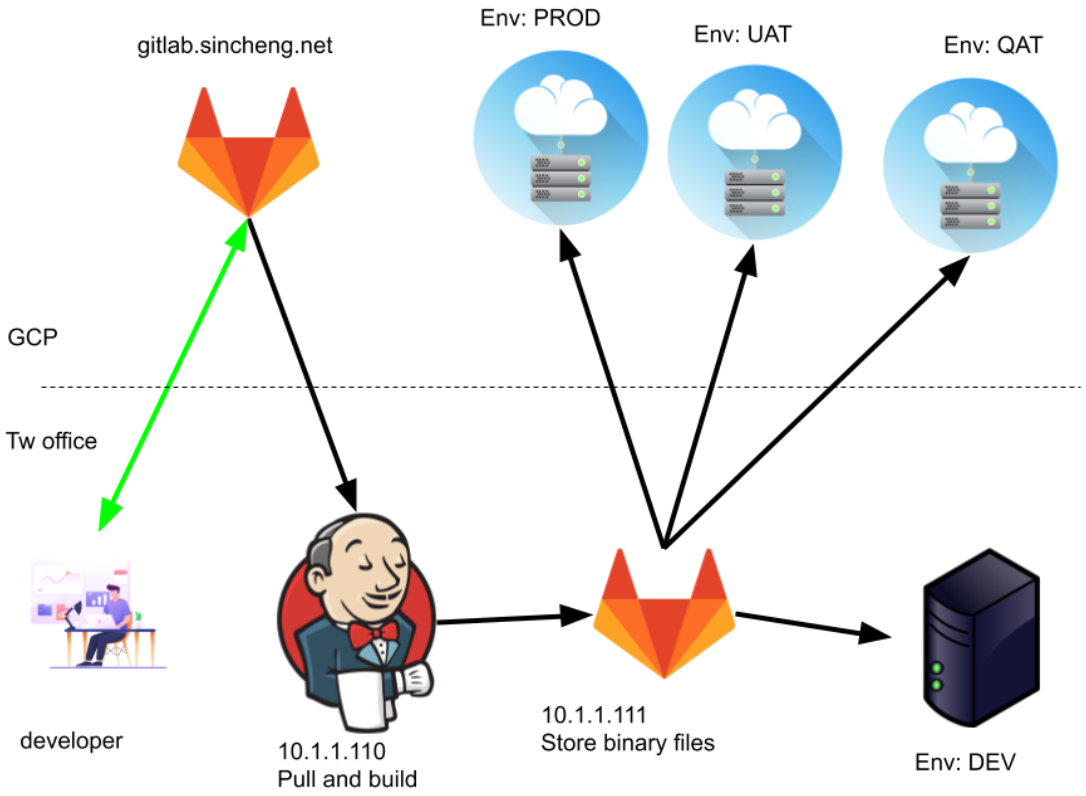
jenkins in sc.local (twoffice)



servers

local jenkins 10.1.1.110	http://jenkins.sc.local:8080/
local gitlab 10.1.1.111 (binary repo)	http://gitlab.sc.local/
gcp gitlab	https://gitlab.sincheng.net/

accounts

local jenkins 10.1.1.110	webgui: admin / e01833ae6e2f4e8289e6aabd464cf22d
local gitlab 10.1.1.111 (binary repo)	webgui: root / 36meszp6H6LwHVA deploy / DJzNoUz0OysR (account for binary push
gcp gitlab	deploy@sincheng.net / ?ub==KcfmdtM5kE,

	(account for gcp gitlab.sincheng.net alias to ray.huang@sincheng.net)

deploymgr (local
deploymgr_gcp (gcp

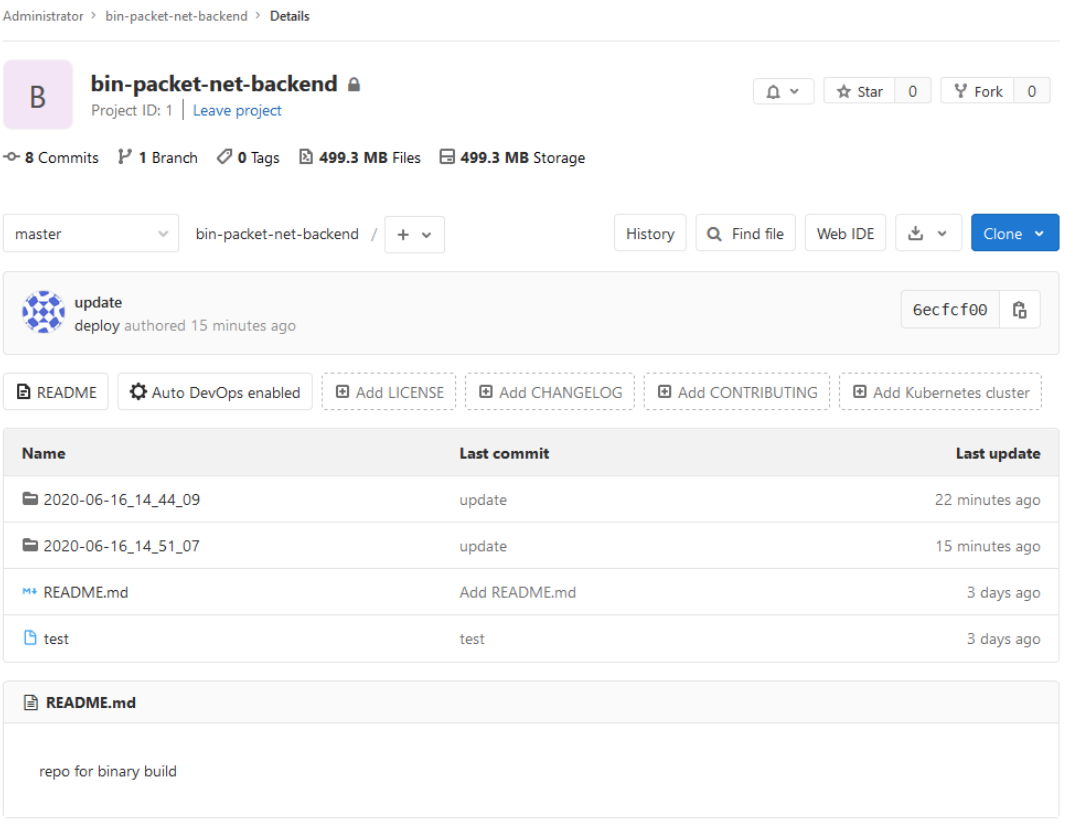
jenkins_output

project= http://jenkins.sc.local:8080/job/build-packet-net-backend/
console output

```
[master 6ecfcf0] update
 8 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 2020-06-16_14_51_07/activity-1.0.jar
 create mode 100644 2020-06-16_14_51_07/admin-1.0.jar
 create mode 100644 2020-06-16_14_51_07/auth-1.0.jar
 create mode 100644 2020-06-16_14_51_07/dataAnalyze-1.0.jar
 create mode 100644 2020-06-16_14_51_07/file-manager-1.0.jar
 create mode 100644 2020-06-16_14_51_07/gateway-1.0.jar
 create mode 100644 2020-06-16_14_51_07/pay-1.0.jar
 create mode 100644 2020-06-16_14_51_07/user-1.0.jar
+ git push origin master
To gitlab.sc.local:root/bin-packet-net-backend.git
   e8d737f..6ecfcf0  master -> master
Not sending mail to unregistered user charlie0275@gmail.com
Not sending mail to unregistered user austin.wang@sincheng.net
Not sending mail to unregistered user henk.li@sincheng.net
Not sending mail to unregistered user charlie.lee@sincheng.net
Sending e-mails to: ray.huang@sincheng.net
Finished: SUCCESS
```

local gitlab

project = git@gitlab.sc.local:root/bin-packet-net-backend.git



deploy to dev

[deploymgr@centos7 ~]$ cat deploy.sh
to be continued

deploymgr pull from local gitlab with binary files

```
[deploymgr@centos7 packet-net-backend]$ git remote -v
origin  git@gitlab.sc.local:root/bin-packet-net-backend.git (fetch)
origin  git@gitlab.sc.local:root/bin-packet-net-backend.git (push)
[deploymgr@centos7 packet-net-backend]$ git pull origin master
From gitlab.sc.local:root/bin-packet-net-backend
 * branch            master     -> FETCH_HEAD
[deploymgr@centos7 packet-net-backend]$ ll
total 8
drwxrwxr-x. 2 deploymgr deploymgr 188 Jun 17 11:31 2020-06-16_14_44_09
drwxrwxr-x. 2 deploymgr deploymgr 188 Jun 17 11:31 2020-06-16_14_51_07
-rw-rw-r--. 1 deploymgr deploymgr  21 Jun 17 11:31 README.md
-rw-rw-r--. 1 deploymgr deploymgr   9 Jun 17 11:31 test
[deploymgr@centos7 packet-net-backend]$ pwd
/home/deploymgr/gitlab.sc.local/packet-net-backend
```

- cd $(ls -dt */ | head -n1) # cd 進入最新的目錄
- cd $(ls -dt */ | head -n2 | tail -n1) # cd進入倒數第二新的目錄

branch

- master (develop
- UAT
- PROD

deploy to uat

to be continued

deploy to prod

to be continued

####################OLD DATA################################

flow
---
rd > gcp gitlab(code) > jenkins (compile, test) > binary upload > production update

project for test
---
packet-net-backend (這等於是deploy.sh 中 1

	在那台機器，會跑什麼指令 或批次
rd	目前自動佈署只在10.1.1.101，deploymgr下的deploy.sh(會調/opt/gm/{module}/下的start.sh ray> 如下附錄[note1^]
gitlab	(source在gcp cd-tw)應該是要從這裡拉code來build 我們依環境與分支對應了master -> local QAT -> qat PROD -> uat PROD -> production 這邊的話，我覺得先最簡單的。 如果要autodeploy，jenkins應該要webhook gitlab [note3^]，沒有就不用。 ray> webhook要上，只能先上在qa 最多到 uat，prod一定是手動上版 jenkins ui 點下去 git checkout (環境 master,QAT UAT,PROD，或一開就對應好，就不用這步) git pull orgiin (執行 test package ….) production環境的佈署應該是用uat的建置好就直接佈署 ray> 這段可以和上面的一起看，就是 rd要上傳至gcp的gitlab (packet-net-backend 專案)，會怎上 怎進版，怎換分支之類的
jenkins	(看ray兄你建哪)這個拉code就是git指令，run test ,run compile,run rd 這個部份都是用個人帳號，因為異動也是由個人帳號上傳。

	ray>以 packet-net-backend專案，您現在拉code下來會怎做，應會有建議的帳號與git指令
compile	(jenkins server)這個稍等 "C:\Program Files\Java\jdk1.8.0_251\bin\java.exe" -Dmaven.multiModuleProjectDirectory=C:\Source\packet-net-backend "-Dmaven.home=C:\Program Files\JetBrains\IntelliJ IDEA 2020.1.1\plugins\maven\lib\maven3" "-Dclassworlds.conf=C:\Program Files\JetBrains\IntelliJ IDEA 2020.1.1\plugins\maven\lib\maven3\bin\m2.conf" "-Dmaven.ext.class.path=C:\Program Files\JetBrains\IntelliJ IDEA 2020.1.1\plugins\maven\lib\maven-event-listener.jar" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2020.1.1\lib\idea_rt.jar=55086:C:\Program Files\JetBrains\IntelliJ IDEA 2020.1.1\bin" -Dfile.encoding=UTF-8 -classpath "C:\Program Files\JetBrains\IntelliJ IDEA 2020.1.1\plugins\maven\lib\maven3\boot\plexus-classworlds-2.6.0.jar;C:\Program Files\JetBrains\IntelliJ IDEA 2020.1.1\plugins\maven\lib\maven3\boot\plexus-classworlds.license" org.codehaus.classworlds.Launcher -Didea.version2020.1.1 package ↑↑↑ 提供的是ide的指令，jenkins環境可能沒有ide，maven會有，指令還是要測一下 ray>這個您寫現在您本地怎做就好，jenkins怎做我這兒測他
test	(jenkins server)這個稍等 "C:\Program Files\Java\jdk1.8.0_251\bin\java.exe" -Dmaven.multiModuleProjectDirectory=C:\Source\packet-net-backend "-Dmaven.home=C:\Program Files\JetBrains\IntelliJ IDEA 2020.1.1\plugins\maven\lib\maven3" "-Dclassworlds.conf=C:\Program Files\JetBrains\IntelliJ IDEA 2020.1.1\plugins\maven\lib\maven3\bin\m2.conf" "-Dmaven.ext.class.path=C:\Program Files\JetBrains\IntelliJ IDEA 2020.1.1\plugins\maven\lib\maven-event-listener.jar" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2020.1.1\lib\idea_rt.jar=55215:C:\Program Files\JetBrains\IntelliJ IDEA 2020.1.1\bin" -Dfile.encoding=UTF-8 -classpath "C:\Program Files\JetBrains\IntelliJ IDEA 2020.1.1\plugins\maven\lib\maven3\boot\plexus-classworlds-2.6.0.jar;C:\Program Files\JetBrains\IntelliJ IDEA 2020.1.1\plugins\maven\lib\maven3\boot\plexus-classworlds.license" org.codehaus.classworlds.Launcher -Didea.version2020.1.1 test ↑↑↑ 提供的是ide的指令，jenkins環境可能沒有ide，maven會有，指令還是要測一下 ray>這個您寫現在您本地怎做就好，jenkins怎做我這兒測他
binary upload	(jenkins operate to local gitlab)如果是上傳到gitlab，那就是git 指令，git add,git commit git push …. ray>這段我們暫時沒有，我來補
production update	這個不太確定是什麼

--rd本地執行腳本

```
#!/bin/bash
#遠端ssh認證需先設定好
#
#

#環境別
targetEnv=prod
#專案根目錄
rootProjectDir=/d/sincheng.net/Source/packet-net-backend/
#佈署檔案放置處
rootDir=/d/sincheng.net/release/packet-net-backend/
targetDir=${rootDir}${targetEnv}/
targetDataDir=${rootDir}${targetEnv}/dataAnalyze/
targetJobDir=${rootDir}${targetEnv}/job/


projects=(admin auth gateway pay activity user file-manager)
projectsData=(dataAnalyze)
projectsJob=(job)

echo "targetEnv:${targetEnv}"
echo "targetDir:${targetDir}"

mkdir -p $targetDir

for((i=0; i<${#projects[@]}; i++));
do
  cp -f ${rootProjectDir}${projects[$i]}/target/${projects[$i]}-1.0.jar ${targetDir}
done

scp ${targetDir}*.jar (你的帳號)@34.80.88.184:~

##job
echo "################################"

echo "targetJobDir:${targetJobDir}"

mkdir -p $targetJobDir

for((i=0; i<${#projectsJob[@]}; i++));
do
  cp -f ${rootProjectDir}${projectsJob[$i]}/target/${projectsJob[$i]}-1.0.jar
${targetJobDir}
done

scp ${targetJobDir}*.jar (你的帳號)@35.229.137.233:~

##資料分析
echo "################################"

echo "targetDataDir:${targetDataDir}"
```

	mkdir -p $targetDataDir
	
	for((i=0; i<${#projectsData[@]}; i++));
	do
	  cp -f ${rootProjectDir}${projectsData[$i]}/target/${projectsData[$i]}-1.0.jar ${targetDataDir}
	  if [${projectsData[$i]} == 'dataAnalyze']; then
	  ##資料分析設定檔
	  cp -f ${rootProjectDir}${projectsData[$i]}/target/classes/environment.properties ${targetDataDir}
	  cp -f ${rootProjectDir}${projectsData[$i]}/target/classes/environment_${targetEnv}.properties ${targetDataDir}
	  fi
	done
	
	scp ${targetDataDir}* (你的帳號)@35.229.191.247:~
	
	↑↑↑
	上傳後調用腳本，建了deploymgr_gcp，應該也是用這個帳號上傳
	
	ray> 我們現在把程式放到正式環境的方法

[note1]:

[root@centos7 ~]# find / -name deploy.sh
/home/deploymgr/deploy.sh

log放在同目錄20200527143255153595208.deploy
有一個 $1 的參數，接數字 做 不同程式的動作，到jenkins改string (switch) combobox

1	auth cp /home/deploymgr/auth-1.0.jar /opt/gm/auth/ cd /opt/gm/auth/;sudo /opt/gm/auth/start.sh	5	admin sudo cp -f /home/deploymgr/admin-1.0.jar /opt/gm/admin/ cd /opt/gm/admin/;sudo /opt/gm/admin/start.sh	9	xx
2	gatewayh5	6	**activity**	1	dataAnalyze

	sudo cp -f /home/deploymgr/gateway-1.0.jar /opt/gm//gateway/ cd /opt/gm/gateway;sudo /opt/gm/gateway/member.sh start		**sudo cp -f /home/deploymgr/activity-1.0.jar /opt/gm/activity/** **cd /opt/gm/activity;sudo /opt/gm/activity/start.sh**	0	sudo docker cp /home/deploymgr/dataAnalyze-1.0.jar ${nimbusID}:/topo/ sudo docker cp /home/deploymgr/admin-1.0.jar ${nimbusID}:/topo/ sudo docker cp /home/deploymgr/environment.properties ${nimbusID}:/topo/ sudo docker cp /home/deploymgr/environment_${currentEnv}.properties ${nimbusID}:/topo/ sudo docker-compose -f /usr/local/src/storm-mono-docker/docker-compose.yml stop sudo docker-compose -f /usr/local/src/storm-mono-docker/docker-compose.yml start #add hosts for cid in $(sudo docker ps | awk '{print $1}' | grep -v 'CONTAINER'); do sudo docker exec -i $cid bash -c "echo '10.1.1.101 centos7' >> /etc/hosts" &> /dev/null
3	gatewayAdmin sudo cp -f /home/deploymgr/gateway-1.0.jar /opt/gm//gateway/ cd /opt/gm/gateway;sudo /opt/gm/gateway/admin.sh start	7	#user sudo cp -f /home/deploymgr/user-1.0.jar /opt/gm/user/ cd /opt/gm/user;sudo /opt/gm/user/start.sh		
4	pay sudo cp -f /home/deploymgr/pay-1.0.jar /opt/gm/pay/ cd /opt/gm/pay/;sudo /opt/gm/pay/start.sh	8	file-manager sudo cp -f /home/deploymgr/file-manager-1.0.jar /opt/gm/file-manager/ cd /opt/gm/file-manager;sudo /opt/gm/file-manager/start.sh		

```
[deploymgr@centos7 ~]$ cat deploy.sh  | grep cp
   sudo cp -f /home/deploymgr/auth-1.0.jar /opt/gm/auth/
   sudo cp -f /home/deploymgr/gateway-1.0.jar /opt/gm//gateway/
   sudo cp -f /home/deploymgr/gateway-1.0.jar /opt/gm//gateway/
   sudo cp -f /home/deploymgr/pay-1.0.jar /opt/gm/pay/
   sudo cp -f /home/deploymgr/admin-1.0.jar /opt/gm/admin/
   sudo cp -f /home/deploymgr/activity-1.0.jar /opt/gm/activity/
   sudo cp -f /home/deploymgr/user-1.0.jar /opt/gm/user/
   sudo cp -f /home/deploymgr/file-manager-1.0.jar /opt/gm/file-manager/
      sudo docker cp /home/deploymgr/dataAnalyze-1.0.jar ${nimbusID}:/topo/
      sudo docker cp /home/deploymgr/admin-1.0.jar ${nimbusID}:/topo/
      sudo docker cp /home/deploymgr/environment.properties ${nimbusID}:/topo/
      sudo docker cp /home/deploymgr/environment_${currentEnv}.properties
${nimbusID}:/topo/
```

用kill 先去之前的java 程式 (pidof java | wc ，這台有34 個 java程式
再到/opt/gm/function/start.sh 啟用
部份的程式是用docker 跑的

看起包好的jar檔也放在這個目錄下 (/home/deploymgr

```
[root@centos7 deploymgr]# ll
total 564608
-rw-r--r--. 1 root      root              0 May 27 14:32 20200527143255153595208.deploy
-rw-r--r--. 1 root      root              0 May 27 14:35 20200527143550026608068.deploy
-rw-r--r--. 1 deploymgr deploymgr  91051067 Jun  5 10:02 activity-1.0.jar
-rw-r--r--. 1 deploymgr deploymgr  92147610 Jun  5 10:02 admin-1.0.jar
-rw-r--r--. 1 deploymgr deploymgr  87223497 Jun  5 10:02 auth-1.0.jar
-rw-r--r--. 1 deploymgr deploymgr  35241936 Jun  5 10:02 dataAnalyze-1.0.jar
-rwxrw-r--. 1 deploymgr deploymgr        37 May 27 09:36 deploy2.sh
-rwxrw-r--. 1 deploymgr deploymgr      3643 May 27 14:54 deploy.sh
-rw-r--r--. 1 deploymgr deploymgr        21 Jun  5 10:02 environment_local.properties
-rw-r--r--. 1 deploymgr deploymgr       422 Jun  5 10:02 environment.properties
-rw-r--r--. 1 deploymgr deploymgr  38781052 Jun  5 10:02 file-manager-1.0.jar
-rw-r--r--. 1 deploymgr deploymgr  35201195 Jun  5 10:02 gateway-1.0.jar
-rw-r--r--. 1 deploymgr deploymgr 102472433 Jun  5 10:02 pay-1.0.jar
-rw-r--r--. 1 deploymgr deploymgr  96010110 Jun  5 10:02 user-1.0.jar
[root@centos7 deploymgr]#
```

[note2]:
maven
[root@tw-jenkins-110 ~]# mvn -version
Apache Maven 3.5.4 (Red Hat 3.5.4-5)
Maven home: /usr/share/maven
Java version: 1.8.0_252, vendor: Oracle Corporation, runtime:
/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.252.b09-2.el8_1.x86_64/jre
Default locale: en_US, platform encoding: UTF-8
OS name: "linux", version: "4.18.0-147.8.1.el8_1.x86_64", arch: "amd64", family: "unix"

https://waylau.com/build-java-project-with-maven/
mvn compile

mvn package

[note3]:
ref: https://github.com/jenkinsci/gitlab-plugin#jenkins-to-gitlab-authentication
section: Jenkins-to-GitLab authentication

deploy_api_token (in user deploy
token: PY5t6orGZSfVy5oXg74T