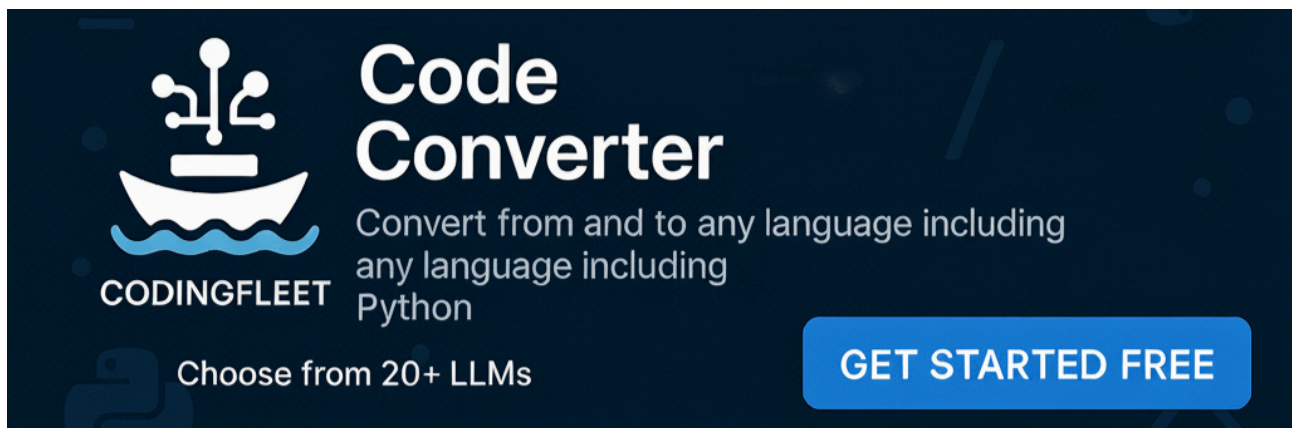


How to Execute Shell Commands in a Remote Machine in Python - The Python Code

By Abdeladim Fadheli

Published: 2019-11-11 · Archived: 2026-04-05 22:43:47 UTC

[Abdeladim Fadheli](#) · 4 min read · Updated may 2024 · [Ethical Hacking](#)



The advertisement banner features a dark blue background with a white icon of a boat carrying a circuit board. To the right of the icon, the text reads 'Code Converter' in large white font, followed by 'Convert from and to any language including any language including Python' in smaller white font. Below the icon, the text 'CODINGFLEET' is written in white. At the bottom left, it says 'Choose from 20+ LLMs'. On the right side, there is a blue button with white text that says 'GET STARTED FREE'.

Get a head start on your coding projects with our [Python Code Generator](#). Perfect for those times when you need a quick solution. Don't wait, try it today!

Have you ever wanted to quickly execute certain commands remotely on your Linux machine? or do you want to routinely execute some lines of code in your server to automate stuff? In this tutorial, you will learn how to write a simple Python script to remotely execute shell commands on your Linux machine.

RELATED: [How to Brute-Force SSH Servers in Python](#).

We will be using the paramiko library; let's install it:

```
pip3 install paramiko
```

Defining some connection credentials:

```
import paramiko

hostname = "192.168.1.101"
username = "test"
password = "abc123"
```

In the above code, I've defined the hostname, username, and password, this is my local Linux box, you need to edit these variables for your case, or you may want to make command-line argument parsing using the argparse

module as we usually do in such tasks.

Note that it isn't safe to connect to SSH using credentials like that. You can configure your SSH listener daemon to accept only public authentication keys instead of using a password. However, for demonstration purposes, we will be using a password.

Executing Shell Commands

Now, let's create a list of commands you wish to execute on that remote machine:

```
commands = [  
    "pwd",  
    "id",  
    "uname -a",  
    "df -h"  
]
```

In this case, simple commands output useful information about the operating system.

The below code is responsible for initiating the SSH client and connecting to the server:

```
# initialize the SSH client  
client = paramiko.SSHClient()  
# add to known hosts  
client.set_missing_host_key_policy(paramiko.AutoAddPolicy())  
try:  
    client.connect(hostname=hostname, username=username, password=password)  
except:  
    print("[!] Cannot connect to the SSH Server")  
    exit()
```

Now let's iterate over the commands we just defined and execute them one by one:

```
# execute the commands  
for command in commands:  
    print("="*50, command, "="*50)  
    stdin, stdout, stderr = client.exec_command(command)  
    print(stdout.read().decode())  
    err = stderr.read().decode()  
    if err:  
        print(err)
```

Here are my results:

```
===== pwd =====  
/home/test  
  
===== id =====  
uid=1000(test) gid=0(root) groups=0(root),27(sudo)  
  
===== uname -a =====  
Linux rockikz 4.17.0-kali1-amd64 #1 SMP Debian 4.17.8-1kali1 (2018-07-24) x86_64 GNU/Linux  
  
===== df -h =====  
Filesystem      Size  Used Avail Use% Mounted on  
udev            1.9G   0  1.9G   0% /dev  
tmpfs           392M  6.2M  386M   2% /run  
/dev/sda1       452G  410G   19G  96% /  
tmpfs           2.0G   0  2.0G   0% /dev/shm  
tmpfs           5.0M   0  5.0M   0% /run/lock  
tmpfs           2.0G   0  2.0G   0% /sys/fs/cgroup  
tmpfs           392M  12K  392M   1% /run/user/131  
tmpfs           392M   0  392M   0% /run/user/1000
```

Awesome, these commands were successfully executed on my Linux machine!

[Related: Build 35+ Ethical Hacking Scripts & Tools with Python EBook](#)

Executing Scripts

Now that you know how to execute commands one by one, let's dive a little bit deeper and execute entire shell (.sh) scripts.

Consider this script (named "script.sh"):

```
cd Desktop  
mkdir test_folder  
cd test_folder  
echo "$PATH" > path.txt
```

After the SSH connection, instead of iterating for commands, now we read the content of this script and execute it:

```
# read the BASH script content from the file  
bash_script = open("script.sh").read()  
# execute the BASH script  
stdin, stdout, stderr = client.exec_command(bash_script)  
# read the standard output and print it  
print(stdout.read().decode())  
# print errors if there are any
```

```
err = stderr.read().decode()
if err:
    print(err)
# close the connection
client.close()
```

exec_command() method executes the script using the default shell (BASH, SH, or any other) and returns standard input, standard output, and standard error, respectively. We will read from stdout and stderr if there are any, and then we will close the SSH connection.

After the execution of the above code, a new file test_folder was created in Desktop and got a text file inside that which contained the global \$PATH variable:

```
fullclip@rockikz:~/Desktop$ ls
HELLO test_folder
fullclip@rockikz:~/Desktop$ cd test_folder/
fullclip@rockikz:~/Desktop/test_folder$ ls
path.txt
fullclip@rockikz:~/Desktop/test_folder$ cat path.txt
/usr/local/bin:/usr/bin:/bin:/usr/games
fullclip@rockikz:~/Desktop/test_folder$
```

[GET -10% OFF: Build 35+ Ethical Hacking Scripts & Tools with Python EBook](#)

Conclusion

As you can see, this is useful for many scenarios. For example, you may want to manage your servers only by executing Python scripts remotely; you can do anything you want!

And by the way, If you want to run more complex jobs on a remote server, you might want to look into [Ansible](#) instead.

You can also use [Fabric](#) library, as it is a high-level Python library designed just to execute shell commands remotely over SSH. It builds on top of [Invoke](#) and [Paramiko](#).

Feel free to edit the code as you wish; for example, you may want to parse command-line arguments with argparse.

If you're into cyber security, then I highly encourage you to take [our Ethical Hacking with Python EBook](#), where we build 35+ hacking tools and scripts from scratch using Python!

READ ALSO: [How to Create a Reverse Shell in Python](#).

Happy Coding ♥

Finished reading? Keep the learning going with our [AI-powered Code Explainer](#). Try it now!

[View Full Code](#) [Analyze My Code](#)

Sharing is caring!

Read Also

```
client = paramiko.SSHClient()
# add to know hosts
client.set_missing_host_key_policy(paramiko.AutoAddPo
try:
    client.connect(hostname=hostname, username=username
except socket.timeout:
    # this is when host is unreachable
    print(f"{RED}[!] Host: {hostname} is unreachable,
    return False
except paramiko.AuthenticationException:
```

```
while True:
    # receive the command from the server
    command = s.recv(BUFFER_SIZE).decode()
    if command.lower() == "exit":
        # if the command is exit, just break out of t
        break
    # execute the command and retrieve the results
    output = subprocess.getoutput(command)
    # send the results back to the server
    s.send(output.encode())
```

```
*] Internal link: https://www.thepythoncode.com/article/building-network-scanner-using-scapy
*] Internal link: https://www.thepythoncode.com/article/control-mouse-python
*] Internal link: https://www.thepythoncode.com/article/generate-read-qr-code-python
*] Internal link: https://www.thepythoncode.com/article/top-python-libraries-for-data-science
!] External link: https://twitter.com/ThePythonCode1
!] External link: https://www.facebook.com/thepythoncode
!] External link: https://www.github.com/x4nth055
*] Internal link: https://www.thepythoncode.com/author/1
!] External link: javascript://void(0);
*] Internal link: https://www.thepythoncode.com/article/getting-started-with-scapy
!] External link: https://scapy.readthedocs.io/en/latest/installation.html
!] External link: https://en.wikipedia.org/wiki/Address_Resolution_Protocol
!] External link: https://en.wikipedia.org/wiki/Broadcasting_(networking)
!] External link: https://en.wikipedia.org/wiki/Classless_Inter-Domain_Routing
!] External link: https://github.com/x4nth055/pythoncode-tutorials/blob/master/scapy/networking.py
*] Internal link: https://www.thepythoncode.com/article/building-arp-spoofers-using-scapy
*] Internal link: https://www.thepythoncode.com/article/make-dns-spoof-python
!] External link: https://en.wikipedia.org/wiki/Security_hacker
!] External link: https://en.wikipedia.org/wiki/Domain_Name_System
```

Comment panel

Source: <https://www.thepythoncode.com/article/executing-bash-commands-remotely-in-python>