checks are periodically performed performed by Sonarr and on certain events. The resulting warnings and errors are listed here to give advice on how to resolve them.

System Warnings

Currently installed .NET Framework is old and unsupported

- ▶ Sonarr uses the .NET Framework. We need to build Sonarr against the lowest supported version still used by our users. Occasionally we increase the version we build against to be able to utilize new features. Apparently you haven't applied the appropriate Windows updates in a while and need to upgrade .NET to be able to use newer versions of Sonarr.
- ▶ Upgrading the .NET Framework is very straightforward on Windows, although it often requires a restart.

Currently installed .NET Framework is supported but upgrading is recommended

- ▶ Sonarr uses the .NET Framework. We need to build Sonarr against the lowest supported version still used by our users. Upgrading to newer versions allows us to build against newer versions and use new Framework features.
- ▶ Upgrading the .NET Framework is very straightforward on Windows, although it often requires a

Currently installed mono version is old and unsupported

- ▶ Sonarr v4 is written in .NET and v3 required Mono. Mono 5.20 is the absolute minimum for Sonarr.
- ▶ The upgrade procedure for Mono varies per platform.



Mono is no longer supported starting in Sonarr version 4.0

Currently installed SQLite version is not supported

- ▶ Sonarr stores its data in an SQLite database. The SQLite3 library installed on your system is too old. Sonarr requires at least version 3.9.0.



Note that Sonarr uses `libSQLite3.so` which may or may not be contained in a SQLite3 upgrade package.

Package Maintainer Message

- ▶ Your package maintainer has a message for you. This is controlled by your maintainer and not Sonarr.

New update is available

- ▶ Rejoice, the developers have released a new update. This generally means awesome new features and squashed piles of bugs (right?). Apparently you do not have Auto-Updating enabled, so you will have to figure out how to update on your platform. Pressing the Install button on the System => Updates page is probably a good starting point.



This warning will not appear if your current version is less than 14 days old

Cannot install update because startup folder and/or UI folder are not writable by the user

This means Sonarr will be unable to update itself. You'll have to update Sonarr manually or set the permissions on Sonarr's Startup directory (the installation directory) to allow Sonarr to update itself.

Cannot install update because startup folder is in an App Translocation folder

In macOS Sierra, Apple added a strange security feature called App Translocation (sometimes known as Gatekeeper Path Randomization) which means that after downloading an application, if you do not move the resulting application somewhere (anywhere!), with the Finder (you must use the Finder!), the application will be run as if it is located at a randomly chosen path by the system.

Updating will not be possible to prevent deleting AppData on Update

Sonarr detected that AppData folder for your Operating System is located inside the directory that contains the Sonarr binaries. Normally it would be C:\ProgramData for Windows and, ~/.config for linux.

Please look at System => Info to see the current AppData & Startup directories.

This means Sonarr will be unable to update itself without risking data-loss.

If you're on Linux, you will probably have to change the home directory for the user that is running Sonarr and copy the current contents of the ~/.config/Sonarr directory to preserve your database.

Failed to resolve the IP Address for the Configured Proxy Host

Review your proxy settings and ensure they are accurate

Ensure your proxy is up, running, and accessible

Proxy Failed Test

Your configured proxy failed to test successfully, review the HTTP error provided and/or check logs for more details.

System Time is off by more than 1 day

System time is off by more than 1 day. Scheduled tasks may not run correctly until the time is corrected

Review your system time and ensure it is synced to an authoritative time server and accurate

MediaInfo Library Could not be Loaded

MediaInfo Library could not be loaded. Sonarr requires MediaInfo (**libmediainfo**) to evaluate the video attributes of files.

Mono Legacy TLS enabled

Mono 4.x tls workaround still enabled, consider removing MONO_TLS_PROVIDER=legacy environment option.

Download Clients

No download client is available

A properly configured and enabled download client is required for Sonarr to be able to download media. Since Sonarr supports different download clients, you should determine which best matches your requirements. If you already have a download client installed, you should configure Sonarr to use it and create a category. See Settings => Download Client.

Unable to communicate with download client

Sonarr was unable to communicate with the configured download client. Please verify if the download client is operational and double check the url. This could also indicate an authentication error.

This is typically due to improperly configured download client. Things you can typically check:

Your download clients IP Address if its on the same bare metal machine this is typically 127.0.0.1

The Port number of that your download client is using these are filled out with the default port number but if you've changed it you will need to have the same one entered into Sonarr.

Ensure SSL encryption is not turned on if you're using both your Sonarr instance and your download client on a local network. See the SSL FAQ entry for more information.

Download clients are unavailable due to failure

One or more of your download clients is not responding to requests made by Sonarr. Therefore Sonarr has decided to temporarily stop querying the download client on it's normal 1 minute cycle, which is normally used to track active downloads and import finished ones. However, Sonarr will continue to attempt to send downloads to the client, but will in all likelihood fail.

You should inspect System => Logs to see what the reason is for the failures.

If you no longer use this download client, disable it in Sonarr to prevent the errors.

Enable Completed Download Handling

- ▶ Sonarr requires Completed Download Handling to be able to import files that were downloaded by the download client. It is recommended to enable Completed Download Handling. (Completed Download Handling is enabled by default...)

Docker bad remote path mapping

- ▶ This error is typically associated with bad docker paths within either your download client or Sonarr
- ▶ An example of this would be:
 - ▶ Download client: Download Path: /mnt/user/downloads:/downloads
 - ▶ Sonarr: Download Path: /mnt/user/downloads:/data
- ▶ Within this example the download client places its downloads into /downloads and therefore tells Sonarr when its complete that the finished episode is in /downloads. Sonarr then comes along and says "Okay, cool, let me check in /downloads" Well, inside Sonarr you did not allocate a /downloads path you allocated a /data path so it throws this error.
- ▶ The easiest fix for this is CONSISTENCY if you use one scheme in your download client, use it across the board.
- ▶ Team Sonarr is a big fan of simply using /data.
 - ▶ Download client: /mnt/user/data/downloads:/data/downloads

- ▶ Sonarr: /mnt/user/data:/data
- ▶ Now within the download client you can specify where in /data you'd like to place your downloads, now this varies depending on the client but you should be able to tell it "Yeah download client place my files into." /data/torrents (or usenet)/tv and since you used /data in Sonarr when the download client tells Sonarr it's done Sonarr will come along and say "Sweet, I have a /data and I also can see /torrents (or usenet)/tv all is right in the world."
- ▶ There are many great write ups: our wiki [Docker Guide](#) and TRaSH's [Hard links and Instant Moves \(Atomic-Moves\)](#) [🔗](#) . Now these guides place heavy emphasis on Hard links and Atomic moves, but the general concept of containers and how path mapping works is the core of these discussions.
- ▶ See [TRaSH's Remote Path Guide](#) [🔗](#) for more information.

Downloading into Root Folder

- ▶ Within the application, a root folder is defined as the configured media library folder. This is not the root folder of a mount. Your download client has an incomplete or complete (or is moving completed downloads) into your root (library) folder.
- ▶ This frequently causes issues - including data loss - and should not be done. To fix this change your download client so it is not placing downloads within your root folder. Note that 'placing' also includes if your download client category is set to your root folder or if NZBGet/SABnzbd have sort enabled and are sorting to your root folder.
- ▶ Please note that this check looks at all defined/configured root folders added not only root folders currently in use. In other words, the folder your download client downloads into or moves completed downloads to, should not be the same folder you have configured as your root/library/final media destination folder in the *arr application.
- ▶ Configured Root Folders (aka Library folders) can be found in [Settings => Media Management => Root Folders](#)
- ▶ One example is if your downloads are going into `/data/downloads` then you have a root folder set as `/data/downloads`.
- ▶ It is suggested to use paths like `/data/media/` for your root folder/library and `/data/downloads/` for your downloads.
- ▶ Review our [Docker Guide](#) and TRaSH's [Hard links and Instant Moves \(Atomic-Moves\) Guide](#) [🔗](#) for more information on the correct and optimal path setup. Note that the concepts apply for docker and non-docker




Your download folder where your download client places the downloads and your root/library folder MUST be separate. *Arr will import the file(s) from your download client's folder into your library. The download client should not move anything or download anything to your library.

Bad Download Client Settings

- ▶ The location your download client is downloading files to is causing problems. Check the logs for further information. This may be permissions or attempting to go from windows to linux or linux to windows without a remote path map.

Bad Remote Path Mapping

- ▶ The location your download client is downloading files to is causing problems. Check the logs for further information. This may be permissions or attempting to go from windows to linux or linux to windows without a remote path map. See [TRaSH's Remote Path Guide](#)  for more information.

Permissions Error

- ▶ Sonarr or the user sonarr is running as cannot access the location your download client is downloading files to. This is typically a permission issue.


Remote File was removed part way through processing

- ▶ A file accessible via a remote path map appears to have been removed prior to processing completing.

Remote Path is Used and Import Failed

- ▶ Check your logs for more info; Refer to our Troubleshooting Guides

Downloading into Root Folder

- ▶ Within the application, a root folder is defined as the configured media library folder. This is not the root folder of a mount. Your download client has an incomplete or complete (or is moving completed downloads) into your root (library) folder.
- ▶ This frequently causes issues - including data loss - and should not be done. To fix this change your download client so it is not placing downloads within your root folder. Note that 'placing' also includes if your download client category is set to your root folder or if NZBGet/SABnzbd have sort enabled and are sorting to your root folder.
- ▶ Please note that this check looks at all defined/configured root folders added not only root folders currently in use. In other words, the folder your download client downloads into or moves completed downloads to, should not be the same folder you have configured as your root/library/final media destination folder in the *arr application.
- ▶ Configured Root Folders (aka Library folders) can be found in [Settings => Media Management => Root Folders](#)
- ▶ One example is if your downloads are going into `/data/downloads` then you have a root folder set as `/data/downloads`.
- ▶ It is suggested to use paths like `/data/media/` for your root folder/library and `/data/downloads/` for your downloads.
- ▶ Review our [Docker Guide](#) and TRaSH's [Hard links and Instant Moves \(Atomic-Moves\) Guide](#)  for more information on the correct and optimal path setup. Note that the concepts apply for docker and non-docker



Your download folder where your download client places the downloads and your root/library folder MUST be separate. *Arr will import the file(s) from your download client's folder into your library. The download client should not move anything or download anything to your library.

Completed Download Handling is disabled

- ▶ It's required to use Completed Download Handling since it provides better compatibility for the unpacking and post-processing logic of various download clients. With it, Sonarr will only import a download once the download client reports it as ready.
- ▶ If completed download handling is disabled:
 - ▶ All imports must be handled manually
 - ▶ This healthcheck will persist and cannot be dismissed or disabled
 - ▶ Episodes will always be missing in Sonarr and eligible to be grabbed in perpetuity
 - ▶ Episodes manually moved and renamed by the user into the series' folder in Sonarr's library folder will possibly be picked up by the twice daily rescan if named properly and thus not be missing at that point.
 - ▶ The user will need to manually unmonitor or configure an On Grab custom script to unmonitor episodes.
 - ▶ FlexGet is likely a better tools for one's usecase if they do not wish to use Sonarr's media library management functionalites and simply require something to parse rss feeds and send releases to the download client



* Completed Download Handling only works properly if the download client and Sonarr are on the same machine since it gets the path to be imported directly from the download client otherwise a remote map is needed.

* Completed Download Handling requires Sonarr has read and write access to the completed download directory

Download Client Removes Completed Downloads

- ▶ It's required that your download client retain its history of completed downloads until Sonarr has imported them. If history retention is disabled then *Arr may not see the completed download before it is removed from the download client. Your download client should be set to keep (usenet) and pause not remove (torrents) downloads after completion: **either indefinitely or for at least 14 days.**
- ▶ Sabnzb: Switches => Post Processing => Keep Jobs **must** be set to 14 days or greater OR be set to Keep All History
- ▶ Removing completed downloads from your client can be managed by Sonarr and enabled via the download client settings in *Arr. Thus *Arr can ensure that your download client history is cleaned up.

Indexers

No indexers available with automatic search enabled, Sonarr will not provide any automatic search results

- ▶ Simply put you do not have any of your indexers set to allow automatic searches
Go into Settings > Indexers, select an indexer you'd like to allow Automatic Searches and then click save.

No indexers available with RSS sync enabled, Sonarr will not grab new releases automatically

- ▶ Sonarr uses the RSS feed to pick up new releases as they come along. [See the FAQ for more information](#)
- ▶ To correct this issue go to Settings > Indexers, select an indexer you have and enable RSS Sync.

No indexers are enabled

- ▶ Sonarr requires indexers to be able to discover new releases. All your indexers are disabled or you do not have any indexers added.

Enabled indexers do not support searching

- ▶ None of the indexers you have enabled and available support searching. This means Sonarr will only be able to find new releases via the RSS feeds. But searching for series (either Automatic Search or Manual Search) will never return any results. The only way to remedy it is to add another indexer.

No indexers available with Interactive Search Enabled

- ▶ None of the indexers you have enabled and available support interactive searching. This means the application will only be able to find new releases via the RSS feeds or an automatic search.

Indexers are unavailable due to failures

- ▶ Errors occurs while Sonarr tried to use one of your indexers. To limit retries, Sonarr will not use the indexer for an increasing amount of time (up to 24h).
- ▶ This mechanism is triggered if Sonarr was unable to get a response from the indexer (could be caused DNS, proxy/vpn connection, authentication, or an indexer issue), or was unable to fetch the nzb/torrent file from the indexer. Please inspect the logs to determine what kind of error caused the problem.
- ▶ You can prevent this warning by disabling the affected indexer.
- ▶ Run a **Test** on the indexer to force Sonarr to recheck the indexer; please note that the Health Check warning will not always disappear immediately and you may need to run the **Check Health** Task

Jackett All Endpoint Used

- ▶ The Jackett /all endpoint is convenient, but that is its only benefit. Everything else is potential problems, so adding each tracker individually is now required.
- ▶ [Even Jackett's Devs says it should be avoided and should not be used.](#) [↗](#)
- ▶ Using the /all endpoint has no advantages, only disadvantages:
 - ▶ you lose control over indexer specific settings (categories, search modes, etc.)
 - ▶ mixing search modes (IMDB, query, etc.) might cause low-quality results
 - ▶ indexer specific categories (>= 100000) cannot be used.
 - ▶ slow indexers will slow down the overall result
 - ▶ total results are limited to 1000
 - ▶ if one of the trackers in /all returns an error, *Arr will disable it and now you do not get any results.

Solutions

- ▶ Add each tracker in Jackett manually as an indexer in *Arr
- ▶ Check out [Prowlarr](#) which can sync indexers to *Arr and from the Lidarr/Radarr/Readarr development team.
- ▶ Check out [NZBHydra2](#) [↗](#) which can sync indexers to *Arr. But do not use their single aggregate endpoint and use `multi` if sync will be used.

Media & Lists

Series Removed from TheTVDB

- ▶ The affected series was/were removed from [The TVDb](#) [↗](#) . This usually happens because the series is a duplicate or considered part of a different series. Alternatively, TVDb may treat it as a Series; hopefully TMDb does as well so [Radarr](#) can be used instead. To correct this you will need to remove the affected series and add the correct series if applicable.

Lists are unavailable due to failures

- ▶ Typically this simply means that Sonarr is no longer able to communicate via API or via logging in to your chosen list provider. Your best bet if the problem persists is to contact them in order to rule them out, as their systems maybe overloaded from time to time.
- ▶ Review System => Events filtered for Warning (Warning & Errors) to see the historical failures or check logs for details.

Import List Missing Root Folder

- ▶ One or more of your import lists are configured to a root folder that is not accessible to Sonarr.
- ▶ This may be permissions issues, a missing mount, or simply needing to update the lists after reorganizing your setup.

Missing Root Folder

- ▶ A root folder is added to Sonarr and does not exist or is not accessible
- ▶ This error is typically identified if a Series is looking for a root folder but that root folder is no longer available.
- ▶ This error may also be if a list is still pointed at a root folder but that root folder is no longer available.
- ▶ If you would like to remove this warning simply find the series that is still using the old root folder and edit it to the correct root folder.
- ▶ Easiest way to find the problem series is to:
 - ▶ Go to the Mass Editor Tab
 - ▶ Create a custom filter with the old root folder path
 - ▶ Select mass edit on the top bar and from the Root Paths drop down select the new root path that you want these series to be moved to.
 - ▶ Next you will receive a pop-up that states Would you like to move the Series folders to 'root path' ? This will also state This will also rename the Series folder per the Series folder format in settings. Simply select No if

the you do not want Lidarr to move your files

- ▶ Run the Check Health Task in System => Tasks

Series Path Mount is Read Only

A mount containing a series path is read only and is not writable by the user Sonarr is running as.







Disk Space

- ▶ This section will show you available disk space
- ▶ In docker this can be tricky as it will typically show you the available space within your Docker image.

About

- ▶ This will tell you about your current install of Sonarr.

More Info

- ▶ Home Page: Sonarr's [home page](#) 
- ▶ Wiki: You're here already.
- ▶ Reddit: [r/sonarr](#) 
- ▶ Discord: Join our [discord](#) 
- ▶ Donations: If you're feeling generous and would like to donate, click [the Sonarr donation page](#) 
- ▶ Source: [GitHub](#) 
- ▶ Feature Requests: Got a great idea drop it [on the GitHub issues page](#) 

Tasks

Scheduled

- ▶ This section lists all scheduled tasks that Sonarr runs
- ▶ Application Check Update - This will run every on the displayed schedule in the UI, checking to see if Sonarr is on the most current version then triggering the update script to update Sonarr. Settings => Update



Note: If on Docker this will not update your container as a new image will need to be downloaded.

- ▶ Backup - This will run a backup of your Sonarr's database on a set schedule more details on this can be found here. More information about backups can be found System => Backups.

- ▶ Check Health - Check Health will run on the displayed schedule in the UI checking the overall health of your Sonarr. To see a list of possible health related issues see the Wiki Entry on Health Checks.
- ▶ Clean Up Recycle Bin - The recycling bin will be cleared out on the displayed schedule in the UI. This will only be used if the recycling bin is set in File Management
- ▶ Housekeeping - On the displayed schedule in the UI this will dust out all the cobwebs, sweeps and vacuums the floors, mops, shines, and even makes nice neat little folded notes just for you. But does not take out the trash. That it just was not paid enough for.
- ▶ Import List Sync - On the displayed schedule in the UI this will run your Lists and import any possible new series. More info about lists can be found Settings => Lists.
- ▶ Messaging Cleanup - On the displayed schedule in the UI this cleans up those messages that appear in the bottom left corner of Sonarr
- ▶ Refresh Monitored Downloads - This goes through and refreshes the downloads queue located under Activity. Essentially ping your download client to check for finished downloads.
- ▶ Refresh Series - This goes through and refreshes all the metadata for all monitored and unmonitored series
- ▶ RSS Sync - This will run the RSS Sync. This can be changed in Settings => Indexers => Options. More information on the RSS function can be found on our [FAQ](#)



All these tasks can be ran manually outside their scheduled times by hitting the icon to the far right of each of the tasks.

Queue

The queue will show you running and upcoming tasks as well as a history of recently ran tasks as well as how long those tasks took.

Backup



If you're looking for how to back/restore your Sonarr instance click [the Sonarr backup FAQ](#).

Within the Backup section you will be presented with previous backups (unless you have a fresh install that hasn't made any backups).

- ▶ Backup Now - This option will trigger a manual backup of your Sonarr's database
- ▶ Restore Backup - This will open a new screen to restore from a previous backup
 - ▶ By selecting Choose File this will prompt your browser to open a dialog box to restore from a Sonarr Zip backup
- ▶ If you have any previous backups and would like to download them from Sonarr to be placed in a non standard location you simply can select one of these files to download them

- ▶ Off to the right of each of the previous download you have two options.
 - ▶ Restore (Clock Icon) - To restore from a previous backup - This will open a new window to confirm you want to restore from this backup
 - ▶ Delete (Trashcan) - To delete a previous backup

Updates

The update screen will show the past 5 updates that have been made as well as the current version you are on. This page will also display the update notes from the Developers telling you what has been fixed or added to Sonarr (Rejoice!)



A Maintenance Release contains bug fixes and other various improvements. Take a look at the commit history on Github for specifics.

Events

The events tab will show you what has been happening within your Sonarr. This can be used to diagnose some light issues. However, this does not replace Trace Logs discussed in Logging.



Events are the equivalent of INFO Logs.

- ▶ Components - This column will tell you what component within Sonarr has been triggered
- ▶ Message - This column will tell you what message as been sent from the component from the previous column.
- ▶ Gear Icon - This option will allow you to change how many Components/Messages are displayed per page (Default is 50)
- ▶ Options at the top of the page
 - ▶ Refresh - This option will refresh the current page, pulling a new events log
 - ▶ Clear - This will clear the current events log allowing you to start from fresh

Log Files

- ▶ This page will allow you to download and see what current log files are available for Sonarr.
- ▶ On the top row there are several options to allow you to control your log files.
- ▶ The top row on the far left there is a dropdown that will allow you to switch from Log files and Updater Log Files

- ▶ Log Files - The bread and butter of any support issue more on log files can be found here.
- ▶ Updater Log Files - This will show the log files associated with Sonarr's updater script



If you're on docker this will be empty as you should be updating by downloading a new docker image

- ▶ Refresh - This will refresh the current page and display any newly created logs
- ▶ Delete - This will clear all logs allowing you to start from fresh
- ▶ File Name - This will display the file name associated with the log
- ▶ Last Written - This is the local time that this particular log file was written to.
 - ▶ Sonarr uses rolling log files limited to 1MB each. The current log file is always sonarr.txt, for the the other files sonarr.0.txt is the next newest (the higher the number the older it is) up to 51 log files total. This log file contains `fatal`, `error`, `warn`, and `info` entries.
 - ▶ When Debug log level is enabled, additional sonarr.debug.txt rolling log files will be present, up to 51 files. This log files contains `fatal`, `error`, `warn`, `info`, and `debug` entries. It usually covers a ~40h period.
 - ▶ When Trace log level is enabled, additional sonarr.trace.txt rolling log files will be present, up to 51 files. This log files contains `fatal`, `error`, `warn`, `info`, `debug`, and `trace` entries. Due to trace verbosity it only covers a couple of hours at most.