本文档涉及到的所有软件都在[\\172.16.0.52\superxon\ATE\GouJunping\Software](file:///\\172.16.0.52\superxon\ATE\GouJunping\Software)目录下

**Git**

**一．基本信息：**

1. 网址：<http://172.16.0.31/>

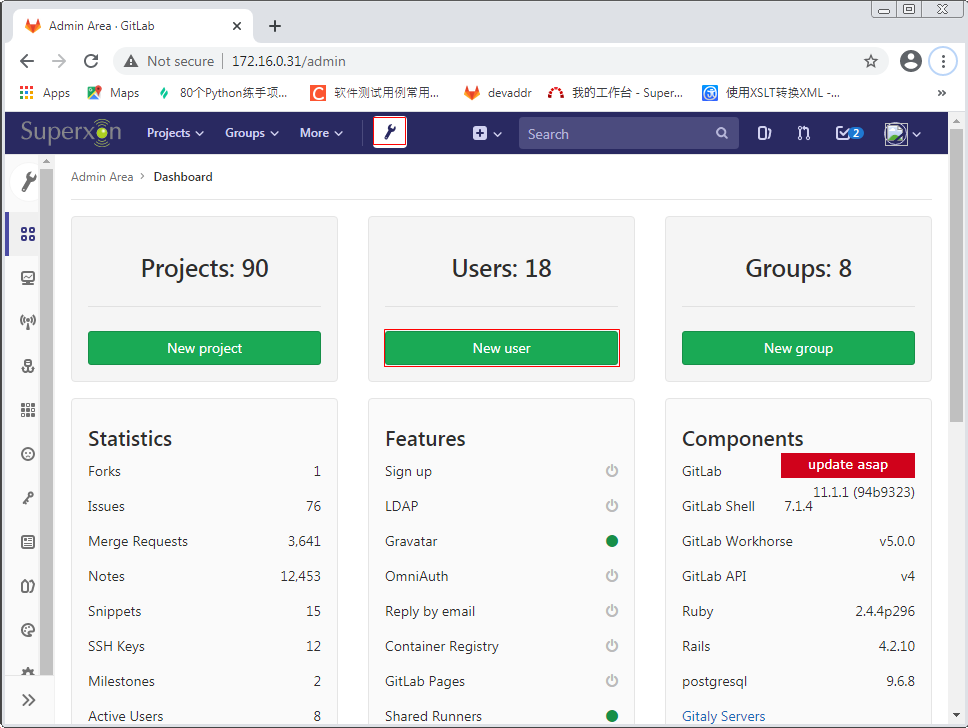
2. 服务器IP：172.16.0.31，登录用户名：superxon，登录密码：Superxon

3. GitLab的管理员账号：root，密码：12345678

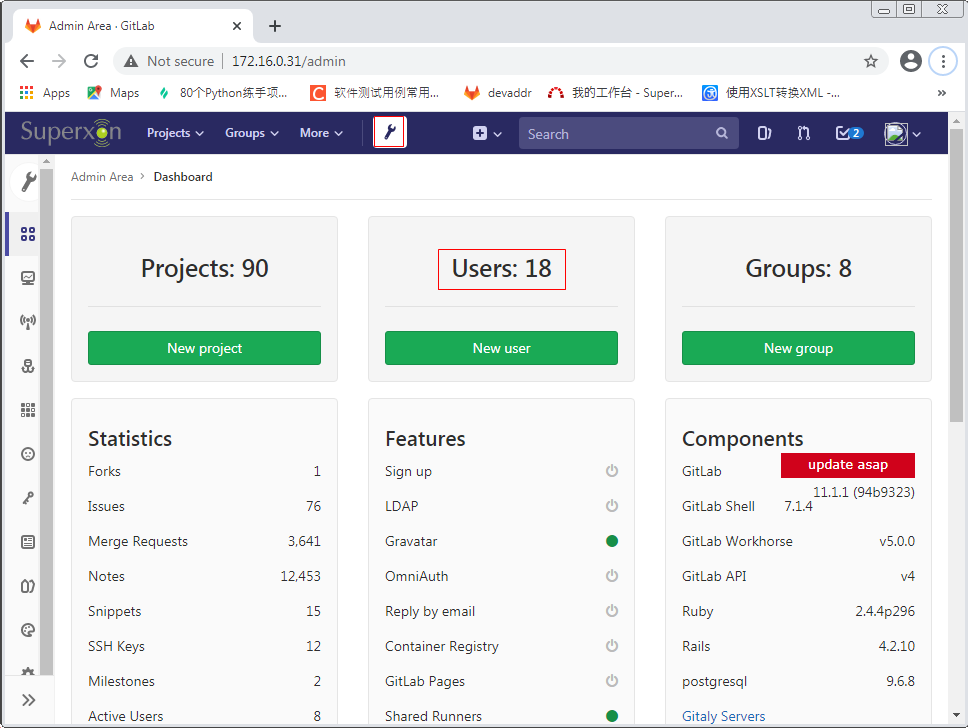
**二．常用操作：**

1. 用户管理：

新建用户：点击顶部扳手图标 -> 点击New user，注意：用户的Email请使用公司个人的cn邮箱



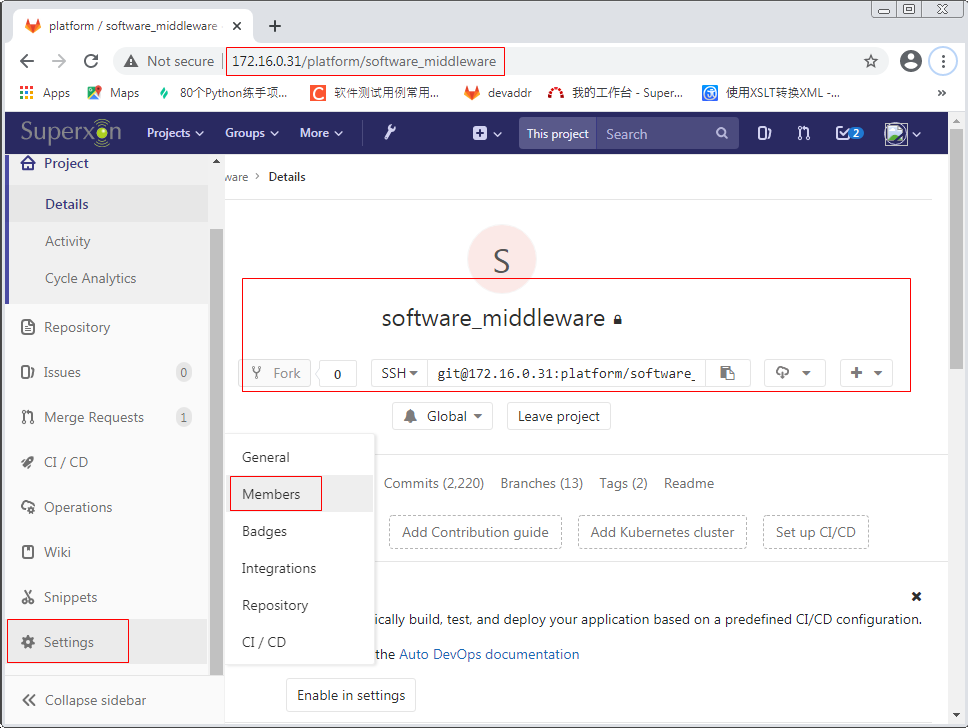
老用户管理：点击顶部扳手图标 -> 点击User: 18



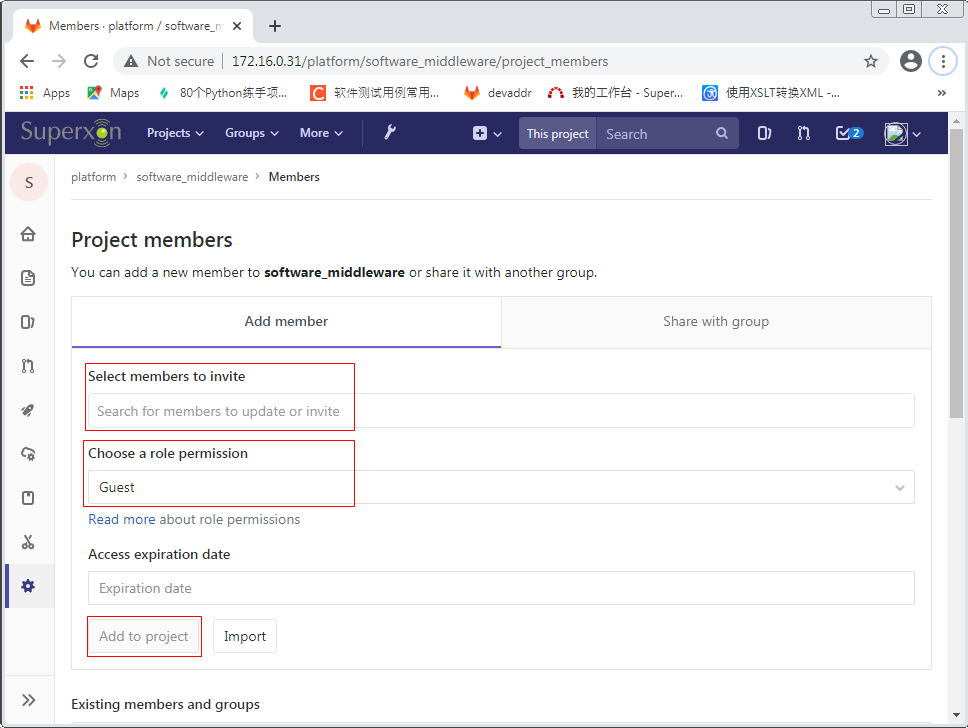
2. Projects和Groups管理跟User管理类似

3. Project和Group权限管理：

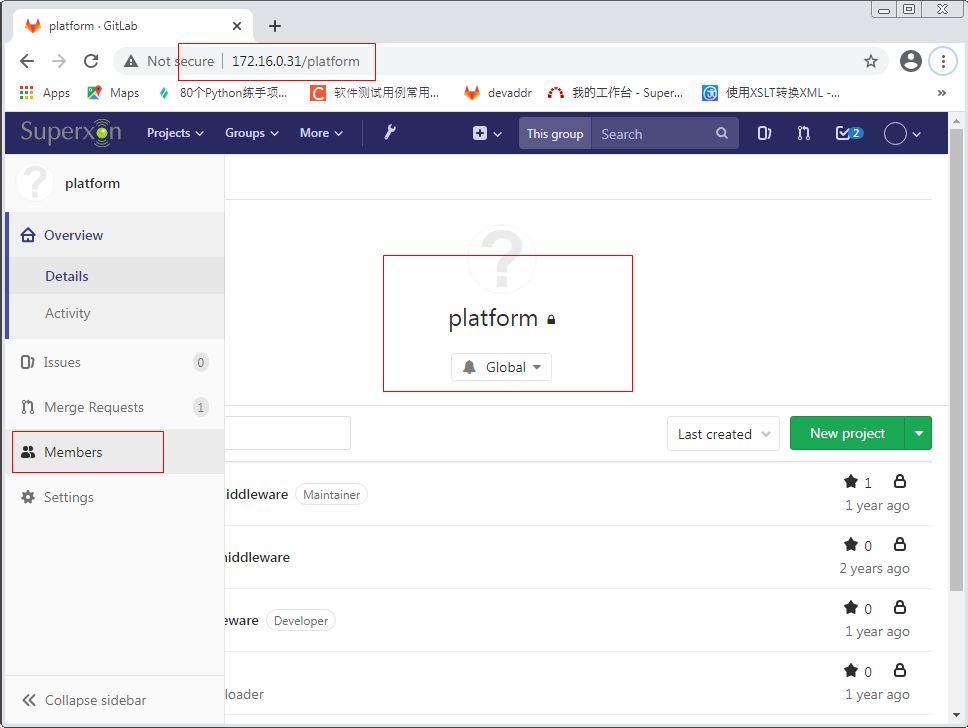
进入到工程页面过后，鼠标放到左下Settings位置，然后在弹出的菜单中点击Members



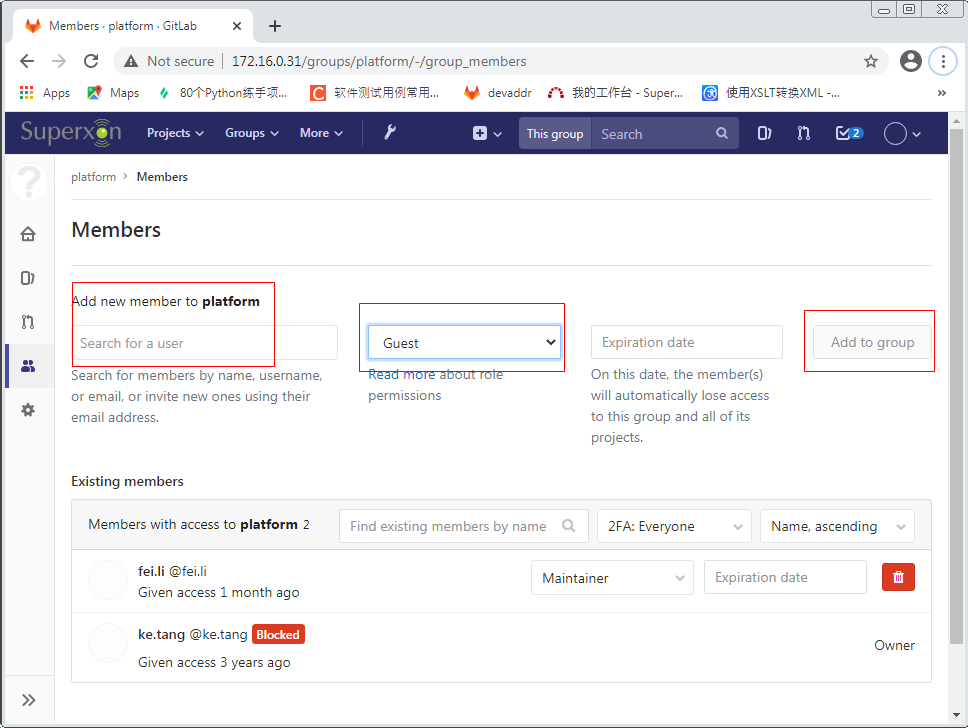
新的页面如下，选择对应的账号和权限，然后点击Add to project



也可以直接对整个Group添加权限，这样 Group下的所有Projects都具有相同的权限；添加权限方式类似，进入到group的页面，然后点击左侧的Members



分别添加账号，选择权限，点击Add to group



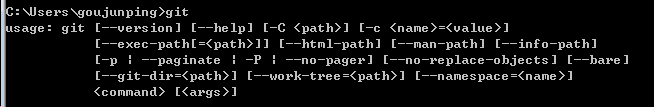
4. 客户端安装和配置：

分别安装以下两个软件

Git-2.24.0.2-32-bit.exe

TortoiseGit-2.7.0.0-64bit.msi

安装完成后在cmd中输入git会提示如下信息:



在命令行输入如下命令配置用户和邮箱：

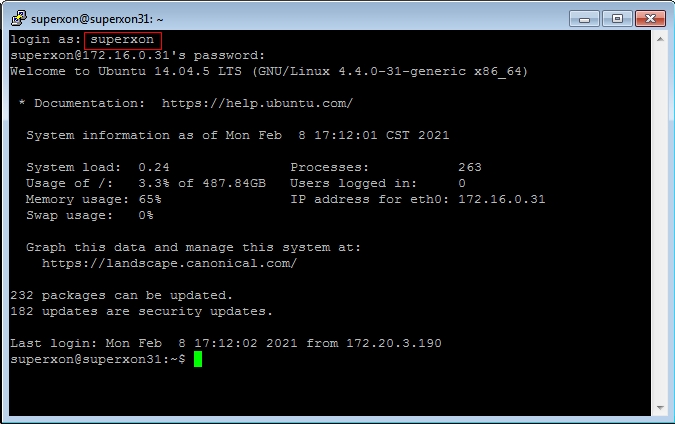
git config --global user.name “username” //username是自己的账户名

git config --global user.email “username@superxon.cn” // username@superxon.cn为创建用户时配置的邮箱

5. 自动创建带模板的工程

安装putty-0.70-installer.msi

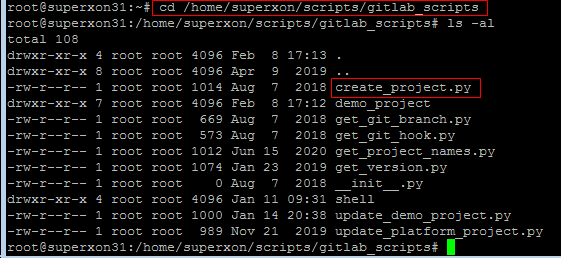
登录IP：172.16.0.31，用户名：superxon，密码：Superxon



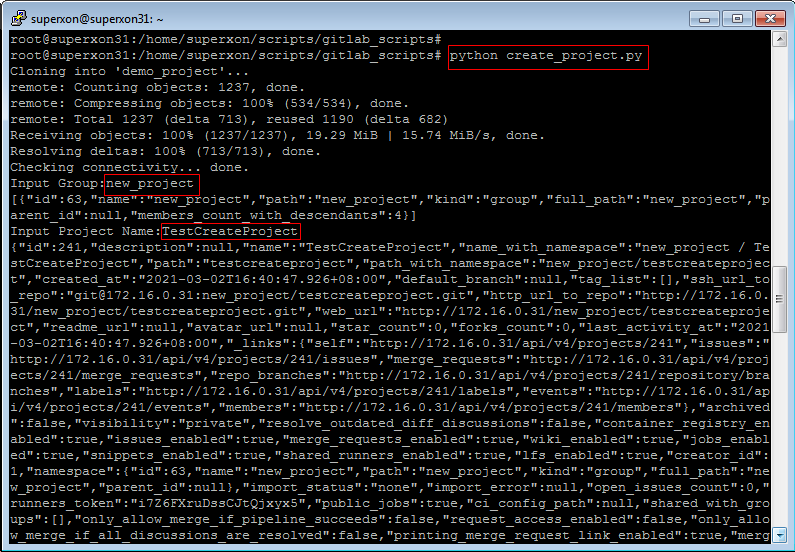
输入”su –“命令，密码Superxon，切换到root账号



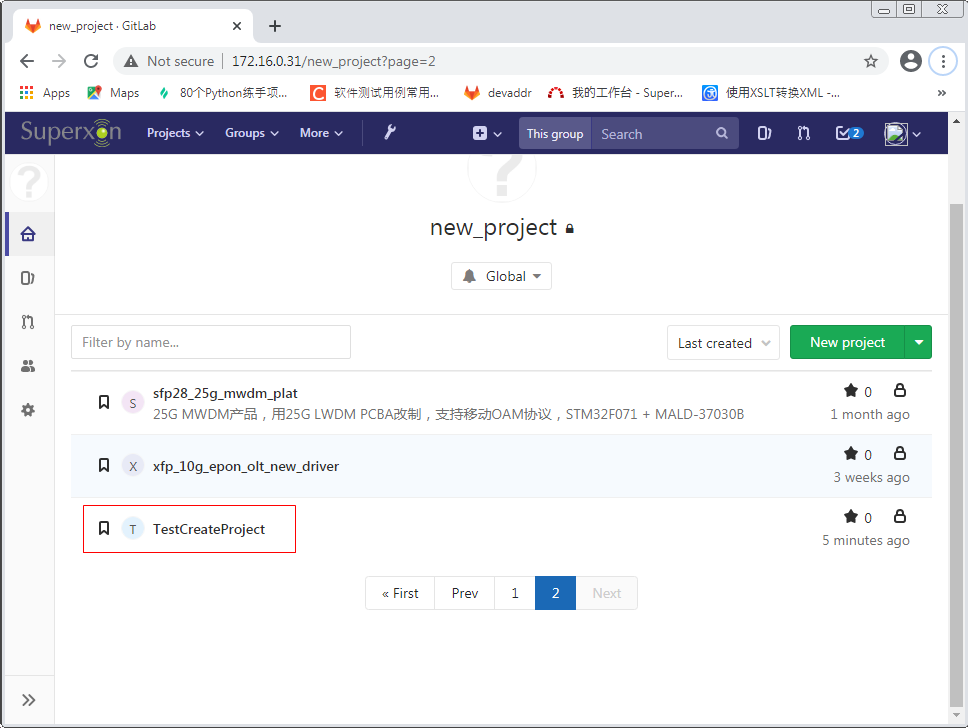
输入命令cd /home/superxon/scripts/gitlab\_scripts切换目录



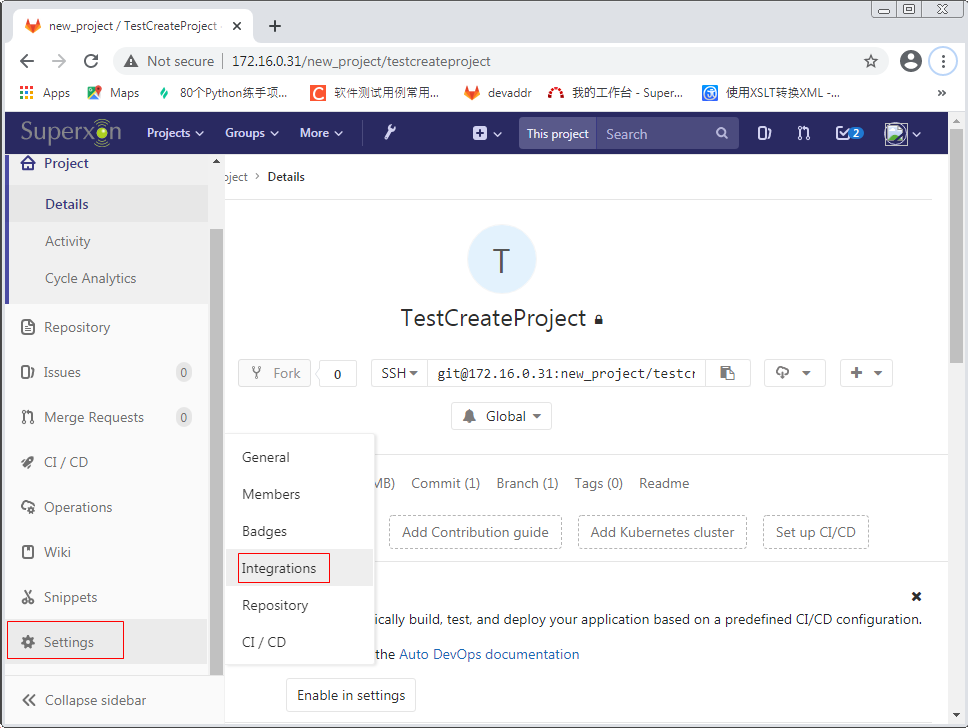
执行python create\_project.py命令，在提示Input Group的地方输入group的名字，在提示Input Project Name的地方输入新建工程的名字，等待脚本执行完成



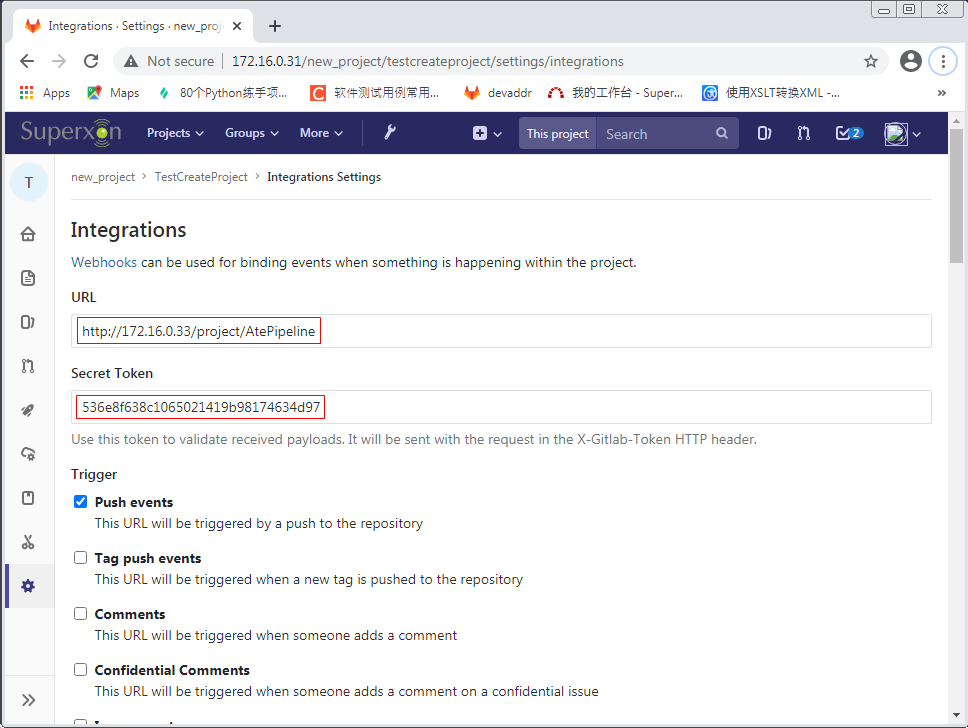
在Gitlab端可以看到刚创建的工程:



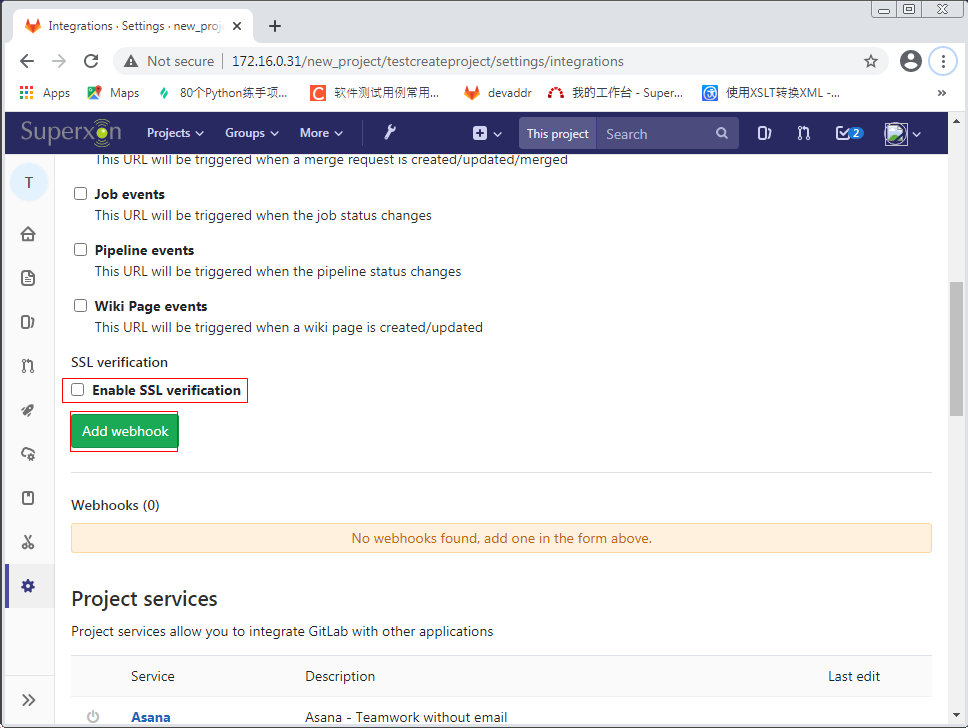
点击工程名，进入工程中，配置触发Jenkins出包任务：点击左下的Settings -> Integrations



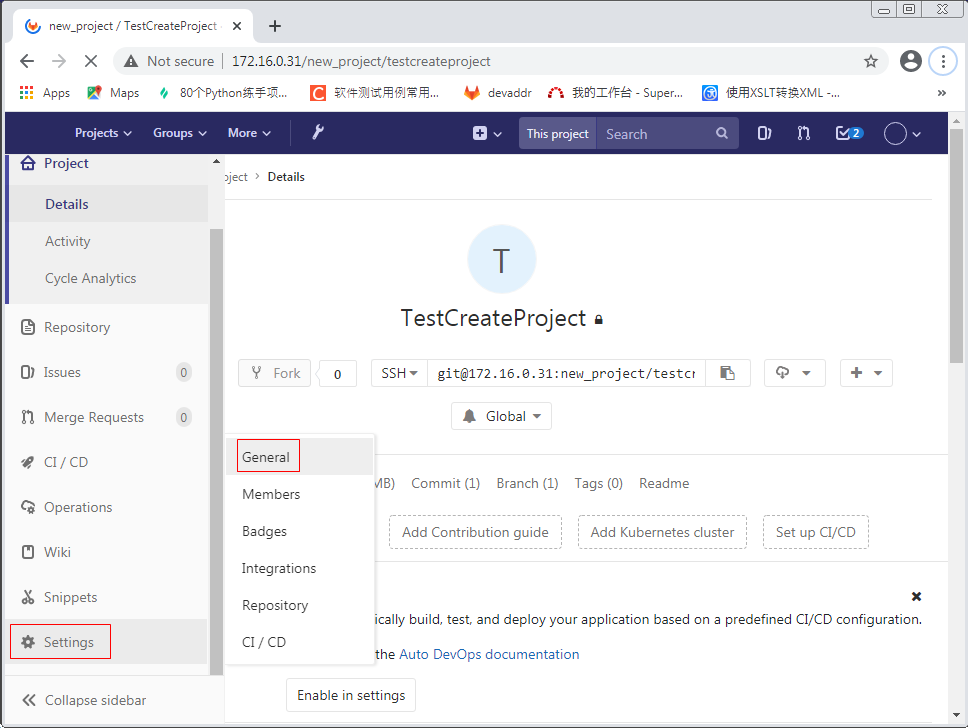
配置URL和Secret Token，URL为Jenkins服务器中的一个任务路径，Token为该任务中配置时随机生成的一个Token，除非Jenkins中配置变更了，否则gitlab中所有JOB都配置图中所示的值。这个两个值的生成会在Jenkins任务配置中讲解



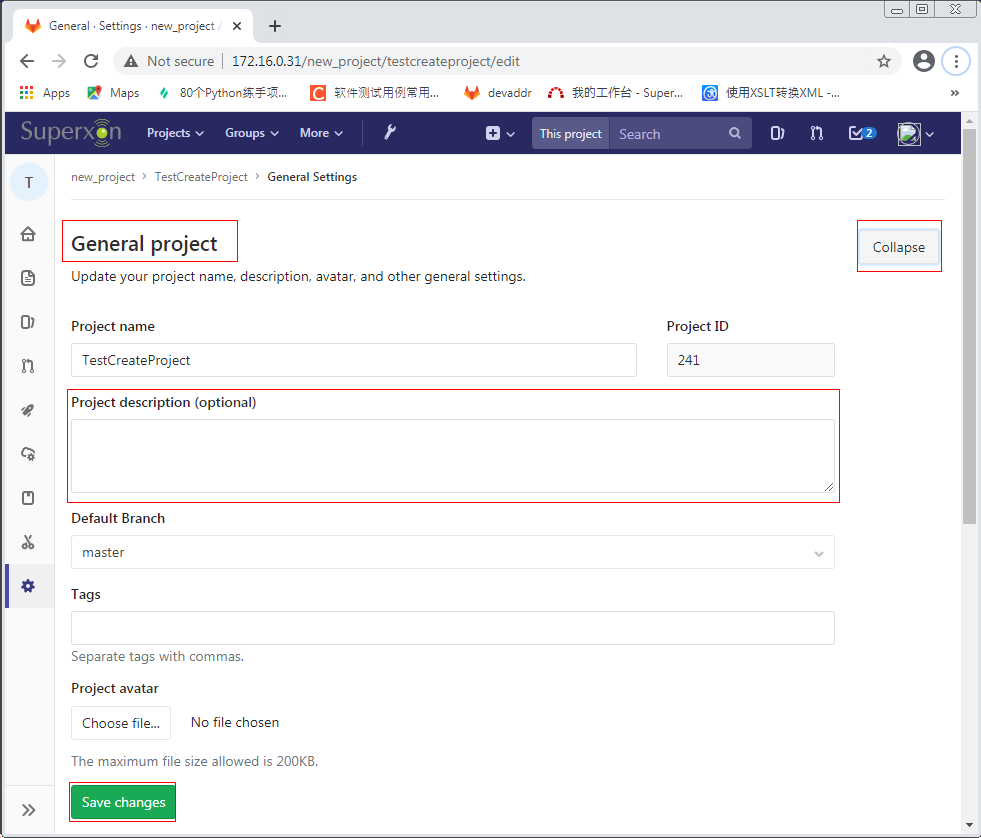
不要勾选Enable SSL verification，点击Add webhook即可



配置备注：在工程页面点击Settings中的General



展开General project，填如描述信息，点击Save changes即可



注意：这样创建的模板是根据<http://172.16.0.31/ate/demo_project>工程创建的，除了公共的脚本、工具或者文档，其他文件请不要上传到demo\_project工程中

**三．部分工程说明**

1. new\_project下的所有工程为新架构固件

2. project下的所有工程为老架构固件

3. ate group下的team工程存放部门管理文件

4. ate group下的demo\_project工程为创建新工程的模板

5. ate group下的evaluation\_board工程为评估板固件

6. ate group下的scripts工程为自动创建工程的脚本

7. platform group下的software\_middleware工程为动态库和通用软件

8. platform group下的tool\_middleware工程存放所有归档的工具类软件，里面的工具都归档在EXP-YZ0000123-TOOL\_KIT编号中

9. platform group下的bootloader工程存放芯片的bootloader代码

**Jenkins**

**一．基本信息：**

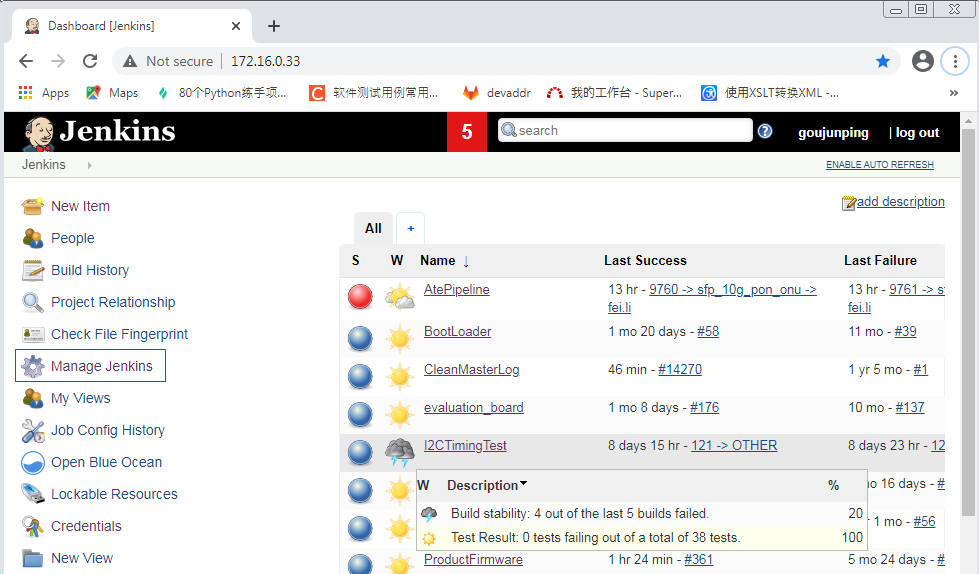
1. 网址：<http://172.16.0.33/>, admin账户密码admin

2. 服务器IP：172.16.0.33，登录用户名：superxon，登录密码：Superxon

3. JENKINS\_HOME路径/var/lib/jenkins

**二．常用操作:**

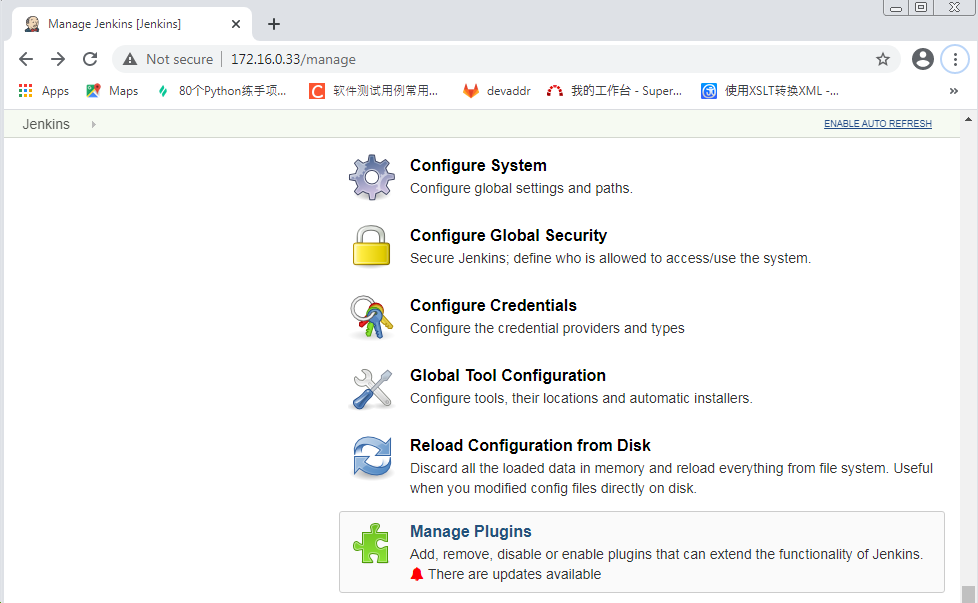
1. jenkins管理，点击Manage Jenkins



Configure System: 里面包括全局变量的配置(里面包括一些用户名和密码)，邮箱服务器的配置

Configure Global Security: 权限的配置

Manage Nodes: 节点管理



2. 节点管理：Jenkins节点包括master和slave，master是Jenkins服务器运行的虚拟机，slave是执行编译测试的虚拟机

Windows-172.16.0.35：固件编译节点，在何川管理的服务器上

Windows-172.16.0.36：软件编译节点，在何川管理的服务器上

Windows-172.16.0.37：软件编译节点，在何川管理的服务器上

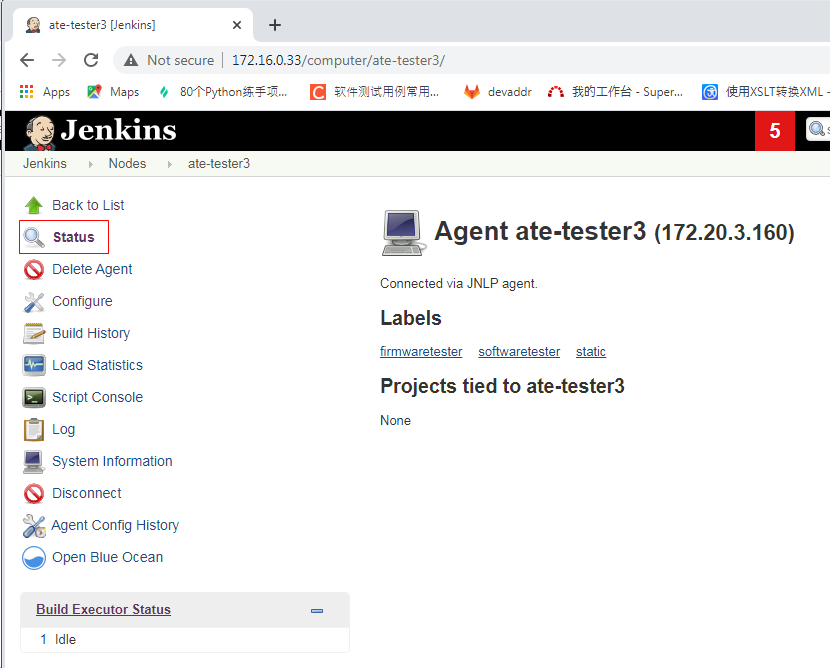
ate-tester1：自动化测试节点, 也包括代码静态解析, 172.20.3.164，在我们自己管理的服务器上

ate-tester2：自动化测试节点, 也包括代码静态解析, 172.20.3.158，在我们自己管理的服务器上

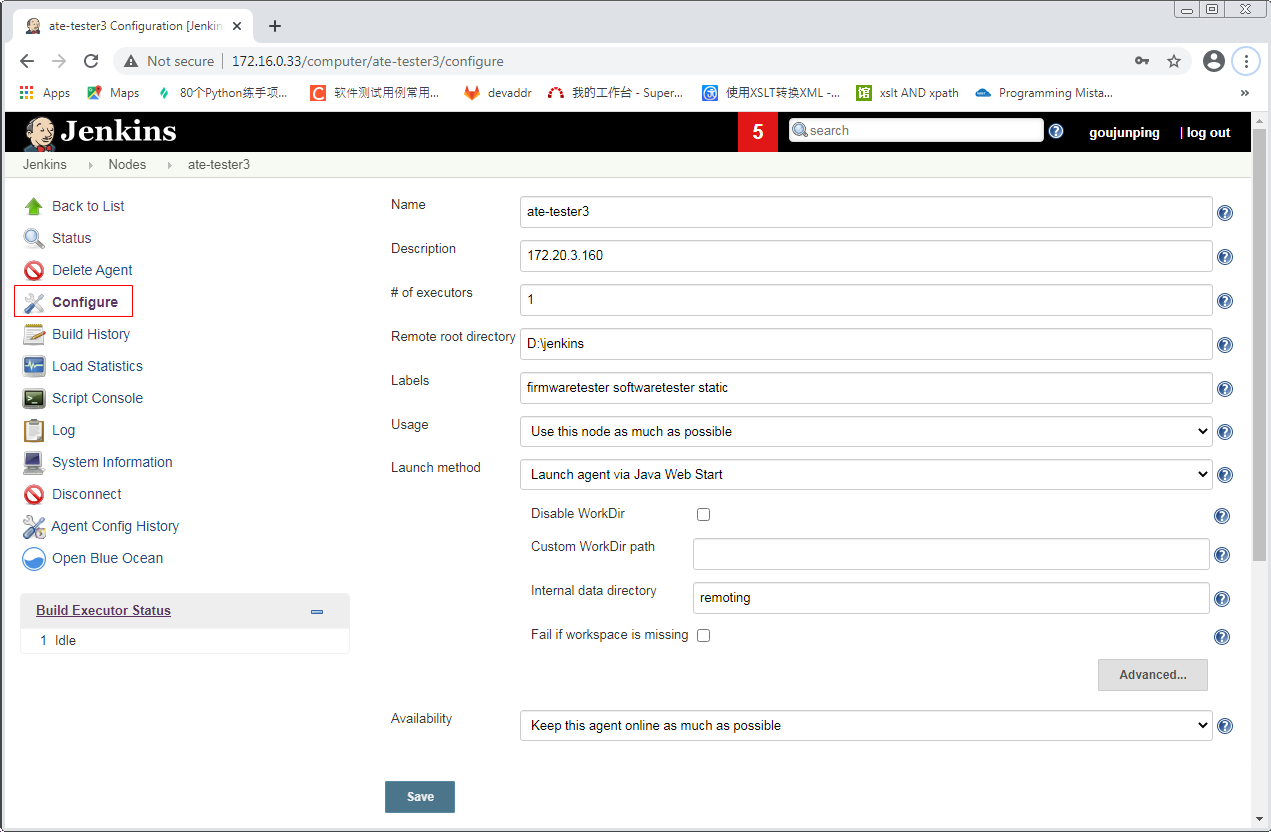
ate-tester3：自动化测试节点, 也包括代码静态解析, 172.20.3.160，在我们自己管理的服务器上

以上虚拟机的登录账号：administrator, 密码：123

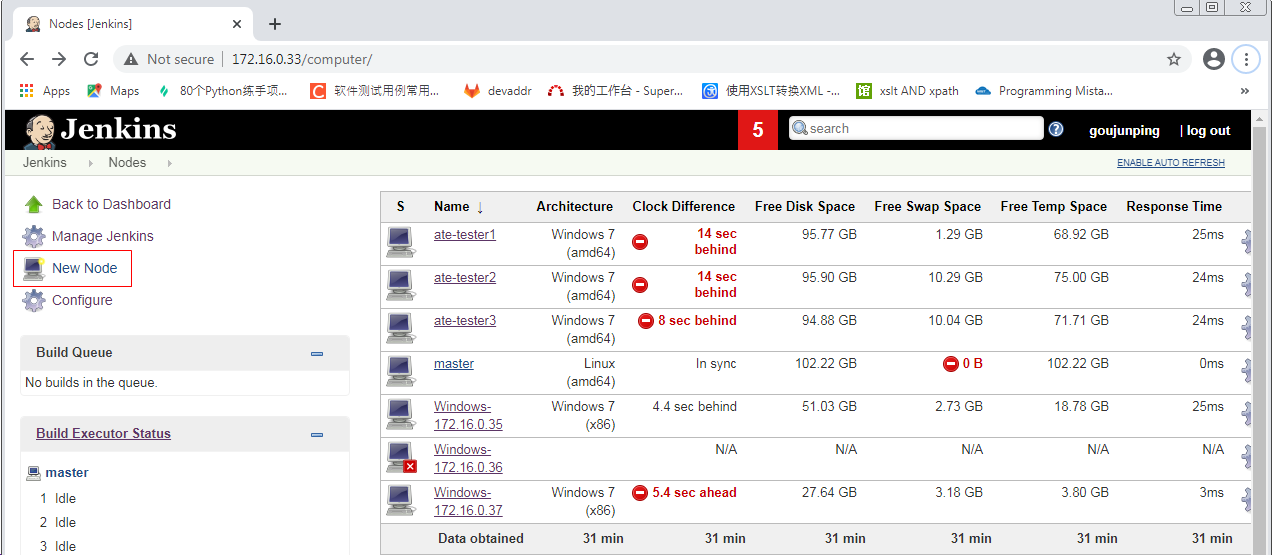
节点状态：Labels为节点标签，可以在配置任务的时候选择在那个标签上执行



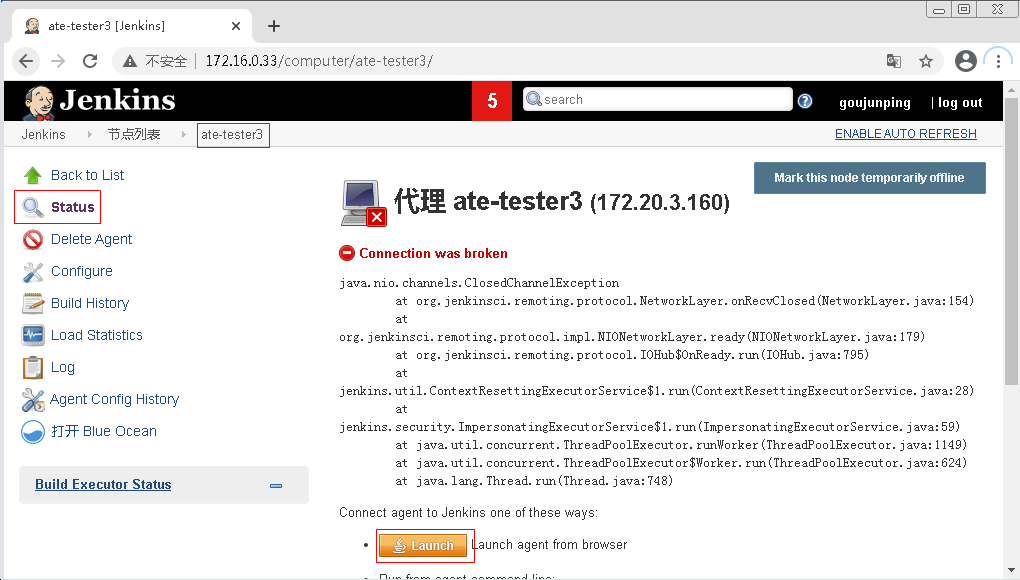
节点配置：Name节点名，Description描述，Remote root directory在虚拟机上执行任务的根目录，Lables标签(一般会把同类的多个节点配置为同一个标签)，Launch method连接方式(统一使用Launch agent via Java Web Start)



新建节点:



节点连接:新的虚拟机需要安装jdk-8u191-windows-x64.exe，远程桌面登录到要连接的虚拟机 -> 使用浏览器打开节点的status页面 -> 点击Launch会下载一个slave-agent.jnlp的文件 -> 双击该文件



等待自动连接成功，会显示一个jenkins agent的界面，上面提示Connected则说明已经连接成功了



连接成功后可以关闭web浏览器，退出远程桌面，Jenkins就可以访问该节点了；但是节点重启后又需要手动重复连接，可以根据如下步骤将其安装为一个Windows服务，则系统启动后会自动连接：

点击Jenkins agent界面中的File -> 点击菜单中的Install as server –> 等待安装成功后会自动关闭Jenkins agengt界面

3. 任务管理

AtePipeline: Git上project和new\_project下所有工程的编译测试任务

BootLoader: bootloader出包任务

ClCleanMasterLog: master服务器日志清理任务

evaluation\_board: 评估板固件出包任务

I2CTimingTest: I2C速率和异常时序测试任务

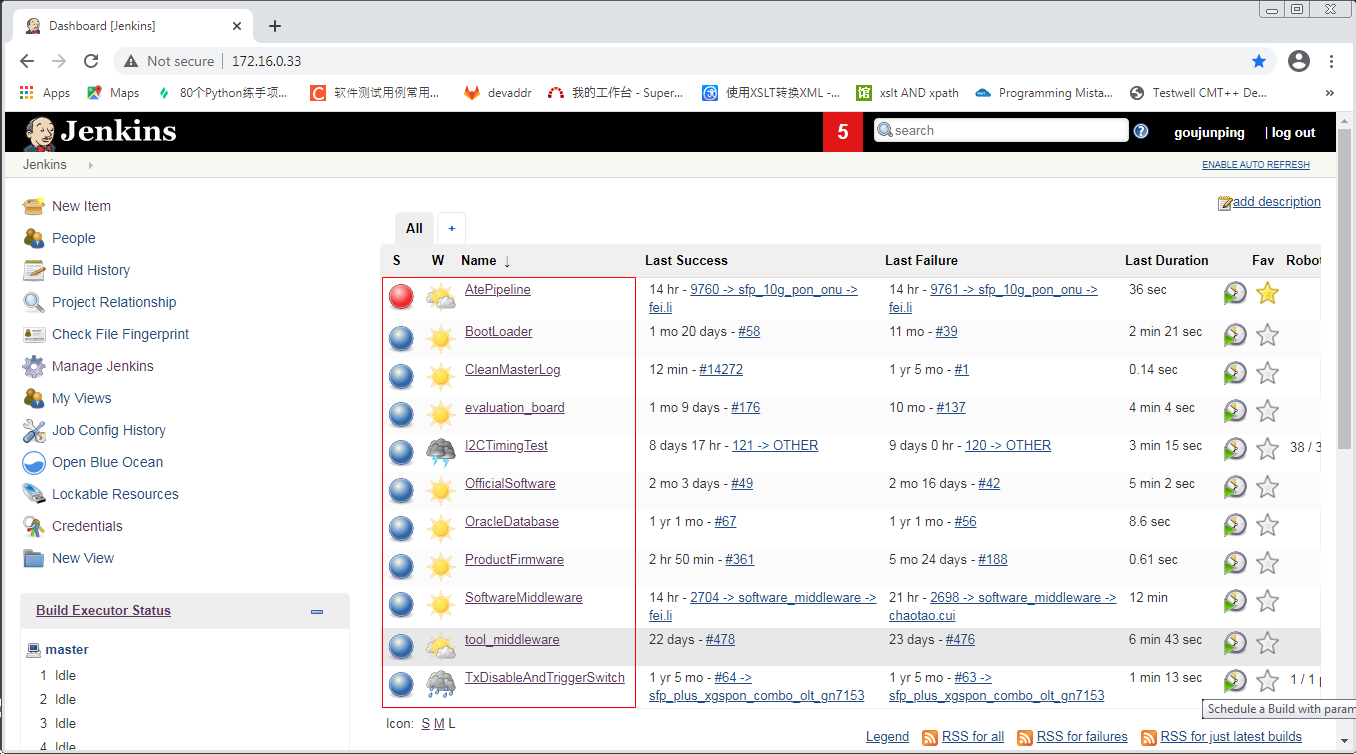
OfficialSoftware: 软件正式包出包任务

OracleDatabase: 数据库操作接口任务，未使用

ProductFirmware: 每天统计当前产品使用的固件版本

tool\_middleware: 工具包出包任务

TxDisableAndTriggerSwitch: TxDisable和Trigger切换测试

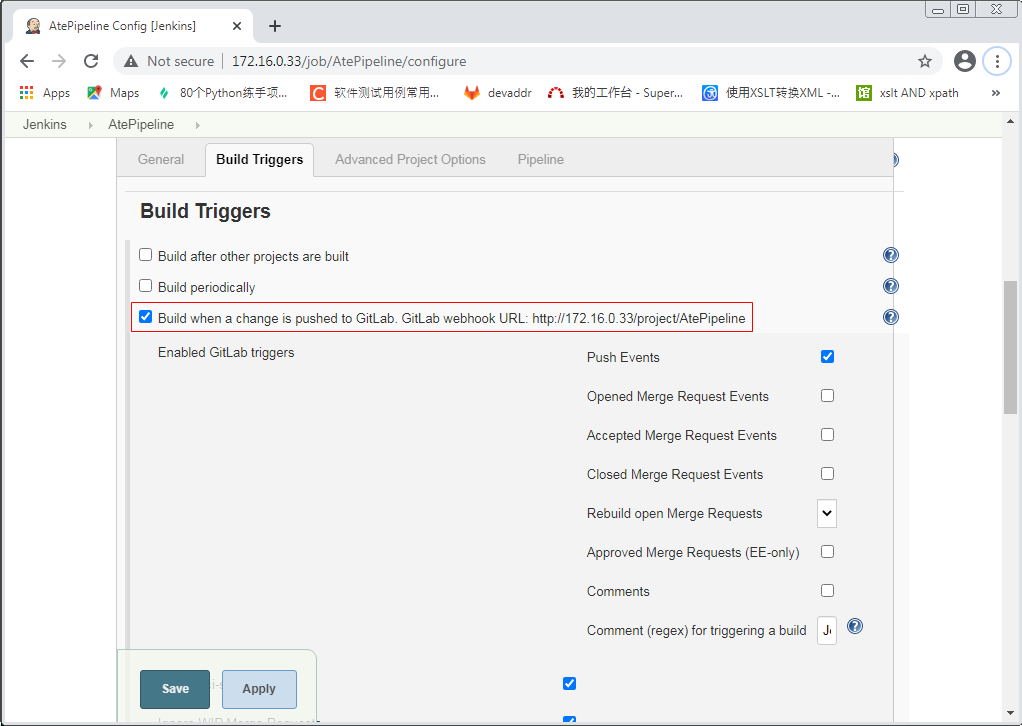


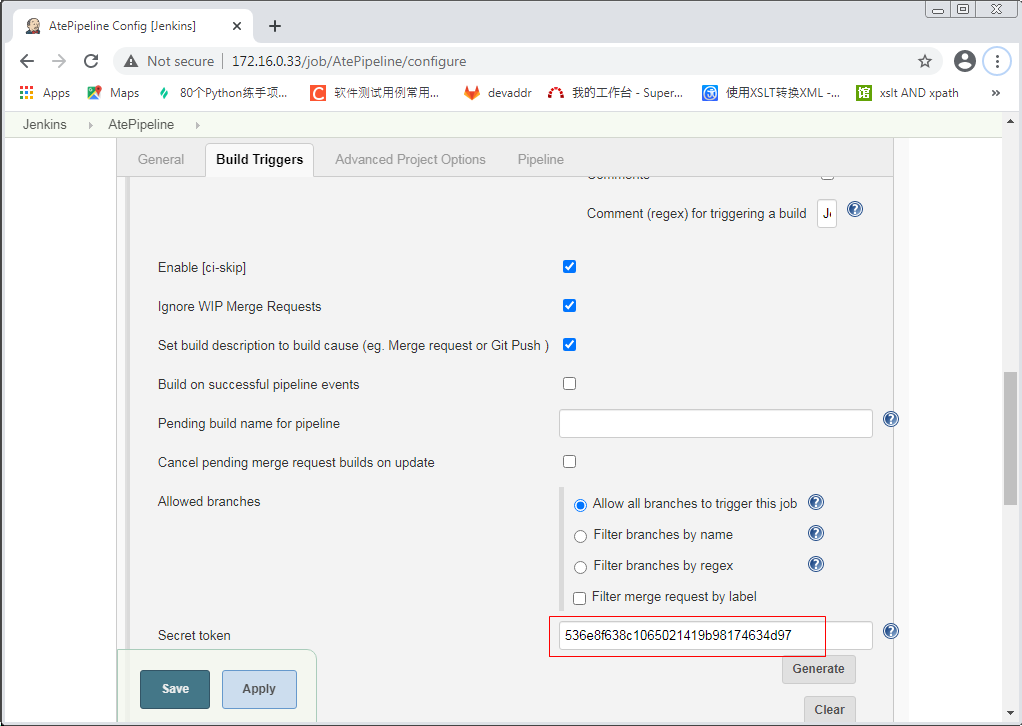
4. 任务配置

<https://blog.csdn.net/duyusean/article/details/85397241>

<https://www.jenkins.io/zh/doc/book/pipeline/syntax/>

Git触发配置





AtePipeLine任务配置脚本注释见《AtePipeLine配置脚本.txt》

从配置脚本可以看出，所有步骤都是调用的02 ci\04 common\common\_script.bat脚本，该脚本根据参数执行对应的操作：

第一个参数: 可以为compile, static, test, post; 分别代表编译，静态解析，测试和后处理, 后处理一般是日志上传

第二个参数: 对于compile, static, test, 第二个参数为firmware或software; 对于post, 第二个参数为'compile', 'static', 'test'

第三个参数: 如果第一个参数为post, 才会又第三个参数firmware或software

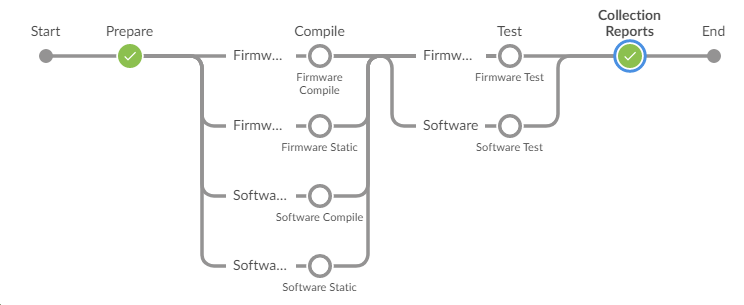
调用举例：

call common\_script.bat compile firmware //表示编译固件

call common\_script.bat test firmware //表示测试固件

call common\_script.bat post test firmware //表示上传固件测试日志

5. AtePipeline执行流程



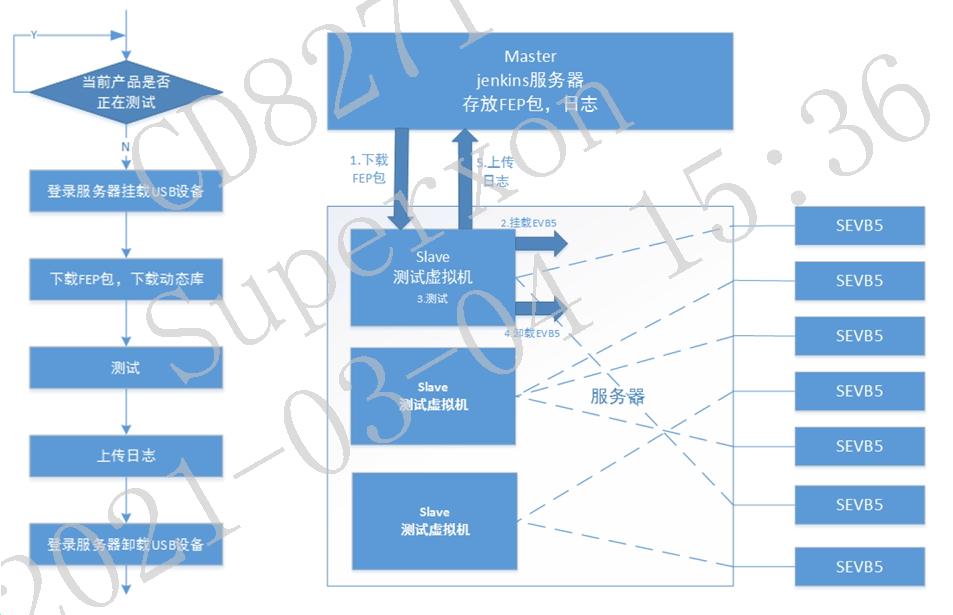
**固件自动化测试**

1. **测试环境**
2. 服务器IP：172.20.2.189, 登录账号: root, 登录密码: !Ate123, 网页客户端地址: <https://172.20.2.189/>

依赖库的Python库：robotframework, xlrd, lxml, openpyxl, paramiko, pywin32, pyvim, pyVmomi

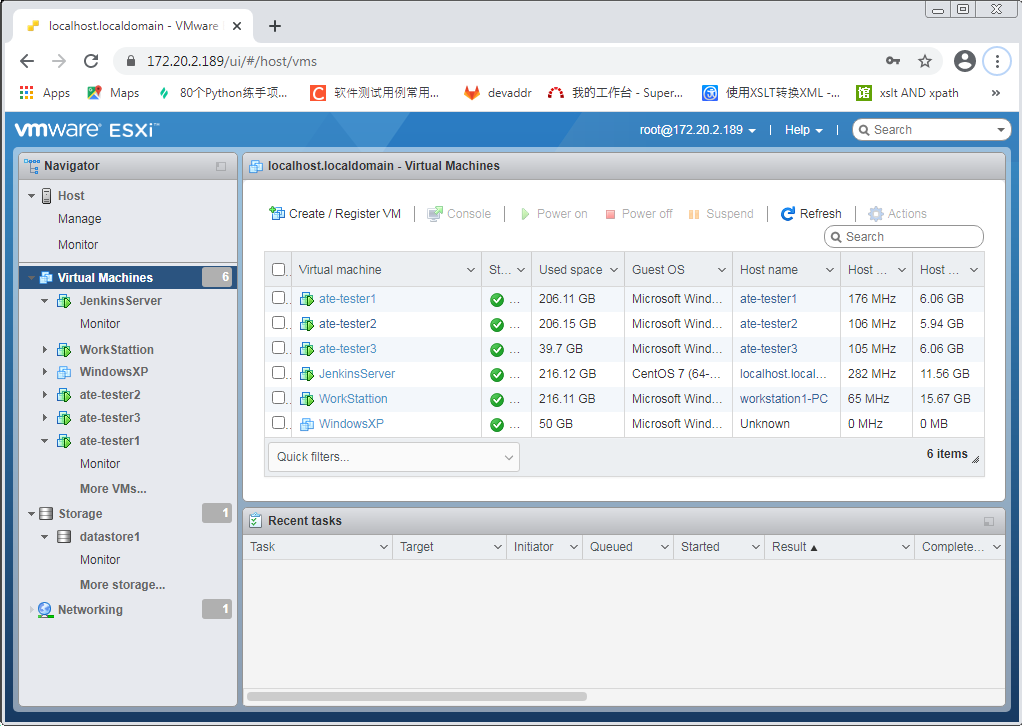
安装命令：pip install robotframework

1. 测试流程和环境框图:

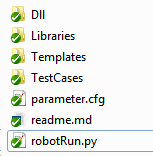


1. 测试电脑管理

用浏览器登录服务器可以对虚拟机进行管理，包括虚拟机创建、删除、重启以及修改配置等



1. **测试脚本**
2. 新固件增加测试脚本
3. 将测试板安装到机架上
4. USB线接入任意HUB口
5. 测试脚本上传到工程的02 ci\02 firmware\hlt目录下
6. 测试脚本文件名及配置信息介绍：



Dll：相关的动态库

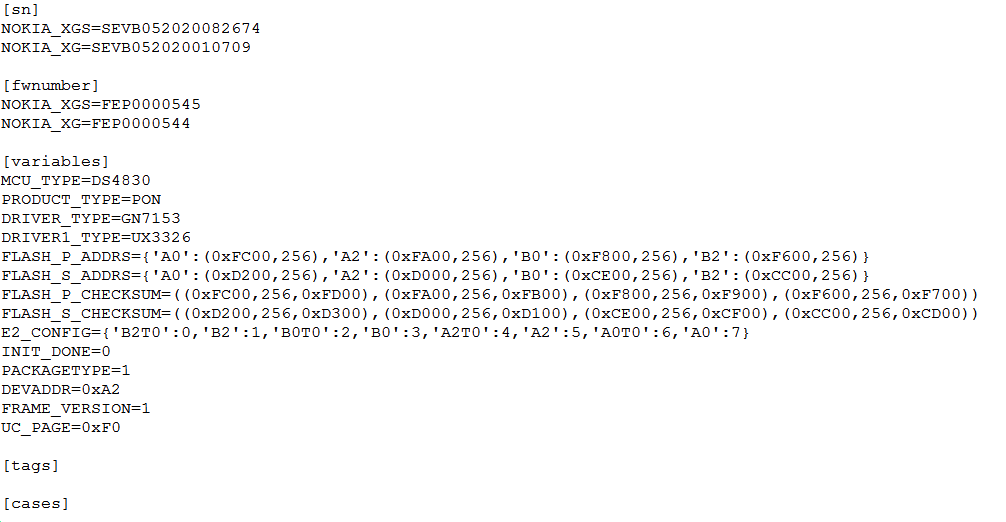
Libraries：底层的测试函数库

Templates：Excel模板，包括EED和TRT报告

TestCases：所有测试用例及用例需要的参数

robotRun.py：测试入口，如果不带参数，会测试所有固件，否则测试单个固件

Parameter.cfg：配置文件



**[sn]**配置前面接入服务器的SEVB5板SN，key一般为客户加方案，工程下每个固件一个配置，如果多个固件使用同一个测试板，就配置相同的SN

**[fwnumber]**固件编号，key必须跟[sn]中配置的key一一对应

**[variables]**全局变量

MCU\_TYPE: MCU类型，DS4830，F320等

PRODUCT\_TYPE: 产品类型，PON或者DCOM

DRIVER\_TYPE: Driver类型

FLASH\_P\_ADDRS: 主Flash地址

FLASH\_S\_ADDRS: 备Flash地址

FLASH\_P\_CHECKSUM: 主Flash Checksum

FLASH\_S\_CHECKSUM: 备Flash Checksum

E2\_CONFIG: E2读写权限配置

INIT\_DONE: 初始化完成标志

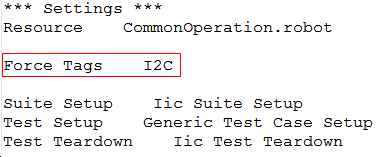
PACKAGETYPE: 封装类型

DEVADDR: SuperMaster设备地址

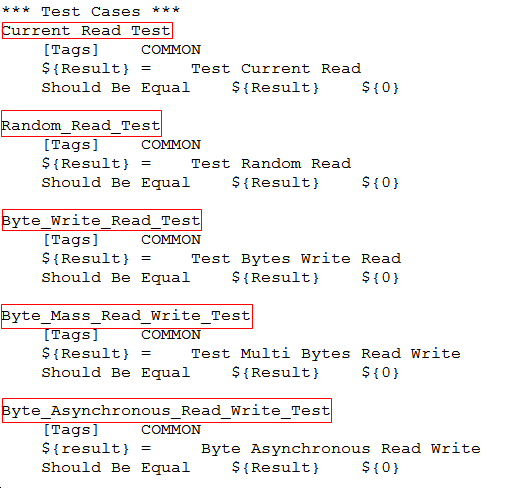
FRAME\_VERSION: 固件架构版本

US\_PAGE: Supermaster table页

**[tags]**如果配置了对应的tag, 则只跑tag下的case，比如配置了I2C这个tag，那么就只跑I2C相关的用例，否则跑所有case，tag可以在.robot文件的Force Tags中查询到，可以配置多个tag



**[cases]**用例名, 如果配置了则跑对应的用例，否则跑所有用例，可以配置多个用例



**时序板相关测试**

依赖的Python库：pyserial, robotframework

时序板相关的文档为《时序测试板-固件设计思路说明20190304.docx》

和《时序测试板-通信协议说明20190304.docx》

1. **时序板命令测试**

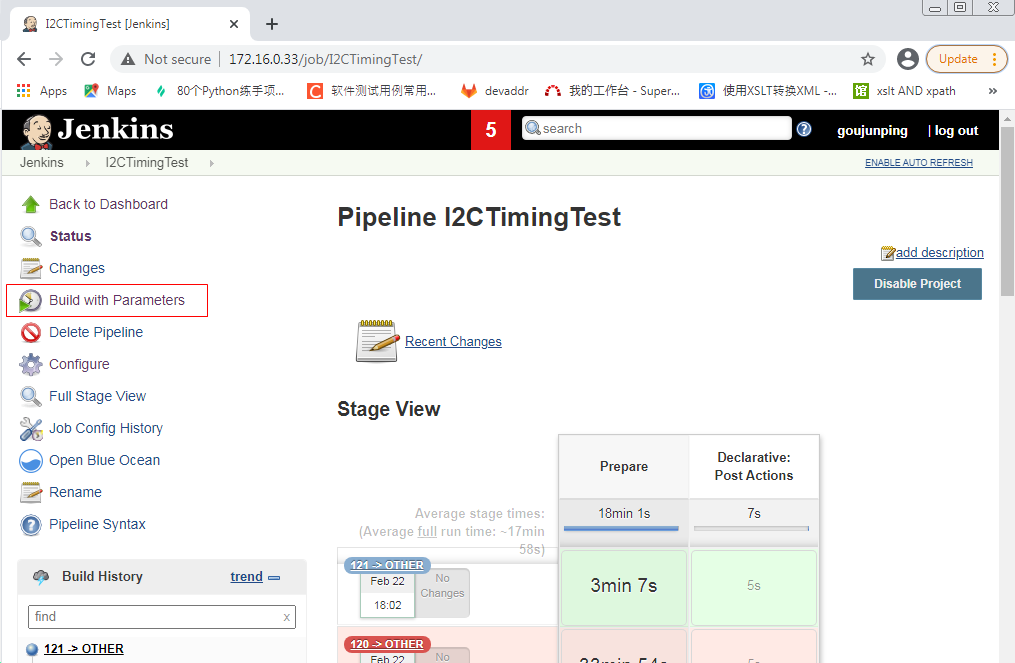
该脚本主要测试时序板的命令

1. 脚本路径：evaluation\_board\02 ci\hlt\SEVB0040
2. 执行方式：在cmd中，进入evaluation\_board\02 ci\hlt\SEVB0040目录下，执行命令robot Sevb40Cases.robot文件；其中时序板的SN配置在Sevb40Cases.robot文件中
3. **异常时序测试**
4. 脚本路径：tool\_middleware\04 src\04 test\AbnormalTimingTest
5. 执行方式：在cmd中，进入tool\_middleware\04 src\04 test\AbnormalTimingTest目录下，执行命令robot AbnormalTiming.robot文件；其中时序板的SN配置在AbnormalTiming.robot文件中
6. **I2C最大速率测试**
7. 脚本路径：tool\_middleware\04 src\04 test\I2CRateTest\I2CRateReliabilityTest.py
8. 需要配置环境变量：CUSTOMERTYPE客户类型，可以为ZTE、NOKIA以及其他的；INTERVAL每个速率间隔时间，单位秒；SEVB40SN时序板的SN号
9. 执行方式：在cmd中，进入tool\_middleware\04 src\04 test\I2CRateTest目录下，执行命令Python I2CRateReliabilityTest.py文件

**四. Jenkins中I2CTimingTest任务执行异常时序和I2C最大速率测试**

1. 将对应的子板和模块插入机架时序板中

2. 点击Jenkins中任务页面中的Build with Parameters



3. 参数介绍

TESTTYPE: 测试类型，ALL包括I2C异常时序和IIC速率测试；IICRATE只测试IIC速率；TIMING只测试异常时序

CUSTOMERTYPE: 客户类型，除了ZTE和NOKIA，其他任何产品都选OTHER

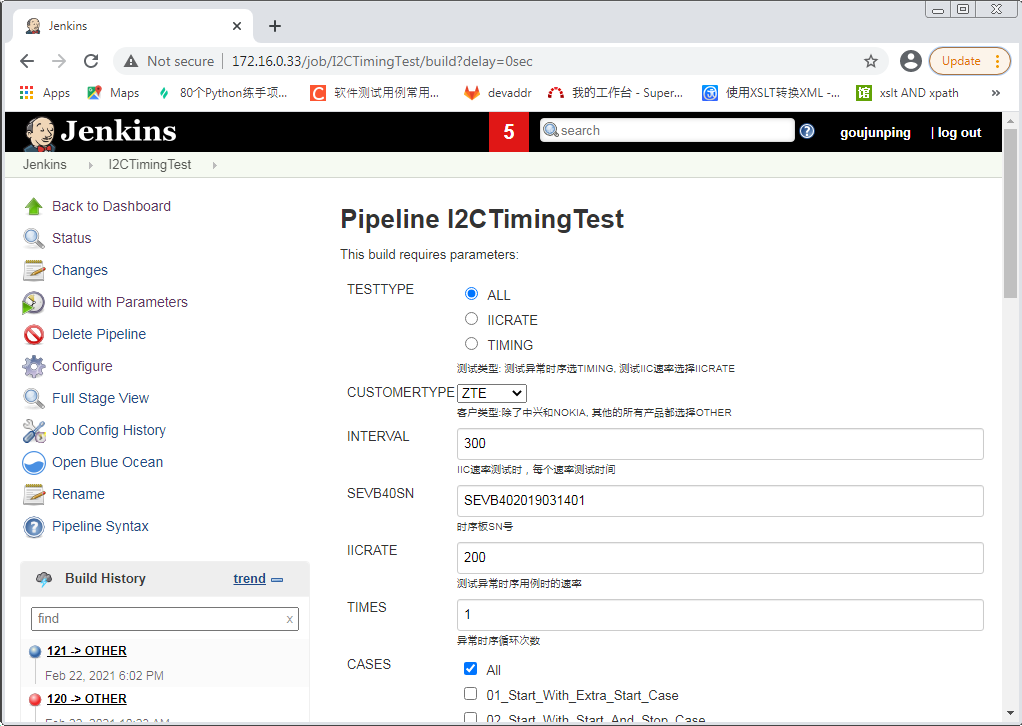
INTERVAL: IIC速率测试时，每个速率测试的时间，默认为300秒

SEVB40SN：时序板SN，如果不更换时序板，则使用默认值即可

IICRATE：异常时序测试时使用的IIC速率

TIMES：测试异常时序循环次数

CASES：异常时序用例选择



**代码静态解析**

脚本路径：02 ci\02 firmware\lint

**一．Cloc代码行数统计**

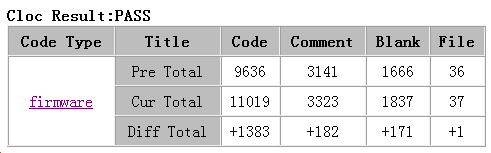
1. 软件cloc-1.64.exe位置在C:\Windows\System32
2. 执行命令在sw-cloc.bat脚本中
3. 统计结果：

Code: 代码行数

Comment：注释行数

Blank：空行行数

File：文件个数



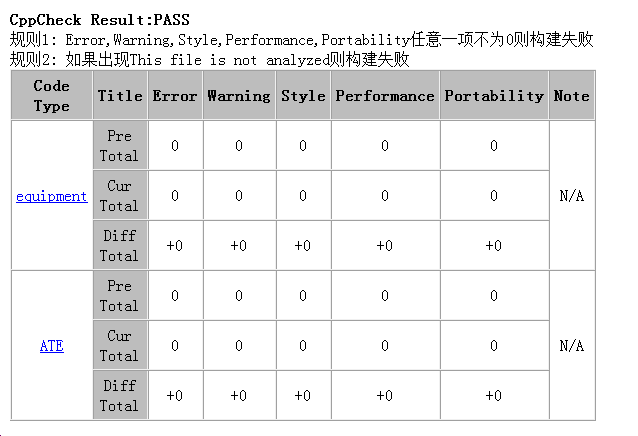
**二．CppCheck**

1. 软件cppcheck.exe 在每个工程的05 tools\04 common\Cppcheck路径下

2. 依赖Python库：Pygments

3. 执行命令在sw-cppcheck.bat脚本中

4. 统计结果

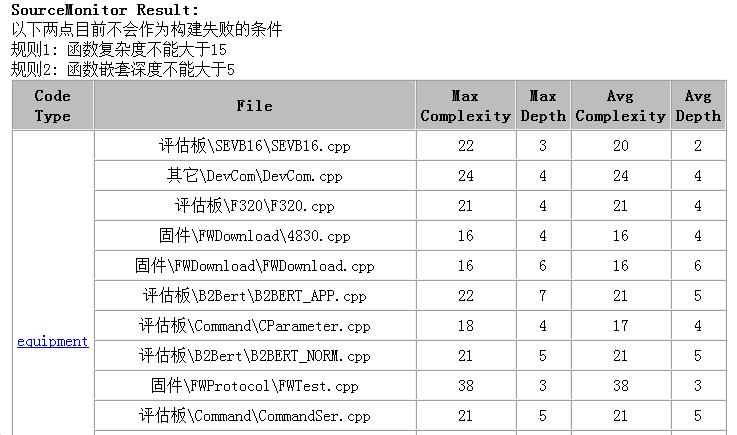


**三．SourceMonitor代码复杂度检查**

1. 软件SourceMonitor.exe位置C:\Program Files (x86)\SourceMonitor目录下，安装包SMSetupV3-5-14.exe

2. 执行命令在sw-sourcemonitor.bat脚本中

3. 统计结果

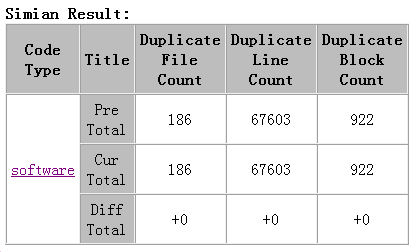


**四．Simian代码重复度检查**

1. 软件simian-2.5.10.exe位置C:\simian-2.5.10\bin，安装包：simian-2.5.10.tar.gz

2. 执行命令在sw-duplication.batt脚本中

3. 统计结果



**通用软件出包(software\_middleware)**

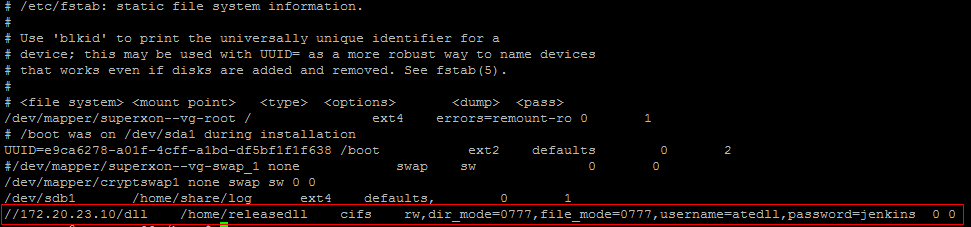
**一．执行流程**



备注1：在编译通用软件的时候，需要拷贝软件依赖的动态库相关文件，该功能在software\_compile.py脚本中实现：如果是出正式包，动态库编译时直接将编译结果拷贝到本地的D:\Jenkins\workspace\software目录下，软件编译是直接从该目录取；如果是非正式包，则从Jenkins服务器的/home/share/dll/master目录下取

备注2：编译结束后，无论成功还是失败，都会将所有编译结果拷贝到Jenkins服务器的/var/lib/jenkins/userContent目录下，该功能在file\_transfer.py脚本中实现

备注3：当所有步骤执行成功后(任意步骤失败则无需拷贝)，需要把动态库相关的文件拷贝到网盘中，以便产线在运行软件的时候能够从网盘取用，该过程在collection\_log.sh脚本中实现：如果是release分支，就拷贝到/home/releasedll/目录下，其他分支拷贝到/home/share/目录下；整个拷贝过程都在master服务器上进行，/home/releasedll/目录是网盘172.20.23.10挂载到master服务器上的，挂载配置文件为/etc/fstab

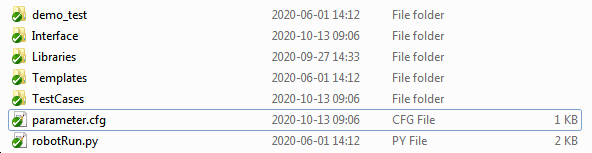


172.20.23.10网盘服务器是何川维护的

**二．动态库测试**

1. 脚本路径：software\_middleware\02 ci\03 software\hlt

2. 脚本功能说明：



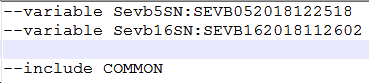
Interface: 动态库接口函数封装

Libraries: 动态库测试用例实现

Templates: 测试用例表格

TestCases: 测试用例

Parameter.cfg: 配置文件，分别为SEVB5板的SN，SEVB16的SN，以及tag



robotRun.py: 测试入口，不带参数

**HW数据回传工具**

1. 脚本路径tool\_middleware\04 src\03 software\生产相关工具\HW\_DataBack
2. 执行流程



**EED导入工具**

1. 脚本路径：tool\_middleware\04 src\03 software\EED相关工具\EEDModule

**FQC软件**

1. 脚本路径：software\_middleware\04 src\03 software\ATE\FQC
2. 为了兼容WindowsXP系统，该软件是用python34版本开发的，python34相关的安装软件在Pyqt\_Winxp文件夹下
3. 格式配置:



在没有任何参数的情况下点击查询会查询数据storagemanage\_config表中的所有配置信息；可以单独指定客户进行查询，也可以单独指定产品类型进行查询，或者两个条件都指定

没有指定任何参数的情况下查询：



指定客户的情况下查询：



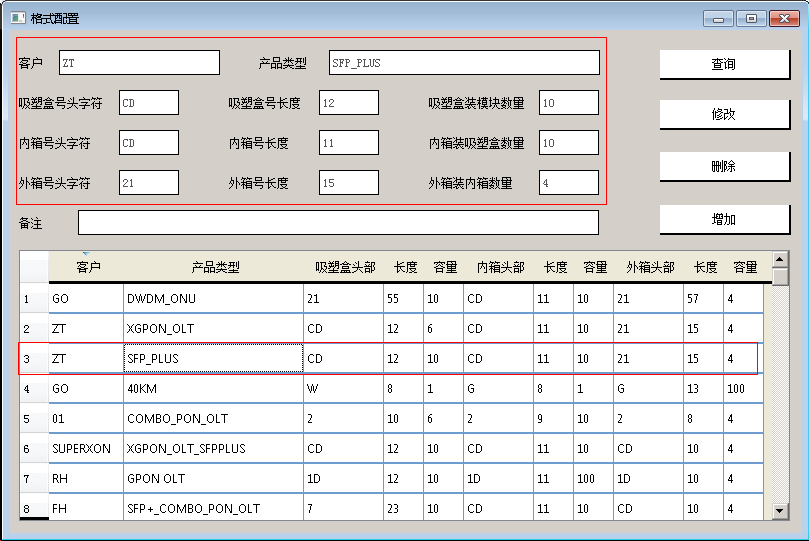
指定产品类型的情况下查询：



客户和产品类型都指定的情况下查询:



查询完成后点击下面表格中的任意一行，就会自动把该行的数据填到上面的输入框中，比如这里点击第3行；然后可以根据已有的信息进行增、删、改



注意：这里的查询、修改、删除和增加都是根据客户和产品类型这两个条件来确认一条数据的

产品类型是从数据库autodt\_spec\_tracking表的PN\_TYPE字段获取的

**OSA报告导入软件**

1. 脚本路径: tool\_middleware\04 src\03 software\生产相关工具\OSAImport
2. 要求：
3. 账号必须有autodt\_process\_log表和optics\_data表的写权限
4. Excel表头必须为”Superxon OSA Test Report”, 且一个Excel只能有一个这样的Sheet
5. 表头信息最好填写完整, 表头的总数量必须跟下面模块SN的数量相同
6. 模块SN不能有重复
7. 模块的数据目前只要求VBR必须有

**注意：**

1. 大部分的python软件都是使用的python3.4.4版本打包的, 以保证软件能在WindowsXP系统上运行, python3.4.4对应的PyQt以及其他库都在Pyqt\_Winxp目录下; 打包方式为pyinstaller -F main.py, 这里的main.py泛指软件的主文件;
2. 只有HW回传数据的软件是用python3.6.8打包的
3. 部分老的Win7电脑如果允许不起来python程序，请安装Windows6.1-KB2999226-x64.msu补丁后再试