

32 git服务器搭建

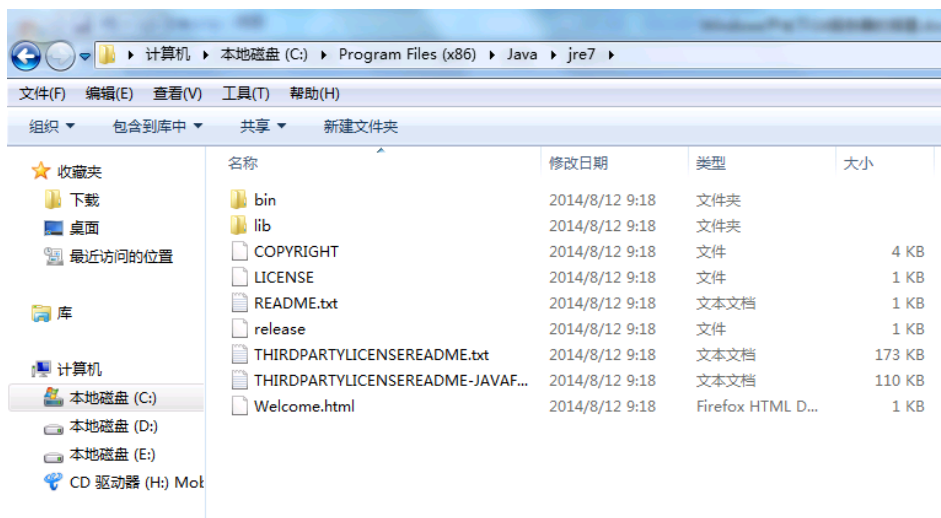
by-zhouzixing

Windows平台下Git服务器搭建

第一步：下载Java,下载地址：http://www.java.com/zh_CN/



第二步：安装Java.安装步骤不再详述。



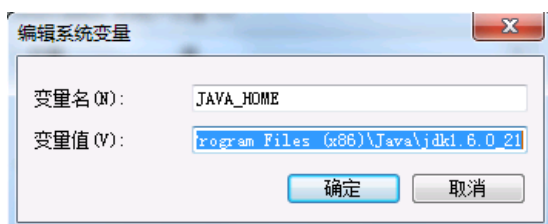
第三步：配置Java环境变量。

右键“计算机”=>“属性”=>“高级系统设置”=>“高级”=>“环境变量”=>“系统变量”。

1. 新建：

变量名：JAVA_HOME

变量值：D:\Program Files (x86)\Java\jdk1.6.0_21【具体要根据你自己的安装路径，我的是安装在D盘的】



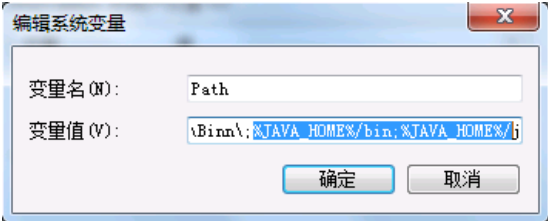
2. 新建：

变量名：CLASSPATH

变量值: .;%JAVA_HOME%/lib/dt.jar;%JAVA_HOME%/lib/tools.jar

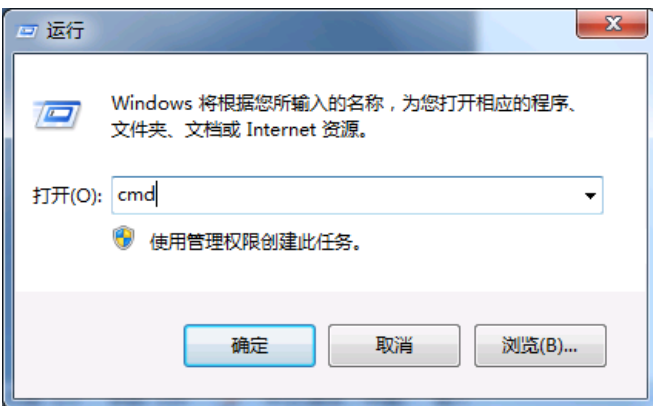


3. 添加: 找到PATH变量, 选择编辑。把%JAVA_HOME%/bin;%JAVA_HOME%/jre/bin添加到“变量值”的结尾处。



第四步: 验证Java是否安装成功。

打开命令窗口=> 输入CMD => 确定。



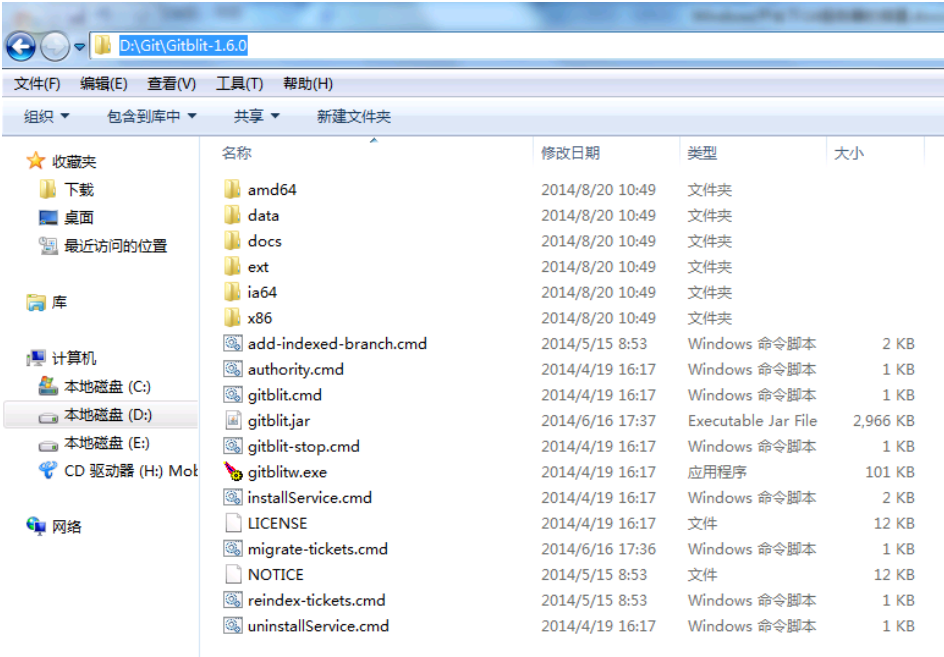
在命令窗口中, 输入java命令: javac,回车, 如果出现如下信息, 说明安装成功。



第五步: 下载Gitblit.下载地址: <http://www.gitblit.com/>



第六步：解压缩下载的压缩包即可，无需安装。



第七步：创建用于存储资料的文件夹。



第八步：配置gitblit.properties 文件。

1. 找到Git目录下的data文件下的gitblit.properties文件，“记事本”打开。

```
gitblit.properties - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

# Allow an authenticated user to create a destination repository on a push if
# the repository does not already exist.
#
# Administrator accounts can create a repository in any project.
# These repositories are created with the default access restriction and authorization
# control values. The pushing account is set as the owner.
#
# Non-administrator accounts with the CREATE role may create personal repositories.
# These repositories are created as VIEW restricted for NAMED users.
# The pushing account is set as the owner.
#
# SINCE 1.2.0
git.allowCreateOnPush = true

# Global setting to control anonymous pushes.
#
# This setting allows/rejects anonymous pushes at the level of the receive pack.
# This trumps all repository config settings. While anonymous pushes are convenient
# on your own box when you are a lone developer, they are not recommended for
# any multi-user installation where accountability is required. Since Gitblit
# tracks pushes and user accounts, allowing anonymous pushes compromises that
# information.
#
# SINCE 1.4.0
git.allowAnonymousPushes = false

# The default access restriction for new repositories.
# Valid values are NONE, PUSH, CLONE, VIEW
# NONE = anonymous view, clone, & push
# PUSH = anonymous view & clone and authenticated push
# CLONE = anonymous view, authenticated clone & push
# VIEW = authenticated view, clone, & push
#
# SINCE 1.0.0
git.defaultAccessRestriction = PUSH
```

2. 找到git.repositoriesFolder(资料库路径)，赋值为第七步创建好的文件目录。

```
gitblit.properties - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

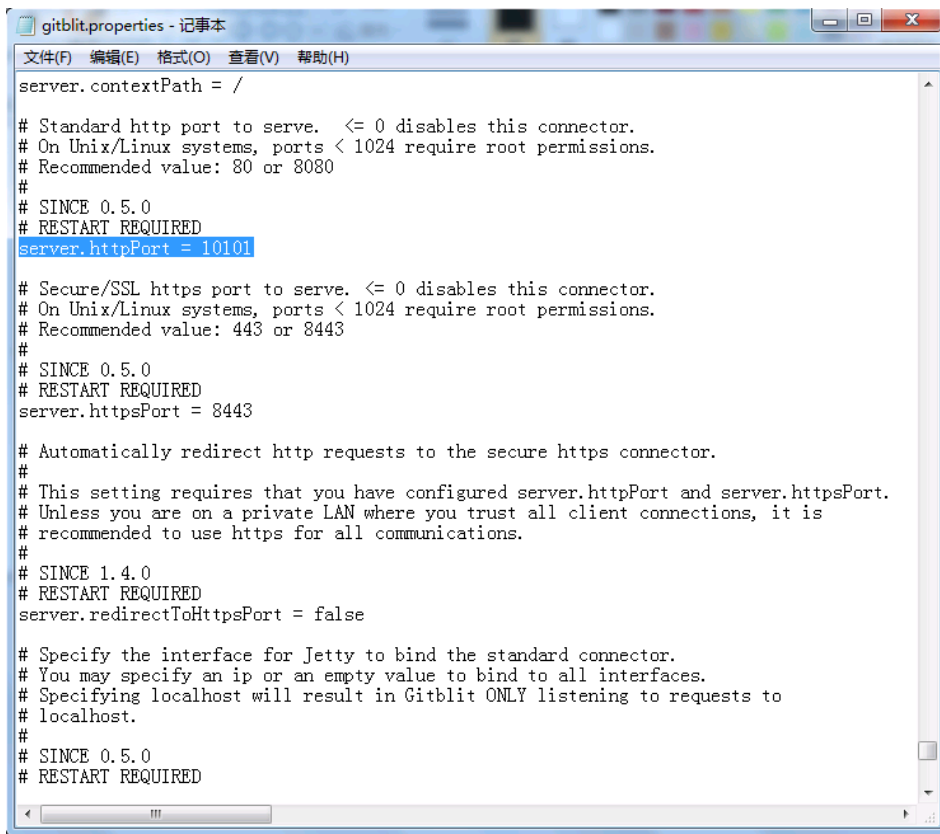
#
# e.g. "${baseFolder}/git" becomes "git", if --baseFolder is unspecified
#
# Git Servlet Settings
#
# Base folder for repositories.
# This folder may contain bare and non-bare repositories but Gitblit will only
# allow you to push to bare repositories.
# Use forward slashes even on Windows!!
# e.g. c:/gitrepos
#
# SINCE 0.5.0
# RESTART REQUIRED
# BASEFOLDER
git.repositoriesFolder = D:\Git\GitRepository

# Build the available repository list at startup and cache this list for reuse.
# This reduces disk io when presenting the repositories page, responding to rpcs,
# etc, but it means that Gitblit will not automatically identify repositories
# added or deleted by external tools.
#
# For this case you can use curl, wget, etc to issue an rpc request to clear the
# cache (e.g. https://localhost/rpc?req=CLEAR_REPOSITORY_CACHE)
#
# SINCE 1.1.0
git.cacheRepositoryList = true

# Search the repositories folder subfolders for other repositories.
# Repositories MAY NOT be nested (i.e. one repository within another)
# but they may be grouped together in subfolders.
# e.g. c:/gitrepos/libraries/mylibrary.git
#       c:/gitrepos/libraries/myotherlibrary.git
#
# SINCE 0.5.0
```

注意其中的D:\Git\GitRepository 中的“\”一定要用“/”。

3. 找到server.httpPort，设定http协议的端口号



```
gitblit.properties - 记事本
文件(F)  编辑(E)  格式(O)  查看(V)  帮助(H)

server.contextPath = /

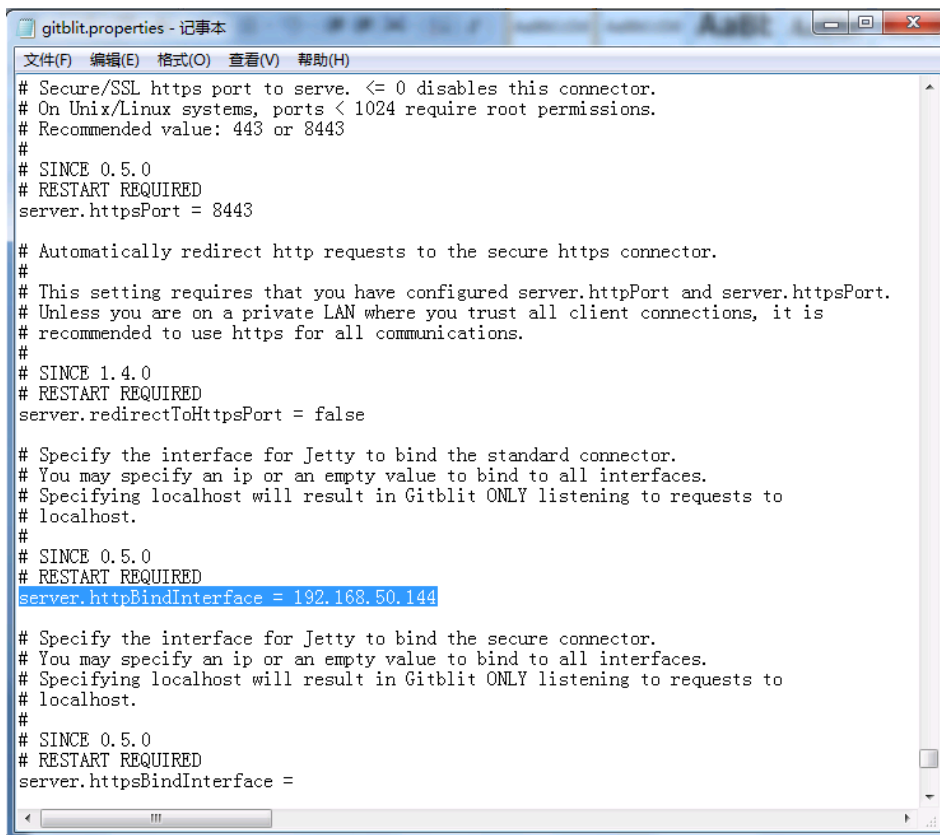
# Standard http port to serve. <= 0 disables this connector.
# On Unix/Linux systems, ports < 1024 require root permissions.
# Recommended value: 80 or 8080
#
# SINCE 0.5.0
# RESTART REQUIRED
server.httpPort = 10101

# Secure/SSL https port to serve. <= 0 disables this connector.
# On Unix/Linux systems, ports < 1024 require root permissions.
# Recommended value: 443 or 8443
#
# SINCE 0.5.0
# RESTART REQUIRED
server.httpsPort = 8443

# Automatically redirect http requests to the secure https connector.
#
# This setting requires that you have configured server.httpPort and server.httpsPort.
# Unless you are on a private LAN where you trust all client connections, it is
# recommended to use https for all communications.
#
# SINCE 1.4.0
# RESTART REQUIRED
server.redirectToHttpsPort = false

# Specify the interface for Jetty to bind the standard connector.
# You may specify an ip or an empty value to bind to all interfaces.
# Specifying localhost will result in Gitblit ONLY listening to requests to
# localhost.
#
# SINCE 0.5.0
# RESTART REQUIRED
```

4. 找到server.httpBindInterface, 设定服务器的IP地址。这里就设定你的服务器IP。



```
gitblit.properties - 记事本
文件(F)  编辑(E)  格式(O)  查看(V)  帮助(H)

# Secure/SSL https port to serve. <= 0 disables this connector.
# On Unix/Linux systems, ports < 1024 require root permissions.
# Recommended value: 443 or 8443
#
# SINCE 0.5.0
# RESTART REQUIRED
server.httpsPort = 8443

# Automatically redirect http requests to the secure https connector.
#
# This setting requires that you have configured server.httpPort and server.httpsPort.
# Unless you are on a private LAN where you trust all client connections, it is
# recommended to use https for all communications.
#
# SINCE 1.4.0
# RESTART REQUIRED
server.redirectToHttpsPort = false

# Specify the interface for Jetty to bind the standard connector.
# You may specify an ip or an empty value to bind to all interfaces.
# Specifying localhost will result in Gitblit ONLY listening to requests to
# localhost.
#
# SINCE 0.5.0
# RESTART REQUIRED
server.httpBindInterface = 192.168.50.144

# Specify the interface for Jetty to bind the secure connector.
# You may specify an ip or an empty value to bind to all interfaces.
# Specifying localhost will result in Gitblit ONLY listening to requests to
# localhost.
#
# SINCE 0.5.0
# RESTART REQUIRED
server.httpsBindInterface =
```

5. 找到server.httpsBindInterface, 设定为localhost

```
gitblit.properties - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

# You may specify an ip or an empty value to bind to all interfaces.
# Specifying localhost will result in Gitblit ONLY listening to requests to
# localhost.
#
# SINCE 0.5.0
# RESTART REQUIRED
server.httpBindInterface = 192.168.50.144

# Specify the interface for Jetty to bind the secure connector.
# You may specify an ip or an empty value to bind to all interfaces.
# Specifying localhost will result in Gitblit ONLY listening to requests to
# localhost.
#
# SINCE 0.5.0
# RESTART REQUIRED
server.httpsBindInterface = localhost

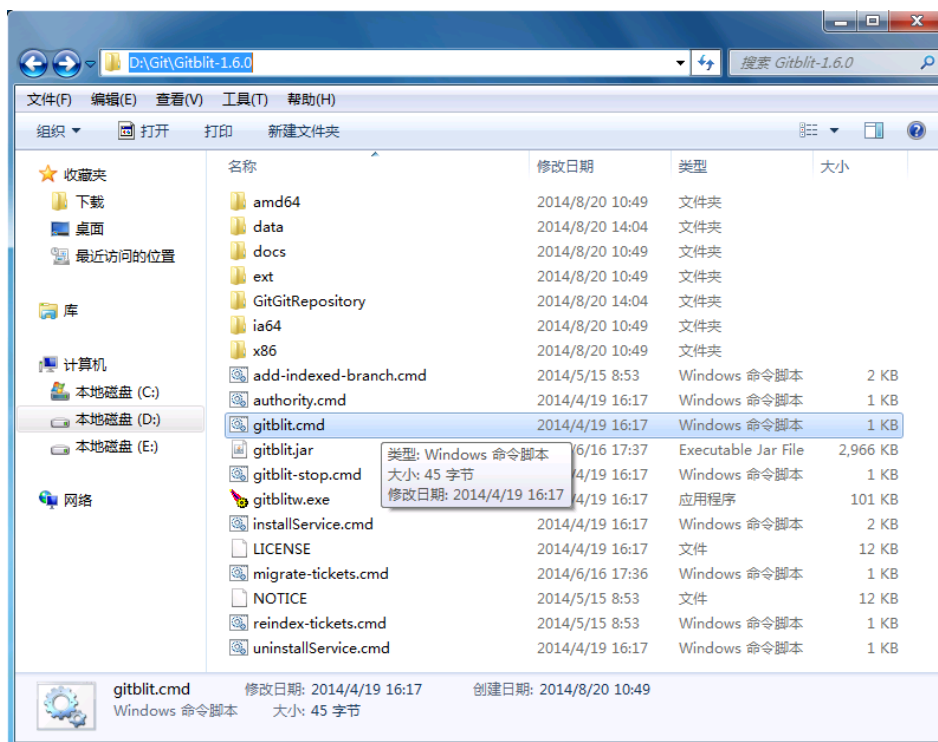
# Alias of certificate to use for https/SSL serving. If blank the first
# certificate found in the keystore will be used.
#
# SINCE 1.2.0
# RESTART REQUIRED
server.certificateAlias = localhost

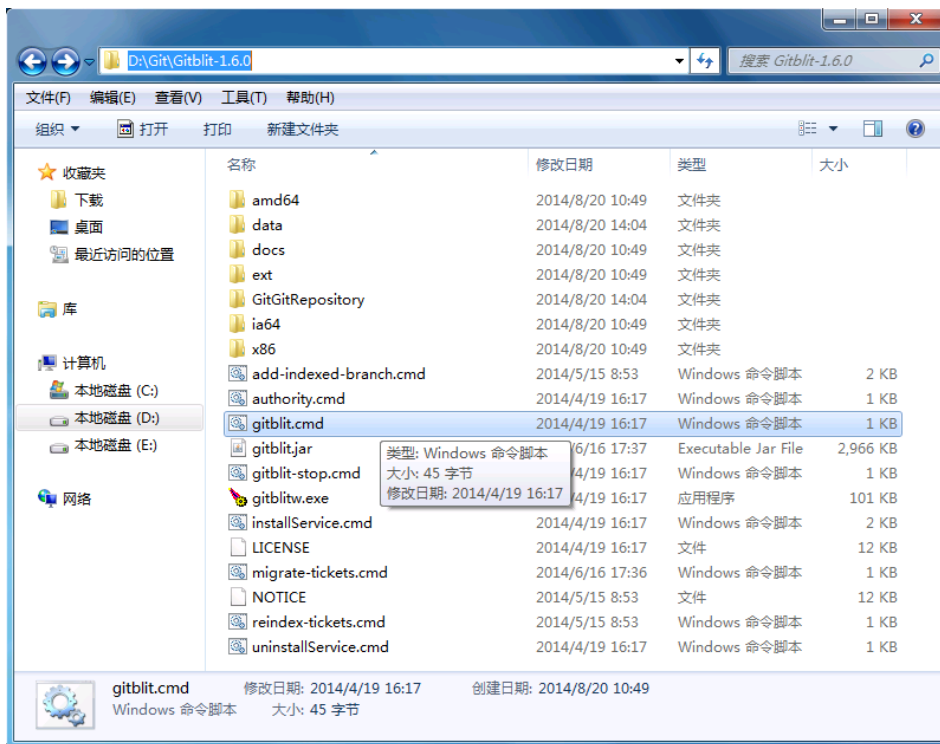
# Password for SSL keystore.
# Keystore password and certificate password must match.
# This is provided for convenience, its probably more secure to set this value
# using the --storePassword command line parameter.
#
# If you are using the official JRE or JDK from Oracle you may not have the
# JCE Unlimited Strength Jurisdiction Policy files bundled with your JVM. Because
# of this, your store/key password can not exceed 7 characters. If you require
# longer passwords you may need to install the JCE Unlimited Strength Jurisdiction
# Policy files from Oracle.
#
```

6. 保存，关闭文件。

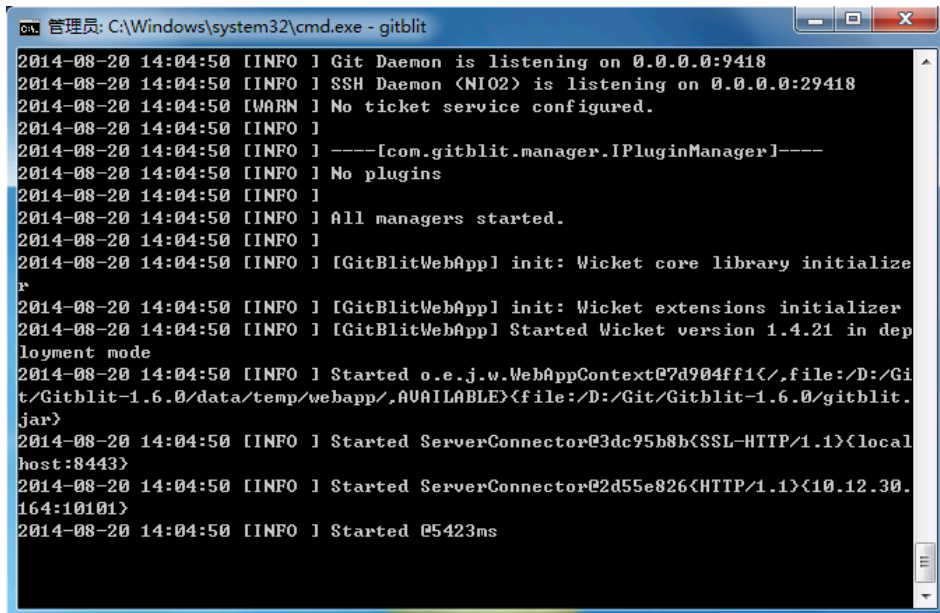
第九步：运行gitblit.cmd 批处理文件。

1. 找到bitblit目录中的gitblit.cmd文件，双击。

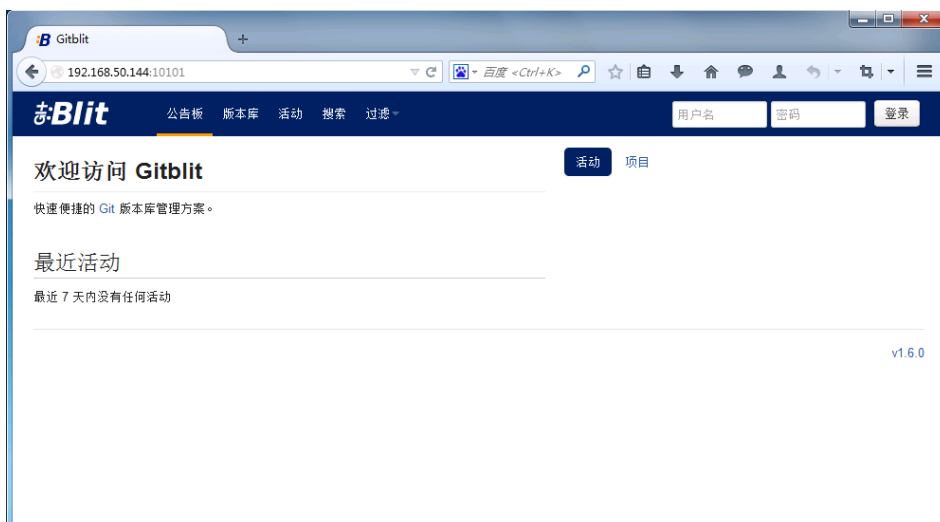




2. 运行结果如下，运行成功。

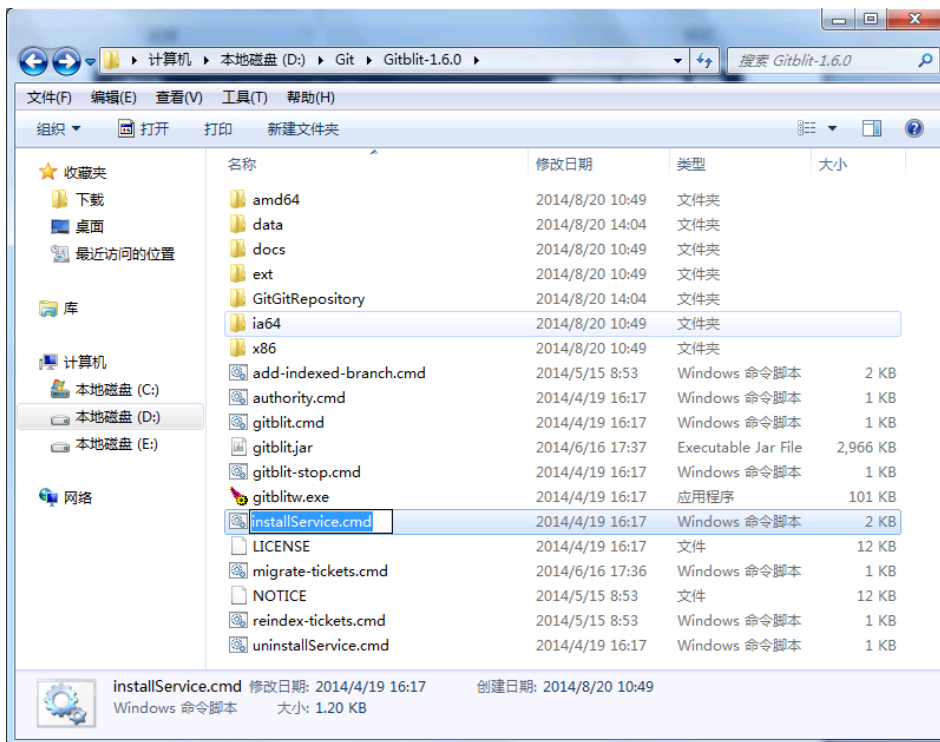


第十步：在浏览器中打开,现在就可以使用GitBlit了。

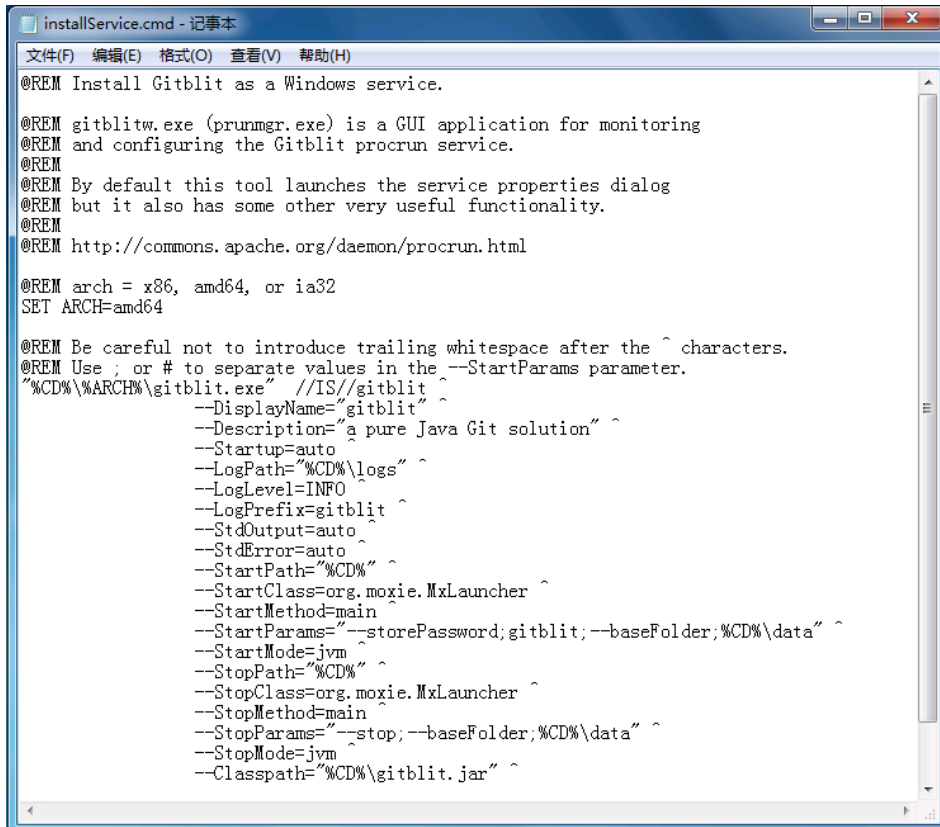


第十一步：设置以Windows Service方式启动GitBlit。

1. 在Gitblit目录下，找到installService.cmd文件。



2. 用“记事本”打开。



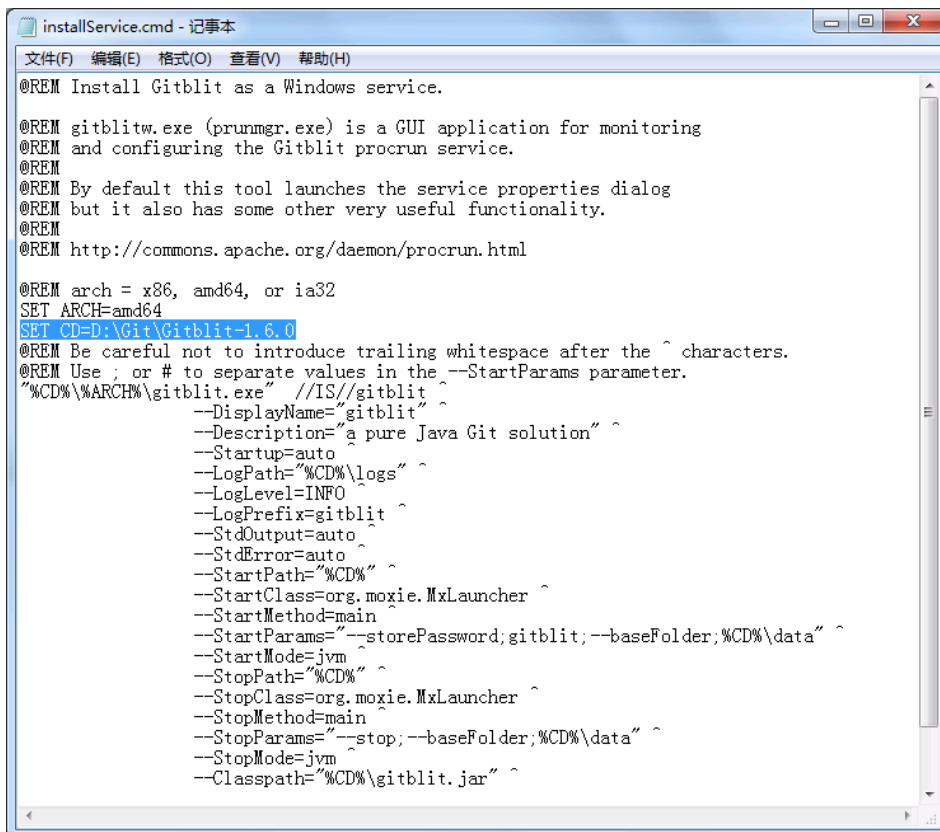
3. 修改 ARCH

32位系统: SET ARCH=x86

64位系统: SET ARCH=amd64

4. 添加 CD 为程序目录

SET CD=D:\Git\Gitblit-1.6.0(你的实际目录)



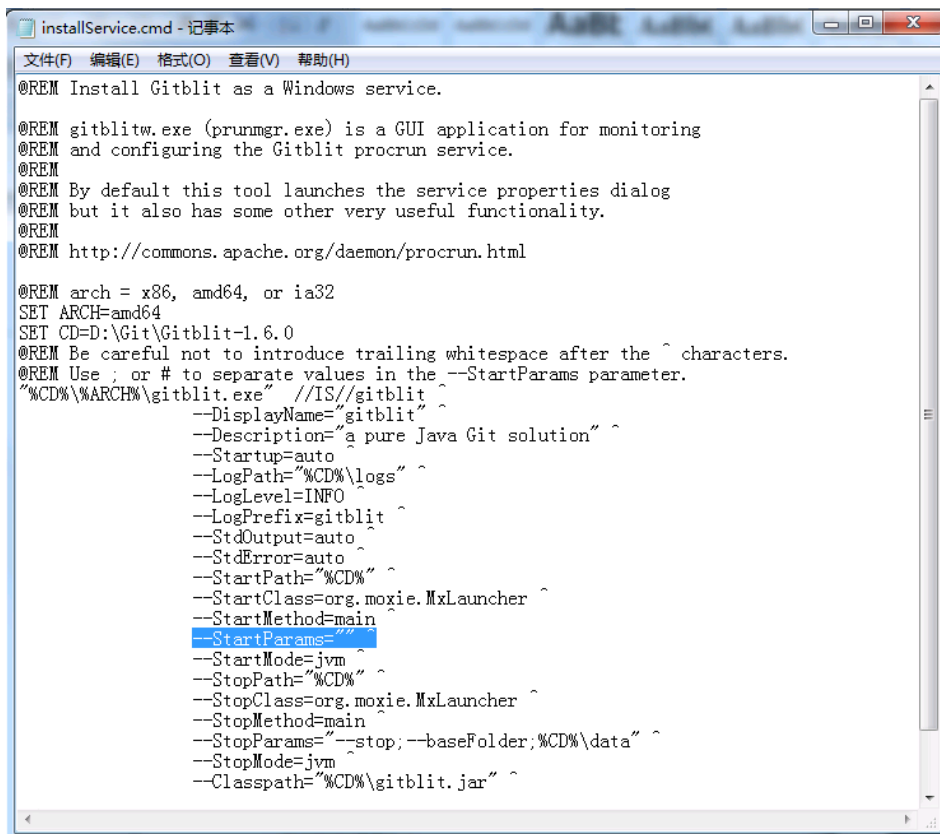
```
installService.cmd - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

@REM Install Gitblit as a Windows service.

@REM gitblitw.exe (prunmgr.exe) is a GUI application for monitoring
@REM and configuring the Gitblit procrun service.
@REM
@REM By default this tool launches the service properties dialog
@REM but it also has some other very useful functionality.
@REM
@REM http://commons.apache.org/daemon/procrun.html

@REM arch = x86, amd64, or ia32
SET ARCH=amd64
SET CD=D:\Git\Gitblit-1.6.0
@REM Be careful not to introduce trailing whitespace after the ^ characters.
@REM Use ; or # to separate values in the --StartParams parameter.
"%CD%\%ARCH%\gitblit.exe" //IS//gitblit ^
    --DisplayName="gitblit" ^
    --Description="a pure Java Git solution" ^
    --Startup=auto ^
    --LogPath="%CD%\logs" ^
    --LogLevel=INFO ^
    --LogPrefix=gitblit ^
    --StdOutput=auto ^
    --StdError=auto ^
    --StartPath="%CD%" ^
    --StartClass=org.moxie.MxLauncher ^
    --StartMethod=main ^
    --StartParams="--storePassword;gitblit;--baseFolder;%CD%\data" ^
    --StartMode=jvm ^
    --StopPath="%CD%" ^
    --StopClass=org.moxie.MxLauncher ^
    --StopMethod=main ^
    --StopParams="--stop;--baseFolder;%CD%\data" ^
    --StopMode=jvm ^
    --Classpath="%CD%\gitblit.jar" ^
```

5. 修改StartParams里的启动参数，给空就可以了。



```
installService.cmd - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

@REM Install Gitblit as a Windows service.

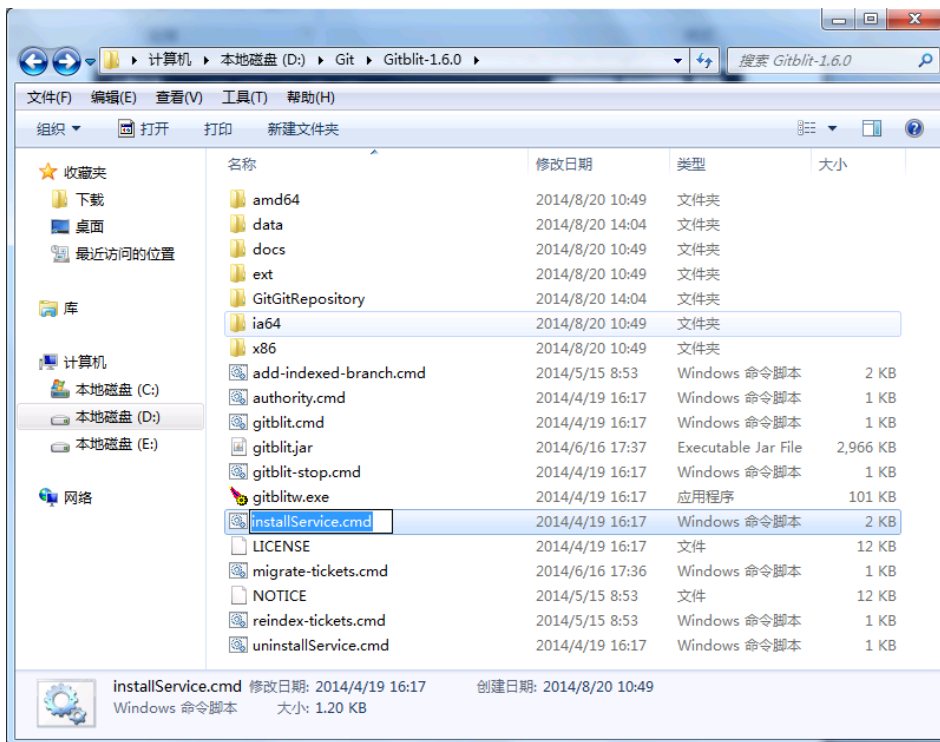
@REM gitblitw.exe (prunmgr.exe) is a GUI application for monitoring
@REM and configuring the Gitblit procrun service.
@REM
@REM By default this tool launches the service properties dialog
@REM but it also has some other very useful functionality.
@REM
@REM http://commons.apache.org/daemon/procrun.html

@REM arch = x86, amd64, or ia32
SET ARCH=amd64
SET CD=D:\Git\Gitblit-1.6.0
@REM Be careful not to introduce trailing whitespace after the ^ characters.
@REM Use ; or # to separate values in the --StartParams parameter.
"%CD%\%ARCH%\gitblit.exe" //IS//gitblit ^
    --DisplayName="gitblit" ^
    --Description="a pure Java Git solution" ^
    --Startup=auto ^
    --LogPath="%CD%\logs" ^
    --LogLevel=INFO ^
    --LogPrefix=gitblit ^
    --StdOutput=auto ^
    --StdError=auto ^
    --StartPath="%CD%" ^
    --StartClass=org.moxie.MxLauncher ^
    --StartMethod=main ^
    --StartParams="" ^
    --StartMode=jvm ^
    --StopPath="%CD%" ^
    --StopClass=org.moxie.MxLauncher ^
    --StopMethod=main ^
    --StopParams="--stop;--baseFolder;%CD%\data" ^
    --StopMode=jvm ^
    --Classpath="%CD%\gitblit.jar" ^
```

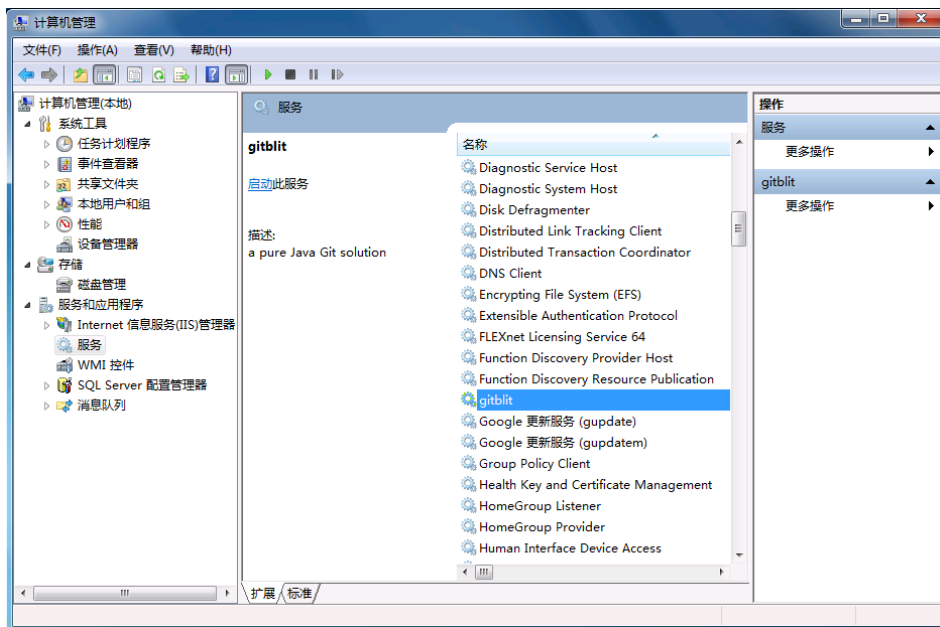
6. 保存，关闭文件。

第十二步：以Windows Service方式启动Gitblit。

1. 双击Gitblit目录下的installService.cmd文件(以管理员身份运行)。



2. 在服务的服务管理下，就能看到已经存在的gitblit服务了。



3. 平时使用时，保持这个服务是启动状态就可以了。

以上就是Windows平台下，配置GitV服务器的全部的全部内容。