SOMEBYTES
www.Somebytes.de

EEAttachments

Documentation

Sven Ilius

24.01.2025

EEAttachments – Documentation

Table of Contents

Functional Overview

EEAttachments is a service which can export attachments from Exchange mailboxes in order to use the data in your own workflow.

In order to access the Exchange server, Exchange Webservice (EWS) is used, the Exchange versions of which are available as of 2007 SP1.

The service can run on the server on which Exchange is installed but, as EWS is used, it can also run on any other computer. The advantage of this is that Hosted-Exchange can also be used for synchronization.

With EEAttachments it is possible for you to include several Exchange servers in the configuration as well as configure several synchronization jobs.

In order to use the program, Exchange as of version 2007 SP1 is required.

EEAttachments – Documentation

Possible User Scenarios

Your mailbox contains mails with xml ad attachments. Your idea is to store the xml data inside a database. So use EEAttachments to extract the attachments from the folder your mails are stored and further process it with your own batch-files, powershell or import-program in order to get it into the database.

Configuration

After successful installation, you can adjust the settings in the configuration program under START=>Programs=>EEAttachments.

With the configuration you introduce the Exchange-Server connection to the service as well as the folders you want to be able to export attachments for.

Service Configuration

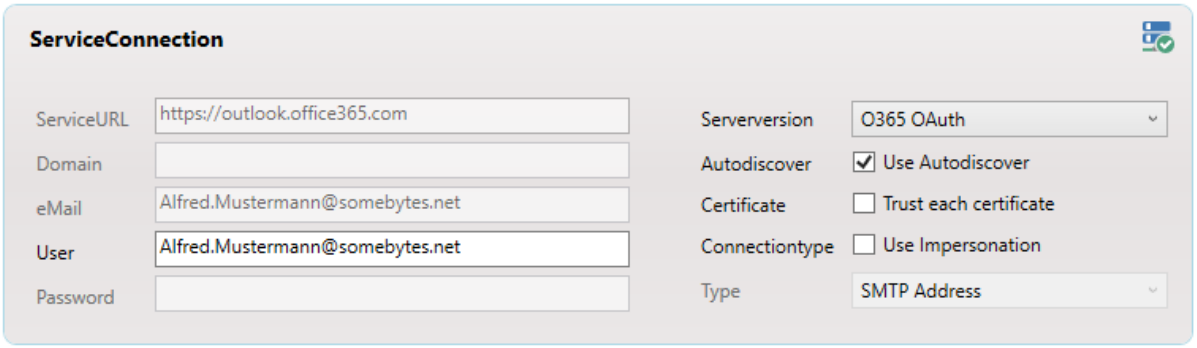
In the menu, under [Serviceconnection], you can set up and delete Exchange web service, Microsoft Graph, Google or CalDAV connections.

Create EWS access to Exchange or Microsoft 365

Create EWS-Access: Via the menu item [Serviceconnection] => [Create EWS access], you will get a new entry in the service list. This allows you to connect to the desired Exchange server. The EWS name is a freely definable name, it is used to identify your connections within the synchronization tasks.

In the simplest case, the connection can be established via "Autodiscover". All you have to do is enter your user’s name, password and email address. All information that is necessary for the connection is retrieved from the server.

If you are using Office 365, then it is possible to select “Microsoft 365” for the version and then the “Connect” button. You will be redirected to your Microsoft 365 registration, where you can enter the appropriate username and password.



If the options mentioned above fail, you have to configure the connection manually, which means that you also have to know and enter the Service URL. To do this, remove the tick from "Autodiscover" and enter the relevant information.

The following information is required for manual configuration:

1. Exchange Webservice URL
2. Username
3. Password
4. Optionally, you can also enter a proxy if you need to communicate via proxy.

Now enter the service URL of your Exchange server: e.g. https: //YOUR.EXCHANGE.SERVER If you call up the web service URL (https: // YOUR.EXCHANGE.SERVER/EWS/Exchange.asmx) as a test via browser, an XML file should be displayed after authentication. To do this, open the context menu of the edit field and select "Test URL".

Now enter a user account that has access to the data you want.

If your certificate contains errors, you have the option to use the checkbox at "Trust each certificate". However, this can result in you violating your company's security guidelines.

The following parameters can be used for the service connection:

Impersonation: If you would like to access data that lies in a mailbox different from yours, you can also use „Impersonation“, i.e. the right to access another person’s data. (Overtaking of another user’s identity)

Impersonation is also helpful if you use a password policy that requires users to change their password monthly. Otherwise you would have to keep track of this in EEAttachments. To be able to use impersonation, you need a user with a mailbox that acts as a service user for EEAttachments. Furthermore, this service user must be authorized accordingly.

Configuration of impersonation in Exchange:

- **Exchange 2007**, per Exchange Management Shell: In order for a user to be able to do this, the right **ms-Exch-EPI-Impersonation** has to be set up for this user on the server. In addition, the mailbox you would like to access needs to have the right: **ms-exch-epi-may-impersonate**.

- **Exchange 2010 and up**:

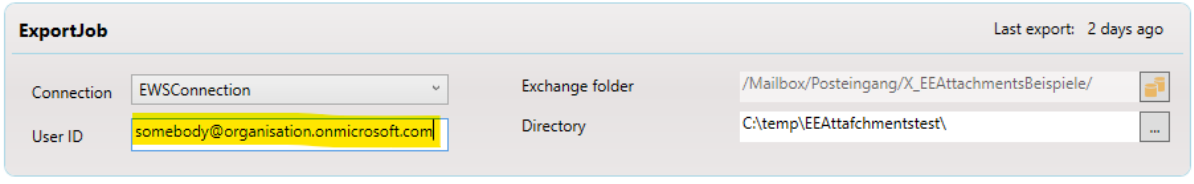
    - **Via Admin-Center:** In Exchange Admin Center you find Permissions=>Administrative Roles, you will find a role named "ApplicationImpersonation" which allows administrators to adopt the identity of the selected users. Do these steps in order to have access to user mailboxes:
        1. Create a new administrative role group.
        2. Enter name and add "ApplicationImpersonation" as a role.
        3. Add the service user, who will have access to the user mailboxes.

    - **Via Exchange Management Shell:** Since Exchange 2010 it's done with "Application Impersonation Management Role". To do this, the user to be accessed must be listed in a so-called ManagementScope (similar to a group). Assuming that you want a user named "ServiceAcc" to be able to access the data of the user "Testuser", you would have to enter the following in the Management Shell:
        - `new-managementscope -name:Impersonation4EEAttachments -RecipientRestrictionFilter:"Name -eq 'Testuser'"`
        - `new-ManagementRoleAssignment -Name:ImpersonationAssignment4EEAttachments -Role:ApplicationImpersonation -User:' ServiceAcc @somebytes.local' -CustomRecipientWriteScope: Impersonation4EEAttachments`

      The commands create a scope in which the test user is located. A role is assigned to the scope, which allows the user ServiceAcc to access the mailboxes of the users who are in the scope.

      EEAttachments provides you with a pre-configured configuration (usable in Exchange 2010 and up) in the context menu of the corresponding service connection after the configuration of the web service connections and the synchronization jobs (entry of the user addresses for User ID). Call up "Configuration help for impersonation".

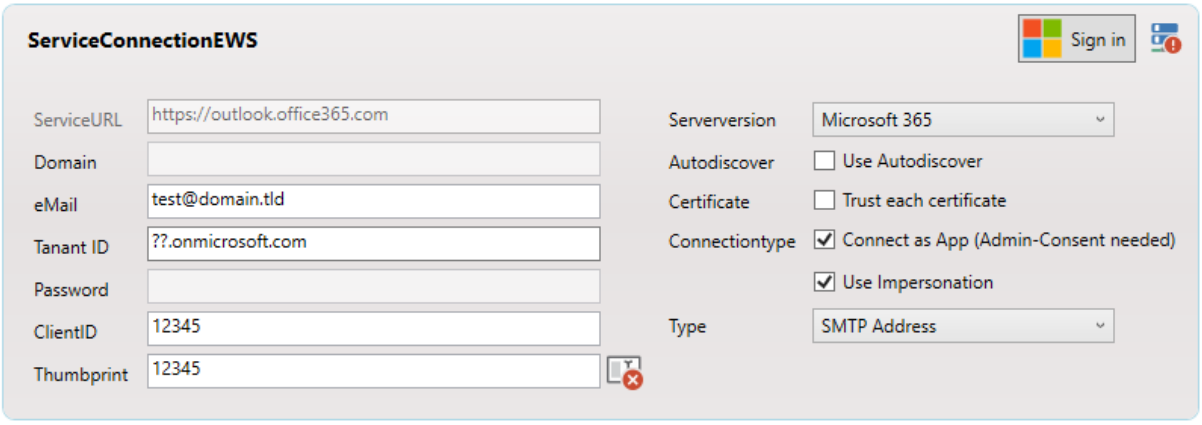
In order to be able to access the desired mailbox of a user with the service user via "impersonation", the corresponding SMTP address of the user must be entered in the synchronization job.



EEAttachments – Documentation

Server version: In order to be able to use the right functions, the service needs to know which version is used by your Exchange server.

Access as an application: With Microsoft 365 it is also possible to access your mailboxes via application and EWS.



For this purpose, it is necessary to add a new registry in Azure portal => App registries:

1. New registration => "Only accounts in this organization directory (single client)".
   Redirect URI "Public Client/Native (mobile and desktop)"
   https://www.somebytes.com/EEAttachments-authorizeready

2. API Permissions:

   - (Office 365 Exchange Online) full_access_as_app.

3. Generate yourself a certificate in the client and add it to "Certificates & Secrets".

   Here is an example of how they can generate the certificate:

   $Cert = New-SelfSignedCertificate -CertStoreLocation "Cert:\LocalMachine\My" -FriendlyName "EWS4EEAttachments" -NotAfter (Get-Date).AddYears(5) -KeyExportPolicy Exportable -Subject "Zertifikat für EEAttachments - EWS"

   Get-ChildItem "Cert:\LocalMachine\My\$($Cert.thumbprint)" | Export-Certificate -FilePath C:\temp\EWSEEAttachments.cer

   $Cert.thumbprint

Alternatively, you can have EEAttachments generate the Powershell script by clicking on the service connection panel on the right and selecting the corresponding menu item.

Enter the clientID, applicationID and the fingerprint of the certificate in the corresponding fields in the service connection at EEAttachments.

4. Optionally, you can specify an existing mailbox in the E-mail field so that the service can check whether the connection is working without an export task.
5. After clicking on "Connect" the administrator approval page appears with which you can allow the application as a company application.

You can find the client ID in the Azure AD Admin Center, under Properties. (Either a GUID or the domain e.g. contoso.onmicrosoft.com).

Microsoft Graph Connection

To add a Microsoft Graph connection to your Microsoft 365 account, follow these steps:

In the Service Connection menu, click on "Microsoft Graph Connection" Click on "Connect" and provide the corresponding account details.

Furthermore, there is the possibility to access the mailboxes of your entire client as an app. For this it is necessary to add a new registration in Azure portal => App registrations:

1.) New registration => "Only accounts in this organization directory (single client)".

2.) Redirect URI "Public Client/Native (mobile and desktop)"
https://www.somebytes.com/eeattachments-authorizeready

3.) API Permissions:

- Mail.Read
- Mail.ReadWrite
- User.Read.All
- User.Read
- Mail.Send

4.) Generate a certificate in the client and add it to "Certificates & Secrets".

Here is an example how to generate the certificate:

$Cert = New-SelfSignedCertificate -DnsName domain,tld -CertStoreLocation "Cert:\LocalMachine\My" -FriendlyName "MicrosoftGraph4EEAttachments" -NotAfter (Get-Date).AddYears(5) -KeyExportPolicy Exportable -Subject "MicrosoftGraph Certificate for EEAttachments"

Get-ChildItem "Cert:\LocalMachine\My\$($Cert.thumbprint)" | Export-Certificate -FilePath C:\App-Certificate-EEAttachments-MS-Graph\msgraphEEAttachments.cer

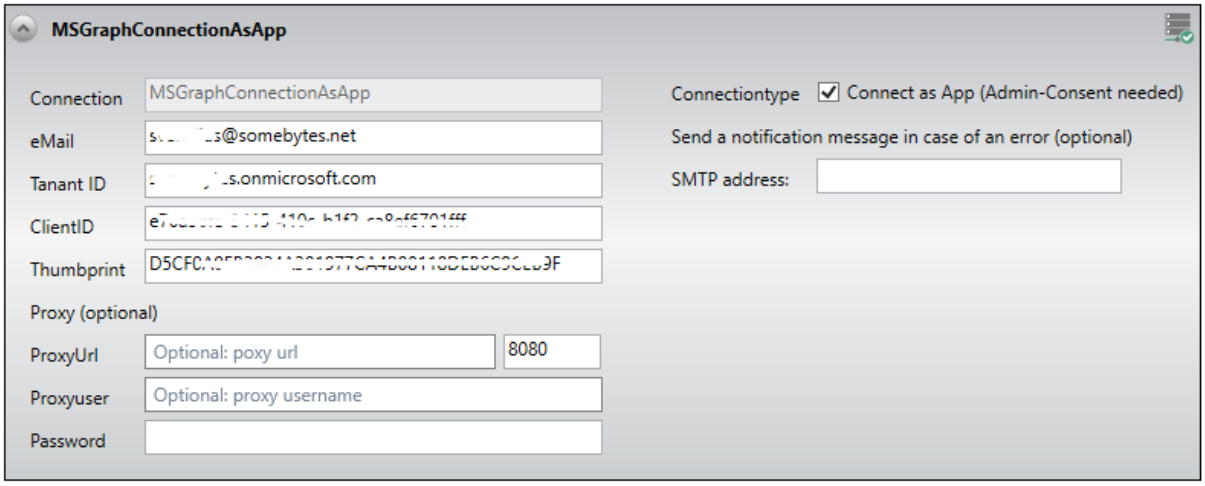
$Cert.thumbprint

5.) Enter the tenantID, applicationID and the thumbprint of the certificate in the corresponding fields in the service connection at EEAttachments.

6.) Optionally, you can specify an existing mailbox in the E-mail field so that the service can check whether the connection is working without an export task.

7.) After clicking on "Connect" the administrator approval page appears with which you can allow the application as a company application.

You can find the client ID in the Azure AD Admin Center, under Properties. (Either a GUID or the domain e.g. contoso.onmicrosoft.com).



If you use this connection in the export jobs, then you must specify the mailbox you want to access in the User ID field.

Gmail Connection

To add a Google Gmail connection, click on Gmail connection in the Service connection menu.

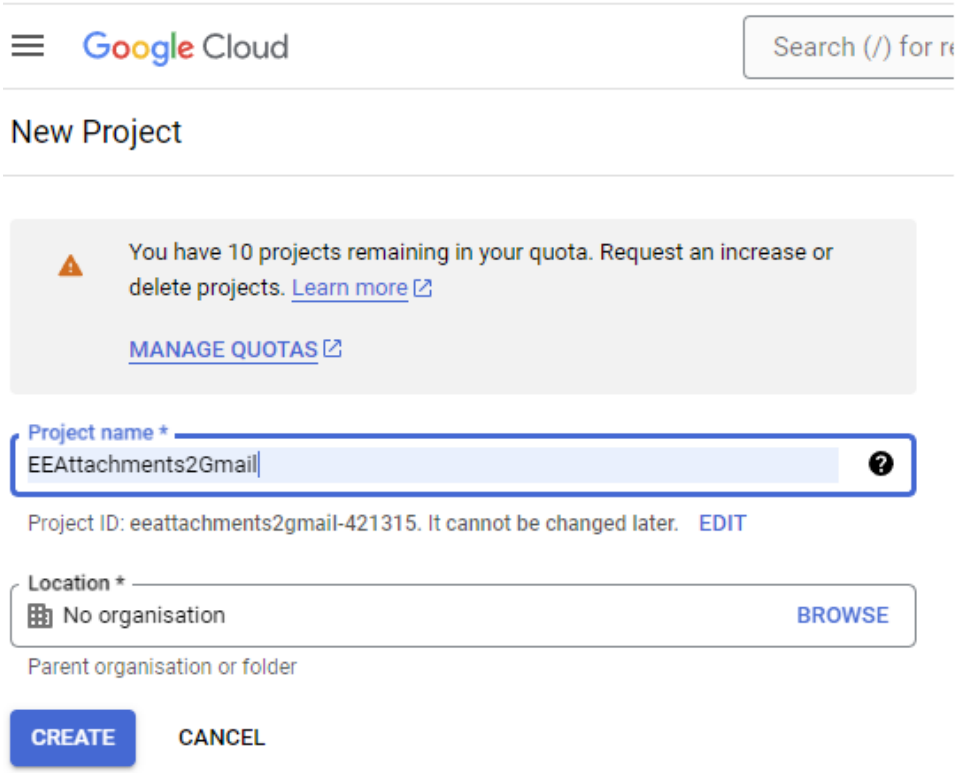
To fill the service dialog with data, you must perform the API registration in the desired Google account.

Info: I did the following procedure for you in previous versions, but this is now considered a security risk by Google, so the following measure is actually for your own safety and you have to do your own Api activation.

To do this, proceed as follows:

Log in to your Google account and navigate to the Google Cloud Console.

Continue to APIs and services and create a new project there. (at the top of the screen)



You can freely choose the name. You only need to enter something for Organization if you have a company account, as this is where you will register the app for your users as an admin.

After you have created the project, select it in the drop-down field if it has not already been done automatically.

Now click on “ENABLE APIS AND SERVICES” and search for “Gmail”.

Select “Gmail API” and “Enable”.

Then click on “OAuth consent screen” and select “External” if you are using a private account or “Internal” if you are using a workspace account.

1. application name: e.g. “EEAttachments”
2. user support e-mail: Your e-mail address
3. contact details of the developer: Your e-mail address.
4. now click “Save and continue”
5. on the next screen you must select the access areas
6. select “Add or remove scopes” and enter the following in the “Manually add scopes” field:

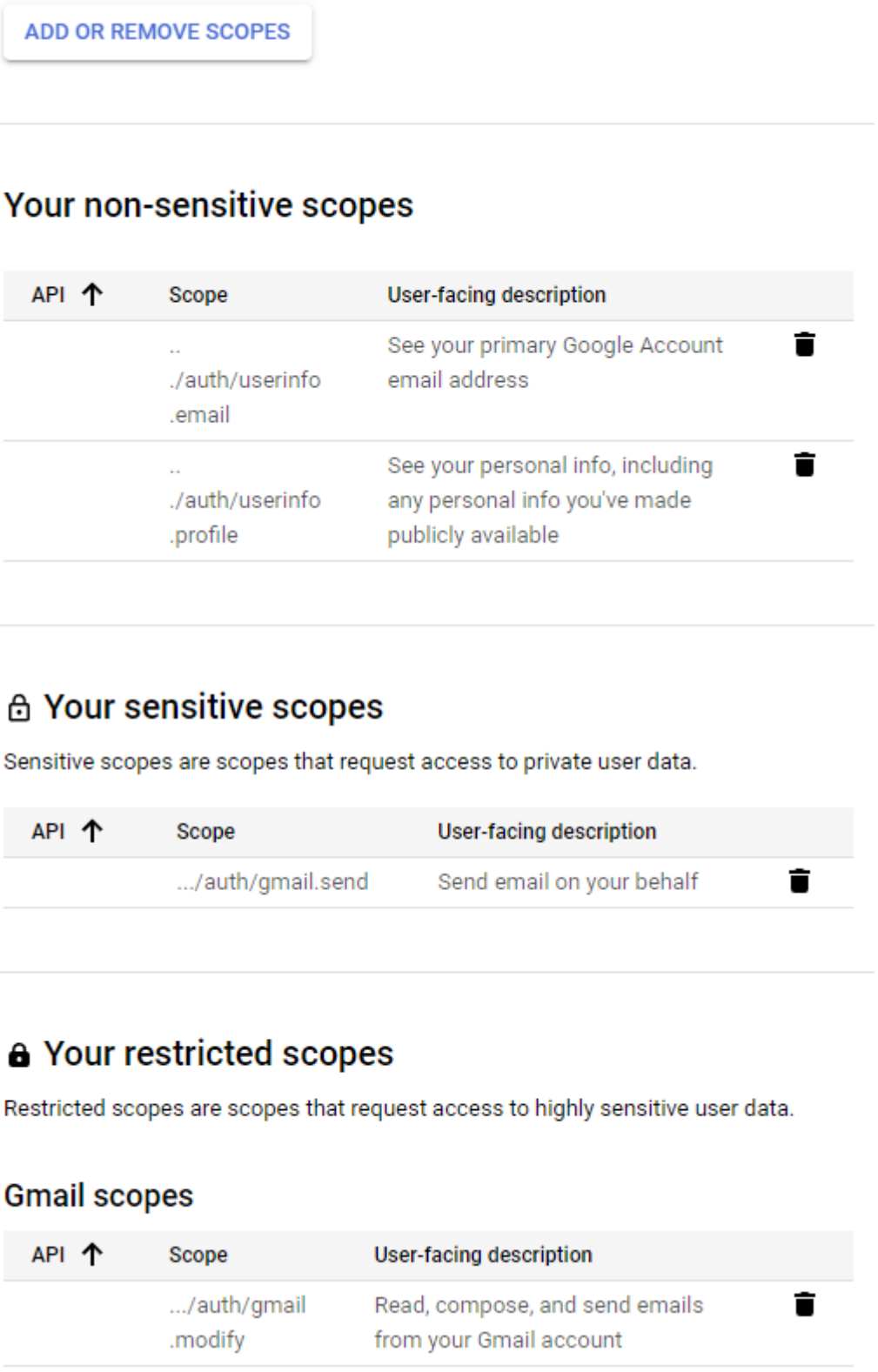
   https://www.googleapis.com/auth/userinfo.profile

   https://www.googleapis.com/auth/userinfo.email

https://www.googleapis.com/auth/gmail.modify

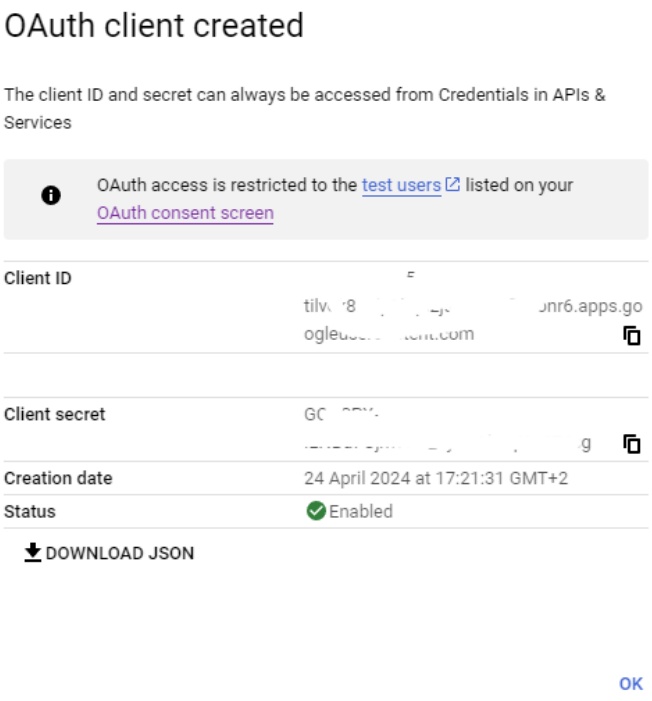
https://www.googleapis.com/auth/gmail.send

7. click “Add to table” and refresh. You should get a screen like this:



8. after “Save and continue” you must specify a test user. If you set up the application internally, you do not need the test, as the application does not have to run in test mode and “External”. The test user is again your own e-mail address.
9. now click on “Save and continue” and return to the desktop.
10. the application does not have to be published and therefore does not have to be approved by Google, which would also be a greater challenge.

Finally, you will need the login data for EEAttachments. You can create these in the “Google Cloud Console” dashboard under “Login data”. Here you will find a link “Create credentials”. Here you select “OAuth client ID” and then select “Desktop app” as the application type. After clicking on “Create”, you will receive a dialog in which you can copy the client ID and the client key for the service dialog from EEAttachments

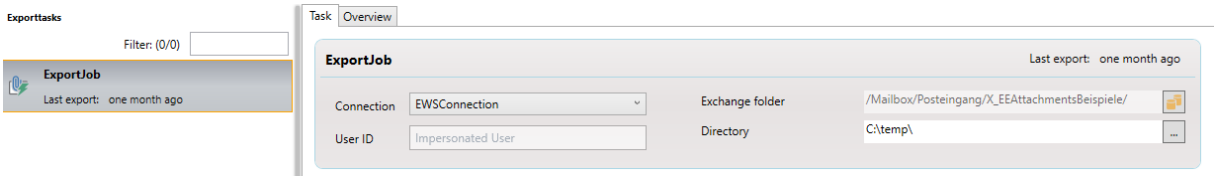


Now add this information to the EEAttachments dialog:

And select “Sign in with Google”. Sign in with the user entered under “Test user”. You should now be connected to Gmail.

Configuration of Exportjobs

To export attachments from Exchange folders you need to add a job in the second part of the configurator. A job includes the ServiceConection you defined earlier, a folder inside a mailbox and a folder on your file system.



Assign a name of your choice for the job and select a service and a folder of the respective Exchange server.

Connection: Is the Exchange webservice you have configured above.

User ID: If you have selected “Impersonation” or “Access as app” in the service-connection, you can enter the user whose mailbox you want to access here.

> Tip for professionals: If you want to create a large number of jobs for your users, we recommend using the automatic job generator, which you start by right-clicking on the User ID field “Create jobs according to this template”. You will receive a dialog in which you must enter the e-mail addresses of the users. If the corresponding folders are found in the mailbox, the wizard will generate the jobs according to the selected template.

Exchange folder: Please select the folder in which you want to search for attachments to export.

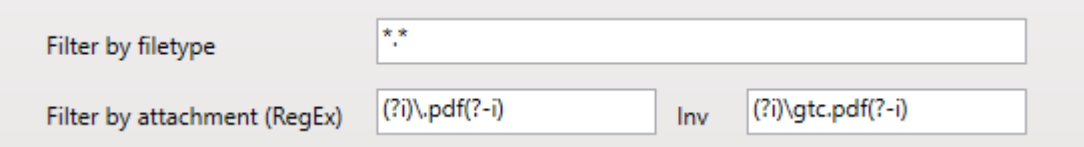
Directory: Is the directory on your filesystem where the exported attachments are located.

Filter by type: Here you specify which files the service should export. To export everything, enter *.*. Otherwise, you can use any file extension here, but you must separate them with a semicolon.

Filter by attachment (RegEx): This filter allows you to filter the attachment to be exported more precisely. A RegEx filter is applied to the file name. (You can test the RegEx filter here http://regexstorm.net/tester)

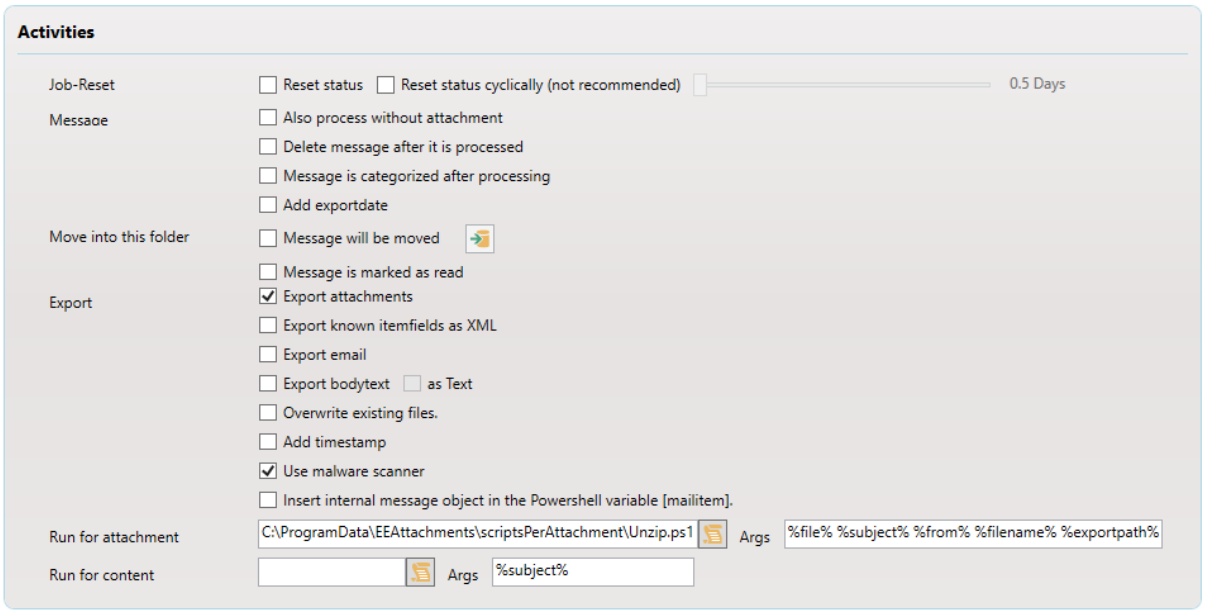
- One use case would be the following: You want to export all PDF files that have “invoice” in their name, regardless of case. To implement this, you can use the following regular expression: (?i).*invoice.*\.pdf(?-i)
- Another example: You want to export all PDF files that end with .pdf, but not those that are named gtc.pdf. In this case, the filter is available again as an exclusion filter. You can enter the

following expression in the Inv field. (?i)gtc\.pdf(?-i) or gtc\.pdf if you do not want to differentiate between upper and lower case.



Further filters: There are other filters which are used to search in the “From”, “Subject” or “Body” field. These filters also work with RegEx.

Time filter: You can set which mails are processed as soon as the status has been reset. The filter can be applied to messages that are younger than the set date or newer. You need the time filter for current mails if you have a large number of mails in your mailbox and want to ensure that not all of them are processed.
The filter can also work inversely so that only messages older than the set date are processed. It only makes sense to use this setting with the automatic reset of the status, as all messages must always (cycle can be configured) be queried for the date. An example of this is the archiving of mails that are older than X days.



Job-Reset: The job internally remembers a status with which it recognizes the last processed message so that it can continue processing the next time a mail arrives.

If you set this checkbox, this status is reset and the service processes all mails again until the status is reached again (the checkmark is then automatically removed). You therefore only need this checkbox in the event of an error or to test your processing scripts.

However, there are exceptions, namely if you would like to archive mails that are older than X days. In this case, all mails must be run through again and again to read the date. For this reason, there is the option “Reset status cyclically” (every X days) This option makes particular sense with the time filter, which only processes messages if they are older than X days.

Process message without attachment: You can use the checkbox to ensure that messages are processed despite failed filtering or even without an attachment

Process message without attachment: You can use the checkbox to ensure that messages are processed despite failed filtering or even without an attachment.

Add export date: With Exchange Messages, additional information can be added to the message, which is visible in the header of the mail in Outlook. The export date is inserted here.

Move message: Messages can be moved within the mailbox to another folder after they have been successfully processed.

Mark e-mail as read: The mail is marked as read after successful processing. You can also use this option to ensure that mails are not processed more than once by mistake if you use the “Process unread mails only” filter at the same time.

Export attachments: Mail attachments are exported if they have passed the filter, for example for further processing.

Export mail content as XML: With Exchange, the messages can also be provided as XML if you want to process XML.

Export e-mail: Mails can be exported as EML files. EML files can be read by all possible mail programs.

Export mail text: You can use this to export the mail text as HTML or text.

Start program: For each exported document, you can start a program, PowerShell script or batch file. This allows you to ensure that the attachment is processed further. The batch file receives the file name, the subject, the sender address or other fields that you may wish to process as parameters.

Overwrite existing files: If you use this option, files with the same name are overwritten, otherwise files are created in which additional characters are added.

Append timestamp: A date is added to the exported attachments.

Use malware scanner: This allows you to check files and scripts for viruses before they are exported or executed.

Pass EWS message object to PowerShell: If you want to read or edit the message object directly, you can use this option. You will get a session variable named mailitem.

Processing:

You can specify different scripts for processing the message and for processing the attachments.

Which parameters are transferred to batch processing depends on which variables you define under Parameters.

Here you can find useful parameters:

%file% = path to the exported file.

%filename% = filename without directory.
%xmlfile% = path to XML – E-Mail content.
%Subject% = Subject of the Mail.
%From% = From field of the E-Mail.
%taskname% = Name of the EEAttachments Export-Task.
%exportpath% = The path which you have entered in the Export-Task.

These are fields you can use as well:

%ActualWork% %AssignedTime% %BillingInformation% %ChangeCount% %Companies% %CompleteDate% %Contacts% %DelegationState% %Delegator% %DueDate% %IsComplete% %IsRecurring% %IsTeamTask% %Mileage% %Mode% %Owner% %PercentComplete% %Recurrence% %StartDate% %Status% %StatusDescription% %TotalWork% %Alias% %AssistantName% %Birthday% %BusinessHomePage% %Children% %Companies% %CompanyName% %CompleteName% %ContactSource% %Department% %DirectoryId% %DirectReports% %DisplayName% %EmailAddresses% %FileAs% %FileAsMapping% %Generation% %GivenName% %HasPicture% %ImAddresses% %Initials% %JobTitle% %Manager% %ManagerMailbox% %MiddleName% %Mileage% %MSExchangeCertificate% %NickName% %Notes% %OfficeLocation% %PhoneNumbers% %PhoneticFirstName% %PhoneticFullName% %PhoneticLastName% %Photo% %PhysicalAddresses% %PostalAddressIndex% %Profession% %SpouseName% %Surname% %WeddingAnniversary% %AdjacentMeetingCount% %AdjacentMeetings% %AllowNewTimeProposal% %InternetMessageId% %BccRecipients% %CcRecipients% %ConversationTopic% %From% %InternetMessageId% %IsDeliveryReceiptRequested% %IsRead% %IsReadReceiptRequested% %IsResponseRequested% %ReceivedBy% %ReceivedRepresenting% %ReplyTo% %Sender% %ToRecipients% %MimeContent% %AppointmentReplyTime% %AppointmentSequenceNumber% %AppointmentState% %AppointmentType% %ConferenceType% %ConflictingMeetingCount% %ConflictingMeetings% %DeletedOccurrences% %Duration% %End% %EndTimeZone% %EnhancedLocation% %ICalDateTimeStamp% %ICalRecurrenceId% %ICalUid% %IsAllDayEvent% %IsCancelled% %IsMeeting% %IsOnlineMeeting% %IsRecurring% %IsResponseRequested% %JoinOnlineMeetingUrl% %LegacyFreeBusyStatus% %Location% %MeetingRequestWasSent% %MeetingWorkspaceUrl% %MyResponseType% %NetShowUrl% %OnlineMeetingSettings% %OptionalAttendees% %Organizer% %OriginalStart% %Recurrence% %RequiredAttendees% %Start% %StartTimeZone% %TimeZone% %When% %ArchiveTag% %Body% %Categories% %Culture% %DateTimeCreated% %DateTimeReceived% %DateTimeSent% %DisplayCc% %DisplayTo% %Flag% %HasAttachments% %IconIndex% %Importance% %InReplyTo% %IsAssociated% %IsDraft% %IsFromMe% %IsReminderSet% %IsResend% %IsSubmitted% %IsUnmodified% %ItemClass% %LastModifiedName% %LastModifiedTime% %ReminderDueBy% %ReminderMinutesBeforeStart% %RetentionDate% %Sensitivity% %Size% %Subject% %TextBody%

This link shows you the documentation of these fields:

http://msdn.microsoft.com/en-us/library/microsoft.exchange.webservices.data.emailmessageschema_fields(v=exchg.80).aspx

If you use EEAttchments to start PowerShell scripts EEAttachments will add your arguments as session-variables (Get-Variable). Also you will get a variable which contains the full MailItem in order to read or change it. You will find it in the variable “mailitem”.

EEAttachments – Documentation

In the program folder of EEAttachments you will find some PowerShell scripts like the unzip.ps1 which can unzip your attachments after the export.

Menu

Manage Exchange Webservice Connections

In the menu, under [EWS], you can set up and delete Exchange web service connections.

In addition, the menu contains the item [licenses] with which you can manage your licenses.

You can also find the menu items in the corresponding list boxes in the form of a context menu.

Manage Jobs

In the menu, under [Jobs] you can set up and delete synchronization jobs. A job sets up a connection between two folders of one mailbox or between two folders of different mailboxes.

Manage EEAttachments Services

Menu [Service]

This is where you save your settings and control the EEAttachments service.

You can also find the service if you enter "services.msc" under Run or double-click on the icon at the bottom right of the status bar. Behind Log file you will find a log window and the log file, which is useful for debugging your scripts. The output of your scripts is also listed here.

The EEAttachments service runs as SYSTEM user by default, but you can also enter a different user here if you wish. To configure the service, enter services.msc in the command prompt or go to "Services" in the Control Panel. Here you will find the EEAttachments service and can change the account by right-clicking => Properties => Log in.

Cybersecurity

The program can check your scripts, attachments and arguments for malware and uses the "Antimalware Scan Interface (AMSI)", which works together with Windows Defender or other anti-virus programs. You can specify in each job whether you want to use the scanner or not. If malware is found, the file is not processed and a corresponding error appears in the log.

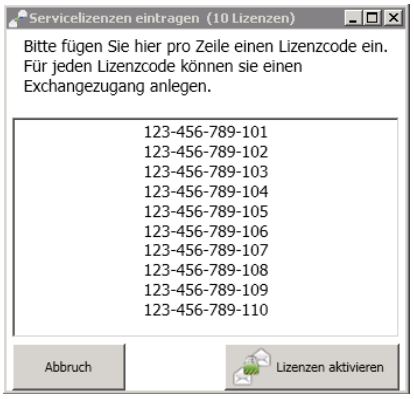
You can also save the configuration files in encrypted form by creating a DWORD value of 1 with the name UseConfigEncryption in the registry key HKLM\Software\Somebytes\EEAttachments. After you have

restarted the configurator and the configuration has been saved, the service can use the encrypted configuration.

Save your Powershell scripts or batch files in a directory that can only be accessed by the user of the Windows service (EEAttachments) to prevent an attacker from being able to inject code that may then be executed with elevated rights.

Another attack option is the internal MessageItem, which you can write to a PowerShell variable for further processing. Only activate the option in the export job if you actually need to work with the MessageItem in the script.

Licensing



EEAttachments requires a license for every Mailbox you would like to access. Enter the licenses received from Somebytes horizontally in the dialog and restart the service. If you would like to reach several Mailboxes in your configuration, for which you have no license, those remaining will not be loaded.

Troubleshooting

- Should you encounter problems with the service, you can look up the service's error messages in the file EEAttachments.log under C:\programdata\EEAttachments\logs\ which you can discuss with info@somebytes.net as well. You can change the logging behavior with NLog.config from the NLog project. (https://nlog-project.org/config/).
- Further log data (please also see Logging in the configurator) can be found under C:\ProgramData\EEAttachments\logs\.
- In C:\ProgramData\EEAttachments\ you can also the configuration files for EEAttachments (.conf).
- By default, EEAttachments is run as a SYSTEM user. You can, however, enter another user if you wish. In order to configure the service, please enter services.msc at the command prompt or go to "Services" within the control panel. Here you will find the service EEAttachments.