DEVOPS BOOTCAMP

by TWN

Artifact Repository Manager with Nexus

Key Takeaways

TECHWORLD WITH NANA

Introduction to Artifact Repository Manager

TECHWORLD WITH NANA

What is an Artifact Repository?

- **Storage of build artifacts** produced by continuous integration and makes them available for automated deployment to different deployment environments
- Provides a **central location**

- Artifacts are applications built into a **single file**
- There are different artifact **formats:**

JAR, WAR, ZIP, TAR,etc.

- Artifact repository needs to **support this specific format**
- Repository for each file/artifact type

TECHWORLD WITH NANA

What is an Artifact Repository Manager?

- Can **store many different artifact types**
- Which is great, because as a company you often produce different types of artifacts

Instead of **having different repositories for each** artifact type

Just 1 repository for managing all your different artifact types

TECHWORLD WITH NANA

Public vs Private Artifact Repository Managers

Public Repository Managers

- There are public repository managers
- For example libraries/frameworks you use as a dependency
- You can make own project publically available there:

Public

Nexus - one of most popular repository manager

- It's a private central store
- You can **upload** and **store** different built artifacts
- **Retrieve** (download) artifacts later
- For internal usage in your company

Nexus

Private

TECHWORLD WITH NANA

Nexus - Artifact Repository Manager

- Has open source and commercial version
- Host own repositories
- Proxy repositories (intermediary to another repository):

Fetch everything from 1 place (Nexus): Company internal and public artifacts

- You can have multiple repositories for different formats or different teams in your company
- **Supported formats** in Nexus:

TECHWORLD WITH NANA

Features of Repository Manager

- Integration with LDAP
- Flexible and powerful REST API for integration with other tools

Important, because you need to integrate it into your CI/CD Pipeline

- Backup and Restore
- Multi-Format Support (different file types - zip, tar, docker etc)
- Metadata Tagging (labelling and tagging artifacts)
- Cleanup Policies (automatically delete files that match condition)
- Search functionality (across projects, artifact repos etc.)
- User token support for system user authentication

TECHWORLD WITH NANA

Install and Run Nexus on Cloud Server

TECHWORLD WITH NANA

Install and Run Nexus on Cloud Server - 1

Summary of Steps - 1st Part:

1. **Create Ubuntu Server (Droplet)** - min 4GB RAM & 2 CPUs
2. **Open SSH port 22**
3. **Install Java 8**
4. **Download and Install Nexus**
5. **Create "nexus" user and group** - Best Practice: Run applications with own user

TECHWORLD WITH NANA

Install and Run Nexus on Cloud Server - 2

With Nexus installation you get:

```
Bootcamp — root@ubuntu-s-1vcpu-1gb-fra1-01: /opt — ssh root@46.101.157.191 — 102×26
~/Demo-projects/Bootcamp — root@ubuntu-s-1vcpu-1gb-fra1-01: /opt — ssh root@46.101.157.191        ..react-example
root@ubuntu-s-1vcpu-1gb-fra1-01:/opt# ls
latest-unix.tar.gz  nexus-3.28.1-01  sonatype-work
root@ubuntu-s-1vcpu-1gb-fra1-01:/opt# ls nexus-3.28.1-01/
NOTICE.txt  OSS-LICENSE.txt  PRO-LICENSE.txt  bin  deploy  etc  lib  public  system
root@ubuntu-s-1vcpu-1gb-fra1-01:/opt# ls sonatype-work/nexus3/
clean_cache  log  orient  tmp
root@ubuntu-s-1vcpu-1gb-fra1-01:/opt#
```

Sonatype-work:
contains own config for Nexus and data

- subdirectories depending on your Nexus configuration
- IP address that accessed Nexus
- Logs of Nexus App
- Your uploaded files and metadata
- You can use this folder for backup

Nexus folder:
contains runtime and application of Nexus

TECHWORLD WITH NANA

Install and Run Nexus on Cloud Server - 3

Summary of Steps - 2nd Part:

1. Make Nexus own the folders
2. Start with nexus user
3. Open port 8081 with firewall
4. Access from browser

Starting Nexus...

- Services should not run with root user permissions
- So best practice: create own user for service (like in this case nexus)
- Should have only the permission for that specific service

TECHWORLD WITH NANA

Install and Run Nexus on Cloud Server - 4

Run Nexus...

1. Make Nexus own the folders

2. Start with nexus user

```
root@ubuntu-s-1vcpu-1gb-fra1-01:/opt# ls -l
total 162600
-rw-r--r-- 1 root root 166494026 Oct 19 14:35 latest-unix.tar.gz
drwxr-xr-x 9 root root      4096 Oct 25 09:06 nexus-3.28.1-01
drwxr-xr-x 3 root root      4096 Oct 25 09:06 sonatype-work
root@ubuntu-s-1vcpu-1gb-fra1-01:/opt# chown -R nexus:nexus nexus-3.28.1-01
root@ubuntu-s-1vcpu-1gb-fra1-01:/opt# chown -R nexus:nexus sonatype-work
root@ubuntu-s-1vcpu-1gb-fra1-01:/opt# ls -l
total 162600
-rw-r--r-- 1 root  root  166494026 Oct 19 14:35 latest-unix.tar.gz
drwxr-xr-x 9 nexus nexus      4096 Oct 25 09:06 nexus-3.28.1-01
drwxr-xr-x 3 nexus nexus      4096 Oct 25 09:06 sonatype-work
root@ubuntu-s-1vcpu-1gb-fra1-01:/opt#
```

```
nexus@ubuntu-s-4vcpu-8gb-fra1-01:~$ /opt/nexus-3.28.1-01/bin/nexus start
Starting nexus
nexus@ubuntu-s-4vcpu-8gb-fra1-01:~$ ps aux | grep nexus
root       14850  0.0  0.0  10104  3768 pts/0    S    12:38   0:00 su - nexus
nexus      14851  0.0  0.0  10056  5032 pts/0    S    12:38   0:00 -bash
nexus      15531  281 11.5 6353596 943024 pts/0  Sl   12:39   0:31 /usr/lib/jvm/java-8-openjdk-amd64/j
re/bin/java -server -Dinstall4j.jvmDir=/usr/lib/jvm/java-8-openjdk-amd64/jre -Dexe4j.moduleName=/opt/n
exus-3.28.1-01/bin/nexus -XX:+UnlockDiagnosticVMOptions -Dinstall4j.launcherId=245 -Dinstall4j.swt=fal
se -Di4jv=0 -Di4jv=0 -Di4jv=0 -Di4jv=0 -Di4jv=0 -Xms2703m -Xmx2703m -XX:MaxDirectMemorySize=2703m -XX:
+UnlockDiagnosticVMOptions -XX:+LogVMOutput -XX:LogFile=../sonatype-work/nexus3/log/jvm.log -XX:-OmitS
tackTraceInFastThrow -Djava.net.preferIPv4Stack=true -Dkaraf.home=. -Dkaraf.base=. -Dkaraf.etc=etc/kar
af -Djava.util.logging.config.file=etc/karaf/java.util.logging.properties -Dkaraf.data=../sonatype-wor
k/nexus3 -Dkaraf.log=../sonatype-work/nexus3/log -Djava.io.tmpdir=../sonatype-work/nexus3/tmp -Dkaraf.
startLocalConsole=false -Djava.endorsed.dirs=lib/endorsed -Di4j.vpt=true -classpath /opt/nexus-3.28.1-
01/.install4j/i4jruntime.jar:/opt/nexus-3.28.1-01/lib/boot/nexus-main.jar:/opt/nexus-3.28.1-01/lib/boo
t/activation-1.1.1.jar:/opt/nexus-3.28.1-01/lib/boot/jakarta.xml.bind-api-2.3.3.jar:/opt/nexus-3.28.1-
01/lib/boot/jaxb-runtime-2.3.3.jar:/opt/nexus-3.28.1-01/lib/boot/txw2-2.3.3.jar:/opt/nexus-3.28.1-01/l
ib/boot/istack-commons-runtime-3.0.10.jar:/opt/nexus-3.28.1-01/lib/boot/org.apache.karaf.main-4.2.9.ja
r:/opt/nexus-3.28.1-01/lib/boot/osgi.core-6.0.0.jar:/opt/nexus-3.28.1-01/lib/boot/org.apache.karaf.spe
cs.activator-4.2.9.jar:/opt/nexus-3.28.1-01/lib/boot/org.apache.karaf.diagnostic.boot-4.2.9.jar:/opt/n
exus-3.28.1-01/lib/boot/org.apache.karaf.jaas.boot-4.2.9.jar com.install4j.runtime.launcher.UnixLaunch
er start 9d17dc87 0 0 org.sonatype.nexus.karaf.NexusMain
```

TECHWORLD WITH NANA

Install and Run Nexus on Cloud Server - 5

Run Nexus...

3. Open port 8081 with Firewall

4. Access from browser

TECHWORLD WITH NANA

Overview of Nexus

TECHWORLD WITH NANA

Login & Default Repositories

Login

- With **default "admin" user** that automatically gets created to administer Nexus

Default Repositories

- You get default repositories with different TYPES when you install Nexus

TECHWORLD WITH NANA

Repository Types & Formats

Repository Types

- **Proxy**: intermediary to another repository
- **Group**: containing a group of repositories
- **Hosted**: for hosting company-internal artifacts

Each technology (python, java, docker) has its artifact type (jar, image), which can be saved in its own artifact repository format

Repository Formats

- **maven2**: for java jar artifacts
- **nuget**: for .NET packages
- **docker**: for Docker images

TECHWORLD WITH NANA

Publish artifact to repository

You can push or "PUBLISH" artifacts to Nexus with different tools

- Using built-in commands of build tools, like Maven or Gradle
- For that, you need to **configure** each build tool **with Nexus credentials and address**

X X X X X

TECHWORLD WITH NANA

Nexus REST API

You can also **interact with Nexus using its REST API** to do multiple things:

- query for different information: list of repositories | list of components | ...
- download an artifact
- upload an artifact

How to interact?

- Using a tool like curl or wget to execute http request
- You need to provide user and credential of a Nexus user

```
[\W]$ curl -u user:pwd -X GET 'http://167.99.248.163:8081/service/rest/v1/repositories'
```

```
[\W]$ curl -u nana:Bibilo123 -X GET 'http://167.99.248.163:8081/service/rest/v1/components?repository=
maven-snapshots'
```

To use Nexus API or any API:
You should **always reference the documentation**. Not learn by heart, because 1) API changes 2) there are a lot of endpoints, so you can't learn that much

TECHWORLD WITH NANA

Component vs Asset

Components

- **abstract**
- what we are uploading
- term "component" refers to any type or format

Assets

- **actual physical packages/files**
- 1 component = 1 or more assets

TECHWORLD WITH NANA

Cleanup Policies & Scheduled Tasks

Cleanup Policies

- Clean up old artifacts to make space. You can define logic like:

→ **Delete all artifacts older than 30 days**

→ **Delete all artifacts, which haven't been used/downloaded for more than 10 days**

- You can create own cleanup policy for each repository

Scheduled Tasks

- Nexus allows you to schedule tasks that will be applied to all or a specific repository on a configurable schedule

TECHWORLD WITH NANA