



# Exchange EWS Plugin

Bacula Systems Documentation

---

## Contents

1	Scope	2
2	Features	3
3	Architecture	4
4	Installation	6
5	Configuration	8
6	Operations	18
7	Best Practices	39
8	Limitations	41
9	Troubleshooting	42

## Contents

---

The following article aims at presenting the reader with information about the **Bacula Enterprise Exchange EWS Plugin** (Exchange Plugin based on EWS - Exchange Web Services). The document briefly describes the target technology of the plugin, defines the scope of its operations, and presents its main features.

Through subchapters, more in-depth information can be found about the following topics:

### 1 Scope

**Bacula Enterprise Exchange Plugin** currently supports the following platforms:

- Exchange Server 2019
- Exchange Server 2016
- Exchange Server 2013 with Service Pack 1 or later.

The underlying version of the Windows operative system can be any of the supported ones associated to each Exchange Server product version according to the official Microsoft documentation. As an example, at the time of writing this document, this information can be found here: <https://learn.microsoft.com/en-us/exchange/plan-and-deploy/system-requirements?view=exchserver-2019> .

This plugin is available since **Bacula Enterprise 16.2**, and needs to be deployed in a Linux host.

**See also:**

- Go to *Exchange EWS Features*
- Go to *Exchange EWS Installation*
- Go to *Exchange EWS Configuration*
- Go to *Exchange EWS Operations*

- Go to *Exchange EWS Best Practices*
- Go to *Exchange EWS Limitations*
- Go to *Exchange EWS Troubleshooting*

Go back to the *Exchange EWS plugin main page*.

## 2 Features

The main feature of **Bacula Enterprise Exchange EWS Plugin** is to offer backup and restore of Exchange Server environments at item level, which is the major possible granularity for Exchange services. This includes: emails, attachments, calendar appointments, tasks, contacts and folder structures.

In addition to the main goal, this plugin permits the user to adjust the overall functions to fit his environment offering large flexibility to select the target information to protect, to filter it because of privacy reasons or to do it efficiently through different strategies involving the parallelization of the implied operations.

### 2.1 General Features

Below, there is a list of general features this plugin offers:

- Backup and restore Exchange Server items
- Microsoft EWS API based backups
- Multi-service backup in the same job (email, calendar, contact and/or 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
- Mailbox discovery capabilities
- List/query and auto-generation capabilities if combined with ScanPlugin
- Restore objects to Exchange Server
  - To original mailbox
  - To any other mailbox
  - To a different Exchange Server (cross-server restore)
- Restore any object to filesystem
- Full, Incremental & Differential backups
  - Advanced delta function for improved performance
- Mail folder, messages, appointments, contacts, tasks and attachments granularity for backup and restore
- Email addresses and folders selection capabilities for backup
- Backup and restore MIME objects for migration purposes
- Emails indexed at item level into Bacula Catalog

- Advanced search capabilities for restore operations
- Generation of user-friendly restore report into the destination mailbox
- Privacy filters for emails:
  - Ability to exclude email message fields from the catalog
  - Exclude private or spam messages through powerful filtering capability by rules.

## 2.2 Protected Items

Below, there is a list of all the items that can be backed up and restored using this plugin:

- User mailboxes
- Shared mailboxes
- Mailbox folder structure
- In-place archiving
- Emails and associated attachments
- Calendar appointments and associated attachments
- Contacts and associated attachments
- User tasks and associated attachments.

See also:

- Go back to *Exchange EWS Scope*
- Go to *Exchange EWS Architecture*
- Go to *Exchange EWS Installation*
- Go to *Exchange EWS Configuration*
- Go to *Exchange EWS Operations*
- Go to *Exchange EWS Best Practices*
- Go to *Exchange EWS Limitations*
- Go to *Exchange EWS Troubleshooting*

Go back to the *Exchange EWS plugin main page*.

## 3 Architecture

**Bacula Enterprise Exchange EWS Plugin** is a Bacula File Daemon plugin built over the **Exchange EWS (Exchange Web Services) API** to perform all of its operations to retrieve from and feed to the target Exchange service. The plugin runs a Java Daemon which uses a custom extension of the EWS Managed API SDK originally built by Microsoft.

All the information is obtained using secure and encrypted HTTPS queries to Exchange Server from the File Daemon (and through the mentioned Java Daemon), where the plugin is installed. All the requests are performed over the following endpoint: <https://exchange.hostname/EWS/EWS/Exchange.asmx>

To get more information about EWS, visit: <https://learn.microsoft.com/en-us/exchange/client-developer/exchange-web-services/start-using-web-services-in-exchange>

The metadata of every backed up item is stored in Bacula using JSON format. If MIME option is enabled, the information is also stored with that format (RFC 2077 for emails). Any attachment associated to a given item is downloaded and stored as it is. The download process is done locally to the host, and then sent to the Bacula Storage Daemon.

Backup and restore processes use different parallelization techniques in order to maximize performance, and overcome latency times when doing each needed request to EWS. Parallelization of several backup jobs is also supported.

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

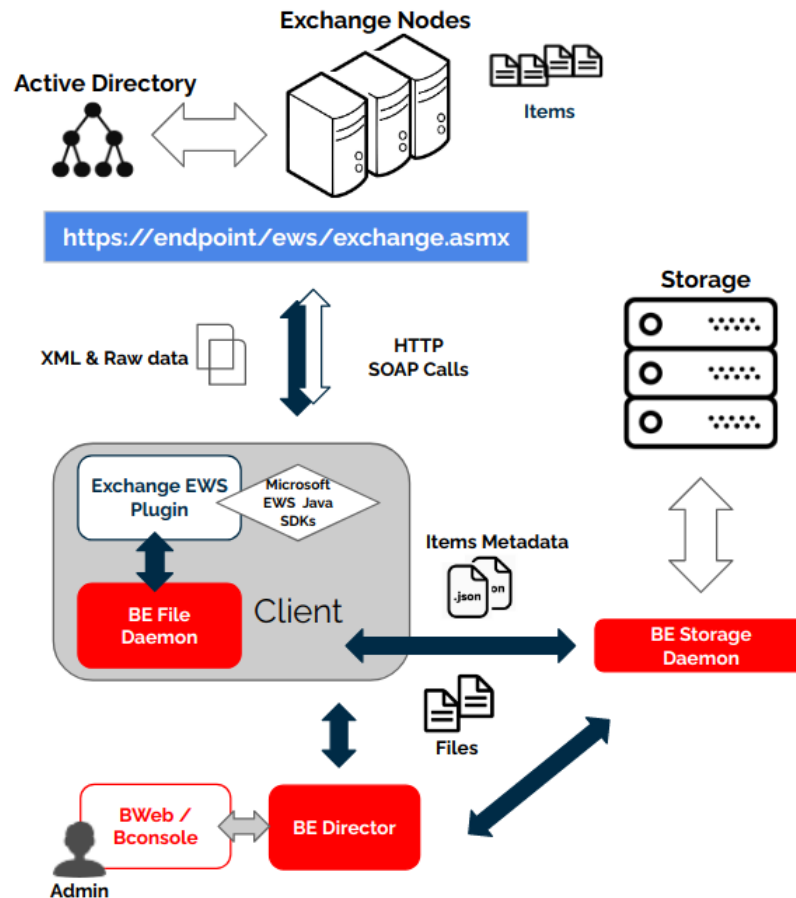


Fig. 1: Exchange EWS Plugin Architecture

**See also:**

- Go back to *Exchange EWS Scope*
- Go back to *Exchange EWS Features*
- Go to *EWS Installation*
- Go to *Exchange EWS Configuration*
- Go to *Exchange EWS Operations*
- Go to *Exchange EWS Best Practices*
- Go to *Exchange EWS Limitations*
- Go to *Exchange EWS Troubleshooting*

Go back to the [Exchange EWS plugin main page](#).

## 4 Installation

This article describes how to install Bacula Enterprise Exchange EWS Plugin.

### 4.1 Prerequisites

- The Bacula File Daemon and the Exchange EWS Plugin need to be installed on the host that is going to connect to the Exchange Server.
- The plugin is implemented over a Java layer, and even if it backs up a Windows product, it must be deployed in a host running Linux. It is possible to use any of the supported **Linux** distributions of Bacula Enterprise, including Red Hat Linux, Debian, Ubuntu or Suse Linux Enterprise Server as some examples.
- The plugin works through a Java daemon, therefore Java needs to be installed into the host through a JRE or JDK package (openjdk-8-jre for example). Installed Java environment needs to be in version 8 or above and the Java binary must be available in the system PATH.
- Memory and computation requirements completely depend on the plugin configuration and useage (parallelization, size of data to backup, 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 using up to 512Mb of RAM in demanding scenarios (usually it will be much less). In some situations this could be higher. Memory limits can be adjusted (see Out of memory).
- Exchange EWS Service is used to perform all plugin operations. Therefore, it must be up and accessible through HTTPS from the host where Bacula FD and the plugin are going to be deployed. To do this, Outlook service needs to be installed on the host where the Bacula Enterprise Exchange Plugin is going to connect to fetch and protect the data.
- In order to fetch the data, the connection to EWS is done using Basic Authentication with username and password. An administrator user (a user belonging to the 'Organization Management' group) properly configured to access the mailboxes of any target user is needed. Details about how to configure such user are given in the next sections.
- EWS endpoint is usually served through HTTPS protocol. The certificate needs to be valid and the included CN (example: myhost.com) needs to match the endpoint configured in the plugin parameters.

### 4.2 Installation Methods

- `EWSInstallationWithBIM` (recommended)
- `EWSInstallationPackageManagers`
- `EWSInstallationManual`

## 4.3 Result

The package installs the following elements:

- Jar libraries in `/opt/bacula/lib` (such as `bacula-exchange-ews-plugin-x.x.x.jar` and `bacula-exchange-ews-plugin-libs-x.x.x.jar`). 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’.
- Plugin connection file (`e2ws-fd.so`) in the plugins directory (usually `/opt/bacula/plugins`). Note that `e2ws` acronym means Exchange EWS.
- Backend file (`e2ws_backend`) that invokes the jar files in `/opt/bacula/bin`. This backend file searches for the most recent `bacula-exchange-ews-plugin-x.x.x.jar` file in order to launch it, even though usually we should have only one file.

Once the plugin is installed, it should be possible to see it loaded through a status client command in `bconsole` (‘Plugin:’ line must contain ‘e2ws’):

Listing 1: **Status client**

```
*st client
Automatically selected Client: 127.0.0.1-fd
Connecting to Client 127.0.0.1-fd at 127.0.0.1:8102

127.0.0.1-fd Version: 16.0.5 (05 April 2023) x86_64-pc-linux-gnu ubuntu 22.04
Daemon started 14-abr-23 10:14. Jobs: run=2 running=0 max=100.
Ulimits: nofile=1024 memlock=2026356736 status=ok
Heap: heap=827,392 smbytes=436,939 max_bytes=5,100,087 bufs=153 max_bufs=248
Sizes: boffset_t=8 size_t=8 debug=600 trace=1 mode=1,2010 bwlimit=0kB/s
Crypto: fips=no crypto=OpenSSL 3.0.2 15 Mar 2022
APIs: !GPFS
Plugin: bpipe(2) e2ws(1.0.0)
```

### See also:

- Go back to [Exchange EWS Scope](#)
- Go back to [Exchange EWS Features](#)
- Go back to [Exchange EWS Architecture](#)
- Go to [Exchange EWS Configuration](#)
- Go to [Exchange EWS Operations](#)
- Go to [Exchange EWS Best Practices](#)
- Go to [Exchange EWS Limitations](#)

Go back to the [Exchange EWS plugin main page](#).

## 5 Configuration

The following chapter present the information on how to configure admin user, and fileset.

### 5.1 Admin User Configuration

Bacula Enterprise Exchange EWS Plugin needs an administrator user to access to the server and to retrieve the information to back up.

This admin user needs to be able to: - Impersonate other users with Full rights (and therefore access their mailboxes)  
- Have mailbox discovery abilities.

To configure impersonation, it is necessary to run the following command in Powershell:

Listing 2: **Impersonation**

```
New-ManagementRoleAssignment -name:impersonationAssignmentName -  
↔Role:ApplicationImpersonation -User:<AdminUserName>
```

For more information about impersonation, visit: <https://learn.microsoft.com/en-us/exchange/client-developer/exchange-web-services/how-to-configure-impersonation>

In order to provide full permissions to the admin user for the impersonated mailboxes, this command needs to be run in Powershell for every user:

Listing 3: **Mailbox permissions**

```
Add-MailboxPermission -Identity <user-to-impersonate@mydomain.com> -User <admin-  
↔user@mydomain.com> -AccessRights fullaccess
```

Mailbox discover capabilities are enabled with the following command in Powershell:

Listing 4: **Discovery Management**

```
Add-RoleGroupMember -Identity "Discovery Management" -Member <AdminUserName>
```

---

**Note:** Be aware that the effect of all these commands can take time. Especially the Mailbox discovery capabilities may need more than 30 minutes to be activated.

---

---

**Important:** In addition to running this command, the Admin user needs to have his mailbox activated. Otherwise, discovery capabilities won't work.

---

**See also:**

- Go to *Fileset Configuration*

Go back to the *main configuration page*.



## 5.2 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 invoke the Exchange EWS Plugin:

Listing 5: **Fileset E2WS**

```
FileSet {
  Name = FS_E2WS
  Include {
    Options {
      signature = MD5
      ...
    }
    Plugin = "e2ws: <e2ws-parameter-1>=<e2ws-value-1> <e2ws-parameter-2>=<e2ws-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 exchange servers, in case of existing, should be using different filesets and different jobs.

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

Below, in the subsections, there are lists that present all the parameters you can use to control Exchange EWS Plugin behavior.

### Fileset Connection Parameters

The following parameters control the connection of the Exchange EWS Plugin to the Exchange Server.

Option	Required	Default	Values	Example	Description
<b>end-point</b>	Yes		A hostname or IP address	win19-cl1-exch	Hostname or IP address that matches the DN of the SSL Certificate of the Exchange service
<b>admin_domain</b>	No		A domain name	MYEX-CHANGE-DOMAIN	The users domain name. If admin_user is including already the domain, this parameter must not be set
<b>admin_user</b>	Yes		Email address or username (with or without the domain prefix)	myadmin@mydomain.com	An email address, or the username of the admin user that has permissions to impersonate all the other users. The format can be an email address, a single username (then admin_domain needs to be filled in) or domainusername. For simplicity, it is recommended to use the email address
<b>admin_password</b>	Yes		A password string	G3934kdTt0348	The password associated to the admin user

---

**Note:** The admin user must have his mailbox enabled and working. Otherwise, discovery operations will fail.

---

**See also:**

- Go to *Fileset Backup Parameters*
- Go to *Fileset Common Parameters*
- Go to *Fileset Tuning Parameters*
- Go to *Fileset Advanced Parameters*
- Go to *Fileset Examples*

Go back to the *Fileset Configuration* article.

**Fileset Backup Parameters**

The following list of parameters control what is going to be included into the associated backup:

Option	Required	Default	Values	Example	Description
<b>service</b>	No		email, contact, calendar, task, (list parameter: it can contain 0, 1 or more elements separated by ',')	email	Establish the service or services that will be backed up. If this is not set, the plugin will try to backup all supported services.
<b>user</b>	No		Valid email addresses of existing users on the selected exchange service separated by ','	AlexW@yourdomain.com LeeY@yourdomain.com	Backup mailboxes associated to this list of users. If no user is provided the plugin will use discovery mechanism and include any user with an active mailbox
<b>user_exclude</b>	No		Valid email addresses of existing users on the selected exchange service separated by ','	LauraG@yourdomain.com AmandaT@yourdomain.com	Exclude selected mailboxes. If this is the only parameter found for selection, all elements will be included and this list will be excluded
<b>user_regex_include</b>	No		Valid regex	.*@management\.\mydomain.com	Backup matching user mailboxes. Please, only provide list parameters (user + user_exclude) or regex ones. But do not try to combine them
<b>user_regex_exclude</b>	No		Valid regex	.*@guests\.\mydomain.com	Exclude matching user mailboxes. 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 user selection, all users will be included and matching users will be excluded
<b>folder</b>	No		Folder names separated by ','	inbox,company	Backup only the list of folders of this parameter from the mailboxes of the selected users. If no folder is provided, all folders will be included
<b>folder_exclude</b>	No		Folder names separated by ','	travel,personal	Exclude selected folders from the mailboxes of the selected users. If this is the only parameter found for folder selection, all folders will be included and this list will be excluded
<b>folder_regex_include</b>	No		Valid regex	.*my-company	Backup matching folders by name. Please, only provide list parameters or regex ones. But do not try to combine them
<b>folder_regex_exclude</b>	No		Valid regex	.*private	Exclude matching user folders by name. Please, only provide list parameters or regex ones. But do not try to combine them. If this is the only parameter found for folder selection, all folders will be included and matching folders will be excluded
<b>exclude_attachments</b>	No	No	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	Yes	Exclude any attachment from backup
<b>mime</b>	No	No	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	Yes	Backup raw MIME file of items, in addition to the item objects themselves (regular objects are backed up as json formatted objects)
<b>archive</b>	No	No	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	Yes	Include In-place archiving tree folders and items of the selected mailboxes
<b>email_exclude_expr</b>	No	No	String representing a valid Boolean JavaScript expression regarding email message fields	emailSubject.includes('private') && email.IsRead	Exclude from backup all messages that match the provided expression of their respective owners.
<b>email_exclude_index_strip</b>	No	No	String representing	/.*pri-	Exclude only from indexing (catalog email tables)

**See also:**

- Go back to *Fileset Connection Parameters*
- Go to *Fileset Common Parameters*
- Go to *Fileset Tuning Parameters*
- Go to *Fileset Advanced Parameters*
- Go to *Fileset Examples*

Go back to the *Fileset Configuration* article.

## Fileset Common Parameters

These parameters are controlling generic aspects of the behavior of the Exchange EWS Plugin, it is possible to find also these parameters in other Bacula Enterprise Plugins with similar effects, so you may be familiar with them.

Op-tion	Re-quired	Default	Values	Example	Description
<b>abort_on_error</b>	No	No	No, Yes	Yes	If set to <b>Yes</b> : Abort job as soon as any error is found with any element. If set to <b>No</b> : 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
<b>con-fig_file</b>	No		The path pointing to a file containing any combination of plugin parameters	/opt/bacula/etc/e2ws.conf	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
<b>log</b>	No	/opt/bacula/working/e2ws-debug.log	Working directory with enough permissions for File Daemon to create a file with the provided name	/tmp/e2ws.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.
<b>de-bug</b>	No	0	0, 1, 2, 3, 4, 5, 6, 7, 8, 9	Debug level. Greater values generate more debug information	Generates the working/e2ws/e2ws-debug.log* files containing debug information which is more complete with a greater debug number
<b>path</b>	No	/opt/bacula/working	Working path with enough permissions for File Daemon to create any internal (and usually temporary) plugin file	/mnt/my-vol/	Uses this path to store metadata and temporary files

**See also:**

- Go back to *Fileset Connection Parameters*
- Go back to *Fileset Backup Parameters*
- Go to *Fileset Tuning Parameters*
- Go to *Fileset Advanced Parameters*
- Go to *Fileset Examples*

Go back to the [Fileset Configuration](#) article.

## Fileset Tuning Parameters

These set of parameters can be used to fine-tune the behavior of the plugin to be more flexible in cases of bad network environments, or when significant job concurrency is happening, etc. It is not necessary to modify them for the great majority of the cases:

Option	Required	Default	Values	Example	Description
<b>backup_queue_size</b>	No	10	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 api concurrent requests and consequently, throttling and cpu/memory consumption for both, the FileDaemon and the Exchange server. It is only needed to modify this parameter, in general, if you are need a ver precise control of your concurrency levels
<b>concurrent_threads</b>	No	10	0-100	1	Number of maximum concurrent backup threads running in parallel in order to open data for running download actions. 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
<b>concurrent_listing_threads</b>	No	2	0-20	1	Number of maximum concurrent backup page listing threads running in parallel in order to fetch sets of data. This parameter will also affect api concurrent requests
<b>api_listing_page_size</b>	No	100	10-500	1	Number of items got in each page for multi-page requests to EWS API
<b>general_network_retries</b>	No	5	Positive integer (number of retries)	10	Number of retries for failed requests to the EWS API
<b>general_network_delay</b>	No	50	Positive integer (seconds)	100	Delay between retries to the EWS API

### See also:

- Go back to [Fileset Connection Parameters](#)
- Go back to [Fileset Backup Parameters](#)
- Go back to [Fileset Common Parameters](#)
- Go to [Fileset Advanced Parameters](#)
- Go to [Fileset Examples](#)

Go back to the [Fileset Configuration](#) article.

## Fileset Advanced Parameters

The following parameters are advanced ones, and they should not be modified in the great majority of cases:

Option	Re-quired	Default	Values	Ex-ample	Description
<b>stream_sleep</b>	No	1	Positive integer (1/10 seconds)	5	Time to sleep when reading header packets from FD and not having a full header available
<b>stream_max_wait</b>	No	120	Positive integer (seconds)	360	Max wait time for FD to answer packet requests
<b>time_max_last_modified_log</b>	No	43200	Positive integer (seconds)	43200	Maximum time to wait to overwrite a debug log that was marked as being used by other process
<b>logging_max_file_size</b>	No	50MB	String size	300MB	Maximum size of a single debug log file Generates the working/e2ws/e2ws-debug.log* files containing debut information which is more complete with a greater debug number
<b>logging_max_backup_index</b>	No	25	Positive integer (number of files)	50	Maximum number of log files to keep
<b>log_rolling_file_patterns</b>	No	%.log.%d{MM, MMM}.log.gz	String, Yes	Yes	Log patter for rotated log files
<b>split_config_file</b>	No	=	Character	:	Character to be used in config_file parameter as separator for keys and values
<b>opener_queue_timeout_secs</b>	No	3600	Positive integer (seconds)	3600	Timeout when internal object opener queue is full
<b>publisher_queue_timeout_secs</b>	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 parallel and also failed backups will generate several log files. For example: `e2ws-debug-0.log`, `e2ws-debug-1.log...`

See also:

- Go back to [Fileset Connection Parameters](#)
- Go back to [Fileset Backup Parameters](#)

- Go back to *Fileset Common Parameters*
- Go back to *Fileset Tuning Parameters*
- Go to *Fileset Examples*

Go back to the *Fileset Configuration* article.

## Fileset Examples

In this section, some fileset examples are presented:

Listing 6: **Fileset: for all data belonging to a user**

```
FileSet {
  Name = fs-e2ws-adelev
  Include {
    Options { signature = MD5 }
    Plugin = "e2ws: endpoint=myexchange.myorg.com admin_user=ex-admin@myorg.com admin_
↪password=xxxxxxx user=adelev@myorg.com"
  }
}
```

Listing 7: **Fileset: using a config file**

```
FileSet {
  Name = fs-e2ws-adelev
  Include {
    Options { signature = MD5 }
    Plugin = "e2ws: config_file=/opt/bacula/etc/e2ws.settings user=adelev@myorg.com"
  }
}
```

Config file contents is stored in the same File Daemon host in `/opt/bacula/etc/e2ws.`

```
↪settings:
endpoint=myexchange.myorg.com
admin_user=ex-admin@myorg.com
admin_password=xxxxxxx
```

Listing 8: **Fileset: Backup only emails**

```
FileSet {
  Name = fs-e2ws-adelev-email
  Include {
    Options { signature = MD5 }
    Plugin = "e2ws: config_file=/opt/bacula/etc/e2ws.settings service=email_
↪user=adelev@myorg.com"
  }
}
```

Listing 9: **Fileset: Backup emails and appointments of all users**

```
FileSet {
  Name = fs-e2ws-email-calendar
```

(continues on next page)

```

Include {
  Options { signature = MD5 }
  Plugin = "e2ws: config_file=/opt/bacula/etc/e2ws.settings service=email,calendar"
}
}

```

Listing 10: **Fileset: Backup only email folders: inbox and important custom folder**

```

FileSet {
  Name = fs-e2ws-all-inbox-important
  Include {
    Options { signature = MD5 }
    Plugin = "e2ws: config_file=/opt/bacula/etc/e2ws.settings service=email,
↪folder=inbox,important"
  }
}

```

Listing 11: **Fileset: Backup emails in mime format for two users**

```

FileSet {
  Name = fs-e2ws-mime-u1-u2
  Include {
    Options { signature = MD5 }
    Plugin = "e2ws: config_file=/opt/bacula/etc/e2ws.settings service=email mime=true,
↪user=user1@myorg.com,user2@myorg.com"
  }
}

```

Listing 12: **Fileset: Backup emails and contact in mime format for two users, but exclude attachments**

```

FileSet {
  Name = fs-e2ws-user1-user2-no-attach
  Include {
    Options { signature = MD5 }
    Plugin = "e2ws: config_file=/opt/bacula/etc/e2ws.settings service=email,contact,
↪mime=true exclude_attachments=true user=user1@myorg.com,user2@myorg.com"
  }
}

```

Listing 13: **Fileset: Backup all services from all users starting with 'org'**

```

FileSet {
  Name = fs-e2ws-org-users
  Include {
    Options { signature = MD5 }
    Plugin = "e2ws: config_file=/opt/bacula/etc/e2ws.settings user_regex_include=\\"org.
↪*\\"
  }
}

```



**Listing 14: Fileset: Backup one user reducing the concurrency configuration**

```
FileSet {
  Name = fs-e2ws-user1-min
  Include {
    Options { signature = MD5 }
    Plugin = "e2ws: config_file=/opt/bacula/etc/e2ws.settings user=user1@myorg.com
↳concurrent_threads=1 concurrent_listing_threads=1"
  }
}
```

**Listing 15: Fileset: Backup one user maximizing the concurrency configuration**

```
# Warning: This configuration could provoke throttling issues
FileSet {
  Name = fs-e2ws-user1-max
  Include {
    Options { signature = MD5 }
    Plugin = "e2ws: config_file=/opt/bacula/etc/e2ws.settings user=user1@myorg.com
↳concurrent_threads=50 concurrent_listing_threads=10 backup_queue_size=500 api_list_
↳page_size=500"
  }
}
```

**Listing 16: Fileset: Backup all services, all users, but exclude emails where the subject contains 'private'**

```
FileSet {
  Name = fs-e2ws-exclude-private
  Include {
    Options { signature = MD5 }
    Plugin = "e2ws: config_file=/opt/bacula/etc/e2ws.settings email_exclude_index_
↳expr=\"emailSubject.includes('private')\"""
  }
}
```

**See also:**

- Go back to [Fileset Connection Parameters](#)
- Go back to [Fileset Backup Parameters](#)
- Go back to [Fileset Common Parameters](#)
- Go back to [Fileset Tuning Parameters](#)
- Go back to [Fileset Advanced Parameters](#)

Go back to the [Fileset Configuration](#) article.

**See also:**

- Go back to [Admin User Configuration](#).

Go back to the [main configuration page](#).

### See also:

- Go back to *Exchange EWS Scope*
- Go back to *Exchange EWS Features*
- Go back to *Exchange EWS Architecture*
- Go back to *Exchange EWS Installation*
- Go to *Exchange EWS Operations*
- Go to *Exchange EWS Best Practices*
- Go to *Exchange EWS Limitations*
- Go to *Exchange EWS Troubleshooting*

Go back to the *Exchange EWS plugin main page*.

## 6 Operations

The following article describes details regarding backup, restore or list operations with **Bacula Enterprise Exchange EWS Plugin**.

### 6.1 Backup in Exchange EWS Plugin

Backup jobs in Exchange EWS plugin behave as any other backup job in Bacula Enterprise once the fileset has been created, as described in the configuration section. Below, some special features of the plugin that happen at backup time are described, as well as the file structure that a backup creates.

#### Backup File Structure

Items are formatted in the backup catalog in order to not include sensitive information. They are included in a path in the following format:

```
/@e2ws/domain.name/users/user@domain.name/foldername/shortId_itemDate.itemExtension
```

Depending on the type of the item, `itemExtension` will be:

- Message: `.msg`
- Appointment: `.pp`
- Task: `.task`
- Contact: `.con`
- Contact group: `.con.gr`

Mime files will have the extra word 'mime' in their extension. For example:

```
/@e2ws/testlab.local/users/ex-admin@testlab.local/regress_20230417125041/  
AAPfLdTrAAA=_r20230417-125325.mime.msg
```

Attachments will be stored together with item objects:

- They include their original name (file name)
- They have a special extension “.att”
- They include the attachment type (file or item)

- The first part of the attachment name is the name of the parent message.

Here is an example of an attachment :

```
/@e2ws/testlab.local/users/ex-admin@testlab.local/inbox/AAPfLdTuAAA=_r20230417-125329.  
msg.Prompta.gen.file.att  
/@e2ws/testlab.local/users/ex-admin@testlab.local/inbox/AAPfLdTuAAA=_r20230417-125329.msg  
→ Parent message
```

## MIME Objects Backup

Based on the fileset parameter **mime**, it is possible to get mime formatted items as well as regular objects (which are in json format). This kind of objects can be useful to get if there is any plan of using the information outside the Exchange service (e.g. for migration purposes).

---

**Note:** Activating this option has a performance penalty, and the backup time will be significantly higher, as for every email the information will be requested twice (one for the regular format, so .json message file plus attachments; another one for mime format, so to get a unified .mime.msg file containing the message and attachments).

---

At restore time, if the restore operation is done via EWS services and not to any local filesystem, selected mime objects are automatically ignored. It means if only those .mime.msg files were manually selected during a restore session over Exchange, the restore won't restore any file. While doing the same over a Local filesystem, destination will end up with those .mime.msg files restored. Usually, the selection would include both kind of files (a folder, a whole backup). In that situation, the restore will be simply successful, while those .mime.msg won't be used.

## Email Privacy Filters

Bacula Systems is aware of one of many privacy concerns that may arise when tools like the Exchange EWS 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 the plugin offers in order to deal with that problem is the possibility to exclude messages. This is a very powerful 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_exclude_expr` fileset parameter will exclude completely the selected messages
- `email_exclude_index_expr` fileset parameter will exclude the selected messages from the index (MetaEmail catalog table).

Not only messages can be excluded, but also select only a subset of email fields to be included in the indexed information using `email_fields_exclude_index` fileset parameter.

All three 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

---

**Note:** 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_index` parameter.

Then, for `email_exclude_index_expr` and `email_exclude_expr`, use them in a valid boolean expression in **Javascript** language syntax. Some examples are provided below:

Listing 17: **Expression to exclude messages where subject includes the word 'private'**

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

Listing 18: **Complex expression to exclude messages that are not read and are Draft or their folder name is named Private**

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

Listing 19: **Expression to exclude messages based on the received or sent date**

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

Listing 20: **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 as they can have doubts about the correct behavior of their prepared expressions. In order to help with that, Exchange EWS Plugin presents a query method that allows to test those expressions against a static preloaded set of data.

There are two commands available:

- Show command
- Test command

The show command will show the static data in json format, so it is possible to see the contents to adapt the expressions to test command - it will apply the expression parameters to the preloaded static data.

The test command has the following format:

Listing 21: **Expression tester Show command**

```
.query client=<your-fd-client> plugin="e2ws: endpoint=<ews-endpoint> admin_user=  
↳<username> admin_password=<password>" parameter=email-expr-show
```

The show command has the following format:

Listing 22: **Expression tester Test command**

```
.query client=<your-fd-client> plugin="e2ws: endpoint=<ews-endpoint> admin_user=  
↳<username> admin_password=<password> email_exclude_expr = \"<your-js-expression>\"  
↳parameter=json|email-expr-test  
// Or  
.query client=<your-fd-client> plugin="e2ws: endpoint=<ews-endpoint> admin_user=  
↳<username> admin_password=<password> email_exclude_index_expr = \"<your-js-expression>  
↳\" parameter=json|email-expr-test
```

---

**Note:** You need to provide a valid endpoint and user credentials, even if it's not really used to process any data.

---

The test command produces JSON output with objects with the exact format that is received from Microsoft and, consequently, the same format that is stored in backup. Note that 'total' value at the end, where the value of 12 total preloaded messages is shown.

Listing 23: **Expression tester Show command output**

```
.query client=<your-fd-client> plugin="e2ws: endpoint=<ews-endpoint> admin_user=  
↳<username> admin_password=<password>" 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, purple. You 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"  
        }  
      }  
    ]  
  }  
}
```

(continues on next page)

```
],
"from": {
  "emailAddress": {
    "address": "elon@other.com"
  }
},
"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"
    ]
  },
  {
    "total": "12"
  }
}

```

The test command on its side will produce two different outputs. The first part presents the same format as 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 24: Expression tester Test command, index part output

```

.query client=<your-fd-client> plugin="e2ws: endpoint=<ews-endpoint> admin_user=
↪<username> admin_password=<password>" parameter=json|email-expr-show
....
{
  "meta-email-12": {
    "EmailId": "",
    "EmailOwner": "test@test.com",
    "EmailTenant": "ews.test",
    "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": "e2ws"
  }
},
{
  "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, which 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 to the backup is applied, but also it is applied to the index:

Listing 25: **Expression tester Test command, index part output**

```
.query client=127.0.0.1-fd plugin="e2ws: endpoint=<ews-endpoint> admin_user=<username>
↪admin_password=<password> email_exclude_expr=\"emailFrom == 'elon@other.com'\" email_
↪exclude_index_expr=\"emailSubject.includes('private')\" parameter=json|email-expr-test
...
{
  "meta-email-4": {
    "EmailId": "",
    "EmailOwner": "test@test.com",
    "EmailTenant": "ews.test",
    "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": "e2ws"
  }
},
{
  "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 .....
```



## Delta Backup

The Microsoft EWS API provides a Delta function to track changes of some objects. Bacula Enterprise Exchange EWS Plugin uses this function in order to speed up Incremental/Differential processes.

Delta function has the following important characteristics:

- Delta tokens can expire at some point, or even become invalid due to internal Microsoft issues. If that happens, the plugin will try to start a new Delta cycle.
- There are two delta types of tokens implied: one for the folder structure, another for every folder that has changes inside.
- 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 the 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 a 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.

Here, we can see an example of the contents of the file, with one execution 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 folders or users:

Listing 26: `deltaLink`

```
{
  "deltaServices" : {
    "e2ws" : {
      "entities" : {
        "ex-admin@testlab.local" : {
          "containers" : {
            "AAMkADkwMWYyMWQwLWZjZmMtNDU3NS1iMmM3LWVmMTRkNTQ0MjVjYQAUAAAAAD2ghxkrOnXTJN4mEgVv12nAQAtUQmYk+IgrB3K
            AAPfLb1CAAA=" : {
              "deltaEntries" : [
                {
                  "date" : "Apr 14, 2023, 10:14:42 AM",
                  "delta" :
                    "H4sIAAAAAAEAGNgYGcAAotqE0tHE2NTA0ddZ3NHC10TR2djXSdnJ2ddNyMnC2cnczdLU10D2vBgveDKv0TgksSSV0fEvMSiSgYr
                    j8GaaK3+QMuKS4JSk1Mzy1JTQjJzU0nwrU9icYlnXnFJYl5yqncqKb71zS9K9SxJzS32zwtOLSpLLSLByXDfhgNxUW5iUTYk1rgYG
                    sckYG3UDOGZMfKbjuPaV2Yc5p/
                    f0MzPd1dzoXMDAy8DEwg7RwM9jViBcesLntwSAEFOUFYqB1rIwMDL6OAZ6+jn4gRQxupm5hYOVooB2I5ZD4S9H4MHAYiM2Q+OjOYm
                    Xj0sKYLgXwr5brZ7Dd7L+v6q5Sj/2soE1YfomCCo00paMRRbo2b1aYNIYr94knDYyMTA0AAAcVKZJwMAAA==" ,

```

(continues on next page)

```

      "job" : "pluginTest.2023-04-14_10.14.42_03"
    },
  ],
  "description" : "regress_20230414101303",
  "id" :
  ↪ "AAMkADkwMWYyMWQwLWZjZmMtNDU3NS1iMmM3LWVmMTRkNTQ0MjVjYQAUAAAAAD2ghxkrOnXTJN4mEgVv12nAQAtUQmYk+IgrB3K.
  ↪ AAPfLb1CAA="
    },
    "MailboxFolders" : {
      "deltaEntries" : [
        {
          "date" : "Apr 14, 2023, 10:14:42 AM",
          "delta" :
          ↪ "H4sIAAAAAAAEAk2be1ST5x3HX6HeQJCgwqzIXaVqJEhCiEBpLsSkBpRytG6oECBCLAAxAkqFeQFXT6TUuU5AgWCptlQp60qsh4qX
          ↪ h83t/veZ/
          ↪ 3uSMIowX7Fb1Bq1BKI2USpVgtV0aLpUp1pFilVqnF2vmqaLVKrlXIZJKyV1PmpZSYM1MKjYUmtdfstJQIC/
          ↪ hJbUF+lsmizxIU/
          ↪ Owyk+X1vAKzIFHn55nMhU9+lqaWSzQRURKpWKJIKIul0YpssSoiWiXWqCIj5SqFm1Ir0QpR6iJLislSbLlIkGs152abXC/
          ↪ 8/7kn5uAmCjz2MHyLX5ZksRktmbokjKkFk/7XE/pnn+Lt/
          ↪ bfHL0nSzz1C9vkYX3rny4ChBnDy+pvpqYMInF2Z277sY3un4q1Hj7B/B0/6xX/bvQqXjC5fFtU/
          ↪ fM0UQXPgpH4jyhSg/
          ↪ QXD1p6ZD1D9EBUBUIEQFVQWRIVAVChEzYComRA1C6LCIOoFiJoNUX043y9HO+IBUQboXZ4I3cuTuzQcFH+NclD8NUoIbFP6QM/
          ↪ rR1CE/DXKEeEUiJoMRYi1AJMgiJ9CBzUVop6HqGlQGY4enduyo2bcu0aMgVDFBbhRIji780dVCx0rwkQ5SUIY/
          ↪ gpo0TtNWoVRKVD1BGisiEqByp5+9h5ND/
          ↪ lDtX5IIgKhagZEDUTomZBVBhEvQBRsyFqDkTN5ad6XWzfQ9QwQjUJwPulHV0EUQaISoRaAC+IEkGUN0QBdcN0iSFqHkSFQ5QEoiIgr
          ↪ gKidEFUNUU0QtR+idkM138p/
          ↪ ryvij1VQhBkQ1QLRWCVC1F5ELUaosvQVQBBrayGqEKKIKoYouohqgGiGqGVivEQ5Q2Nvjz5KfvMF5in2GejoyDKBSon/
          ↪ tGXg+IfETko/jGbg3oyZuPfbWebun62L/9j0e6KfPuwj18nr8mTGFKfaR91wr4pT5M6TRRW/sW4vQN6UP/
          ↪ HGjij9SPKTW+W77nDrU0Q9T1HWeZpJ9VvNlVxdkUEtNB1ed30YtzTY/fNTMX6ZBTGnOnbo/fUYtzT3R/
          ↪ fa310nnfSwXfsIfaShTurrs7cV66khfC72oXE8szbCpA3qXEEvXhLSM+VUctTT04dW014ilFkum3Pcjaun6B2GH91NKHdde/
          ↪ 0e33u0XerCkXjebZz88RCm1pz+r8drPjzbQ1unLj2/8UXf/
          ↪ U36pJyt991c81j43SC1dGXNkFNCbMqSBbbrTva31LZRS+40Kdb+aqwDeKHb61pzSTi11+rve9MrfTS2t6GxP+j11+tqBQB3ns0eZD
          ↪ StQwsl9uf28P23ycWp5IOXS1+uo36iQ0bb3DNr12hDTOW4mdeX3LdnWkkabvuHH1/
          ↪ 9Q7tiAJJUaH4tNtcP8DUo6S5oh3Drp0UgcqXlX3e7fcg7QRg700JXWJsR4Ax0fu0o1+ja8Ja0W1lpT2/
          ↪ xJy7Tr3o6yfxQZf0cstf307spocTS5rr3C9bc5W+kZzCljeaCSs41lGdG2rJp++V1JmJp6/
          ↪ tuwYGf86cfxpS2Hq8+Qjvh7bp3rPxbzUvU9fTEId+BA5yD3pFDVE6l7eM8vJERRtrreqMs/Xq+B7F08EBs/
          ↪ tI3udepfzih5VT6wVd7z5A2fa590lncwqkXgFk0UxqXlDMhj30xa+T4l1NpyqDKntjSRX0aRB/
          ↪ UWYml4RdqdsX83d8Iqa069wS8zjiSEs11tZdHxJLN61ZXnX5KH/
          ↪ 6YqbU01Jxu5840h1bj+0fWEos3dXfIJsPrPYwpbWRYs6/OUMsrU/LmnvsKrG0adtR246z/E8/
          ↪ kil9643Hnb7EkXZ4nQqCqGWin79TSD1f1SHT9JSaSqx9JR+xW1JPLH0nObmh7HbiaWfful97rIOYum18tXNw70JpV8c+uJGeCV/
          ↪ 5Y9nStt18X+mrl1fNOVPfMS5w/
          ↪ ssqcG18sSxa4lHKIaxe1LE5YX8ZZrAl04bq1LTTiT6D05HldP1xBsyhoknNwSuAA516JjSs5+P7RURR+p7Xap5MJk/
          ↪ 0kVM6WbaevzPicMNzUi//eox/kiXMKU5m385uZM40uQTy9qLaac8fyaw/
          ↪ G75ke+IpSfb2fsBhbNK1jSM4cTo+4Q91GGPjdVcm0Ff6RWlnRgzJWK0gXEk4V2S5t1BFL7/
          ↪ tZ4zZxbnCPnHp1Kg2apu6roo00MahM69IDPKgmpnSLvi0A2ONjRpp4auHKawf4Iz3MkiZ3HDzdSjvmvXk+avHy4neA3Uhm1cpWJ13
          ↪ kjzWVK/zbUhKsfx5Tea0sXUQ/QNq79S8YmzlZq5CCyU2nRQDCSfgFT+obfcNlm4vQrx5+v7K/
          ↪ nj7SQkfws8ZAA6RcxpZNSQ8+WEjd9U+c2z2M05Y+0nimNGks9+G9+aQNTGhWwfKeNepdnT+9p/xJqqe2vzQHAV/
          ↪ h7s2Z8ITv9ai5uIF7nP9WiFBFu23c63rn/
          ↪ rSvPuPd0np6Ntyp9NFH1bIVxA9qwt7c76uB2YmGKe144PUQ2I5jSw+UJ00CljvY0nZznf9WYbzwX9d/
          ↪ AO8yNqGhTQAA",
        }
      ]
    }
  "job" : "pluginTest.2023-04-14_10.14.42_03"
}

```

```

        ],
        "description" : "mailboxfolders",
        "id" : "MailboxFolders"
    }
},
"jobName" : "pluginTest"
}

```

**See also:**

- Go to [Restore in Exchange EWS Plugin](#)
- Go to [List & Query](#)

Go back to the [Operations](#) article.

## 6.2 Restore in Exchange EWS Plugin

Exchange EWS plugin is able to restore to any local filesystem mounted over the host where the File Daemon is running, or to the Exchange environment. The restore method is selected based on the value of the `where` parameter at restore time:

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

When using the Exchange EWS restore method, the following parameters are available to control the restore behavior under 'Plugin Options' menu during a `bconsole` restore session:

Option	Required	Default	Values	Example	Description
<b>destination_user</b>	No		Existing email address on the target Exchange service	AlexW@DestinationUser.com	Destination User where restore data will be uploaded. If no user is set, every selected file will be restored in the original account
<b>destination_archive</b>	No		0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	yes	Restore using the in-place archiving tree instead the regular mailbox tree
<b>send_report</b>	No	1	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	0	Send an email to the destination user with a report containing the result of all restored (or failed) items
<b>foreign_container_generation</b>	No	1	0, no, No, false, FALSE, false, off ; 1, yes, Yes, TRUE, true, on	0	Generate a general folder to put inside restored items coming from different mailboxes. For example, if we restore emails from user <code>a@domain.com</code> into Mailbox of user <code>b@domain.com</code> , with this option enabled the plugin will generate an automatic folder <code>a@domain.com</code> inside the destination restore folder used over destination user <code>b@domain.com</code>
<b>end-point</b>	No	Original backup value	A hostname or IP address	win19-cl1-exch	Cross-server restore: Hostname or IP address that matches the DN of the SSL Certificate of the Destination Exchange service
<b>admin_domain</b>	No	Original backup value	A domain name	MYEX-CHANGE-DOMAIN	Cross-server restore: The users domain name. If <code>admin_user</code> is including already the domain, this parameter must not be set
<b>admin_user</b>	No	Original backup value	Email address or username (with or without the domain prefix)	myadmin@mydomain.com	Cross-server restore: An email address, or the username of the admin user that has permissions to impersonate all the other users. The format can be an email address, a single username (then <code>admin_domain</code> needs to be filled in) or domainusername. For simplicity, it is recommended to use the email address
<b>admin_password</b>	No	Original backup value	A password string	G3934kdr10348	Cross-server restore: The password associated to the admin user
<b>debug</b>	No	0	0, 1, 2, 3, 4, 5, 6, 7, 8, 9	3	Change debug level

## Restore Use Cases

The following restore scenarios are supported:

- Restore whole directories, specific items (email, task, contact, appointments 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 to the local filesystem (general restore where parameter must be set to a path).
- It is possible to control replacement behavior (items are compared using exchange id) with the generic replace option of Bacula.
- Restore to a different exchange server:
  - Restore parameters implied: `endpoint`, `admin_domain`, `admin_user`, `admin_password`.

Some particularities to remark:

- 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: 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.

## Restore Example Session

In the following restore example session, we restore into the original mailbox all the emails of the backup, inside the 'restored' folder.

---

**Note:** It is also possible to run backup or restore operations from any of the Bacula Graphical User Interfaces.

---

### Listing 27: Restore bconsole session

```
Connecting to Director 127.0.0.1:8101
1000 OK: 10002 127.0.0.1-dir Version: 16.0.5 (05 April 2023)
Enter a period to cancel a command.
*restore
Automatically selected Catalog: MyCatalog
Using Catalog "MyCatalog"

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.
```

(continues on next page)

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\_E2WS

```

+-----+-----+-----+-----+-----+-----+
| jobid | level | jobfiles | jobbytes | starttime | volumename |
+-----+-----+-----+-----+-----+-----+
| 1 | F | 32 | 10,632,924 | 2023-04-17 12:52:41 | TEST-2023-04-17:0 |
+-----+-----+-----+-----+-----+-----+

```

You have selected the following JobId: 1

Building directory tree for JobId(s) 1 ...

31 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: /

\$ mark \*

31 files marked.

\$ done

Bootstrap records written to /tmp/regress/working/127.0.0.1-dir.restore.2.bsr

The Job will require the following (\*=>InChanger):

Volume(s)	Storage(s)	SD Device(s)
TEST-2023-04-17:0	File	FileStorage

Volumes marked with "\*" are in the Autochanger.

31 files selected to be restored.

Using Catalog "MyCatalog"

```
Run Restore job
JobName:      RestoreFiles
Bootstrap:    /tmp/regress/working/127.0.0.1-dir.restore.2.bsr
Where:        /tmp/regress/tmp/bacula-restores
Replace:      Always
FileSet:      Full Set
Backup Client: 127.0.0.1-fd
Restore Client: 127.0.0.1-fd
Storage:      File
When:         2023-04-17 13:16:33
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
 12: JobId
 13: Plugin Options
Select parameter to modify (1-13): 9
Please enter the full path prefix for restore (/ for none): /
Run Restore job
JobName:      RestoreFiles
Bootstrap:    /tmp/regress/working/127.0.0.1-dir.restore.2.bsr
Where:
Replace:      Always
FileSet:      Full Set
Backup Client: 127.0.0.1-fd
Restore Client: 127.0.0.1-fd
Storage:      File
When:         2023-04-17 13:16:33
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
12: JobId
13: Plugin Options
Select parameter to modify (1-13): 13
Automatically selected : e2ws: service=email endpoint=w16-cl02-exch admin_user=ex-
↳admin@testlab.local admin_password=Bacula18 debug=4 user="ex-admin@testlab.local"
↳folder="REGRESS_20230417125041"
Plugin Restore Options
Option                               Current Value           Default Value
destination_user:                    *None*                 (*None*)
destination_path:                    *None*                 (*None*)
send_report:                         *None*                 (1)
foreign_container_generation:        *None*                 (1)
send_invitations_mode:              *None*                 (AllCopy)
endpoint:                            *None*                 (*None*)
admin_domain:                       *None*                 (*None*)
admin_user:                          *None*                 (*None*)
admin_password:                     *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 Exchange)
  3: send_report (Send report of the restore operation to the affected user)
  4: foreign_container_generation (Generate a general container (usually a folder) to
↳put inside restored objects coming from different entities)
  5: endpoint (Destination Exchange endpoint)
  6: admin_domain (Destination Exchange endpoint admin user domain)
  7: admin_user (Destination Exchange endpoint admin user)
  8: admin_password (Destination Exchange endpoint admin password)
  9: debug (Change debug level)
Select parameter to modify (1-10): 2
Please enter a value for destination_path: restore
Plugin Restore Options
Option                               Current Value           Default Value
destination_user:                    *None*                 (*None*)
destination_path:                    restore                 (*None*)
send_report:                         *None*                 (1)
foreign_container_generation:        *None*                 (1)
endpoint:                            *None*                 (*None*)
admin_domain:                       *None*                 (*None*)
admin_user:                          *None*                 (*None*)
admin_password:                     *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.2.bsr
Where:

```



(continued from previous page)

```
Replace:      Always
FileSet:     Full Set
Backup Client: 127.0.0.1-fd
Restore Client: 127.0.0.1-fd
Storage:     File
When:       2023-04-17 13:16:33
Catalog:    MyCatalog
Priority:    10
Plugin Options: User specified
OK to run? (Yes/mod/no): yes
Job queued. JobId=3
```

#### Listing 28: Restore job result

```
*llist joblog jobid=3
time: 2023-04-17 13:16:57
logtext: 127.0.0.1-dir JobId 3: Start Restore Job RestoreFiles.2023-04-17_13.16.54_11

time: 2023-04-17 13:16:57
logtext: 127.0.0.1-dir JobId 3: Restoring files from JobId(s) 1

time: 2023-04-17 13:16:57
logtext: 127.0.0.1-dir JobId 3: Connected to Storage "File" at 127.0.0.1:8103 with TLS

time: 2023-04-17 13:16:57
logtext: 127.0.0.1-dir JobId 3: Using Device "FileStorage" to read.

time: 2023-04-17 13:16:57
logtext: 127.0.0.1-dir JobId 3: Connected to Client "127.0.0.1-fd" at 127.0.0.1:8102.
↳with TLS

time: 2023-04-17 13:16:57
logtext: 127.0.0.1-fd JobId 3: Connected to Storage at 127.0.0.1:8103 with TLS

time: 2023-04-17 13:16:57
logtext: 127.0.0.1-sd JobId 3: Ready to read from volume "TEST-2023-04-17:0" on File.
↳device "FileStorage" (/tmp/regress/tmp).

time: 2023-04-17 13:16:57
logtext: 127.0.0.1-sd JobId 3: Forward spacing Volume "TEST-2023-04-17:0" to addr=260

time: 2023-04-17 13:16:57
logtext: 127.0.0.1-fd JobId 3: e2ws: Plugin log of this job available in: /tmp/regress/
↳working/e2ws/e2ws-debug-0.log

time: 2023-04-17 13:16:57
logtext: 127.0.0.1-fd JobId 3: e2ws: Backend connection to testlab.local stablished

time: 2023-04-17 13:16:57
logtext: 127.0.0.1-fd JobId 3: e2ws: Jar Version: 1.0.0 | Java version: 11.0.18

time: 2023-04-17 13:16:57
```

(continues on next page)

```
logtext: 127.0.0.1-fd JobId 3: e2ws: Starting backend restore process
time: 2023-04-17 13:16:57
logtext: 127.0.0.1-fd JobId 3: e2ws: Restore to Microsoft Exchange Service
time: 2023-04-17 13:16:57
logtext: 127.0.0.1-fd JobId 3: e2ws: No destination entity provided. Trying to restore_
↳each item into its original owner entity
time: 2023-04-17 13:16:57
logtext: 127.0.0.1-fd JobId 3: e2ws: Destination Path: restore
time: 2023-04-17 13:16:57
logtext: 127.0.0.1-fd JobId 3: e2ws: Generate report: enabled
time: 2023-04-17 13:16:57
logtext: 127.0.0.1-fd JobId 3: e2ws: Foreign container generation: enabled
time: 2023-04-17 13:16:57
logtext: 127.0.0.1-fd JobId 3: e2ws: Send invitations mode for appointments: AllCopy
time: 2023-04-17 13:17:03
logtext: 127.0.0.1-sd JobId 3: End of Volume "TEST-2023-04-17:0" at addr=10694381 on_
↳device "FileStorage" (/tmp/regress/tmp).
time: 2023-04-17 13:17:03
logtext: 127.0.0.1-sd JobId 3: Elapsed time=00:00:06, Transfer rate=1.780 M Bytes/second
time: 2023-04-17 13:17:19
logtext: 127.0.0.1-fd JobId 3: e2ws: Report sent to:ex-admin@testlab.local
time: 2023-04-17 13:17:19
logtext: 127.0.0.1-fd JobId 3: e2ws: No more items to restore. Restore ended
time: 2023-04-17 13:17:19
logtext: 127.0.0.1-dir JobId 3: Bacula 127.0.0.1-dir 16.0.5 (05Apr23):
  Build OS:                x86_64-pc-linux-gnu ubuntu 22.04
  JobId:                    3
  Job:                      RestoreFiles.2023-04-17_13.16.54_11
  Restore Client:           "127.0.0.1-fd" 16.0.5 (05Apr23) x86_64-pc-linux-gnu,ubuntu,22.
↳04
  Where:
  Replace:                  Always
  Start time:               17-abr-2023 13:16:57
  End time:                 17-abr-2023 13:17:19
  Elapsed time:             22 secs
  Files Expected:           31
  Files Restored:           31
  Bytes Restored:           10,669,371 (10.66 MB)
  Rate:                     485.0 KB/s
  FD Errors:                0
  FD termination status:    OK
```

SD termination status: OK  
Termination: Restore OK

time: 2023-04-17 13:17:19  
logtext: 127.0.0.1-dir JobId 3: Begin pruning Jobs older than 6 months .

time: 2023-04-17 13:17:19  
logtext: 127.0.0.1-dir JobId 3: No Jobs found to prune.

time: 2023-04-17 13:17:19  
logtext: 127.0.0.1-dir JobId 3: Begin pruning Files.

time: 2023-04-17 13:17:19  
logtext: 127.0.0.1-dir JobId 3: No Files found to prune.

time: 2023-04-17 13:17:19  
logtext: 127.0.0.1-dir JobId 3: End auto prune.

    jobid: 3  
    job: RestoreFiles.2023-04-17\_13.16.54\_11  
    name: RestoreFiles  
  purgedfiles: 0  
    type: R  
    level: F  
    clientid: 1  
    clientname: 127.0.0.1-fd  
    jobstatus: T  
  jobstatuslong: Completed successfully  
    schedtime: 2023-04-17 13:16:33  
    starttime: 2023-04-17 13:16:57  
    endtime: 2023-04-17 13:17:19  
    realendtime: 2023-04-17 13:17:19  
    realstarttime: 2023-04-17 13:16:57  
    jobtdate: 1,681,730,239  
  volsessionid: 3  
  volsessiontime: 1,681,728,758  
    jobfiles: 31  
    jobbytes: 10,669,371  
    readbytes: 10,629,087  
    joberrors: 0  
  jobmissingfiles: 0  
    poolid: 0  
    poolname:  
  priorjobid: 0  
  priorjob:  
  filesetid: 0  
  fileset:  
  hascache: 0  
  comment:  
  reviewed: 0  
  isvirtualfull: 0  
    rate: 485

```

compressratio: 0
  statusinfo:
writestorage:
  writedevic:
lastreadstorage: File
lastreaddevice: FileStorage

```

## 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 Exchange EWS Plugin** includes an option to generate a restore report in the user mailbox destination. 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 an Email restore session:

BEE Restore report of restore session launched on: 2023-03-30 16.58.41

infobaculaenterprise@noreply.com  
Thu 3/30, 4:59 PM  
Exchange S. Admin

☞ Reply all

### BE Exchange EWS Plugin: Restore Report

Destination entity: ex-admin@testlab.local Date: 2023/03/30 16:58:46

#	Type	Details	Path	Description	Result
1	emailmessage	moses.beck@example.com:2023-03-30 16.58.49	REGRESS_20230330165838/regress_20230330165824	EmailMessage restored successfully in destination path	SUCCESS
2	emailmessage	marina.rutledge@example.com:2023-03-30 16.58.49	REGRESS_20230330165838/regress_20230330165824	EmailMessage restored successfully in destination path	SUCCESS
3	emailmessage	priscilla.calhoun@example.com:2023-03-30 16.58.48	REGRESS_20230330165838/regress_20230330165824	EmailMessage restored successfully in destination path	SUCCESS
4	emailmessage	ulysses.cleveland@example.com:2023-03-30 16.58.48	REGRESS_20230330165838/regress_20230330165824	EmailMessage restored successfully in destination path	SUCCESS
5	emailmessage	marion.dickerson@example.com:2023-03-30 16.58.48	REGRESS_20230330165838/regress_20230330165824	EmailMessage restored successfully in destination path	SUCCESS
6	emailmessage	kaitlin.love@example.com:2023-03-30 16.58.48	REGRESS_20230330165838/regress_20230330165824	EmailMessage restored successfully in destination path	SUCCESS
7	emailmessage	rebecca.ochoa@example.com:2023-03-30 16.58.48	REGRESS_20230330165838/regress_20230330165824	EmailMessage restored successfully in destination path	SUCCESS
8	emailmessage	clinton.dawson@example.com:2023-03-30 16.58.48	REGRESS_20230330165838/regress_20230330165824	EmailMessage restored successfully in destination path	SUCCESS
9	emailmessage	dorian.herring@example.com:2023-03-30 16.58.47	REGRESS_20230330165838/regress_20230330165824	EmailMessage restored successfully in destination path	SUCCESS
10	emailmessage	rena.houston@example.com:2023-03-30 16.58.47	REGRESS_20230330165838/regress_20230330165824	EmailMessage restored successfully in destination path	SUCCESS

Summary

Restored with Error:	0
Restored with Warning:	0
Restored Successfully:	10

Copyright © 2008 - 2023 Bacula Systems™ SA — BaculaSystems. All rights reserved. Version 1.0.0

Fig. 2: Restore Email Example Report

### See also:

- Go back to *Backup in Exchange EWS Plugin*

- Go to *List & Query*

Go back to the *Operations* article.

### 6.3 List & Query

It is possible to list information using the `bconsole .ls` command and providing a path. In general, we need to provide a path representing a folder inside the user mailbox. In addition, it is also possible to list the users from a given exchange endpoint through a `.query` command.

Below, there are some examples:

List users:

Listing 29: Query example: Users

```
*.query client=127.0.0.1-fd plugin="e2ws: endpoint=xxxx admin_user=xxxxx@my.domain admin_
↪password=xxxxx" parameter=user
user=ex-admin@testlab.local
displayName=Exchange S. Admin
guid=82190796-3221-4f26-8efb-c52ffdc2c4d2
reference=/o=BaculaSystems/ou=Exchange Administrative Group (FYDIBOHF23SPDLT)/
↪cn=Recipients/cn=df20c005a2b64a3a82d431e97f12ce78-Exchange S
user=support@testlab.local
displayName=First S. User
guid=48b2b400-44ab-40e5-995c-fede3b4453c3
reference=/o=BaculaSystems/ou=Exchange Administrative Group (FYDIBOHF23SPDLT)/
↪cn=Recipients/cn=8efecafa2b724e85990b1047fca7cba3-First S. U
user=beuser@testlab.local
displayName=Backup Exec
guid=ca497f2d-5b03-4959-8dc8-78f475905a10
reference=/o=BaculaSystems/ou=Exchange Administrative Group (FYDIBOHF23SPDLT)/
↪cn=Recipients/cn=3f1b5ef6f9ef4a00bd5ea92206d7675c-Backup Exe
user=support2@testlab.local
displayName=Second D. User
guid=527e07d5-a961-4d4b-bc7a-ec00ebc09c32
reference=/o=BaculaSystems/ou=Exchange Administrative Group (FYDIBOHF23SPDLT)/
↪cn=Recipients/cn=ccc701ae568643329c87dbcaa8f0acfd-Second D.
user=CompanyMeeting@testlab.local
displayName=Bsys room
guid=8b357da2-1249-4fbb-ad27-070272ef7524
reference=/o=BaculaSystems/ou=Exchange Administrative Group (FYDIBOHF23SPDLT)/
↪cn=Recipients/cn=93b1df2aff8a4fadbf15d58b73c69999-Bsys room
```

List inbox emails:

Listing 30: List example: Inbox emails

```
*.ls client=127.0.0.1-fd plugin="e2ws: endpoint=xxxxxxxx admin_user=xxxx@mydomain.com_
↪admin_password=xxxx user=ex-admin@testlab.local service=email" path=Inbox
Connecting to Client 127.0.0.1-fd at 127.0.0.1:8102
drwxr-xr-x 1 nobody nogroup -1 1970-01-01 00:59:59 /@e2ws/testlab.
↪local/users/ex-admin@testlab.local/Inbox/
-rw-r----- 1 nobody nogroup 386774 2023-04-14 10:16:15 /@e2ws/testlab.
↪local/users/ex-admin@testlab.local/Inbox/AAPfLckyAAA=_r20230414-101615.msg
```

(continues on next page)

(continued from previous page)

```
-rw-r----- 1 nobody nogroup          381535 2023-03-30 17:04:26 /@e2ws/testlab.  
↪local/users/ex-admin@testlab.local/Inbox/AAPXSgqFAAA=_r20230330-170426.msg  
-rw-r----- 1 nobody nogroup          381535 2023-03-30 17:04:09 /@e2ws/testlab.  
↪local/users/ex-admin@testlab.local/Inbox/AAPXSgqEAAA=_r20230330-170409.msg  
-rw-r----- 1 nobody nogroup          385825 2023-03-30 17:03:22 /@e2ws/testlab.  
↪local/users/ex-admin@testlab.local/Inbox/AAPXSgqDAAA=_r20230330-170322.msg  
-rw-r----- 1 nobody nogroup          385598 2023-03-30 17:02:10 /@e2ws/testlab.  
↪local/users/ex-admin@testlab.local/Inbox/AAPXSgqCAAA=_r20230330-170210.msg  
-rw-r----- 1 nobody nogroup          386081 2023-03-30 17:00:59 /@e2ws/testlab.  
↪local/users/ex-admin@testlab.local/Inbox/AAPXSgqBAAA=_r20230330-170059.msg  
-rw-r----- 1 nobody nogroup          382465 2023-03-30 16:59:48 /@e2ws/testlab.  
↪local/users/ex-admin@testlab.local/Inbox/AAPXSgqAAAA=_r20230330-165948.msg  
-rw-r----- 1 nobody nogroup          381644 2023-03-30 16:59:05 /@e2ws/testlab.  
↪local/users/ex-admin@testlab.local/Inbox/AAPXSgp%2FAAA=_r20230330-165905.msg  
-rw-r----- 1 nobody nogroup          384487 2023-03-30 16:58:09 /@e2ws/testlab.  
↪local/users/ex-admin@testlab.local/Inbox/AAPXSgp+AAA=_r20230330-165809.msg  
-rw-r----- 1 nobody nogroup          381568 2023-03-30 16:57:44 /@e2ws/testlab.  
↪local/users/ex-admin@testlab.local/Inbox/AAPXSgp9AAA=_r20230330-165744.msg  
-rw-r----- 1 nobody nogroup          381978 2023-03-30 16:57:18 /@e2ws/testlab.  
↪local/users/ex-admin@testlab.local/Inbox/AAPXSgp8AAA=_r20230330-165718.msg  
-rw-r----- 1 nobody nogroup          381402 2023-03-30 16:55:51 /@e2ws/testlab.  
↪local/users/ex-admin@testlab.local/Inbox/AAPXSgp7AAA=_r20230330-165551.msg  
...
```

**See also:**

- Go back to [Backup in Exchange EWS Plugin](#)
- Go back to [Restore in Exchange EWS Plugin](#)

Go back to the [Operations](#) article.

**See also:**

- Go back to [Exchange EWS Scope](#)
- Go back to [Exchange EWS Features](#)
- Go back to [Exchange EWS Architecture](#)
- Go back to [Exchange EWS Installation](#)
- Go back to [Exchange EWS Configuration](#)
- Go to [Exchange EWS Best Practices](#)
- Go to [Exchange EWS Limitations](#)
- Go to [Exchange EWS Troubleshooting](#)

Go back to the [Exchange EWS plugin main page](#).

## 7 Best Practices

The following article presents best practices regarding jobs distribution, concurrency and performance.

### 7.1 Jobs Distribution

It is recommended to split the target backup between different users, or even having one job per user. This way errors in one job will not invalidate a whole backup cycle, where some users have been successful, and others experienced errors. This also makes it easier to identify the cause of the error.

**See also:**

- Go to *Concurrency*
- Go to *Performance*

Go back to the *Best Practices* article.

### 7.2 Concurrency

When using Exchange EWS APIs, it is possible to find a variety of boundaries that need to be considered. We highlight some of them below:

- Exchange EWS Throttling: <https://learn.microsoft.com/en-us/exchange/client-developer/exchange-web-services/ews-throttling-in-exchange>
- Capabilities of the host serving the Exchange Service
- Usage of the service during the backup window
- Internet Information Server (IIS) limits: <https://learn.microsoft.com/en-us/exchange/architecture/client-access/client-message-size-limits?view=exchserver-2019>

If a boundary is crossed, the corresponding request will usually fail. Bacula Exchange EWS Plugin is prepared to wait some amount of time and retry it, so it has a certain level of resiliency. However, it is crucial to plan an adequate strategy to backup all the elements without needing to reach any boundary on a regular basis. This means to control how many concurrent requests are done during the backup window.

A single job implements some parallelism which can be reduced until a point, if necessary, using the following parameters:

- `backup_queue_size` - this variable controls the size of internal queues communicating 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.
- `concurrent_threads` which controls the number of simultaneous processes fetching and downloading data. This can be reduced or increased to directly affect the concurrency level of a single job.
- `concurrent_listing_threads` controls a different pool of threads intended only to fetch information from the API. It can be reduced to 1, but increasing it over the default values won't change significantly the behavior of the plugin.

The recommended strategy to backup a new environment is to plan a step-by-step testing scenario before putting it into production, where the number of users and the concurrency of the jobs are increased progressively. Other important point is the timing schedule as some boundaries are related to time-frames (number of request per amount of time). If you detect you reach boundaries when running all your backups during a single day of the week, try to increase the time window, and spread the load through it in order to achieve better performance results.

**See also:**

- Go back to *Jobs Distribution*
- Go to *Performance*

Go back to the *Best Practices* article.

## 7.3 Performance

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

- Exchange latency and bandwidth
- Network infrastructure
- FD Host hardware
- FD Load
- Ratio number of elements/size
- And many more.

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

As a reference and regarding the number of elements and their size:

- Many little objects to protect: More objects per second, but smaller speed (MB/s).
- Big files to protect: Fewer 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`) should work well for most scenarios, however, fine-tune is possible if we define one job per user, and we control how many of them run in parallel, together to decrease the `concurrent_threads` value in order to avoid throttling or Exchange server capacity problems.

There are many possible strategies to use this plugin, so it is recommended to study what suits your needs best before deploying the jobs in your entire environment, so you can get the best possible results:

- You can have a job per user and all services.
- You can have multiple entities and only some services inside a job.
- You can split your workload through a schedule, or try to run all your jobs together.
- You can run jobs in parallel or take advantage 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 data to backup or select precisely what elements you really need inside each service (folders).
- And more.

### See also:

- Go back to *Jobs Distribution*
- Go back to *Concurrency*

Go back to the *Best Practices* article.

### See also:

- Go back to *Exchange EWS Scope*
- Go back to *Exchange EWS Features*



- Go back to [Exchange EWS Architecture](#)
- Go back to [Exchange EWS Installation](#)
- Go back to [Exchange EWS Configuration](#)
- Go back to [Exchange EWS Operations](#)
- Go to [Exchange EWS Limitations](#)
- Go to [Exchange EWS Troubleshooting](#)

Go back to the [Exchange EWS plugin main page](#).

## 8 Limitations

The following article presents limitations of Exchange EWS Plugin.

- The `restart` command has limitations with plugins, as it initiates the Job from scratch rather than continuing it. Bacula determines whether a Job is restarted or continued, but using the `restart` command will result in a new Job.

More detailed limitations:

### 8.1 Protection Scope

Only the items that are listed in the [features' section](#) of this document are backed up. This means that backup with this plugin **does not include** elements such as the mailbox configuration, mailbox rules or any other database element outside the items the users can directly work with.

If you are interested in including also those elements into your backup strategy, consider the combination of this plugin with the Bacula Enterprise Exchange VSS Plugin, that works at database level.

Legacy public folders, when not connected to a shared and accessible mailbox are not supported. They will be supported in future versions of this plugin.

#### See also:

- Go to [Backup of Attachments and Files](#)
- Go to [Empty Files](#)

Go back to the main [Limitations](#) article.

### 8.2 Backup of Attachments and Files

In general, this plugin backups two types of information:

- Objects
- Files.

Objects are elements representing some item in Exchange such as a calendar event, a contact, an email, etc., while files are attachments of those items.

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 perform metadata checks and to improve overall performance, as this the 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 backend script (`e2ws_backend`) 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 you run in parallel.

**See also:**

- Go back to *Protection Scope*
- Go to *Empty Files*

Go back to the main *Limitations* article.

## 8.3 Empty Files

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

**See also:**

- Go back to *Protection Scope*
- Go back to *Backup of Attachments and Files*

Go back to the main *Limitations* article.

**See also:**

- Go back to *Exchange EWS Scope*
- Go back to *Exchange EWS Features*
- Go back to *Exchange EWS Architecture*
- Go back to *Exchange EWS Installation*
- Go back to *Exchange EWS Configuration*
- Go back to *Exchange EWS Operations*
- Go back to *Exchange EWS Best Practices*
- Go to *Exchange EWS Troubleshooting*

Go back to the *Exchange EWS plugin main page*.

## 9 Troubleshooting

In this article, there are suggested solutions to common situations that can cause trouble during the usage of the Exchange EWS plugin.

## 9.1 Certificate Problem

The certificate associated to the configured endpoint should be a valid one, or the plugin will reject to connect to it for security reasons.

The Common Name (CN) of the certificate should match the hostname used in the endpoint variable. Otherwise, the plugin will raise errors like: “The request failed. The request failed. Host name ‘10.10.10.99’ does not match the certificate subject provided by the peer”.

If the certificate CN uses a hostname, we need to use that hostname from the plugin configuration, instead of the IP address. If needed, because the DNS Server used by the File Daemon host cannot resolve that hostname, the IP-hostname association should be added to the local `/etc/hostname` file (or equivalent local hostname configuration, depending on the Operative System).

On the other hand, if the certificate is not valid but it is needed to tell the plugin to trust it, the procedure is to add it to the local keystore of the Java Virtual Machine running on the File Daemon. An example of how to do it is given below:

Listing 31: **Throttling unlimited**

```
keytool -cacerts -storepass changeit -importcert -alias ewscert -file certificate.cer
```

Note that:

- keytool should be available in the PATH of the system. If it’s not, you should look for it inside the Java JRE installation path (bin directory).
- changeit is the default password of the Java keystore. You should change it.
- certificate.cer is the file containing the Exchange certificate to import. You should download it, for instance, using a browser, and connect to Outlook.

**See also:**

- Go to [Out of Memory](#)
- Go to [Throttling](#)

Go back to the main [Troubleshooting](#) article.

## 9.2 Out of Memory

If you ever face *OutOfMemory* errors of the Java daemon (you will find them in the `e2ws-debug.err` file), you are very likely using a high level of concurrency through `internal_concurrent_threads` parameter and/or parallel jobs.

To overcome this situation you can:

- a) Reduce `concurrent_threads` parameter.
- b) Reduce the number of jobs running in parallel.
- c) If you cannot do that, you should increase JVM memory.

To increase JVM memory, you will need to:

Create this file: `‘/opt/bacula/etc/e2ws_backend.conf’`.

Below, an example of the contents: `E2WS_JVM_MIN=2G E2WS_JVM_MAX=8G`

Those values will define the MIN (`E2WS_JVM_MIN`) and MAX (`E2WS_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. 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/e2ws_backend.conf` won't be modified through package upgrades, so your memory settings will be persistent.

**See also:**

- Go back to [Certificate Problem](#)
- Go to [Throttling](#)

Go back to the main [Troubleshooting](#) article.

## 9.3 Throttling

It is possible to manage Exchange throttling policies, and increase them if it detected a high number of requests rejected while doing backup jobs. Below, there is an example of how to configure an unlimited throttling policy for a given account:

Listing 32: **Throttling unlimited**

```
New-ThrottlingPolicy BaculaNoThrottling
Set-ThrottlingPolicy BaculaNoThrottling -RCAMaxConcurrency unlimited -RcaMaxBurst_
↪unlimited -RcaRechargeRate unlimited -RcaCutoffBalance unlimited
Set-Mailbox <user or service account> -ThrottlingPolicy BaculaNoThrottling
```

**See also:**

- Go back to [Certificate Problem](#)
- Go back to [Out of Memory](#)

Go back to the main [Troubleshooting](#) article.

**See also:**

- Go back to [Exchange EWS Scope](#)
- Go back to [Exchange EWS Features](#)
- Go back to [Exchange EWS Architecture](#)
- Go back to [Exchange EWS Installation](#)
- Go back to [Exchange EWS Configuration](#)
- Go back to [Exchange EWS Operations](#)
- Go back to [Exchange EWS Best Practices](#)
- Go back to [Exchange EWS Limitations](#)

Go back to the [Exchange EWS plugin main page](#).