一些要安装的插件和配置

Plugin Manager

Updates   Available   **Installed**   Advanced

Name ↓	Enabled	
Timestamper 1.17 Adds timestamps to the Console Output Report an issue with this plugin	✓	⊗

Dashboard ‣ **Global Tool Configuration**

Ant installations

Add Ant

List of Ant installations on this system

Maven

Maven installations

Add Maven

⠿ Maven
Name

m2

MAVEN_HOME

/usr/local/apache-maven-3.8.6

☐ Install automatically ?

Plugin Manager

Updates   **Available**   Installed   Advanced

Install	Name ↓
	AnsiColor 1.0.2
☑	Miscellaneous
	Adds ANSI coloring to the Console Output

==其实github配置可以不用，因为我后面没用github了，太慢了，换成gitee了，或者gitlab：==

🏰 全局凭据 (unrestricted)

Credentials that should be available irrespective of domain specification to requirements matching.

ID	Name	Kind	Description	
🧑 yk-github20221111	frankyangkun/****** (我的github)	Username with password	我的github	🔧

Icon:　S　M　L

用户名密码github不支持了，所以下面的配置是不对的，不需要下图这个凭据了：

改成直接使用token：

Global Pipeline Libraries

Sharable libraries available to any Pipeline jobs running on this system. These libraries will be trusted, meaning they run without "sandbox" restrictions and may use @Grab.

Library
Name   ?

jenkinslin

Default version   ?

main

Cannot validate default version until after saving and reconfiguring.

☐ Load implicitly   ?

☑ Allow default version to be overridden   ?

☑ Include @Library changes in job recent changes   ?

☐ Cache fetched versions on controller for quick retrieval   ?
Retrieval method

Modern SCM ▾

Loads a library from an SCM plugin using newer interfaces optimized for this purpose. The recommended option when available.
Source Code Management

GitHub ▾

Credentials   ?

- 无 - ▾   ⊕Add ▾

⚠ Credentials are recommended

◉ Repository HTTPS URL
Repository HTTPS URL   ?

https://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@github.com/frankyangkun/jenkinslib.git

Credentials ok. Connected to https://github.com/frankyangkun/jenkinslib.   [Validate]

○ Repository Scan - Deprecated Visualization

| General | Build Triggers | Advanced Project Options | **Pipeline** |

Branch Specifier (blank for 'any')   ?

*/main

[Add Branch]

Repository browser   ?

(Auto)

Additional Behaviours

Clean before checkout
   ?
   ☑ Delete untracked nested repositories   ?

[Add ▾]

Script Path   ?

ci.jenkinsfile

☑ **Lightweight checkout**   ?

遇到的问题

重要问题：（已解决）无法访问/bitnami/jenkins/home（权限问题）


```
Console Output

Started by user user
Loading library jenkinslib@main
Examining frankyangkun/jenkinslib
Attempting to resolve main as a branch
Resolved main as branch main at revision 2e71a6c1a735126770ff706d3051e187dbf67bec
java.nio.file.AccessDeniedException: /bitnami/jenkins/home/workspace/MicroserviceDemo@libs
        at java.base/sun.nio.fs.UnixException.translateToIOException(UnixException.java:90)
        at java.base/sun.nio.fs.UnixException.rethrowAsIOException(UnixException.java:111)
        at java.base/sun.nio.fs.UnixException.rethrowAsIOException(UnixException.java:116)
        at java.base/sun.nio.fs.UnixFileSystemProvider.createDirectory(UnixFileSystemProvider.java:389)
        at java.base/java.nio.file.Files.createDirectory(Files.java:690)
        at java.base/java.nio.file.Files.createAndCheckIsDirectory(Files.java:797)
        at java.base/java.nio.file.Files.createDirectories(Files.java:783)
        at hudson.FilePath.mkdirs(FilePath.java:3609)
        at hudson.FilePath.access$1100(FilePath.java:212)
        at hudson.FilePath$Write.invoke(FilePath.java:2445)
        at hudson.FilePath$Write.invoke(FilePath.java:2433)
        at hudson.FilePath.act(FilePath.java:1200)
        at hudson.FilePath.act(FilePath.java:1183)
        at hudson.FilePath.write(FilePath.java:2430)
        at org.jenkinsci.plugins.workflow.libs.SCMSourceRetriever.doRetrieve(SCMSourceRetriever.java:199)
        at org.jenkinsci.plugins.workflow.libs.SCMSourceRetriever.retrieve(SCMSourceRetriever.java:137)
        at org.jenkinsci.plugins.workflow.libs.LibraryAdder.retrieve(LibraryAdder.java:204)
        at org.jenkinsci.plugins.workflow.libs.LibraryAdder.add(LibraryAdder.java:142)
        at org.jenkinsci.plugins.workflow.libs.LibraryDecorator$1.call(LibraryDecorator.java:125)
        at org.codehaus.groovy.control.CompilationUnit.applyToPrimaryClassNodes(CompilationUnit.java:1087)
        at org.codehaus.groovy.control.CompilationUnit.doPhaseOperation(CompilationUnit.java:624)
```

问题：（已解决）jenkins job名不能包含大写字母

顺便说下命名问题：

主要原因在这儿：home虽然是1001普通用户的（我的登录用户user就是普通用户）

```
/bitnami/jenkins
$ ls -al
total 12
drwxrwxrwx  3 root root   18 Nov  9 13:11 .
drwxr-xr-x  3 root root   21 Jun 10 15:09 ..
drwxr-xr-x 20 1001 root 8192 Nov 11 07:24 home
$
```

但是里面的workspace不属于1001用户：不知道什么时候变的

```
drwx------   2 1001 root 4096 Nov 11 03:32 secrets
drwxr-xr-x   2 1001 root  182 Nov 11 07:21 updates
drwxr-xr-x   2 1001 root   24 Nov  9 13:12 userContent
drwxr-xr-x   3 1001 root   55 Nov 11 07:21 users
drwxr-xr-x  11 1001 root 4096 Nov  9 13:11 war
drwxr-xr-x   3 root root   37 Nov 10 07:36 workspace
# pwd
/bitnami/jenkins/home
#
```

而且workspace里所有者全是root的：

```
# ls -al workspace
total 12
drwxr-xr-x  3 root root   37 Nov 10 07:36 .
drwxr-xr-x 20 1001 root 8192 Nov 11 07:24 ..
drwxr-xr-x  3 root root   17 Nov 10 07:36 demo-maven-microservice
#
```

我目前jenkins的user用户是1001普通用户，应该是无法操作root的

用**docker exec -it -u 0 32c7b3e8ce4c /bin/sh**以root用户进入jenkins容器，
改成1001：都变成1001的了：

```
# chown -R 1001:root workspace
# ls -al
```

```
# ls -al
total 12
drwxr-xr-x  3 1001 root   37 Nov 10 07:36 .
drwxr-xr-x 20 1001 root 8192 Nov 11 07:24 ..
drwxr-xr-x  3 1001 root   17 Nov 10 07:36 demo-maven-microservice
#
```

但是如果jenkins重启了pod，会不会新的pod又恢复成root了，我之前从来没遇
到过这个问题

重启了下，重新进入容器，还好没变：也可能是重启后新的容器本身就是这个权
限

```
drwxr-xr-x  11 1001 root 4096 Nov  9 13:11 war
drwxr-xr-x   3 1001 root   37 Nov 10 07:36 workspace
# ls -al workspace
total 12
drwxr-xr-x  3 1001 root   37 Nov 10 07:36 .
drwxr-xr-x 20 1001 root 8192 Nov 11 08:04 ..
drwxr-xr-x  3 1001 root   17 Nov 10 07:36 demo-maven-microservice
#
```

终于成功了，确实是这个原因：

但我刚才的操作之后重启**jenkins**，是新的**pod**了才对，为什么配置还能保持？
应该是我的**jenkins**做了持久化，配置信息保存在了**nfs**，所以**chown**的操作也保存了。

ps：我估计不是这个原因，应该是我当前的**jenkins**不知道什么原因导致权限变了，重启**jenkins**后，是新部署的**jenkins**了，权限自然就恢复了。反正肯定都是权限造成的。

问题：（已解决）重新部署或重启jenkins（web上restart操作）后，jenkins容器内sonar-scanner和vim都没了

有个严重问题，我jenkins重启后，pod名称没变，但是里面的sonar-scanner都没了，vim也没了，不知道什么原因

应该是每次jenkins pod内的容器都变了，每次exec进入的都是新的，但是因为我的jenkins部署时，Dockerfile里包含了maven的内容，所以每次新部署的容器都包含了dockerfile里的内容，所以mvn是保留的，可是后来我单独进入容器安装的sonarscanner和vim肯定就没法保存了。

所以 这么看来，我的**sonarscanner**安装，**vim**安装也最好都集成到**jenkins-**

maven的dockerfile里

不然可能每次重启jenkins，就都无法保存。

问题：（已解决）为什么sonarscanner和vim不能自动保存到nfs（只能和maven一样，集成到dockerfile里）

ps：因为nfs保存的都是jenkins本身相关的数据，比如上面的job，运行数据，配置等，sonarscanner和vim都是我另外安装到jenkins所在的容器本身的os上的，跟jenkins没关系，重启后实际上os都是重装了，所以肯定是全部清理了，所以要想重启后保存，只能和maven一样，集成到dockerfile里

补充：jenkinsfile里添加runOpts参数定义

补充：其实后面视频里也遇到过这个问题，只需要在jenkinsfile添加上这个变量即可：然后就可以手动执行了，不一定非要commit来触发

我直接加了个：


```groovy
#!groovy

@Library('jenkinslib') _

//func from share library
def build = new org.devops.build()
//def deploy = new org.devops.deploy()
def tools = new org.devops.tools()
def gitlab = new org.devops.gitlab()

//不一定非要commit操作才能触发，手动执行也可触发
def runOpts

//env
String buildType = "${env.buildType}"
```

问题：（已解决）maven没找到本地仓库.m2目录（编译完成后就会生成）

还有个问题：我的maven没找到本地仓库.m2目录，但是看之前的笔记，虽然有个这个：

看了下配置文件，确实没定义：只有注释


```
<settings xmlns="http://maven.apache.org/SETTINGS/1.2.0"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:schemaLocation="http://maven.apache.org/SETTINGS
  <!-- localRepository
   | The path to the local repository maven will use to store
   |
   | Default: ${user.home}/.m2/repository
  <localRepository>/path/to/local/repo</localRepository>
  -->

  <!-- interactiveMode
   | This will determine whether maven prompts you when it nee
   | maven will use a sensible default value, perhaps based on
   | the parameter in question.
   |
   | Default: true
  <interactiveMode>true</interactiveMode>
  -->

  <!-- offline
```
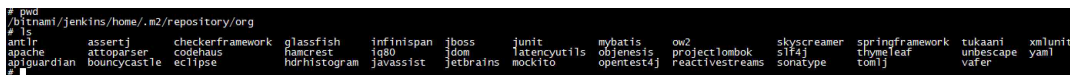
这是网上查到的：应该加上这一行

`<localRepository>${user.home}/.m2/repository</localRepository>`

但最终其实是没加的，之前是找了的，不知道为什么。

补充：后来我编译成功后，找到了这个目录：但是没找到我的项目

但是运行日志里又有：

```
Downloaded from central: https://repo.maven.apache.org/maven2/com/google/j2objc/j2objc-annotations/1.3/j2objc-annotations-1.3.jar (8.8 kB at 1.2 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/com/google/guava/guava/28.2-android/guava-28.2-android.jar (2.6 MB at 341 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/commons/commons-lang3/3.7/commons-lang3-3.7.jar (500 kB at 65 kB/s)
[INFO] Replacing main artifact with repackaged archive
[INFO]
[INFO] --- maven-install-plugin:2.5.2:install (default-install) @ 046-springboot-dubbo-ssm-provider ---
[INFO] Installing /bitnami/jenkins/home/workspace/microservice-demo-20221111/046-springboot-dubbo-ssm-provider/target/046-springboot-dubbo-ssm-provider-v1.0.jar to
/bitnami/jenkins/home/.m2/repository/com/frank/springboot/046-springboot-dubbo-ssm-provider/v1.0/046-springboot-dubbo-ssm-provider-v1.0.jar
[INFO] Installing /bitnami/jenkins/home/workspace/microservice-demo-20221111/046-springboot-dubbo-ssm-provider/pom.xml to
/bitnami/jenkins/home/.m2/repository/com/frank/springboot/046-springboot-dubbo-ssm-provider/v1.0/046-springboot-dubbo-ssm-provider-v1.0.pom
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  04:27 min
[INFO] Finished at: 2022-11-14T02:23:05Z
[INFO] ------------------------------------------------------------------------
[Pipeline] }
[Pipeline] // dir
[Pipeline] dir
Running in /bitnami/jenkins/home/workspace/microservice-demo-20221111/047-springboot-dubbo-ssm-consumer
[Pipeline] {
```

Removing intermediate container bb2194d69491
 ---> 533177b29ad9
ProgressMessage{id=null, status=null, stream=null, error=null, progress=null, progressDetail=null}
Successfully built 533177b29ad9
Successfully tagged frankyang1988/047-springboot-dubbo-ssm-consumer:v20220322
[INFO] Built frankyang1988/047-springboot-dubbo-ssm-consumer:v20220322
[INFO]
[INFO] --- maven-install-plugin:2.5.2:install (default-install) @ 047-springboot-dubbo-ssm-consumer ---
[INFO] Installing /bitnami/jenkins/home/workspace/microservice-demo-20221111/047-springboot-dubbo-ssm-consumer/target/047-springboot-dubbo-ssm-consumer-1.0-SNAPSHOT.jar to
/bitnami/jenkins/home/.m2/repository/com/frank/springboot/047-springboot-dubbo-ssm-consumer/1.0-SNAPSHOT/047-springboot-dubbo-ssm-consumer-1.0-SNAPSHOT.jar
[INFO] Installing /bitnami/jenkins/home/workspace/microservice-demo-20221111/047-springboot-dubbo-ssm-consumer/pom.xml to
/bitnami/jenkins/home/.m2/repository/com/frank/springboot/047-springboot-dubbo-ssm-consumer/1.0-SNAPSHOT/047-springboot-dubbo-ssm-consumer-1.0-SNAPSHOT.pom
[INFO] --
[INFO] BUILD SUCCESS
[INFO] --
[INFO] Total time: 02:37 min
[INFO] Finished at: 2022-11-14T02:25:47Z
[INFO] --
[Pipeline])
[Pipeline] // dir
[Pipeline])

其实是有的，目录进错了：

重要问题：（已解决）始终报Incorrect username or password（gitee只能用账号密码，不能用私钥公钥方法）

在《Git、GitLab、GitHub、Gitee（码云）的区别》中确认过，我的gitee密码没问题，但始终报错：

```
getProjectName按/分割结果gitlab: [https:, , gitee.com, frankyang1988, 044-springboot-dubbo-ssm-parent.git]
[Pipeline] echo
项目名称: 044-springboot-dubbo-ssm-parent.git
[Pipeline] checkout
The recommended git tool is: NONE
Warning: CredentialId "fd1f2048-303b-42a2-8328-9e57063de47f" could not be found.
 > git rev-parse --resolve-git-dir /bitnami/jenkins/home/workspace/microservice-demo-20221111/044-springboot-dubbo-ssm-parent.git/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://gitee.com/frankyang1988/044-springboot-dubbo-ssm-parent.git # timeout=10
Fetching upstream changes from https://gitee.com/frankyang1988/044-springboot-dubbo-ssm-parent.git
 > git --version # timeout=10
 > git --version # 'git version 2.36.1'
 > git fetch --tags --force --progress -- https://gitee.com/frankyang1988/044-springboot-dubbo-ssm-parent.git +refs/heads/*:refs/remotes/origin/* # timeout=10
ERROR: Error fetching remote repo 'origin'
hudson.plugins.git.GitException: Failed to fetch from https://gitee.com/frankyang1988/044-springboot-dubbo-ssm-parent.git
    at hudson.plugins.git.GitSCM.fetchFrom(GitSCM.java:1003)
    at hudson.plugins.git.GitSCM.retrieveChanges(GitSCM.java:1244)
    at hudson.plugins.git.GitSCM.checkout(GitSCM.java:1308)
    at org.jenkinsci.plugins.workflow.steps.scm.SCMStep.checkout(SCMStep.java:129)
    at org.jenkinsci.plugins.workflow.steps.scm.SCMStep$StepExecutionImpl.run(SCMStep.java:97)
    at org.jenkinsci.plugins.workflow.steps.scm.SCMStep$StepExecutionImpl.run(SCMStep.java:84)
    at org.jenkinsci.plugins.workflow.steps.SynchronousNonBlockingStepExecution.lambda$start$0(SynchronousNonBlockingStepExecution.java:47)
    at java.base/java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:515)
    at java.base/java.util.concurrent.FutureTask.run(FutureTask.java:264)
    at java.base/java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1128)
    at java.base/java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:628)
    at java.base/java.lang.Thread.run(Thread.java:829)
Caused by: hudson.plugins.git.GitException: Command "git fetch --tags --force --progress -- https://gitee.com/frankyang1988/044-springboot-dubbo-ssm-parent.git
+refs/heads/*:refs/remotes/origin/*" returned status code 128:
stdout:
stderr: remote: ←[31m[session-8c492fa3] Username for 'https: Incorrect username or password (access token)←[0m
fatal: Authentication failed for 'https://gitee.com/frankyang1988/044-springboot-dubbo-ssm-parent.git/'

    at org.jenkinsci.plugins.gitclient.CliGitAPIImpl.launchCommandIn(CliGitAPIImpl.java:2671)
    at org.jenkinsci.plugins.gitclient.CliGitAPIImpl.launchCommandWithCredentials(CliGitAPIImpl.java:2096)
    at org.jenkinsci.plugins.gitclient.CliGitAPIImpl.access$500(CliGitAPIImpl.java:84)
    at org.jenkinsci.plugins.gitclient.CliGitAPIImpl$1.execute(CliGitAPIImpl.java:618)
    at hudson.plugins.git.GitSCM.fetchFrom(GitSCM.java:1001)
    ... 11 more
```

好像是需要在 gitlab机器上生成公钥私钥，然后把公钥拷贝到jenkins上作为凭据
之前的笔记：

继续：按帖子里的说法，gitlab上的ssh key是配置jenkins所在机器的公钥，我之前好像是配置的gitlab上的公钥
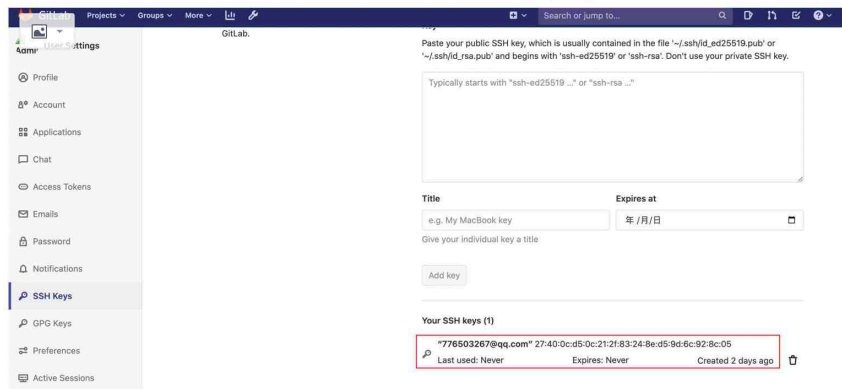
确实应该是配置jenkins所在机器的，因为是要建立gitlab和jenkins的连接

我当时为什么要配成gitlab的？当时以为导入github项目需要，其实导入github只需要github的personal access token，不需要配置gitlab所在机器的公钥，
前面我理解错了，已经补充笔记了

jenkins所在机器node3的公钥，配到gitee：

jenkins所在机器node3的 <mark>私钥</mark>，配到jenkins：

可是还是不行：反正之前的最开始gitee确实没这么配置过，只是gitlab这样配过

General   Build Triggers   Advanced Project Options   **Pipeline**

Definition

Pipeline script from SCM

SCM ?

Git

Repositories ?

Repository URL ?

https://gitee.com/frankyang1988/jenkinslib.git

Failed to connect to repository : Command "git ls-remote -h -- https://gitee.com/frankyang1988/jenkinslib.git HEAD" returned status code 128:
stdout:
stderr: remote: ←[31m[session-cd43968f] Username for 'https: Incorrect username or password (access token)←[0m
fatal: Authentication failed for 'https://gitee.com/frankyang1988/jenkinslib.git/'

Credentials ?

node3-jenkins-privatekey (node3-jenkins-privatekey)   Add ▾

==删掉gitee上的公钥==

补充：关键在于jenkinsfile中checkout时的**==CredentialId==参数**

（**gitlab用的是私钥，gitee只能用账号密码**）

应该是这里的问题：==CredentialId==


```
[Pipeline] echo
>>>>>>>>>>>获取父工程代码<<<<<<<<<<<
[Pipeline] }

[Pipeline] // ansiColor
[Pipeline] echo
getProjectName按/分割结果gitlab: [https:, , gitee.com, frankyang1988, 044-springboot-dubbo-ssm-parent.git]
[Pipeline] echo
项目名称：044-springboot-dubbo-ssm-parent.git
[Pipeline] checkout
The recommended git tool is: NONE
Warning: CredentialId "fd1f2048-303b-42a2-8328-9e57063de47f" could not be found.
 > git rev-parse --resolve-git-dir /bitnami/jenkins/home/workspace/microservice-demo-20221111/044-springboot-dubbo-ssm-parent.git/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://gitee.com/frankyang1988/044-springboot-dubbo-ssm-parent.git # timeout=10
Fetching upstream changes from https://gitee.com/frankyang1988/044-springboot-dubbo-ssm-parent.git
 > git --version # timeout=10
 > git --version # 'git version 2.36.1'
 > git fetch --tags --force --progress -- https://gitee.com/frankyang1988/044-springboot-dubbo-ssm-parent.git +refs/heads/*:refs/remotes/origin/*
ERROR: Error fetching remote repo 'origin'
hudson.plugins.git.GitException: Failed to fetch from https://gitee.com/frankyang1988/044-springboot-dubbo-ssm-parent.git
	at hudson.plugins.git.GitSCM.fetchFrom(GitSCM.java:1003)
	at hudson.plugins.git.GitSCM.retrieveChanges(GitSCM.java:1244)
	at hudson.plugins.git.GitSCM.checkout(GitSCM.java:1308)
	at org.jenkinsci.plugins.workflow.steps.scm.SCMStep.checkout(SCMStep.java:129)
	at org.jenkinsci.plugins.workflow.steps.scm.SCMStep$StepExecutionImpl.run(SCMStep.java:97)
	at org.jenkinsci.plugins.workflow.steps.scm.SCMStep$StepExecutionImpl.run(SCMStep.java:84)
	at org.jenkinsci.plugins.workflow.steps.SynchronousNonBlockingStepExecution.lambda$start$0(SynchronousNonBlockingStepExecution.java:47)
	at java.base/java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:515)
```

```
'RelativeTargetDirectory', relativeTargetDir: "${parentName}"]], userRemoteConfigs: [[credentialsId: 'fd1f2048-303b-42a2-8328-9e57063de47f', url: '${srcUrl_
```

```
RemoteConfigs: [[credentialsId: 'fd1f2048-303b-42a2-8328-9e57063de47f', url: '${srcUrl_interface}']]])
'RelativeTargetDirectory', relativeTargetDir: "${interfaceName}"]], userRemoteConfigs: [[credentialsId: 'fd1f2048-303b-42a2-8328-9e57063de47f', url: '${srcU
```
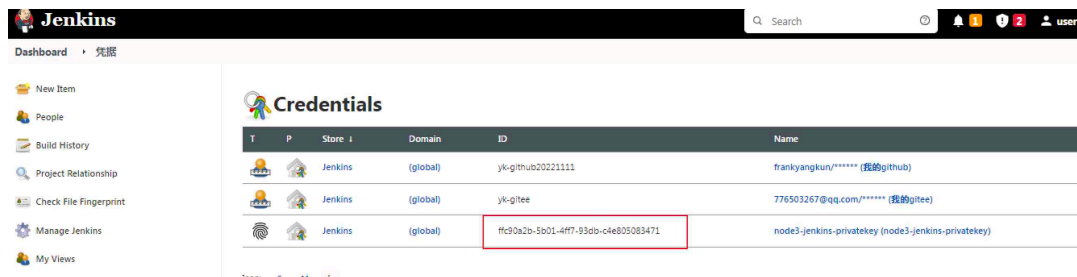
.t后缀)

```
RemoteConfigs: [[credentialsId: 'fd1f2048-303b-42a2-8328-9e57063de47f', url: '${srcUrl_provider}']]])
'RelativeTargetDirectory', relativeTargetDir: "${providerName}"]], userRemoteConfigs: [[credentialsId: 'fd1f2048-303b-42a2-8328-9e57063de47f', url: '${srcU
```

缀)

```
RemoteConfigs: [[credentialsId: 'fd1f2048-303b-42a2-8328-9e57063de47f', url: '${srcUrl_consumer}']]])
```

根据之前的笔记，这个id是jenkins配置的私钥：因为我的jenkins重新部署了，凭据全没了，所以才会报上面的错误，现在是这个

修改jenkinsfile：

```
Dir: "${parentName}"]], userRemoteConfigs: [[credentialsId: 'ffc90a2b-5b01-4ff7-93db-c4e805083471', url: '${srcUrl_parent}']]
```

```
-303b-42a2-8328-9e57063de47f', url: '${srcUrl_interface}']]])
Dir: "${interfaceName}"]], userRemoteConfigs: [[credentialsId: 'ffc90a2b-5b01-4ff7-93db-c4e805083471', url: '${srcUrl_interfac
```

```
-303b-42a2-8328-9e57063de47f', url: '${srcUrl_provider}']]])
Dir: "${providerName}"]], userRemoteConfigs: [[credentialsId: 'ffc90a2b-5b01-4ff7-93db-c4e805083471', url: '${srcUrl_provider}
```

```
-303b-42a2-8328-9e57063de47f', url: '${srcUrl_consumer}']]])
Dir: "${consumerName}"]], userRemoteConfigs: [[credentialsId: 'ffc90a2b-5b01-4ff7-93db-c4e805083471', url: '${srcUrl_consumer}
```

gitee上的公钥还是要加上：

但是pipeline这里不能用它，这个私钥是给jenkinsfile里checkout代码时用的

还是要用gitee的账号密码：

为什么还是不对：

换成另一个凭据：就是gitee本身的账号密码

Credentials

T	P	Store ↓	Domain	ID	Name
👤	🏠	Jenkins	(global)	yk-github20221111	frankyangkun/****** (我的github)
👤	🏠	Jenkins	(global)	yk-gitee	776503267@qq.com/****** (我的gitee)
👆	🏠	Jenkins	(global)	ffc90a2b-5b01-4ff7-93db-c4e805083471	node3-jenkins-privatekey (node3-jenkins-privatekey)

```
r: "${parentName}"]], userRemoteConfigs: [[credentialsId: 'yk-gitee', url: '${srcUrl_parent}']]])
```

```
03b-42a2-8328-9e57063de47f', url: '${srcUrl_interface}']]])
r: "${interfaceName}"]], userRemoteConfigs: [[credentialsId: 'yk-gitee', url: '${srcUrl_interface}']]
```
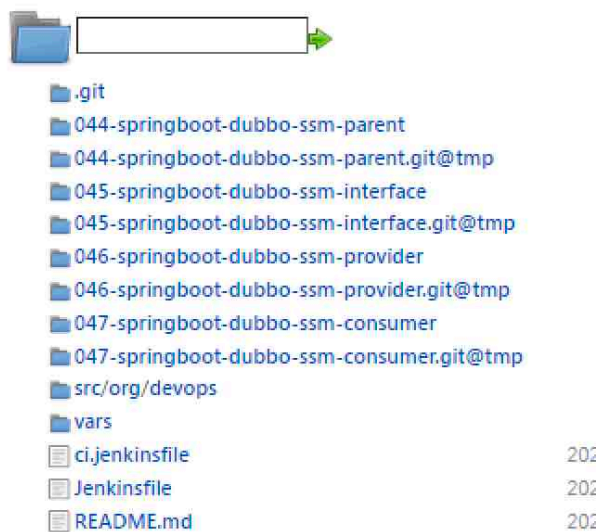
```
03b-42a2-8328-9e57063de47f', url: '${srcUrl_provider}']]])
r: "${providerName}"]], userRemoteConfigs: [[credentialsId: 'yk-gitee', url: '${srcUrl_provider}']]])
```

```
03b-42a2-8328-9e57063de47f', url: '${srcUrl_consumer}']]])
r: "${consumerName}"]], userRemoteConfigs: [[credentialsId: 'yk-gitee', url: '${srcUrl_consumer}']]])
```

终于可以了：

```
[Pipeline] echo
>>>>>>>>>>拉取父工程代码<<<<<<<<<<
[Pipeline] ]

[Pipeline] // ansiColor
[Pipeline] echo
getProjectName按/分割结果gitlab: [https:, , gitee.com, frankyang1988, 044-springboot-dubbo-ssm-parent.git]
[Pipeline] echo
项目名称: 044-springboot-dubbo-ssm-parent.git
[Pipeline] checkout
The recommended git tool is: NONE
using credential yk-gitee
 > git rev-parse --resolve-git-dir /bitnami/jenkins/home/workspace/microservice-demo-20221111/044-springboot-dubbo-ssm-parent.git/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://gitee.com/frankyang1988/044-springboot-dubbo-ssm-parent.git # timeout=10
Fetching upstream changes from https://gitee.com/frankyang1988/044-springboot-dubbo-ssm-parent.git
 > git --version # timeout=10
 > git --version # 'git version 2.36.1'
using GIT_ASKPASS to set credentials 我的gitee
 > git fetch --tags --force --progress -- https://gitee.com/frankyang1988/044-springboot-dubbo-ssm-parent.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git rev-parse origin/master^{commit} # timeout=10
Checking out Revision 81022ce2d0689d3b8276bed8d6f5714c78e37ec2 (origin/master)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f 81022ce2d0689d3b8276bed8d6f5714c78e37ec2 # timeout=10
Commit message: "父工程（重新整合springboot微服务demo. 引入rabbitmq实现）"
First time build. Skipping changelog.
[Pipeline] echo
getProjectName按/分割结果gitlab: [https:, , gitee.com, frankyang1988, 044-springboot-dubbo-ssm-parent.git]
[Pipeline] echo
项目名称: 044-springboot-dubbo-ssm-parent.git
[Pipeline] echo
projectNameList:[044-springboot-dubbo-ssm-parent, git]
[Pipeline] echo
项目名称去掉.git后: 044-springboot-dubbo-ssm-parent
[Pipeline] sh
```

代码都下载了:

Workspace

- .git
- 044-springboot-dubbo-ssm-parent
- 044-springboot-dubbo-ssm-parent.git@tmp
- 045-springboot-dubbo-ssm-interface
- 045-springboot-dubbo-ssm-interface.git@tmp
- 046-springboot-dubbo-ssm-provider
- 046-springboot-dubbo-ssm-provider.git@tmp
- 047-springboot-dubbo-ssm-consumer
- 047-springboot-dubbo-ssm-consumer.git@tmp
- src/org/devops
- vars
- ci.jenkinsfile   202
- Jenkinsfile   202
- README.md   202

==这里就要总结一下了，和之前不一样了，私钥的用法不适用于gitee，之前gitlab是用的私钥方案，要特别注意。==

补充：换成gitee后，jenkinsfile需要修改的地方

```
11    //不一定非要commit操作才能触发，手动执行也可触发
12    def runOpts
13    //换成gitee时，无法获取gitlab的webhook中的projectId，和commitSha，定义一个空的测试用
14    def projectId
15    def commitSha
16
```

```
193        }
194      success{
195        script{
196          println("success..")
197          //换成gitee时，无法获取gitlab的webhook中的projectId，注释掉
198          //gitlab.ChangeCommitStatus(projectId,commitSha,"success")
199        }
200      }
201      failure{
202        script{
203          println("failure..")
204          //gitlab.ChangeCommitStatus(projectId,commitSha,"failed")
205        }
206      }
207      aborted{
208        script{
209          println("aborted..")
210          //gitlab.ChangeCommitStatus(projectId,commitSha,"canceled")
211        }
212      }
```

由于我资源的不足，所以必须把k8s的gitlab先卸载，换成soarqube，等测试通过后，集成到jenkins，jenkinsfile写好，然后把sonarqube的内容注释掉，再把gitlab换回来，再加上容器部署的内容。保证jenkins流程的完整，jenkinsfile内容的完整。

由于jenkins重新部署过，第一次编译比较耗时：

Pipeline microservice-demo-20221111

📝 Recent Changes

Stage View

		Declarative: Checkout SCM	拉取代码	项目构建	Declarative: Post Actions
Average stage times: (Average full run time: ~12min 26s)		3s	9s	1min 18s	489ms
#15 Nov 14 10:13	No Changes	9s	23s	11min 44s	468ms
#14 Nov 14 10:11	No Changes	4s	18s	1s failed	397ms

Workspace

📁 046-springboot-dubbo-ssm-provider / target / ➡️

- 📁 classes
- 📁 generated-sources/annotations
- 📁 generated-test-sources/test-annotations
- 📁 maven-archiver
- 📁 maven-status/maven-compiler-plugin
- 📁 surefire-reports
- 📁 test-classes/com/frank/springboot
- 📄 046-springboot-dubbo-ssm-provider-v1.0.jar   2022年11月14日 上午2:23:02   42.52 MB 📄 view
- 📄 046-springboot-dubbo-ssm-provider-v1.0.jar.original 2022年11月14日 上午2:22:09   13.13 KB 📄 view

📦 (all files in zip)

Workspace

📁 **047-springboot-dubbo-ssm-consumer** / **target** / [_____] ➡

    📁 **classes**

    📁 **docker**

    📁 **generated-sources/annotations**

    📁 **generated-test-sources/test-annotations**

    📁 **maven-archiver**

    📁 **maven-status/maven-compiler-plugin**

    📁 **surefire-reports**

    📁 **test-classes**

    📄 **047-springboot-dubbo-ssm-consumer-1.0-SNAPSHOT.jar**   2022年11月14日 上午2:23:47   30.07 MB 📰 **view**

    📄 **047-springboot-dubbo-ssm-consumer-1.0-SNAPSHOT.jar.original**   2022年11月14日 上午2:23:44   13.03 KB 📰 **view**

    👜 **(all files in zip)**