



ORGanizer for Salesforce Connector App

Administrator Guide v.1.0



ORGanizer for Salesforce Connector App

Administrator Guide

Table of Contents

Introduction	3
Installation	3
Configuration	3
Permission Sets	3
Custom Settings	3
Sharing configuration.....	4
Data Model	4
Backup.....	4
Backup Item	5
Backup Log	5
Connector App Webservices.....	6
ORGanizer for Salesforce Connector App Usage	6
Populate Data	6
Import ORGanizer Data Wizard	7
Connect with the ORGanizer for Salesforce browser extension	10
Reveal password & token from within the Lightning App	16
Connector App Webservices: Developer Details	18



Introduction

The **ORGanizer for Salesforce Connector App** is a Salesforce AppExchange package that is able to connect to **ORGanizer for Salesforce browser extension** (<https://organizer.solutions>, already published on the [AppExchange](#)) for secure Salesforce credentials storage (more features will be added in the future).

ORGanizer for Salesforce extension is a productivity tool for Salesforce professionals (developers, administrators and users) which delivers login capabilities and credentials storage plus tens of utility plugins to improve Salesforce working tasks.

The idea behind the Connector App is a way to let companies store the updated Salesforce org credentials into a centralized org (where the package is installed) so that each employee when needed can *grab* (or download) a specific credential into their instance of ORGanizer for Salesforce extension, if and only if the administrator of the centralized org let them access (via standard Salesforce sharing features) the credential record.

Credentials are protected in the centralized org with a hash key and the AES-256 encryption algorithm (stored in a password field on the credential object): when an employee needs a credential and is given access to the specific record by the centralized org admin, he can send the encryption key to himself via mail so he can proceed to download the credentials on the extension.

Installation

Simply install the package from AppExchange and select “Install for Admins only”.

Configuration

The package needs very low configuration:

- Permission set assignment to users and administrators
- Custom setting configuration for debugging and org-wide address configuration (optional)
- Sharing rules / manual sharing configurations to let users access only specific record

Permission Sets

The Connector App comes with 2 permission sets:

- **ORGanizer Backup Admin**: meant for admins
- **ORGanizer Backup Permissions**: meant for regular users

They convey access to the basic features and operations needed by admins and users.

Custom Settings

The Connector App comes with the **ORGanizer Backup Settings** custom setting that holds 2 fields:

- **Enable Integration Logs**: by enabling this feature (as an org default or for specific users) integration logs will be stored in the dedicated **Backup Log** object, that admins can inspect to check any issue on the connection
- **Org-wide Email Address**: set the email address used to send the secret to users, which must be configured as an Organization Wide Address (this prevents email to be considered spam on the users' mail services)

Sharing configuration

Use Sharing Rules configuration or manual sharing to share the backup records with the users in your company (standard Salesforce feature).

Data Model

Below it is shown the simple data model of the package:

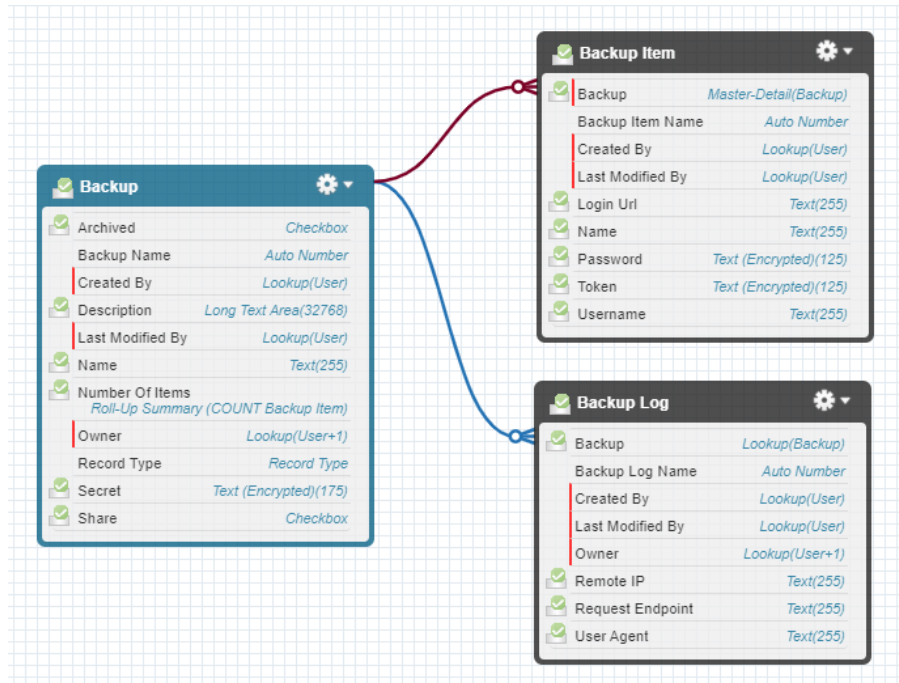


Figure 1 – ORGanizer for Salesforce Connector App data model schema

Backup

The Backup object is the parent record that identifies a “group of credentials” that can be downloaded into ORGanizer for Salesforce extension.

This object contains:

- SHA-256 secret (used to encrypt password/token when transmitting to-from ORGanizer extension)
- Share: identifies a record as shared (just an informative field)
- Archived: identifies a record as archived, so it won’t be retrieved by ORGanizer for Salesforce
- Number of records: roll-up summary field that shows up the number of child credential records
- Name/Description: informative fields to identify the backup package

When a “Backup” record is created, a trigger automatically generates a random SHA-256 key that is stored in the “Secret” field and is used to automatically encrypt the child record passwords for transmission when requested from the ORGanizer for Salesforce extension.

Backup BKG-00000001	
Related	Details
Backup Name	BKG-00000001
Name	Bolis' Demo Logins
Description	Logins related to demo orgs owned by Bolis
Number Of Items	2
Created By	Tommaso Bolis, 25/03/2019 14.05
Owner	Tommaso Bolis
Share	☒
Archived	☐
Last Modified By	Tommaso Bolis, 25/03/2019 15.00

Figure 2 – Backup record example

Backup Item

This object contains the actual credential for a given org:

- Name: identification name
- Username: login username
- Password: encrypted password using a password field (automatically obfuscated using SHA256 passkey)
- Token: security token (automatically obfuscated using SHA256 passkey)
- Login Url: requested login endpoint

The password and token values are securely stored as password fields, both automatically obfuscated using a trigger. When a user requests a credential from the browser extension, it is obfuscated and encrypted on the fly using the SHA-256 algorithm and must be decrypted using the shared secret on the extension side.

Backup Item BKI-00000001	
Related	Details
Backup Item Name	BKI-00000001
Backup	BKG-00000001
Name	DE 001
Username	demo01@test.demo
Password	*****
Token	*****
Login Url	https://login.salesforce.com
Created By	Tommaso Bolis, 25/03/2019 14.05
Last Modified By	organizer.dev@enree.co, 17/04/2019 20.19

Figure 3 – Backup Item example

Backup Log

The Backup Log object is a system object that is used to track, only if enabled on the dedicated Custom Setting, access to the Connector App from the extension.

It contains:

- Backup lookup (if the request contained the backup ID)
- Request endpoint
- Remote IP address

- User Agent

Backup Log	
BKL-00000454	
Related	Details
Backup Log Name	BKL-00000454
Owner	organizer.dev@enree.co
Backup	BKG-00000001
Request Endpoint	/orgnz_backupper/api/v1.0/backup-management/backups/a041t000007wzwEAAQ
Remote IP	79.33.81.235
User Agent	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/92.0.4515.131 Safari/537.36
Created By	organizer.dev@enree.co, 11/08/2021 11:00
Last Modified By	organizer.dev@enree.co, 11/08/2021 11:00

Figure 4 - Backup Log record example

Connector App Webservices

The Connector App talks to the ORGanizer for Salesforce browser extension via 3 REST webservices (enabled via the dedicated permission set):

- **orgnz_backupper.EchoWsr_1_0**: via this webservice the browser extension is able to get if the Connector App package is installed in the remote org
- **orgnz_backupper.BackupWsr_1_0**: main service to get the list of all available/shared backup packages
- **orgnz_backupper.BackupItemWsr_1_0**: service to get the credentials list for a given Backup package

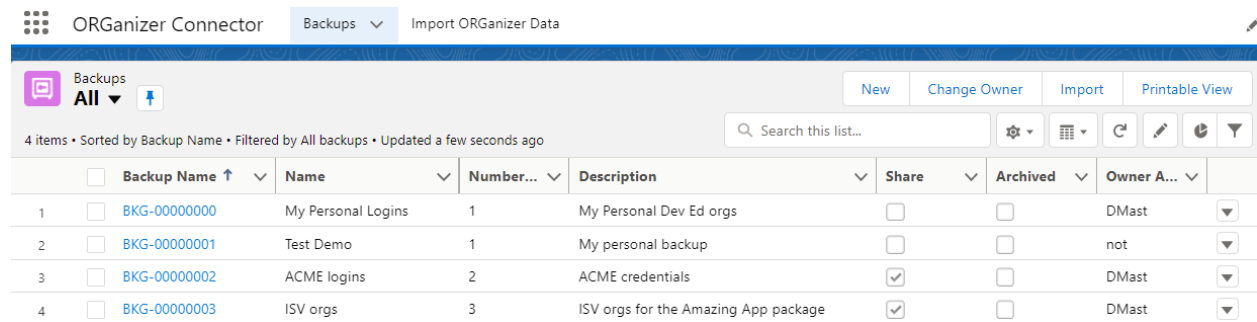
Each service needs a valid session ID obtained by ORGanizer for Salesforce browser extension (shown below): it is **strongly recommended** not to expose these services publicly (e.g., with a Customer Community / Salesforce Site).

ORGanizer for Salesforce extension PRO users will have this integration automatically setup, while “free tier” users will have to code their way to consume this service if they need it (e.g. for automation purposes for example or to develop a tool to massively extract the credentials stored in the Connector App): have a look at the end of this guide for more details.

ORGanizer for Salesforce Connector App Usage

Populate Data

Access the **ORGanizer Connector** app and create Backup and Backup Item records manually:



	Backup Name ↑	Name	Number...	Description	Share	Archived	Owner A...
1	BKG-00000000	My Personal Logins	1	My Personal Dev Ed orgs	☐	☐	DMast
2	BKG-00000001	Test Demo	1	My personal backup	☐	☐	not
3	BKG-00000002	ACME logins	2	ACME credentials	☒	☐	DMast
4	BKG-00000003	ISV orgs	3	ISV orgs for the Amazing App package	☒	☐	DMast

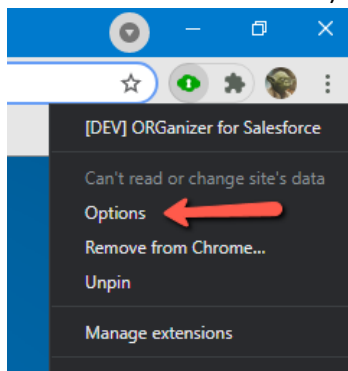
Figure 5 – Connector App configured

Import ORGanizer Data Wizard

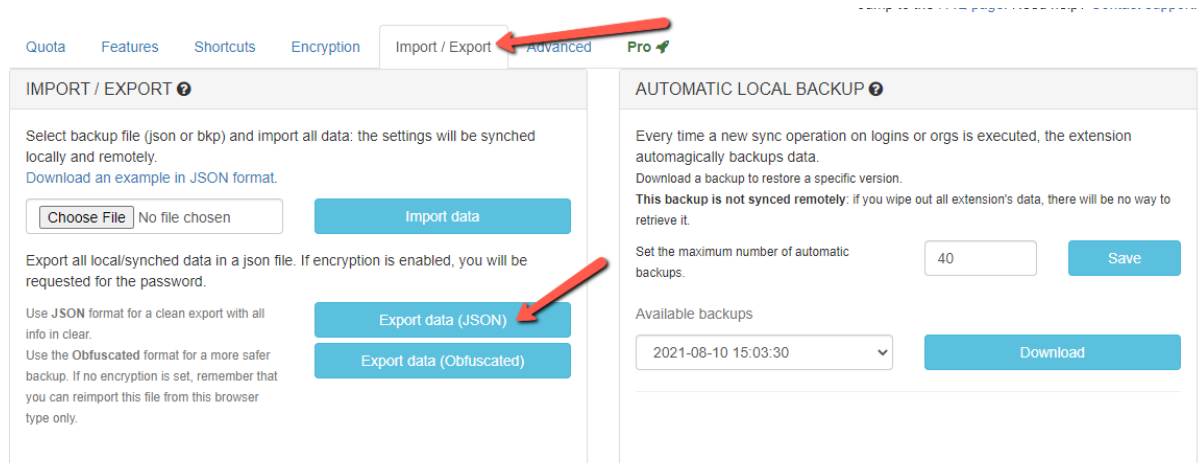
In alternative, it is possible to use the **Import ORGanizer Data** wizard which lets you get a valid backup JSON got from the ORGanizer for Salesforce browser extension and put the logins into a given Backup record on the Connector App.

To get a backup from the extension, follow the following steps:

1. Right click on the ORGanizer for Salesforce browser extension icon and select Options (only for Chrome / Firefox / Opera; on Firefox click on “Manage Extension” and then select Options from the extension choice menu)



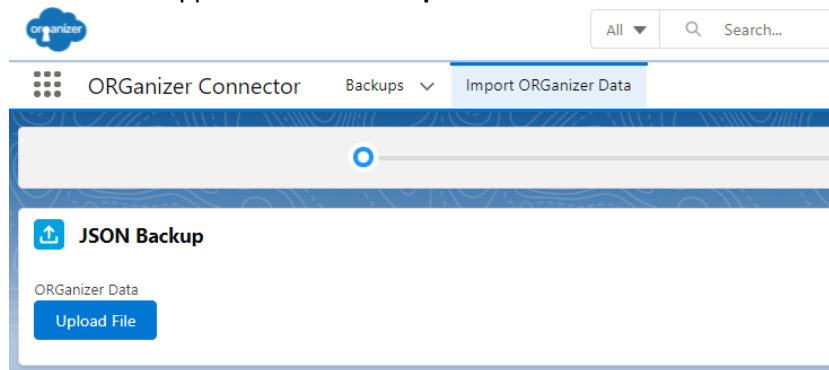
- 2.
3. Select the “IMPORT / EXPORT” tab and in the “IMPORT/EXPORT” panel click the “Export data (JSON)” button:



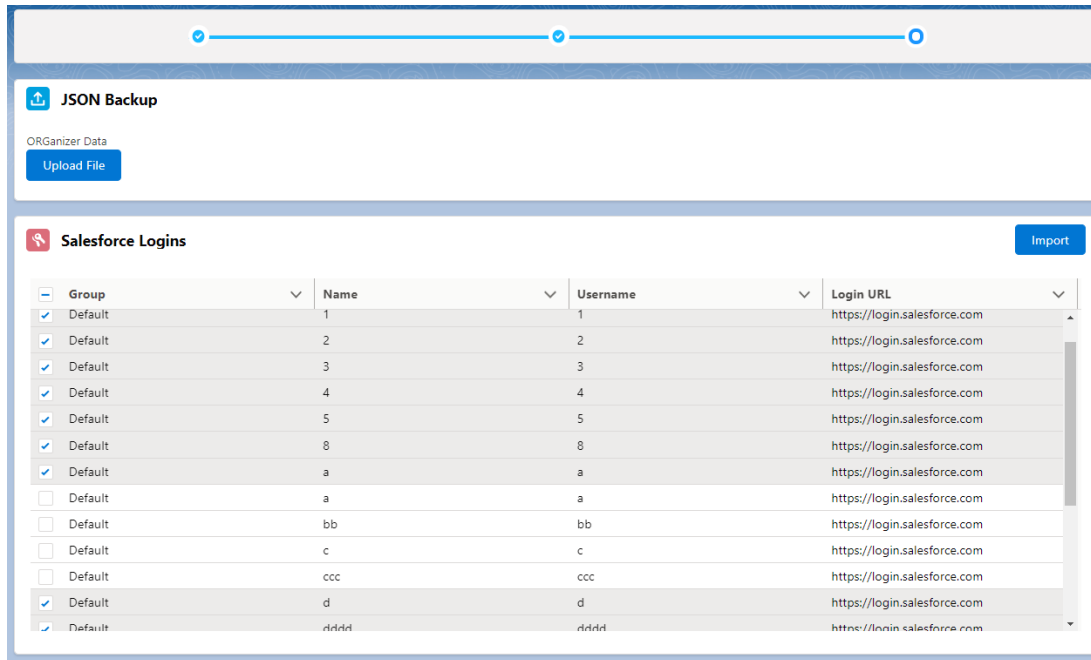
4. Download the backup file (which have a similar format):

```
1 {
2   "accounts": [
3     {
16    },
17    {
30    },
31    {
32      "d": 1626708827070,
33      "f": false,
34      "g": "Default",
35      "lo": false,
36      "lp": "0",
37      "lt": false,
38      "n": "3",
39      "p": "1",
40      "r": "0",
41      "s": true,
42      "t": "",
43      "u": "3"
44    },
45    {
46      "d": 1628586829336,
47      "f": false,
```

5. Go back to the Connector App and select the **Import ORGanizer Data** wizard:



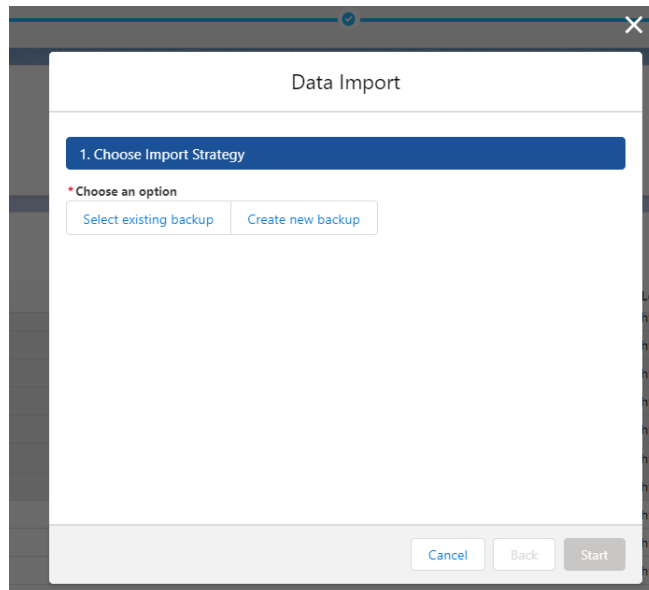
6. Click the **Upload File** button and select the file exported in 3). The UI shows the available credentials on the file and lets you select the logins to be imported:



The screenshot shows the ORGanizer Data interface. At the top, there's a progress bar with three steps. Below it, the 'JSON Backup' section has an 'Upload File' button. The 'Salesforce Logins' section features a table with columns: Group, Name, Username, and Login URL. The table contains 12 rows. The first 7 rows have 'Default' in the Group column and are selected with checkboxes. The last 5 rows are not selected. An 'Import' button is located at the top right of the table.

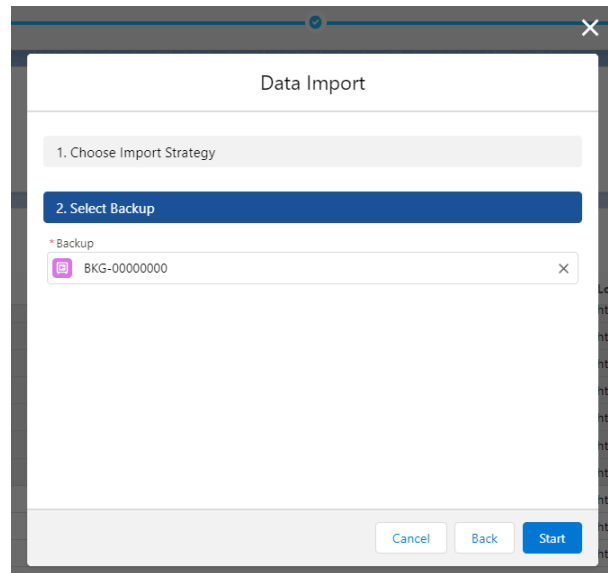
Group	Name	Username	Login URL
☒ Default	1	1	https://login.salesforce.com
☒ Default	2	2	https://login.salesforce.com
☒ Default	3	3	https://login.salesforce.com
☒ Default	4	4	https://login.salesforce.com
☒ Default	5	5	https://login.salesforce.com
☒ Default	8	8	https://login.salesforce.com
☒ Default	a	a	https://login.salesforce.com
☐ Default	a	a	https://login.salesforce.com
☐ Default	bb	bb	https://login.salesforce.com
☐ Default	c	c	https://login.salesforce.com
☐ Default	ccc	ccc	https://login.salesforce.com
☒ Default	d	d	https://login.salesforce.com
☒ Default	dddd	dddd	https://login.salesforce.com

- Click the **Import** button to start the record creation process. A popup requests to select an existing Backup record or to create a new one:

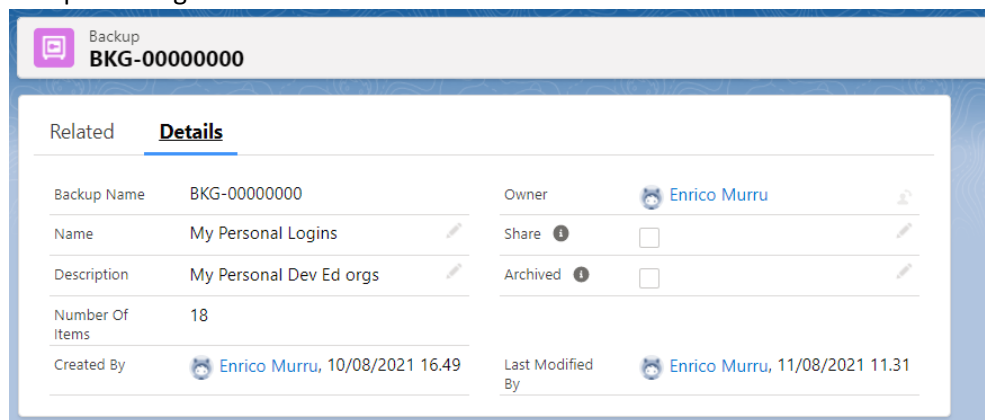


The screenshot shows a 'Data Import' popup window. It has a title bar with a close button. The main content area is titled '1. Choose Import Strategy'. Below this, there's a section labeled '* Choose an option' with two buttons: 'Select existing backup' and 'Create new backup'. At the bottom of the window, there are three buttons: 'Cancel', 'Back', and 'Start'.

- In this example we'll be selecting an existing backup record:



9. Click **Start** to start the import process. If everything went ok, the backup record will be populated with the imported logins:



Connect with the ORGanizer for Salesforce browser extension

The Connector App is a free AppExchange app that at this time is just a password store manager and that in the future will gain other free features, but in order to be used coupled with ORGanizer for Salesforce browser extension, the extension needs to get a license (the extension is free to use anyway).

The package itself won't prevent anyone from consuming the webservices used by ORGanizer for Salesforce extension in PRO mode (see in "Connector App Webservices" chapter) and "free tier" users can still develop their way to consume using any programming language: there is an example on how to consume the webservices at the end this guide with the details of each service.

Install the extension from <https://organizer.solutions/index.html#getit>.

Once the extension is enabled, to talk to the Connector App you need to input in the extension a valid credential to the org where the app is installed:

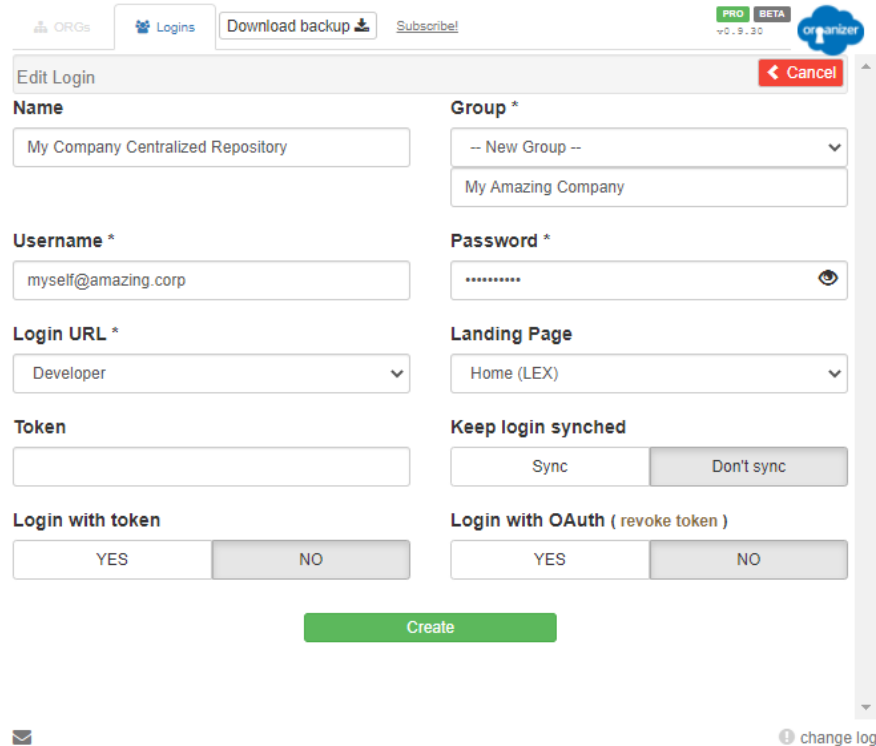


Figure 6 – ORGanizer for Salesforce extension: login creation

You can use OAuth login or token login to enable API access.

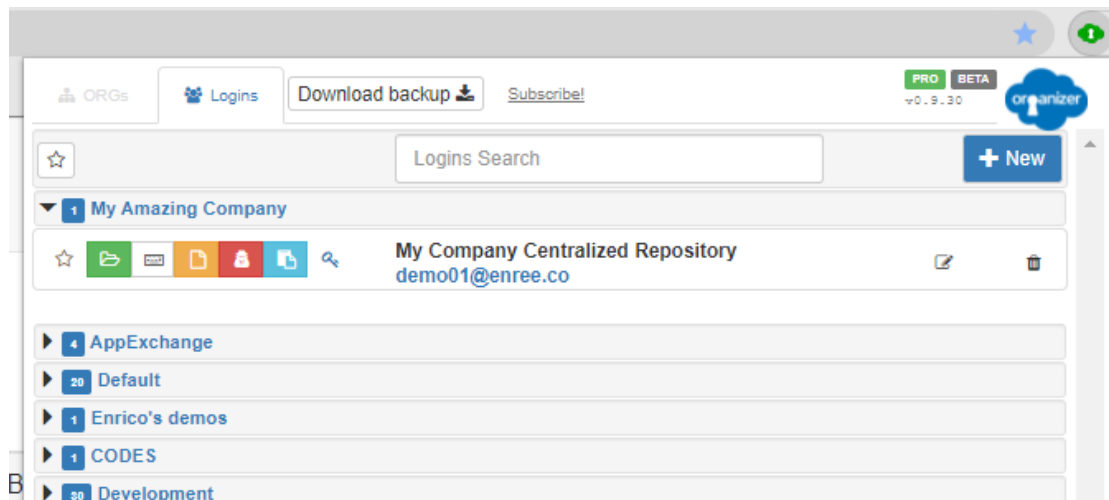
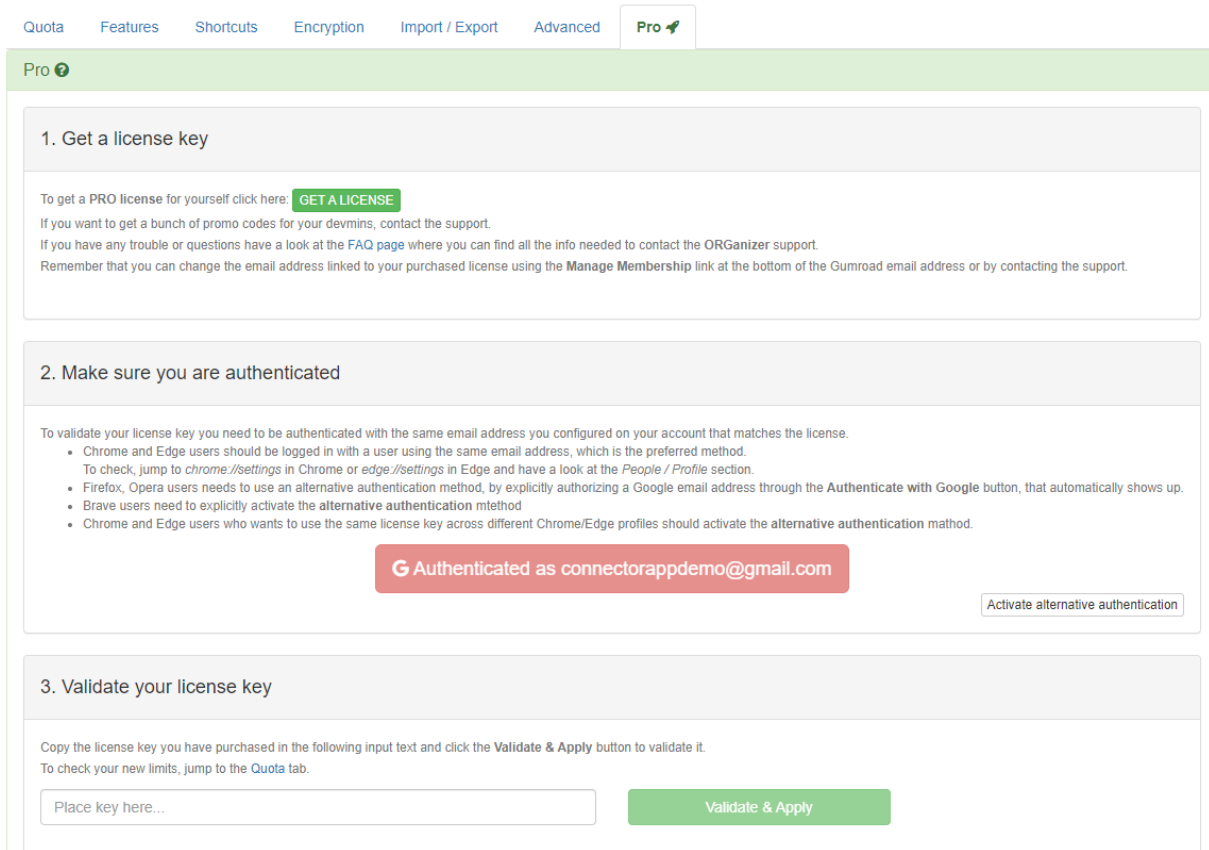


Figure 7 – ORGanizer for Salesforce popup page with stored credentials

Once the login is saved, you can start the Connector App connection process.

The last step is to activate a **FULL** or **TEAM** PRO license that you have to purchase from the ORGanizer for Salesforce site (following [this link](#)).

Once you have obtained your license key, click on the Options page (as shown previously) and click on the “PRO” tab:



The screenshot shows the 'Pro' tab selected in the top navigation bar. The page is divided into three main sections:

- 1. Get a license key**

To get a PRO license for yourself click here: [GET A LICENSE](#)

If you want to get a bunch of promo codes for your devmins, contact the support.

If you have any trouble or questions have a look at the [FAQ page](#) where you can find all the info needed to contact the ORganizer support.

Remember that you can change the email address linked to your purchased license using the **Manage Membership** link at the bottom of the Gumroad email address or by contacting the support.
- 2. Make sure you are authenticated**

To validate your license key you need to be authenticated with the same email address you configured on your account that matches the license.

 - Chrome and Edge users should be logged in with a user using the same email address, which is the preferred method.
 - To check, jump to *chrome://settings* in Chrome or *edge://settings* in Edge and have a look at the *People / Profile* section.
 - Firefox, Opera users needs to use an alternative authentication method, by explicitly authorizing a Google email address through the **Authenticate with Google** button, that automatically shows up.
 - Brave users need to explicitly activate the **alternative authentication** method
 - Chrome and Edge users who wants to use the same license key across different Chrome/Edge profiles should activate the **alternative authentication** method.

G Authenticated as connectorappdemo@gmail.com

[Activate alternative authentication](#)
- 3. Validate your license key**

Copy the license key you have purchased in the following input text and click the **Validate & Apply** button to validate it.

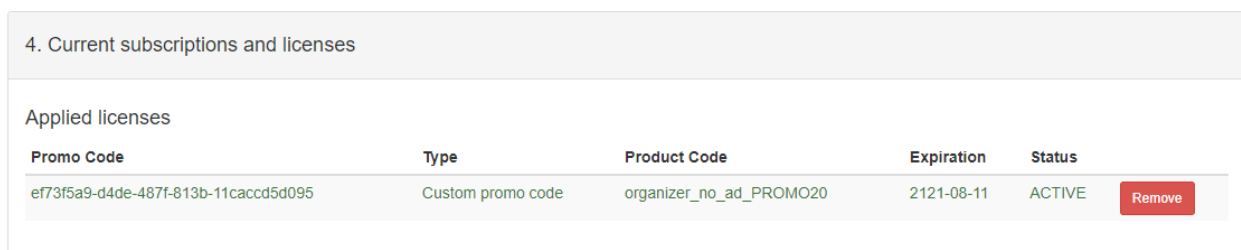
To check your new limits, jump to the [Quota](#) tab.

Place key here... [Validate & Apply](#)

Make sure you are authenticated with the email address registered with your license (in this example connectorappdemo@gmail.com, using Chrome profile sync feature): alternatively, you can hit the “Activate alternative authentication” button and login using Google OAuth flow (this avoids using Chrome’s profile email address).

Insert the provided license key on the “3. Validate your license key” panel and hit “Validate & Apply”.

If everything goes ok, the last panel shows the new license:



The screenshot shows the '4. Current subscriptions and licenses' panel. It contains a table of applied licenses.

Promo Code	Type	Product Code	Expiration	Status	
ef73f5a9-d4de-487f-813b-11cacc5d095	Custom promo code	organizer_no_ad_PROMO20	2121-08-11	ACTIVE	Remove

Refresh the Options page and jump again in the “Import/Export” tab but now focus on the “ORganizer Connector (PRO)” panel:

ORGanizer Connector (PRO) ?

ORGanizer Connector AppExchange app lets you store company-wide passwords on a dedicated package right on your CRM.

1. Select a login to initiate a connection with the ORGanizer Connector app

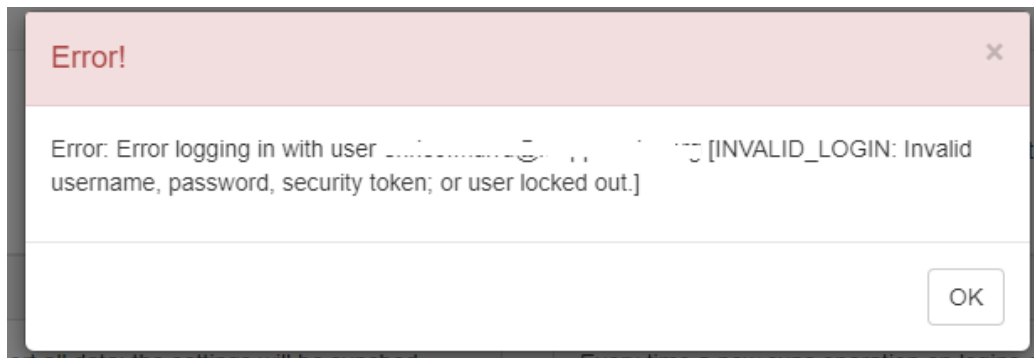
My Company Centralized Repository (demo01@enree.co) ▼

Open Connection

Imported logins will replace local logins with the same username. **Imported logins are automatically flagged as "not synced"**.

Select the right login configuration and click the **Open Connection** button.

If there are issues with the connection the system shows an error:



If everything goes ok, the page shows the list of available backups:

1. Select a login to initiate a connection with the ORGanizer Connector app

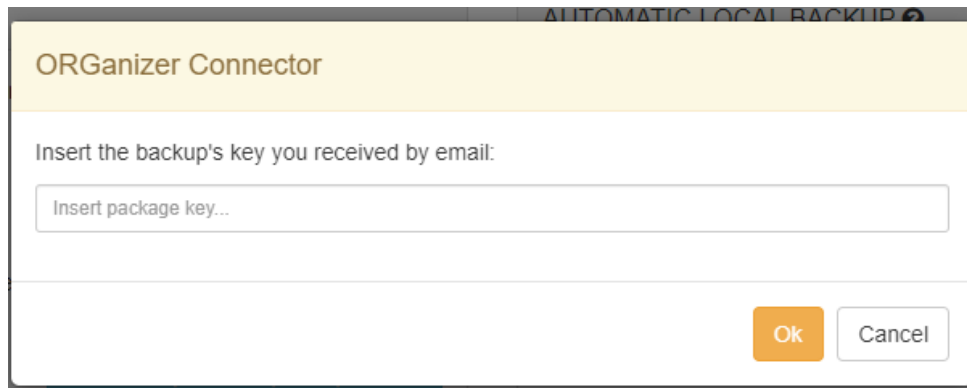
My Company Centralized Repository (demo01@enree.co) ▼

Open Connection

2. Select a package

Name	Description	Logins count	Shared	
acme logins	ACME credentials	2	✓	<button>Get Backup</button>
isv orgs	ISV orgs for the Amazing App package	3	✓	<button>Get Backup</button>
my personal logins	My Personal Dev Ed orgs	18	-	<button>Get Backup</button>
test demo	My personal backup	1	-	<button>Get Backup</button>

Now that your user has access to the backup list, click the **Get Backup** button to retrieve the available logins. The system will ask for the secret key:



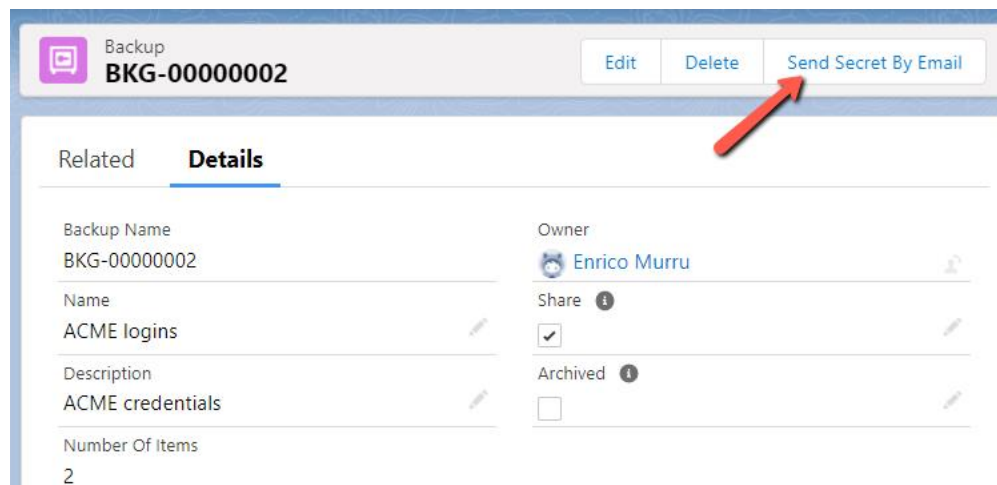
AUTOMATIC LOCAL BACKUP

ORGanizer Connector

Insert the backup's key you received by email:

Ok Cancel

To get the key open the backup record and click the **Send Secret By Email**:

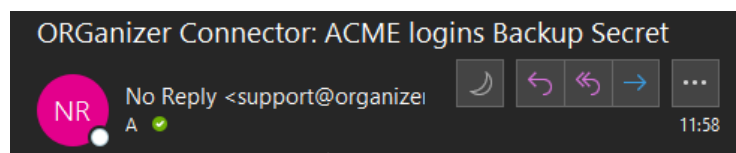


Backup **BKG-00000002** Edit Delete **Send Secret By Email**

Related **Details**

Backup Name	BKG-00000002	Owner	Enrico Murru
Name	ACME logins	Share	☒
Description	ACME credentials	Archived	☐
Number Of Items	2		

A message has just arrived into the user's email address with the secret:



Dear Enrico Murru,
here is the secret for backup [ACME logins \(BKG-00000002\)](#):
7gqj9TQcEsk0JhXhpLTTVbVAF+SphEz7NPT+faxadhQ=

Kind Regards,
ORGanizer Connector App

Copy the secret on the clipboard and paste into the extension popup and the list of all available logins will be shown:

ORGanizer Connector (PRO) ?

ORGanizer Connector AppExchange app lets you store company-wide passwords on a dedicated package right on your CRM.

1. Select a login to initiate a connection with the ORGanizer Connector app

My Company Centralized Repository (demo01@enree.co) Open Connection

2. Select a package

Name	Description	Logins count	Shared	
acme logins	ACME credentials	2	✓	Get Backup
isv orgs	ISV orgs for the Amazing App package	3	✓	Get Backup
my personal logins	My Personal Dev Ed orgs	18	-	Get Backup
test demo	My personal backup	1	-	Get Backup

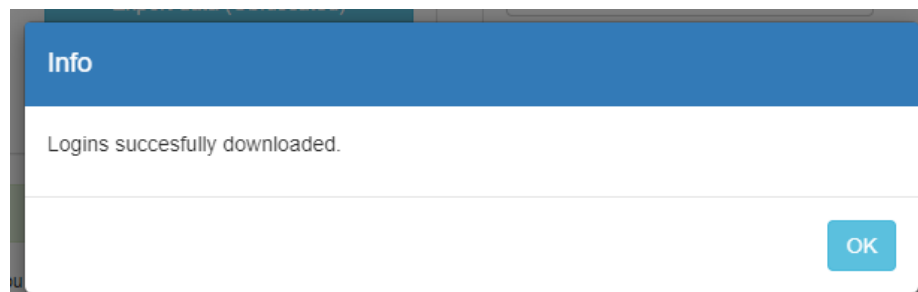
3. Select the logins you want to import locally and a group from the list

	Name	Username	Login URL	Last Modified
☐	acme production	prod@acme.com	https://login.salesforce.com	8/11/2021, 10:45:50 AM by Enrico Murru
☐	acme stage	stage@acme.com	https://test.salesforce.com	8/11/2021, 10:46:06 AM by Enrico Murru

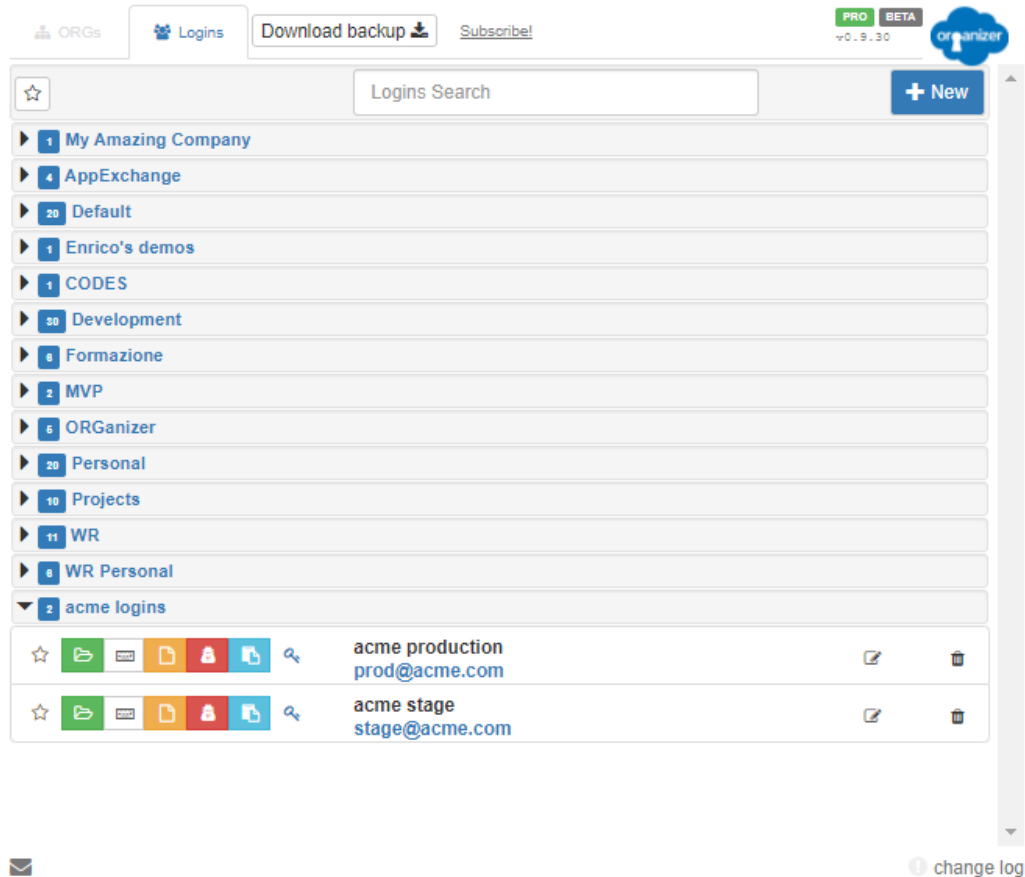
-- Use package's name (acme logins) -- Import Logins

It is possible to select the requested logins and select a logins group (defaults to the backup record name).

Click **Import Logins** to import the selected logins (the extension will require again the secret key to be input):



If you open the ORGanizer for Salesforce extension's popup (click on the extension's icon), you can see the new imported logins:

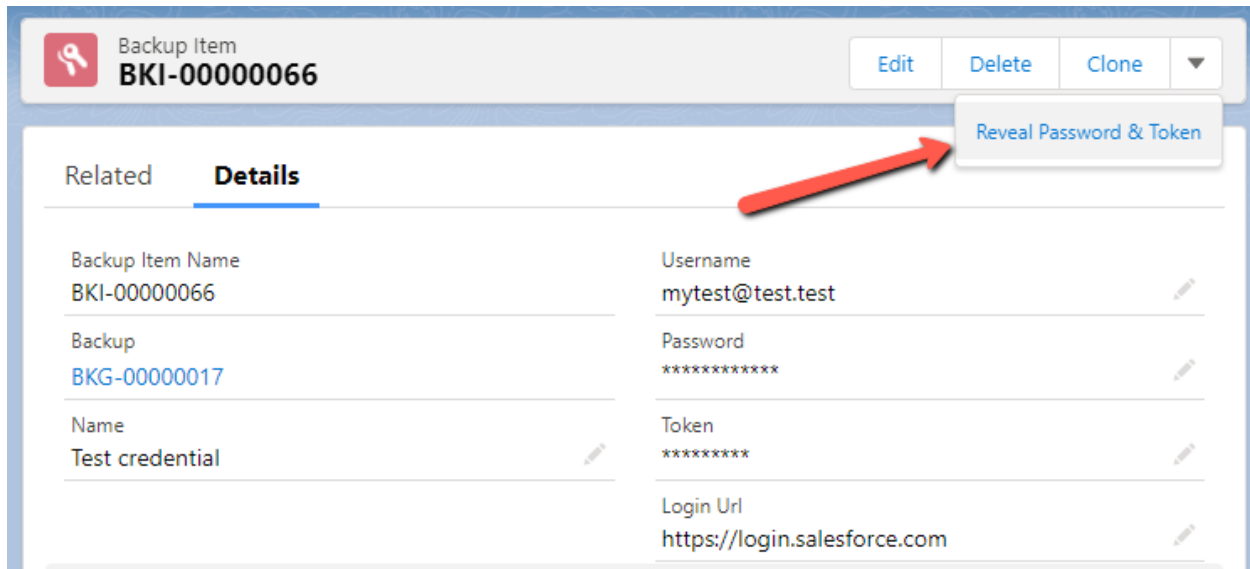


The logins are marked as “not synced” by default (you can change it editing manually each credential) and the key if the username, so they are updated at each synchronization.

Reveal password & token from within the Lightning App

In case ORGanizer for Salesforce browser extension is not used at all, any user that has full access to the Backup can reveal the password and token of any item they have access using the secret.

Open a Backup Item record and select the “Reveal Details” quick action:



Backup Item
BKI-00000066

Edit Delete Clone ▼

Reveal Password & Token

Related **Details**

Backup Item Name	BKI-00000066	Username	mytest@test.test
Backup	BKG-00000017	Password	*****
Name	Test credential	Token	*****
		Login Url	https://login.salesforce.com

The popup requires to input the secret (that you can send via mail using the “Send Secret by Email” action on the Backup record, seen earlier):

Reveal Password & Token

Insert the Backup's secret to reveal this item's password and token

* Secret

.....

Cancel Reveal

By clicking the “Reveal” button the password and token will be temporary shown to the requesting user:



Reveal Password & Token

Here are the requested details:

Password

TestPassword

Token

TestToken

Close

Connector App Webservices: Developer Details

As said earlier, the integration between ORGanizer for Salesforce browser extension and the ORGanizer for Salesforce Connector AppExchange app is delivered out of the box only for extension PRO users.

Nevertheless, the webservices exposed by the Connector App is accessible from outside with no limitations.

From [organizer connector integration script](#) GitHub repository it is possible to download a script that lets users, after configuration, to download and decrypt the data contained in the backups.

This let's users create their own integration if needed and if they don't want to get a PRO license.

The logic is stored into the **index.js** file and uses NodeJS standard Crypto module to handle encryption/decryption of required parameters and results to obtain the item details stored in the ORGanizer for Salesforce Connector app.

Details on configuration are available on the repository.