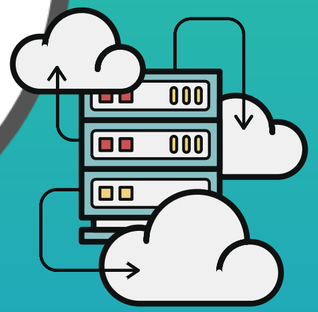
DEVOPS BOOTCAMP

by TWN



Cloud & Infrastructure as a Service (IaaS)

Key Takeaways

TECHWORLD WITH NANA

Throughout the DevOps Bootcamp, we will install and configure

Nexus
Artifact Repository

Jenkins
Build Automation

Deploy own
Applications on Servers

&
more

- We **don't install** it **locally** on our laptop!
- **More realistic**: Install them on remote dedicated servers in cloud

What is Infrastructure as a Service (IaaS)?

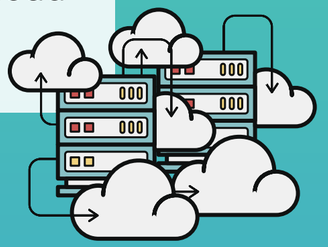
- Offers **compute, storage and networking resources on demand**
- Instead of:

Company buys own server

- You manage own servers and infrastructure
- If something breaks, you need to fix it

Delegate Infrastructure Management

- Move your physical infrastructure to cloud
- You just rent the servers on demand



What is Cloud Computing?

- Cloud Computing is the **delivery of computing services** - including servers, storage, databases, networking, software - **over the internet** ("the cloud")
- **IaaS is 1 of 4 types of cloud services**. Others: Software as a Service (SaaS), Platform as a Service (PaaS) and Serverless

TECHWORLD WITH NANA

Infrastructure as a Service Providers

Most used IaaS Providers

- AWS is the most used : **Powerful, but complex (learn in later module)**

- **DigitalOcean**: Easier to start

Setup Server on DigitalOcean (DO) - 1

to run packaged java application (jar file)

- Servers on DigitalOcean are called "Droplets"
- Droplets are Linux-based virtual machines (VMs)

Summary of Steps:

1. **Pre-Requisite**: Create DigitalOcean account (with new signup credits - free tier)
2. **Configure SSH keys**
3. **Create a Droplet** with Linux Ubuntu distribution
4. **Open SSH port 22** on server using firewall configuration
5. **SSH into the server** using its public IP address
6. **Install Java** to run Java applications on it

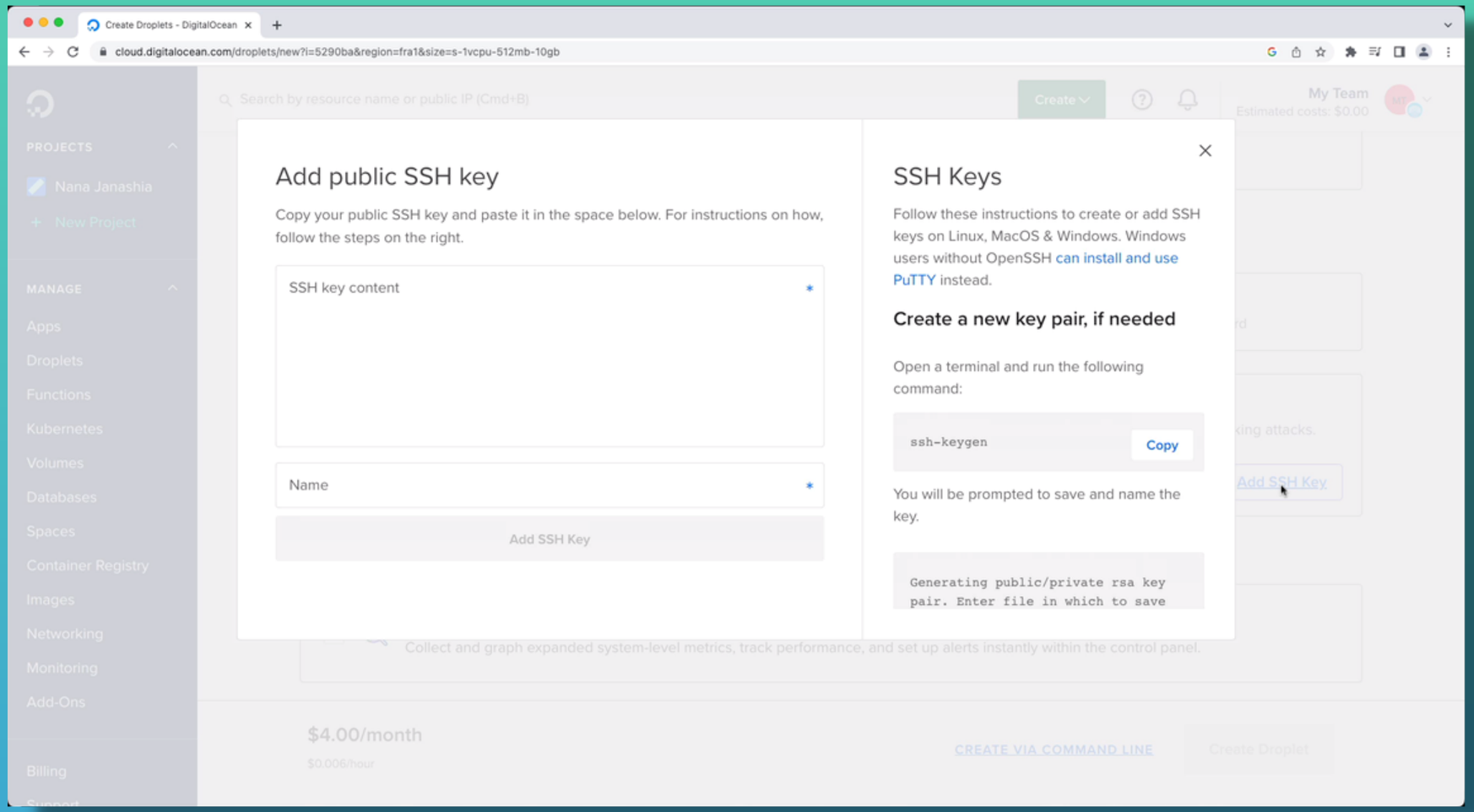
TECHWORLD WITH NANA

Setup Server on DigitalOcean (DO) - 2

to run packaged java application (jar file)

Configure SSH keys

- To be able to access any server on DO from local computer using SSH



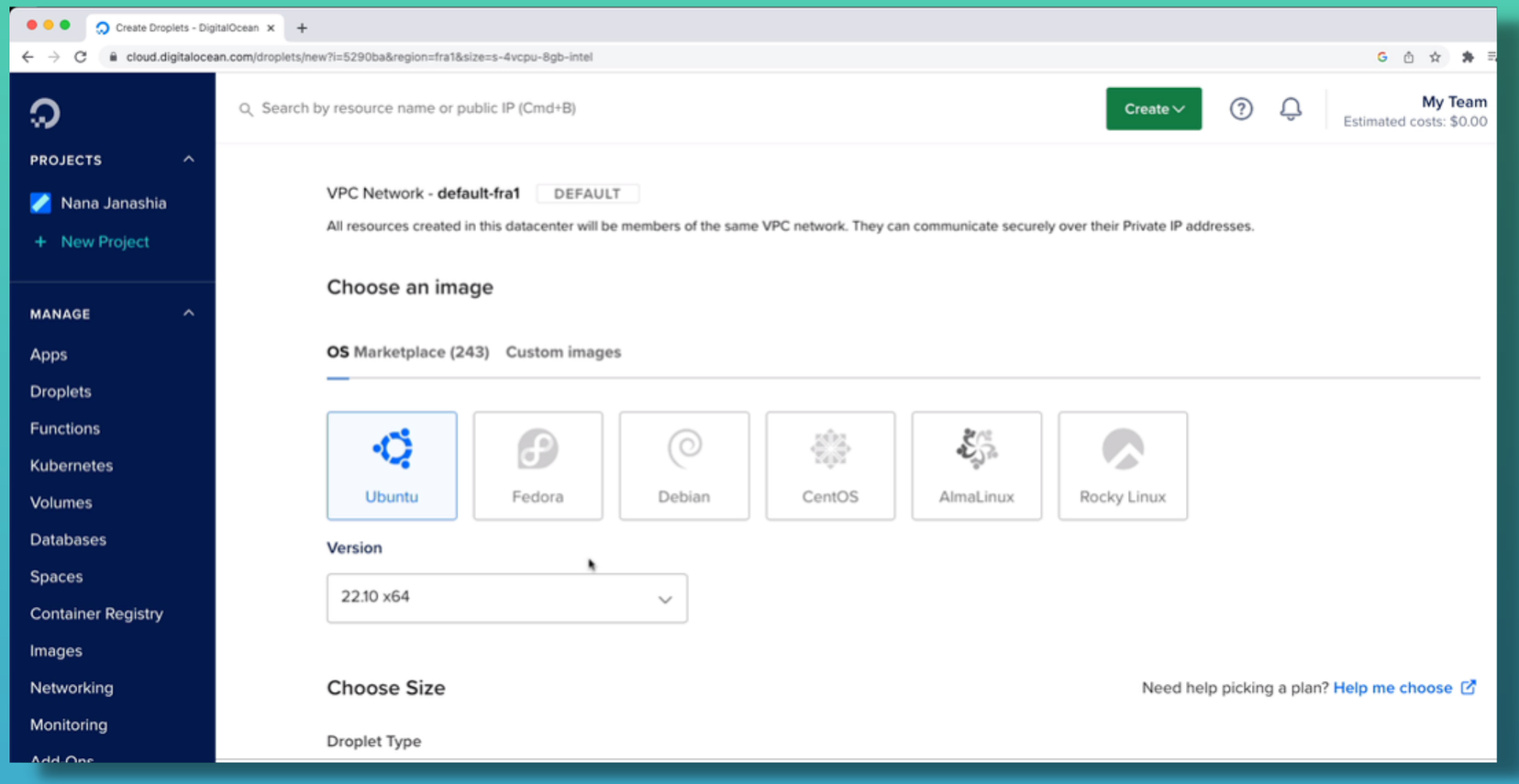
TECHWORLD WITH NANA

Setup Server on DigitalOcean (DO) - 3

to run packaged java application (jar file)

Create a Droplet

- With Linux Ubuntu distribution



TECHWORLD WITH NANA

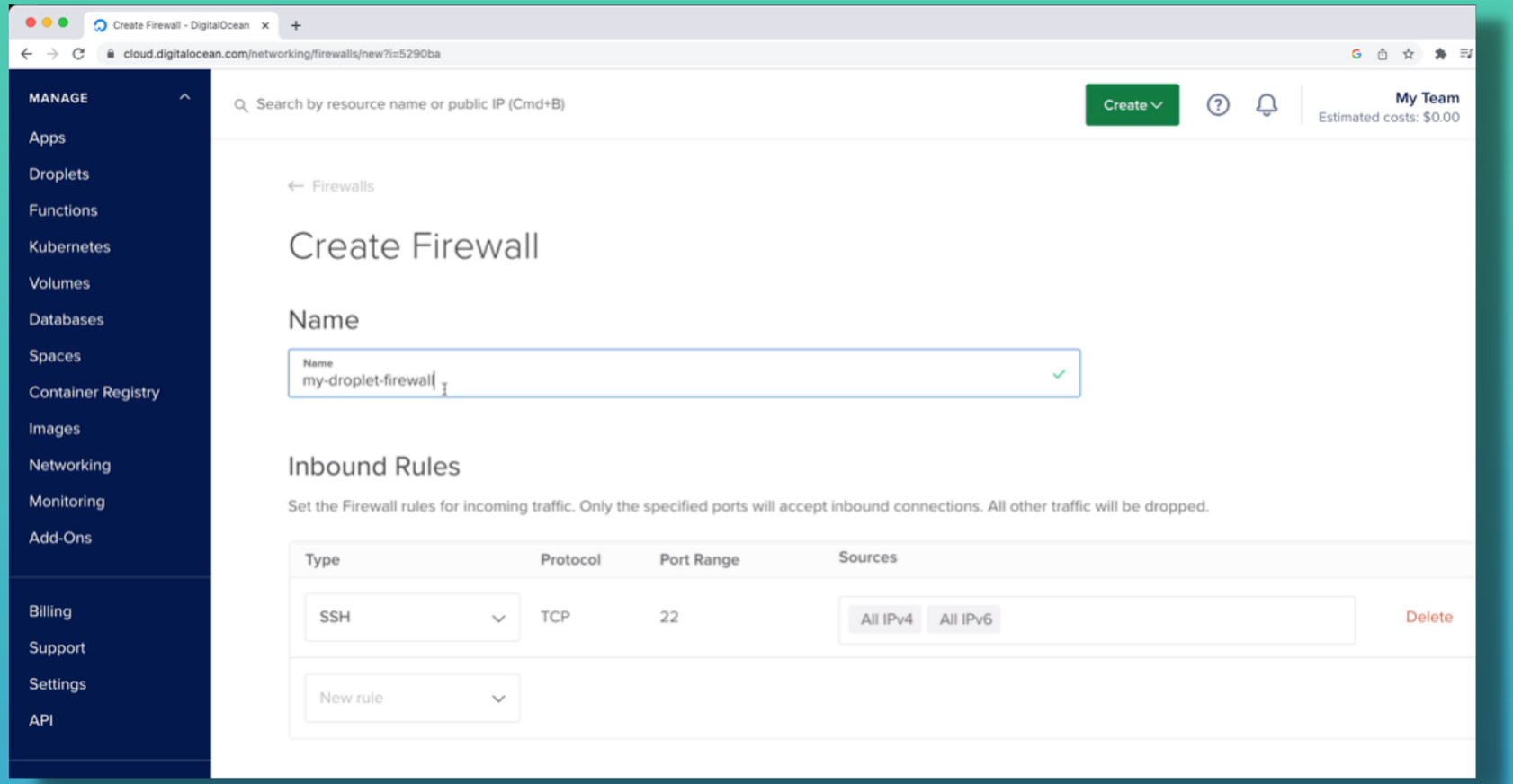
Setup Server on DigitalOcean (DO) - 4

to run packaged java application (jar file)

Open SSH port 22

- Open SSH port 22 on the server by creating a new **Firewall configuration**

- **Inbound** Rules = for incoming traffic
- **Outbound** Rules = for outgoing traffic



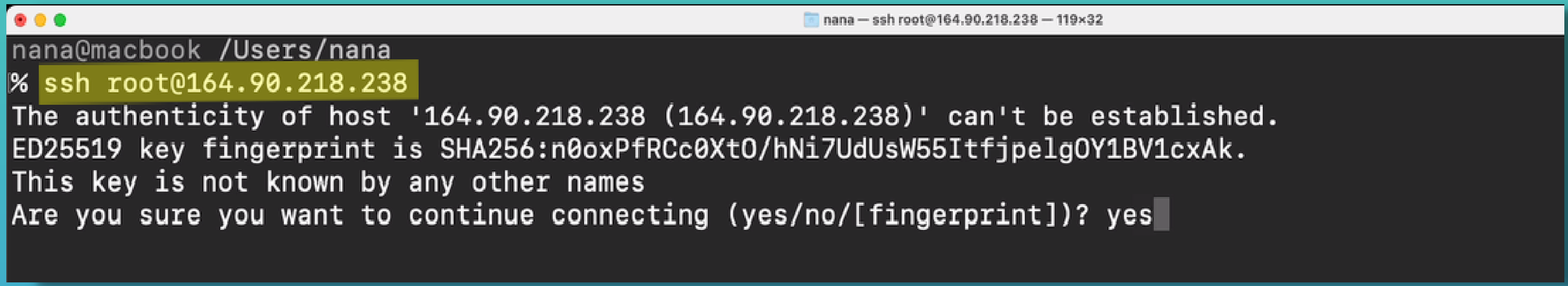
TECHWORLD WITH NANA

Setup Server on DigitalOcean (DO) - 5

to run packaged java application (jar file)

SSH into the server

- By using its public IP address



Setup Server on DigitalOcean (DO) - 6

to run packaged java application (jar file)

Install Java

1. Check if java is installed and get **installation commands**

```
root@ubuntu-s-1vcpu-1gb-fra1-01:~# java

Command 'java' not found, but can be installed with:

apt install openjdk-11-jre-headless  # version 11.0.8+10-0ubuntu1~20.04, or
apt install default-jre              # version 2:1.11-72
apt install openjdk-8-jre-headless   # version 8u265-b01-0ubuntu2~20.04
apt install openjdk-13-jre-headless  # version 13.0.3+3-1ubuntu2
apt install openjdk-14-jre-headless  # version 14.0.1+7-1ubuntu1

root@ubuntu-s-1vcpu-1gb-fra1-01:~# apt update
```

2. **Install java** to run Java applications on it

```
root@ubuntu-s-1vcpu-1gb-fra1-01:~# apt install openjdk-8-jre-headless
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  ca-certificates-java fontconfig-config fonts-dejavu-core java-common libavahi-client3
  libavahi-common-data libavahi-common3 libcups2 libfontconfig1 libjpeg-turbo8 libjpeg8 liblcms2-2
  libnspr4 libnss3 libpcsclite1 libxi6 libxrender1 libxtst6 x11-common
Suggested packages:
  default-jre cups-common liblcms2-utils pcscd libnss-mdns fonts-dejavu-extra fonts-ipafont-gothic
  fonts-ipafont-mincho fonts-wqy-microhei fonts-wqy-zenhei fonts-indic
The following NEW packages will be installed:
  ca-certificates-java fontconfig-config fonts-dejavu-core java-common libavahi-client3
  libavahi-common-data libavahi-common3 libcups2 libfontconfig1 libjpeg-turbo8 libjpeg8 liblcms2-2
  libnspr4 libnss3 libpcsclite1 libxi6 libxrender1 libxtst6 openjdk-8-jre-headless x11-common
0 upgraded, 20 newly installed, 0 to remove and 16 not upgraded.
Need to get 30.7 MB of archives.
After this operation, 111 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```

TECHWORLD WITH NANA

Run packaged Java Application - 1

on prepared DigitalOcean server

In **real world**, applications will run on a remote server!

Server prepared to run java application

Summary of Steps:

1. **Build jar file**
2. **Copy to remote server**
3. **Run the application**

Run packaged Java Application - 2

on prepared DigitalOcean server

1. Build Jar File

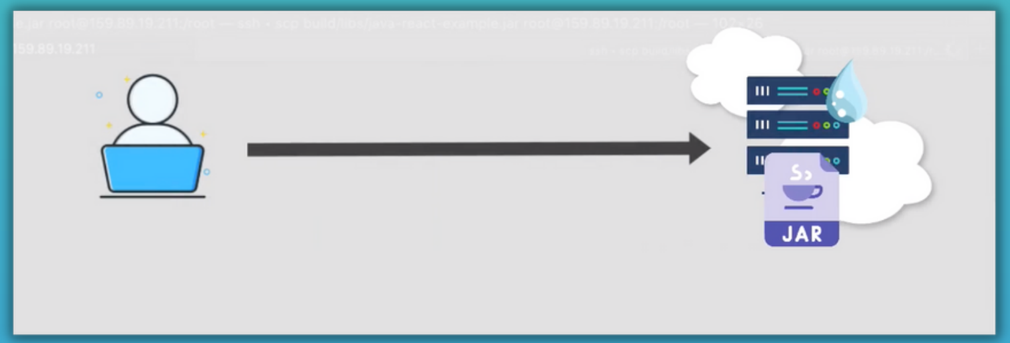
```
nana@macbook /Users/nana/java-
% gradle build

BUILD SUCCESSFUL in 8s
7 actionable tasks: 7 executed
```

2. Copy to Remote Server

```
nana@macbook /Users/nana/java-react-example [master]
% scp build/libs/java-react-example.jar root@164.90.218.238:/root
java-react-example.jar
```

command — .jar file source locally — user — public IP — destination

remote server information

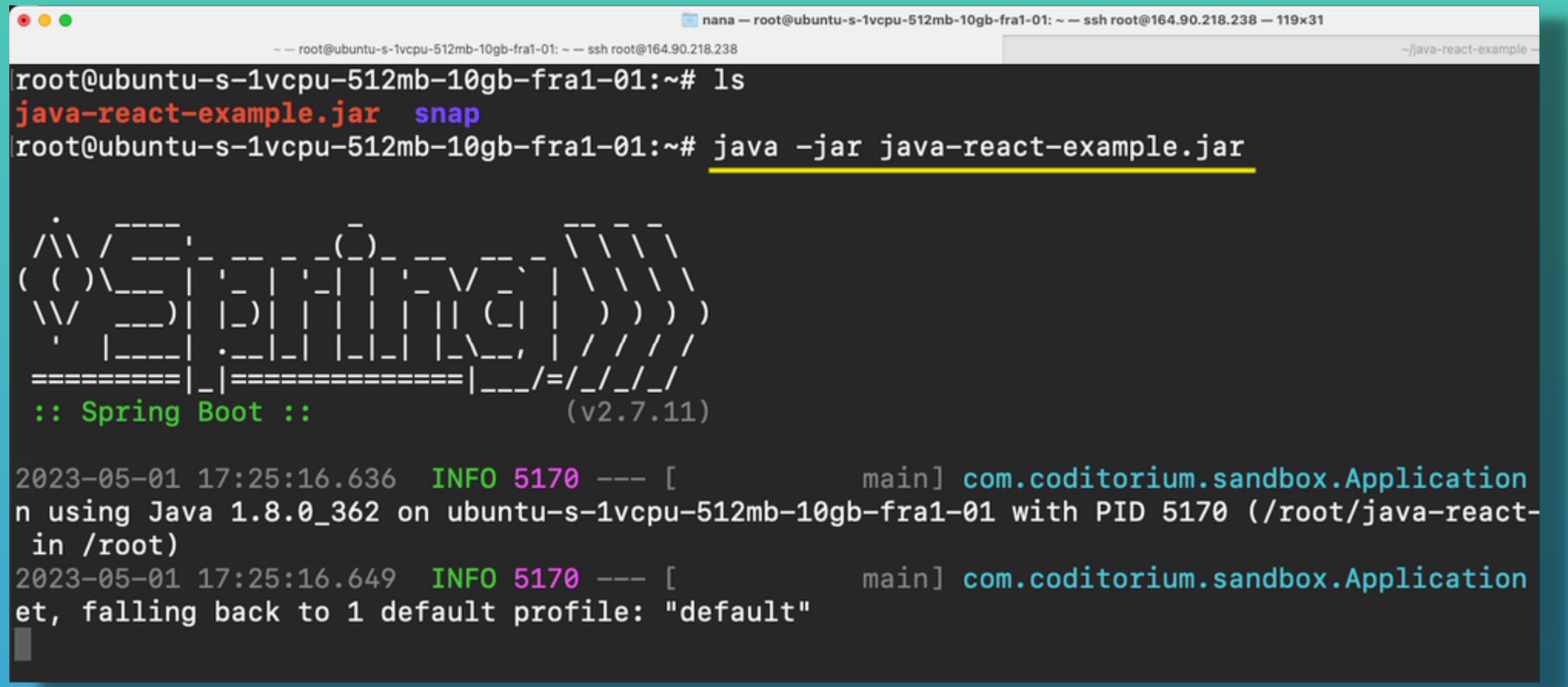


TECHWORLD WITH NANA

Run packaged Java Application - 3

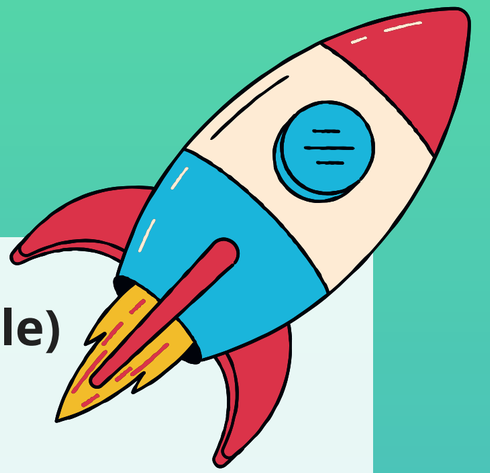
on prepared DigitalOcean server

3. Run the application



TECHWORLD WITH NANA

Security Best Practices



- **Create separate Linux user on remote server (as you learned in Linux module)**
  - Every cloud platform's configuration for their remote servers is different
  - On a droplet, by default you work with the "root" user
  - Create a new "admin" user
  - Using the "admin" user, create own users for each application you run (e.g. nexus, jenkins, my-app)

- Give it only the permission it needs to run that application

- **Don't work with the root user!**