Microsoft 365 Plugin

Bacula Systems Documentation

Contents

Contents

1 Overview

This whitepaper presents how to protect the most relevant elements of Microsoft 365 services using **Bacula Enterprise**.

1.1 Features

The **Bacula Enterprise Microsoft 365 Plugin** is a very easy to deploy and configure plugin supporting the following M365 services:

- OneDrive
- Emails
- Mailbox settings
- Sharepoint Online
- Calendars
- Contacts
- OneNote
- Tasks
- Teams
- Chats

It is shipped with advanced parallelization, resiliency, and flexibility features in addition to covering most of the possible M365 use case scenarios. A full feature list is presented below:

- Common features
  - Microsoft Graph API based backups
  - Multi-service backup in the same task
  - Multi-service parallelization capabilities
  - Multi-thread single service processes
  - Automatic parallelization of fetching processes
  - Generation of user-friendly report for restore operations
  - Network resiliency mechanisms
  - Latest Microsoft Authentication mechanisms
  - Discovery/List/Query capabilities
  - Restore objects to Microsoft 365
    - To original entity
    - To any other entity
    - To a different tenant (cross-tenant restore)
  - Restore any object to filesystem
  - Owner data protection feature:
    - Notify data owner about restore actions of his data
    - Do not proceed further until he enters his M365 credentials

  - Incremental & Differential backups
    - Advanced delta function for improved performance (for selected services)
- Backup and Restore of Exchange Online Mailboxes
  - Mailfolder, message and attachment granularity for restore
  - Email addresses and mailfolders selection capabilities for backup
  - Mailbox settings protection
  - Folder rules protection
  - Restore objects to Microsoft 365 or to any file-system
  - Restore MIME messages to any filesystem
  - Incremental & Differential backup
  - Support for user mailboxes and shared mailboxes
  - User categories protection
  - Fully indexed information into Bacula Catalog
  - Advanced search capabilities for restore operations
  - Ability to filter input data by date
  - Ability to exclude message fields from backup or from index
  - Exclude private or spam messages through powerful filtering capability
- Backup and Restore of OneDrive for Business & Sharepoint Document libraries
  - Backup and Restore of User drives
  - Backup and Restore of Group drives
  - Backup and Restore of Sharepoint document libraries
    - Include/Exclude system libraries
    - Include/Exclude hidden libraries
  - Backup main entity drive unit, but also any other unit
  - Advanced selection capabilities
    - Target entities (Users/Groups/Sites)
      - List Include/List Exclude/Regex include/Regex Exclude…
    - Folder selection capabilities for backup
      - List Include/List Exclude
    - File selection capabilities for backup
      - Regex include/Regex Exclude…
    - Drive unit selection capabilities
  - Folder and file granularity for restore
  - Computed hash check at backup and restore time
  - Backup and restore of permissions shares
  - Backup and restore of shared elements to each entity

  - Backup and restore of OneDrive file versions
- Backup and Restore of Sharepoint Sites
  - Backup of Site Objects (MS Graph Object)
    - Backup Site Sharing permissions
  - Backup of full Site Templates (PnP Powershell Provisioning):
    - Site metadata
    - Lists metadata
    - ListItems metadata
    - WebPages metadata
    - DocumentLibraries metadata
    - Support for Sub-Sites
  - Restore backed up sites as new sites using Site Template (PnP Powershell Provisioning)
  - Backup/Restore Lists Objects (MS Graph Object)
  - Backup/Restore ListItems (MS Graph Object)
  - Backup/Restore Document Libraries Drive Items
- Backup and Restore of Contacts/People
  - Backup/restore Contacts
  - Backup/restore groups of contacts
  - Backup Organizational contacts (Read-only)
- Backup and Restore of Tasks
  - Backup/restore User todo tasks
  - Backup/restore Teams Planner tasks
- Backup and Restore of Calendars
  - Backup/Restore user calendars
  - Backup/Restore user groups of calendars
  - Backup/Restore groups calendar
    - Calendar permissions
  - Backup/Restore Events
    - Support for Attachments
      - File Attachments
      - Reference Attachments
      - Item Attachments (including backup of MIME objects)
- Backup and Restore of Onenote
  - Backup/Restore of Notebooks
    - User notebooks
    - Site notebooks

        - Group notebooks
    - Backup/Restore of SectionGroups/Sections
    - Backup/Restore of Pages
        - Support for Page resources: Images and files
- Backup and Restore of Teams
    - Backup/Restore of Team
        - Public and Private Teams
        - Team entity
        - Team members and roles
        - Team installed apps
    - Backup/Restore of Team Channels
        - Public and Private Channels
        - Channel entity
        - Channel members
        - Channel tabs
        - Channel messages
        - Channel messages hosted contents
- Backup and Restore of Chats
    - Backup/Restore of Chat
        - Chat entity
        - Chat installed apps
        - Chat tabs
        - Chat messages
        - Chat messages hosted contents

1.2 Requirements

Microsoft 365 Personal, Family, Microsoft Home & Student subscriptions **are not supported** for backup/restore purposes.

It is necessary to have **full administrative access to the target Tenant** to protect in order to provide the required permissions to the Azure Application linked to this Bacula Enterprise Microsoft 365 Plugin.

Currently the plugin must be installed on a Linux based OS (RH, Debian, Ubuntu, SLES ..) where a Bacula Enterprise File Daemon is installed. *Bacula Systems will address support for running this plugin on a Windows platform in a future version.*

The OS where the File Daemon is installed must have installed Java version 8 or above.

If the Sharepoint module is going to be used, the OS where File Daemon is installed must also have the following packages installed:

- PowerShell v7.2.1 or above
- PnP Powershell v1.9.0 or above

Memory and computation requirements completely depend on the usage of this plugin (parallelization, environment size, etc). However, it is expected to have a minimum of 4GB RAM in the server where the File Daemon is running. By default, every job could end up using up to 512Mb of RAM in demanding scenarios (usually it will be less). However, there can be particular situations where this could be higher. This memory limit can be adjusted internally (see out-of-memory). Refer to the Scope section below for any service specific requirements.

1.3 Why Protect Microsoft 365?

This is a common question that arises frequently among IT and Backup professionals, so it is important to have a clear picture of it. It is true that Microsoft offers some services intended to prevent data loss:

- As with any cloud data, Microsoft 365 data is geo-replicated using Azure cloud to several destinations automatically and transparently. Therefore, complete data loss because of hardware failures are very unlikely to happen.
- Data Loss Protection service: Policy based services capable of detecting filtered content and act upon it encrypting it or modifying it in order to protect it (remove headers, etc). This is not a backup tool, is a service to prevent undesired actions to the content stored in Microsoft 365 (for example sharing confidential information with the wrong people).
- Retention policies of Microsoft 365: Microsoft retains a maximum of 30 days of deleted information from active subscriptions. Therefore it is possible to recover accidental deleted items inside that period. For more information:
  - Subscription Retention
  - High availability and business continuity

There are no other protection mechanisms for data protection. Below is a listing of challenges not covered:

- No Ransomware protection: If data suffers an attack and becomes encrypted, data is lost.
- No malicious attacker protection: If data is deleted permanently, data is lost.
- No real point-in-time recovery for Exchange 365, and recoveries of partially deleted files are limited to 30 days.
- Point in time recovery for OneDrive/Sharepoint limited to 30 days from deletion.
- It is not possible to align data protection of Microsoft 365 services to general retention periods or policies longer than 30 days.
- No automated way to extract any data from the cloud to save it in external places (this could lead to eventual compliance problems)

2 Scope

Bacula Enterprise Microsoft 365 Plugin is applicable on environments using any enterprise plan where target services of this plugin are included:

- Office 365 platform
- Exchange Online
- OneDrive for Business
- Sharepoint Online

The lack of any of the services in your subscription could lead to authentication problems (See troubleshooting section *App Registration Error*).

For more details about Microsoft 365 plans, visit these links:

- https://www.microsoft.com/microsoft-365/enterprise/compare-office-365-plans
- https://docs.microsoft.com/en-us/office365/servicedescriptions/office-365-platform-service-description/office-365-plan-options

This paper presents solutions for **Bacula Enterprise** version 12.8 and later, and is not applicable to prior versions.

Note: Important considerations

Before using this plugin, please carefully read the elements discussed in this section.

2.1 Onedrive Recycle Bin

Onedrive Recycle Bin cannot be protected with this plugin. This is a Microsoft limitation coming from their exposed REST APIs, where it is not possible to access Recycle Bin information. If future versions of Microsoft REST APIs officially include this function, Bacula Systems will include it as a new feature of this plugin.

2.2 Empty files

In general, empty files (files with 0 byte contents) are simply not backed up by Microsoft 365 plugin. In particular, email attachments or onedrive files will show a message in the joblog to inform about empty files detected and so not processed.

2.3 Backup of Attachments and Files

In general, this plugin backups two types of information:

- Objects
- Files

Objects are elements representing some entity in Microsoft 365 such as a calendar event, a contact, an email, a team, etc.

Files are attachments, hosted contents or OneDrive files.

While objects are directly streamed from memory to the backup engine, files need to be downloaded to the FD host before being sent. This is done in order to make some metadata checks and to improve overall performance, as this way operations can be parallelized. Every file is removed just after being completely downloaded and sent to the backup engine.

The path used for this purpose is established by the ‘path’ plugin variable, that usually is set up in the m365_backend script with the value: /opt/bacula/working

Inside the path variable, a ‘spool’ directory will be created and used for those temporary download processes.

Therefore, it is necessary to have at least enough disk space available for the size of the largest file in the backup session. If you are using concurrency between jobs or through the same job (by default this is the case through the concurrent_threads=5 parameter), you would need at least that size for the largest file multiplied by the number of operations in parallel you run.

2.4 Accurate Mode and Virtual Full Backups

Accurate mode and Virtual Full backups are not supported. These features will be addressed in future versions of this plugin.

2.5 Details on the Sharepoint backup/restore

The M365 plugin overcomes the current limitations of the MS Graph API on backup/restore of SharePoint sites by using the Powershell PnP opensource project:

- https://learn.microsoft.com/en-us/powershell/sharepoint/sharepoint-pnp/sharepoint-pnp-cmdlets

This project has been designed to abstract many complexities of the Sharepoint Online APIs and provides all sorts of automation processes. Among them, site template provisioning, which is the main part of the method **Bacula Enterprise** Microsoft 365 Plugin uses in order to provide backup/restore capabilities for Sharepoint Online.

2.6 Site Template Provisioning

Microsoft can modify at any time (and they do it frequently) the properties, structure or limitations of any element belonging to a site kind (team site, communication site, project site, etc) whicn can cause the restore provision process to fail. It is even common to see native structures of Sharepoint Online trigger errors with particular items/values and this error comes from the Sharepoint Online API.

As a result, is not possible to guarantee that a given template is going to work in a provision process. Many times, the template might need to be partially modified before being applied (removing some particular item, removing some particular column..). That part of the process is not possible to predict or automate, so we strongly recommend frequent testing of site restore processes to detect any need of manual restore processes before a real restore is required.

2.7 Particular List/ListItems Restores

It is possible to select and restore a particular list or listItems, without restoring an entire Sharepoint Site, and pointing the operation to an existing Site. This is done using the Graph API, but as we have previously exposed, this API has many limitations with the Sharepoint API. One of the limitations is that lists or list items with advanced elements (location, images, etc) are not supported. Therefore this feature should only be used for user created lists implying simple elements like texts, checks, dates, etc.

2.8 Site Backups In Parallel

Backing up sites in parallel from the same FD is not supported.

2.9 Email Limitations

The following items associated to Outlook elements (Exchange Online service) are not supported due to the fact they are currently not accessible using the APIs this plugin relies on (MS Graph API and PnP Powershell):

- **Sharing permissions** associated to **MailFolders**
- **Public Folders**

2.10 Onenote Limitations

The following items associated to Onenote elements are not supported due to the fact they are currently not accessible using the APIs this plugin relies on (MS Graph API and PnP Powershell):

- **Sharing permissions** associated to **Notebooks**
- **Shared elements** from one entity to another one. To protect them it is necessary to make it always from the source entity.

Notebook 5000 files limitation

Onenote notebooks can only store up to 5000 different items into their associated OneDrive Libraries, this a M365 limitation.

When they surpass that limit they become unaccessible, not only for backup but also for accessing them through any other Microsoft native interface.

When a job finds this kind of notebooks, an error message will be generated and the notebook won't be included in the backup.

You can get more information about this limit and how to solve it in the following Microsoft documentation link:

- https://learn.microsoft.com/en-us/microsoftteams/troubleshoot/teams-onenote-integration/issue-access-notebook

2.11 External Personal Calendars

External calendars (as Google Calendar) connected to Microsoft accounts are not supported. They are external elements not exposed through MS APIs, so M365 Plugin cannot protect that information.

2.12 Calendar Events icons

Little icons that appear on the left of a calendar event title (usually auto-generated by the Outlook service) cannot be backed up, as they are not exposed through the MS Graph API the plugin is relying on.

2.13 Chats types

For the chats module, only types 'group' or 'one_on_one' can be backed up. Other types like: 'meeting' or 'unknownFutureValue' will be ignored.

2.14 MS Cloud APIs General Disclaimer

MS Cloud APIs are Microsoft property and they can change or evolve at any time. In particular, the Graph API is actively developed, containing new features every week, even if the version number of the service (1.0) is not changed as a result of any of those additions:

- https://developer.microsoft.com/en-us/graph/changelog

This situation is significantly different from traditional on-premise software, where each update is clearly numbered and controlled for a given server, so applications consuming that software, can clearly state what is offered and what are the target supported versions.

Microsoft is committed to trying to not break any existing functionality that could affect external applications. However, this situation can happen and therefore, cause some ocasional problems with this plugin. Bacula Systems controls this with an advanced automatic monitoring system which is always checking the correct behavior of existing features, and will react quickly to that hypothetical event, but please be aware of the nature and implications of this kind of cloud technologies.

3 Architecture

Bacula Enterprise Microsoft 365 Plugin is using the **Microsoft Graph API** to perform almost all of its operations. Therefore, the plugin is working at the maximum granularity that the service provides.

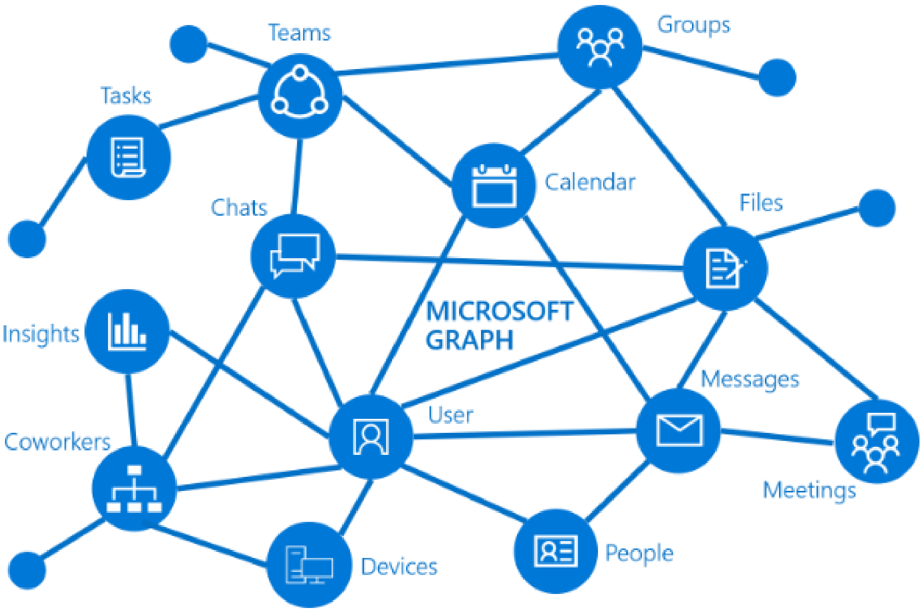


Fig. 1: Microsoft Graph

All the information is obtained using secure and encrypted HTTPS queries to Microsoft 365 from the File Daemon where the plugin is installed. All the requests are performed over the following endpoints:

- Graph API to manage all the services (except Sharepoint Online): https://graph.microsoft.com
- Login endpoint: https://login.microsoftonline.com/
- Sharepoint Online service endpoint: https://{tenantname}.sharepoint.com/

To get more information about Graph API, visit: https://learn.microsoft.com/en-us/graph/overview

The plugin will contact an Azure registered app named **bacula-m365-plugin** and will use it as a bridge to download the required data or objects during the time of a backup and send them to the Storage Daemon. Conversely, the plugin will receive them from an SD and perform uploads as needed during a restore.

The implementation is done through a Java Daemon, therefore Java is a requirement in the FD host. For more information about the bacula-m365-plugin, please, consult *Authorization* section.

Below is a simplified vision of the architecture of this plugin inside a generic **Bacula Enterprise** deployment:

Listed below is the information that can be protected using this plugin:

- Email

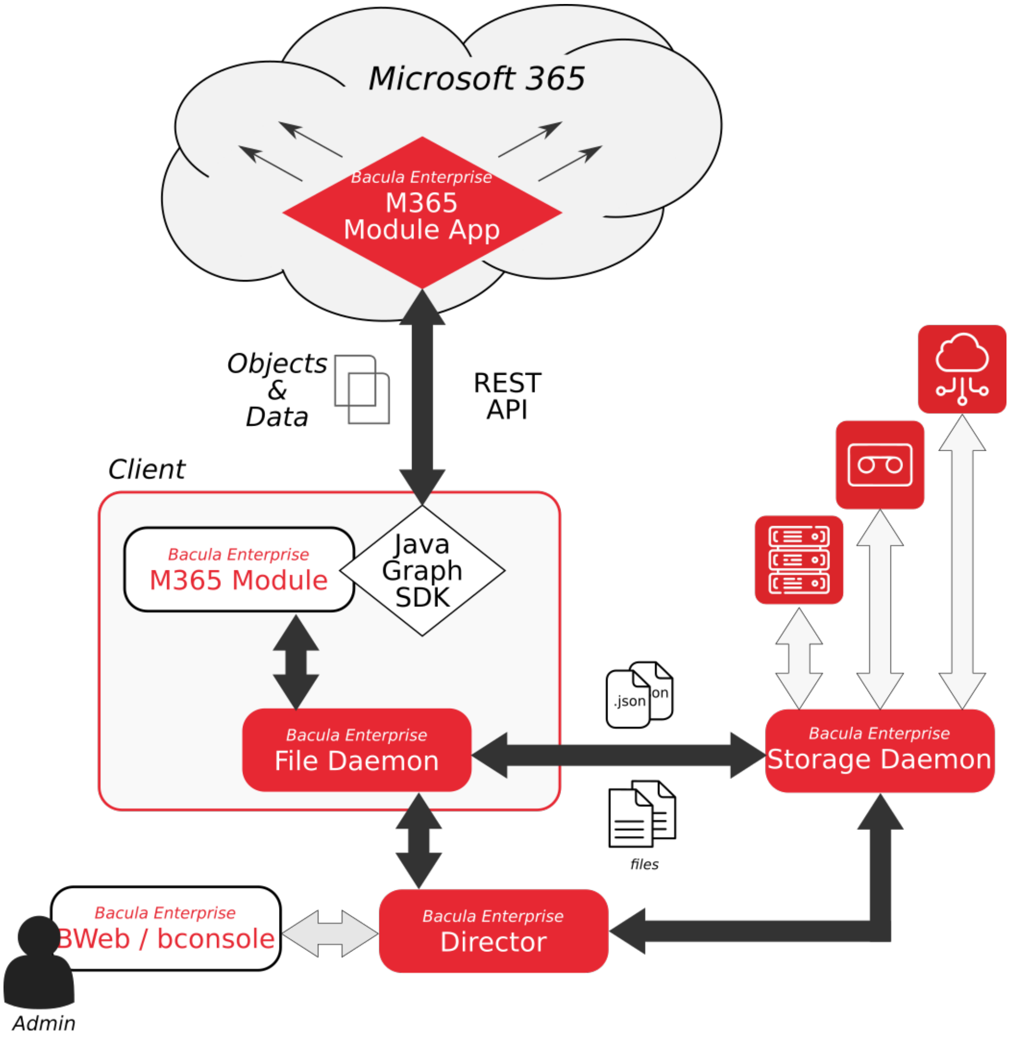


Fig. 2: M365 Plugin Architecture

  - Common MailFolders (Inbox, Deleted Items, Drafts..)
  - User MailFolders & SubFolders
  - Messages & EventMessages
  - Attachments (ItemAttachments, FileAttachments and ReferenceAttachments)
  - Mailbox settings
  - Folder Rules
- OneDrive
  - Onedrive for Business of Users, for each drive unit
    - Folders
    - Files
    - File Versions
  - Group libraries, for each drive unit
    - Folders
    - Files
    - File Versions
  - Sharepoint site libraries, for each drive unit
    - Folders
    - Files
    - File Versions
  - Shared permissions (direct access, share links, expiration times..)
  - SharedWithMe Objects
- Sharepoint
  - Pnp Site template
    - Site metadata
    - Lists metadata
    - ListItems metadata
    - WebPages metadata
  - Site Object
    - Site sharing permissions
  - Lists Objects
  - ListItem Objects
- Contacts/People
  - Contact object
  - Name of Groups of contacts
  - Organizational contact object
- Tasks

    - User Todo lists
    - User Todo tasks
- Calendars
    - Calendar objects
    - Calendar group objects
        - Calendar permissions
    - Events objects
    - Attachments (ItemAttachments, FileAttachments and ReferenceAttachments)
        - MIME objects where possible
- Notebooks
    - Notebook objects
    - Section objects
    - SectionGroup objects
    - Pages
        - Page contents (Html formatted) - Page resources
            - Page image files
            - Page object files (any other file apart from images)
- Teams
    - Team objects
    - Team settings
    - Team members and associated roles
    - Team installed apps
    - Channel objects
        - Channel tabs
        - Channel chat messages
            - Chat messsages hosted contents
- Chat
    - Chat objects
    - Chat installed apps
    - Chat tabs
    - Channel chat messages
        - Chat messsages hosted contents

All the information of each object is stored in JSON format (except for Pnp site template, which is stored in XML), preserving all their original values. When the plugin works with objects containing additional data (MIME files for messages, data for attachments and files of OneDrive, etc), that data is also backed up.

4 Services and Features

In this section we will dig into how this plugin behaves for each particular service, describing special features and and behaviors that require an extended description.

4.1 OneDrive

Bacula Enterprise Microsoft 365 Plugin can protect OneDrive Business units associated to users, groups or sites. It is possible to utilize advanced selection methods to decide exactly what is backed up, as well as control precisely which items to restore and their destinations. The information protected with this service is:

- Files
- File versions
- Items M365 metadata (files and folders)
- Shared permissions

Files will keep their names in the catalog and will be included in a path like this:

- `/@m365/tenant.name/entitykind/ownerentityname/drives/unitname/root:/path/to/file/name-file.extension`

  (where `entitykind` can be users, sites or groups)

Version History

OneDrive and SharePoint can be configured to *retain the history for files/items*.

Onedrive hash check

Onedrive service stores a hash for every file hosted, using Microsoft algorithm QuickXOR. Bacula Enterprise Microsoft 365 Plugin calculates this hash and compares it to the Microsoft hashes at backup time, and also at restore time in order to ensure data integrity. Debug mode shows information about these hashes.

Onedrive shares

Bacula Enterprise Microsoft 365 Plugin is able to backup and restore shared elements. These kind of elements require a special treatment, as they are composed of two parts:

- In the source account, shared elements include special information about the permissions of the share (who and how the share must work)
- In the destination account, shared elements appear within an special category called ‘SharedWithMe’.
  - SharedWithMe elements are not directly accessible from the destination account, as they are links to the source account. Therefore, Bacula Enterprise Microsoft 365 Plugin will query the source account in order to get these elements when they are part of the backup target. The source of those files must be located inside the same tenant the plugin is protecting, otherwise it will not be able to access them.

Shared permissions

Bacula Enterprise Microsoft 365 Plugin will query for the permissions of an item if this item has been shared directly. This means the plugin will not backup inherited permissions. In order to have inherited permissions in a backup, the top element where the original shared permissions were set needs to be included in the backup and in the restore. As an example, if a directory is shared but we restore only specific files contained in it, those files will not be shared as they were in originally. It is necessary to restore the whole directory in order to replicate the original inherited permissions as they were at the time of backup.

The method to store shared permissions is to include them as 'extended_attributes' of every file. This implies that permissions can only be restored directly to the Microsoft 365 service. A File Daemon restore to a local filesystem will only restore files, and shared permissions will not be restored.

Shared permissions can include links. Links are pre-generated URLs that can include, password, expiration dates and other configuration parameters as scopes, types, or affected identities. Shared permissions restores have special characteristics that must be considered and they are described below:

- Permission will be generated exactly as it was, but it will be a new permission object. This is similar to the situation with files. A restored file has the same contents as the original, but can include slightly different metadata because creation process was different.
  - If permission had a static link, a link will be generated, but the associated URL will be different from the original.
  - If a link had a password, the password will not be restored. Every link will be restored without a password (restore job log will warn about all links that had a password)
    - This is a limitation of OneDrive Business, as it does not allow passwords to be generated from APIs.
- Permissions can send a notification message to destination users.
  - This is configured at restore time with a parameter. Consequently, if different shares require different treatment, different restore sessions will be needed.
- Possible unexpected problems with permissions at restore time are treated as Warnings, not as Errors - as long as the file was restored successfully.

Shared permissions are not restored by default. You need to enable the option 'drive_restore_shared_permissions' during the restore session.

Shared with me

SharedWtihMe elements of each target account, if included in the FileSet, are backed up in a predefined directory called SharedWithMe inside the top folder of every selected account. For example, for a given account youraccout@yourdomain.com in a tenant called yourtenant:

Listing 1: **SharedWithMe**

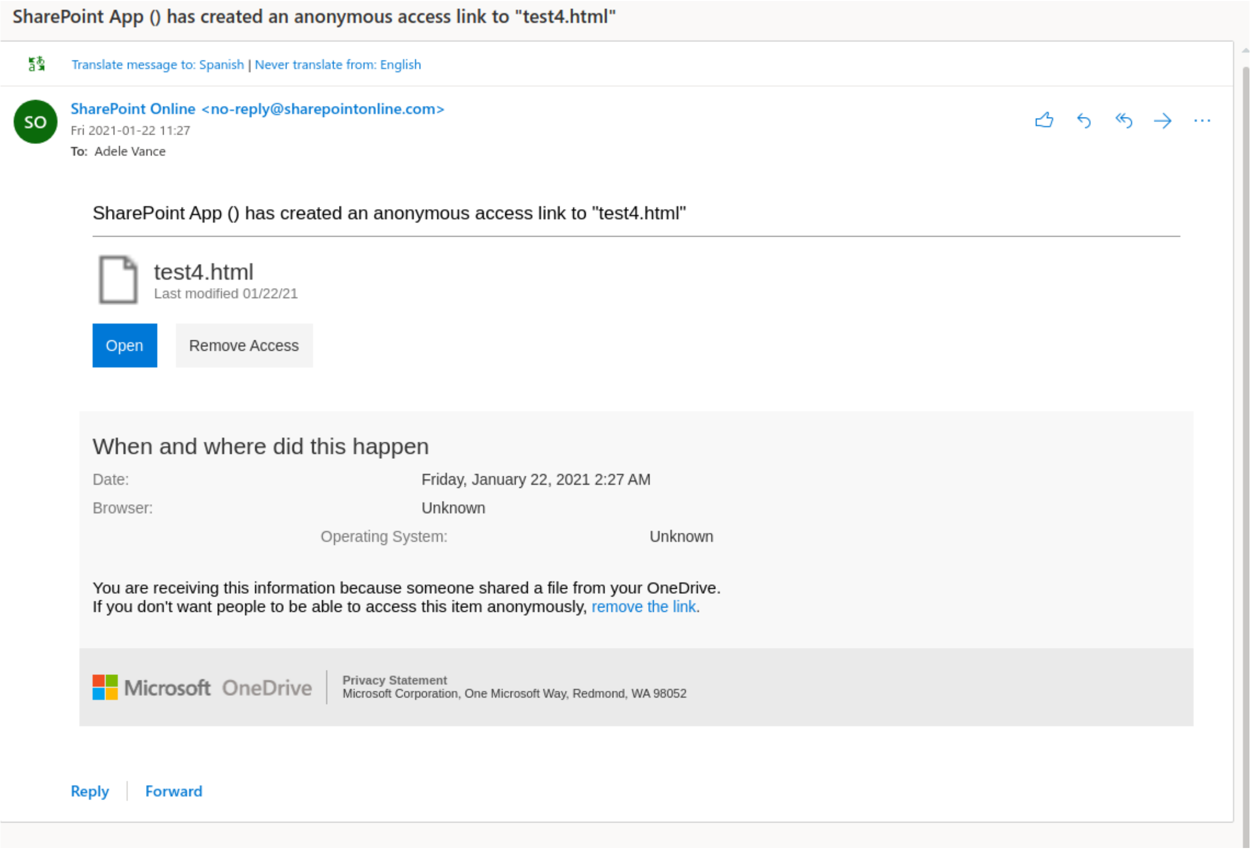
```
``/@m365/yourtenant/users/youraccout@yourdomain.com/drive/onedrive/root:/sharedWithMe/``
```

At restore time, sharedWithMe elements are treated as any other regular file. However, it is important to note that sharedWithMe files, as we are in the receiver account, have no share permissions.

SharedWithMe elements included for backup are only the ones belonging to the current tenant. Backup of sharedWithMe elements from external tenants are not supported.

The plugin has an special parameter at restore time allowing it to skip sharedWithMe elements even if they are selected. This feature is intended to facilitate full restores where source and destination accounts are included. Please, note that a restore of a source account with share elements will present those elements to any receiver account if you enable the option to restore share permissions, as we have discussed in 3.2.1.

Please, be aware that restoring permissions from the plugin will generate automatic M365 reporting emails in the user Inbox as the following example shows:



SharePoint App () has created an anonymous access link to "test4.html"

Translate message to: Spanish | Never translate from: English

SharePoint Online <no-reply@sharepointonline.com>
Fri 2021-01-22 11:27
To: Adele Vance

SharePoint App () has created an anonymous access link to "test4.html"

test4.html
Last modified 01/22/21

When and where did this happen

Date: Friday, January 22, 2021 2:27 AM
Browser: Unknown
Operating System: Unknown

You are receiving this information because someone shared a file from your OneDrive.
If you don't want people to be able to access this item anonymously, remove the link.

Microsoft OneDrive | Privacy Statement
Microsoft Corporation, One Microsoft Way, Redmond, WA 98052

Fig. 3: Restored permissions warning message

Please, go to the Configuration section of this document to see how to set up the sharedWithMe skip option.

Note: If you enable the backup for file versions (*Calendars*) and sharedwithme elements, sharedwithme elements will include all their versions in the backup.

Backup parameters

The list below shows the specific backup parameters that can be set up in order to control the behavior of the drive module.

In order to select the drive module, the common *service* parameter must be equals or be containing the value *drive*.

Entities that can include one drive units are: users, groups or sites.

Option	Required	Default	Values	Example	Description
drives	No		Valid names or ids of existing drives on the selected tenant belonging to the selected entity (user, group or site) separated by ,	documents, images, b!12344490zcdfs09ewrqwf	Will backup only selected drive units belonging to the specified entity (user, group or site)
drives_exclude	No		Valid names or ids of existing drives on the selected tenant belonging to the selected entity (user, group or site) separated by ,	onedrive	Will backup all drives except the excluded in this lists, belonging to the specified entity (user, group or site)
drives_regex_include	No		Valid regex	*.pages	Backup matching drive units (based in the drive unit name)
drives_regex_exclude	No		Valid regex	^site.*	Exclude matching drive units (based in the drive unit name)
drive_files	No		Strings representing existing **folders** for the given users separated by ,	Customers, Partners	Backup only specified **folders** belonging to the selected users
drive_files_exclude	No		Strings representing existing **folders** for the given users separated by ,	Personal	Exclude selected **folders** belonging to the selected users
drive_files_regex_include	No		Valid regex	.*Company	Backup matching drive folders. Please, only provide list parameters (files + files_exclude) or regex ones. But do not try to combine them.
drive_files_regex_exclude	No		Valid regex	.*Plan	Exclude matching drive folders from the selection. Please, only provide list parameters (files + files_exclude) or regex ones. But do not try to combine them. If this is the only parameter found for selection, all elements will be included and this list will be excluded.
drive_system_include	No	No	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	Yes	Include system documentLibraries of sharepoint (sitepages, siteassets, etc).
drive_shared_with_me	No	Yes	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	No	Include SharedWithMe elements of every target entity in the backup process
drive_version_history	No	No	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	Yes	Include Onedrive former versions of every file into the backup process
drive_permissions	No	Yes	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	No	Include Onedrive files sharing permissions as part of the backup. If you need higher performance, it is recommended to disable this option
drive_disable_hashcheck	No	No	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	No	Disable the hash checking mechanism for each file downloaded. This is only recommended for document libraries where the hash stored at M365 level is detected to be incorrect for an important number of files (specially in sharepoint jobs)

- Note: In previous versions, instead of drive_files* parameters, it was possible to use files_* parameters. They are deprecated parameters now as we recommend to use module specific parameters in order to have better control and more possibilities inside a single fileset. However, jobs using files_* parameters will still work.

Restore

The list below shows the subset of restore parameters that can be used to control the behavior of onedrive module restore operations:

- destination_user, destination_group, destination_site, destination_drive, destination_path, send_report, allow_duplicates
- drive_skip_sharedwitme, skip_versions, restore_share_permissions, drive_send_invitations, drive_invitations_message
- debug, foreign_container_generation

Use cases

The following restore scenarios are supported:

- Restore files, directories, or file versions to original entity drive or to a different entity drive
  - Restore parameters implied: **destination_user, destination_group, destination_site**
- Restore file(s)/dir(s) or file version(s) to original path or to a different path
  - Restore parameters implied: **destination_path**
- Restore file(s)/dir(s) or file version(s) to original drive unit or to a different drive unit
  - Restore parameters implied: **destination_drive**
- Restore file(s)/dir(s) or file version(s) to local file system (general restore **where** parameter must be set to a path)
- It is possible to make general restore selections, but avoid restoring versions
  - Restore parameters implied: **skip_versions**
- It is possible to restore sharing permissions of implied files
  - Restore parameters implied: **restore_shared_permissions, drive_send_invitations, drive_invitations_message**
- It is possible to make general restore selections, but specify if backed up shared elements must be considered
  - Restore parameters implied: **drive_skip_sharedwitme**
- It is possible to control whether or not duplicate elements are allowed (based on file id):
  - Restore parameters implied: **allow_duplicates**

Particularities:

- If no **destination entity** is provided, the destination entity will be looked for inside the backed up path, so the destination user will be the same as the original one
- If no **destination_path** is provided, the destination path will be the same as the original one
  - If a destination entity was provided, but no destination_path was provided and the selected file did not belong to the destination entity:
    - A new folder will automatically be created inside the target entity

    - For each ‘foreign’ entity, a new folder will be created
    - Inside each ‘foreign’ entity folder, the original path structure will be preserved when restoring the files
        - *Unless the parameter **foreign_container_generation** is disabled

For more details about the behavior of each parameter, please check the general section of restore parameters.

Fileset examples

Full OneDrive of one user:

Listing 2: **Fileset Example**

```
FileSet {
   Name = fs-m365-drive-adelev
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=drive tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-eb5d-
   42nf-bac7-7b019fd284g5 user=adelev@baculaenterprise.onmicrosoft.com"
   }
}
```

Folders of one user, but include sharedWithMe elements:

Listing 3: **Fileset Example**

```
FileSet {
   Name = fs-m365-drive-adelev-shared
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=drive tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-eb5d-
   42nf-bac7-7b019fd284g5 user=adelev@baculaenterprise.onmicrosoft.com drive_files=\
↪"dir1,dir2\"
   drive_shared_with_me=yes"
   }
}
```

Full OneDrive of one group, include version history:

Listing 4: **Fileset Example**

```
FileSet {
   Name = fs-m365-drive-devteam-versions
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=drive tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-eb5d-
   42nf-bac7-7b019fd284g5 group=\"Dev Team\" drive_version_history=true"
   }
}
```

Full OneDrive of one sharepoint site:

Listing 5: **Fileset Example**

```
FileSet {
   Name = fs-m365-drive-live
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=drive tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-eb5d-
   42nf-bac7-7b019fd284g5 site=\"Live @ MyEnterprise\""
   }
}
```

Specific drives of a sharepoint site, including some system:

Listing 6: **Fileset Example**

```
FileSet {
   Name = fs-m365-drive-live-2-drives
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=drive tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-eb5d-
   42nf-bac7-7b019fd284g5 site=\"Live @ MyEnterprise\" drive_system_include=yes drives=b!
↪7Tf3z7izSES7kjOHD-YM-
   0kScgNXL6lFnL4O2LWLMy4yH2tqunhTSpSwmPR5a0hq,b!7Tf3z7izSES7kjOHD-YM-
↪0kScgNXL6lFnL4O2LWLMy41XlWc2E0pTKfPzlKxhztE"
   }
}
```

Exclude directories of two users:

Listing 7: **Fileset Example**

```
FileSet {
   Name = fs-m365-drive-adjon-users-notemp
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=drive tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-eb5d-
   42nf-bac7-7b019fd284g5 user=\"adelev@baculaenterprise.onmicrosoft.com,
↪jonis@baculaenterprise.onmicrosoft.com\"
   drive_files_exclude=\"*.temporary\""
   }
}
```

4.2 Email / Mailboxes

Bacula Enterprise Microsoft 365 Plugin can protect M365 Mailboxes associated to users. It is possible to utilize advanced selection methods to decide exactly what is backed up (folders included/excluded, users included/excluded), as well as control precisely which messages or attachments to restore and where (original user's account or another user's account). The information protected with this service is:

- Folders (Inbox, Deleted Items, Sent…)
  - Messages (Metadata and contents)
  - Attachments
    - File Attachments
    - Item Attachments
    - Reference Attachments
- Folder Rules
- Mailbox Settings
- Outlook Categories

Public folders are not supported and **folder sharing permissions are not protected** (see *Email Limitations*). On the other hand, folders are backed up always from their original accounts:

- This means if a folder 'folderA' is shared from user 'U1' to user 'U2', if you only backup the entire mailbox of U2, contents of folderA will not be protected. In order to protect folderA you will need to add some additional job protecting U1 and folderA into the backup set or simply add the entire mailbox of U1.

User mailboxes and **shared mailboxes** are supported, it is only required to setup the correct address in user* parameters (shared mailboxes are treated as users). For **groups**, please note that they **have no real mailbox**, therefore it is not possible to backup the emails belonging to a given group. However, protecting emails of any of the users that is member of those groups will result in protecting all the received emails of that group.

Mailbox backup includes the following features:

- Incremental backup with Delta function:
  - Delta function is applied for each folder individually
- MIME object backup:
  - Based on the fileset parameter **email_mime** it is possible to get mime messages as well as the M365 objects. These kind of objects can be useful to have if there is any plan of using the emails outside the M365 service.
  - At restore time, if the restore operation is done via M365 services and not to a local FileSystem, mime_objects are automatically ignored.
- Multilevel-attachments:
  - M365 supports having messages or event objects as the attachment of a given message object. These attachments can have additional attachments. This situation is what M365 Plugin considers multilevel-attachments. Some particular objects such as Files, when present in this multilevel way, are got embedded in the metadata of the message attached. This implies that the plugin must get them in a single HTTP call and load them entirely in memory. If there is not enough memory in the system, the variable **multilevel_attach** can be disabled and those objects will be ignored.
  - Multilevel-attachments are backed up at the same level. This means that a message with this structure:
    - Base Message has 1 Message-Attachment, Message-Attachment has 1 more internal Message-Attachment 2, and Message-Attachment 2 has a File attachment

        - When restored, the result will be all Attachments being restored to the same level, not one inside the another as they were originally. For example:
        - Base Message
            - Message-Attachment 1
            - Message-Attachment 2
            - File Attachment

Messages will be formatted in the catalog in order to not include sensitive information and will be included in a path like this:

- `/@m365/tenant.name/users/user@tenant.com/email/foldername/2021-04-19 09.58.52..J-123.10.msg`
    - Where the message name is composed as: messageDate..J-JobId.JobIndex.msg
        - messageDate corresponds with the receivedDate
        - JobIndex is an internal index relative to a backlup job execution
        - The extension .msg corresponds to a file containing an email
        - Mime messages will have the extra word ‘mime’ in their extension. For example:
            - `/@m365/tenant.name/users/user@tenant.com/email/foldername/2021-04-19 15.54.06..J-2021-04-19_15.54.45_03.10.mime.msg`
- `/@m365/tenant.name/users/user@tenant.com/email/mailbox\_settings.mail.set`
    - Mailbox Settings
- `/@m365/tenant.name/users/user@tenant.com/email/rules/nameRule.mail.rule`
    - Mailbox Rules
- `/@m365/tenantname/users/usermail/outlook\_category_name.olk.cat/`
- Mailbox Categories

Attachments will be stored together with message objects:

- They include their original name (file name)
- They have an extension about their type (.file.att, .ref.att or .item.att)
- The first part of the attachment name is the name of the parent message

Here are a few attachment examples:

```
/@m365/tenant.name/users/user@tenant.com/email/foldername/2021-04-19
15.54.06..J-123.10.Aliquid.gen.1.file.att
/@m365/tenant.name/users/user@tenant.com/email/foldername/2021-04-19
15.54.06..J-123.10.Dolor.gen.4.ref.att
/@m365/tenant.name/users/user@tenant.com/email/foldername/2021-04-19
15.54.06..J-123.10.Sapien Nostrum Aperiri Unum - t.3.item.att
/@m365/tenant.name/users/user@tenant.com/email/foldername/2021-04-19
15.54.06..J-123.10.Veri.gen.2.file.att
/@m365/tenant.name/users/user@tenant.com/email/foldername/2021-04-19
15.54.06..J-123.10.msg → Parent message
```

Backup parameters

The list below shows the specific backup parameters that can be set up in order to control the behavior of the email module.

In order to select the email module, the common *service* parameter must be equals or be containing the value *email*.

Entities that can include mailboxes are: users.

Option	Required	Default	Values	Example	Description
email_files	No		Strings representing existing **mailfolders** for the given users or groups separated by ‘,’	Inbox, Sent	Backup only specified **mailfolders** belonging to the selected users
email_files_exclude	No		Strings representing existing **mailfoders** for the given users or groups separated by ‘,’	Archive, Personal	Exclude selected **mailfolders** belonging to the selected users
email_files_regex_include	No		Valid regex	.*Company	Backup matching mailfolders. Please, only provide list parameters (files + files_exclude) or regex ones. But do not try to combine them.
email_files_regex_exclude	No		Valid regex	.*Plan	Exclude matching mailfolders from the selection. Please, only provide list parameters (files + files_exclude) or regex ones. But do not try to combine them. If this is the only parameter found for selection, all elements will be included and this list will be excluded.
email_multilevel_attach	No	Yes	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	No	Include multilevel attachments in backup. A multilevel attachment is, for example, a message attached to a message containing more elements attached inside. Under certain circumstances, multilevel attachments could imply higher memory needs, as some of them are embedded in the metadata of the objects to backup, so they need to be caught in a single call and be loaded entirely in memory. This parameter is intended to control the behavior against that situation.
email_exclude_attachments	No	No	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	Yes	Exclude any attachment from backup
email_mime	No	No	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	Yes	Backup raw MIME file of every email, in addition to the Message object itself
email_settings	No	No	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	Yes	Backup mailbox settings of included users
email_rules	No	No	0, no, No, false, FALSE, false, off ;	Yes	Backup mailbox folder rules of included users

Note: In previous versions, instead of email_files* parameters, it was possible to use files_* parameters. They are deprecated parameters now as we recommend to use module specific parameters in order to have better control and more possibilities inside a single fileset. However, jobs using files_* parameters will still work. The following fields: email_messages_exclude_expr, email_messages_exclude_index_expr, email_fields_exclude, email_fields_exclude_index, email_filter_received_from are available from Bacula Enterprise version 14.0.

Restore

The list below shows the subset of restore parameters that can be used to control the behavior of email module restore operations:

- destination_user, destination_path, send_report, allow_duplicates, debug, foreign_container_generation

Use cases

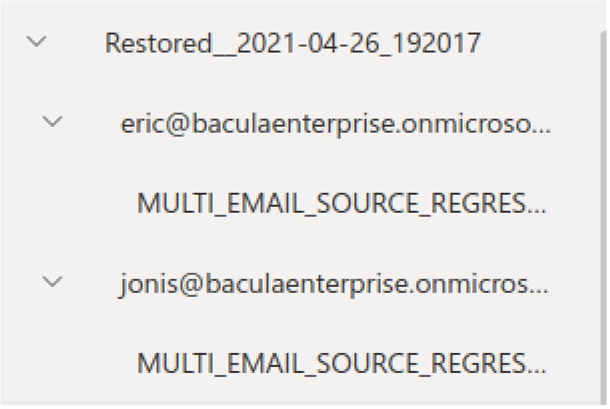
The following restore scenarios are supported:

- Restore directories, emails, or attachments to original user or to a different user
  - Restore parameters implied: **destination_user**
- Restore directories, emails, or attachments to original path or to a different path
  - Restore parameters implied: **destination_path**
- Restore files, directories, or file versions to local file system (general restore **where** parameter must be set to a path)
- It is possible to control whether or not duplicate elements are allowed (based on file id):
  - Restore parameters implied: **allow_duplicates**

Particularities:

- If no **destination_user** is set, every message will be restored into its original mailbox
- If no **destination_path** is set, every message will be restored into its original path
  - If the selection contains messages from several users:
    - Original user messages will be restored in their original location
    - For other users, a special folder will be created with the email address of each of them, containing the full path and messages of the restored objects, unless the parameter **foreign_container_generation** is disabled
    - Example:
- Restored elements will be duplicated by default, unless **allow_duplicates** variable is disabled
  - Even when disabling that variable, messages will be checked by id. So if there is an element with the same information but different ID, it will not be considered to be a duplicate

For more details about the behavior of each parameter, please check the general section of restore parameters.



Restored__2021-04-26_192017

eric@baculaenterprise.onmicroso...

MULTI_EMAIL_SOURCE_REGRES...

jonis@baculaenterprise.onmicros...

MULTI_EMAIL_SOURCE_REGRES...

Fig. 4: Restore of emails from 2 different users over a third mailbox without destination_path result in auto-generated Restore_date folder containing those 2 foreign users with the restored folder inside of them

Messages exclude expressions

Bacula Systems is aware about one of many privacy concerns that may arise when tools like this M365 Plugin enables the possibility to backup and restore data coming from different users, so the backup administrator can restore potentially private data at his will. Moreover, emails are usually one of the most critical items in terms of privacy.

One of many strategies this plugin offers in order to deal with that problem is the possibility to exclude messages. This is a very powerfull feature where it is possible to use quite flexible expressions that allow to select a subset of messages and simply exclude them from the backup:

- email_messages_exclude_expr new fileset parameter

Or only from the index (from the catalog)

- email_messages_exclude_index_expr new fileset parameter

Not only messages can be excluded but also select only a subset of email fields to be included in the protected information. It is possible to exclude fields from the backup:

- email_fields_exclude new fileset parameter

Or only from the index (from the catalog):

- email_fields_exclude_index new fileset parameter

Please, be aware that if the fields are excluded from backup, the restore operation to M365 can fail easily, since for instance emails cannot exist in M365 without From field, Subject field, etc.

All four discussed expressions are based on an internal structure of fields to work with. Below you can see the entire list of fields that you can use:

- emailTags
- emailSubject
- emailFolderName
- emailFrom
- emailTo
- emailCc
- emailBodyPreview

- emailImportance
- emailTime
- emailIsRead
- emailIsDraft

Please note that it is very important to write the fields exactly as written above.

These fields can be used in a comma separated list in the 'email_fields_exclude' parameter and also 'email_fields_exclude_index' parameter.

Then, for 'email_messages_exclude_expr' and 'email_messages_exclude_index_expr' use them in a valid boolean expression in **Javascript** language syntax. Some examples are provided below:

Listing 8: **Expression to exclude messages where subject includes the word 'private'**

```
emailSubject.includes('private')
```

Listing 9: **Complex expression to exclude messages that are not read and are Draft or their folder name is named Private**

```
!emailIsRead && (emailIsDraft || emailFolderName == 'Private')
```

Listing 10: **Expression to exclude messages based on the received or sent date**

```
!emailTime < Date.parse('2012-11-01')
```

Listing 11: **Expression to exclude messages using a regex based on emailFrom**

```
/.*private.com/.test(emailFrom)
```

Note: This feature is available since Bacula Enterprise version 14.0

Expression tester

This expression mechanism can sometimes be uncertain for end users, where they can have doubts about the correct behavior of their prepared expressions. In order to help with that, M365 Plugin presents a query method that allows to test those expressions against a static pre-loaded set of data.

There are two commands available:

- Show command: It will show the static data in json format, so it is possible to see the contents to adapt the expressions to test
- Test command: It will apply the expression parameters to the pre-loaded static data

The test command has the following format:

Listing 12: **Expression tester Show command**

```
.query client=<your-fd-client> plugin="m365: tenant=<your-tenant-id>" parameter=email-
↪expr-show
```

The show command has the following fomat

Listing 13: **Expression tester Test command**

```
.query client=<your-fd-client> plugin="m365: tenant=<your-tenant-id> email_messages_
↪exclude_expr = \"<your-js-expression>\"" parameter=json|email-expr-test
// Or
.query client=<your-fd-client> plugin="m365: tenant=<your-tenant-id> email_messages_
↪exclude_index_expr = \"<your-js-expression>\"" parameter=json|email-expr-test
```

Please, not that you need to provide a valid tenantId, even if it's not really used to process any data.

The test command produces some JSON output with objects with the exact format that is received from Microsoft and, consequently the same format that is stored in backup. Please not the 'total' value at the end, where the value of 12 total pre-loaded messages is shown

Listing 14: **Expression tester Show command output**

```
.query client=<your-fd-client> plugin="m365: tenant=<your-tenant-id>"␣
↪parameter=json|email-expr-show
....
    "email-12": {
      "body": {
        "content": "These are the contents in text format of the 12 email of test data.␣
↪It has the following categories:orange, black, white, purpleYou can try to filter this␣
↪body using any JS method like /.*12.*/.test(emailBody) or emailBody.includes(12)",
        "contentType": "TEXT"
      },
      "ccRecipients": [
        {
          "emailAddress": {
            "address": "danny@other.com"
          }
        },
        {
          "emailAddress": {
            "address": "lucas@other.com"
          }
        },
        {
          "emailAddress": {
            "address": "terese@other.com"
          }
        }
      ],
      "from": {
        "emailAddress": {
          "address": "elon@other.com"
        }
```

(continues on next page)

(continued from previous page)

```
      },
      "hasAttachments": false,
      "isDraft": false,
      "isRead": false,
      "replyTo": [
        {
          "emailAddress": {
            "address": "elon@other.com"
          }
        }
      ],
      "sentDateTime": {
        "dateTime": {
          "date": {
            "year": 2021,
            "month": 12,
            "day": 5
          },
          "time": {
            "hour": 11,
            "minute": 30,
            "second": 0,
            "nano": 0
          }
        },
        "offset": {
          "totalSeconds": 0
        }
      },
      "subject": "This is private subject 12",
      "toRecipients": [
        {
          "emailAddress": {
            "address": "laura@other.com"
          }
        },
        {
          "emailAddress": {
            "address": "jack@other.com"
          }
        },
        {
          "emailAddress": {
            "address": "john@other.com"
          }
        }
      ],
      "categories": [
        "orange",
        "black",
        "white",
        "purple"
```

(continues on next page)

(continued from previous page)

```
      ]
    }
  },
  {
    "total": "12"
  }
```

The test command, on its side will produce two different outputs. The first part presents the same format than the show format, and those are the messages that would be included in the backup. The second part presents a different format, so an output like:

Listing 15: **Expression tester Test command, index part output**

```
.query client=<your-fd-client> plugin="m365: tenant=<your-tenant-id>"␣
↪parameter=json|email-expr-show
....
      {
    "meta-email-12": {
      "EmailId": "",
      "EmailOwner": "test@test.com",
      "EmailTenant": "johndoe.onmicrosoft.com",
      "EmailTags": "orange,black,white,purple",
      "EmailSubject": "This is private subject 12",
      "EmailFolderName": "/",
      "EmailFrom": "elon@other.com",
      "EmailTo": "laura@other.com,jack@other.com,john@other.com",
      "EmailCc": "danny@other.com,lucas@other.com,terese@other.com",
      "EmailInternetMessageId": "",
      "EmailBodyPreview": "",
      "EmailImportance": "",
      "EmailConversationId": "",
      "EmailSize": 235,
      "EmailIsRead": 0,
      "EmailIsDraft": 0,
      "EmailHasAttachment": 0,
      "Type": "EMAIL",
      "Version": 1,
      "Plugin": "m365"
    }
  },
  {
    "total-backup": "12"
  },
  {
    "total-index": "12"
  }
```

That part represents the information that would be indexed in the backup (included into the catalog). You can also see the total entries at the end, that are very useful to quickly compare with the original 12 value and so, knowing if our expression is filtering the expected data or not. Below we provide an example where some filtering is applied to the backup, but also to the index:

Listing 16: **Expression tester Test command, index part output**

```
.query client=127.0.0.1-fd plugin="m365: tenant=xxxx-xx-xxxx-xxx-xxxx email_messages_
↪exclude_expr=\"emailFrom == 'elon@other.com'\" email_messages_exclude_index_expr=\
↪"emailSubject.includes('private')\"" parameter=json|email-expr-test
...
     {
    "meta-email-4": {
      "EmailId": "",
      "EmailOwner": "test@test.com",
      "EmailTenant": "johndoe.onmicrosoft.com",
      "EmailTags": "orange,black,white,purple",
      "EmailSubject": "This is orange subject 8",
      "EmailFolderName": "/",
      "EmailFrom": "bob@company.com",
      "EmailTo": "laura@company.com,jack@company.com,john@company.com",
      "EmailCc": "danny@company.com,lucas@company.com,terese@company.com",
      "EmailInternetMessageId": "",
      "EmailBodyPreview": "",
      "EmailImportance": "",
      "EmailConversationId": "",
      "EmailSize": 232,
      "EmailIsRead": 0,
      "EmailIsDraft": 0,
      "EmailHasAttachment": 0,
      "Type": "EMAIL",
      "Version": 1,
      "Plugin": "m365"
    }
  },
  {
    "total-backup": "6"
  },
  {
    "total-index": "4"
  }
```

In case your expression is not valid, the plugin will also inform about that with the following message:

- error=Error listing elements. Cause: Predicate test error!! Review your query ....

Fileset examples

Backup Full MailBox of some users, but excluding some folders:

Listing 17: **Fileset Example**

```
FileSet {
   Name = fs-m365-drive-adjon-users-notemp
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=email tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-eb5d-
```

(continues on next page)

(continued from previous page)

```
   42nf-bac7-7b019fd284g5 user=\"adelev@baculaenterprise.onmicrosoft.com,
↪jonis@baculaenterprise.onmicrosoft.com\"
   email_files_exclude=\"*.temporary\""
   }
}
```

Backup all MailBoxes:

Listing 18: **Fileset Example**

```
FileSet {
   Name = fs-m365-email-all
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=email tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-eb5d-
   42nf-bac7-7b019fd284g5"
   }
}
```

Backup only the Inbox folder of some users:

Listing 19: **Fileset Example**

```
FileSet {
   Name = fs-m365-email-2user-inbox
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=email tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-eb5d-
   42nf-bac7-7b019fd284g5 user="peter@mycompany.com,john@mycompany.com" email_files=inbox
↪"
   }
}
```

Backup some users and include MIME messages:

Listing 20: **Fileset Example**

```
FileSet {
   Name = fs-m365-email-2user-mime
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=email tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-eb5d-
   42nf-bac7-7b019fd284g5 user="peter@mycompany.com,miriam@mycompany.com" email_mime=yes"
   }
}
```

Disable backup of multi-level attachments:

Listing 21: **Fileset Example**

```
FileSet {
   Name = fs-m365-email-multilevel-attach
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=email tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-eb5d-
   42nf-bac7-7b019fd284g5 user="peter@mycompany.com,miriam@mycompany.com" multilevel_
↪attach=no"
   }
}
```

Well known folders

Microsoft 365 can present the folders information in local languages to the user.

In general, there is no 'multilanguage' support, in the sense that folders must be included with their original name. For example, if you create a folder named 'books', you cannot expect it to be backed up if you use something like 'livres' or 'libros' from other languages. You need to use the real name that was used to create such folder.

There is one very important special case though, which is 'well known folders'. Well known folders are folders like 'inbox', 'outbox', 'archive'... A full list can be found here: https://docs.microsoft.com/en-us/graph/api/resources/mailfolder?view=graph-rest-1.0

That kind of folders can be 'found' by the plugin using their 'well known name', instead their internal id, as it's the general case. Therefore, for them it is possible to get the folder using their English well known word even if the user sees the folder with a translated word.

For example, to backup inbox it is needed to use 'inbox' even if for some user it is 'Posteingang' or 'boîte de réception'. Microsoft 365 Plugin will recognize this special words and will query the information through them.

To summarize:

- Well known folders -> Use English word
- Other user folders -> Use original name

4.3 Calendars

Bacula Enterprise Microsoft 365 Plugin can protect M365 Calendars associated to users or groups:

- Groups on M365 have only one calendar assigned, without the possibility to add more
- Users can utilize calendar selection features using the files* parameters. Target users can also be customized using user* parameters.

It is important to note that the Calendar of a group can only be accessed using delegated permissions. Delegated permissions mean the plugin will impersonate a logged in user in order to access the required information. Therefore, to backup a calendar of a group, the logged in user needs to belong to that group.

The information protected with this module is detailed below:

- For Users:
  - Calendar Groups
  - Calendars

        - Events (Metadata and contents)
            - Master events are backed up only once
            - Restoring master event results in the application of all event instances
        - Attachments
            - File Attachments
            - Item Attachments
            - Reference Attachments

- For Groups
    - Calendar
        - Events (Metadata and contents)
            - Master events are backed up only once
            - Restoring master event results in the application of all event instances
        - Attachments * File Attachments * Item Attachments * Reference Attachments
        - Calendar for groups uses *Delegated permissions*

Calendar module includes the following features:

- Incremental backup (it is done without Delta function)
- Multilevel-attachments:
    - M365 supports having messages or event objects as the attachment of a given message object. Those attachments can present additional attachments. This situation is what the M365 Plugin considers multilevel-attachments. Some particular objects like Files, when present in this multilevel way, are embedded in the metadata of the message attached. This implies that the plugin must get them in a single HTTP call and load them entirely in memory. If there is not enough memory available on the system, the variable **multilevel_attach** can be disabled and those objects will be ignored.
    - Multilevel attachmets are backed up at the same level. This means that a message with this structure:
        - Base Event have 1 Event-Attachment, Event Attachment 1 has a Message Attachment 2, and Message Attachment 2 has a File attachment
        - When restored, the result will be all Attachments being restored to the same level, not one inside the another as they were originally. For example:
        - Base Event
            - Event-Attachment 1
            - Message-Attachment 2
            - File Attachment

Catalog structure for Calendar and Events presents the same structure as the Email module (changing ‘email’ for ‘calendar’ in the path ; event extensions are .evt in comparison to the .msg extension of messages). For more information, please check section *4.2 Email/Mailboxes*

Backup parameters

The list below shows the specific backup parameters that can be set up in order to control the behavior of the calendar module.

In order to select the calendar module, the common *service* parameter must be equals or be containing the value *calendar*.

Entities that can include calendars are: users (calendar groups, calendars and events) and groups (one calendar and events).

Option	Required	Default	Values	Example	Description
calendar_files	No		Strings representing existing **calendars** for the given users or groups separated by ‘,’	Holidays, Meetings	Backup only specified **calendars** belonging to the selected entities (Users, Groups, Sites..)
calendar_files_exclude	No		Strings representing existing **calendars** for the given users or groups separated by ‘,’	Trainings, TeamBeer	Exclude selected **calendars** belonging to the selected entities
calendar_files_regex_include	No		Valid regex	.*Company	Backup matching calendars. Please, only provide list parameters (files + files_exclude) or regex ones. But do not try to combine them.
calendar_files_regex_exclude	No		Valid regex	.*Plan	Exclude matching calendars from the selection. Please, only provide list parameters (files + files_exclude) or regex ones. But do not try to combine them. If this is the only parameter found for selection, all elements will be included and this list will be excluded.
calendar_multilevel_attach	No	Yes	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	No	Include multilevel attachments in backup. A multilevel attachment is, for example, a message attached to a message containing more elements attached inside. Under certain circumstances, multilevel attachments could imply higher memory needs, as some of them are embedded in the metadata of the objects to backup, so they need to be caught in a single call and be loaded entirely in memory. This parameter is intended to control the behavior against that situation.
calendar_exclude_attachments	No	No	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	Yes	Exclude any attachment from backup

Note: In previous versions it was possible to use files_* parameters instead of calendar_files* parameters,. These parameters are now deprecated. We recommend to use module specific parameters in order to have better control and more possibilities inside a single fileset. However, jobs using files_* parameters will still work.

Restore

The list below shows the subset of restore parameters that can be used to control the behavior of calendar module restore operations:

- destination_user, destination_group, destination_path, send_report, allow_duplicates
- restore_share_permissions, calendar_name, debug, foreign_container_generation

Use cases

The following restore scenarios are supported:

- Restore calendars, events, or attachments to the original user or to a different user
  - Restore parameters implied: **destination_user**
- Restore calendars, events, or attachments to the original calendar group or to a different calendar group
  - Restore parameters implied: **destination_path**
- Restore calendars, events, or attachments to the original calendar or to a different calendar
  - Restore parameters implied: **calendar_name**
- Restore calendars, events, or attachments to local file system (general restore **where** parameter must be set to a path)
- It is possible to control whether or not duplicate elements are allowed (based on file id):
  - Restore parameters implied: **allow_duplicates**

Particularities:

- If no **destination_user** is set, every event will be restored into its original calendar
- If no **calendar_name** is set, every event will be restored into its original calendar
  - If the selection contains events from several users:
    - Original user messages will be restored in their original location
    - For other users, a special calendar will be created with the email address of each of them in the name, unless the parameter **foreign_container_generation** is disabled
      - If **foreign_container_generation** is disabled, a calendar with the same name present in the original user will be looked for in the destination user, if it does not exist, an error will be triggered
  - Restored elements will be duplicated by default, unless the **allow_duplicates** variable is disabled
    - Even disabling that variable, events will be checked by id. So if there is an element with the same information but a different ID, it will not be considered to be a duplicate

For more details about the behavior of each parameter, please check the general section of restore parameters.

Fileset examples

Backup all calendars of one user:

Listing 22: **Fileset Example**

```
FileSet {
   Name = fs-m365-calendar-adelev
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=calendar tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-
   eb5d-42nf-bac7-7b019fd284g5 user=adelev@baculaenterprise.onmicrosoft.com"
   }
}
```

Backup all calendars from all users or groups:

Listing 23: **Fileset Example**

```
FileSet {
   Name = fs-m365-calendar-all
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=calendar tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-
   eb5d-42nf-bac7-7b019fd284g5"
   }
}
```

Backup all calendars from a group:

Listing 24: **Fileset Example**

```
FileSet {
   Name = fs-m365-calendars-dev-team
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=calendar tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-
   eb5d-42nf-bac7-7b019fd284g5 group=\"Dev Team\" "
   }
}
```

Backup only one calendar of some users:

Listing 25: **Fileset Example**

```
FileSet {
   Name = fs-m365-calendar-2user-inbox
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=calendar tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-
   eb5d-42nf-bac7-7b019fd284g5 group="Dev Team,Marketing team" calendar_files=CompanyCal"
```

(continues on next page)

(continued from previous page)

```
   }
}
```

Activate backup of multi-level attachments:

Listing 26: **Fileset Example**

```
FileSet {
   Name = fs-m365-calendar-multilevel-attach
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=calendar tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-
   eb5d-42nf-bac7-7b019fd284g5 user="peter@mycompany.com,miriam@mycompany.com" calendar_
↪multilevel_attach=yes"
   }
}
```

4.4 Contacts / People

Bacula Enterprise Microsoft 365 Plugin can protect M365 Contacts associated to users, but also organizational contacts that are common to a given tenant. It is possible to select what users to backup (user* parameters) and also the specific contact folders to include/exclude (files* parameters).

The information protected by this module is detailed below:

- Users:
  - Contact Folders (Main contact folders and any others)
  - Contacts
- Tenant
  - Organization contacts

Contact module includes the following features:

- Incremental backup with delta function for user contacts
  - Delta function is applied to each Contact Folder
  - Organization contacts do not support incremental backup
- **Organization contacts are read-only** in M365 Graph
  - Only local restore is supported

Organization contacts can be located in Microsoft 365 Admin Center > Users > Contacts:

Catalog structure for contacts is presented below:

- User contacts are stored in:
  - `/@m365/tenantname/users/usermail/contact/folderName/`
    - Default folder name is ‘Contact’
    - Each element has the special extension ‘.con’
- Organization contacts are stored in:

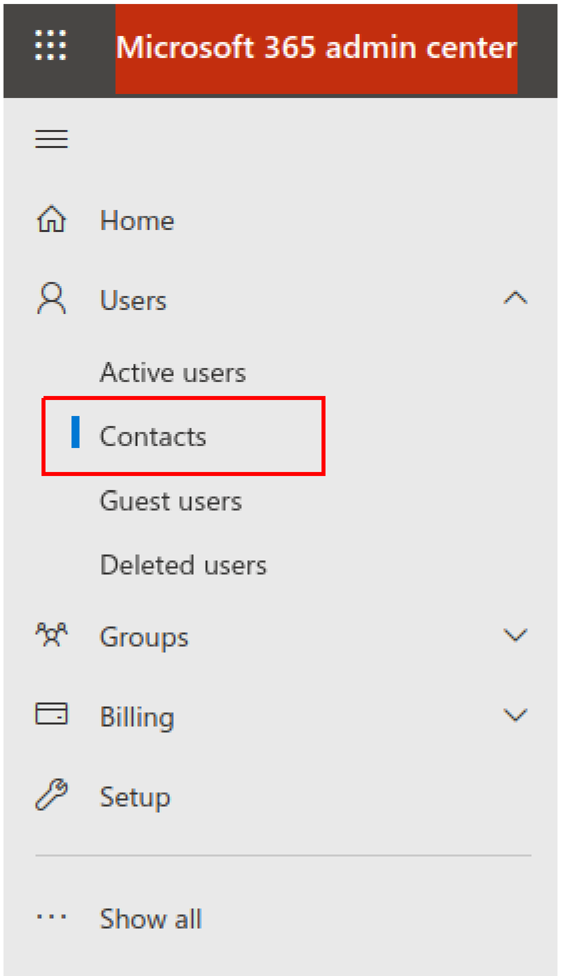


Fig. 5: Organization contacts location

- `/@m365/tenantname/users/organization/contact/Contacts/`
  - Each element has the special extension '.orgcon'

Backup parameters

The list below shows the specific backup parameters that can be set up in order to control the behavior of the contact module.

In order to select the contact module, the common *service* parameter must be or contain the value *contact*.

Entities that can include mailboxes are: users.

Option	Required	Default	Values	Example	Description
contact_files	No		Strings representing existing **contactFolders** for the given users separated by ','	Customers, Partners	Backup only specified **contactFolders** belonging to the selected users
contact_files_exclude	No		Strings representing existing **contactFolders** for the given users separated by ','	Personal	Exclude selected **contactFolders** belonging to the selected users
contact_files_regex_include	No		Valid regex	.*Company	Backup matching contact folders. Please, only provide list parameters (files + files_exclude) or regex ones. But do not try to combine them.
contact_files_regex_exclude	No		Valid regex	.*Plan	Exclude matching contact folders from the selection. Please, only provide list parameters (files + files_exclude) or regex ones. But do not try to combine them. If this is the only parameter found for selection, all elements will be included and this list will be excluded.
contact_organization	No	No	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	No	Include Organizational contacts (common to all the people in the tenant). It is recommended to include organizational contacts only in a single backup, as part of the backup strategy, but not for every user.

- Note: In previous versions, instead of contact_files* parameters, it was possible to use files_* parameters. They are deprecated parameters now as we recommend to use module specific parameters in order to have better control and more possibilities inside a single fileset. However, jobs using files_* parameters will still work.

Restore

The list below shows the subset of restore parameters that can be used to control the behavior of contact module restore operations:

- destination_user, destination_path, send_report, allow_duplicates, debug, foreign_container_generation

Use cases

The following restore scenarios are supported:

- Restore user contacts to the original user or to a different user
  - Restore parameters implied: **destination_user**
- Restore user contacts to the original contact folder group or to a different contact folder
  - Restore parameters implied: **destination_path**
- Restore user contacts to local file system (general restore **where** parameter must be set to a path)

Particularities are the same as *email module*.

For more details about the behavior of each parameter, please check the general section of *restore parameters*.

Fileset examples

Backup all contacts of one user:

Listing 27: **Fileset Example**

```
FileSet {
   Name = fs-m365-contacts-adelev
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=contact tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-
   eb5d-42nf-bac7-7b019fd284g5 user=adelev@baculaenterprise.onmicrosoft.com"
   }
}
```

Backup all contacts from all users and include organization contacts:

Listing 28: **Fileset Example**

```
FileSet {
   Name = fs-m365-contacts-all
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=contact tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-
   eb5d-42nf-bac7-7b019fd284g5 contact_organization=yes"
   }
}
```

4.5 Sharepoint

Sharepoint is a special module in the Bacula Enterprise Microsoft 365 Plugin. All the other modules rely only on the M365 Graph API to perform all of their operations. This is the current recommendation from Microsoft and that is exactly the strategy that the M365 Plugin is following. However, the Graph Sharepoint API presents several limitations which do not allow the plugin to perform a full backup/restore operation at the time of the writing this document:

- It is not possible to create Sites
- It is not possible to get or upload web pages
- It is not possible to create some types of columns for Sharepoint lists
- It is not possible to create or update items including many types of allowed Sharepoint fields

That makes a full process of backup/restore not possible today using just the MS Graph API. On the other hand, Microsoft has native Sharepoint Online APIs. These APIs are expected to be entirely replaced in the future with MS Graph APIs, but today they are supporting more advanced functions for Sharepoint Online than MS Graph does.

Sharepoint Online APIs are fully featured and significantly complex. They may be used for many different purposes. In order to simplify many automation operations over Sharepoint Online using these APIs, as well as others over other M365 services, the PnP PowerShell project was created, and the Bacula Enterprise Microsoft 365 Plugin relies on it for doing its export/import procedures involving the parts not supported by Graph APIs.

The information protected by this module is detailed below, distinguishing what is protected using each method:

- Sites Metadata (MS Graph)
  - Sites Sharing permissions (MS Graph)
- Site Templates (PnP Powershell) which includes:
  - Site metadata
  - Lists metadata
  - ListItems metadata
  - WebPages metadata
  - DocumentLibraries metadata
- Lists (MS Graph Object)
- ListItems (MS Graph Object)
  - ListItem versions
- Document Libraries Files (MS Graph Drive Items)

    - Files versions

The catalog stores the information with the following structure:

- `/@m365/tenantName/sites/siteName/`
    - siteName.site → Graph Object
    - siteName.site.template.xml → PnP template
    - drives/ → document libraries containing files
        - documents
        - sitePages
        - ...
    - lists/ → Lists containing metadata
        - documents
        - sitePages
        - ...

Version History

OneDrive and SharePoint can be configured to *retain the history for files/items*.

Sharepoint backup can protect site images. It is important to note the images are usually stored inside the 'Site Assets' list, which is a system list. In order to backup system lists it is necessary to activate the 'include_system' backup parameter.

Incremental and Differential levels are supported. However, please note that those level will not download lists metadata. Those elements are only included in Full backups, as they do not present modification dates to handle.

Backup parameters

The list below shows the specific backup parameters that can be set up in order to control the behavior of the sharepoint module.

In order to select the sharepoint module, the common *service* parameter must be equals or be containing the value *sharepoint*.

Option	Required	Default	Values	Example	Description
lists	No		Valid names of existing lists from the selected sharepoint site separated by ‘,’	siteassets, documents	Backup only selected lists of the included sharepoint sites
lists_exclude	No		Valid names of existing lists from the selected sharepoint site separated by ‘,’	sitepages	Backup all lists of the included sharepoint sites except the ones listed in this parameter
lists_regex_include	No		Valid regex	*.pages	Backup matching lists
lists_regex_exclude	No		Valid regex	^site.*	Exclude matching lists
sharepoint_subsites	No	No	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	Yes	Include subsites in selected sites for sharepoint backup
sharepoint_hidden_lists	No	No	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	Yes	Include hidden sharepoint lists of selected sites for sharepoint backup
sharepoint_system_include	No	No	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	Yes	Include system documentLibraries (sitepages, siteassets, etc).
sharepoint_version_history	No	No	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	Yes	Include item former versions of every list item into the backup process

Restore

The list below shows the subset of restore parameters that can be used to control the behavior of sharepoint module restore operations:

- send_report, local_path_json_objects, skip_versions, restore_share_permissions, debug -> Common restore parameters
- sharepoint_newsite_name, sharepoint_newsite_owner -> To be used for full site restore operations
- destination_site, destination_path, sharepoint_list_name -> To be used for list item restore operations
- sharepoint_local_template_path -> To be used for full site restore operations where we need to apply a local filesystem template, previously restored or manually built

Use Cases

The following restore scenarios are supported:

- Restore an entire site or a particular sub-site using a new site name (combines PnP and Graph actions)
  - Restore parameters implied: **sharepoint_newsite_name,sharepoint_newsite_owner**
  - ***Note:** This restore method will restore locally Graph objects like listItems or any other not applicable in the path pointed by: **local_path_json_objects**
- Restore files from document libraries to their original location or to a different document library (Graph based)
  - Refer to section *Restore*, as document libraries are managed as drive objects
- Restore listItems to their original list or to a list with the same structure in the same or different site (Graph based)
  - listItems can only contain simple fields (text, numbers..)
  - Restore parameters implied: **sharepoint_list_name**
- Restore a list to its original site or to a different site (Graph based)
  - lists can only contain simple columns (text, numbers..)
  - Restore parameters implied: **destination_site**
- Restore any information to a local filesystem (general restore **where** parameter must be set to a path)

Please note that the following scenarios are not currently supported:

- To simultaneously restore a site and its subsites:
  - It is necessary to specifically select the parent site and unmark its subsites in order to restore the parent site
  - It is necessary to select only one subsite for each restore job in order to restore them
- Restoring a site template over an existing site is not supported. It is required to always assign **sharepoint_newsite_name,** and use with a non-existing name

For more details about the behavior of each parameter, please check the general section of *restore parameters*.

Fileset examples

Backup sharepoint site:

Listing 29: **Fileset Example**

```
FileSet {
   Name = fs-m365-site-company-events
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=sharepoint tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-
   eb5d-42nf-bac7-7b019fd284g5 site=\"Company Events\" "
   }
}
```

Backup sharepoint site including hidden lists:

Listing 30: **Fileset Example**

```
FileSet {
   Name = fs-m365-site-company-events-hidden
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=sharepoint tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-
   eb5d-42nf-bac7-7b019fd284g5 site=\"Company Events\" system_include=yes sharepoint_
↪hidden_lists=yes"
   }
}
```

Backup sharepoint site and its sub-sites:

Listing 31: **Fileset Example**

```
FileSet {
   Name = fs-m365-site-company-events-sub
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=sharepoint tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-
   eb5d-42nf-bac7-7b019fd284g5 site=\"Company Developments\" sharepoint_subsites=yes "
   }
}
```

Backup a particular sharepoint sub-site:

Listing 32: **Fileset Example**

```
FileSet {
   Name = fs-m365-site-company-devs
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=sharepoint tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-
```

(continues on next page)

(continued from previous page)

```
   eb5d-42nf-bac7-7b019fd284g5 site=\"Company Developments/Doc Site\" "
   }
}
```

Backup specific sharepoint site lists:

Listing 33: **Fileset Example**

```
FileSet {
   Name = fs-m365-site-company-events-docs-travel
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=sharepoint tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-
   eb5d-42nf-bac7-7b019fd284g5 site=\"Company Events\" lists=\"documents, travel lists\"
↪"
   }
}
```

4.6 Onenote

The M365 Plugin Onenote module is able to protect any notebook belonging to users, groups, or sites of the base tenant. The pages of the notebook and any reference file are extracted in HTML format.

The information protected with this module is detailed below:

- For Users, groups or sites:
  - Notebook metadata
  - SectionGroup metadata
  - Section metadata
  - Page Html contents
  - Page Resources
    - Object files
    - Image files

Onenote module includes the following features:

- Incremental backup (it is done without the Delta function)
- It is possible to exclude images or file resources associated to pages from the backup
- It is possible to control the destination notebook, section, or section group at restore time (in addition to the general destination entity: user, group or site)

The catalog stores the information with the following structure:

- `/@m365/tenantName/entityKind/entityName/onenote/`
  - notebooks/notebookName
    - notebokName.note.book
      - sectionGroupName1

        - sectionGroupName2
        - …
    - sectionA
        - page1.note.page.html
        - resource1.note.res.png
        - resource1.note.res.pdf
        - …

Please, note that Onenote is probably the most sensitive module to throttling limits in Microsoft 365 APIs: https://docs.microsoft.com/es-es/graph/throttling

It is recommended to select the information to be protected of it and spread the jobs during time in order to avoid rejections for this reason.

Backup parameters

The list below shows the specific backup parameters that can be set up in order to control the behavior of the onenote module.

In order to select the onenote module, the common *service* parameter must be or contain the value *onenote*.

Option	Required	Default	Values	Example	Description
notebooks	No		Valid names of existing notebooks from the selected entities to backup separated by','	My notebook, John @ Work	Backup only selected notebooks of the included entities (users, groups, or sites)
notebooks_exclude	No		Valid names of existing notebooks from the selected entities to backup separated by','	Quicknotes	Backup all notebooks of the included entities except the ones listed in this parameter
notebooks_regex_include	No		Valid regex	*.buylist	Backup matching notebooks
notebooks_regex_exclude	No		Valid regex	^Work.*	Exclude matching notebooks
onenote_exclude_page_images	No	No	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	Yes	Do not backup images from pages of selected notebooks
onenote_exclude_page_files	No	No	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	Yes	Do not backup objects (files that are not images) from pages of selected notebooks

Restore

The list below shows the subset of restore parameters that can be used to control the behavior of onenote module restore operations:

- destination_user, destination_site, destination_group, send_report, local_path_json_objects, debug -> Common restore parameters
- notebook_name, notesection_name, notesection_group_name -> Control onenote destination element. Usually you only need to set up a non existing notebook_name to restore one entire notebook into a new one.

Use cases

The following restore scenarios are supported:

- Restore user, group, or site notebooks to their original entity or to a different entity
  - Restore parameters implied: **destination_user**, **destination_group**, **destination_site**
- To restore information to its original location, do not set the following parameters:
  - Restore over a new notebook: **notebook_name**
  - Restore over a new section: **notesection_name**
  - Restore over a new section group: **notesectiongroup_name**
- Restore to local file system (general restore **where** parameter must be set to a path)

Particularities about **foreign_container_generation** present the same behavior as the *email module*.

For more details about the behavior of each parameter, please check the general section of restore parameters.

Fileset examples

Backup all notebooks of a user:

Listing 34: **Fileset Example**

```
FileSet {
        Name = fs-m365-peter-onenote
        Include {
                Options { signature = MD5 }
                Plugin = "m365: service=onenote tenant=57uia43-d107-17a2-a2g2-
↪aa53c10tdahc objectid=56ddf1h9-eb5d-42nf-bac7-7b019fd284g5 user=\"peter@mycompany.com\
↪" "
        }
}
```

Backup specific notebooks of a site:

Listing 35: **Fileset Example**

```
FileSet {
        Name = fs-m365-site-notebooks-plan
        Include {
                Options { signature = MD5 }
                Plugin = "m365: service=onenote tenant=57uia43-d107-17a2-a2g2-
↪aa53c10tdahc objectid=56ddf1h9-eb5d-42nf-bac7-7b019fd284g5 site=\"Company Events\"␣
↪notebooks=places,planning"
        }
}
```

Do not backup files or images:

Listing 36: **Fileset Example**

```
FileSet {
        Name = fs-m365-peter-onenote-noimages
```

(continues on next page)

(continued from previous page)

```
        Include {
                Options { signature = MD5 }
                Plugin = "m365: service=onenote tenant=57uia43-d107-17a2-a2g2-
↪aa53c10tdahc objectid=56ddf1h9-eb5d-42nf-bac7-7b019fd284g5 user=\"peter@mycompany.com\
↪" onenote_exclude_page_images=yes onenote_exclude_page_files=yes"
        }
}
```

4.7 Teams

The M365 Plugin Teams module is able to protect one or more teams of the Microsoft 365 Teams service. Public and private teams are supported, as well as public or private channels.

The information protected with this module is detailed below:

- For groups:
  - Team metadata
  - Team settings metadata
  - Team members metadata
  - Team installed applications
  - Team channels (Public channels and private channels)
    - Channel metadata
    - Channel members for private channels
    - Channel tabs metadata
    - Channel messages - Chat messages - Hosted contents

The Teams module includes the following features:

- Incremental/Differential backup
- It is possible to exclude tabs or applications from the backup
- It is possible to select what channels to include into the backup
- It is possible to restore the data as a new Team in M365

Incremental backup of teams do not download Channel metadata of existing channels, installed apps or channel tabs. Those elements are only included in Full backups, as they do not present modification dates to handle.

The catalog stores the information with the following structure:

- `/@m365/tenantName/group/groupName/teams/`
  - team/teamName
    - teamName.team
      - apps/
      - app1.teamapp
      - app2.teamapp
      - …

        - channels/
        - channel1Name/ (tabs/tab1.tab, tabs/tab2.tab)
        - 2021-04-19 10.58.52..J-123.1.channel.msg
        - 2021-04-19 09.58.52..J-123.2.channel.msg
        - 2021-04-19 09.58.52..J-123.2.file.1.host.cnt
        - 2021-04-19 09.58.52..J-123.2.file.2.host.cnt
        - channel2Name/
        - …

It is important to note how the information is distributed in M365, since many modules are related to each other. When talking about MS Teams this is very important to understand, as some information accessible from Teams is stored in other M365 locations that are already managed through other parts of the M365 Plugin.

For example, it is possible to include attachments (which is a different concept than hosted contents) in chat messages. These files are stored into the drive unit associated to the group associated to the team, so that information should be backed up simply including the drive module in the fileset.

Some of the most important relations to have in mind are:

- A team is also directly attached to a group. So we need to use group parameters in the fileset in order to control 'what team' we want to backup.
- A team can use a notebook from onenote: This is the group onenote
- A team can store files: This is the onedrive unit for the group.
- A team can have a calendar: This is the calendar of the group.
- A team has a Team Site in Sharepoint: This can also be protected using the sharepoint module (and site parameters).

Backup parameters

The list below shows the specific backup parameters that can be set up in order to control the behavior of the teams module.

In order to select the teams module, the common *service* parameter must be equals or be containing the value *teams*.

Option	Required	Default	Values	Example	Description
teams_channels	No		Valid names of existing channels from the selected teams to backup separated by','	General, Development	Backup only selected channels of the included teams (from groups parameters)
teams_channels_exclude	No		Valid names of existing channels from the selected teams to backup separated by','	Fun Channel	Backup all channels of the included teams (from groups parameters) except the ones listed in this parameter
teams_channels_regex_include	No		Valid regex	*.corp	Backup matching channels
teams_channels_regex_exclude	No		Valid regex	^Private.*	Exclude matching channels
teams_apps	No	Yes	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	No	Backup team apps from selected teams
teams_channel_tabs	No	Yes	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	No	Backup team channel tabs from selected teams

Restore parameters

The list below shows the subset of restore parameters that can be used to control the behavior of teams module restore operations:

- send_report, debug -> Common restore parameters
- team_name -> Control destination team: If a non existing one is provided, a new team will be created. If an existing team name is provided, it will be used to restore channels/apps/tabs inside. If this parameter is not set, original team name will be considered and the same logic will be applied.
- team_channel_name -> Control destination channel name: If a non existing one is provided, a new channel will be created. If an existing channel name is provided, it will be used to restore messages inside. If this parameter is not set, original channel name will be considered and the same logic will be applied.
- team_private_channels_mode -> Decide what to do about restoring private channels:
  - DELEGATED: Default value. Will ask for delegated permissions of the private channel owner
  - PUBLIC: Will convert private channels into public ones
  - SKIP: Won't restore the private channel and will simply skip it
- team_guest_members_enable -> Enable guest members (asking for interaction if needed)

Use cases

The following restore scenarios are supported:

- Restore teams data into a newly generated Team: Set team_name with a new name.
- Restore channels into an existing Team with their original name: Set team_name to the existing team.
- Restore a channel with a new name into an existing Team: Set team_name to the existing team name and team_channel_name to a new name.
- Restore messages into an existing Team/Channel: Set team_name and team_channel_name to both existing objects.

If you restore teams data into a new generated team, users and dates will be preserved as they originally were. However, if you restore data of a team inside an existing team, please be aware that messages won't keep their original users or sent dates. This is only possible for new generated teams, existent ones do not allow to create such messages, so M365 Plugin adds a text prefix with that information, but all messages are actually sent with team owner user or channel owner (private channels case).

For private channels, if delegated mode is selected, they will be restored using the same strategy than messages into an existing team (previx with user and date). If they are converted to public, users and dates will be preserved, but the mode of the channel will be converted to public (this is due to Graph API limitations).

Please note that you can select particular channels or messages and restore them into the local filesystem (as json files) or into a new team/existing (it is not mandatory to select everything to have the restore working).

Microsoft Teams and Groups are associated entities. Therefore, the restore process will generate also a group with the same name of the Team.

If you want to restore also onedrive data, onenote data, etc, you could restore first teams data and then, restore any other data using as destination_group the name of the generated team (which is the same name the new generated group gets).

Fileset examples

Backup all teams from the tenant:

Listing 37: **Fileset Example**

```
FileSet {
        Name = fs-m365-all-teams
        Include {
                Options { signature = MD5 }
                Plugin = "m365: service=teams tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-eb5d-42nf-bac7-7b019fd284g5"
        }
}
```

Backup specific team:

Listing 38: **Fileset Example**

```
FileSet {
        Name = fs-m365-site-notebooks-plan
        Include {
                Options { signature = MD5 }
                Plugin = "m365: service=teams tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-eb5d-42nf-bac7-7b019fd284g5 group=MyTeam"
        }
}
```

Backup specific team including drive files and onenote:

Listing 39: **Fileset Example**

```
FileSet {
        Name = fs-m365-site-notebooks-plan
        Include {
                Options { signature = MD5 }
```

(continues on next page)

(continued from previous page)

```
                Plugin = "m365: service=teams,drive,onenote tenant=57uia43-d107-17a2-
↪a2g2-aa53c10tdahc objectid=56ddf1h9-eb5d-42nf-bac7-7b019fd284g5 group=MyTeam"
        }
}
```

Backup ‘General’ channel and do not backup tabs:

Listing 40: **Fileset Example**

```
FileSet {
        Name = fs-m365-peter-teams-noimages
        Include {
                Options { signature = MD5 }
                Plugin = "m365: service=teams tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-eb5d-42nf-bac7-7b019fd284g5 group=MyTeam teams_channels=General␣
↪teams_channel_tabs=no"
        }
}
```

4.8 Chat

The M365 Plugin Chat module is able to protect any chat where a given user is participating.

Please, note that this module needs Delegated permissions to work, so you will need to login using the Device Auth Flow

in order to have enough permissions to get the data: *Delegated permissions*.

The information protected with this module is detailed below:

- For Users:
  - Chat metadata
  - Chat tabs metadata
  - Chat installed apps metadata
  - **Chat messages**
    - Hosted contents

Chat module includes the following features:

- Incremental/Differential backup
- It is possible to exclude tabs or applications from the backup

The catalog stores the information with the following structure:

- `/@m365/tenantName/user/userName/chats/`
  - chat1/ - chatName.chat
    - **apps/**
      - app1.chatapp
      - app2.chatapp
      - …
    - **tabs/**

    - tab1.chat.tab
    - tab2.chat.tab
    - …
    - 2021-04-19 10.58.52..J-123.1.chat.msg
    - 2021-04-19 09.58.52..J-123.2.chat.msg
    - 2021-04-19 09.58.52..J-123.2.file.1.host.cnt
    - 2021-04-19 09.58.52..J-123.2.file.2.host.cnt
    - …
  - chat2/ - …

Backup parameters

The list below shows the specific backup parameters that can be set up in order to control the behavior of the chat module.

In order to select the chat module, the common *service* parameter must be or contain the value *chat*.

Option	Required	Default	Values	Example	Description
chat_topics	No		Valid topics associated to existing chats from the selected user(s) to backup separated by','	ProjectA, Important	Backup only selected chats of the included users
chat_topics_exclude	No		Valid topics associated to existing chats from the selected user(s) to backup separated by','	Fun Channel	Backup all chats of the included users except the ones listed in this parameter
chat_topics_regex_include	No		Valid regex	*.corp	Backup matching chats (by topic)
chat_topics_regex_exclude	No		Valid regex	^Private.*	Exclude matching chats (by topic)
chat_apps	No	Yes	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	No	Backup chat apps from selected user chats
chat_tabs	No	Yes	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	No	Backup team chat tabs from selected user chats
chat_exclude_empty	No	No	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	Yes	Exclude any chat that has no messages inside
chat_exclude_type_group	No	No	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	Yes	Exclude chats of type 'group'
chat_exclude_type_one_on_one	No	No	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	Yes	Exclude chats of type 'one_on_one'

Restore parameters

The list below shows the subset of restore parameters that can be used to control the behavior of teams module restore operations:

- destination_user -> User that will be used to restore chats to it. Usually, it should be the same owner user of the messages to restore
- send_report, debug -> Common restore parameters
- chat_topic -> Control destination chat (add a new one)

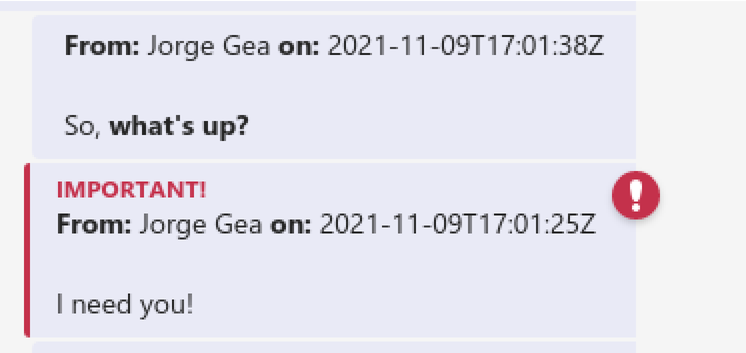
Use cases

Depending on the chat type, it is possible to:

- Group chats:
  - Restore chats as new ones, specifying a new topic with the restore variable 'chat_topic' (group chats)
  - Restore chat messages inside original chats (if they where deleted)
- One on one chats:
  - One on one chats can have only one instance per pair of users. Therefore, the restore can only create chat messages with the implied user in case they were deleted before

As with any other plugin, local filesystem restore is possible too and we would get all objects (chat, chatMessages…) as json files, except hosted contents that will be restored as the original files.

Please note that restore to M365 service function is not able to restore messages in the exact form that they were generated because of Microsoft security policy around this service. It is not possible to create a chat message with a different date or user than the real ones that are being used for the current connection. Therefore when the restore is performed what Bacula M365 Plugin does is to add a little header for each message containing that information, but the messages will be all generated using the 'destination_user' or the original user in case that variable was not used. Example:



From: Jorge Gea **on:** 2021-11-09T17:01:38Z

So, **what's up?**

IMPORTANT!
From: Jorge Gea **on:** 2021-11-09T17:01:25Z

I need you!

Fileset examples

Backup all chats of a user:

Listing 41: **Fileset Example**

```
FileSet {
        Name = fs-m365-peter-chat
        Include {
                Options { signature = MD5 }
                Plugin = "m365: service=chat tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-eb5d-42nf-bac7-7b019fd284g5 user=\"peter@mycompany.com\" "
        }
}
```

Backup chats of all users, but do not backup tabs or installed apps:

Listing 42: **Fileset Example**

```
FileSet {
        Name = fs-m365-peter-chat-noimages
        Include {
                Options { signature = MD5 }
                Plugin = "m365: service=chat tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-eb5d-42nf-bac7-7b019fd284g5 chat_apps=no chat_tabs=no"
        }
}
```

4.9 Tasks

Bacula Enterprise Microsoft 365 Plugin can protect M365 Todo Tasklists associated to users and M365 Planner tasks associated to Teams. Todo Taskslists will be considered in case user* parameters are defined, while Planner tasks will be considered in case group* parameters are defined.

For Todo Tasklists:

- It is possible to select what users to backup (user* parameters) and also the specific tasklists to include/exclude (tasklists* parameters).

The information protected by this module is detailed below:

- Users:
  - Task Lists
  - Tasks
- Groups:
  - Planner plan
  - Planner buckets
  - Planner tasks

This module includes the following features:

- Incremental/Differential backup

It is important to note here that Incremental or Differential backups won't take modified tasks. The reason is task objects do not store the information of last modification date, so we cannot determine if they were modified after creation. Therefore the recommendation is to run regularly Full backups to retrieve any modified task.

Please note the following limitatons:

- Todo Tasks LinkedResources are not supported. As of Bacula version 14.0, these objects are returned as empty entities from Graph API. This limitation is expected to be addressed in the future.
- Planner tasks won't keep their original order once they are restored and visualized through the Teams App or using the internet browser.

Catalog structure for tasks is presented below:

- User tasks are stored in:
  - `/@m365/tenantname/users/usermail/tasks/tasklist_name`
    - Default folder name is 'Tasks'
    - Each element has the special extension '.todo.task'
- Group planner tasks are stored in:
  - `/@m365/tenantname/group/groupdisplayname/tasks/plan_name/plan_bucket/planner_task_name`
    - Tasks have the special extension '.planner.task'. However other task related objects are stored also with different extensions. These objects are: task details, assignedto task board task format, progress task board task format and bucket task board task format

Please note that this module needs **Delegated permissions** to work, so it is necessary to login using the Device Auth Flow in order to have enough permissions to get the data: *Delegated permissions*

Backup parameters

The list below shows the specific backup parameters that can be set up in order to control the behavior of the tasks module.

In order to select the tasks module, the common *service* parameter must be equals or be containing the value *tasks*.

Entities that can include mailboxes are: users.

Option	Required	Default	Values	Example	Description
tasklists	No		Strings representing existing todo **tasklists** for the given users separated by ‘,’	ProjectA, ProjectB	Backup only specified **tasklists** belonging to the selected users
tasklists_exclude	No		Strings representing existing todo **tasklists** for the given users separated by ‘,’	Personal	Exclude selected **tasklists** belonging to the selected users
tasklists_regex_include	No		Valid regex	.*Company	Backup matching tasklists. Please, only provide list parameters (files + files_exclude) or regex ones. But do not try to combine them.
tasklists_regex_exclude	No		Valid regex	.*Plan	Exclude matching tasklists from the selection. Please, only provide list parameters (files + files_exclude) or regex ones. But do not try to combine them. If this is the only parameter found for selection, all elements will be included and this list will be excluded.
plannerplans	No		Strings representing existing planner **plans** for the given groups separated by ‘,’	ProjectA, ProjectB	Backup only specified **plannerplans** belonging to the selected groups
plannerplans_exclude	No		Strings representing existing planner **plans** for the given groups separated by ‘,’	Personal	Exclude selected planner **plans** belonging to the selected groups
plannerplans_regex_include	No		Valid regex	.*Company	Backup matching planner plans. Please, only provide list parameters (files + files_exclude) or regex ones. But do not try to combine them.
plannerplans_regex_exclude	No		Valid regex	.*Plan	Exclude matching planner plans from the selection. Please, only provide list parameters (files + files_exclude) or regex ones. But do not try to combine them. If this is the only parameter found for selection, all elements will be included and this list will be excluded.

Restore

The list below shows the subset of restore parameters that can be used to control the behavior of tasks module restore operations:

- destination_user, destination_group, send_report, debug, tasklist_name, tasklist_skip_sharedwithme

Use cases

The following restore scenarios are supported:

- Restore task lists and their tasks into to their original user or to a different user
  - Restore parameters implied: **destination_user**
  - Destination tasklist (new or existent one): tasklist_name
  - Skip or not shared tasklists: tasklist_skip_sharedwithme
- Restore planner plans and their tasks into to their original groups (teams) or to a different group (team)
  - Restore parameters implied: **destination_group**
  - Destination plan (new or existent one): plan_name
  - Create or not associated tab: plan_create_tab

For more details about the behavior of each parameter, please check the general section of restore parameters.

Fileset examples

Backup all todo tasks of one user:

Listing 43: **Fileset Example**

```
FileSet {
   Name = fs-m365-tasks-adelev
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=tasks tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-
   eb5d-42nf-bac7-7b019fd284g5 user=adelev@baculaenterprise.onmicrosoft.com"
   }
}
```

Backup todo tasklist ‘work’ of one user:

Listing 44: **Fileset Example**

```
FileSet {
   Name = fs-m365-tasks-adelev
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=tasks tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-
   eb5d-42nf-bac7-7b019fd284g5 user=adelev@baculaenterprise.onmicrosoft.com␣
↪tasklists=work"
   }
}
```

Backup all todo tasks from all users

Listing 45: **Fileset Example**

```
FileSet {
   Name = fs-m365-tasks-all
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=tasks tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-
   eb5d-42nf-bac7-7b019fd284g5"
   }
}
```

Backup all planner plans of one group:

Listing 46: **Fileset Example**

```
FileSet {
   Name = fs-m365-planner-dev
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=tasks tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-
   eb5d-42nf-bac7-7b019fd284g5 group=DevTeam"
   }
}
```

Backup all planner plans from all groups

Listing 47: **Fileset Example**

```
FileSet {
   Name = fs-m365-planner-all
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=tasks tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-
   eb5d-42nf-bac7-7b019fd284g5"
   }
}
```

Backup planner plan ‘product’ of one user:

Listing 48: **Fileset Example**

```
FileSet {
   Name = fs-m365-tasks-adelev
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=tasks tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-
   eb5d-42nf-bac7-7b019fd284g5 group=DevTeam plannerplans=product"
   }
}
```

5 Special features

In the following section, special features and behaviors are detailed.

5.1 Item versions

OneDrive and SharePoint can be configured to retain the history for files/items. Depending on the service and configuration, a new version can be created for each edit, each time the file is saved, manually, or never. Previous versions of a document may be retained for a finite period of time depending on admin settings which may be unique per user or location. This feature is usually enabled and the information may be accessed through the ‘Version history’ option available once a file has been selected.

Bacula Enterprise Microsoft 365 Plugin is able to backup this information if a special **drive_version_history.** backup parameter is activated.

Versions are backed up in a special way:

- They are backed up as a regular file. This means a version has its own full metadata as the parent file itself has. All the metadata is the same as the file contains, except for size, dates and name.
- The name of the file is modified, so at restore time you can see the version number and the version date in the filename. Example:
  - Parent file: myDoc.doc
  - Versions:
    - myDoc###v25.0_2021-01-19_234537.doc
    - myDoc###v24.0_2021-01-17_212537.doc
    - myDoc###v23.0_2021-01-12_104537.doc
    - …
    - *Notice that the extension of the file is kept in order to easily identify a possible name modification in OneDrive, once the file is restored
- Versions are not restored by default. You need to disable the special restore parameter ‘drive_skip_versions’, setting it to 0.

File versions are backed up in all backup levels (Full, Incremental, Differential), this means you can track all the changes of the files in your backup. For example, every Incremental run is going to backup only the new modified versions since the last Full or Incremental execution.

Here is an example of a file backed up, listed in a restore session:

Listing 49: **deltaLink in restore session**

```
Automatically selected FileSet: FS_M365_DRIVE
+-------+-------+----------+-------------+---------------------+-------------------+
| jobid | level | jobfiles | jobbytes    | starttime           | volumename        |
+-------+-------+----------+-------------+---------------------+-------------------+
|    11 | F     |      190 | 332,978,505 | 2021-01-22 10:39:34 | TEST-2021-01-22:0 |
|    12 | I     |        1 |         550 | 2021-01-22 10:43:05 | TEST-2021-01-22:0 |
+-------+-------+----------+-------------+---------------------+-------------------+
You have selected the following JobIds: 11,12

Building directory tree for JobId(s) 11,12 ...
+++++++++++++++++++++++++++++++++++++++++++++++
188 files inserted into the tree.

You are now entering file selection mode where you add (mark) and
remove (unmark) files to be restored. No files are initially added, unless
you used the "all" keyword on the command line.
Enter "done" to leave this mode.

cwd is: /
$ cd @m365/baculaenterprise/users/adelev@baculaenterprise.onmicrosoft.com/drive/root:
cwd is: /@m365/baculaenterprise/users/adelev@baculaenterprise.onmicrosoft.com/drive/
↪root:/
$ ls
Docs/
pluginTest.drive.deltaLink
sharedWithMe/
versionedSampleWeb###v1.0_2020-11-25_153507.html
versionedSampleWeb###v2.0_2020-11-25_153507.html
versionedSampleWeb###v3.0_2020-12-02_180612.html
versionedSampleWeb###v4.0_2020-12-02_180612.html
versionedSampleWeb###v5.0_2020-12-03_134422.html
versionedSampleWeb###v6.0_2020-12-02_180612.html
versionedSampleWeb.html
docVersioned###v1.0_2020-11-26_123244.txt
docVersioned###v2.0_2020-11-26_123244.txt
docVersioned###v3.0_2020-12-02_180613.txt
docVersioned###v4.0_2020-12-02_180613.txt
docVersioned.txt
$
```

Note: It is important to keep in mind that versions have no Delta function and cannot be filtered by date. Therefore, the process to decide if a version needs to be backed up or not requires the plugin to walk through all existent versions of a modified item. In some situations this could have some undesired impact on backup performance.

5.2 Delta Backup

The Microsoft Graph API provides a Delta function to track changes of some objects. Bacula Enterprise Microsoft 365 Plugin uses this function in order to speed up Incremental/Differential processes. Please note that Delta function is not a mandatory requirement and that Incremental or Differential backups will function also with the services that currently do not support it.

Delta function has some important characteristics:

- In OneDrive entities backup it can only be used for full entities (full user, full group, or full site). This means that selecting specific paths to backup will not trigger the Delta function.
- Delta tokens can expire at some point, or even become invalid due to internal Microsoft issues. If this situation happens, the plugin will try to start a new Delta cycle
- Any situation where the Delta function cannot be used will trigger a regular Full/Inc/Diff where every element is listed and selected or discarded according to the item dates.

The Delta backup cycle is described below:

- Full backup: All entity elements are backed up. A token (token_1) is generated and this token is stored locally by the FD.
- Incremental 1 backup: token_1 is used to retrieve changes since token_1's generation so every change is backed up. A new token is generated and stored locally by the FD.
- Incremental 2 backup: token_2 is used to retrieve changes since token_2's generation so every change is backed up. A new token is generated and stored locally by the FD.
- And so on…

Tokens are stored in an file placed in a path defined by the **path** parameter of the plugin. The name is: jobname.deltaLink

The file stores tokens required for every execution and it is renewed (emptied) during every Full backup execution.

This file is also backed up in the backup itself, so it can be restored manually, before an Incremental/Differential execution in case it was lost and in case you don't want to run a Full backup again.

Services supporting Delta backup are: **email**, **onedrive** and **contacts**.

Here we can see an example of the contents of the file, with 3 executions and one user entity involved. The structure is tree-based, so it is easy to understand what would be generated in case of backing up other services or entities:

Listing 50: **deltaLink**

```
{
"jobName": "M365-DRIVE-ADELE-BACK",
"deltaServices": {
      "DRIVE": {
         "entities": {
            "4bfec6ba-6e0c-455f-a86d-1dbfdd1c5754": {
               "id": "4bfec6ba-6e0c-455f-a86d-1dbfdd1c5754",
               "name": "adelev@baculaenterprise.onmicrosoft.com",
               "containers": {
                  "b!7Tf3z7izSES7kjOHD-YM-0kScgNXL6lFnL4O2LWLMy41XlWc2E0pTKfPzlKxhztE": {
                  "id": "b!7Tf3z7izSES7kjOHD-YM-
↪0kScgNXL6lFnL4O2LWLMy41XlWc2E0pTKfPzlKxhztE",
                  "description": "onedrive",
                  "deltaEntries": [
                     {
```

(continues on next page)

(continued from previous page)

```
                     "job": "M365-DRIVE-ADELE-BACK.2021-04-13_13.42.42_21",
                     "date": "Apr 13, 2021 1:42:42 PM",
                     "delta":

↪"MzslMjM0OyUyMzE7Mzs5YzU1NWUzNS00ZGQ4LTRjMjktYTdjZi1jZTUyYjE4NzNiNDQ7NjM3NTM5MTA5NjczMzAwMDAwOzI4Nzc5
7JTIzaXRCQmNGclZjVGpya0x1YmZhdEpWWHpWdlolMjUyRkxPN1FJYmM4OXNGJTI1MkZZZkxZJTI1M0Q7JTIzMA"
                     },
                     {
                     "job": "M365-DRIVE-ADELE-BACK.2021-04-13_13.43.47_22",
                     "date": "Apr 13, 2021 1:43:47 PM",
                     "delta":

↪"MzslMjM0OyUyMzE7MTtjZmY3MzdlZC1iM2I4LTQ0NDgtYmI5Mi0zMzg3MGZlNjBjZmI7NjM3NTM5MTEwMzExNDAwMDAwOzI4Nzc5
                    ␣
↪7JTIzaXRCQmNGclZjVGpya0x1YmZhdEpWWHpWdlolMjUyRkxPN1FJYmM4OXNGJTI1MkZZZkxZJTI1M0Q7JTIzMA
↪"
                     },
                     {
                     "job": "M365-DRIVE-ADELE-BACK.2021-04-13_13.44.39_23",
                     "date": "Apr 13, 2021 1:44:39 PM",
                     "delta":

↪"MzslMjM0OyUyMzE7MTtjZmY3MzdlZC1iM2I4LTQ0NDgtYmI5Mi0zMzg3MGZlNjBjZmI7NjM3NTM5MTEwODMwMTAwMDAwOzI4Nzc5
7JTIzaXRCQmNGclZjVGpya0x1YmZhdEpWWHpWdlolMjUyRkxPN1FJYmM4OXNGJTI1MkZZZkxZJTI1M0Q7JTIzMA"
                     }
                  ]
               }
            }
         }
      }
   }
}
```

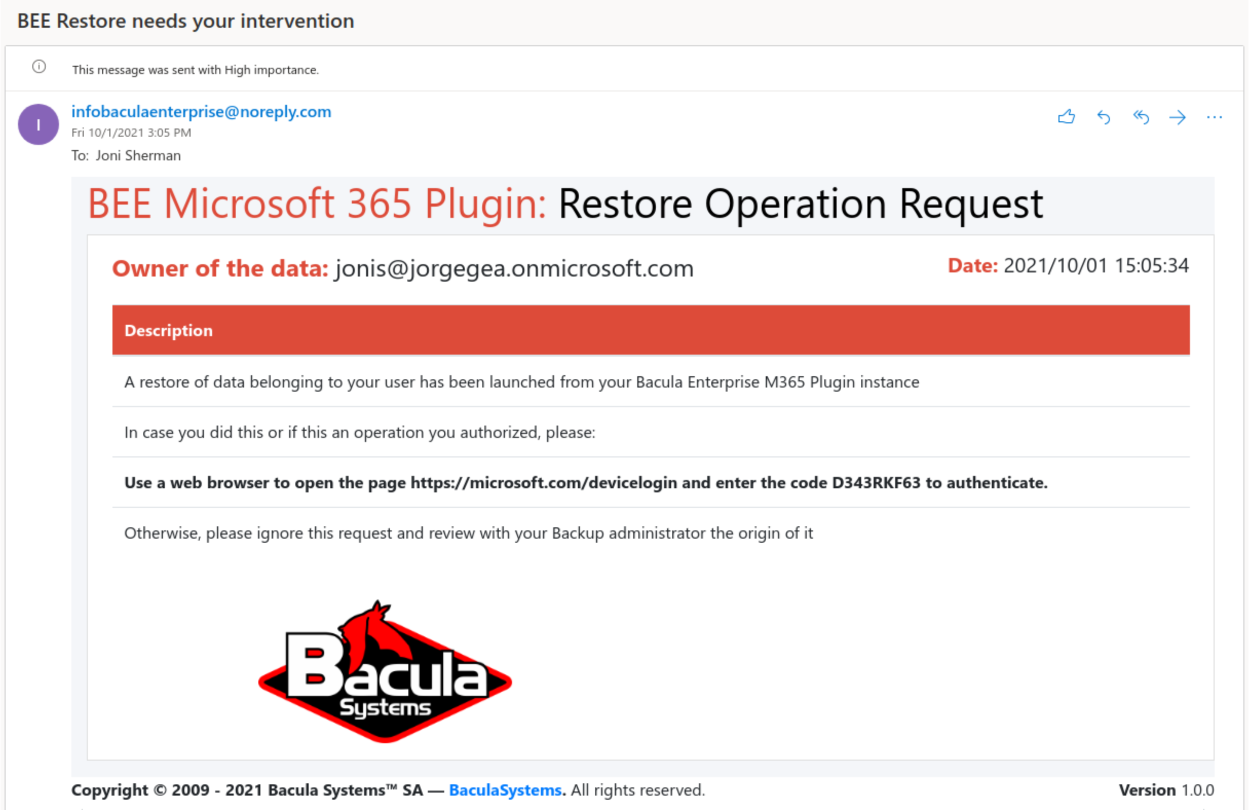
5.3 Data owner restore protection

Bacula Systems is aware about one of many privacy concerns that may arise when tools like this M365 Plugin enables the possibility to backup and restore data coming from different users, so the backup administrator can restore potentially private data at will. To address this concern, **Bacula Enterprise** Microsoft 365 Plugin includes the **Owner restore protection** feature.

This feature is enabled at configuration time with the parameter 'owner_restore_protection' (please check the Configuration section of this document for further information). Once it is enabled, any restore operation to a different user than the original owner will trigger the intervention of the owner of the data. The restore job will be paused and a message asking to enter a Microsoft 365 page and enter a code will be logged in the joblog. A similar message will also be sent via email to the affected user:

If the user does not confirm the operation within 15 minutes, the restore will fail and no data will be restored. However, if the user knows that operation and wants to approve it, as soon as the owner credentials are confirmed, the restore will resume and the data will be processed as usual and restored to any configured destination.

Note: This feature is available starting with version 14.0 of Bacula Enterprise



BEE Restore needs your intervention

This message was sent with High importance.

infobaculaenterprise@noreply.com

Fri 10/1/2021 3:05 PM

To: Joni Sherman

BEE Microsoft 365 Plugin: Restore Operation Request

Owner of the data: jonis@jorgegea.onmicrosoft.com

Date: 2021/10/01 15:05:34

Description
A restore of data belonging to your user has been launched from your Bacula Enterprise M365 Plugin instance
In case you did this or if this an operation you authorized, please:
Use a web browser to open the page https://microsoft.com/devicelogin and enter the code D343RKF63 to authenticate.
Otherwise, please ignore this request and review with your Backup administrator the origin of it

Copyright © 2009 - 2021 Bacula Systems™ SA — BaculaSystems. All rights reserved. **Version** 1.0.0

Fig. 6: Restore Operation Request example

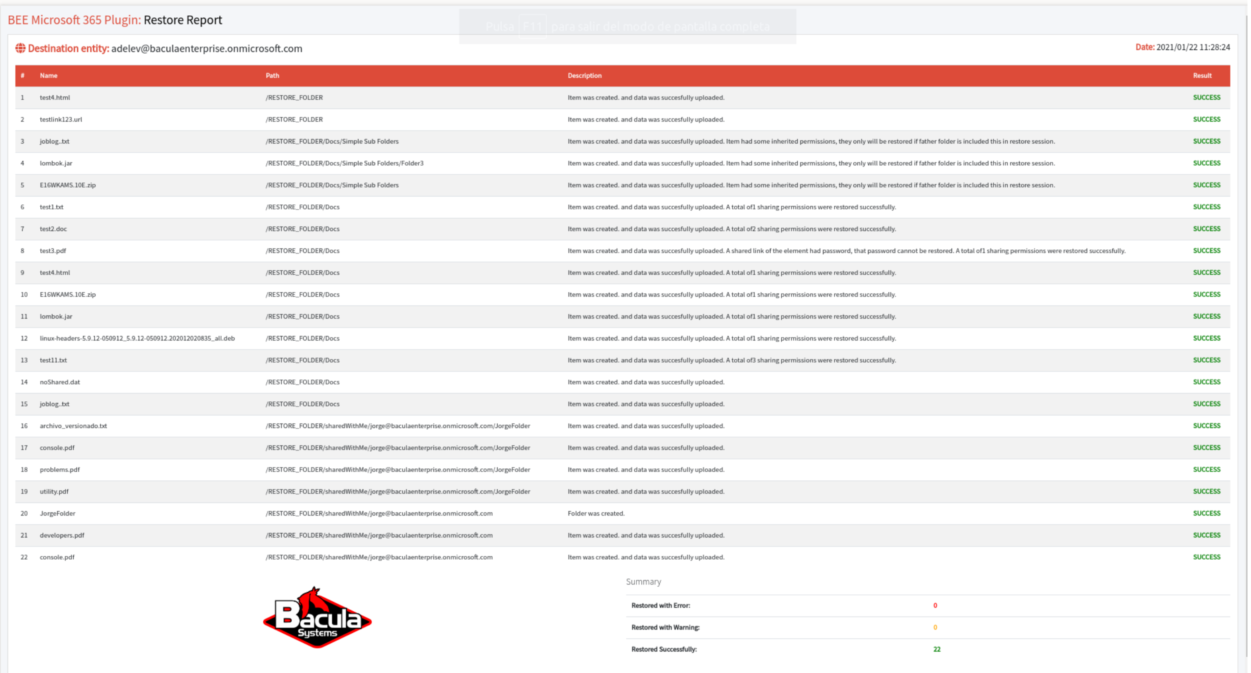
5.4 User restore report

Files and emails can represent very sensitive information for end-users. For that reason, information included in backup/restore logs is not exhaustive by default. For example, email restores do not include information such as the subject or sender when they are displayed in the backup log. However, for reporting and controlling purposes, the information of what has been exactly restored, what permissions have been applied, and other information can be useful and necessary for the affected user.

Bacula Enterprise Microsoft 365 Plugin includes an option to generate a restore report in the affected user's service (Drive or Mailbox). The restore report contains detailed information about the items that have been restored successfully, if any of them had any trouble during the restore, and it also reports the date when the action was performed.

The generation of the report can be enabled/disabled in the bconsole restore session. If enabled, depending on the service, the report can generate an HTML file or an email in the Inbox of the affected user.

The image below shows an example report from a Drive restore session:



BEE Microsoft 365 Plugin: Restore Report

Destination entity: adelev@baculaenterprise.onmicrosoft.com

Date: 2021/01/22 11:28:24

#	Name	Path	Description	Result
1	test4.html	/RESTORE_FOLDER	Item was created. and data was succesfully uploaded.	SUCCESS
2	testlink123.url	/RESTORE_FOLDER	Item was created. and data was succesfully uploaded.	SUCCESS
3	joblog..txt	/RESTORE_FOLDER/Docs/Simple Sub Folders	Item was created. and data was succesfully uploaded. Item had some inherited permissions, they only will be restored if father folder is included this in restore session.	SUCCESS
4	lombok.jar	/RESTORE_FOLDER/Docs/Simple Sub Folders/Folder3	Item was created. and data was succesfully uploaded. Item had some inherited permissions, they only will be restored if father folder is included this in restore session.	SUCCESS
5	E16WKAMS.10E.zip	/RESTORE_FOLDER/Docs/Simple Sub Folders	Item was created. and data was succesfully uploaded. Item had some inherited permissions, they only will be restored if father folder is included this in restore session.	SUCCESS
6	test1.txt	/RESTORE_FOLDER/Docs	Item was created. and data was succesfully uploaded. A total of1 sharing permissions were restored successfully.	SUCCESS
7	test2.doc	/RESTORE_FOLDER/Docs	Item was created. and data was succesfully uploaded. A total of2 sharing permissions were restored successfully.	SUCCESS
8	test3.pdf	/RESTORE_FOLDER/Docs	Item was created. and data was succesfully uploaded. A shared link of the element had password, that password cannot be restored. A total of1 sharing permissions were restored successfully.	SUCCESS
9	test4.html	/RESTORE_FOLDER/Docs	Item was created. and data was succesfully uploaded. A total of1 sharing permissions were restored successfully.	SUCCESS
10	E16WKAMS.10E.zip	/RESTORE_FOLDER/Docs	Item was created. and data was succesfully uploaded. A total of1 sharing permissions were restored successfully.	SUCCESS
11	lombok.jar	/RESTORE_FOLDER/Docs	Item was created. and data was succesfully uploaded. A total of1 sharing permissions were restored successfully.	SUCCESS
12	linux-headers-5.9.12-050912_5.9.12-050912.202012020835_all.deb	/RESTORE_FOLDER/Docs	Item was created. and data was succesfully uploaded. A total of1 sharing permissions were restored successfully.	SUCCESS
13	test11.txt	/RESTORE_FOLDER/Docs	Item was created. and data was succesfully uploaded. A total of3 sharing permissions were restored successfully.	SUCCESS
14	noShared.dat	/RESTORE_FOLDER/Docs	Item was created. and data was succesfully uploaded.	SUCCESS
15	joblog..txt	/RESTORE_FOLDER/Docs	Item was created. and data was succesfully uploaded.	SUCCESS
16	archivo_versionado.txt	/RESTORE_FOLDER/sharedWithMe/jorge@baculaenterprise.onmicrosoft.com/JorgeFolder	Item was created. and data was succesfully uploaded.	SUCCESS
17	console.pdf	/RESTORE_FOLDER/sharedWithMe/jorge@baculaenterprise.onmicrosoft.com/JorgeFolder	Item was created. and data was succesfully uploaded.	SUCCESS
18	problems.pdf	/RESTORE_FOLDER/sharedWithMe/jorge@baculaenterprise.onmicrosoft.com/JorgeFolder	Item was created. and data was succesfully uploaded.	SUCCESS
19	utility.pdf	/RESTORE_FOLDER/sharedWithMe/jorge@baculaenterprise.onmicrosoft.com/JorgeFolder	Item was created. and data was succesfully uploaded.	SUCCESS
20	JorgeFolder	/RESTORE_FOLDER/sharedWithMe/jorge@baculaenterprise.onmicrosoft.com	Folder was created.	SUCCESS
21	developers.pdf	/RESTORE_FOLDER/sharedWithMe/jorge@baculaenterprise.onmicrosoft.com	Item was created. and data was succesfully uploaded.	SUCCESS
22	console.pdf	/RESTORE_FOLDER/sharedWithMe/jorge@baculaenterprise.onmicrosoft.com	Item was created. and data was succesfully uploaded.	SUCCESS

Summary

Restored with Error:	0
Restored with Warning:	0
Restored Successfully:	22

Fig. 7: Restore Drive Example Report

6 Installation

The Bacula File Daemon and the Microsoft 365 Plugin need to be installed on the host that is going to connect to the cloud based services. The plugin is implemented over a Java layer, therefore it can be deployed on the platform better suited for your needs among any of the officially supported platforms of **Bacula Enterprise** (RHEL, SLES, Debian, Ubuntu, Windows, etc).

Please, note that you may want to deploy your File Daemon and the plugin on a virtual machine directly deployed in Azure Cloud in order to reduce the latency between it and the Microsoft Graph API and experience modest performance gains. However, this option is **only recommended in case of having a very stable connection between the File Daemon and the Storage Daemon**, which means a special, dedicated connection with Azure or when the Storage Daemon is deployed in the cloud as well. This is not usually the case - thus the data needs to traverse the Internet with standard and shared connections from the File Daemon to the Storage Daemon. Disconnections while transmitting data between these two daemons may make jobs fail and cause large timeouts that are difficult to manage and stabilize. The best strategy strategy for this kind of scenarios is to deploy the File Daemon and the Plugin to the same host as the destination Storage Daemon is installed. This way, disconnections between the two daemons will not happen, while disconnections between the FD and M365 will be transparently recovered (when possible), so jobs will finish successfully.

The system must have Java >= 8 installed (openjdk-8-jre for example) and the Java executable should be available in the system PATH.

The Sharepoint module depends on the Powershell and the PnP Powershell modules. Therefore, they also need to be installed before installing the Bacula packages (see section *PnP.Powershell* below).

6.1 PnP.Powershell

Install PowerShell

In order to install PowerShell it is necessary to follow the instructions for the particular OS involved which may be found in the github site of the project:

- https://github.com/PowerShell/PowerShell

For example, if using Debian, these are the instructions:

- https://docs.microsoft.com/en-us/powershell/scripting/install/installing-powershell-core-on-linux?view=powershell-7.1#debian-9

The procedure is shown below, some dependencies are installed, a repository is added, and then the apt package manager is used to install the tool:

Listing 51: **Install PowerShell**

```
# Install system components
sudo apt-get update
sudo apt-get install -y curl gnupg apt-transport-https

# Import the public repository GPG keys
curl https://packages.microsoft.com/keys/microsoft.asc | sudo apt-key add -

# Register the Microsoft Product feed
sudo sh -c 'echo "deb [arch=amd64] https://packages.microsoft.com/repos/microsoft-debian-
↪stretch-prod stretch
main" > /etc/apt/sources.list.d/microsoft.list'

# Update the list of products
sudo apt-get update

# Install PowerShell
sudo apt-get install -y powershell

# Start PowerShell
pwsh
```

Install PnP.Powershell

In order to install the PnP.Powershell module, once Powershell is already installed, we simply need to run the command:

- Install-Module -Name “PnP.PowerShell”

Inside a Powershell session. Below we provide an example:

Listing 52: **Install PnP.PowerShell**

```
yourworkstation:~$ pwsh
PowerShell 7.2.0
Copyright (c) Microsoft Corporation.

https://aka.ms/powershell
Type 'help' to get help.

PS /home/john> Install-Module -Name "PnP.PowerShell"
```

6.2 Bacula Packages

We are taking Debian Buster as the example base system to proceed with the installation of the **Bacula Enterprise** Microsoft 365 Plugin. In this system, the installation is most easily done by adding the repository file suitable for the existing subscription and the Debian version utilized. An example would be /etc/apt/sources.list.d/bacula.list with the following content:

Listing 53: **APT**

```
# Bacula Enterprise
deb https://www.baculasystems.com/dl/@customer-string@/debs/bin/@version@/buster-64/␣
↪buster main
deb https://www.baculasystems.com/dl/@customer-string@/debs/m365/@version@/buster-64/␣
↪buster m365
```

After that, a run of apt update is needed:

Listing 54: **APT install**

```
apt update
```

Then, the plugin may be installed using:

Listing 55: **APT install**

```
apt install bacula-enterprise-m365-plugin
```

The plugin has two different packages implied that should be installed automatically with the command shown:

- bacula-enterprise-m365-plugin
- bacula-enterprise-m365-plugin-libs

Alternately, manual installation of the packages may be done after downloading the poackages from your Bacula Systems provided download area, and then using the package manager to install. An example:

Listing 56: **APT install**

```
dpkg -i bacula-enterprise-*
```

The package will install the following elements:

- Jar libraries in /opt/bacula/lib (such as bacula-m365-plugin-x.x.x.jar and bacula-m365-plugin-libs-x.x.x.jar). Please note that the version of those jar archives is not aligned with the version of the package. However, that version will be shown in the joblog in a message like 'Jar version:X.X.X'.

Note: Version in Jar Name

Version is included in the name of .jar files from Bacula Enterprise version 14.0.4. Before that, libraries were composed by: bacula-m365-plugin.jar, bacula-meta-plugin-1.0.0.jar and bacula-m365-plugin-libs-1.0.0.jar

- Plugin connection file (m365-fd.so) in the plugins directory (usually /opt/bacula/plugins)
- Backend file (m365_backend) that invokes the jar files in /opt/bacula/bin. This backend file searches for the most recent bacula-m365-plugin-x.x.x.jar file in order to launch it, even thought usually we should have only one file.
- A collection of powershell files used in the Sharepoint module in /opt/bacula/bin.

7 Configuration

7.1 Authorization

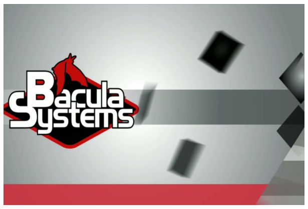
The first step in order to use the **Bacula Enterprise** Microsoft 365 Plugin is to authorize it to handle data of the target tenant to backup.

There are two possible strategies in order to allow the communication between the Bacula Enterprise Plugin and your tenant:

- **Method A (DEPRECATED): Common app model**
  - Register the pre-existing Bacula Systems **bacula-m365-plugin** Azure AD app into your tenant.
  - The communication will happen through this multi-tenant application.
  - Application Id and associated secrets are internal to the plugin.
  - Microsoft Graph limits associated to an application are common for everyone using this application (multi-customer).
  - For future new permissions you just need to click on 'Grant permissions' from Azure AD enterprise apps section
- **Method B (RECOMMENDED): Standalone app model**
  - Register the pre-existing Bacula Systems **bacula-m365-registratror** Azure AD app into your tenant. Then add a standalone application in your tenant.
  - Then add your own standalone application in your tenant calling the appropriate automatic command from bconsole
  - A **bacula-m365-plugin-standalone** Azure AD application will be created specifically for your tenant. The communication will happen through it.
  - Application Id and associated secrets need to be correctly set and they can be managed by you.

- Microsoft Graph limits associated to an application are specific to your standalone application
- For future new permissions you need to re-run the bconsole add-app command (which uses bacula-m365-registrator) or do it yourself manually

To walk through the process described in Method B with the help of a video, click on the image below:



Note: Authentication Method B is only available from Bacula Enterprise 12.8.2

The first method is simpler and faster to setup, however it is only advised for **testing purposes**. The second method needs a few extra variables to manage and it is recommended for medium or large environments. It is more secure and it can offer better performance.

Backup and restore operations will be using in general the 'Application permissions' model, where the application has enough privileges to perform all the operations without impersonating any user. However, some specific modules need to employ 'Delegated permissions'. To know more about them, please go to *Delegated permissions*

Note: Starting from Bacula Enterprise version 14.0 you can also perform these authorization tasks directly using BWeb, to see more details, please go to section *BWeb Management Console*

The sections below will show how to use both methods. For any of them, the first step is to find your Tenant ID:

How to find the Tenant ID

In order to find the Tenant ID you only need to login to the Azure portal (portal.azure.com) and take a look at the 'Overview' page of the service Azure Active Directory. Just click in the service as the below image shows:

Once there, you will find the Tenant ID in the box highlighted in the below image:

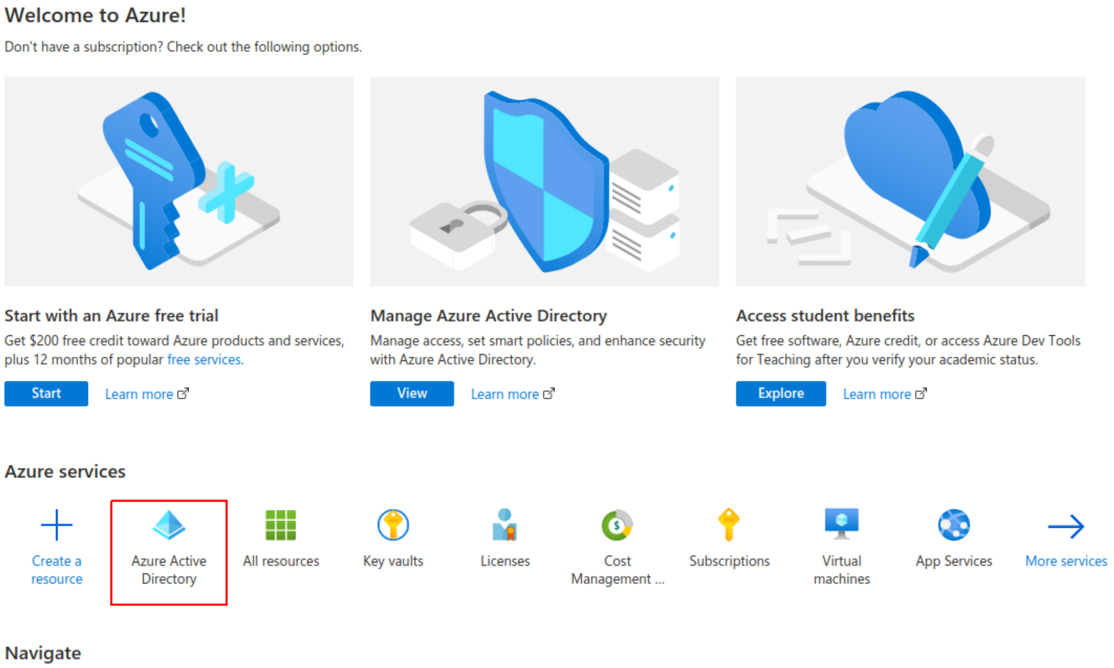
Authorization Method A: Common app model with 'bacula-m365-plugin' (DEPRECATED)

Bacula Systems has a registered application in Azure AD named **bacula-m365-plugin**. This section will show how to authorize it to perform backup and restore operations over your target tenant. All required steps to complete this authorization process are presented below.

Please, **use this connection method for testing purposes as it is deprecated**. For production systems, please go ahead with **Authorization method B**: *Authorization Method B: Standalone app model with 'bacula-m365-registrator' (RECOMMENDED)*.

Most of the procedures described in this section must be done by a **tenant administrator**. A tenant administrator is a user who has been assigned the Azure AD role **Global administrator**.

Below we show an schema of how this authorization method works:



Welcome to Azure!

Don't have a subscription? Check out the following options.

Start with an Azure free trial

Get $200 free credit toward Azure products and services, plus 12 months of popular free services.

Manage Azure Active Directory

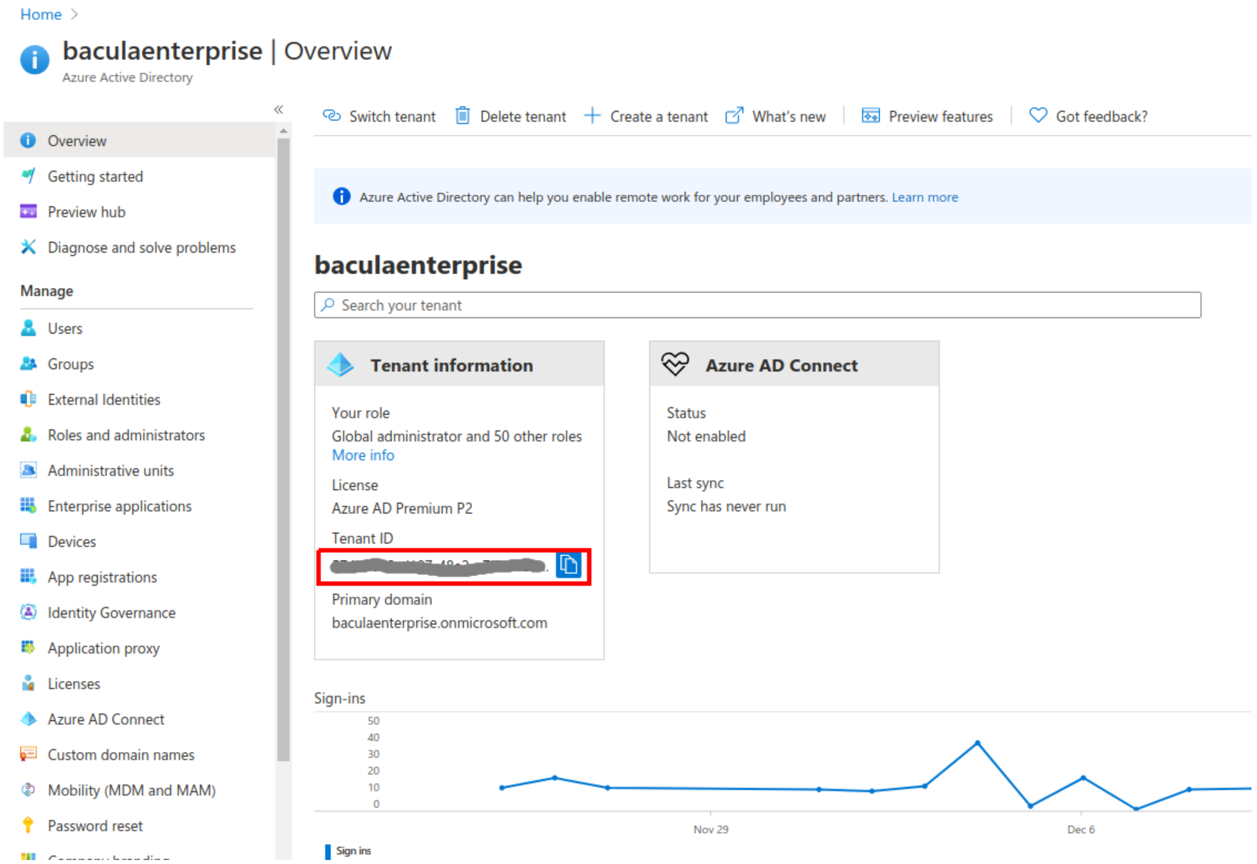
Manage access, set smart policies, and enhance security with Azure Active Directory.

Access student benefits

Get free software, Azure credit, or access Azure Dev Tools for Teaching after you verify your academic status.

Azure services

Navigate



baculaenterprise | Overview

Azure Active Directory

Azure Active Directory can help you enable remote work for your employees and partners. Learn more

baculaenterprise

Tenant information

Your role
Global administrator and 50 other roles
More info

License
Azure AD Premium P2

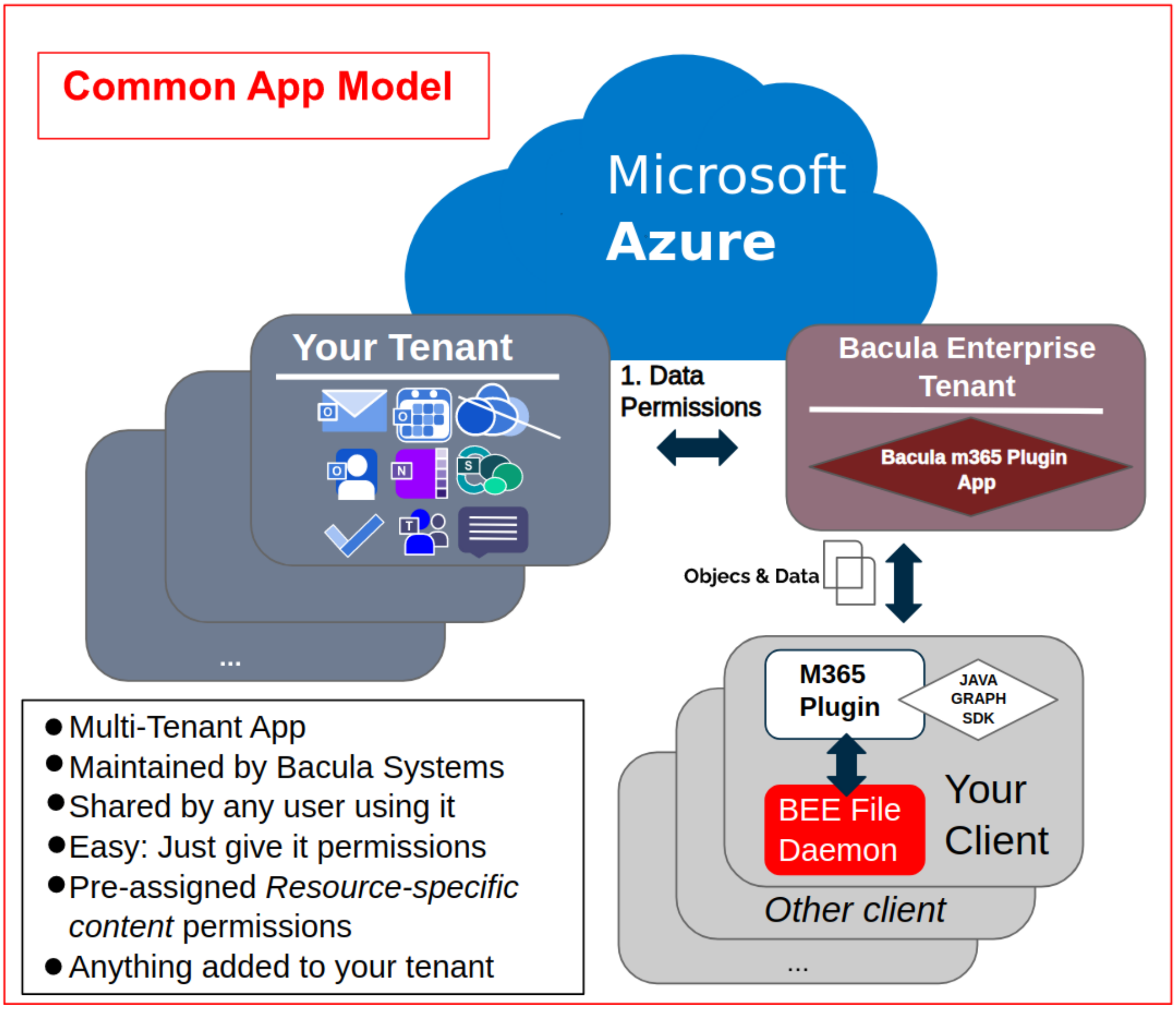
Tenant ID
[redacted]

Primary domain
baculaenterprise.onmicrosoft.com

Azure AD Connect

Status
Not enabled

Last sync
Sync has never run



- Multi-Tenant App
- Maintained by Bacula Systems
- Shared by any user using it
- Easy: Just give it permissions
- Pre-assigned *Resource-specific content* permissions
- Anything added to your tenant

A-1. Authorize bacula-m365-plugin

A **tenant administrator** must run the following query in a web browser, replacing **{tenantId}** with the value obtained in the first step:

Listing 57: **Authorization URL**

```
https://login.microsoftonline.com/{tenantId}/adminconsent?client_id=14a0b71a-d9ca-496c-
↪b4c0-76a3cbb5dc33&state=12345&redirect_uri=https://www.baculasystems.com/m365-plugin-
↪auth/common
```

You can get the exact same URL from the plugin command line itself, once you have installed it in a client named {your_client_name} using the following special Query command.

Listing 58: **Authorization URL**

```
*.query plugin="m365:" client={your_client_name} parameter=register:{tenantId}
```

Here is an execution example where the instructions are displayed and you need to open the provided URI:

Listing 59: **Query command for tenant URL**

```
*.query plugin="m365:" client=127.0.0.1-fd parameter=register:57uia43-d107-17a2-a2g2-
↪aa53c10tdahc
console=---- M365 PLUGIN REGISTER COMMAND ----
console=-------------------------------------
console=-------------------------------------
info=Open the following URI in a browser with your tenant admin credentials
console=-------------------------------------
console=-------------------------------------
uri=https://login.microsoftonline.com/57uia43-d107-17a2-a2g2-aa53c10tdahc/adminconsent?
↪client_id=14a0b71a-d9ca-496c-b4c0-76a3cbb5dc33&state=12345&redirect_uri=https://www.
↪baculasystems.com/m365-plugin-auth/common
console=-------------------------------------
console=-------------------------------------
info=Once you have accepted the provided permissions to the bacula-m365-plugin app..
info=You can get your ObjectId using the command below
console=-------------------------------------
console=-------------------------------------
command=.query plugin="m365: tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc" client={your_
↪client_name} parameter=objectid
console=-------------------------------------
```

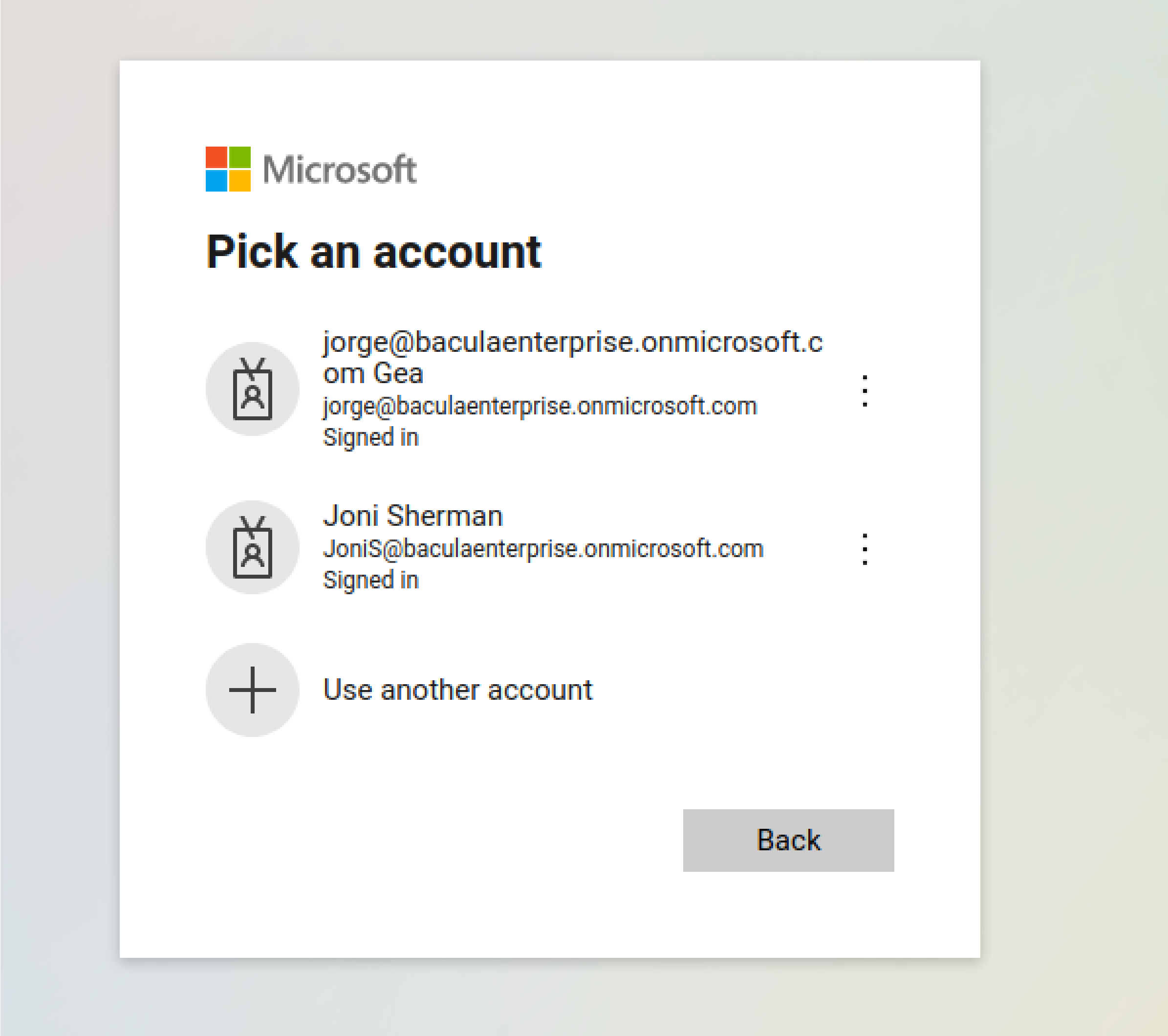
When opening the URI, Microsoft 365 will ask for credentials. You need to authenticate as a **tenant admin**:

…and once they are correctly provided, the app will ask for all the required permissions to backup and restore all of the supported elements:

The application needs **all of the listed permissions to be able to work**. If any of them is missing, backup or restore operationms will fail.

The image and list shown here are illustrative and some additional permissions may be needed as the plugin evolves. However, we provide also here a text list of current permissions needed:

- Graph
  - Read and write user chat messages → Backup/Restore of Team Channels and Chats



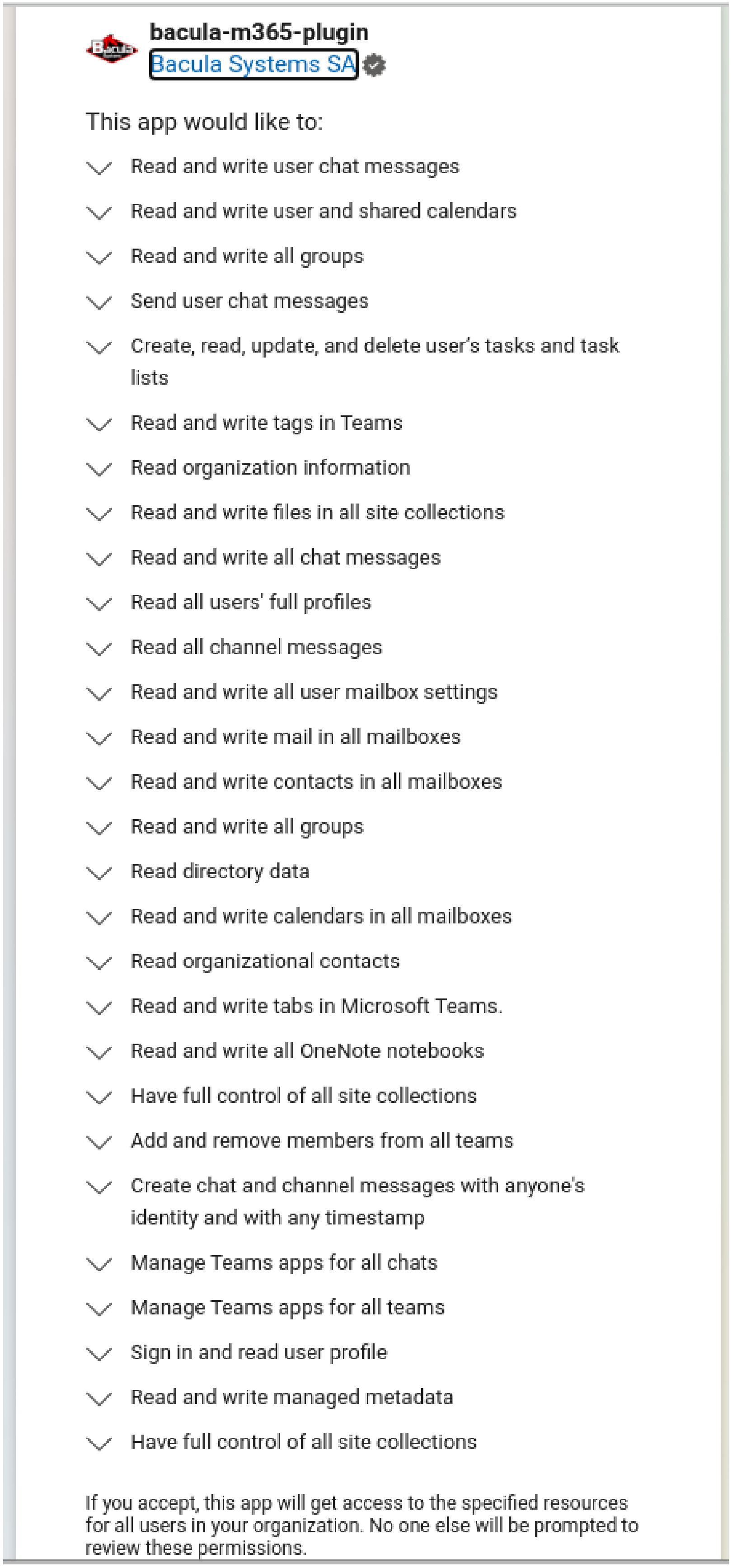
Microsoft

Pick an account

jorge@baculaenterprise.onmicrosoft.com Gea
jorge@baculaenterprise.onmicrosoft.com
Signed in

Joni Sherman
JoniS@baculaenterprise.onmicrosoft.com
Signed in

Use another account



bacula-m365-plugin

Bacula Systems SA

This app would like to:

- Read and write user chat messages
- Read and write user and shared calendars
- Read and write all groups
- Send user chat messages
- Create, read, update, and delete user’s tasks and task lists
- Read and write tags in Teams
- Read organization information
- Read and write files in all site collections
- Read and write all chat messages
- Read all users' full profiles
- Read all channel messages
- Read and write all user mailbox settings
- Read and write mail in all mailboxes
- Read and write contacts in all mailboxes
- Read and write all groups
- Read directory data
- Read and write calendars in all mailboxes
- Read organizational contacts
- Read and write tabs in Microsoft Teams.
- Read and write all OneNote notebooks
- Have full control of all site collections
- Add and remove members from all teams
- Create chat and channel messages with anyone's identity and with any timestamp
- Manage Teams apps for all chats
- Manage Teams apps for all teams
- Sign in and read user profile
- Read and write managed metadata
- Have full control of all site collections

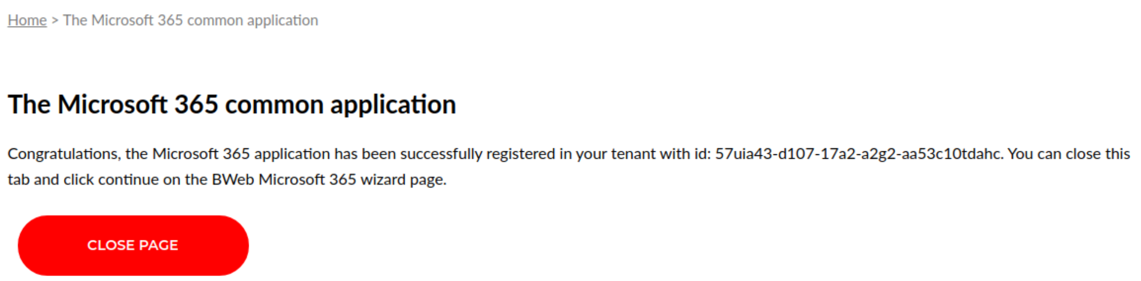
If you accept, this app will get access to the specified resources for all users in your organization. No one else will be prompted to review these permissions.

  - Read and write user and shared calendars → Backup/Restore of group Calendars
  - Read and write all groups → Creation of Teams
  - Send user chat messages → Restore of Chats
  - Create, read, update and delete user's tasks and task lists → Backup/Restore of Tasks
  - Read and write tags in Teams → For future support of Team tags
  - Read organization information → Find tenant name
  - Read and write files in all sites collections → Backup/Restore OneDrive and Sharepoint
  - Read and write all chat messages → Backup/Restore of Team Channels
  - Read all users full profiles → Find users to Backup/Restore
  - Read all channel messages → Backup of Team Channels
  - Read and write all user mailbox settings → Outlook categories and more future mailbox information
  - Read and write contacts in all mailboxes → Backup/Restore contacts
  - Read directory data → Security checks of objectid, list groups, etc
  - Read and write calendars in all mailboxes → Backup/Restore User Calendars
  - Read organizational contacts → Backup organizational contacts
  - Read and write tabs in Microsoft Teams → Backup of Microsoft Teams tabs
  - Read and write all OneNote notebooks → Backup/Restore OneNote service*
  - Have full control of all site collections → Backup/Restore Sharepoint
  - Add and remove members from all Teams → Backup/Restore of Microsoft Teams members
  - Create chat and channel messages with anyone's identity and with any timestamp
  - Manage Teams apps for all chats → Backup of Microsoft Teams apps
  - Manage Teams apps for all teams → Backup of Chats apps
  - Read and write the names, descriptions, and settings of all channels of all Teams → Backup of Microsoft Teams Channels
  - Add and remove members from all channels → Backup of Microsoft Teams Channels
  - Read and write managed metadata → Security checks of objectid, list groups, etc
  - Sign in and read user profile → Ability to connect to the tennant
  - etc.
- Sharepoint Online
  - Full management of all site collections → Backup/Restore Sharepoint (PnP)
  - Full management of Term Store → Backup/Restore Sharepoint (PnP)

Once it is confirmed, the browser will use the 'redirect_uri', which is a page on baculasystems.com that will confirm the result of the registration process.

The generated URI contains the parameter **admin_consent=True** if the action was successful, and you will see a confirmation message in that case. Otherwise, the operation may have not been successful for some reason and you will see an error message.

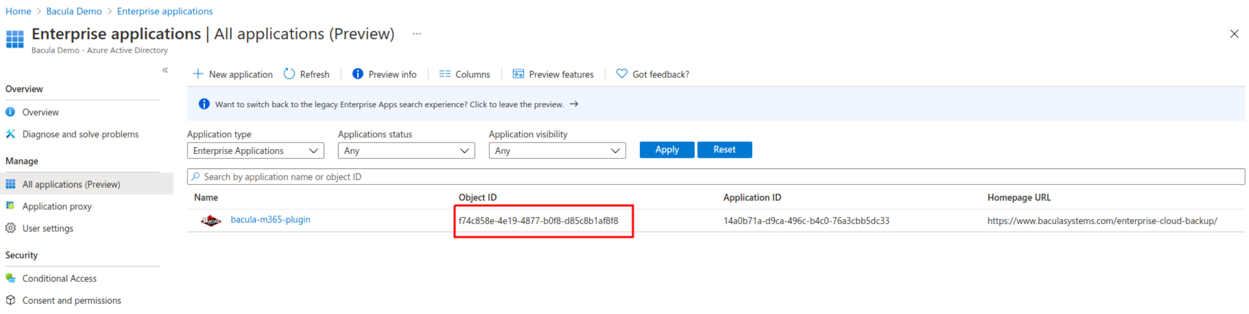
Once that action is done, the tenant where our app now has permissions will show the plugin in the Enterprise Apps section:



Home > The Microsoft 365 common application

The Microsoft 365 common application

Congratulations, the Microsoft 365 application has been successfully registered in your tenant with id: 57uia43-d107-17a2-a2g2-aa53c10tdahc. You can close this tab and click continue on the BWeb Microsoft 365 wizard page.



Home > Bacula Demo > Enterprise applications
Enterprise applications | All applications (Preview)
Bacula Demo - Azure Active Directory
Overview
Overview
Diagnose and solve problems
Manage
All applications (Preview)
Application proxy
User settings
Security
Conditional Access
Consent and permissions
New application
Refresh
Preview info
Columns
Preview features
Got feedback?
Apply
Reset

Want to switch back to the legacy Enterprise Apps search experience? Click to leave the preview. →

Application type: Enterprise Applications

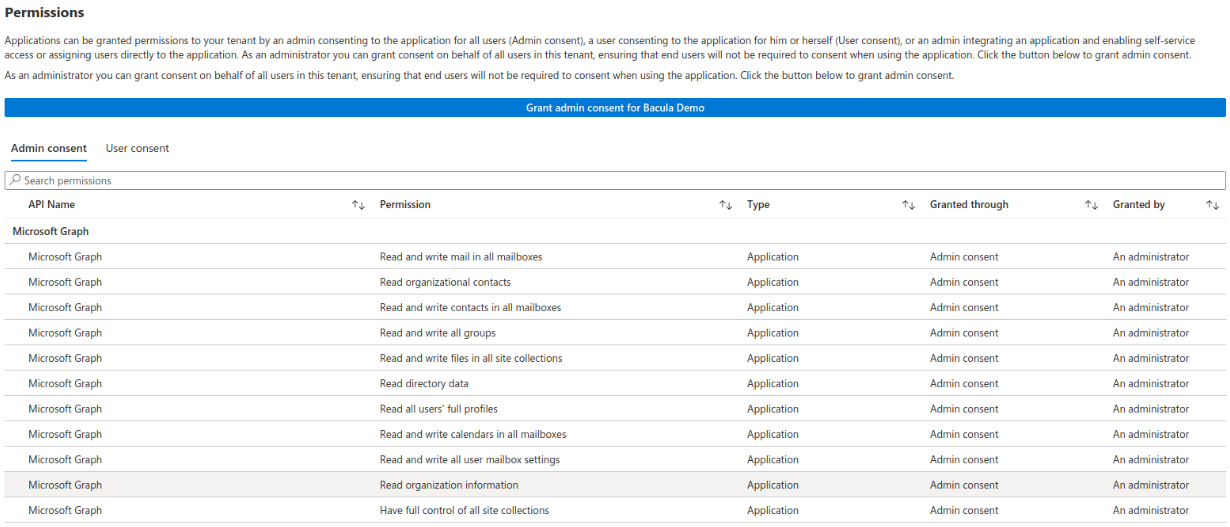
Applications status: Any

Application visibility: Any

Search by application name or object ID

Name	Object ID	Application ID	Homepage URL
bacula-m365-plugin	f74c858e-4e19-4877-b0f8-d85c8b1af8f8	14a0b71a-d9ca-496c-b4c0-76a3cbb5dc33	https://www.baculasystems.com/enterprise-cloud-backup/

Clicking on the app, the tenant admin can always see the permissions assigned:



Permissions

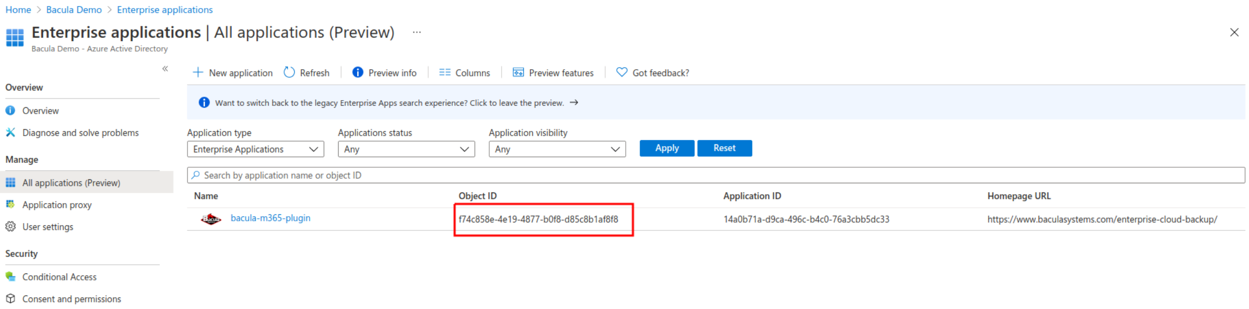
Applications can be granted permissions to your tenant by an admin consenting to the application for all users (Admin consent), a user consenting to the application for him or herself (User consent), or an admin integrating an application and enabling self-service access or assigning users directly to the application. As an administrator you can grant consent on behalf of all users in this tenant, ensuring that end users will not be required to consent when using the application. Click the button below to grant admin consent.

As an administrator you can grant consent on behalf of all users in this tenant, ensuring that end users will not be required to consent when using the application. Click the button below to grant admin consent.

API Name	Permission	Type	Granted through	Granted by
Microsoft Graph				
Microsoft Graph	Read and write mail in all mailboxes	Application	Admin consent	An administrator
Microsoft Graph	Read organizational contacts	Application	Admin consent	An administrator
Microsoft Graph	Read and write contacts in all mailboxes	Application	Admin consent	An administrator
Microsoft Graph	Read and write all groups	Application	Admin consent	An administrator
Microsoft Graph	Read and write files in all site collections	Application	Admin consent	An administrator
Microsoft Graph	Read directory data	Application	Admin consent	An administrator
Microsoft Graph	Read all users' full profiles	Application	Admin consent	An administrator
Microsoft Graph	Read and write calendars in all mailboxes	Application	Admin consent	An administrator
Microsoft Graph	Read and write all user mailbox settings	Application	Admin consent	An administrator
Microsoft Graph	Read organization information	Application	Admin consent	An administrator
Microsoft Graph	Have full control of all site collections	Application	Admin consent	An administrator

A-2. Grab Object ID

The plugin needs a final parameter that is unique to each tenant and the plugin app. This is **ObjectID**, and may be obtained from the Overview app page, once step 2 has been completed:


Home > Bacula Demo > Enterprise applications
Enterprise applications | All applications (Preview)
Bacula Demo - Azure Active Directory
New application
Refresh
Preview info
Columns
Preview features
Got feedback?
Overview
Overview
Diagnose and solve problems
Manage
All applications (Preview)
Application proxy
User settings
Security
Conditional Access
Consent and permissions
Apply
Reset
Search by application name or object ID

Want to switch back to the legacy Enterprise Apps search experience? Click to leave the preview. →

Application type: Enterprise Applications
Applications status: Any
Application visibility: Any

Name	Object ID	Application ID	Homepage URL
bacula-m365-plugin	f74c858e-4e19-4877-b0f8-d85c8b1af8f8	14a0b71a-d9ca-496c-b4c0-76a3cbb5dc33	https://www.baculasystems.com/enterprise-cloud-backup/

The plugin can also obtain it from the command line using another special Query command. You can see that this exact command is also suggested in the command that shows the register URL:

Listing 60: **Authorization URL**

```
*.query plugin="m365: tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc" client={your_client_name} parameter=objectid
```

Here is an execution example:

Listing 61: **Authorization URL**

```
*.query plugin="m365: tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc" client=127.0.0.1-fd parameter=objectid
console=---- M365 PLUGIN OBJECTID COMMAND ----
console=-------------------------------------
objectid=56ddf1h9-eb5d-42nf-bac7-7b019fd284g5
console=-------------------------------------
```

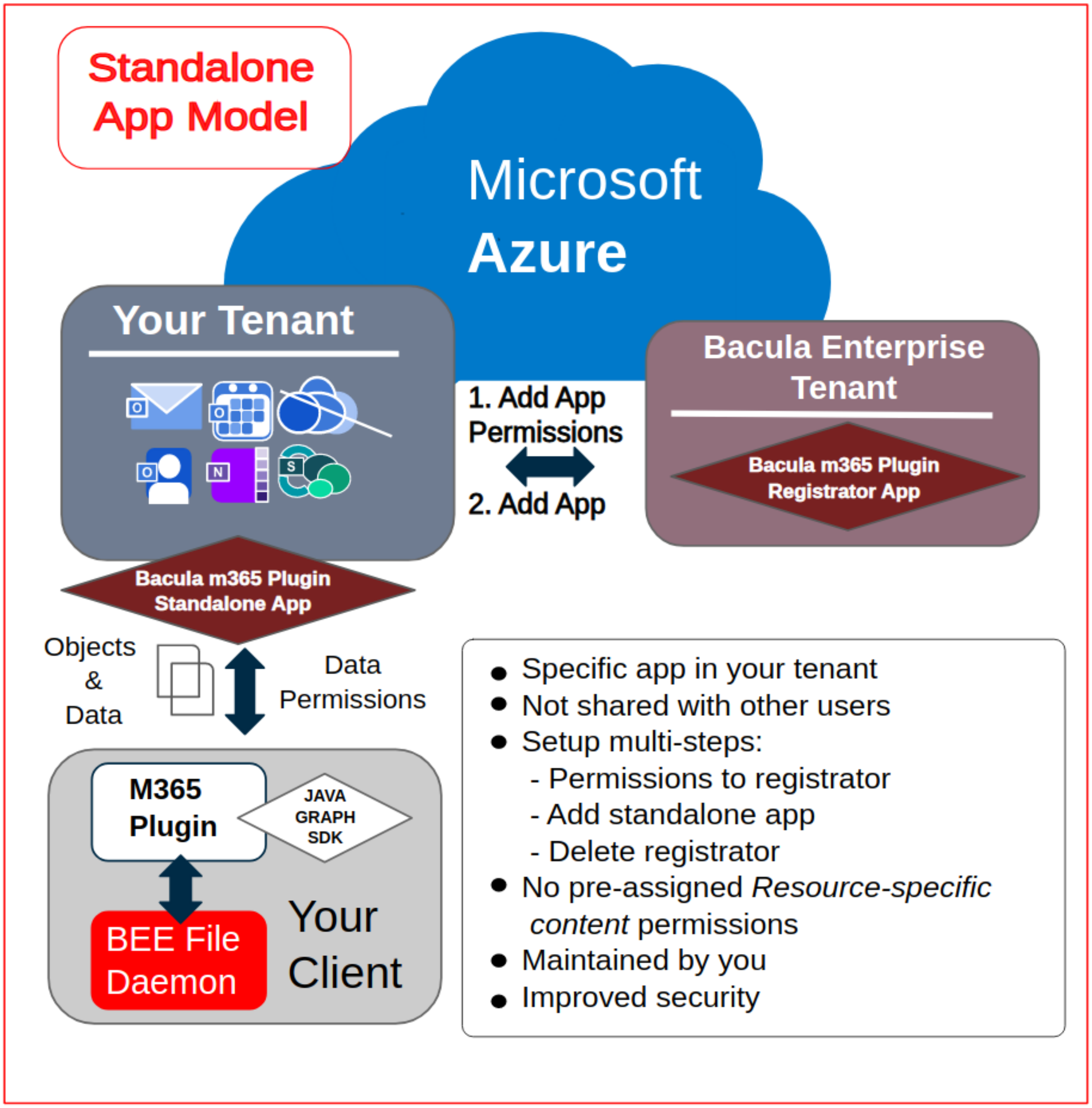
Authorization Method B: Standalone app model with 'bacula-m365-registrator' (RECOMMENDED)

Bacula Systems has a registered application in Azure AD named **bacula-m365-registrator**. This section will show how to authorize it to perform application management operations over your target tenant.

Once you authorize it, you will be able to add a new **bacula-m365-plugin-standalone** specifically for your tenant that you can directly control and that will be used only in your environment.

This is the **recommended authorization method** for any production environment.

Below we show an schema of how this authorization method works:



- Specific app in your tenant
- Not shared with other users
- Setup multi-steps:
  - Permissions to registrator
  - Add standalone app
  - Delete registrator
- No pre-assigned *Resource-specific content* permissions
- Maintained by you
- Improved security

B-1. Authorize bacula-m365-registrator

A **tenant admin** must run the following query in a web browser, replacing **{tenantId}** with the value obtained in the first step:

Listing 62: **Authorization URL**

```
https://login.microsoftonline.com/{your_tenant_id}/adminconsent?client_id=8ed4fad3-3f63-
↪44fb-b7e0-e324895d7df9&state=12345&redirect_uri=https://www.baculasystems.com/m365-
↪plugin-auth/standalone
```

You can get the exact same URL from the plugin command line itself, once you have installed it in a client named {your_client_name} using the following special Query command.

Listing 63: **Authorization URL**

```
*.query plugin="m365:" client={your_client_name} parameter=register-registrator:
↪{tenantId}
```

Here is an execution example where the instructions are displayed and you need to open the provided URI:

Listing 64: **Query command for tenant URL**

```
*.query client=127.0.0.1-fd plugin="m365:" parameter=register-registrator:57uia43-d107-
↪17a2-a2g2-aa53c10tdahc
console=---- M365 PLUGIN REGISTER COMMAND ----
console=-------------------------------------
console=-------------------------------------
info=Open the following URI in a browser with your tenant admin credentials
console=-------------------------------------
console=-------------------------------------
uri=https://login.microsoftonline.com/57uia43-d107-17a2-a2g2-aa53c10tdahc/adminconsent?
↪client_id=8ed4fad3-3f63-44fb-b7e0-e324895d7df9&state=12345&redirect_uri=https://www.
↪baculasystems.com/m365-plugin-auth/standalone
console=-------------------------------------
```

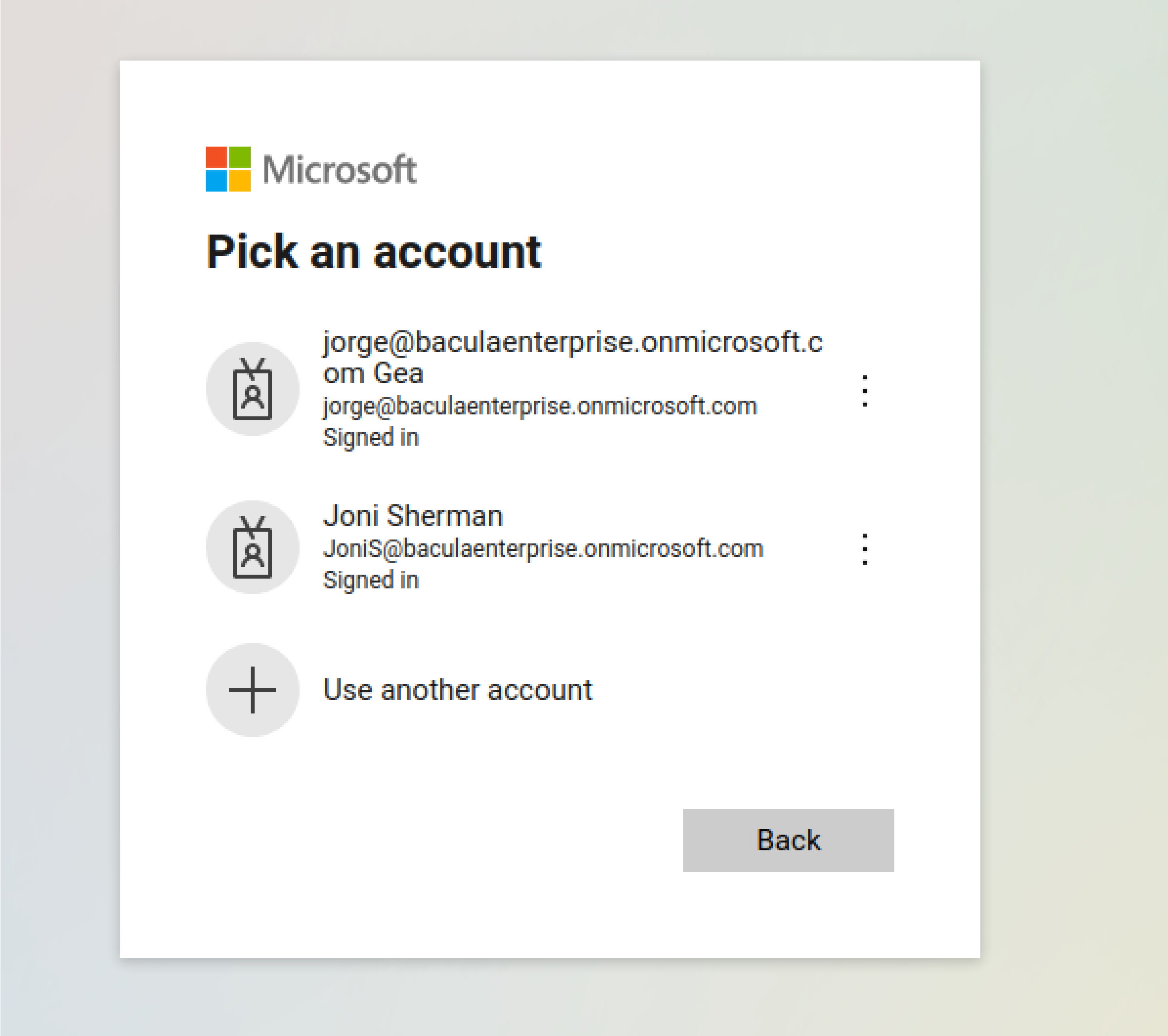
When opening the URI, Microsoft 365 will ask for credentials. You need to authenticate as a **tenant admin**:

…and once they are correctly provided, the app will ask for all the required permissions to backup and restore all of the supported elements:

The application needs **all of the listed permissions to be able to work**. If any of them is missing, the final command to add the application will fail.

The image shown here is illustrative and some additional permission may be needed as the plugin evolves. To facilitate that information, we are listing the permissions in text alongside the reason for their need.

- Graph - Sign in and read user profile: Connect to the tenant - Read and write all groups: Necessary to manage some permission grants (delegated form) - Read directory data: Find your tenant information, check service principals (delegated form) - Manage all delegated permission grants: Generate delegated permissions grants for the end app - Read and write all applications: Create/Update the new application (client credentials) - Read organization information: Find your tenant information and show user-friendly progress - Read and write all applications: Create/Update the new application (delegated form) - Read and write all groups: Necessary to manage some permission grants (client credentials) - Read directory data: Find your tenant information, check service principals (client credentials) - Manage app permission grants and app role assignments: Add necessary plugin permissions



Microsoft

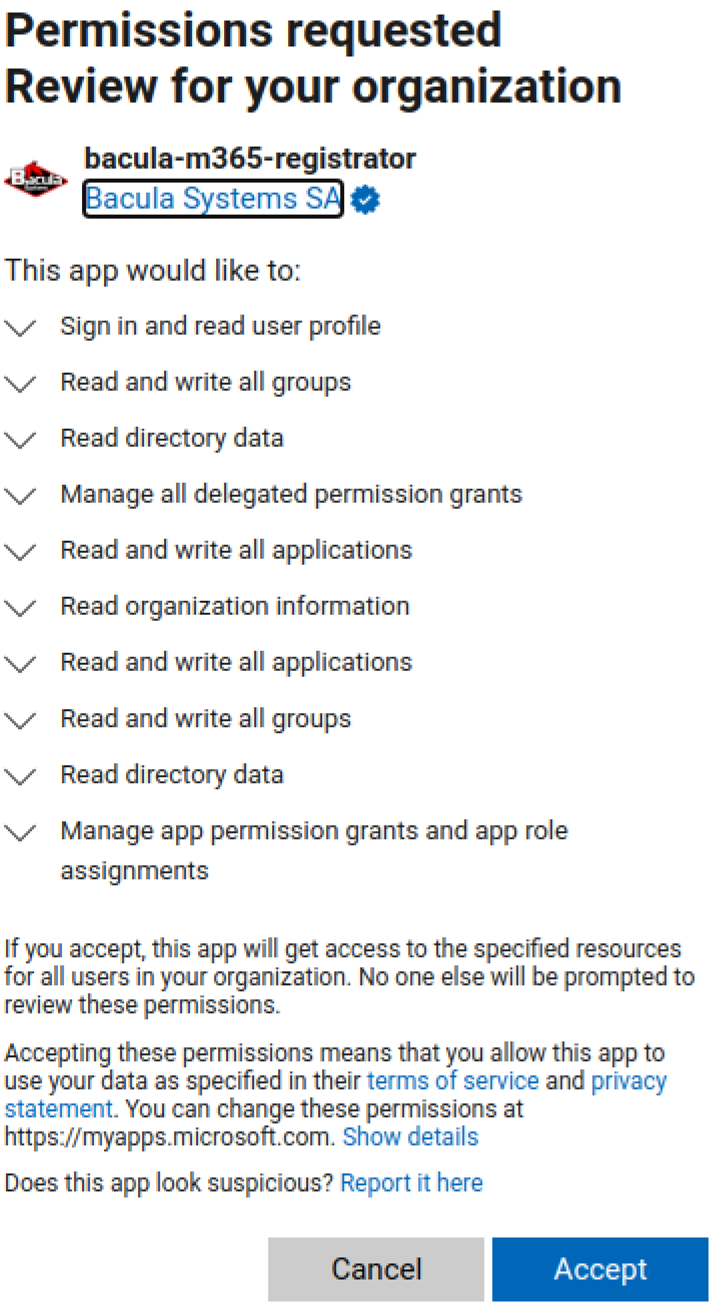
Pick an account

jorge@baculaenterprise.onmicrosoft.c
om Gea
jorge@baculaenterprise.onmicrosoft.com
Signed in

Joni Sherman
JoniS@baculaenterprise.onmicrosoft.com
Signed in

Use another account

Back



Permissions requested
Review for your organization

bacula-m365-registrator
Bacula Systems SA

This app would like to:

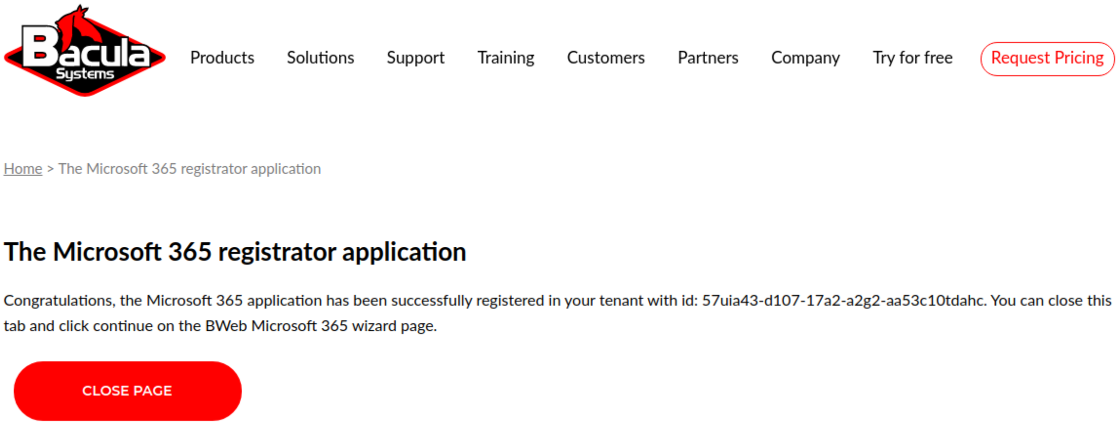
- Sign in and read user profile
- Read and write all groups
- Read directory data
- Manage all delegated permission grants
- Read and write all applications
- Read organization information
- Read and write all applications
- Read and write all groups
- Read directory data
- Manage app permission grants and app role assignments

If you accept, this app will get access to the specified resources for all users in your organization. No one else will be prompted to review these permissions.

Accepting these permissions means that you allow this app to use your data as specified in their terms of service and privacy statement. You can change these permissions at https://myapps.microsoft.com. Show details

Does this app look suspicious? Report it here

Once it is confirmed, the browser will use the 'redirect_uri', which is a page on baculasystems.com that will confirm the result of the registration process.



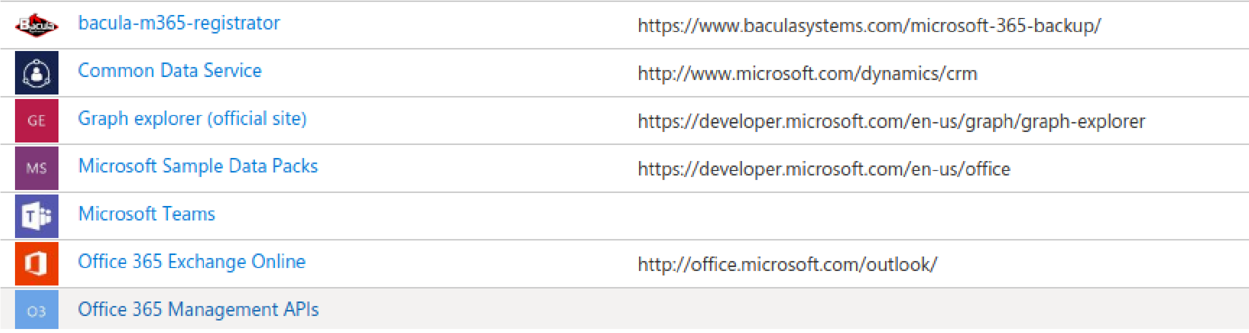
Home > The Microsoft 365 registrator application

The Microsoft 365 registrator application

Congratulations, the Microsoft 365 application has been successfully registered in your tenant with id: 57uia43-d107-17a2-a2g2-aa53c10tdahc. You can close this tab and click continue on the BWeb Microsoft 365 wizard page.

The generated URI contains the parameter **admin_consent=True** if the action was successful, so you will see a confirmation message in that case. Otherwise, the operation may have not been successful for some reason and you will see an error message.

Once that action is done, the tenant where our app now has permissions will show the plugin in the Enterprise Apps section:



	bacula-m365-registrator	https://www.baculasystems.com/microsoft-365-backup/
	Common Data Service	http://www.microsoft.com/dynamics/crm
GE	Graph explorer (official site)	https://developer.microsoft.com/en-us/graph/graph-explorer
MS	Microsoft Sample Data Packs	https://developer.microsoft.com/en-us/office
	Microsoft Teams	
	Office 365 Exchange Online	http://office.microsoft.com/outlook/
O3	Office 365 Management APIs	

Clicking on the app, the tenant admin can always see the permissions assigned:

B-2. Add bacula-m365-plugin-standalone to your tenant

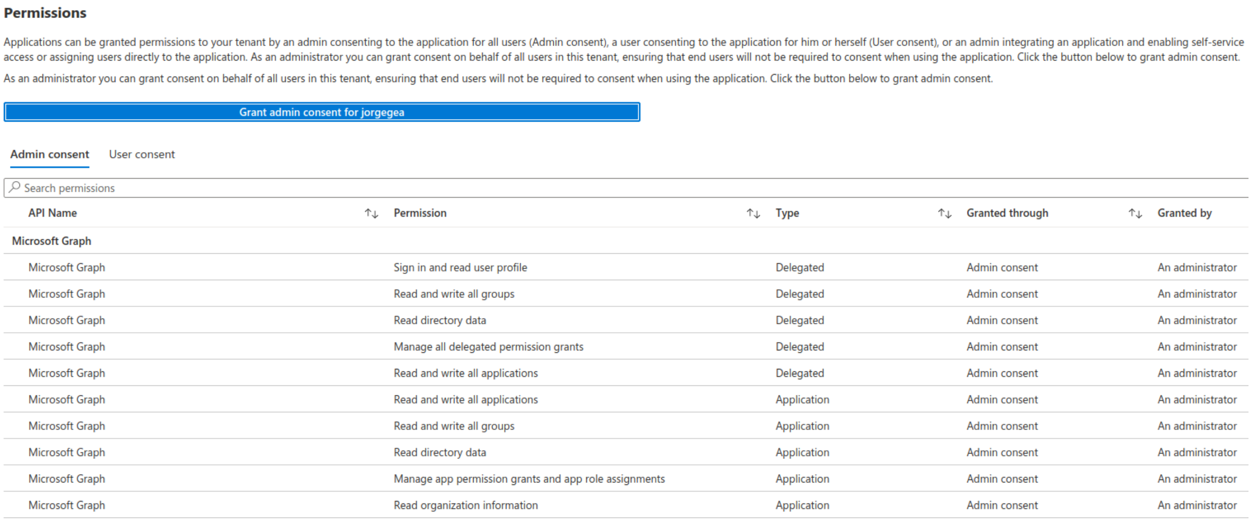
Once the registrator app is successfully registered into the target tenant, it is possible to add the bacula-m365-plugin-standalone application as a 'registered app' using the following bconsole command:

Listing 65: **Add application command**

```
*.query plugin="m365: tenant={tenantId}" client={your_client_name} parameter=add-app
```

This command needs to run operations with 'client credential' permissions, but also with 'delegated permissions' where an admin user must be implied. Therefore, at some point during its execution it will ask for the user interaction in order to complete the authentication process using an special URL and a given code. Once introduced, we will be asked about confirming the sign-in of bacula-m365-registrator:

The process must be performed by a tenant administrator. On the other hand, Microsoft Azure service needs some time in order to present newly created objects. Therefore, the process will be paused a couple of times for about one minute.

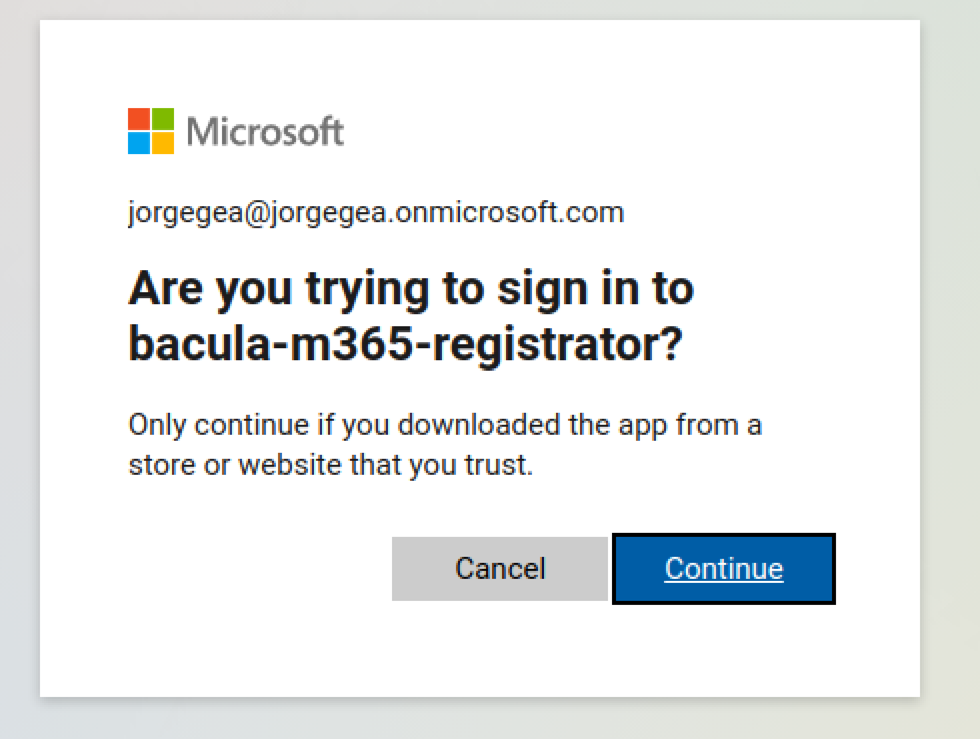


Permissions

Applications can be granted permissions to your tenant by an admin consenting to the application for all users (Admin consent), a user consenting to the application for him or herself (User consent), or an admin integrating an application and enabling self-service access or assigning users directly to the application. As an administrator you can grant consent on behalf of all users in this tenant, ensuring that end users will not be required to consent when using the application. Click the button below to grant admin consent.

As an administrator you can grant consent on behalf of all users in this tenant, ensuring that end users will not be required to consent when using the application. Click the button below to grant admin consent.

API Name	Permission	Type	Granted through	Granted by
Microsoft Graph				
Microsoft Graph	Sign in and read user profile	Delegated	Admin consent	An administrator
Microsoft Graph	Read and write all groups	Delegated	Admin consent	An administrator
Microsoft Graph	Read directory data	Delegated	Admin consent	An administrator
Microsoft Graph	Manage all delegated permission grants	Delegated	Admin consent	An administrator
Microsoft Graph	Read and write all applications	Delegated	Admin consent	An administrator
Microsoft Graph	Read and write all applications	Application	Admin consent	An administrator
Microsoft Graph	Read and write all groups	Application	Admin consent	An administrator
Microsoft Graph	Read directory data	Application	Admin consent	An administrator
Microsoft Graph	Manage app permission grants and app role assignments	Application	Admin consent	An administrator
Microsoft Graph	Read organization information	Application	Admin consent	An administrator



jorgegea@jorgegea.onmicrosoft.com

Are you trying to sign in to bacula-m365-registrator?

Only continue if you downloaded the app from a store or website that you trust.

You need to wait **2-3 minutes** (minimum!) after the first registration command has been completed. If it is executed too soon it could fail, as the registrator may not be completely available yet. If this happens, please, just run the command again.

Please find an example for this procedure below:

Listing 66: **Add application example**

```
*.query client=127.0.0.1-fd plugin="m365: tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc"␣
↪parameter=add-app
console=---- M365 ADD APP ----
info=Creating destination config file in: /opt/bacula/working/m365/baculaenterprise/
↪bacula_m365_config.conf
info=Creating application: bacula-m365-plugin-standalone in tenant: baculaenterprise ...
info=Waiting a bit so MS Azure cache has enough time to show our new Application...
info=Adding certificate key credential ...
info=Adding password credential ...
info=Application successfully created
info=We don't have a valid token. Please:
info=To sign in: use a web browser to open the page https://microsoft.com/devicelogin␣
↪and enter the code CG2QYSK8Y to authenticate. You need to do it with an administrator␣
↪user
info=Adding owner: johndoe@johndoe.onmicrosoft.com to the application ...
info=Owner successfully added
info=Creating service principal for application ...
info=Service principal successfully created
info=Waiting a bit so MS Azure cache has enough time to show our new ServicePrincipal...
info=Granting admin consent for permissions ...
info=Admin consent granted
info=Storing access configuration to the new app in file: /tmp/regress/working/m365/
↪johndoe/bacula_m365_config.conf
info=Access configuration successfully stored
console=---- ADD APP COMPLETED SUCCESSFULLY ----
console=-------------------------------------
info=Use the generated configuration file in your m365 filesets. Just add the property␣
↪config_file=/your/path/bacula_m365_config.conf to the line Plugin="m365: ..."
info=Sample: Plugin = "m365: config_file=/opt/bacula/etc/m365/mytenant/bacula_m365_
↪config.conf service=drive"
console=-------------------------------------
console=-------------------------------------
```

In case the add-app command fails for any reason, please, wait a bit and just run it again.

B-3. Utilize your generated config file

The result of the add-app command is the generation of a configuration file that is intended to be referenced in any future fileset, as the output of the command is indicating.

We recomend to copy that file to a convenience common location (for instance /opt/bacula/etc/filesets) and set the same diretive in any M365 fileset:

Listing 67: **Fileset example**

```
FileSet {
     Name = FS_M365
     Include {
             Options {
                     signature = MD5
                     compression = LZO
             }
             Plugin = "m365: config_file=/opt/bacula/etc/m365/mytenant/bacula_m365_
↪config.conf service=drive"
     }
}
```

The contents of the configuration file are described below:

Listing 68: **Autogenerated config file contents**

```
#Thu Jul 01 12:30:34 CEST 2021                -> Generation Date
secret=xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx        -> Password secret auto-generated
appid=1587ba33-8eer-4512-bgb1-6d65fd393f51    -> Azure AD AppID of your bacula-m365-
↪plugin-standalone
objectid=8jki8705-010p-4ee6-b2aa-c3klmn1b5b9 -> Service principal ID of your Application
tenant=e4wsdf55-de8a-4a83-945a-bfaabb6fa166  -> Your tenant id
```

Please, note that once this application is generated in your tenant, you have full control of it. As a result, you can manually generate a new secret and set the value yourself at your convenience whenever this is needed.

B-4. Optional: Renew your secrets

The add-app command discussed in section B-3 is prepared to automatically regenerate a new password secret if it detects that the current ones are expired or close to expire (6 months). Therefore, you could simply call it again in case you need to renew the secret key which has a maximum life span of 24 months in Azure. It would be possible to do this automatically using a properly configured Admin Job.

B-5. Optional: Remove bacula-m365-registrator app

The registrator application has elevated permissions, as without them it would not be possible to manage applications and assoicated permissions. It is handy to have it if you plan to run automatically the add-app command at some point, as suggested in B-4.

However, if you want to remove it from your tenant after completing the add-app process you could do it manually or you could call the following bconsole command:

Listing 69: **Delete registrator**

```
.query client={your-client} plugin="m365: tenant={yourTenantId}" parameter=delete-
↪registrator
```

Please, be aware that once you do that, you won't be able to run the add-app command. If you need to run it again after deletion, you will need to call again the register-registrator command.

Check permissions

It is possible to check if the service principal of the target tenant (this is shown under 'enterprise applications' in Azure AD) has all the needed permissions to perform plugin operations.

You just need to run the following command:

Listing 70: **Check permissions**

```
.query client={your-client} plugin="m365: tenant={yourTenantId} objectid={yourobjectid}"␣
↪parameter=permissions
```

The ouput will report if everything is ok around permissions, or if there is any missing one:

Listing 71: **Check permissions: ok output**

```
.query client={your-client} plugin="m365: tenant={yourTenantId} objectid={yourobjectid}"␣
↪parameter=permissions
info=All permissions are correctly set
```

Listing 72: **Check permissions: some is missing**

```
.query client={your-client} plugin="m365: tenant={yourTenantId} objectid={yourobjectid}"␣
↪parameter=permissions
error=Delegated permission Chat.ReadWrite was not found in your Azure service principal␣
↪8cee7605-0d5e-42b6-b269-c39f3411b5b9
error=Please review your app permissions
```

In case you are using the standalone application, you will need to run the command with appid and secret too:

Listing 73: **Check permissions**

```
.query client={your-client} plugin="m365: tenant={yourTenantId} objectid={yourobjectid}␣
↪appid={yourappid} secret={yoursecret}" parameter=permissions
```

Please, run this command at any time you have doubts about the correct app permissions.

Please note though, that the command is unable to check if you have activated or not 'Protected APIs' for Teams and Chats. It will check the presence of those permissions, but it is up to Microsoft to enable those particular ones and they do not show (at the time of writing) any flag indicating if they are enabled or not.

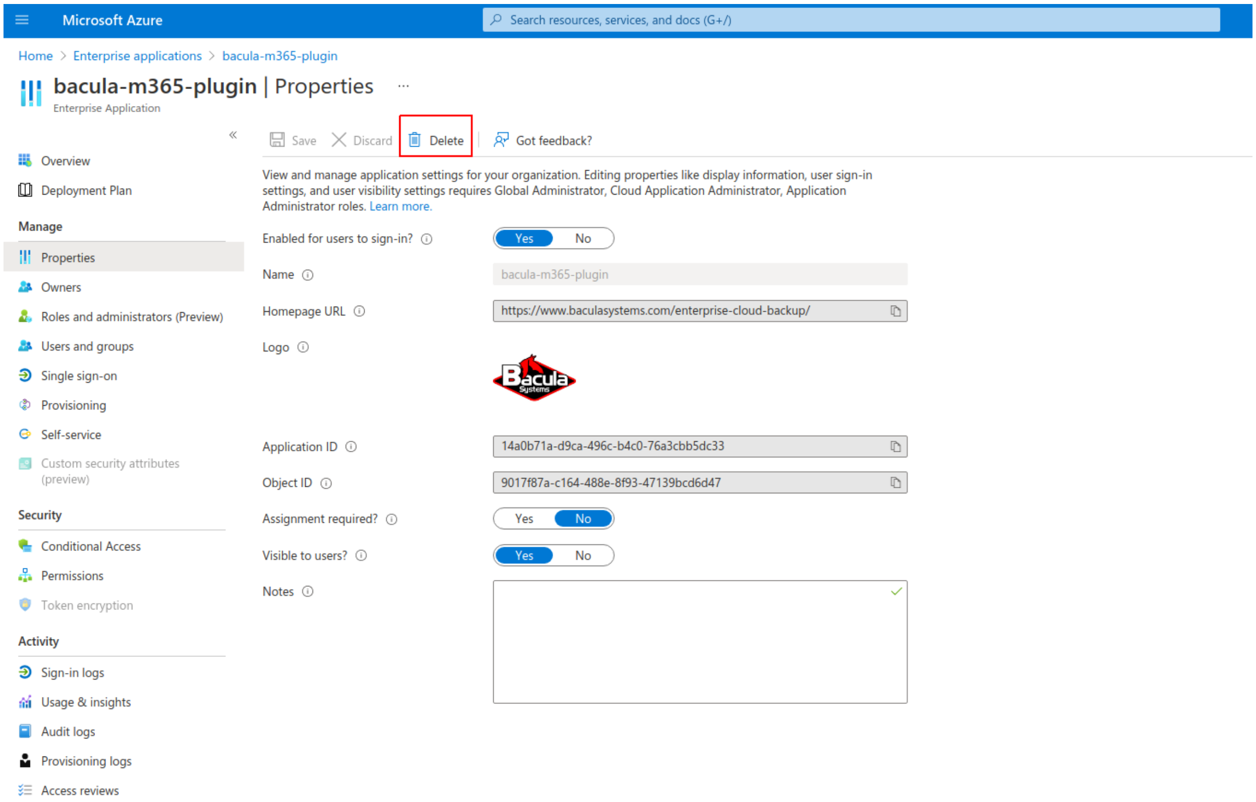
Migration from Authenticaton Method A to Authentication Method B

Perhaps you started to use this plugin using Method A and now you want to switch to Method B in order to enjoy its advantadges. This section will show the steps needed in order to complete such migration.

Migration - Step 1: Remove bacula-m365-plugin from your tenant

The first step is to remove the common app 'bacula-m365-plugin' from your tenant. In order to do it, you need to go to your Azure portal and open the 'Azure Active Directory' application as usual.

From there, please go to 'Enterprise applications' and locate bacula-m365-plugin app. You need to select the application and enter into 'Properties' section. From there, you need to click on the 'Delete' button:



View and manage application settings for your organization. Editing properties like display information, user sign-in settings, and user visibility settings requires Global Administrator, Cloud Application Administrator, Application Administrator roles. Learn more.

Enabled for users to sign-in?	[✓ Yes] No
Name	bacula-m365-plugin
Homepage URL	https://www.baculasystems.com/enterprise-cloud-backup/
Logo	Bacula Systems
Application ID	14a0b71a-d9ca-496c-b4c0-76a3cbb5dc33
Object ID	9017f87a-c164-488e-8f93-47139bcd6d47
Assignment required?	Yes [✓ No]
Visible to users?	[✓ Yes] No
Notes	

Migration - Step 2: Perform Authentication Method B

Once you have removed the common application, you simply need to follow all the steps explained in this document for authentication method B. In case you are using BWeb and you have already registered you tenant, it is recommended to remove it first, and then add it also using method B. Otherwise, for manual configuration, simply follow the steps mentioned in this document: *Authorization Method B: Standalone app model with 'bacula-m365-registrator' (RECOMMENDED)*.

Migration - Step 3: Update your filesets

Once you have your new app in your tenant, you need to update all your filesets and: - Replace objectid with the new value associated to your app in your tenant - Add appid and secret associated to your app in your tenant

7.2 Fileset Configuration

Once the plugin is successfully authorized, it is possible to define regular filesets for backup jobs in Bacula, where we need to include a line similar to the one below, in order to call the M365 Plugin:

Listing 74: **Fileset M365**

```
FileSet {
   Name = FS_M365
   Include {
      Options {
        signature = MD5
        ...
      }
      Plugin = "m365: <m365-parameter-1>=<m365-value-1> <m365-parameter-2>=<m365-value-2>
↪  ..."
   }
}
```

It is **strongly recommended** to use only one ‘Plugin’ line in every fileset. The plugin offers the needed flexibility to combine different modules backup inside the same plugin line. Different tenants, in case of existing, should be using different filesets and different jobs.

Below sub-sections list all the parameters you can use to control M365 Plugin behavior.

In this plugin, any parameter allowing a list of values can be assigned with a list of values separated by ‘,’.

Note: Starting with version 14.0 of Bacula Enterprise you can also perform fileset configuration directly using a dedicated BWeb M365 Wizard. Please go to section *BWeb Management Console* to see more details.

Common parameters

These parameters are common and applicable to all the modules of the M365 Plugin.

Option	Required	Default	Values	Example	Description
abort_on_error	No	No	No, Yes	Yes	If set to **Yes**: Abort job as soon as any error is found with any element. If set to **No**: Jobs can continue even if it they found a problem with some elements. They will try to backup or restore the other and only show a warning
config_file	No		The path pointing to a file containing any combination of plugin parameters	/opt/bacula/m365settings	Allows to define a config file where configure any parameter of the plugin. Therefore you don't need to put them directly in the Plugin line of the fileset
log	No	/opt/bacula/working/m365/m365-debug.log	An existing path with enough permissions for File Daemon to create a file with the provided name	/tmp/m365.log	Generates additional log in addition to what is shown in job log. This parameter is included in the backend file, so, in general, by default the log is going to be stored in the working directory.
debug	No	0	0, 1, 2, 3, 4, 5, 6, 7, 8, 9	Debug level. Greater values generate more debug information	Generates the working/m365/m365-debug.log* files containing debut information which is more complete with a greater debug number
path	No	/opt/bacula/working	An existing path with enough permissions for File Daemon to create any internal plugin file	/mnt/my-vol/	Uses this path to store metadata and temporary files
tenant	**Yes**		A valid tenant id string	57uia43-d107-17a2-a2g2-aa53c10tdahc	The **tenant ID** where the plugin will connect to in order to run backups or restores. Please, check section 5.1 of this paper for more information
objectid	**Yes**		String representing the objectid related to bacula-m365-plugin Azure app and the provided tenant id	56ddf1h9-eb5d-42nf-bac7-7b019fd284g5	The **object ID** of the plugin app of the plugin in Azure, once this is registered in the target tenant. Please, check section 5.1 of this paper for more information
appid	No		String representing the appid associated to bacula-m365-plugin-standalone Azure app registered in the configured tenant id	89tt4hu7-r4h7-kied-56gu-0895kf94jr9d	A valid appid string associated to bacula-m365-plugin-standalone Azure app registered in the configured tenant id. Please, check section 5.1 of this paper for more information
secret	No		String representing the associated appid secret	Jn8.lU-B.3P5gIRTGY6MlXl3t29oQ6Xuf-	The secret associated to the m365 Azure app corresponding to the configured appid string. Please, check section 5.1 of this paper for more information
token_cache_file	No	token_cache.json	A file name located in a valid existing path (it's possible to put only the name of the file and 'working' dir will be used)	my_cache_file.json	The path that will be used to store the login cache for the device code flow authenticated users, wich is relative to the working tenant folder (working/tenant-name/token_cache.json)
service	No		email, drive, sharepoint, contact, calendar, onenote, tasks, teams, chat (list pa-	drive	Establish the service or services that will be backed up. If this is not set, the plugin will try to backup all supported services. It is recommended to split the work among different jobs when several services

Note: The path option is adjustable while M365 jobs are in progress. The running jobs will continue to use the current spool area, and new jobs will use the new spool area defined by the path option.

If specifying more than one service in the fileset, please note that they will run in parallel. Because of that, we recommend to decrease a little the concurrency.

Below some multi-services fileset examples:

Listing 75: **Fileset for all data belonging to a user**

```
FileSet {
   Name = fs-m365-adelev-user
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=drive,email,calendar,contact,onenote,tasks,chat␣
↪tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc objectid=56ddf1h9-eb5d-
   42nf-bac7-7b019fd284g5 user=adelev@baculaenterprise.onmicrosoft.com concurrent_
↪threads=2"
   }
}
```

Listing 76: **Fileset for all data belonging to a group**

```
FileSet {
   Name = fs-m365-devteam-group
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=drive,calendar,onenote,tasks,teams tenant=57uia43-d107-
↪17a2-a2g2-aa53c10tdahc objectid=56ddf1h9-eb5d-
   42nf-bac7-7b019fd284g5 group=DevTeam concurrent_threads=2"
   }
}
```

Listing 77: **Fileset for all data belonging to a site**

```
FileSet {
   Name = fs-m365-mysite-site
   Include {
      Options { signature = MD5 }
      Plugin = "m365: service=sharepoint,drive,onenote tenant=57uia43-d107-17a2-a2g2-
↪aa53c10tdahc objectid=56ddf1h9-eb5d-
   42nf-bac7-7b019fd284g5 site=MySite sharepoint_include_drive_items=no sharepoint_
↪system_include=yes drive_system_include=yes concurrent_threads=2"
   }
}
```

Note: Activating proxy mody will route all the requests through the proxy. However, it is needed DNS resolution to be working separately. Hence, the client where the FD is running will need to have a proper and working DNS Server configured. Other option is to setup the /etc/hosts file manually with the IP addresses of Microsoft authentication servers, but they could change over time. At the time of writing they are: 40.126.31.2 login.microsoftonline.com 20.190.160.131 login.microsoft.com

Advanced common parameters

Following parameters are common to all M365 modules (and even with some other plugins), but are advanced ones. They should not be modified in most common use cases.

Option	Required	Default	Values	Example	Description
stream_sleep	No	1	Positive integer (1/10 secconds)	5	Time to sleep when reading header packets from FD and not having a full header available
stream_max_wait	No	120	Positive integer (seconds)	360	Max wait time for FD to answer packet requests
time_max_last_modify_log	No	86400	Positive integer (seconds)	43200	Maximum time to wait to ovewrite a debug log that was marked as being used by other process
logging_max_file_size	No	50MB	String size	300MB	Maximum size of a single debug log fileGenerates the working/m365/m365-debug.log* files containing debut information which is more complete with a greater debug number
logging_max_backup_index	No	25	Positive integer (number of files)	50	Maximum number of log files to keep
log_rolling_file_pattern	No	m365.log.%d{dd-MMM}.log.gz”	No, Yes	Yes	Log patter for rotated log files
split_config_file	No	=	Character	:	Character to be used in config_file parameter as separator for keys and values
opener_queue_timeout_secs	No	1200	Positive integer (seconds)	3600	Timeout when internal object opener queue is full
publisher_queue_timeout_secs	No	1200	Positive integer (seconds)	3600	Timeout when internal object publisher queue is full

The internal plugin logging framework presents some relevant features that we are going to describe:

- The “.log” files are rotated automatically. Currently each file can be 50Mb at maximum and the plugin will keep 25 files.
  - This behavior can be changed using the internal advanced parameters: logging_max_file_size and logging_max_backup_index
- The “.err” file can show contents even if no real error happened in the jobs. It can show contents too even if debug is disabled. This file is not rotated, but it is expected to be a small file in general. If you still need to rotate it, you can include it in a general rotating tool like ‘logrotate’.
- Backups in paralel and also failed backups will generate several log files. For example: m365-debug-0.log, m365-debug-1.log…

Tuning parameters

These set of parameters are common to all modules and they are advanced ones. They should not be modified in general. They can be used to tune the behavior of the plugin to be more flexible in particular bad network environments or when significant job concurrency is happening, etc.

Option	Required	Default	Values	Example	Description
backup_queue_size	No	100	0-500	1	Number of maximum enqueued internal operations between service static internal threads (there are 3 communicating through queues with the set size: service fetcher, service opener and general publisher to bacula core). This could potentially affect graph api concurrent requests and consequently, Graph throttling. It is only needed to modify this parameter, in general, if you are going to run different jobs in parallel
concurrent_threads	No	10	0-100	1	Number of maximum concurrent backup threads running in parallel in order to fetch or open data for running download actions. This means every service fetcher and service opener will open this number of child concurrent threads. This will affect graph api concurrent requests. Graph API can throttle requests depending on a variety of circumstances, but this parameter impacts it directly. It is only needed to modify this parameter, in general, if you are going to run different jobs in parallel. If you want to have a precise control of your parallelization through different jobs, please set up this value to 1. Please be careful also with the memory requirements, multi-threaded increases very significantly memory consumption per job
concurrent_listing_threads	No	5	0-20	1	Number of maximum concurrent backup page listing threads running in parallel in order to fetch sets of data for some modules. Currently it's only used in the email module. This parameter will also affect graph api concurrent requests. Graph API can throttle requests depending on a variety of circumstances, but this parameter impacts it directly. It is only needed to modify this parameter, in general, if you are going to run different jobs in parallel. If you want to have a precise control of your parallelization through different jobs, please set up this value to 1. Please be careful also with the memory requirements, multi-threaded increases very significantly memory consumption per job
graph_timeout	No	9000	Positive integer (milliseconds)	60000	Graph call timeout inside HttpClient
graph_read_timeout	No	300	Positive integer (milliseconds)	30000	Graph read timeout inside HttpClient
graph_retries	No	5	Positive integer (number of retries)	10	Graph number of retries for retry-candidate requestsInclude some stats information in the joblog. Useful to measure task times
graph_retry_delay	No	5	Posi-	10	Graph delay between retries

Note: graph_list_page_size had default value 500 before BEE 14.0.7. A higher value for this parameter can improve the performance at it reduces the number of API calls done to M365 service. However, the service can also be overloaded and return more HTTP 503 errors (Bad Gateway), especially for the email module. Starting from version 16.0.3, default values for backup_queue_size and concurrent_threads have been increased, also the allowed ranges.

Entity parameters

The following list of parameters are commonly shared through any module used into the same fileset line and are intended to select the target entities to backup. Every module subsection mentions what entities are supported too.

Option	Required	Default	Values	Example	Services	Description
user	No		Valid email addresses of existing users on the selected tenant separated by ‘,’	AlexW@yourdomain.com, LeeY@yourdomain.com	email, drive, contact, calendar, onenote, chat, tasks	Backup mailboxes, drive unit spaces, categories, or whatever service is selected of this list of users. If no user is provided: - The backup will be done for all accessible users of the given tenant if no other entry has value either (group or site parameters)
user_exclude	No		Valid email addresses of existing users on the selected tenant separated by ‘,’	LauraG@yourdomain.com, AmandaT@yourdomain.com	email, drive, contact, calendar, onenote, chat, tasks	Exclude selected mailboxes or onedrive spaces. If this is the only parameter found for selection, all elements will be included and this list will be excluded
user_regex_include	No		Valid regex	.*@management\.mydomain.com	email, drive, contact, calendar, onenote, chat, tasks	Backup matching user mailboxes or onedrive spaces. Please, only provide list parameters (user + user_exclude) or regex ones. But do not try to combine them
user_regex_exclude	No		Valid regex	.*@guests\.mydomain.com	email, drive, contact, calendar, onenote, chat, tasks	Exclude matching user mailboxes or onedrive spaces from the selection. Please, only provide list parameters (user + user_exclude) or regex ones. But do not try to combine them. If this is the only parameter found for selection, all elements will be included and this list will be excluded
site	No		Valid names of existing user sharepoint sites on the selected tenant separated by ‘,’	Communication site, Dev site	drive, sharepoint, onenote	Backup onedrive site library space (*OneDrive service*) belonging to this list of sharepoint sites or the site itself (*Sharepoint service*). If no site name is provided: - **OneDrive**: user and site variables will be checked. If no one has value either, backup will be done for all accessible users, groups or sites of the given tenant - **Sharepoint**: all accessible sites of the tenant will be backed up
site_exclude	No		Valid names of existing user sharepoint sites on the selected tenant separated by ‘,’	Test site	drive, sharepoint, onenote	Exclude listed sites from backup. If this is the only parameter found for selection, all elements will be included and this list will be excluded
site_regex_include	No		Valid regex	.*site	drive,	Backup matching sites. Please, only provide list parame-

Backup parameters

Please, check the specific module pages in order to see backup parameters that are applicable only to each of them:

- *OneDrive*
- *Emails/Mailboxes*
- *Calendars*
- *Contacts*
- *Sharepoint*
- *OneNote*
- *Tasks*
- *Teams*
- *Chat*

Deprecated parameters

The following parameters have been deprecated since version 12.8.3, in order to provide more flexibility. General shared parame

- files -> email_files, drive_files, calendar_files, contact_files
- files_exclude -> email_files_exclude, drive_files_exclude, calendar_files_exclude, contact_files_exclude
- files_regex_include -> email_files_regex_include, drive_files_regex_include, calendar_files_regex_include, contact_files_regex_include
- files_regex_exclude -> email_files_regex_exclude, drive_files_regex_exclude, calendar_files_regex_exclude, contact_files_regex_exclude
- multilevel_attach -> email_multilevel_attach, calendar_multilevel_attach
- exclude_attachments -> email_exclude_attachments, calendar_exclude_attachments
- system_include -> drive_system_include, sharepoint_system_include
- version_history -> drive_version_history, sharepoint_version_history
- backup_threads -> backup_queue_size

Restore parameters

The plugin is able to restore to the local file system on the server where the File Daemon is running or to the Microsoft 365 environment. The method is selected based on the value of the *where* parameter at restore time:

- Empty or '/' (example: where=/) → Microsoft 365 restore will be triggered
- Any other path for where (example: where=/tmp) → Local file system restore will be triggered

When using the Microsoft 365 restore option, the following parameters may be modified by selecting 'Plugin Options' during the bconsole restore session:

Option	Required	Default	Values
destination_user	No		Existing email address on the target Azure AD Tenant
destination_group	No		Existing group name address on the target Azure AD Tena

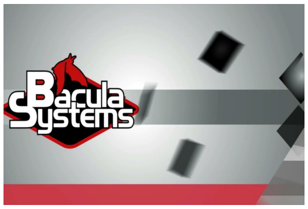
Option	Required	Default	Values
destination_site	No		Existing site name/display name on the target Azure AD T
destination_path	No		Existing path on the selected user (mailfolder path or oned
destination_drive	No		Existing drive name or drive id in the destination entity
send_report	No	0	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, tru
allow_duplicates	No	1	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, tru
drive_skip_sharedwitme	No	0	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, tru
skip_versions	No	1	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, tru
restore_share_permissions	No	0	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, tru
drive_send_invitations	No	0	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, tru
drive_invitations_message	No		String
calendar_name	No		String
sharepoint_list_name	No		String
sharepoint_newsite_name	No		String
sharepoint_newsite_owner	No		String
sharepoint_local_template_path	No		String
notebook_name	No		String
notesection_name	No		String
notesection_group_name	No		String
team_name	No		String
team_channel_name	No		String
team_private_channels_mode	No	DELEGATED	DELEGATED, PUBLIC, SKIP
team_guest_members_enable	No	0	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, tru
chat_topic	No		String
tasklist_name	No		String
tasklist_skip_sharedwithme	No	1	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, tru
plan_name	No		String
plan_create_tab	No	1	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, tru
local_path_json_objects	No	{path}/bacula-restores	String
foreign_container_generation	No	1	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, tru
tenant	No		A valid tenant id string where some bacula m365 plugin a
appid	No		Appid of bacula-m365-plugin-standalone app registered i
objectid	No		Object id associated to set up appid
secret	No		Valid secret associated to appid
debug	No	0	0, 1, 2 ,3, 4, 5, 6, 7, 8, 9

7.3 Automatic Objects Integration

Since Bacula version 16.0.7, a new solution has been introduced, so that each object can be backed up separately with different Jobs to maximize the throughput and the resiliency. It is highly recommended to use this new solution for that purpose - Automatic Object Integration (Scan Plugin). See an example for Microsoft 365Example.

8 Operations

To learn about backup and restore from a video, click on the image below:



8.1 Backup

Microsoft 365 plugin backup configurations currently have one specific requirement in the Job resource. Below we show some examples.

Job Example

The only special requirement with M365 jobs is that Accurate mode backups must be disabled, as this feature is not supported at this time.

Listing 78: **Job Example**

```
Job {
Name = m365-mytenant-backup
FileSet = fs-m365-email-all
Accurate = no
...
}
```

FileSet Examples

The plugin supports enough flexibility to configure almost any type of desired backup. Multiple *Plugin=* lines should not be spoecified in he Include section of a FileSet for the M365 Plugin.

Fileset examples for every supported service are linked below. For common purposes, the following two examples show how to configure an external config file or configure the number of threads:

Setup external config file:

Listing 79: **Fileset Example**

```
FileSet {
   Name = FS_M365_DRIVE
   Include {
      Options {
        signature = MD5
      }
      Plugin = "m365: config_file=/opt/bacula/etc/m365.settings service=drive"
   }
}
```

Listing 80: **Settings file**

```
$ cat /opt/bacula/etc/m365.settings
tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc
objectid=56ddf1h9-eb5d-42nf-bac7-7b019fd284g5
```

Increase number of threads:

Listing 81: **Fileset Example**

```
FileSet {
   Name = fs-m365-email-adelev
   Include {
      Options {
        signature = MD5
      }
      Plugin = "m365: service=email tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc␣
↪objectid=56ddf1h9-eb5d-
   42nf-bac7-7b019fd284g5 user=adelev@baculaenterprise.onmicrosoft.com backup_threads=10"
   }
}
```

More fileset examples for:

- *OneDrive*
- *Emails/Mailboxes*
- *Calendars*
- *Contacts*
- *Sharepoint*
- *OneNote*
- *Tasks*
- *Teams*
- *Chat*

8.2 Restore

Restore operations are done using standard **Bacula Enterprise** bconsole commands.

The *where* parameter controls if the restore will be done locally to the File Daemon's file system or to the Microsoft 365 service:

- where=/ or empty value → Restore will be done over M365
- where=/any/other/path → Restore will be done locally to the File Daemon file system

Restore options are described in the *Restore parameters* section of this document, so here we are going to simply show an example restore session, particularly this example is about OneDrive service:

Listing 82: **Restore Drive Bconsole Session**

```
*restore where=/

First you select one or more JobIds that contain files
to be restored. You will be presented several methods
of specifying the JobIds. Then you will be allowed to
select which files from those JobIds are to be restored.

To select the JobIds, you have the following choices:
1: List last 20 Jobs run
2: List Jobs where a given File is saved
3: Enter list of comma separated JobIds to select
4: Enter SQL list command
5: Select the most recent backup for a client
6: Select backup for a client before a specified time
7: Enter a list of files to restore
8: Enter a list of files to restore before a specified time
9: Find the JobIds of the most recent backup for a client
10: Find the JobIds for a backup for a client before a specified time
11: Enter a list of directories to restore for found JobIds
12: Select full restore to a specified Job date
13: Select object to restore
14: Cancel
Select item: (1-14): 5
Automatically selected Client: 127.0.0.1-fd
Automatically selected FileSet: FS_M365_DRIVE
+-------+-------+----------+-------------+---------------------+-------------------+
| jobid | level | jobfiles | jobbytes    | starttime           | volumename        |
+-------+-------+----------+-------------+---------------------+-------------------+
| 11    | F     | 190      | 332,978,505 | 2021-01-22 10:39:34 | TEST-2021-01-22:0 |
| 12    | I     | 1        | 550         | 2021-01-22 10:43:05 | TEST-2021-01-22:0 |
+-------+-------+----------+-------------+---------------------+-------------------+
You have selected the following JobIds: 11,12

Building directory tree for JobId(s) 11,12 ...␣
↪+++++++++++++++++++++++++++++++++++++++++++++++
188 files inserted into the tree.

You are now entering file selection mode where you add (mark) and
remove (unmark) files to be restored. No files are initially added, unless
you used the "all" keyword on the command line.
Enter "done" to leave this mode.

cwd is: /
$ cd /@m365/baculaenterprise/adelev@baculaenterprise.onmicrosoft.com/drive/root:
cwd is: /@m365/baculaenterprise/adelev@baculaenterprise.onmicrosoft.com/drive/root:/
$ ls
Docs/
pluginTest.drive.deltaLink
sharedWithMe/
test4###v1.0_2020-11-25_153507.html
```

(continues on next page)

(continued from previous page)

```
test4###v2.0_2020-11-25_153507.html
test4###v3.0_2020-12-02_180612.html
test4###v4.0_2020-12-02_180612.html
test4###v5.0_2020-12-03_134422.html
test4###v6.0_2020-12-02_180612.html
test4.html
testlink123###v1.0_2020-11-26_123244.url
testlink123###v2.0_2020-11-26_123244.url
testlink123###v3.0_2020-12-02_180613.url
testlink123###v4.0_2020-12-02_180613.url
testlink123.url
$ mark *
189 files marked.
$ done
Bootstrap records written to /tmp/regress/working/127.0.0.1-dir.restore.3.bsr

The Job will require the following (*=>InChanger):
Volume(s) Storage(s) SD Device(s)
===========================================================================

TEST-2021-01-22:0 File FileStorage

Volumes marked with "*" are in the Autochanger.
189 files selected to be restored.

Using Catalog "MyCatalog"
Run Restore job
JobName: RestoreFiles
Bootstrap: /tmp/regress/working/127.0.0.1-dir.restore.3.bsr
Where: /
Replace: Always
FileSet: Full Set
Backup Client: 127.0.0.1-fd
Restore Client: 127.0.0.1-fd
Storage: File
When: 2021-01-22 11:54:55
Catalog: MyCatalog
Priority: 10
Plugin Options: *None*
OK to run? (yes/mod/no): mod
Parameters to modify:
1: Level
2: Storage
3: Job
4: FileSet
5: Restore Client
6: When
7: Priority
8: Bootstrap
9: Where
10: File Relocation
11: Replace
```

(continues on next page)

(continued from previous page)

```
12: JobId
13: Plugin Options
Select parameter to modify (1-13): 13
Automatically selected : m365: service=drive tenant="574dda43-d107-48e2-a7f2-aa51c10bdaec
↪" objectid="31ddf4b1-
ed5d-432f-bac7-7b946fd23394" user="adelev@baculaenterprise.onmicrosoft.com" drive_
↪version_history=yes debug=3
Plugin Restore Options
Option Current Value Default Value
destination_user: *None* (*None*)
destination_path: *None* (*None*)
send_report: *None* (0)
email_allow_duplicates: *None* (1)
drive_skip_sharedwithme: *None* (0)
drive_skip_versions: *None* (1)
drive_restore_share_permissions: *None* (0)
drive_send_invitations: *None* (0)
drive_invitations_message: *None* (*None*)
debug: *None* (*None*)
Use above plugin configuration? (yes/mod/no): mod
You have the following choices:
1: destination_user (Destination User)
2: destination_path (Destination Path in M365)
3: send_report (Send report of the restore operation to the affected user)
4: email_allow_duplicates (Allow Duplicate Emails)
5: drive_skip_sharedwithme (Skip restoring shared with me elements even if they are␣
↪selected)
6: drive_skip_versions (Skip restoring file former versions (tagged with '###date') even␣
↪if they are selected)
7: drive_restore_share_permissions (Restore share permissions of items so they are␣
↪shared if they originally
were)
8: drive_send_invitations (Send email invitations for restored OneDrive shares)
9: drive_invitations_message (Set invitations message for restored OneDrive shares)
10: debug (Change debug level)
Select parameter to modify (1-10): 2
Please enter a value for destination_path: MY_RESTORE_PATH
Plugin Restore Options
Option Current Value Default Value
destination_user: *None* (*None*)
destination_path: MY_RESTORE_PATH (*None*)
send_report: *None* (0)
email_allow_duplicates: *None* (1)
drive_skip_sharedwithme: *None* (0)
drive_skip_versions: *None* (1)
drive_restore_share_permissions: *None* (0)drive_send_invitations: *None* (0)
drive_invitations_message: *None* (*None*)
debug: *None* (*None*)
Use above plugin configuration? (yes/mod/no): yes
Run Restore job
JobName: RestoreFiles
Bootstrap: /tmp/regress/working/127.0.0.1-dir.restore.3.bsr
```

(continues on next page)

(continued from previous page)

```
Where: /
Replace: Always
FileSet: Full Set
Backup Client: 127.0.0.1-fd
Restore Client: 127.0.0.1-fd
Storage: File
When: 2021-01-22 11:54:55
Catalog: MyCatalog
Priority: 10
Plugin Options: User specified
OK to run? (yes/mod/no): yes
Job queued. JobId=14
```

Restore by service

In this section some example restore configurations will be shown:

- *OneDrive*
- *Email*
- *Calendar*
- *Contacts*
- *Sharepoint*
- *Onenote*
- *Tasks*
- *Teams*
- *Chat*

Cross tenant restore

You can perform cross-tenant restores using the restore variables: - tenantid - appid - objectid - secret

Obviously, it is needed to set up the destination tenant values, where one of the authentication methods should be applied first.

If you are using the standalone app authetication model, you will need to use the four values in your restore session.

If you want to use the common app authentication model, you won't need to put the secret. On the other hand, please, ask the bacula enterprise edition support team in order to get the correct value for appid.

8.3 List

It is possible to list information using the bconsole *.ls* command and providing a path. In general, we need to provide the service parameter and a path representing a folder or object (like calendars).

There are some general commands (user, groups, sites), the rest of the commands need to point to the correct service.

Below some examples:

List general info: Users, groups, sites

Here we are showing these 3 commands using the bconsoel *.ls* command, but notice you may also use them with the query interface (keep your variable values, but apply something like: *.query plugin="… " client=xxxx parameter=xxx*)

Listing 83: **List example: General information**

```
*.ls plugin="m365: tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc objectid=56ddf1h9-eb5d-
↪42nf-bac7-7b019fd284g5"
client=127.0.0.1-fd path=/user
Connecting to Client 127.0.0.1-fd at 127.0.0.1:8102
ptcomm: Starting Plugin Job
ptcomm: Finished reading Plugin Params
ptcomm: Backend connection to M365 stablished
ptcomm: Connected to tenant: baculaenterprise
ptcomm: Starting ListingStart
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /AdeleV@baculaenterprise.
↪onmicrosoft.com
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /AlexW@baculaenterprise.
↪onmicrosoft.com
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /DiegoS@baculaenterprise.
↪onmicrosoft.com
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /jane@baculaenterprise.
↪onmicrosoft.com
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /GradyA@baculaenterprise.
↪onmicrosoft.com
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /HenriettaM@baculaenterprise.
↪onmicrosoft.com
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /IsaiahL@baculaenterprise.
↪onmicrosoft.com
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /JohannaL@baculaenterprise.
↪onmicrosoft.com
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /JoniS@baculaenterprise.
↪onmicrosoft.com
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /john@baculaenterprise.
↪onmicrosoft.com
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /LeeG@baculaenterprise.
↪onmicrosoft.com
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /LidiaH@baculaenterprise.
↪onmicrosoft.com
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /LynneR@baculaenterprise.
↪onmicrosoft.com
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /MeganB@baculaenterprise.
↪onmicrosoft.com
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /MiriamG@baculaenterprise.
↪onmicrosoft.com
```

(continues on next page)

(continued from previous page)

```
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /NestorW@baculaenterprise.
↪onmicrosoft.com
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /PattiF@baculaenterprise.
↪onmicrosoft.com
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /PradeepG@baculaenterprise.
↪onmicrosoft.com
2000 OK estimate files=18 bytes=0
*.ls plugin="m365: tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc objectid=56ddf1h9-eb5d-
↪42nf-bac7-7b019fd284g5"
client=127.0.0.1-fd path=/group
Connecting to Client 127.0.0.1-fd at 127.0.0.1:8102
ptcomm: Starting Plugin Job
ptcomm: Finished reading Plugin Params
ptcomm: Backend connection to M365 stablished
ptcomm: Connected to tenant: baculaenterprise
ptcomm: Starting ListingStart
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Contoso Team
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Dev Team
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /baculaenterprise
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /All Company
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Bacula Systems Team Site
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Dev people Group
2000 OK estimate files=6 bytes=0
*.ls plugin="m365: tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc objectid=56ddf1h9-eb5d-
↪42nf-bac7-7b019fd284g5"
client=127.0.0.1-fd path=/site
Connecting to Client 127.0.0.1-fd at 127.0.0.1:8102
ptcomm: Starting Plugin Job
ptcomm: Finished reading Plugin Params
ptcomm: Backend connection to M365 stablished
ptcomm: Connected to tenant: baculaenterprise
ptcomm: Starting ListingStart
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Communication site
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Adele Vance
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Alex Wilber
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Diego Siciliani
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Grady Archie
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Henrietta Mueller
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Isaiah Langer
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Johanna Lorenz
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Joni Sherman
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Lidia Holloway
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Lynne Robbins
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Megan Bowen
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Pradeep Gupta
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Bacula Systems Team Site
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Work @ Contoso
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Contoso Team
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Live @ Contoso
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Give @ Contoso
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Dev Team
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Dev people Group
```

(continues on next page)

(continued from previous page)

```
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Team Site
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /John Doe
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Jane Doe
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Patti Fernandez
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Lee Gu
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Miriam Graham
-rw-r--r--    1 root    root        -1 2106-02-07 07:28:15 /Nestor Wilke
2000 OK estimate files=28 bytes=0
```

List Onedrive contents

Listing 84: **List example: General information**

```
*.ls plugin="m365: tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc objectid=56ddf1h9-eb5d-
↪42nf-bac7-7b019fd284g5
service=drive user=adelev@baculaenterprise.onmicrosoft.com" client=127.0.0.1-fd path=/
↪Connecting to Client 127.0.0.1-fd at 127.0.0.1:8102
m365: Starting Plugin Job
m365: Finished reading Plugin Params
m365: Connected to tenant: baculaenterprise
m365: Backend connection to M365 stablished
m365: Starting ListingStart
drwxr-xr-x   1 nobody    nogroup            26800469 2020-11-08 01:34:35 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/root/
drwxr-xr-x   1 nobody    nogroup            9876293 2020-12-03 15:28:31 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/Docs/
-rw-r-----   1 nobody    nogroup            217873 2021-04-13 13:29:56 /baculaenterprise/
↪users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/Hats.JPG
-rw-r-----   1 nobody    nogroup   1914266 2021-02-01 17:45:34 /baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/lombok.jar
-rw-r-----   1 nobody    nogroup            785043 2021-02-01 17:45:56 /baculaenterprise/
↪users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/noShared.dat
-rw-r-----   1 nobody    nogroup            4 2021-02-01 17:45:35 /baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/test2.doc
-rw-r-----   1 nobody    nogroup            4 2020-11-25 15:35:07 /baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/test4.html
-rw-r-----   1 nobody    nogroup            42 2020-11-26 12:32:44 /baculaenterprise/
↪users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/testlink123.url
drwxr-xr-x   1 nobody    nogroup            -1 2106-02-07 07:28:15 /baculaenterprise/
↪users
/adelev@baculaenterprise.onmicrosoft.com/sharedWithMe/
2000 OK estimate files=23 bytes=53,600,938
*.ls plugin="m365: tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc objectid=56ddf1h9-eb5d-
↪42nf-bac7-7b019fd284g5
service=drive user=adelev@baculaenterprise.onmicrosoft.com" client=127.0.0.1-fd path=/
↪sharedWithMe
```

(continues on next page)

(continued from previous page)

```
Connecting to Client 127.0.0.1-fd at 127.0.0.1:8102
m365: Starting Plugin Job
m365: Finished reading Plugin Params
m365: Connected to tenant: baculaenterprise
m365: Backend connection to M365 stablished
m365: Starting ListingStart
-rw-r-----   1 nobody    nogroup               11722 2021-04-13 17:42:06 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/sharedWithMe/SharePoint␣
↪App/Aeque.ppt
-rw-r-----   1 nobody    nogroup               4039 2021-04-13 17:42:02 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/sharedWithMe/SharePoint␣
↪App/Voluptatibus.txt
-rw-r-----   1 nobody    nogroup               7697 2021-04-13 17:41:59 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/sharedWithMe/SharePoint␣
↪App/Nullam.jpeg
-rw-r-----   1 nobody    nogroup               21031 2021-04-13 17:41:55 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/sharedWithMe/SharePoint␣
↪App/Graeci.ppt
-rw-r-----   1 nobody    nogroup               21417 2021-04-13 17:41:52 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/sharedWithMe/SharePoint␣
↪App/Habeo.txt
-rw-r-----   1 nobody    nogroup               15354 2021-04-13 17:41:47 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/sharedWithMe/SharePoint␣
↪App/Atomorum.jpeg
-rw-r-----   1 nobody    nogroup               22217 2021-04-13 17:41:43 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/sharedWithMe/SharePoint␣
↪App/Faucibus.ppt
-rw-r-----   1 nobody    nogroup               15714 2021-04-12 11:13:52 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/sharedWithMe/SharePoint␣
↪App/Sodales.ppt
-rw-r-----   1 nobody    nogroup               14371 2021-04-12 11:13:48 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/sharedWithMe/SharePoint␣
↪App/Cum.txt
-rw-r-----   1 nobody    nogroup               11059 2021-04-12 11:13:44 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/sharedWithMe/SharePoint␣
↪App/Sapien.jpeg
-rw-r-----   1 nobody    nogroup               14144 2021-04-12 11:13:41 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/sharedWithMe/SharePoint␣
↪App/Habitasse.ppt
-rw-r-----   1 nobody    nogroup               12594 2021-04-12 11:13:37 /
↪baculaenterprise/users
```

(continues on next page)

(continued from previous page)

```
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/sharedWithMe/SharePoint␣
↪App/Deserunt.txt
-rw-r-----   1 nobody    nogroup               14923 2021-04-12 11:13:32 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/sharedWithMe/SharePoint␣
↪App/Postulant.jpeg
-rw-r-----   1 nobody    nogroup                21307 2021-04-12 11:13:29 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/sharedWithMe/SharePoint␣
↪App/Tation.ppt
drwxr-xr-x   1 nobody    nogroup                0 2021-01-08 14:32:20 /baculaenterprise/
↪users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/sharedWithMe/
↪john@baculaenterprise.onmicrosoft.
com/JohnFolder/
-rw-r-----   1 nobody    nogroup                1442312 2021-01-08 11:57:09 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/sharedWithMe/
↪john@baculaenterprise.onmicrosoft.
com/developers.pdf
-rw-r-----   1 nobody    nogroup                630174 2021-01-08 11:57:08 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/sharedWithMe/
↪john@baculaenterprise.onmicrosoft.
com/console.pdf
2000 OK estimate files=17 bytes=2,280,075*.ls plugin="m365: tenant=57uia43-d107-17a2-
↪a2g2-aa53c10tdahc objectid=56ddf1h9-eb5d-42nf-bac7-7b019fd284g5
service=drive user=adelev@baculaenterprise.onmicrosoft.com" client=127.0.0.1-fd path=/
↪Docs
Connecting to Client 127.0.0.1-fd at 127.0.0.1:8102
m365: Starting Plugin Job
m365: Finished reading Plugin Params
m365: Connected to tenant: baculaenterprise
m365: Backend connection to M365 stablished
m365: Starting ListingStart
drwxr-xr-x   1 nobody    nogroup                9876293 2020-12-03 15:28:31 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/Docs/
drwxr-xr-x   1 nobody    nogroup                793475 2020-11-25 15:34:49 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/Docs/Simple Sub Folders/
-rw-r-----   1 nobody    nogroup                6383460 2020-12-08 10:13:11 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/Docs/E16WKAMS.10E.zip
-rw-r-----   1 nobody    nogroup                1914266 2020-12-08 10:13:32 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/Docs/lombok.jar
-rw-r-----   1 nobody    nogroup                785043 2021-01-12 10:12:04 /
↪baculaenterprise/users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/Docs/noShared.dat
-rw-r-----   1 nobody    nogroup
24 2020-11-25 15:35:05 /baculaenterprise/users
```

(continues on next page)

(continued from previous page)

```
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/Docs/test1.txt
-rw-r-----   1 nobody    nogroup                13 2021-01-11 15:39:57 /baculaenterprise/
↪users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/Docs/test11.txt
-rw-r-----   1 nobody    nogroup                4 2020-11-25 15:35:05 /baculaenterprise/
↪users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/Docs/test2.doc
-rw-r-----   1 nobody    nogroup                4 2020-11-25 15:35:05 /baculaenterprise/
↪users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/Docs/test3.pdf
-rw-r-----   1 nobody    nogroup                4 2020-12-03 16:52:32 /baculaenterprise/
↪users
/adelev@baculaenterprise.onmicrosoft.com/drives/onedrive/root:/Docs/test4.html
2000 OK estimate files=10 bytes=19,752,586
```

Other list examples

Listing 85: **List example: General information**

```
*.ls plugin="m365: tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc objectid=56ddf1h9-eb5d-
↪42nf-bac7-7b019fd284g5 service=email user=adelev@baculaenterprise.onmicrosoft.com"␣
↪client=127.0.0.1-fd path=/Inbox
*.ls plugin="m365: tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc objectid=56ddf1h9-eb5d-
↪42nf-bac7-7b019fd284g5 service=email user=adelev@baculaenterprise.onmicrosoft.com"␣
↪client=127.0.0.1-fd path=/Sent
*.ls plugin="m365: tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc objectid=56ddf1h9-eb5d-
↪42nf-bac7-7b019fd284g5 service=contacts user=adelev@baculaenterprise.onmicrosoft.com"␣
↪client=127.0.0.1-fd path=/
*.ls plugin="m365: tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc objectid=56ddf1h9-eb5d-
↪42nf-bac7-7b019fd284g5 service=contacts user=adelev@baculaenterprise.onmicrosoft.com"␣
↪client=127.0.0.1-fd path=/MyContactsDir
*.ls plugin="m365: tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc objectid=56ddf1h9-eb5d-
↪42nf-bac7-7b019fd284g5 service=outlook_category user=adelev@baculaenterprise.
↪onmicrosoft.com" client=127.0.0.1-fd path=/
*.ls plugin="m365: tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc objectid=56ddf1h9-eb5d-
↪42nf-bac7-7b019fd284g5 service=calendar user=adelev@baculaenterprise.onmicrosoft.com"␣
↪client=127.0.0.1-fd path=/MyCalendar
*.ls plugin="m365: tenant=57uia43-d107-17a2-a2g2-aa53c10tdahc objectid=56ddf1h9-eb5d-
↪42nf-bac7-7b019fd284g5 service=onenote user=adelev@baculaenterprise.onmicrosoft.com"␣
↪client=127.0.0.1-fd path=/MyNotebook/sectionGroups/groupA/sections/secA
```

Performance command

This plugin provides a query command allowing to check the performance of a given setup. The process consist on uploading some randomly generated files of a customizable size (and a customizable number of them) to a target tenant, then download them using the same method the backup itself employs. Upload and download times are measured. An average is calculated and all the information is presented to the user.

Please, note that this command is currently working in single-threaded mode, while the plugin is running with multiple threads (configurable through concurrent_threads variable), so you could have better results with the actual backup process.

The usage of this command is:

Listing 86: **Query perf**

```
*.query client=<your-client> plugin="m365: tenant=<tenantId> objectid=<objectId> user=
↪<your-user> service=<service>" parameter=perf
```

Currently, theh plugin supports two types of performance commands: perf for onedrive (default) and perf for email. So the value service should be: drive or email.

The behavior of this service from performance perspective can differ significantly. Below we detail some important aspects:

- On Onedrive and other services implying user files, it is needed to request a 'download session' for each file.
  - Once we have it, the download starts. This requires significantly more time than just retrieving simple objects.
- In general, objects are not retrieved one by one. We request for lists of objects, that later on are processed.
  - For objects that are not pointing to any other file (as emails without attachments), the time needed per object is very low.

Drive perf command

For Onedrive service, in addition to user, you can also use group or site (site name or siteId) for this command. The final output will be similar to the following:

Listing 87: **Query perf drive output**

```
...
date=2021-09-06 17:25:56
tenant_name=johndoe.onmicrosoft.com
upload-speed-average=3.3 MiB/s
download-speed-average=32.2 MiB/s
latency-average=482.00 ms
console=---- Test END ----
```

You can control the size and the number of files used using the following query parameters: - q_perf_size (in bytes) - q_perf_files (number of files)

By default, their values are: - q_perf_size = 209715200 - q_perf_files = 3

Below we provide a usage example:

Listing 88: **Query perf drive sample**

```
*.query client=127.0.0.1-fd plugin="m365:
tenant=f421f256-748a-4a80-90da-bf47y66fa166 objectid=90173rea-c144-46ye-8f93-
↪4uh87bcd6d47 site=\"johndoe.sharepoint.com,fda66478-aab5-4896-a6ff-e0fb02af2a40,
↪c024ee54-dcc0-4f1e-ab76-671f6eb87fda\" service=drive q_perf_files=5 q_perf_
↪size=314572800" parameter=perf
```

Email perf command

For Email service, specifying a user is mandatory. The command can generate simple emails, but also emails with one attachment. The final output can be similar to the following:

Listing 89: **Query perf drive output**

```
...
console=---- Summary ----
date=2021-11-11 17:05:18
tenant_name=johndoe.onmicrosoft.com
upload-speed-average=44.6 KiB/s
download-speed-average=220.5 KiB/s
latency-average-per-list-call=510.83 ms
latency-average-per-message=36.18 ms
latency-average-attachments=617.64 ms
console=---- Test END ----
```

The behavior of the email perf command is a little different from the Onedrive one. Instead of uploading/downloading one message at a time, here everything is uploaded first. After that everything is downloaded. The reason is here the backup process uses lists of objects and, unless there are attachments implied, that's all, the objects are simply stored from that list.

Similarly to the onedrive perf command, you can control the size and the number of emails. However, there is an extra parameter to control the size of the attachments:

- q_perf_size (in bytes)
- q_perf_size_attachments (in bytes)
  - If not specified, no attachment will be generated
  - If specified, one simple text attachment will be generated together with each email
- q_perf_files (number of files)

By default, their values are:

- q_perf_size = 10240
- q_perf_size_attachments = 0
- q_perf_files = 50

Below we provide a usage example:

Listing 90: **Query perf email sample**

```
*.query client=127.0.0.1-fd plugin="m365:
tenant=f421f256-748a-4a80-90da-bf47y66fa166 objectid=90173rea-c144-46ye-8f93-
↪4uh87bcd6d47 site=\"johndoe.sharepoint.com,fda66478-aab5-4896-a6ff-e0fb02af2a40,
↪c024ee54-dcc0-4f1e-ab76-671f6eb87fda\" service=email q_perf_size_attachment=204800"␣
↪parameter=perf
```

9 Best practices

9.1 Jobs Distribution

It is recommended, to split the target backup between different groups of entities or even having one job per entity (user, group, site, etc). This way errors in one job will not invalidate a whole backup cycle where some entities have been successful and some others had errors. This also makes easier to identify the cause of the error.

9.2 Concurrency

Microsoft public APIs impose a variety of boundaries that need to be considered. If a boundary is crossed, the corresponding API call will fail and the application will need to wait some amount of time to retry, which is different depending on the boundary crossed.

It is crucial to plan an adequate strategy to backup all the elements without reaching API boundaries. A single job implements some parallelism which can be reduced until a point, if necessary, using the variable **backup_queue_size** (default value is 30)**.** This variable controls the size of the internal queues communicating the internal threads, that are designed to fetch, open and send every item to Bacula core. Reducing its size will produce, ultimately (with a value of 1 for example), an execution very similar to a single threaded process. On the othere hand the plugin has **concurrent_threads** which controls the number of simultaneous processes fetching and downloading data (default value is 5).

If you are going to launch different jobs in parallel it is recommended to configure different services for each of them (protect in parallel email, drive, contacts…). However, be careful with the concurrency over the same service (in general, it is recommended a maximum of 4-5 jobs working with the same service) and plan a step-by-step testing scenario before putting it into production. Other important point is the timing schedule, as some boundaries are related to time-frames (number of request per 10 minutes or 1 hour, for example). If you detect you reach boundaries when running all your backups during a single day of the week, please try to use 2 or 3 days and spread the load through them in order to achieve better performance results.

More information about Microsoft 365 Graph API boundaries may be found here:

https://docs.microsoft.com/graph/throttling

9.3 Disk Space

It is necessary to have at least enough disk space available for the size of the largest file in the backup session. If you are using concurrency between jobs or through the same job (by default this is the case through the concurrent_threads=5 parameter), you would need at least that size for the largest file multiplied by the number of operations in parallel you run.

Read more details in the *Backup of Attachments and Files* section.

9.4 Performance

The performance of this plugin is highly dependent on many external factors:

- ISP latency and bandwidth
- Network infrastructure
- FD Host hardware
- FD Load
- ...

In summary, it is not possible to establish an exact reference about how much time a backup will need to complete.

As reference values, in testing environments and using **sequential calls** we had the following results:

- 3000 emails in the same folder of a single mailbox, with emails ranging from 500 to 3000 words - about 10K to 30K in size
- The total time required to back them up was: 88243 ms.
- That implies a total time per email of about **30ms**

Concurrency has also been tested. For example, when using single-threaded mode (concurrent_threads=1) with emails, good values where found for 4-6 concurrent jobs that allowed us to backup around 200 emails per second (sizes between 20K-30K) when no attachments were implied. As more attachments were implied, the number of emails will decrease very significantly, while the overall size speed will be increased. This can be generalized:

- Many little objects to protect -> More objects per second, but less speed (MB/s)
- Big files to protect -> Less objects per second, but greater speed (MB/s)

It is recommended to benchmark your own environment in base to your requirements and needs.

The automatic parallelization mechanism (using concurrent_threads=x, default is 5) should work well for most scenarios, however, fine tune is possible if we define one job per entity and we control how many of them run in pararllel, together to decrease the concurrent_threads value in order to avoid throttling from MS Graph API.

There are many different possible strategies to use this plugin, so please, study what is best suiting for your needs before deploying the jobs for your entire environment, so you can get best possible results:

- You can have a job per entity (users, groups, sites...) and all services
- You can have inside a job multiple entities and only some services
- You can split your workload through an schedule, or try to run all your jobs together.
- You can run jobs in parallel or take advantadge of concurrent_threads and so run less jobs in parallel
- You can select what services to backup or backup them all
- You can backup whole services to backup or select precisely what elements you really need inside each service (folders, paths, exclussions...)
- For delegated permissions services, you can proceed with the login previously to backups (recommended) or you can wait for the job to ask for what it needs
- etc.

9.5 Best practices by service

Each Microsoft 365 service presents its own particularties. Moreover each deployment and each environment can be protected through different strategies depending on the size, number of entities and other parameters. However, there are some general considerations that can be helpful to consider for a selection of services:

Sharepoint

Sharepoint Online is the service where the target data varies the most among the M365 services. Sites can be structured very differently and there exists many different kinds of Sharepoint lists. In addition to that, some of them could have very large document libraries, while others could be very sparse.

For these reasons, it is recommended to split Sharepoint jobs to be as atomic as possible.

Ideally jobs should be split up to a single site backup job. This will help to monitor the health of the backups, to keep good backups for all sites presenting no issues and but to quickly troubleshoot any site presenting any kind of inconvenience. Usually, this is especially needed when Sharepoint service are intensively used.

If sites are created and removed often, another option to avoid a frequent configuration change is to employ the site_regex_include plugin variable and use the naming patterns to set up the jobs, so they will also include future created sites (possible examples: .*Team, .*Web, Dev.* …).

In order to facilitate the job split, it is possible to use the listing or query commands described in *List general info: Users, groups, sites* to have the site names list.

Another best practice is to split Sharepoint sites into two parts with two different jobs: - Pure Sharepoint job: Use sharepoint service (service=sharepoint site=SiteName) and the parameter sharepoint_include_drive_items=no. - Drive part job: Use drive service (service=drive site=SiteName)

Sharepoint jobs, when using the associated powershell commands, cannot be run in parallel. One will wait for any other session to finish, before executing the corresponding commands. Using the above strategy makes Sharepoint jobs to be much more quick and light, so it's a lot easier to control such concurrency. Other ways to control the concurrency in Bacula Enterprise are:

- Designing separated time windows through scheduling
- Sending all Sharepoint jobs to the same pool and the associated storage resource (both of them dedicated to Sharepoint backups) and limiting ConcurrentJobs to the desired value in that particular storage
- Having 1 specific client resource for Sharepoint and use ConcurrentJobs parameter there.

In general, it's not needed to enable system lists, hidden lists or even the versioning of the elements. All those options will impact the performance of the jobs directly, so you should plan carefully the usage of such parameters.

Some Sharepoint sites present an issue with the hashcheck of the files contained in some of its document libraries. If the hash is not correct at M365 side, it does not match the one that is calculated by the plugin locally, while the file is correct. This situation causes a lot of error messages around the hash and also many extra calls to the service to re-check this hash. When this situation happens, it is recommended to disable the hashcheck mechanism for the affected site using the parameter 'drive_disabldrive_disable_hashcheck=true'.

Onenote

Onenote is the most sensitive module to throttling limits in Microsoft 365 APIs: https://learn.microsoft.com/en-us/graph/throttling

It is recommended to be selective with the information to be protected and make one job for each selected entity: that is one job per user, group or site.

In case throttling problems appears with this strategy, jobs will need to be spread into larger time windows.

It is not recommended to include onenote for entity focused strategies where there is one job per user including all services (email, drive, contact, calendar ..). It is recomended to have specific onenote jobs.

Chats and Todo

This service uses delegated permissions. They can be protected together, but we recomend to run the 'login' query command for each individual user before running the actual jobs.

Email

Email service allows you to protect not only email and attachments, but also settings, user categories and an additional MIME copy for each email with its attachments. MIME files are very useful if you are planning to migrate your data from M365 to any other Email tool, however, the cost of getting each MIME file is very high and it duplicates backup time. MIME files are not needed to perform a local restore or a restore over M365. Therefore, it is recommended to disable MIME backup for a general basis and use it only in case of planning a complete migration of the data.

Similarly to other services, it is recommended to split the backup set into small sets of users (ideally one backup per mailbox, but it's also fine to group them by 5 or 10), specially if there is a large number of mailboxes and they contain thousands of emails. To make this easier, the plugin offers regular expression selection parameters (user_regex_include), as well as listing and query commands to analyze the backup target (*List general info: Users, groups, sites*).

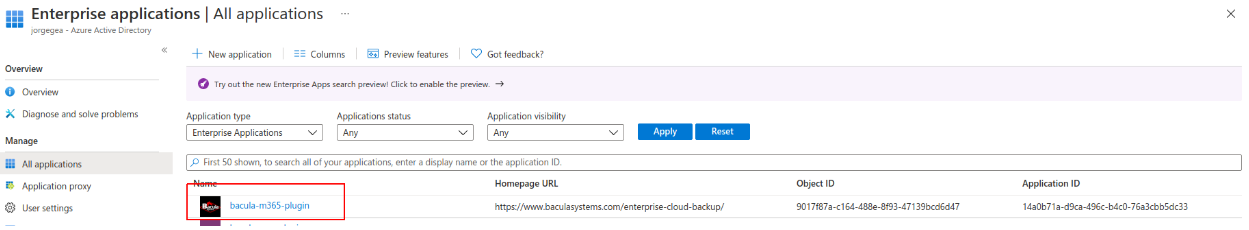
10 Update app permissions

Bacula Enterprise Microsoft 365 Plugin is an application that is under continous evolution. In consequence it may happen that new features require new permissions to function properly.

Bacula Systems will add the required new permissions to the central bacula-m365-plugin app registered in Azure AD. If you are already using the plugin, you will need to update the permissions of your own app (usually bacula-m365-plugin app itself, available under 'enterprise applications' option in your Azure Active Directory application).

To refresh and allow any new permission to your app. Please apply the following procedure:

1. Access the Azure Portal with a Tenant admin user
2. Locate your bacula-m365-plugin app under 'Enterprise applications' and click on it.



Enterprise applications | All applications

jorgegea - Azure Active Directory

Try out the new Enterprise Apps search preview! Click to enable the preview.

Application type: Enterprise Applications

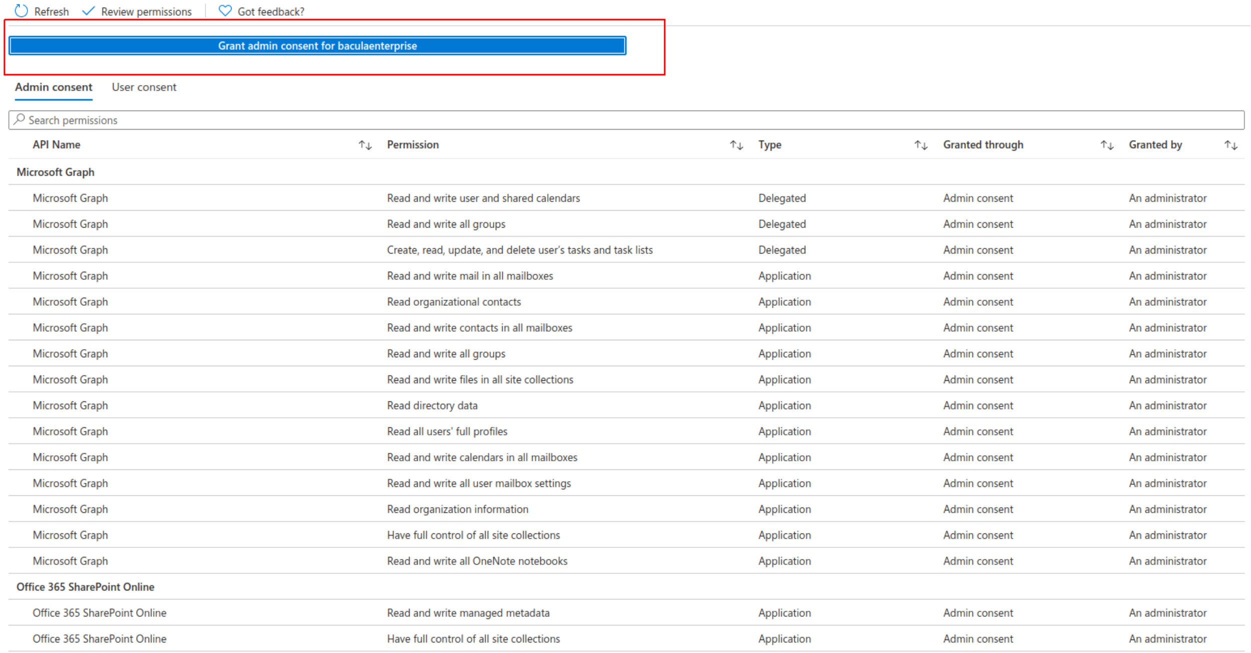
Applications status: Any

Application visibility: Any

First 50 shown, to search all of your applications, enter a display name or the application ID.

Name	Homepage URL	Object ID	Application ID
bacula-m365-plugin	https://www.baculasystems.com/enterprise-cloud-backup/	9017f87a-c164-488e-8f93-47139bcd6d47	14a0b71a-d9ca-496c-b4c0-76a3cbb5dc33

3. In the left menu, click on ‘Permissions’ and then click the big blue button showing ‘Grant admin consent for <yourtenantname>’:



API Name	Permission	Type	Granted through	Granted by
Microsoft Graph				
Microsoft Graph	Read and write user and shared calendars	Delegated	Admin consent	An administrator
Microsoft Graph	Read and write all groups	Delegated	Admin consent	An administrator
Microsoft Graph	Create, read, update, and delete user's tasks and task lists	Delegated	Admin consent	An administrator
Microsoft Graph	Read and write mail in all mailboxes	Application	Admin consent	An administrator
Microsoft Graph	Read organizational contacts	Application	Admin consent	An administrator
Microsoft Graph	Read and write contacts in all mailboxes	Application	Admin consent	An administrator
Microsoft Graph	Read and write all groups	Application	Admin consent	An administrator
Microsoft Graph	Read and write files in all site collections	Application	Admin consent	An administrator
Microsoft Graph	Read directory data	Application	Admin consent	An administrator
Microsoft Graph	Read all users' full profiles	Application	Admin consent	An administrator
Microsoft Graph	Read and write calendars in all mailboxes	Application	Admin consent	An administrator
Microsoft Graph	Read and write all user mailbox settings	Application	Admin consent	An administrator
Microsoft Graph	Read organization information	Application	Admin consent	An administrator
Microsoft Graph	Have full control of all site collections	Application	Admin consent	An administrator
Microsoft Graph	Read and write all OneNote notebooks	Application	Admin consent	An administrator
Office 365 SharePoint Online				
Office 365 SharePoint Online	Read and write managed metadata	Application	Admin consent	An administrator
Office 365 SharePoint Online	Have full control of all site collections	Application	Admin consent	An administrator

4. Wait for a few minutes
5. Refresh the page
6. See the new permissions in the list

New releases needing any new permissions will properly reflect that information with the specific permissions list.

11 Delegated permissions

The great majority of services of the Bacula Enterprise M365 plugin are using the model of ‘Application permissions’. With this model, the plugin application can directly access the implied services once a tenant admin has aproved it.

On the contrary, some other services can require the model of ‘Delegated permissions’ where the plugin needs to impersonate a given user in order to be capable of accessing and restoring the data inside the service. Although we will support more in the future, currently, the only service requesting delegated permissions is:

- Calendar: If any group is included as an entity to protect
- Chat: In order to be able to backup all chat messages belonging to a certain user the user in question will need to delegate the permissions as described below.

Delegated permissions or application permissions are automatically applied when they are needed. However, delegated permissions need user interaction the first time they are invoked, or if the local information about access tokens has been lost for some reason.

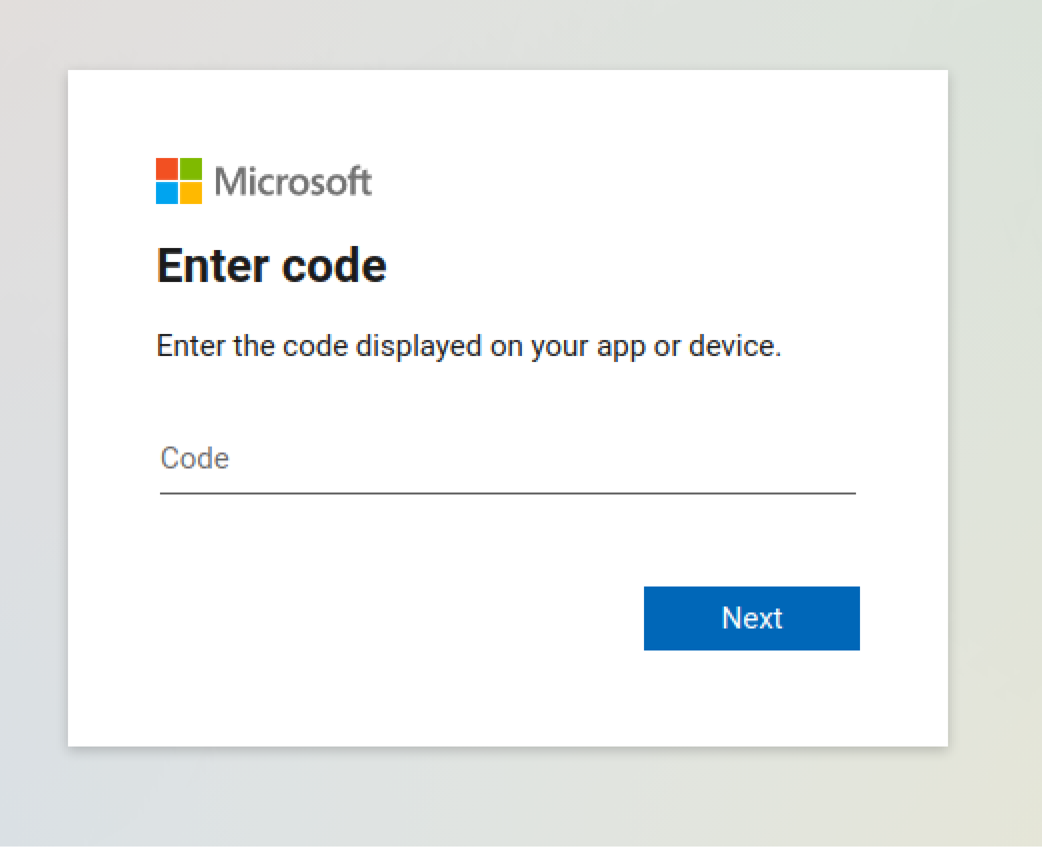
The user interaction can be manually triggered with a special query command that may be run as follows:

Listing 91: **query-login**

```
.query plugin="m365: tenant=e421f055-748a-4a80-90da-bf48b66fa166 objectid=9017f87a-c164-
↪488e-8f93-47139bcd6d47" client=127.0.0.1-fd parameter=login
m365: Starting Plugin Job
m365: Finished reading Plugin Params
m365: Connected to tenant: johndoe
m365: Backend connection to M365 stablished
m365: Starting QueryStart
m365: We don't have a valid token. Please, To sign in, use a web browser to open the␣
↪page https://microsoft.com/devicelogin and enter the code CQ7LETVFT to authenticate.␣
↪You need to do it with an administrator user
```

We only need to follow the instructions, open a browser, enter the code and use the correct user to login (which means a user with access to the resource we want to backup).

For example, if we are working with the calendar of a group, we will need a user with access to that calendar (in general, to protect calendars associated to groups, it will be necessary to create a special backup user with access to all of them and perform the login with, which will avoid the need of performing the login process for every different group).

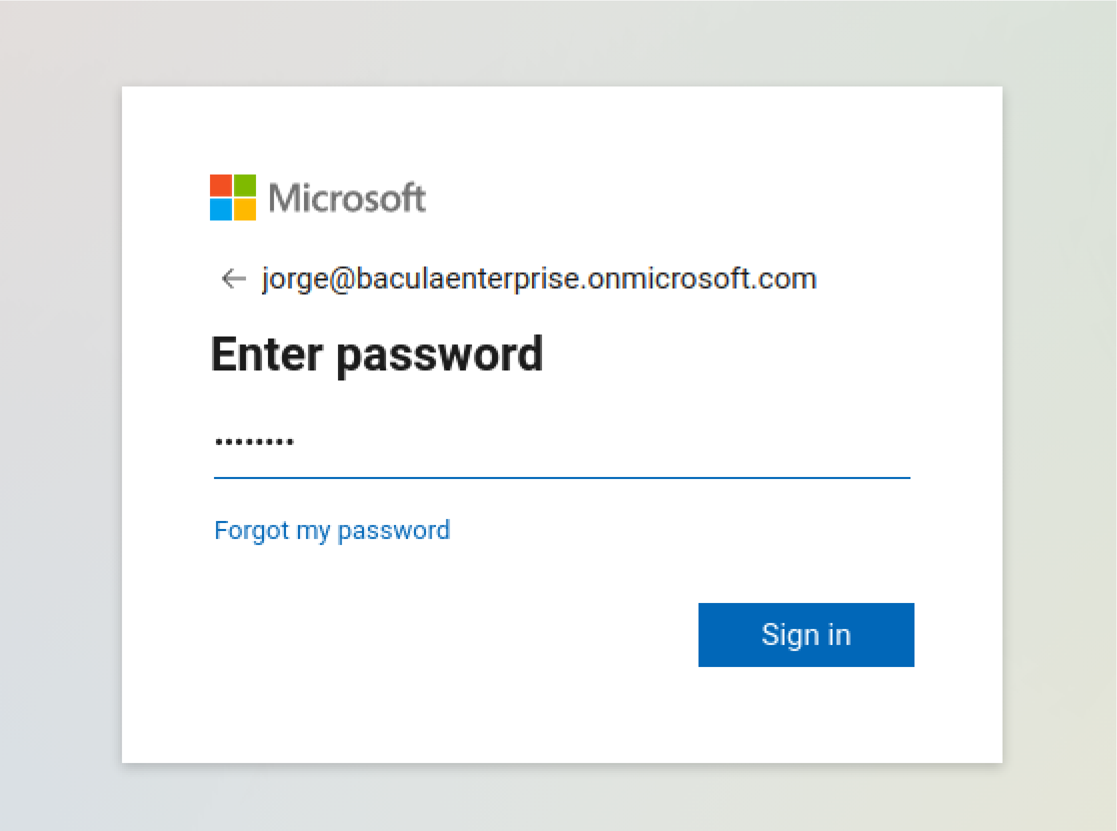


Enter code

Enter the code displayed on your app or device.

Here, if any MFA mechanism is enabled, it will be normally required and the user must complete the associated process. Finally Microsoft will ask to confirm if we want the plugin app to proceed with the operation, and we need to confirm:

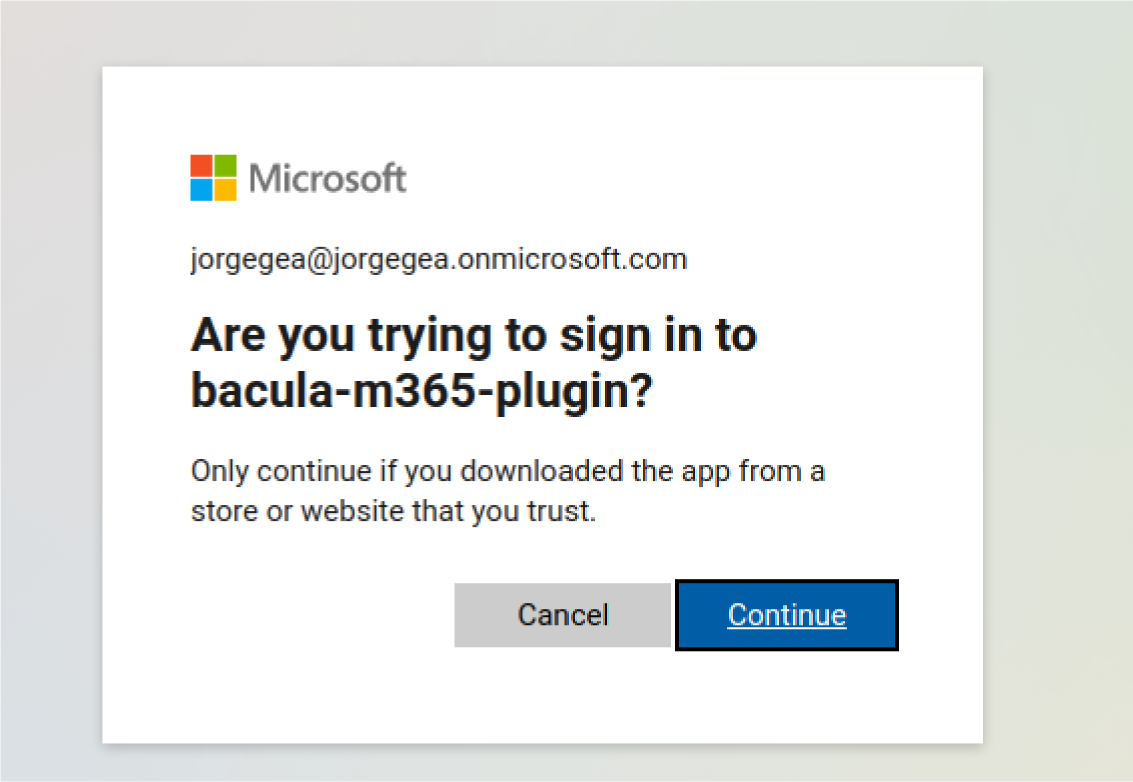
Once the login is complete, the plugin will automatically end the operation and show something like:



jorge@baculaenterprise.onmicrosoft.com

Enter password

••••••••



jorgegea@jorgegea.onmicrosoft.com

Are you trying to sign in to bacula-m365-plugin?

Only continue if you downloaded the app from a store or website that you trust.

Listing 92: **query-login-2**

```
info=We have a valid token got from account: johndoe@johndoe.onmicrosoft.com
info=It will be automatically refreshed when needed. But the current expiration date is:␣
↪2021-06-16T16:18:16Z
info=Backups using services with delegated permissions won't ask for intervention
```

It will also store the token information locally in a special file that it is stored in the working directory inside a folder with the tenant name. For example: /opt/bacula/working/m365/tennantname/bacula-m365-plugin-standalone-token_cache.json

This file is also backed up in the backup operation itself, so it can be restored manually, in case it was lost and in case you don't want to run login operations again

In case the query method is not invoked prior to any job needing it, the job itself will show the message in the joblog and it will be blocked until a user performs the needed interaction described above. Once this happens, the job will automatically continue.

Note: Starting with version 14.0 of Bacula Enterprise you can also see logged in users directly using BWeb, where you can also manually add logged-in users. In order to see more details, please go to section: BWeb Console<./m365-bweb-console>.

12 BWeb Management Console

Bacula Enterprise Microsoft 365 Plugin can be managed from a dedicated user friendly Web console, specifically designed to facilitate tasks as the authentication process for new tenants or the fileset configuration process.

Note: This feature is available starting with version 14.0 of Bacula Enteprise.

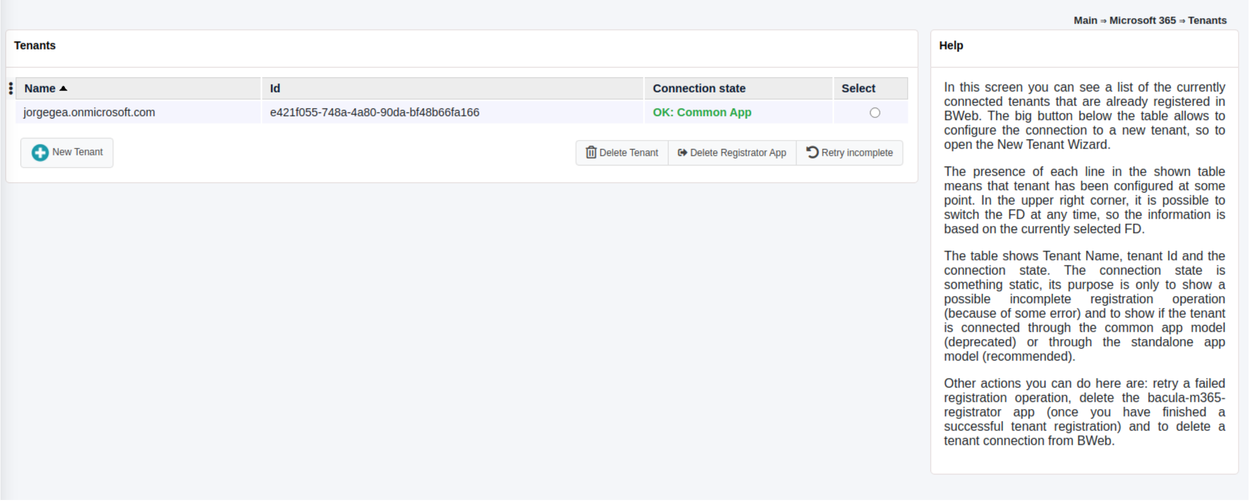
12.1 Menu options

The menu for M365 Plugin is composed by the following entries:

- Microsoft 365
  - Tenants
  - Users
  - Filesets

12.2 Connection

The purpose of this screen is to show a list of the currently connected tenants that are already registered. The presence of each line in the table means that a tenant has been configured at some point.



Name ▲	Id	Connection state	Select
jorgegea.onmicrosoft.com	e421f055-748a-4a80-90da-bf48b66fa166	OK: Common App	○

In this screen you can see a list of the currently connected tenants that are already registered in BWeb. The big button below the table allows to configure the connection to a new tenant, so to open the New Tenant Wizard.

The presence of each line in the shown table means that tenant has been configured at some point. In the upper right corner, it is possible to switch the FD at any time, so the information is based on the currently selected FD.

The table shows Tenant Name, tenant Id and the connection state. The connection state is something static, its purpose is only to show a possible incomplete registration operation (because of some error) and to show if the tenant is connected through the common app model (deprecated) or through the standalone app model (recommended).

Other actions you can do here are: retry a failed registration operation, delete the bacula-m365-registrator app (once you have finished a successful tenant registration) and to delete a tenant connection from BWeb.

In the upper right corner, it is possible to switch the FD at any time, so the information is based on the currently selected FD. After the first time a FD is selected, the information will be remembered using cookies for the next time and for any other M365 Console windows.

The table shows tenant name, tenant id and the connection state. The connection state can indicate a possible incomplete registration operation (because of some error), otherwise it will indicate a successful connection and the type of connection: through the common app or through the standalone app.

The big button placed at the bottom of the table allows us to configure the connection to a new tenant, so it will open the ‘New tenant wizard’ that is described in the section below.

Other actions that can be performed from this tenant list window are:

- Retry operation: It allows to retry a possible incomplete registration, so re-launch the proper internal query command (add-app or the objectid command) using a selected tenant.
- Delete registrator: It allows to delete the registrator app inside the selected tenant. It is not possible to delete a registrator if registration is incomplete . In this case, you must retry operation or delete tenant. This is a possibility only offered for security concerns, as the registrator app has some elevated permissions.
- Delete tenant: It removes the tenant from the internal BWeb configuration.

New tenant wizard

When opening new tenant wizard, we can select the registration model to be used: Standalone or Common.

Details of each model are shown in the images and you can also read more information in this whitepaper under the *Authorization* section.

It is needed to provide the tenant id, which can be found following the whitepaper instructions placed under the same *Authorization* section.

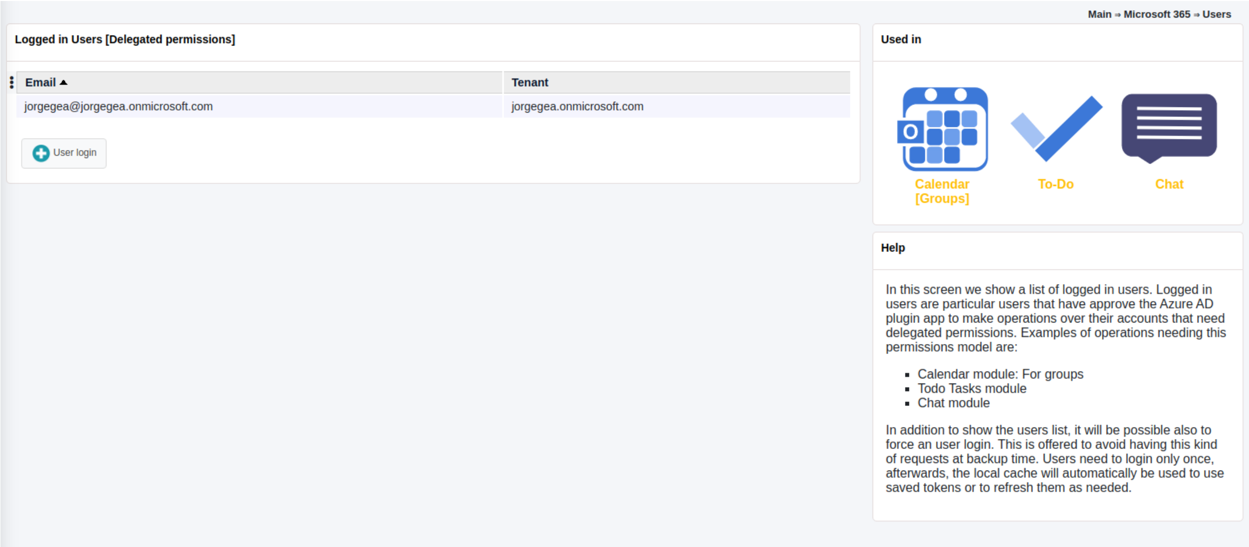
It is recommended to use the Standalone model (first one to appear in the web interface) and be using the common model only for testing purposes.

Once we proceed with one of the models, an URI of Microsoft website will be suggested and you need to provide the needed permissions, then, Microsoft will redirect you to a Bacula Systems page (same as shown in the *Authorization* section).

BWeb will automatically guide you until the end of the process where you will see your tenant correctly added into the tenant list screen.

Logged in users

In this screen a list of logged in users is shown.



Main ⇒ Microsoft 365 ⇒ Users

Logged in Users [Delegated permissions]

Email ▲	Tenant
jorgegea@jorgegea.onmicrosoft.com	jorgegea.onmicrosoft.com

In this screen we show a list of logged in users. Logged in users are particular users that have approve the Azure AD plugin app to make operations over their accounts that need delegated permissions. Examples of operations needing this permissions model are:

- Calendar module: For groups
- Todo Tasks module
- Chat module

In addition to show the users list, it will be possible also to force an user login. This is offered to avoid having this kind of requests at backup time. Users need to login only once, afterwards, the local cache will automatically be used to use saved tokens or to refresh them as needed.

Logged in users are particular users that have approved the Azure AD plugin app to make operations over their accounts that need delegated permissions.

Examples of operations needing this permissions model are:

- Backup of calendar of groups
- Todo Tasks module
- Chats module

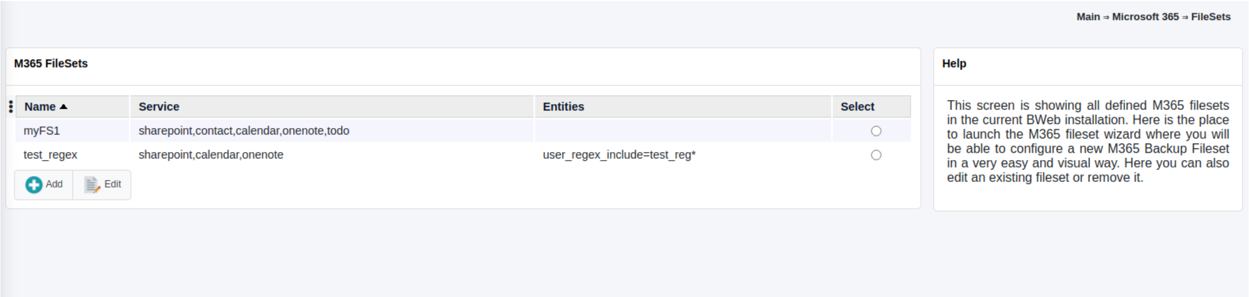
In addition to show the users list, it is also possible to force an user login. This function is offered to avoid having this kind of requests at backup time.

Users need to login only once, afterwards, the local cache will automatically be used to use saved tokens or to refresh them as needed.

For more information about delegated permissions, please take a look to <./m365-delegated-permissions>

12.3 FileSets

The purpose of the FileSets page is to control all the filesets of Microsoft 365 Plugin, so we will see them listed in the main screen:



M365 FileSets

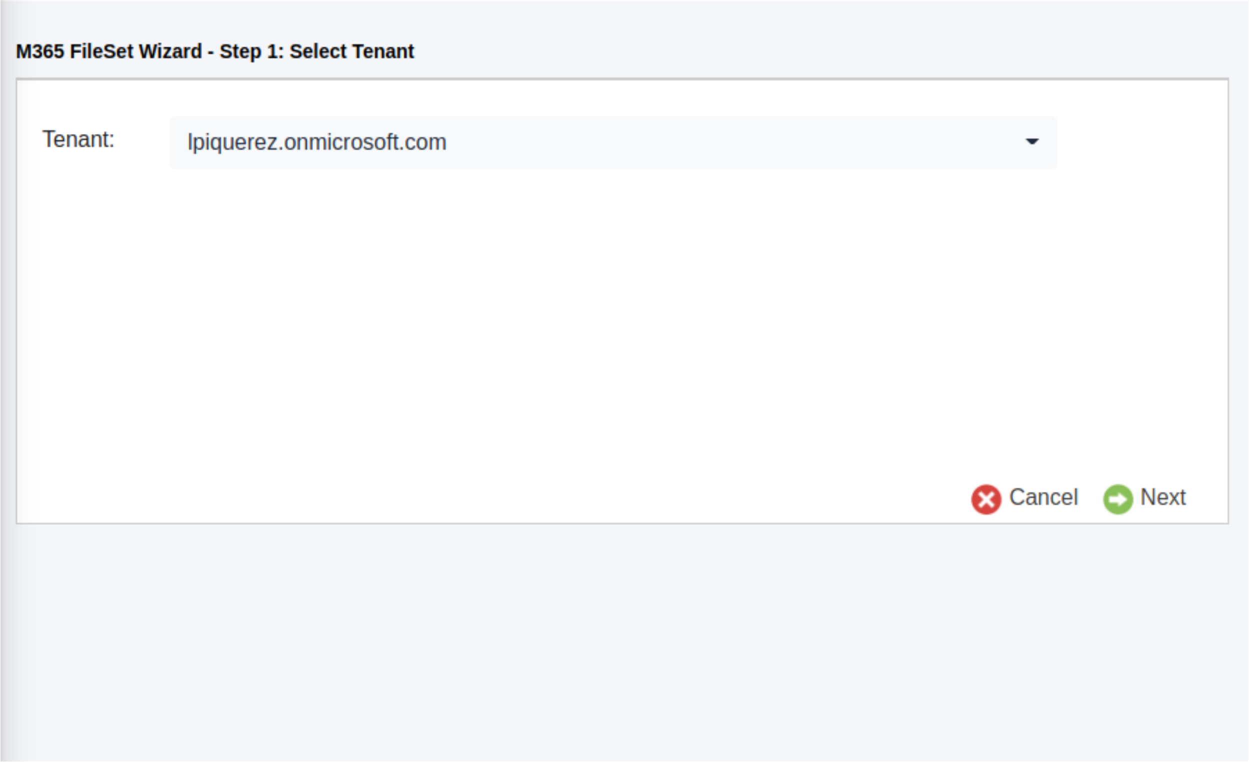
Name ▲	Service	Entities	Select
myFS1	sharepoint,contact,calendar,onenote,todo		○
test_regex	sharepoint,calendar,onenote	user_regex_include=test_reg*	○

Help

This screen is showing all defined M365 filesets in the current BWeb installation. Here is the place to launch the M365 fileset wizard where you will be able to configure a new M365 Backup Fileset in a very easy and visual way. Here you can also edit an existing fileset or remove it.

From there it is possible to create a new M365 Fileset in an user friendly way:

We need to select the tenant to work with as a first step:



M365 FileSet Wizard - Step 1: Select Tenant

Tenant: lpiquerez.onmicrosoft.com

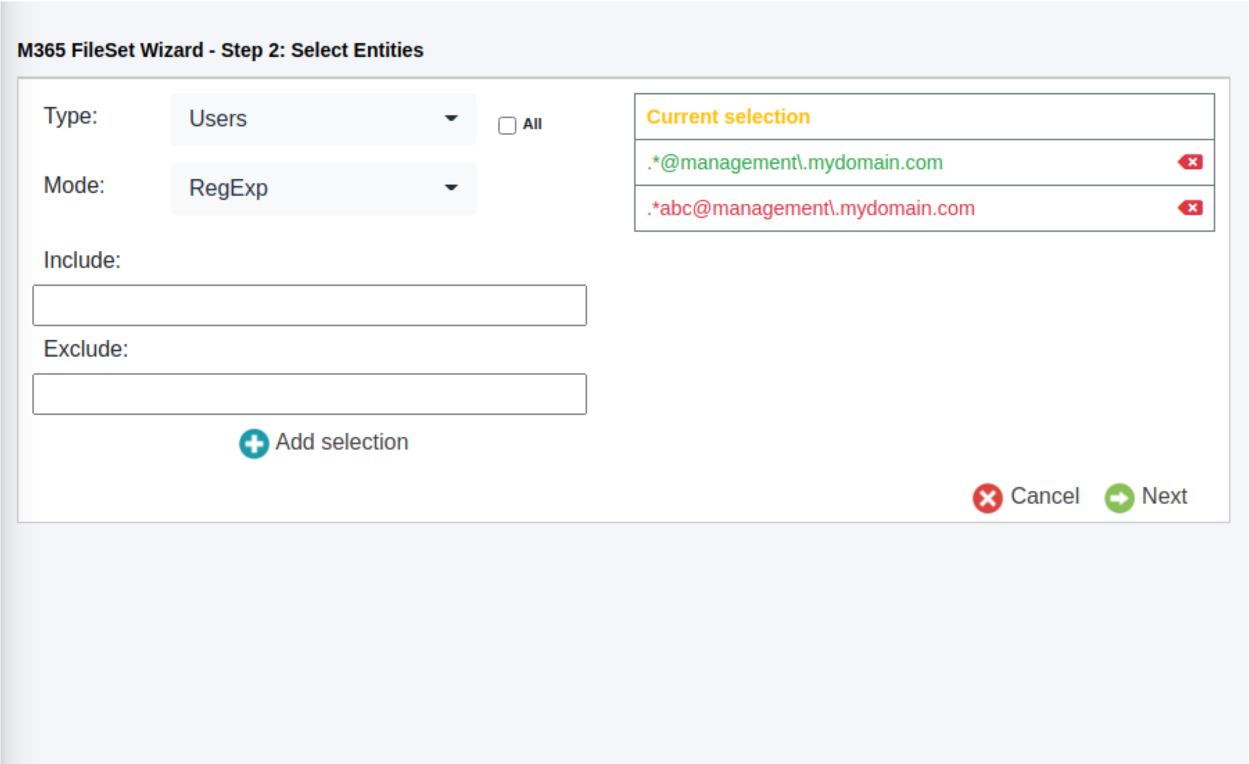
The second step is to select the entities to protect: they can be users, groups or sites. All the services of M365 are 'owned' by any of those entities:

It is possible to dinamically add or remove them, the wizard will store your final selection.

The third step allows us to select the services to protect in a similar way than the entities step, with multiselection allowed, and where we can remove or add services at will:

After selecting the services, we will have one independent screen for each selected service where we will enter the particular configuration for each of the services implied. In the screenshot below, we can see the Sharepoint services:

As a final step, it is possible to setup some advanced parameters and then finish the configuration:



M365 FileSet Wizard - Step 2: Select Entities

Type: Users

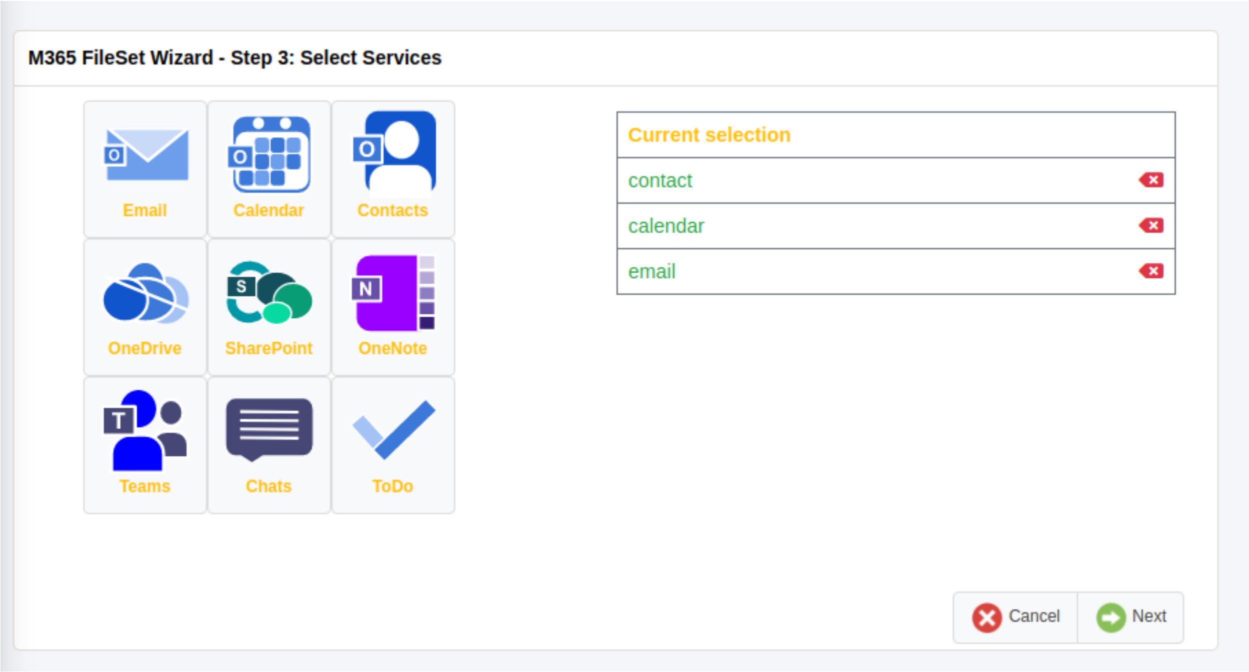
Mode: RegExp

Include:

Exclude:

Current selection

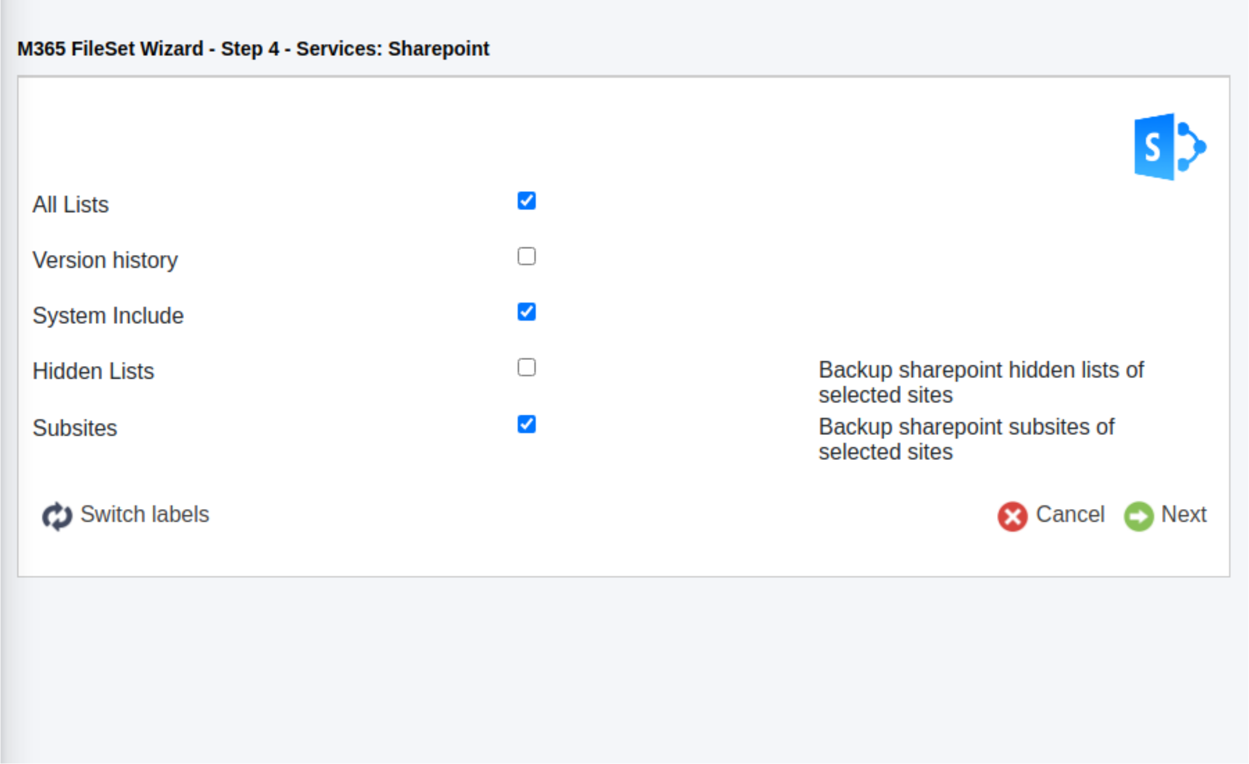
- .*@management\.mydomain.com
- .*abc@management\.mydomain.com



M365 FileSet Wizard - Step 3: Select Services

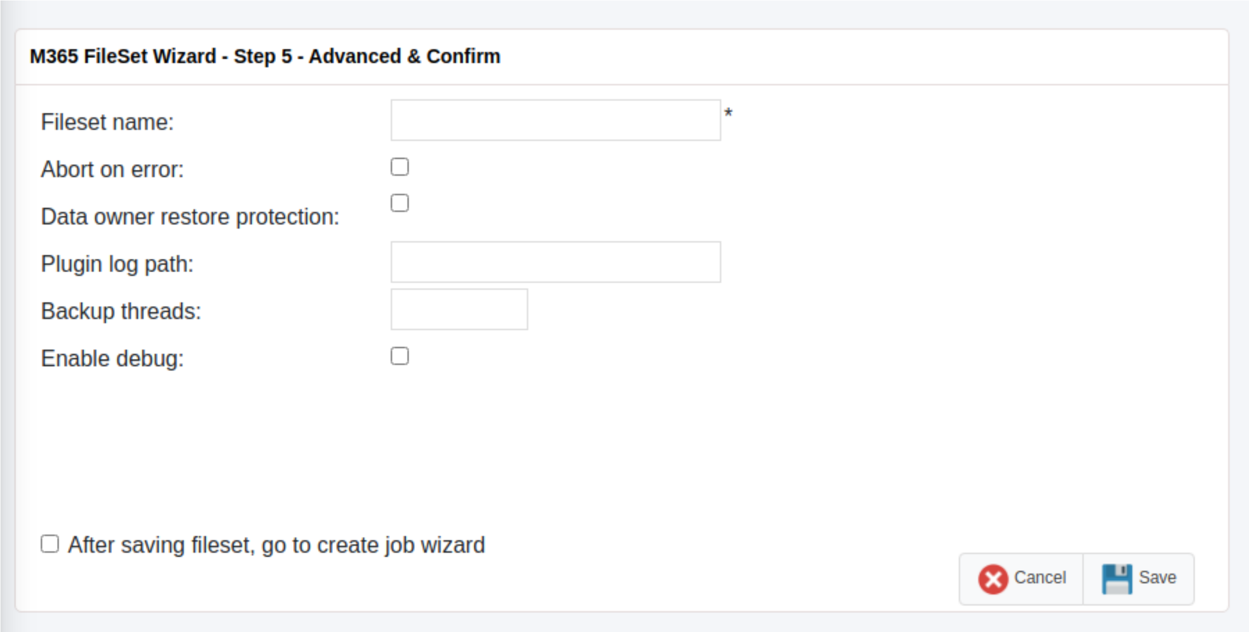
Current selection

- contact
- calendar
- email



M365 FileSet Wizard - Step 4 - Services: Sharepoint

All Lists	☑	
Version history	□	
System Include	☑	
Hidden Lists	□	Backup sharepoint hidden lists of selected sites
Subsites	☑	Backup sharepoint subsites of selected sites



M365 FileSet Wizard - Step 5 - Advanced & Confirm

Fileset name:	*
Abort on error:	□
Data owner restore protection:	□
Plugin log path:	
Backup threads:	
Enable debug:	□

□ After saving fileset, go to create job wizard

From this final screen is possible to go directly to the job configuration wizard and finish a backup configuration or to return to the previous fileset list page.

Please, note that this Wizard contains the most common parameters and it is intended to help to configure the most common use cases. In case you need to setup more advanced parameters or to edit existing filesets, BWeb will allow you to do it using the regular Fileset editor where you will find all plugin parameters.

13 Troubleshooting

Listed in this section are some scenarios that are known to cause issues.

13.1 Pwsh: The specified program requires a newer version of windows

If a Sharepoint backup shows the message referenced in the title, it means the powershell session of the job has conflicted with another open Powershell session.

Solution: Be sure no other Powershell sessions are open and try again

13.2 M365 DataPack Onedrive Hashes

If you test the plugin with the developer Datapack of M365 you may find problems with hash checking of OneDrive autogenerated files. You could see hash failures.

The reason is those auto-generated files have incorrect hashes.

Solution: Use your own uploaded files to test the plugin.

13.3 Sharepoint PnP Template restore problems

As stated in section *Details on the Sharepoint backup/restore*, templates of Sharepoint can present different problems when they are applied over new generated sites. To troubleshoot these issues, the following steps can be taken:

- Restore the site template locally
- Detect the exact part of the template that is causing problems and remove that section from the xml
- Manually apply the site template using PnP.Powershell

Suppose we are experiencing this problem and we have already restored our template in **/tmp/mytemplate.xml** and it is ready to be tested. Below we show how to run manually PnP actions:

1. The first thing to do is run Powershell. Just run the following command in your terminal:

Listing 93: **pwsh**

```
pwsh
```

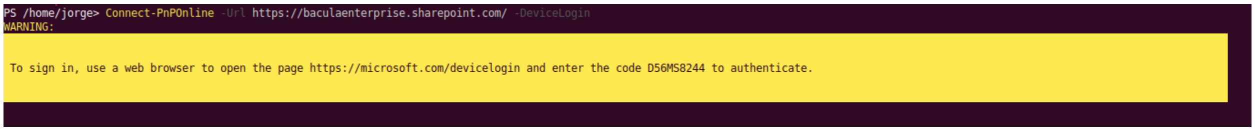
2. Import the Pnp Module:

Listing 94: **pwsh**

```
Import-Module -Name PnP.PowerShell
```

3. Connect to your main site, which is https://tenantname.sharepoint.com

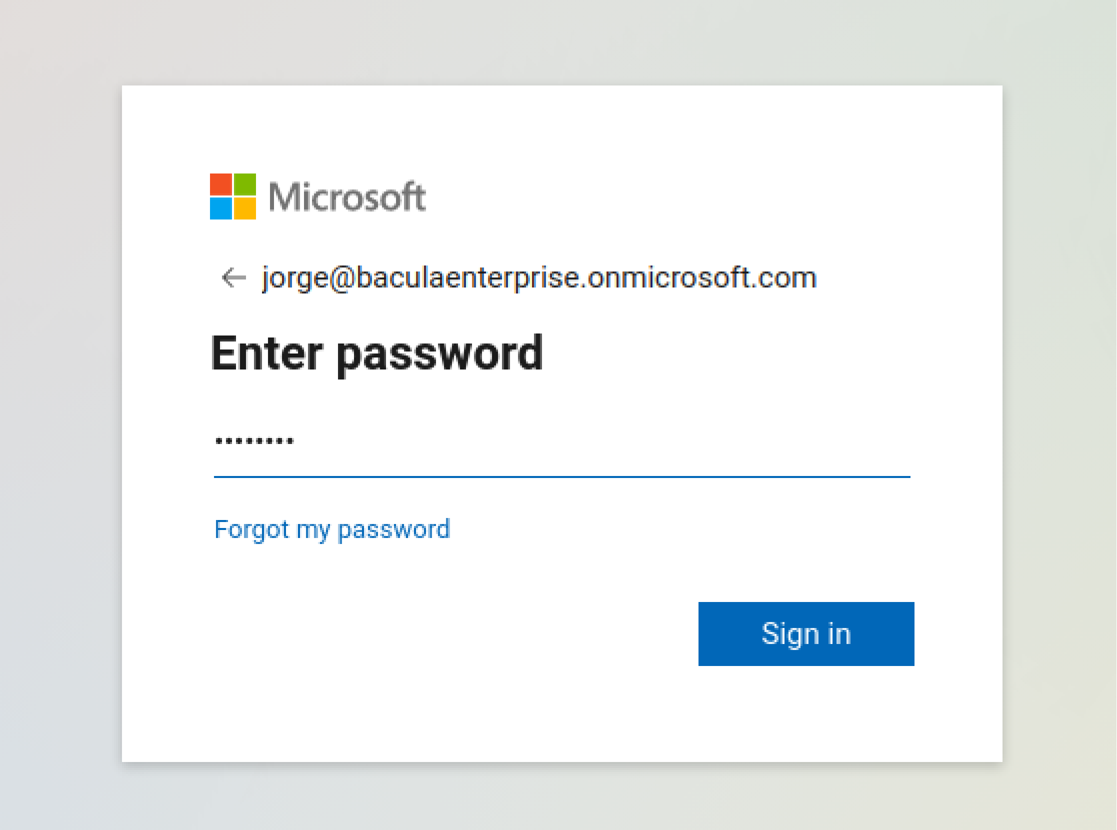
   a. We will use the DeviceLogin method, as this method allows Multifactor Authentication. Just run the command shown with your main site:



```
PS /home/jorge> Connect-PnPOnline -Url https://baculaenterprise.sharepoint.com/ -DeviceLogin
WARNING:

 To sign in, use a web browser to open the page https://microsoft.com/devicelogin and enter the code D56MS8244 to authenticate.
```

   b. Follow the instructions and go to a browser pointing to the provided URL. You will be asked for credentials and you need to login with a tenant administrator user:



jorge@baculaenterprise.onmicrosoft.com

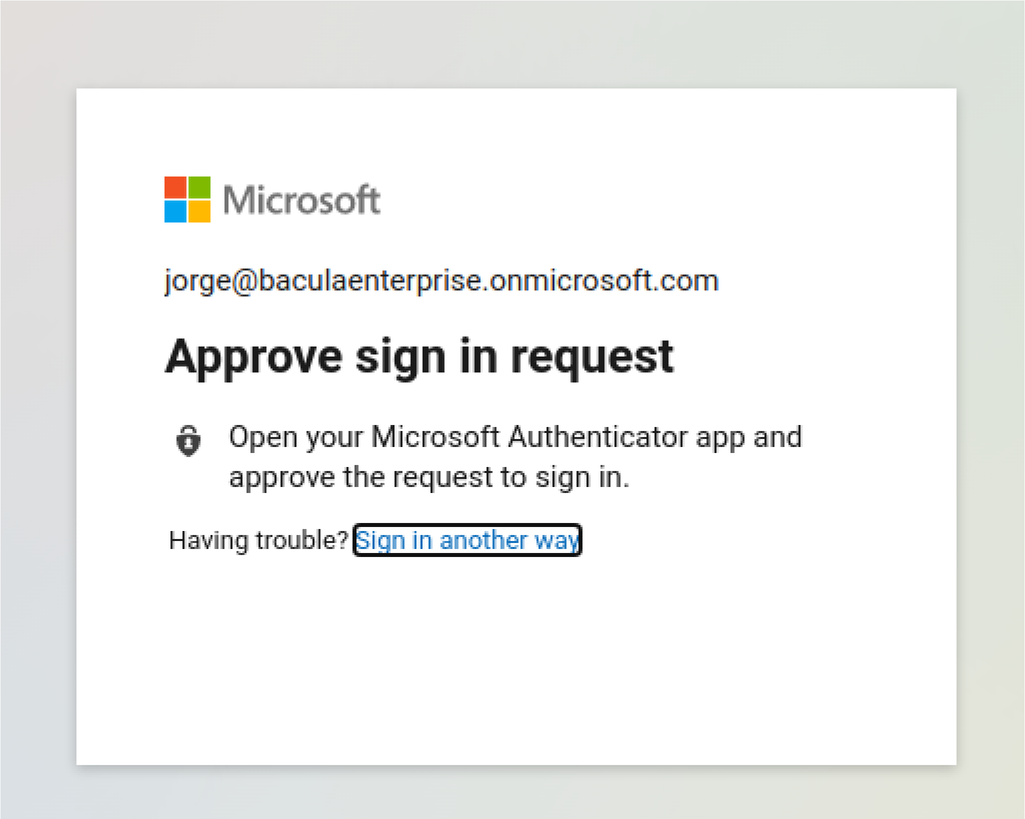
   c. If MFA is enabled, complete the process:

   d. Then put the code shown in the terminal in the first step and click Next (in this example code would be *D56MS8244*):

   e. Approve the required permissions for PnP.Powershell:

   f. Get back to the terminal and check that you are connected:

   g. You can check the correct result of the command Get-PnPSite:

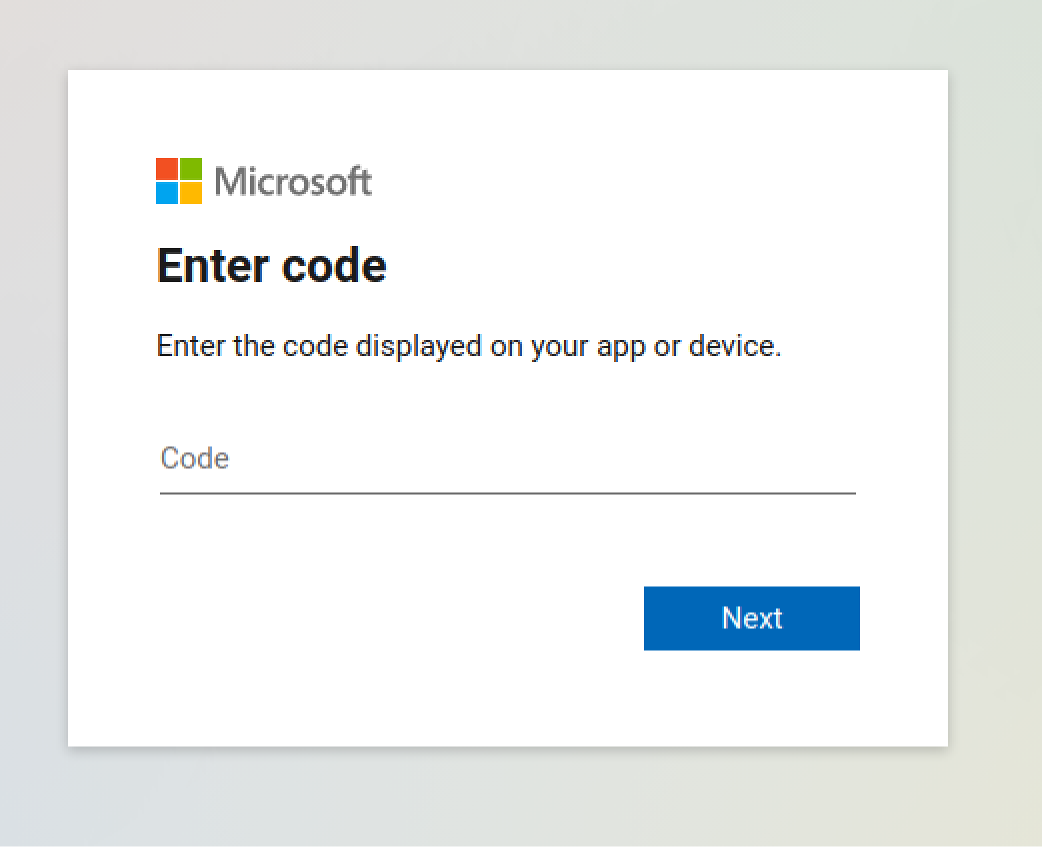


jorge@baculaenterprise.onmicrosoft.com

Approve sign in request

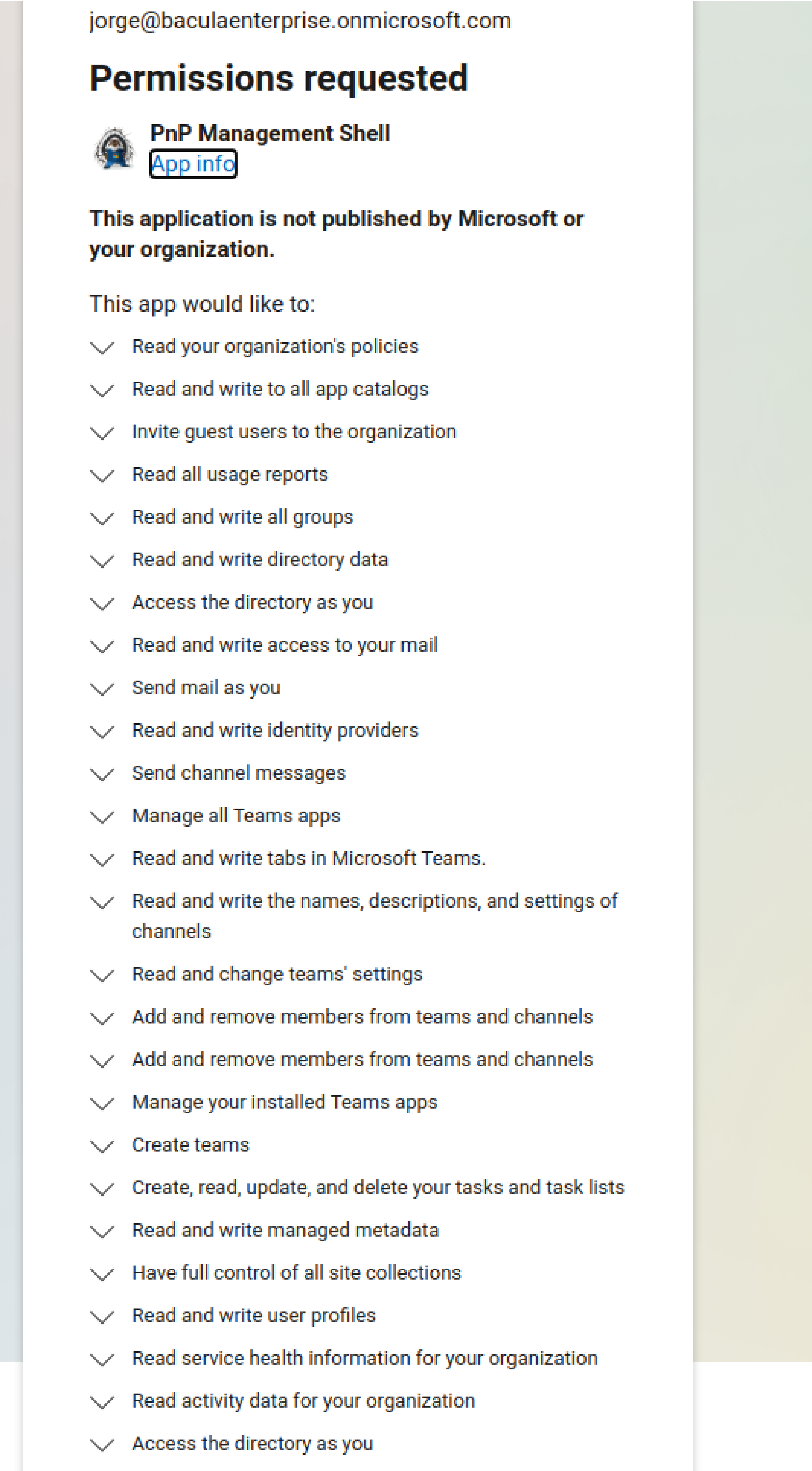
Open your Microsoft Authenticator app and approve the request to sign in.

Having trouble? Sign in another way



Enter code

Enter the code displayed on your app or device.



jorge@baculaenterprise.onmicrosoft.com

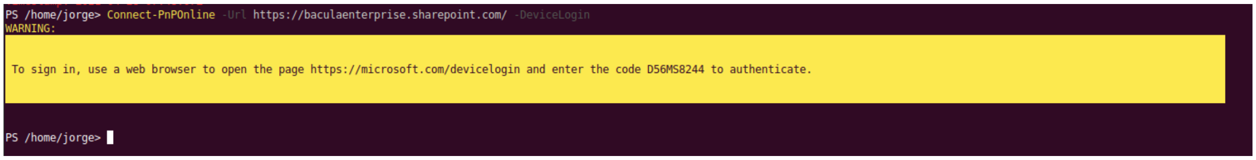
Permissions requested

PnP Management Shell

This application is not published by Microsoft or your organization.

This app would like to:

- Read your organization's policies
- Read and write to all app catalogs
- Invite guest users to the organization
- Read all usage reports
- Read and write all groups
- Read and write directory data
- Access the directory as you
- Read and write access to your mail
- Send mail as you
- Read and write identity providers
- Send channel messages
- Manage all Teams apps
- Read and write tabs in Microsoft Teams.
- Read and write the names, descriptions, and settings of channels
- Read and change teams' settings
- Add and remove members from teams and channels
- Add and remove members from teams and channels
- Manage your installed Teams apps
- Create teams
- Create, read, update, and delete your tasks and task lists
- Read and write managed metadata
- Have full control of all site collections
- Read and write user profiles
- Read service health information for your organization
- Read activity data for your organization
- Access the directory as you



```
PS /home/jorge> Connect-PnPOnline -Url https://baculaenterprise.sharepoint.com/ -DeviceLogin
WARNING:

 To sign in, use a web browser to open the page https://microsoft.com/devicelogin and enter the code D56MS8244 to authenticate.

PS /home/jorge>
```

Listing 95: **pwsh**

```
Get-PnPSite
Url CompatibilityLevel
--- ------------------           https://baculaenterprise.sharepoint.com␣
↪15
```

4. Create your new site using the following commands. Depending on the site kind, you will need to utilize one command or another.

   a. Site kind is determined by @BaseSiteTemplate value which you can find in your template

   b. Replace $var with appropriate values:

      1. newSiteName → NonExistingSiteName in your tenant
      2. newSiteUrl → https://tenantname.sharepoint.com/sites/newSiteName
      3. newSiteOwner → anyadminuser@yourtenant.com
      4. newSiteTimeZone → RegionalSettings > TimeZone attribute value from your template
      5. newSiteLocale → RegionalSettings > LocaleId attribute value from your template
      6. newSiteKind → Commented BaseSiteTemplate attribute value from your template

   c. TeamSite

Listing 96: **pwsh**

```
New-PnPSite -Type TeamSite -Title $newSiteName -Alias $newSiteName -Description "-"␣
↪-Owners
$newSiteOwner -Wait
```

   d. CommunicationSite

Listing 97: **pwsh**

```
New-PnPSite -Type CommunicationSite -Title $newSiteName -Url $newSiteUrl -Owner
↪$newSiteOwner
```

   e. Other:

Listing 98: **pwsh**

```
New-PnPTenantSite -Title $newSiteName -Url $newSiteUrl -Owner $newSiteOwner -
↪TimeZone
$newSiteTimeZone -Lcid $newSiteLocale -Template $newSiteKind
```

5. Now you need to connect to your newly created site. Therefore, repeat step 1, but using your new site URL: https://tenantname.sharepoint.com/sites/newSiteName

6. Once connected to your target tenant, apply the template with these 2 commands:

Listing 99: **pwsh**

```
Set-PnPSite -NoScriptSite:$false
Invoke-PnPSiteTemplate -Path /tmp/mytemplate.xml
```

7. Once the process ends, if everything was OK, you can disconnect with:

Listing 100: **pwsh**

```
Disconnect-PnPOnline
exit
```

13.4 Out of Memory

If you ever face *OutOfMemory* errors of the Java daemon (you will find them in the m365-debug.err file), you are very likely dealing with sub-attachments containing big files or you are using a high level of concurrency through internal concurrent_threads parameter and/or parallel jobs. To overcome this situation you can:

a) Disable the backup of multi-level attachments setting multi_attach=no

b) Reduce concurrent_threads parameter

c) Reduce the number of jobs running in parallel

d) If you cannot do that or you are not using multi-level attachments you should increase JVM memory.

To increase JVM memory that you will need to:

Create a this file: '/opt/bacula/etc/m365_backend.conf'.

Below, an example of the contents:

|M365_JVM_MIN=2G |M365_JVM_MAX=8G

Those values will define the MIN (M365_JVM_MIN) and MAX (M365_JVM_MAX) memory values assigned to the JVM Heap size. In this example we are setting 2Gb for the minimum, and 8Gb for the maximum. In general, those values should be more than enough. Please, be careful if you are running jobs in parallel, as very big values and several jobs at a time could quickly eat all the memory of your host.

The '/opt/bacula/etc/m365_backend.conf' won't be modified through package upgrades, so your memory settings will be persistent.

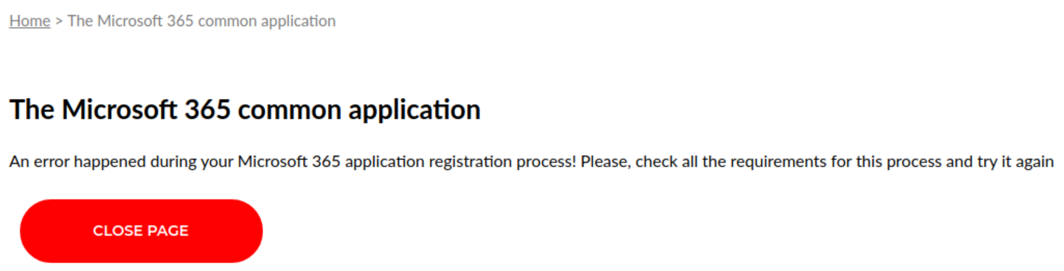
13.5 App Registration Error

When trying to register the bacula-m365-plugin app in a given tenant, the following error can appear in the URL of a resulting page placed on baculasystems.com:

Listing 101: **Registration error**

```
https://www.baculasystems.com/m365-plugin-auth/common?error=access_denied&error_
↪description=AADSTS650052%
3a+The+app+needs+access+to+a+service+(%27https%3a%2f%2f*.dps.mil%2f
↪%27)+that+your+organization+%27cd77bbc0-999e-
44c5-af9d-c11bea178407%27+has+not+subscribed+to+or+enabled.
+Contact+your+IT+Admin+to+review+the+configuration+of+your+service+subscriptions.%0d
↪%0aTrace+ID%3a+afdc0b36-
```

(continues on next page)



Home > The Microsoft 365 common application

The Microsoft 365 common application

An error happened during your Microsoft 365 application registration process! Please, check all the requirements for this process and try it again

(continued from previous page)

```
bad6-4d1f-8d24-9213439c0a01%0d%0aCorrelation+ID%3a+c73a3631-5573-41ee-8220-cc281ea4882f
↪%0d%0aTimestamp%3a+2021-
04-27+09%3a21%3a55Z&error_uri=https%3a%2f%2flogin.microsoftonline.com%2ferror%3fcode%
3d650052&admin_consent=True&state=12345
```

If we look carefully into the error contents we will notice the following message:

- *AADSTS650052*: The app needs access to a service (https://*.dps.mil.com) that your organization cd77bbc0-999e-44c5-af9d-c11bea178407 has not subscribed to or enabled. Contact your IT Admin to review the configuration of your service subscriptions.

Even if it is not a very clear message, the error is pointing us to the fact that our tenant is missing some kind of access or subscription in order to utilize bacula-m365-plugin app. The contents of the error can vary depending on the specific situation of the target tenant.

However, in general, and particularly for the example shown, the problem comes from the fact that **target tenant has no license to utilize Sharepoint Online service**. Without this license/permission, the plugin cannot be registered.

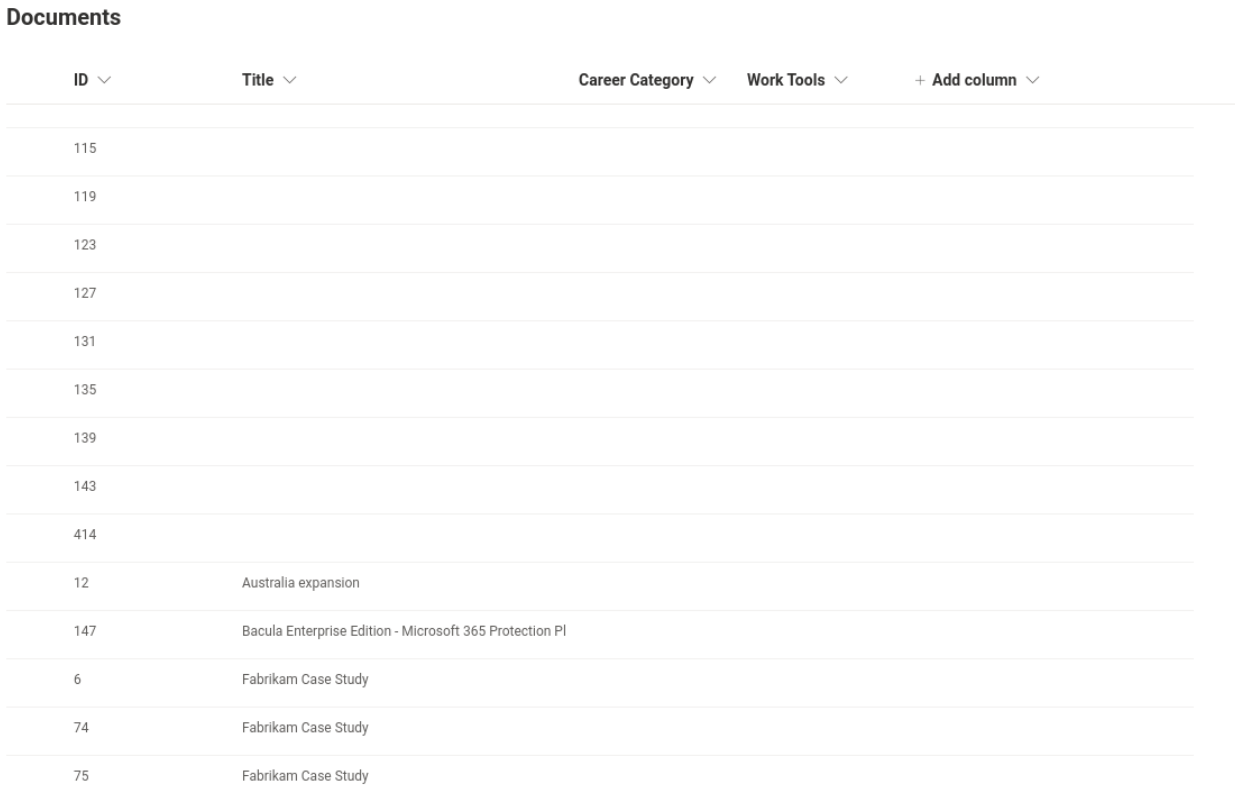
13.6 Empty Items in Group/Team Site Libraries

When working with the OneDrive module and Group entities you may experience a view of a list of apparently empty elements if navigating through them using Sharepoint views:

This is not an error of this plugin. This is a problem with some kind of views available in the Sharepoint Online service, where list items corresponding to drive items are shown. As here we are dealing with drive items, those list items have no title attached.

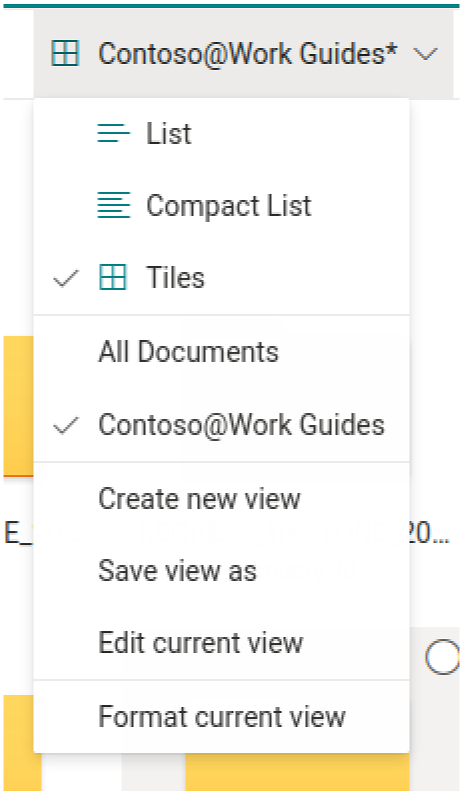
In the particular example shown above, a 'list view' of a document library is selected. However if we select any other view as 'Titles', 'All Documents', etc:

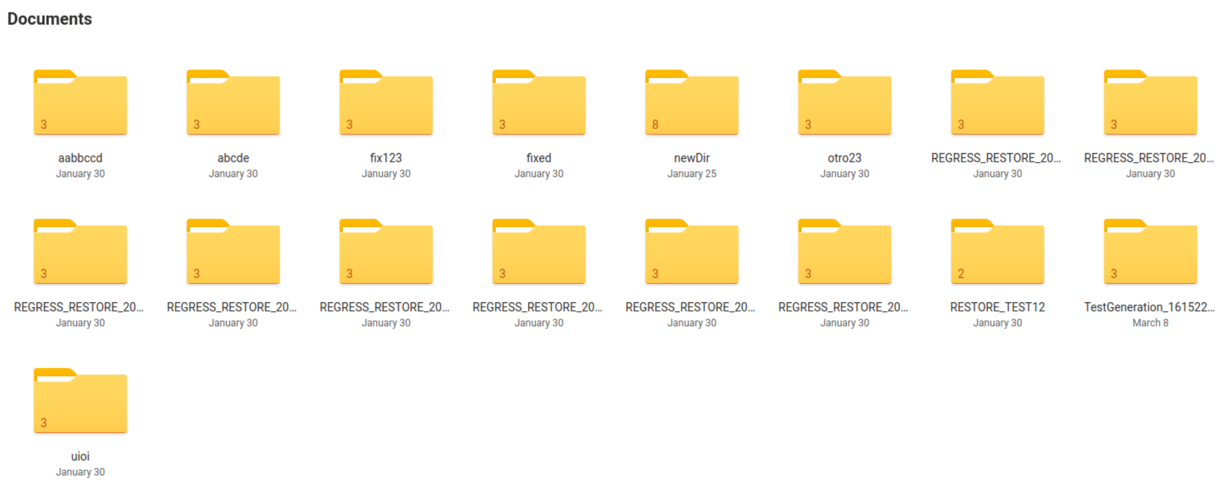
We will be able to see folders and files with their corresponding names:



Documents

ID	Title	Career Category	Work Tools	+ Add column
115				
119				
123				
127				
131				
135				
139				
143				
414				
12	Australia expansion			
147	Bacula Enterprise Edition - Microsoft 365 Protection Pl			
6	Fabrikam Case Study			
74	Fabrikam Case Study			
75	Fabrikam Case Study			





13.7 Empty Chat messages after restoring Teams

After a successful restore operation with the Teams module, it can happen that accessing the data using the web interface of the M365 services will show the associated channels with empty contents. This is usually associated to some bad functions of implied cookies with the web browsers.

If you experience this situation we recommend to open the information using a 'private window/tab' or removing previous cookies, as the information is actually there. In case this strategy does not work, please use the Teams 'desktop app' for your OS, where the behavior is much more solid.

13.8 Chat messages with not displayed contents in Teams

There are several kind of auto-generated messages that rely on other Microsoft Cloud internal data structure. Examples of this are:

- Scheduled meetings through chat actions, not associated to calendar (they belong to 'onlineMeetings' module, which is not included today in Teams backup)
- Messages showing that a Tab was added
- Messages showing contents of external apps

This kind of messages will be restored exactly with the same form that they were got during backup and text contents will be visible. However, as the information linked is attached in the external applications to the old team, that visual content may not be visible as it was in the original message.

13.9 Error restoring teams app

Teams apps are external entities controlled by an undetermined number of external provides. When adding an app to a team, a process belonging to that provider takes place. That process is invoked through the plugin restore process when restoring teams with apps. Sometimes that external process can return unexpected results.

The M365 Plugin will show those situations with a propper message in the joblog that is not treated as error, and the restore will succeed in most cases.

As this is not managed by Bacula Systems or MS Graph API, the approach is just to show the result as it is, when it is not succes
in most cases, will be also successfully restore even if that situation happens.

13.10 Error with Teams App after restoring a Planner plan

Teams desktop applications use cache data that can interfere sometimes with the real data. Restoring a planner plan can be an example. When you try to open a restored planner tab you may see an error message. However, please just click on the reload button and you should see your data without issue.