

## OLCS simplified installation

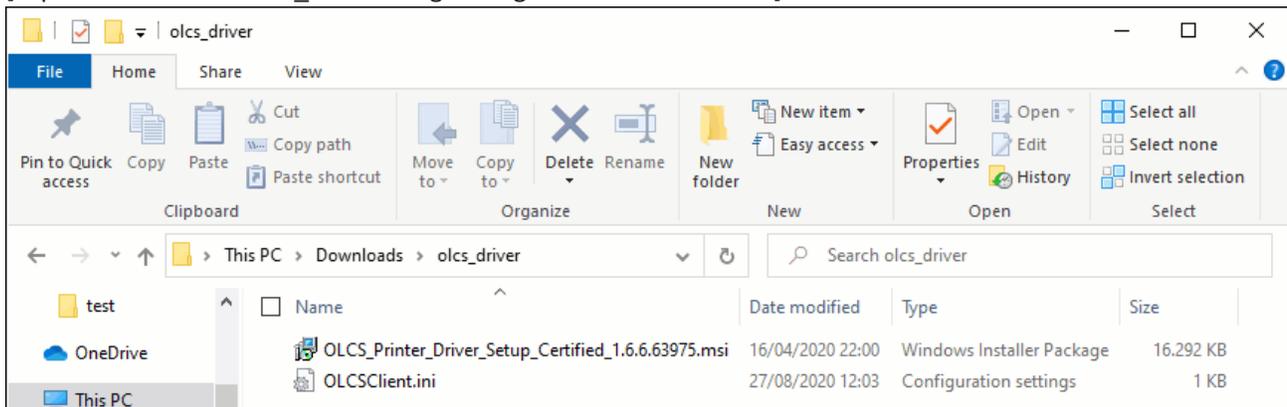
Installation of the OL Connect Send driver is a multi-step process that, although relatively simple, still has a chance of failure. Version 1.7 of OL Connect Send, released with OL Connect 2020.2, has improvements to make the installation more foolproof than before. This article explains what was added, and how this can be used.

### The problem with installing OLCS

The main challenge for a successful installation of the OL Connect client is to correctly apply all the settings for connecting to the server. Get something wrong, and it will not work. To prevent users making typos, we can of course use the preset file; originally meant for silent installations, since 2020.1 (OLCS 1.6.6) this can now also be used with interactive installations.

But working with the preset file comes with a strict recipe: both the installer and the preset file have to sit in the same folder. When both files are downloaded together in a zip, an easy way to mess things up, is to launch the installer from the zip instead of first unpacking the zip. The difference is not always clear to a user.

[caption id="attachment\_33651" align="alignnone" width="872"]



Spot the differences: folder versus zip archive.[/caption]

What would be really nice, is if you only needed to download the installer, and it would be able to get the settings itself. This is now possible with the OLCS 1.7 installer.

Another problem that people sometimes have, is that users can change settings during installation, which can result in a broken installation where the driver cannot reach the server. This is especially problematic for service

providers that onboard external customers; they usually have limited or no access to the client, which makes trouble shooting a lot harder.



Now that the installer can keep control over the downloaded settings, it's possible to prevent users to change things that would cause them to shoot themselves in the foot.

## Settings downloaded by the installer

With version 1.7 of the OL Connect Send client, you can set things up so the user only needs to download the installer itself, and the installer can load the settings from the server. The only thing the user still needs to do, is point out the location to download settings from, which can be a simple copy/paste from the web-site.

The existing ways to install, i.e., loading settings from file, and configuring manually, are still also possible.

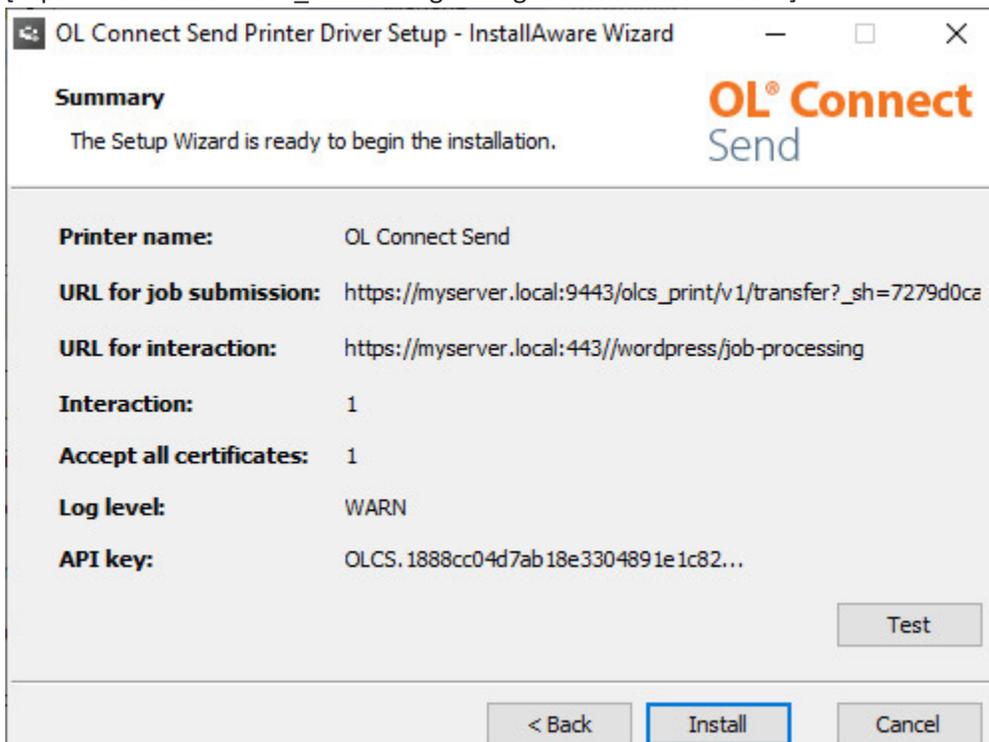
## Locking settings

If the installer downloads the settings itself, they go straight from the server to the installer without the user being able to touch them. We have added the option to lock settings in the installer, so users can also not accidentally break things in the installer.

Before, it was not possible to prevent the user from changing the settings, because a user could always edit the settings file on disk before running the installer.

Locking settings also results in a simplified installation wizard: if settings are locked, the page with communication settings is skipped. Any settings not in the settings file get sensible defaults if possible. If the downloaded settings contain an API key, the page to configure that is also skipped.

[caption id="attachment\_34051" align="aligncenter" width="502"]



The summary page when

settings are locked - no way for a user to break things. [/caption]

To lock settings in the installation wizard, add `LockSettings=1` to the settings file.

## Setting up a server for download URLs

On the server side, the basics of making settings available for download is straightforward. You need two things:

- the same settings file that was used before, now needs to be published on the server;
- the URL to that settings file needs to be presented to the user they can copy/paste it.

In its simplest form, publishing the settings file just means that it has to sit in a public folder of a web server. If all clients use the same URLs, and there are no API keys in the settings file, then this can even be sufficient.

Presenting the URL to the user, can be done on the same page where the driver is downloaded. Copy/pasting can be made even more foolproof with "click to copy" automation.

3. In the installer,
  1. choose to download settings,
  2. and copy/paste the URL below when asked



```
HTTPS://MYSERVER.LOCAL/OLCSCLIENT.INI
```

When implementing the new OLCS client, the new features described above can be used in a number of ways to cater to different needs, and server setups.

### Personalized URLs

If the URLs for submitting jobs or interactive web pages are not the same for every user, then it's necessary to also have a custom URL for downloading settings. It then also makes sense to restrict access to that URL by requiring a user to log in, so it can't be copied by just anyone.

Whether or not to also protect the actual settings file, is a matter of choice. It may also be sufficient to just make the URL "hard to guess".

```
HTTPS://MYSERVER.LOCAL/DOWNLOADS  
/8A56BA13BB3F92A945B836AAFFCAA53020200917113040  
/32493831586AF3D20C6789AE68B07CA620200917113105  
/EB4DFB
```

### API keys

If an API key is included in the settings file that is downloaded, then it is strongly recommended to not only use a URL that is hard to guess, but also require authentication for downloading the settings file itself. Otherwise, the whole purpose of having an API key could be defeated.

The installer can authenticate itself before downloading. This is further explained below. Alternatively, one could omit the API key from the settings file, and have the user copy/paste that separately from their account. As explained above, it is possible to let the user enter the API key, but still prevent them from changing the connection settings.

## Authentication for the installer

To authenticate when downloading settings, the installer supports http BASIC authentication. This is the only form of authentication that is generic enough to be usable when the server is not known up front, while the client cannot be changed, as is the case with our OLCS installer.

As long as the download URL uses https, BASIC authentication can be considered secure enough.

However, if the server uses more advanced authentication methods, then BASIC authentication may not be an option. For instance, if users authenticate through Microsoft, Google, or another *identity provider*, then the server doesn't have the user's password, which it would need to make BASIC authentication work.

In this case the solution is to use techniques like pre-signed URLs. These can be made to require the user to be logged on to the server where they copied the URL from, and the server could even require the user to approve the installers download request from their browser.

[caption id="attachment\_34491" align="alignnone" width="935"]

```
HTTPS://EXAMPLE.COM/SOMETHING? FOO=BAR&BAZ=BARBAZ& X-EMS-ALGORITHM=EMS-
HMAC-SHA256& X-EMS-CREDENTIALS=TH3K3Y%2F20110511%2FUS-
EAST-1%2FHOST%2FAWS4_REQUEST& X-EMS-DATE=20110511T120000Z& X-EMS-
EXPIRES=123456& X-EMS-SIGNEDHEADERS=HOST& X-EMS-
SIGNATURE=FBC9DBB91670E84D04AD2AE7505F4F52AB3FF9E192B8233FEEAE57E9022C2B67
```

Sample of a signed URL, "click to copy" is your friend here. (taken from ~~Security Drops~~)[/caption]

For the user this only makes life easier: the only difference is that they copy/paste a longer URL in the installation wizard, and they don't need to type a username or password. Obviously, implementing this on the server side takes more effort than the straightforward BASIC authentication.

Whatever method is used, when authentication is in play, the download URL should *always* use https.



- Why does the user even need to copy paste that download URL? I know a desktop app that can find its online server by itself.
  - In case of OLCS, we provide a standard client, but the server differs per customer. So we needed a way to have every installation get its settings from a different server.
- Do these features now mean that it is impossible for a user to get the settings file?
  - No. You can still paste the URL in your browser, or a tool like Postman, and that will get you the settings file. These improvements are meant to make installation foolproof, not to block cleverness.
- The download feature seems to be for interactive installation with the setup wizard. What about silent installation?

- The silent installer still needs the settings file present on disk. Silent installation is meant for system administrators who need to deploy the OLCS driver on (many) desktops in their network. These people are expected to be able to handle the settings file.
- What if I want to facilitate both the simplified installation with settings downloaded by the installer, and silent installation that needs a settings file on disk?
  - In that case, the web server needs to provide both options. Just add a link (i.e., "Are you an IT administrator?") on the regular driver download page, and point to a second download page with advanced options.
- Does `LockSettings` also work with a local settings file on disk?
  - Yes. But that of course does not block a user from editing that file manually before running the installer.