

# Netty 环境配置

请务必阅读此章节，有疑问请在章节下留言。

源码地址：<https://github.com/lightningMan/flash-netty>

本小节介绍一下使用本小册的环境搭建，本小册假设读者本机已经有了 java 编程需要的环境。如果读者本机已经安装过 Maven, Git, IntelliJ IDEA 环境，建议直接看文末如何使用本小册代码。

## Maven

Maven 是一个基于对象模型来管理项目构建的项目管理工具，通过配置文件 pom.xml 来配置 jar 包，相对于传统拷贝 jar 包的方式，管理依赖更为方便，如果你的本地没有安装过 Maven，下面的指导将带你一起安装

### 1. 下载

首先，到[下载页面 \(http://maven.apache.org/download.cgi\)](http://maven.apache.org/download.cgi)下载 Maven，由于 Maven 也是使用 java 来编写，所以不同操作系统下载的 maven zip 包是一样的，这里我们选择最新的版本，apache-maven-版本号-src.zip，下载到本地之后解压，接下来我们来看下不同的操作系统配置

### 2. 配置和验证

## windows

1. 假定我们将文件夹解压到 D:\maven，该目录下有 bin、lib 等目录。
2. 通过 我的电脑->属性->高级系统设置->环境变量->系统变量->新建 新建一个环境变量，变量名为 M2\_HOME，然后值为 D:\maven。
3. 找到变量名字为 Path 的环境变量，在值尾部加入：;%M2\_HOME%\bin;，这里需要注意前面的分号。

最后，我们打开 dos 窗口，然后输入 mvn -v，如果出来版本号相关的信息，那么说明我们的 Maven 已经安装成功了。

## linux & mac

假定我们将文件夹解压到 /usr/local/maven，该目录下有 bin、lib 等目录，接下来，我们和 windows 系统一样，需要配置环境变量，我们打开 /etc/profile 文件，然后在尾部加入以下两行

```
export MAVEN_HOME=/usr/local/maven
PATH=$JAVA_HOME/bin:$MAVEN_HOME/bin:$PATH
```

最后，我们在命令行执行 source /etc/profile 让配置生效，最后，我们通过 mvn -v 命令来验证是否生效，如果出来版本号相关的信息，那么说明我们的 Maven 已经安装成功了。

## Git

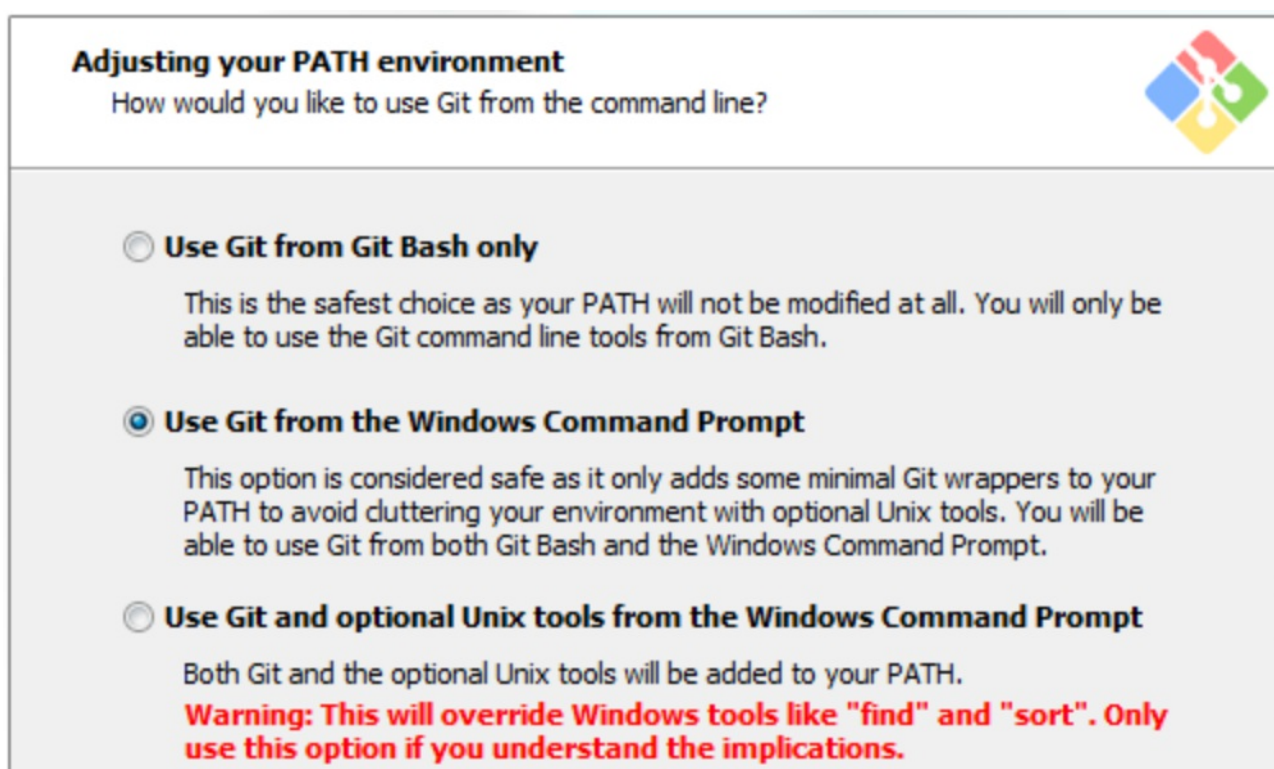
Git 是一个版本管理工具，本小册代码使用 git 来做版本控制，每个章节的代码为一个 git 分支，方便读者循序渐进地学习，接下来我们看一下如何安装配置 git

## 1. 下载安装

windows

1.首先我们去 <https://gitforwindows.org/> (<https://gitforwindows.org/>) 下载最新版本的 git windows 版本。

2.下载完成之后，双击 exe 文件，我们只需要一直选择下一步，安装即可，其中有一步需要注意一下



该步骤为调整环境变量，我们这里选择中间的一项，然后我们继续选择 next 直到安装成功。

3.安装成功之后，我们在任意目录右键，选择 Git Bash Here 这一项，输入 git，如果出来提示，表明安装成功。

mac && linux

1. 如果你使用的是 debian 或 ubuntu，那么直接一条命令 `sudo apt-get install git` 即可完成安装，如果是 CentOS 版本，在命令行执行 `yum install -y git` 即可完成安装
2. mac 系统自带 git，不过默认没有安装，你需要运行 xcode，然后选择菜单 "xcode" -> "Preferences"，选择 "Downloads" 这个 tab 页面，然后再选择 "Command Line Tools"，点击 "Install" 即可完成安装。

## 2. 配置

最后，我们通过在命令行依次输入以下命令来配置你的名字和邮箱，这样在提交代码的时候就能知道作者的信息

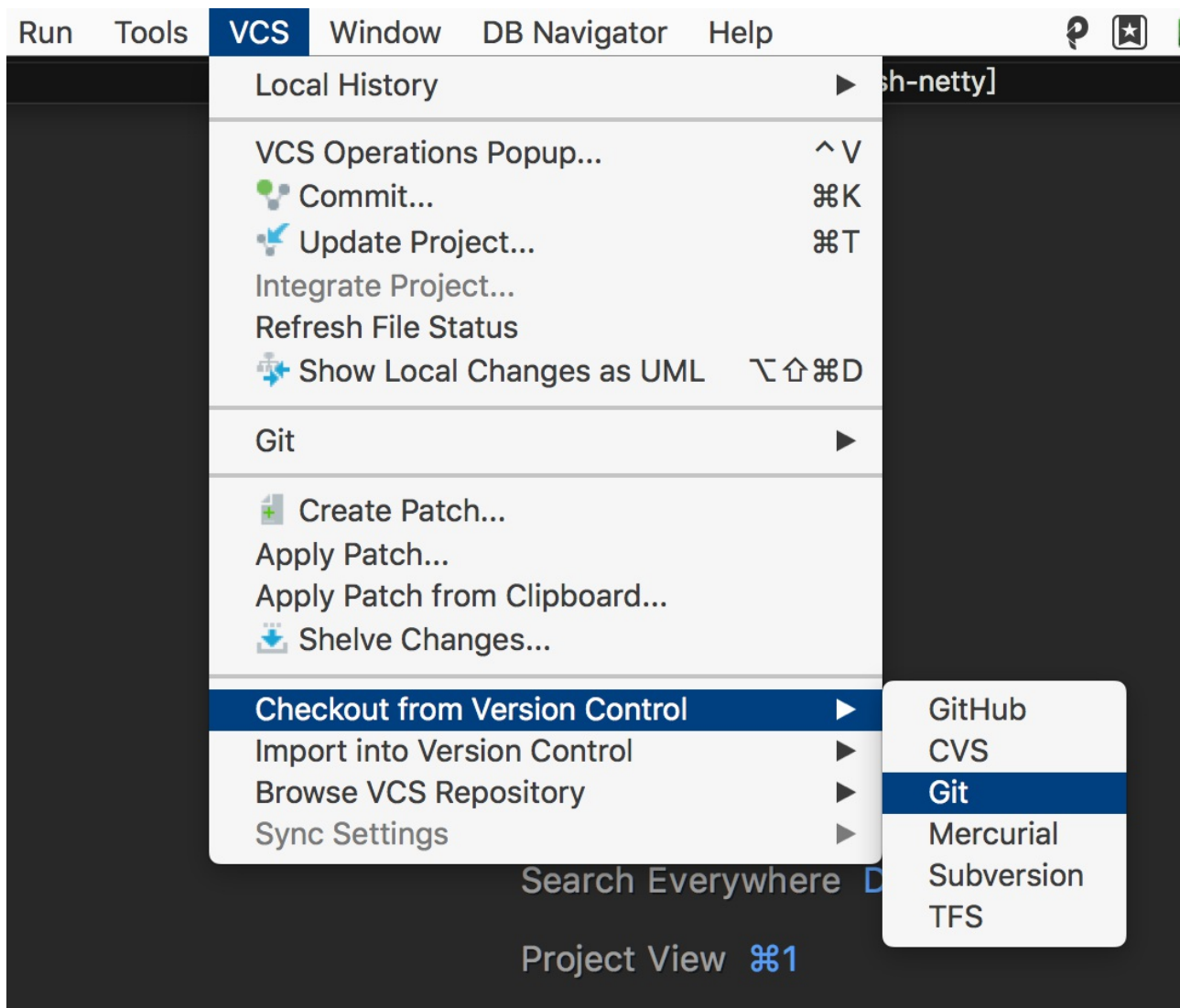
```
git config --global user.name "Your Name"  
git config --global user.email  
"email@example.com"
```

## IntelliJ IDEA

本小册使用 IntelliJ IDEA 作为集成开发环境，当然，如果你非常熟悉 eclipse，也可以使用 eclipse，对于想入门学习 IntelliJ IDEA 的同学，[这里 \(https://www.imooc.com/learn/924\)](https://www.imooc.com/learn/924) 的有我之前录制的的一个免费视频奉献给大家

详细的安装过程和介绍，该视频里面均有介绍，接下来我们来看一下，如何使用本小册的代码。

首先，我们通过如下步骤将代码仓库导入到本地



以上代码仓库为: <https://github.com/lightningMan/flash-netty.git> (<https://github.com/lightningMan/flash-netty.git>)

代码克隆到本地之后，在 IntelliJ IDEA 右下角切换相应的分支，即可找到每一小节对应的完整代码。

## Git Branches

+ New Branch

Checkout Tag or Revision...

## Local Branches

📁 构建客户端与服务端pipeline ▶

★ master ▶

origin/master ▶

pipeline与channelHandler ▶

实现客户端与服务端收发消息 ▶

实现客户端登录 ▶

客户端与服务端双向通信 ▶ origin/客户端与服务端双向通信 ▶

客户端与服务端通信协议编解码 ▶

数据传输载体ByteBuf ▶

## Remote Branches

★ origin/master ▶

origin/pipeline与channelHandler ▶

Checkout as new local branch

☆ origin/实现客户端与服务端收发消息 ▶

Compare

origin/实现客户端登录 ▶

Rebase onto

origin/客户端与服务端双向通信 ▶

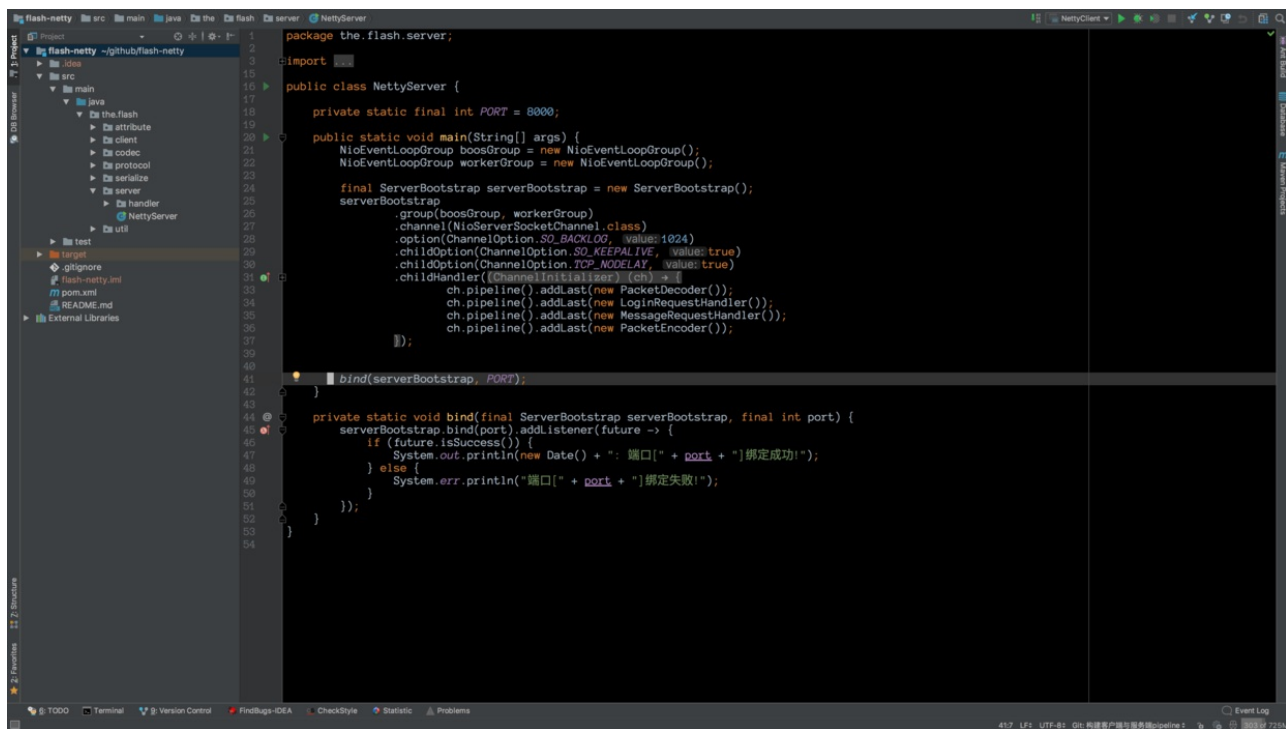
Merge

origin/客户端与服务端通信协议编解码 ▶

Delete

origin/客户端启动流程 ▶

origin/数据传输载体ByteBuf ▶



最后，由于代码里面，我使用了 [lombok](https://www.projectlombok.org/) (<https://www.projectlombok.org/>) 进行自动生成 getter/setter 以及构造函数，需要在 IntelliJ IDEA 中安装一下插件，否则代码会报红，具体安装可以参考以下步骤

先调出配置

然后

接着在弹出来的窗口输入 "lombok"

最后点击 "install" 安装之后重启 IntelliJ IDEA 即可。