Test TCP connectivity with curl

You probably know about curl command: it’s great for downloading web pages or files from a Unix command line. But there’s another great usage curl command has: testing TCP ports connectivity.

Say, you’re helping with some firewall changes and need to confirm that connection from your server to some remote host and specific TCP port still works.

Here’s how you can do it using curl command and its telnet functionality.

Test SSH port connection with curl

In this example, SSH connection works because:

1. We get the “Connected” status
2. We see the SSH version prompt: SSH-2.0-OpenSSH_7.4

greys@server:~ $ curl -v telnet://127.0.0.1:22
* About to connect() to 127.0.0.1 port 22 (#0)
* Trying 127.0.0.1...
* Connected to 127.0.0.1 (127.0.0.1) port 22 (#0)
SSH-2.0-OpenSSH_7.4
^C

Test jBoss port 8080 with curl

This scenario shows that connection is refused (probably because there’s no service running on that port).

IMPORTANT: you would probably get a different message if firewall blocks the 8080 port. Connection refused is a clear sign that port is accessible, but nothing’s responding on it.
See Also

- Using wget and curl
- iptables: keep rules after reboot
- Advanced Unix Commands