

Microsoft Exchange: Exporting Mailboxes to PST

In some cases, you may need to export certain Microsoft Exchange mailboxes to PST files for backup or export purposes. In this guide, I will explain how to perform bulk exports of user mailboxes in Microsoft Exchange.

Backup Exchange Mailbox to PST

Before exporting or backing up data from a user mailbox, you need to do the following.

1. Gain necessary Exchange permissions. Exchange server by default does not allow you to export mailboxes. It is even prohibited for *Exchange Organization Administrators* members.
2. Prepare a storage folder for data export. It is better to use a network file path - we will discuss it below.
3. Create a mailbox export request using the PowerShell command.

You can gain the necessary rights by running the following command in **Exchange Management Shell** with the account having security groups management rights:

```
New-ManagementRoleAssignment -Role "Mailbox Import Export" -User DOMAIN\USER
```

Change *DOMAIN\USER* to the account from which you will perform the export.

Create a network share for the folder you will use as a PST storage. Since a mailbox export can perform on the Exchange server with a Client Access Server (CAS) role, it is necessary to ensure access to the selected folder. A network share is a good option if you use a few CAS servers or prefer to store mailbox data somewhere outside. Assign "Write" permissions to the network share for the user or group with "*Mailbox Import Export*" rights.

Let's export the data using **Exchange Management Shell**.

How to Export Mailbox to PST with PowerShell Script

1. MS Exchange has the command `New-MailboxExportRequest`, which you can use in Exchange Management Shell:
New-MailboxExportRequest -Mailbox user_name -FilePath "\\server_name\ShareName\Filename.pst"
2. Replace *user_name* with target user account name, and paste the network share URI next to the *FilePath* option. You can also use the following options for better exporting customization:
 - **Name** used to name the particular request if you want to execute a few requests per single mailbox. It will just simplify the status check.
 - **ExcludeFolders** and **-IncludeFolders** allows managing whether Exchange will export certain mailbox folders or not.

3. Read the full list of parameters on the [command info page](#).

Note: if you get the “commandlet not found” error after entering the above command, you need to launch Exchange Management Shell using elevated privileges. Just right-click the Exchange Management Shell icon and choose **Run as administrator**.

The Exchange will queue the export request and execute it as soon as possible. If the target mailbox is big or you need to export a few mailboxes - check the exporting status using this command:

Get-MailboxExportRequestStatistics

4. After the export has completed, you will see the desired PST file by the network path specified.
5. If you need to export a few mailboxes at once, then you can create a simple script:

First, let’s create a PowerShell variable to store the list of usernames:

```
$userMailboxes = "user1", "user2", "userN"
```

6. Use any variable name instead of user Mailboxes, and list all exportable mailbox users separated by commas.
If you would like to export all exchange mailboxes, you can use the following command instead of login names:

Use the variable from the previous step as a script parameter:

```
foreach ($User in ($mailboxusers)) { New-MailboxExportRequest -Mailbox $User -FilePath "\\localhost\MailboxExports\${$User}.pst" }
```

7. Replace **\$mailboxusers** with the variable you created (**\$User** means every entry in the variable). Then replace the target file path before the “**`\${\$User}.pst**” part - it helps to name every single PST file with a user account name.
8. You can run the script using Windows Task Scheduler to create mailbox-level backups regularly. But keep in mind that mailbox export is a time-consuming task which puts an additional load on the Exchange server with a CAS role.

Why PST?

In most cases, PST export is not a good way to back up Exchange data because it is slow, complex, and places heavy loads on Exchange servers. That said, it is a good solution if you want simply to make a few copies of a mailbox and move them outside, or if you need to migrate data. In addition, you can implement an additional layer of data protection using PST exports, since this approach allows you to recover calendar events, individual emails, contacts and so on without performing a database restoration process.

In this scenario, you can use third-party tools that support [item-level access to the Exchange](#) backup, or you can write a script to perform the necessary actions using built-in Exchange PowerShell capabilities.

Keep PST Files Safe

If you plan to create an item-level backup using a set of exported PST files, then you should also ensure that your storage is reliable, since large backup files could be corrupted or overwritten by subsequent backup tasks. To ensure greater reliability, we suggest implementing a [3-2-1 backup strategy](#), which allows you to maximize your organization's data resilience in case of disaster. The key is keeping several backup copies of data, spread across separate storage sites.

When choosing your own implementation of a 3-2-1 strategy, keep in mind that most popular cloud storage providers (AWS, Microsoft Azure, Google Cloud Storage platform, etc.) offer [a geo-distributed architecture](#) that allows them to keep data intact even after massive natural disasters.

Conclusion

You can use the Exchange mailbox export feature to move data to or from an Exchange organization. There is also a script that allows you to automate the task for a set of mailboxes.

However, if you are choosing this export feature as a backup tool for Exchange items backup, you should also ensure PST files protection. You can even integrate PST exported files into your existing backup infrastructure.