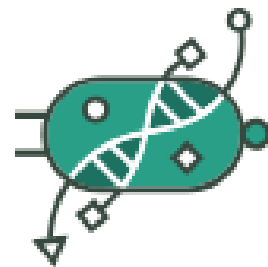


Jun 07, 2024

Setting and Customizing OT-2 for LAP Entries

DOI

dx.doi.org/10.17504/protocols.io.3byl49jxjgo5/v1



Ana Mariya Anhel¹, Lorea Alejaldre¹, Ángel Goñi-Moreno¹

¹Centro de Biotecnología y Genómica de Plantas, Universidad Politécnica de Madrid (UPM)-Instituto Nacional de Investigación y Tecnología Agraria y Alimentaria (INIA/CSIC), Madrid, Spain



Biocomputation Lab

CNB-CSIC

Create & collaborate more with a free account

Edit and publish protocols, collaborate in communities, share insights through comments, and track progress with run records.

Create free account

OPEN  ACCESS



DOI: <https://dx.doi.org/10.17504/protocols.io.3byl49jxjgo5/v1>

Protocol Citation: Ana Mariya Anhel, Lorea Alejaldre, Ángel Goñi-Moreno 2024. Setting and Customizing OT-2 for LAP Entries. **protocols.io** <https://dx.doi.org/10.17504/protocols.io.3byl49jxjgo5/v1>

License: This is an open access protocol distributed under the terms of the **[Creative Commons Attribution License](#)**, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited



Protocol status: Working

We use this protocol and it's working

Created: May 16, 2024

Last Modified: June 07, 2024

Protocol Integer ID: 99941

Keywords: experiments with the lap entry script, lap entries automation script, lap entry script, custom labware to the ot app, lap repository, experimental validation protocol, lap entry run, lap repository entry, files between the robot, uploading custom labware, ot app, running experiment, more information on each script, robot, essential for execution, robot system, documentation, experimental process, execution

Funders Acknowledgements:

Comunidad de Madrid

Grant ID: Y2020/TCS-6555,2019-T1/BIO-14053

MCIN/AEI

Grant ID: CEX2020-000999-S,PID2020-117205GA-I00

ERC

Grant ID: 101044360

Abstract

Automation scripts such as those included in the LAP repository (<https://www.laprepo.cbgp.upm.es/repository/>) require set up steps that not directly part of the experimental process but are essential for execution.

This protocol is divided into 5 sections: connecting to an OT-2, transferring files between the robot and the user system, creating and uploading custom labware to the OT app and robot system and preparing the robot system. These steps establish the necessary environment for running experiments with the LAP entry scripts and user-specific variables.

Not all sections are required for every LAP entry run. For more information on each script in the LAP repository, visit www.laprepo.com and refer to the documentation attached to each entry. LAP repository entries can contain links to protocols.io entry to provide a step by step experimental validation protocol.

Troubleshooting

Before start

This protocol consists of various sections with interlinked steps to transfer files to an OT-2 robot such as variables files, script or customized labware. It is not always necessary to complete all of them to run a LAP entry protocol (i.e your protocol does not use customized labware)

Connect to the OT-2

1 Creation ssh-key connection

An OT-key should be generated to connect to the robot, and it is done with the ssh-keygen command and transferring the public key to the OT.

For more information about how to generate and set the connection between your computer and the Opentrons robot, visit

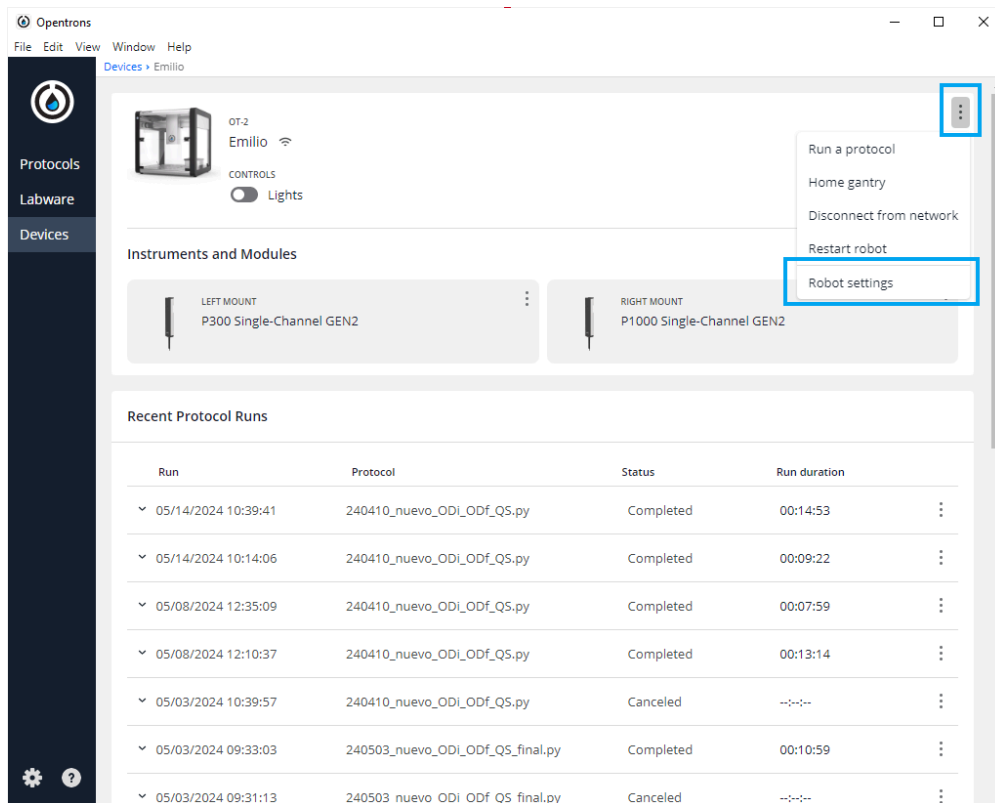
<https://support.opentrons.com/s/article/Setting-up-SSH-access-to-your-OT-2>

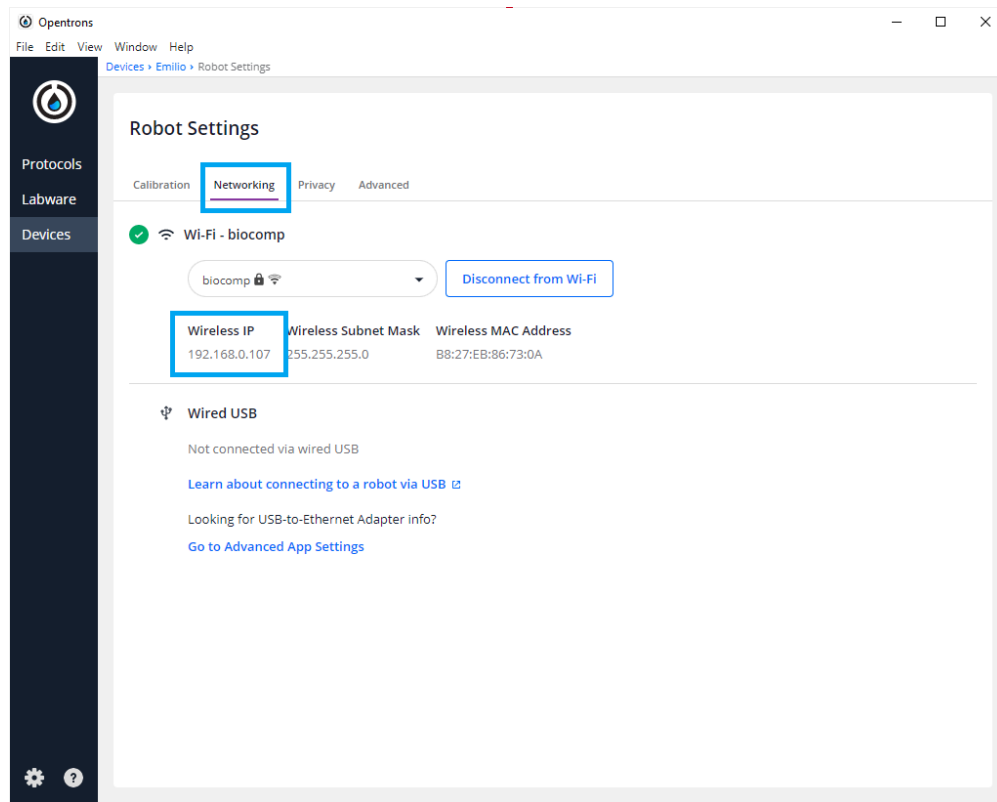
2 Connect to the robot

To connect to the robot's system to perform different actions such as installing packages, check files that are in the system, etc.

2.1 Find the IP of the robot

In the OT app go to *Devices* → *Robot you want to connect* → *Superior 3 dots* → *Robot Settings* → *Networking*





Note

Take into account that the IP of the robot could change from time to time or if you change the network where the robot is connected so it is a good practise to check the IP regularly

Note

In case you are physically connected to the robot via USB the IP will appear in the same window but under the section **Wired USB**

2.2 *Connect to the robot*

Either in a bash (unix systems) or powershell (windows systems) use the command ssh



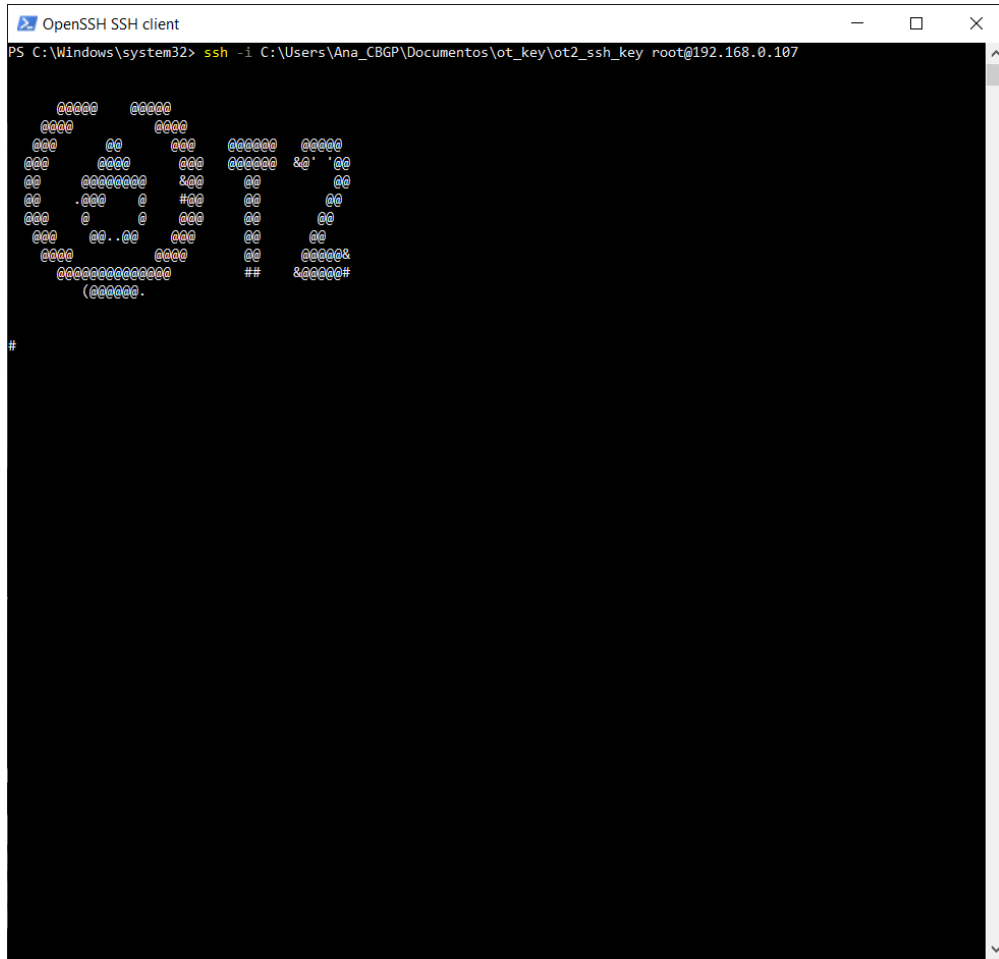
Command

Connect to Linux based OT via ssh

Connect to Linux based OT via ssh

```
ssh -i [path ot_key] root@[Robot_IP]
```

Expected result



Window in a Powershell that shows an OT logo which is a sign of being inside of the robot system

Transferring files between user system and robot system

3 Connect to the robot

You can either connect on the same inalambric network as the robot or by USB

4 Find the IP of the robot

➡ go to step #2.1

5 Transfer the files

In different systems you can/must transfer the files in different ways always taking in account that the system of the OT-2 robot is a Linux Raspberry

5.1 *MacOS/Linux*

We can use the bash or command line to transfer the file(s) with the following command

Command

Passing Files to OT

Passing Files to OT

```
scp -i [ot_key] [file_to_send] root@[IP_OT]:[final_path]
```

The ot_key is referring to the private key created in step 1, [➡ go to step #1](#)

You can face difficulties transferring files in MacOS Ventura (13) and Sonoma (14). These problems can be solved by adding the argument -O (uppercase o) to the command

Command

Transferring files to OT (MacOS 13 and 14)

Transferring files to OT

```
scp -O -i [ot_keypath] [file path to send] root@[IP_robot]:  
[final_path]
```

5.2 *Windows*

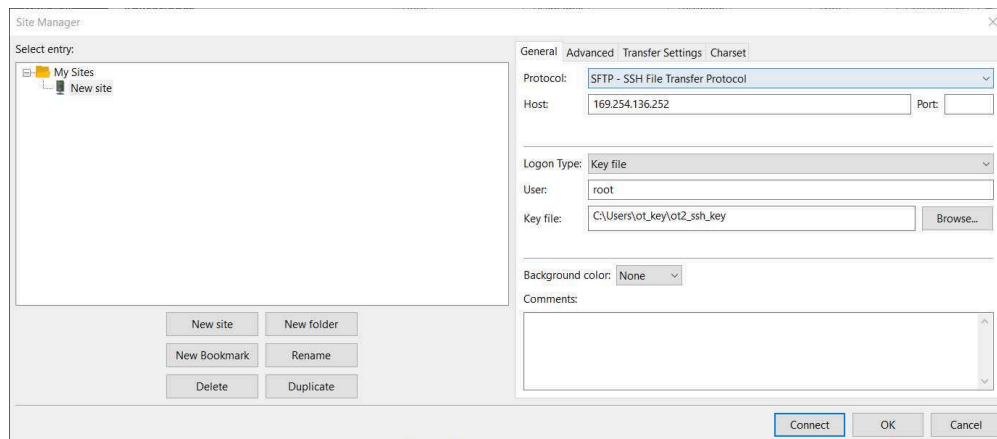
There are several ways to send files from a Windows to a Linux system (for example, with a virtual machine or Windows Powershell in the latest versions of Windows). With the latter you can use the same steps as in 5.1 [↗ go to step #5.1](#)

Another way is using **FileZilla** (<https://filezilla-project.org/download.php?type=client>).

You can set the connection between your computer and the OT-2 robot system as follows.

Open Filezilla → **File (in the toolbar) → Site Manager → New Site (Press button) → Rename Site**

Now that the site is created you can see on the right the settings of that particular site. In the General tab **Change Protocol to SFTP**. Then, introduce in Host the OT IP, change the **Logon Type to key file**, change **User to root** and give the directory where the OT key is. It should look something like the following image



Example of setting the FileZilla to transfer files from Windows (our computer) to Linux (OT)

Then press **Connect**, and we will have a connection between our computer and the robot.

After this connection, we should be able to move the files that we want to the final path of the robot that we desire to send them.

This method can be used as well in some *Linux* and *MacOS* systems

6 Exit the robot

To return to your system you only need to type *exit* in the command line



Preparing Robot OS

7 Check needed packages and versions of LAP entry

In the *Requirements Section* of each LAP Entry you can find that the protocol/entry needs more packages than the ones are installed in the OT-2 system

8 Check python packages of robot

To know which packages are needed to be installed in the opentrons robot you can connect to the robot () and get a list of all the packages and their versions installed in that specific python

Take in account that there are some packages like math or random which versions are specified or linked to the python version

8.1 Connect to robot

⇒ [go to step #2](#)

8.2 Use pip command

You can use the pip command to get the entire packages that are installed in your robot's version of python. Take in account that the packages that are by default in python such as math will not appear in that list

Command

List installed pip packages (Unix)

List installed pip packages

```
pip list
```

To store the packages of the robot in a file for the future you can use the pipe >

Command

List installed pip packages and save (Unix)

List installed pip packages and save

```
pip list > [name_file_store_packages]
```

If you want to store this information in your computer you can transfer that created file

 [go to step #4](#)

You can check for a specific package with the following command

Command

List installed pip specific package (Unix)

List installed pip specific package

```
pip show [name_package]
```

With this last command you can have 2 type of results:

1. A small piece of text giving you the name, version and summary of the package, within other information. This will mean that the package is installed on the python
2. Warning message telling you that the package does not exist. This means that either the package has not been installed in that version of python or that it is not installed with pip (it is by default in python or it is installed with other vias like conda)

9 Install needed packages

To install 1 or more packages you can use any of the following 2 commands



Command

Install package Python

Install package Python

```
python -m pip install [name_package_1] [name_package_2]  
[name_package_3]
```

Command

Install packages with pip

Install packages with pip

```
pip install [name_package_1] [name_package_2] [name_package_3]
```

Make sure that the robot has access to the internet so it can download the packages

Note

For more information about installing packages in the opentrons robots, check the following Opentrons page: <https://support.opentrons.com/s/article/Using-Python-packages-in-Python-API-protocols>

10 Exit the robot

To return to your system you only need to type *exit* in the command line

Uploading custom labware

11 Creation of .json custom labware file

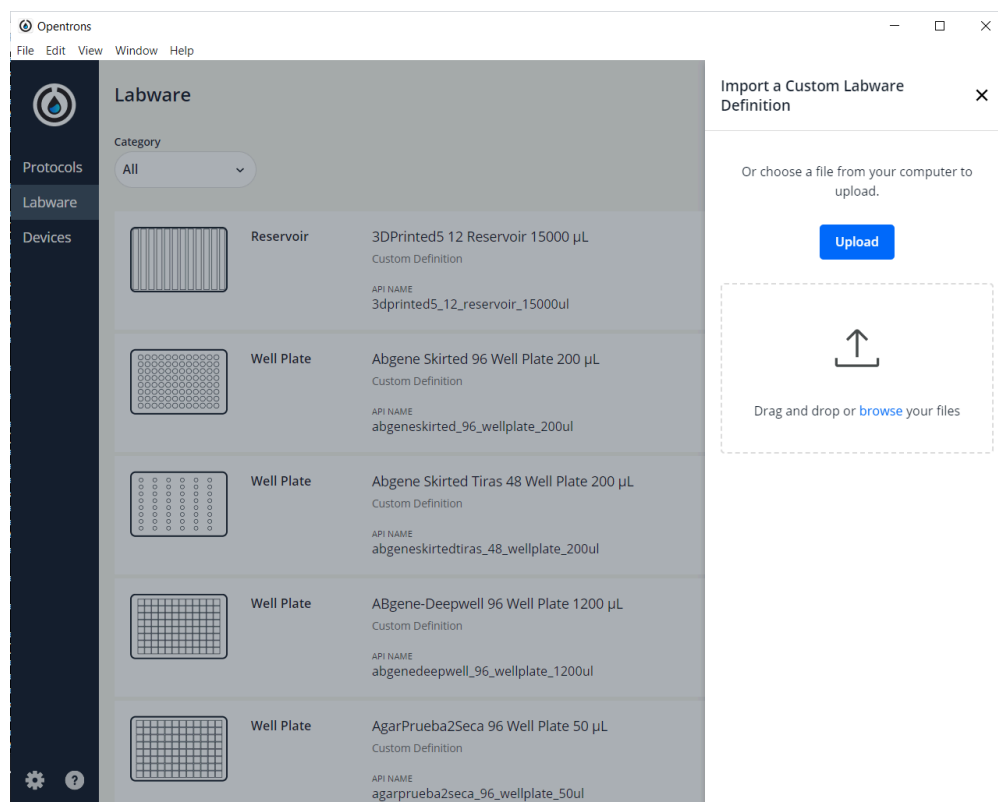
The description file can be obtained by describing the labware dimensions at <https://labware.opentrons.com/create/>

Expected result

You will receive from this page a zip with a .py script in which you can test the labware and a .json file that is the actual definition of the labware

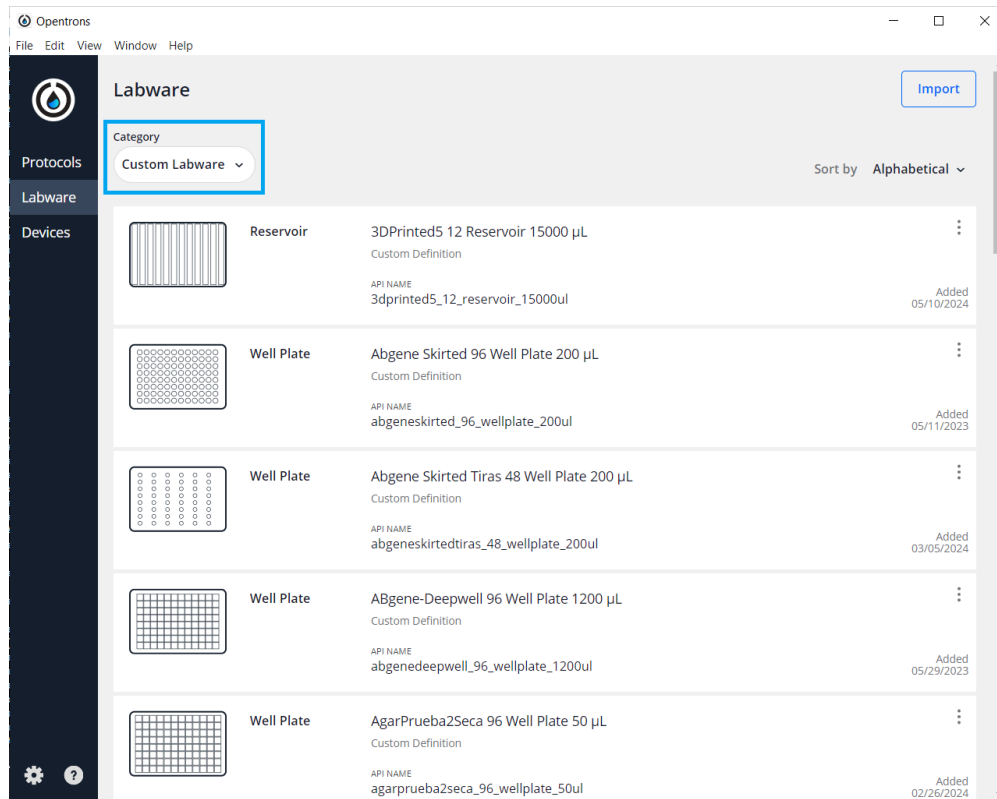
12 Uploading file to the OT App

In the OT app, we need to perform the following route: **Labware** → **Import** → **Choose File** → **Select file we have created in step 9** ➡ [go to step #11](#)



Expected result

After uploading the labware you should be able to see the new labware in the Labware tab of the OT App, all custom labware can be found more easily in the category *Custom Labware*



In some LAP entries or in versions of them there is a need to upload in 2 places the created files so you need to complete as well the following steps.

13 Uploading definition to OT robot system

Note

We do not need to execute this part every time the protocol is used, only when that labware is not included in the OT official labware and the labware file is not in the robot

13.1 *Create a directory with the definition*

We need to create a directory with the API name of the labware, the name that we are going to use to load it in the opentrons

13.2 *Rename .json file*

We need to rename the .json file that the opentrons labware creation to 1.json and place it inside of the directory created in Step 13.1

13.3 *Transfer the directory to a specific location*

To upload the directory with the .json file that represents the custom labware that has been created in step 11 [➡ go to step #11](#) to the robot systems we are going to use the steps from 3 to 6 transferring the file to the final path `/data/labware/v2/custom_definitions/custom_beta` with some changes with the argument `-r` as shown in the following command

Command

Transferring the used custom labware to OT (Linux)

Transferring the used custom labware to OT

```
scp -i [ot_key] -r [custom labware directory path]  
root@[IP_OT] : /data/labware/v2/custom_definitions/custom_beta
```

You can face difficulties transferring files in MacOS Ventura (13) and Sonoma (14). These problems can be solved by adding the argument `-O` (uppercase o) to the command



Command

Transferring the used custom labware to OT (Linux)

Transferring the used custom labware to OT

```
scp -0 -i [ot_key] -r [custom labware directory path]  
root@[IP_OT]:/data/labware/v2/custom_definitions/custom_beta
```