Troubleshooting

- Calendar
  - Missing Shared Calendars
- Contacts
  - Unable to update contacts or events
- Mail
  - Autoconfig for your mail domain fails
  - Database insert problems on MySQL
  - Export threading data
  - Get account IDs
  - Issues connecting to Outlook.com
  - Logging the IMAP/SMTP/Sieve connections
    - Per mail account
    - Globally
  - Timeout and other connectivity issues
  - Manual account synchronization and threading

Calendar

Missing Shared Calendars

Problem:
A user should have access to a shared calendar, but the calendar is not displayed in Nextcloud Calendar or other CalDAV clients (e.g., DAVx[5] or Thunderbird).

Affected Versions:

- Nextcloud Server 31.0.5 and below
- Nextcloud Server 30.0.11 and below

Possible Reason:
A bug in previous versions of Nextcloud Server could mistakenly add a calendar unshare instead of removing the share permission. For example, a user has read access through a group membership, and the owner grants permission to a single user to modify a calendar. When the modify permission is removed, an unshare record is incorrectly created.

Troubleshooting Steps:

1. **Check for Hidden Calendars:**
   It's possible for a user to hide a calendar. Please check in Nextcloud Calendar if the missing calendar is listed in the "hidden" section. If it is, check the box in front of the calendar to enable it again.
2. **List Calendar Shares:**
   Run the command `occ dav:list-calendar-shares <uid>` to list all shares for a user. Look for lines with the Calendar URI/Calendar Name of the missing calendar and Permissions = Unshare. If such a line exists, but the user should have access, you have three options:

1. **Create a User Share and Remove It Again:**
   In most cases, sharing the calendar with the user again (as an individual/user share) will correct the state in the database.
2. **Remove All Calendar Unshares for a User:**
   Run `occ dav:clear-calendar-unshares <uid>`.
3. **Delete Specific Unshares:**
   Some users may have many calendar unshares, so it might be easier to delete only the unwanted unshare. The `Share Id` refers to the ID of a row in the `oc_dav_shares` database table. Delete the row with the matching ID to remove the unshare.

Why isn't there an automated migration to correct the problem?
Unsharing a calendar is a feature, and with the given information, we cannot determine if a calendar was unshared intentionally or as a result of the bug.

Contacts

Unable to update contacts or events

If you get an error like:

```
PATCH https://example.com/remote.php/dav HTTP/1.0 501 Not Implemented
```

it is likely because of a misconfigured web server. Please refer to Troubleshooting WebDAV for troubleshooting steps.

Mail

Autoconfig for your mail domain fails

If autoconfiguration for your domain fails, you can create an autoconfig file and place it as `https://autoconfig.yourdomain.tld/mail/config-v1.1.xml`. For more information please refer to Mozilla's documentation.

Database insert problems on MySQL

If the mail app fails to insert new rows for messages (oc_mail_messages), recipients (oc_mail_recipients) or similar tables, you are possibly not using the 4 byte support.

See Enabling MySQL 4-byte support for how to update your database configuration.

Export threading data

If you encounter an issue with threading, e.g. messages that belong to the same conversation thread don't show up as one, you can export the data the algorithm will use to build threads. We are dealing with sensitive data here, but the command will optionally redact the data with the `--redact` switch. The exported data will then only keep the original database IDs, the rest of the data is randomized. This format does not the export message details, it still contains metadata about how many messages you have and how they relate. Please consider this before posting the data online.

```
sudo -E -u www-data php occ mail:account:export-threads 1393
```

> “□□
>
> 1393 represents the account ID.

The output will look similar to this:

```
[
    {
        "subject": "83379f9bc36915d5024de878386060b5@redacted",
```

```
        "id": "2def0f3597806ecb886da1d9cc323a7c@redacted",
        "references": [],
        "databaseId": 261535
    },
        {
        "subject": "Re: 1d4725ae1ac4e4798b541ca3f3cdce6e@redacted",
        "id": "ce9e248333c44a5a64ccad26f2550f95@redacted",
        "references": [
            "bc95cbaff3abbed716e1d40bbdaa58a0@redacted",
            "8651a9ac37674907606c936ced1333d7@redacted",
            "4a87e94522a3cf26dba8977ae901094d@redacted",
            "a3b30430b1ccb41089170eecbe315d3a@redacted",
            "8e9f60369dce3d8b2b27430bd50ec46d@redacted",
            "46cfa6e729ff329e6ede076853154113@redacted",
            "079e7bc89d69792839a5e1831b1cbc80@redacted",
            "079e7bc89d69792839a5e1831b1cbc80@redacted"
        ],
        "databaseId": 262086
    },
    {
        "subject": "Re: 1d4725ae1ac4e4798b541ca3f3cdce6e@redacted",
        "id": "8dd0e0ef2f7ab100b75922489ff26306@redacted",
        "references": [
            "bc95cbaff3abbed716e1d40bbdaa58a0@redacted",
            "8651a9ac37674907606c936ced1333d7@redacted",
            "4a87e94522a3cf26dba8977ae901094d@redacted",
            "a3b30430b1ccb41089170eecbe315d3a@redacted",
            "8e9f60369dce3d8b2b27430bd50ec46d@redacted",
            "46cfa6e729ff329e6ede076853154113@redacted",
            "079e7bc89d69792839a5e1831b1cbc80@redacted",
            "ce9e248333c44a5a64ccad26f2550f95@redacted",
            "ce9e248333c44a5a64ccad26f2550f95@redacted"
        ],
        "databaseId": 262087
    }
]
```


It's recommended practice to pipe the export into a file, which you can later share with the Mail app community and developers:

```
sudo -E -u www-data php occ mail:account:export-threads 1393 | gzip -c > /tmp/nextcloud-mail-threads-1393.json.gz
```

Get account IDs

For many troubleshooting instructions you need to know the id of a mail account. You can acquire this through the database, but it's also possible to utilize the account export command of occ if you know the UID of the user utilizing the mail account:

```
sudo -E -u www-data php occ mail:account:export user123
```

The output will look similar to this:

```
Account 1393:
- E-Mail: christoph@domain.com
- Name: Christoph Wurst
- IMAP user: christoph
- IMAP host: mx.domain.com:993, security: ssl
- SMTP user: christoph
- SMTP host: mx.domain.com:587, security: tls
```

In this example, `1393` is the account ID.

Issues connecting to Outlook.com

If you can not access your Outlook.com account try to enable the Two-Factor Verification and set up an app password, which you then use for the Nextcloud Mail app.

Logging the IMAP/SMTP/Sieve connections

The Nextcloud Mail app offers an extensive logging system to make it easier identifying and tracking down bugs. As this may include sensitive data, be sure to remove or mask them before posting them publicly.

Per mail account

> *New in version 5.1.0:* Nextcloud 30 or newer

Starting with version 5.1.0 of the mail app, you can enable logging of outgoing IMAP/SMTP/Sieve connections limited to a specific mail account. As that saves a lot of resources on your system, this is the preferred method for debugging issues regarding IMAP/SMTP/Sieve connections.

First, you need to get the accountId for the mail account you like to enable debug logging on.

Please see Get account IDs for more.

Once you know the accountId of the mail account in question, you can use it to enable debug logging by running the following command on the server:

```
sudo -E -u www-data php occ mail:account:debug <accountId> --on
```

All subsequent outgoing connections made by the mail app will then be written to the `data` directory. The file naming follows the following format: `mail-{{userId}}-{{accountId}}-{{protocol}}.log` (e.g., mail-admin-49-imap.log).

The debug logging for that specific account can be disabled once you've collected the necessary data by running the following command on the server:

```
occ mail:account:debug <accountId> --off
```

Globally

This enables logging of the IMAP/SMTP/Sieve connections for **all** mail accounts configured on the server. This should be used with caution as it can put a lot of strain on large environments.

> *New in version 5.1.0:* Nextcloud 30 or newer

To enable the global debug logging on versions 5.1.0 and above, just run the following command on the server:

```
sudo -E -u www-data php occ config:system:set app.mail.debug --value true --type bool
```

All subsequent outgoing connections made by the mail app will then be written to the `data` directory. The file naming follows the following format: `mail-{{userId}}-{{accountId}}-{{protocol}}.log` (e.g., mail-admin-49-imap.log).

The global debug logging can be disabled once you've collected the necessary data by running the following command on the server:

```
sudo -E -u www-data php occ config:system:set app.mail.debug --value false --type bool
```

>
>
> The following steps only apply to version 1.6.2 up to 5.0.8. Restrict logging of outgoing connections to a specific mail account is not available there.

> *New in version 1.6.2:* Nextcloud 20 or newer

To enable the global debug logging, it's necessary to enable both the debug mode as well as debug logging for the whole nextcloud instance by running the following commands on the server:

```
sudo -E -u www-data php occ config:system:set debug --value true --type bool
sudo -E -u www-data php occ config:system:set loglevel --value 0 --type int
```

All subsequent outgoing connections made by the mail app will then be written to the `data` directory. The file naming follows the following format: `horde_{{protocol}}.log` (e.g., horde_imap.log).

Once you've collected the necessary data, it's highly recommended to disable the debug mode as well as resetting the loglevel to the default value by running the following commands:

```
sudo -E -u www-data php occ config:system:set debug --value false --type bool
sudo -E -u www-data php occ config:system:set loglevel --value 2 --type int
```

Timeout and other connectivity issues

You can use OpenSSL to test and benchmark the connection from your nextcloud host to the IMAP/SMTP host.:

```
openssl s_time -connect imap.domain.tld:993
```

The output should look similar to this:

```
Collecting connection statistics for 30 seconds
**********************************************************************************************
**********************************************************************************************
**********************************************************************************************
**********************************************************************************************
**********************************************************************************************
```

```
*************

483 connections in 0.94s; 513.83 connections/user sec, bytes read 0
483 connections in 31 real seconds, 0 bytes read per connection


Now timing with session id reuse.
starting
**************************************************************************************************
**************************************************************************************************
**************************************************************************************************
**************************************************************************************************
**************************************************************************************************
****************************

497 connections in 0.97s; 512.37 connections/user sec, bytes read 0
497 connections in 31 real seconds, 0 bytes read per connection
```

Manual account synchronization and threading

To troubleshoot synchronization or threading problems it's helpful to run the sync from the command line while the user does not use the web interface (reduces chances of a conflict):

```
sudo -E -u www-data php occ mail:account:sync -vvv 1393
```

> “□□
>
> 1393 represents the account ID.

The command offers a `--force` option. Use it wisely as it doesn't perform the same path a typical web triggered sync request would do.

The output will look similar to this:

```
[debug] Skipping mailbox sync for Archive
[debug] Skipping mailbox sync for Archive.2020
[debug] partial sync 1393:Drafts - get all known UIDs took 0s
```

```
[debug] partial sync 1393:Drafts - get new messages via Horde took 0s
[debug] partial sync 1393:Drafts - persist new messages took 0s
[debug] partial sync 1393:Drafts - get changed messages via Horde took 0s
[debug] partial sync 1393:Drafts - persist changed messages took 0s
[debug] partial sync 1393:Drafts - get vanished messages via Horde took 0s
[debug] partial sync 1393:Drafts - persist new messages took 0s
[debug] partial sync 1393:Drafts took 0s
[debug] partial sync 1393:INBOX - get all known UIDs took 0s
[debug] partial sync 1393:INBOX - get new messages via Horde took 0s
[debug] partial sync 1393:INBOX - classified a chunk of new messages took 1s
[debug] partial sync 1393:INBOX - persist new messages took 0s
[debug] partial sync 1393:INBOX - get changed messages via Horde took 1s
[debug] partial sync 1393:INBOX - persist changed messages took 0s
[debug] partial sync 1393:INBOX - get vanished messages via Horde took 0s
[debug] partial sync 1393:INBOX - persist new messages took 0s
[debug] partial sync 1393:INBOX took 2s
[debug] Skipping mailbox sync for Sent
[debug] Skipping mailbox sync for Sentry
[debug] Skipping mailbox sync for Trash
[debug] Account 1393 has 19417 messages for threading
[debug] Threading 19417 messages - build ID table took 1s
[debug] Threading 19417 messages - build root container took 0s
[debug] Threading 19417 messages - free ID table took 0s
[debug] Threading 19417 messages - prune containers took 0s
[debug] Threading 19417 messages - group by subject took 0s
[debug] Threading 19417 messages took 1s
[debug] Account 1393 has 9839 threads
[debug] Account 1393 has 0 messages with a new thread IDs
62MB of memory used
```

□□□ #3
□ cocobook-sync □□□ 2026-10-28 16:30:02 UTC
□ cocobook-sync □□□ 2026-10-29 01:09:25 UTC