

## Mohamed Labouardy

Software Engineer/DevOps Engineer, 3x AWS Certified Interested in Serverless, Containers, Go, Distributed Systems & NLP.

---



## Butler CLI: Export/Import Jenkins Plugins & Jobs

Not long ago, I had to migrate **Jenkins** jobs from an old server to a new one. That's where [StackOverflow](#) comes into the play, below the most voted answers I found:

- [Jenkins CLI](#)
- Copy the jobs directory
- Jenkins Remote API
- [Jenkins Job Import Plugin](#)

In spite of their advantages, those solutions comes with their downsides especially if you have a large number of **jobs** to move or no access root to the server. But, guess what ? I didn't stop there. I have came up with a **CLI** to make your life easier and export/import not only **Jenkins jobs** but also **plugins** like a boss.

To get started, find the [appropriate package](#) for your system and download it. For

linux:

```
1 wget https://s3.us-east-1.amazonaws.com/butlercli/1.0.0/linux/butler
2 chmod +x butler
3 mv butler /usr/local/bin/
```

**Note:** For **Windows** make sure that **butler** binary is available on the **PATH**. [This page](#) contains instructions for setting the **PATH** on **Windows**.

Once done, verify the installation worked, by opening a new terminal session and checking if **butler** is available :

```
1 butler help
```

```
m1labouardy@moby:~$ butler
NAME:
  butler - Import/Export Jenkins Jobs

USAGE:
  butler [global options] command [command options] [arguments...]

VERSION:
  1.0.0

AUTHOR:
  Mohamed Labouardy <mohamed@labouardy.com>

COMMANDS:
  jobs      Jenkins Jobs Management
  plugins   Jenkins Plugins Management
  help, h   Shows a list of commands or help for one command

GLOBAL OPTIONS:
  --help, -h      show help
  --version, -v   print the version
m1labouardy@moby:~$
```

## 1 – Plugins Management

To export **Jenkins jobs**, you need to provide the **URL** of the source **Jenkins** instance:

```
1 butler plugins export --server localhost:8080 --username admin --password admin
```

```
m1labouardy@moby:~/butler$ butler plugins export --server 34.239.121.54:8080 --username admin --password admin
```

NAME	VERSION	DESCRIPTION
bouncycastle-api	2.16.2	bouncycastle API Plugin
structs	1.10	Structs Plugin
script-security	1.39	Script Security Plugin
scm-api	2.2.6	SCM API Plugin
workflow-step-api	2.14	Pipeline: Step API
workflow-api	2.24	Pipeline: API
workflow-support	2.16	Pipeline: Supporting APIs
durable-task	1.17	Durable Task Plugin
workflow-durable-task-step	2.17	Pipeline: Nodes and Processes
credentials	2.1.16	Credentials Plugin
ssh-credentials	1.13	SSH Credentials Plugin
plain-credentials	1.4	Plain Credentials Plugin
credentials-binding	1.13	Credentials Binding Plugin
gradle	1.28	Gradle Plugin
pipeline-input-step	2.8	Pipeline: Input Step

Plugin Name	Version	Full Name
apache-httpcomponents-client-4-api	4.5.3-2.0	Jenkins Apache HttpComponents Client 4.x API Plugin
junit	1.23	JUnit Plugin
windows-slaves	1.3.1	Windows Slaves Plugin
display-url-api	2.2.0	Display URL API
mailer	1.20	Jenkins Mailer Plugin
matrix-auth	2.2	Matrix Authorization Strategy Plugin
antisamy-markup-formatter	1.5	OWASP Markup Formatter Plugin
matrix-project	1.12	Matrix Project Plugin
jsch	0.1.54.1	Jenkins JSch dependency plugin
git-client	2.7.0	Jenkins Git client plugin
pam-auth	1.3	PAM Authentication plugin
authentication-tokens	1.3	Authentication Tokens API Plugin
docker-commons	1.11	Docker Commons Plugin
ace-editor	1.1	JavaScript GUI Lib: ACE Editor bundle plugin
jquery-detached	1.2.1	JavaScript GUI Lib: jQuery bundles (jQuery and jQuery UI) plugin
workflow-scm-step	2.6	Pipeline: SCM Step
workflow-cps	2.42	Pipeline: Groovy
docker-workflow	1.14	Docker Pipeline
jackson2-api	2.8.10.1	Jackson 2 API Plugin
github-api	1.90	GitHub API Plugin
git	3.7.0	Jenkins Git plugin
workflow-job	2.12.2	Pipeline: Job
token-macro	2.3	Token Macro Plugin
github	1.28.1	GitHub plugin

As shown above, **butler** will dump a list of plugins installed to stdout and a new file *plugins.txt* will be generated, with list of installed **Jenkins plugins** with *name* and *version* pairs:

```

1 ouncycastle-api@2.16.2
2 structs@1.10
3 script-security@1.39
4 scm-api@2.2.6
5 workflow-step-api@2.14
6 workflow-api@2.24
7 workflow-support@2.16
8 durable-task@1.17
9 workflow-durable-task-step@2.17
10 credentials@2.1.16
11 ssh-credentials@1.13
12 plain-credentials@1.4
13 credentials-binding@1.13
14 gradle@1.28
15 pipeline-input-step@2.8
16 apache-httpcomponents-client-4-api@4.5.3-2.0
17 junit@1.23
18 windows-slaves@1.3.1
19 display-url-api@2.2.0
20 mailer@1.20
21 matrix-auth@2.2
22 antisamy-markup-formatter@1.5
23 matrix-project@1.12
24 jsch@0.1.54.1
25 git-client@2.7.0
26 pam-auth@1.3
27 authentication-tokens@1.3
28 docker-commons@1.11
29 ace-editor@1.1
30 jquery-detached@1.2.1
31 workflow-scm-step@2.6
32 workflow-cps@2.42
33 docker-workflow@1.14
34 jackson2-api@2.8.10.1
35 github-api@1.90
36 git@3.7.0
37 workflow-job@2.12.2

```

```
38 token-macro@2.3
39 github@1.28.1
```

Now, to import the **plugins** to the new Jenkins instance, use the command below with the **URL** of the Jenkins target instance as an argument:

```
1 butler plugins import --server localhost:8080 --username admin --password admin
```

```
mLabourady@moby:~/butler$ butler plugins import --server 34.239.121.54:8081 --username admin --password admin
Installing bouncycastle-api@2.16.2
Installing structs@1.10
Installing script-security@1.39
Installing scm-api@2.2.6
Installing workflow-step-api@2.14
Installing workflow-api@2.24
Installing workflow-support@2.16
Installing durable-task@1.17
Installing workflow-durable-task-step@2.17
Installing credentials@2.1.16
Installing ssh-credentials@1.13
Installing plain-credentials@1.4
Installing credentials-binding@1.13
Installing gradle@1.28
Installing pipeline-input-step@2.8
Installing apache-httpcomponents-client-4-api@4.5.3-2.0
Installing junit@1.23
Installing windows-slaves@1.3.1
Installing display-url-api@2.2.0
Installing mailer@1.20
Installing matrix-auth@2.2
Installing antisamy-markup-formatter@1.5
Installing matrix-project@1.12
Installing jsch@0.1.54.1
Installing git-client@2.7.0
Installing pam-auth@1.3
Installing authentication-tokens@1.3
Installing docker-commons@1.11
Installing ace-editor@1.1
Installing jquery-detached@1.2.1
Installing workflow-scm-step@2.6
Installing workflow-cps@2.42
Installing docker-workflow@1.14
Installing jackson2-api@2.8.10.1
Installing github-api@1.90
Installing git@3.7.0
Installing workflow-job@2.12.2
Installing token-macro@2.3
Installing github@1.28.1
```

**Butler** will install each plugin on the target Jenkins instance by issuing **API calls**.

## 2 – Jobs Management

The screenshot shows the Jenkins web interface. The top navigation bar includes the Jenkins logo, a search bar, and user information (admin | log out). The main content area displays a table of jobs with columns for Status (S), Weather (W), Name, Last Success, Last Failure, and Last Duration. The jobs listed are Backend, Deployment, Frontend, and Microservice. The Frontend job shows a last failure of 1 min 20 sec - #1. Below the table, there are icons for S, M, and L. At the bottom of the page, it says 'Page generated: Jan 5, 2018 8:12:38 PM UTC REST API Jenkins ver. 2.60.3'.

S	W	Name ↓	Last Success	Last Failure	Last Duration
		Backend	56 sec - #1	N/A	58 ms
		Deployment	23 sec - #1	N/A	15 ms
		Frontend	N/A	1 min 20 sec - #1	1.7 sec
		Microservice	N/A	N/A	N/A

To export **Jenkins jobs**, just provide the **URL** of the source **Jenkins** server:

```
1 butler jobs export --server localhost:8080 --username admin --password admin
```

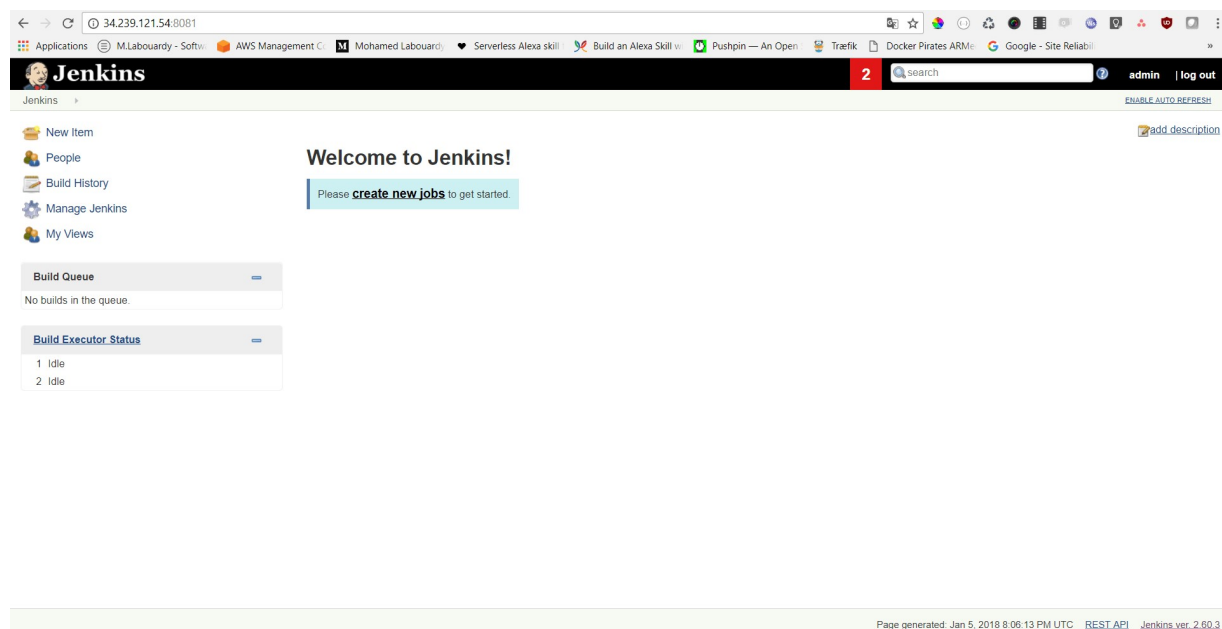
```
m1labouardy@moby:~/butler$ butler jobs export --server 34.239.121.54:8080 --username admin --password admin
Exporting job: Backend
Exporting job: Deployment
Exporting job: Frontend
Exporting job: Microservice
m1labouardy@moby:~/butler$ ls
jobs
m1labouardy@moby:~/butler$
```

A new directory **jobs/** will be created with every job in **Jenkins**. Each job will have its own configuration file *config.xml*.

```
m1labouardy@moby:~/butler$ tree jobs/
jobs/
├── Backend
│   └── config.xml
├── Deployment
│   └── config.xml
├── Frontend
│   └── config.xml
└── Microservice
    └── config.xml

4 directories, 4 files
m1labouardy@moby:~/butler$
```

Now, to import the jobs to the new **Jenkins** instance, issue the following command:



The screenshot shows the Jenkins web interface in a browser. The address bar displays the URL `34.239.121.54:8081`. The page title is "Jenkins". The main content area displays "Welcome to Jenkins!" with a button that says "Please create new jobs to get started". On the left sidebar, there are navigation links for "New Item", "People", "Build History", "Manage Jenkins", and "My Views". Below these are sections for "Build Queue" (showing "No builds in the queue.") and "Build Executor Status" (showing two "Idle" executors). The footer of the page indicates "Page generated: Jan 5, 2018 8:06:13 PM UTC" and "Jenkins ver. 2.60.3".

```
1 butler jobs import --server localhost:8080 --username admin --password admin
```

```
m1labouardy@moby:~/butler$ butler jobs import --server 34.239.121.54:8081 --username admin --password admin
Import job: Backend
```

```

import job: Backend
Import job: Deployment
Import job: Frontend
Import job: Microservice
mlabourdy@moby:~/butler$

```

**Butler** will use the configuration files created earlier to issue API calls to target **Jenkins** instance to create jobs.

Once you are done, check **Jenkins** and you should see your jobs successfully created :

S	W	Name ↓	Last Success	Last Failure	Last Duration
	☀	Backend	N/A	N/A	N/A
	☀	Deployment	N/A	N/A	N/A
	☀	Frontend	N/A	N/A	N/A
	☀	Microservice	N/A	N/A	N/A

Build Queue: No builds in the queue.

Build Executor Status: 1 Idle, 2 Idle.

Page generated: Jan 5, 2018 8:15:49 PM UTC | REST API | Jenkins ver. 2.60.3

Hope it helps ! The CLI is still in its early stages, so you are welcome to contribute to the project in [Github](#).

#### SHARE THIS:



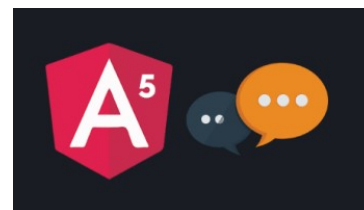
#### RELATED



**Cleanup old Docker images from Nexus Repository**  
In "CI"



**Publish Custom Metrics to AWS CloudWatch**  
In "AWS"



**Chatbot with Angular 5 & DialogFlow**  
In "Angular"

## Comments

0 comments




0 Comments

Sort by **Newest**



Add a comment...

[Facebook Comments Plugin](#)

 January 5, 2018  Mohamed Labouardy  CI, DevOps, Jenkins

Designed by Mohamed Labouardy