

# BiliBot

---

bilibot是一个可以监控bilibiliUP直播以及动态的QQ机器人，并附有关键词回复等功能

## 二代目

---

在使用的酷Q框架关闭后，BiliBot二代目使用了另外一个框架——mirai

BiliBot二代目使用的有：

mirai：<https://github.com/mamoe/mirai>

mirai-http-api：<https://github.com/project-mirai/mirai-api-http>

node-mirai-sdk：<https://github.com/RedBeanN/node-mirai>

## 环境配置

---

### Java

---

mirai运行需要java环境，所以首先要安装java环境

#### linux之ubuntu安装java

进入ubuntu终端后在命令行输入 `sudo apt install openjdk-11-jdk`

```
ubuntu@VM-0-6-ubuntu:~$ sudo apt-get install openjdk-11-jdk
```

安装成功后输入 `java -version` 查看java版本，判断java是否安装成功

```
ubuntu@VM-0-6-ubuntu:~$ java -version
openjdk version "11.0.10" 2021-01-19
OpenJDK Runtime Environment (build 11.0.10+9-Ubuntu-0ubuntu1.18.04)
OpenJDK 64-Bit Server VM (build 11.0.10+9-Ubuntu-0ubuntu1.18.04, mixed mode, sha
ring)
```

### node

---

BiliBot二代目开发使用的框架为node的mirai-http-api插件，所以需要安装node.js

#### linux之ubuntu安装node

教程地址：<https://www.cnblogs.com/niuben/p/12938501.html>

1.进入将要安装node的文件夹，输入以下命令下载node安装包：

```
wget https://nodejs.org/dist/v14.15.5/node-v14.15.5-linux-x64.tar.xz
```

2.解压node安装包：

```
tar -xvf node-v14.15.5-linux-x64.tar.xz
```

3.配置软连接，全局都可以使用node

```
mv node-v14.15.5-linux-x64 node // 修改解压包名称

ln -s /安装node的文件夹/node /usr/bin/node --将node源文件映射到usr/bin下的node文件

ln -s /安装node的文件夹/node/bin/npm /usr/bin/npm
```

#### 4.配置node文件安装路径

```
mkdir node_global

mkdir node_cache

npm config set prefix "node_global"

npm config set cache "node_cache"
```

#### 5.查看node版本判断是否安装成功

```
node -v
```

## 运行

### 安装mirai

找到mirai文件夹，根据你的操作系统选择“mcl-installer-1.0.1-linux-amd64”或是“mcl-installer-1.0.1-windows-amd64.exe”，运行后再根据电脑的环境选择对应的版本

### window系统安装mcl

将“mcl-installer-1.0.1-windows-amd64.exe”放入向要安装mirai的文件夹中运行，再根据系统环境的选择对应的mirai安装包

```
iTXXTech MCL Installer 1.0.1 [OS: windows]
Licensed under GNU AGPLv3.
https://github.com/iTXXTech/mcl-installer

iTXXTech MCL will install to E:\biliBot\mirai\test

Java version <8-15, default: 11>: 11
JRE or JDK <1: JRE, 2: JDK, default: JRE>: 2
Binary Architecture <default: x64>:
Fetching file list for jdk version 11 on x64
Start Downloading: https://mirrors.tuna.tsinghua.edu.cn/AdoptOpenJDK/11/jdk/x64/
windows/OpenJDK11U-jdk_x64_windows_hotspot_11.0.10_9.zip
Downloading: 196126247/196126247
Extracting [17/590] jdk-11.0.10+9/bin/api-ms-win-core-processenvironment-l1-1-0.
Extracting [143/590] jdk-11.0.10+9/conf/security/policy/limited/default_local.po
Extracting [144/590] jdk-11.0.10+9/conf/security/policy/limited/default_US_expor
Extracting [145/590] jdk-11.0.10+9/conf/security/policy/limited/exempt_local.pol
Extracting [148/590] jdk-11.0.10+9/conf/security/policy/unlimited/default_local.
Extracting [149/590] jdk-11.0.10+9/conf/security/policy/unlimited/default_US_exp
Extracting [173/590] jdk-11.0.10+9/jmods/java.management.rmi.jmoddgePackages.hh
```

```

Extracting [548/590] jdk-11.0.10+9/legal/jdk.unsupported/ADDITIONAL_LICENSE_INFO
Extracting [552/590] jdk-11.0.10+9/legal/jdk.unsupported.desktop/ADDITIONAL_LICENSE_INFO
Extracting [553/590] jdk-11.0.10+9/legal/jdk.unsupported.desktop/ASSEMBLY_EXCEPTION
Extracting [590/590] jdk-11.0.10+9/releaseappingsage.txtuffix_list.datINFOFO
Testing Java Executable: E:\biliBot\mirai\test\java\bin\java.exe
openjdk version "11.0.10" 2021-01-19
OpenJDK Runtime Environment AdoptOpenJDK (build 11.0.10+9)
OpenJDK 64-Bit Server VM AdoptOpenJDK (build 11.0.10+9, mixed mode)

Fetching iTXTech MCL Package Info from https://gitee.com/peratx/mirai-repo/raw/master/org/itxtech/mcl/package.json
Mirai Console Loader 公告栏

The latest stable version of iTXTech MCL is 1.0.5
Would you like to download it? (Y/N, default: Y) y
Start Downloading: https://github.com/iTXTech/mirai-console-loader/releases/download/v1.0.5/mcl-1.0.5.zip
Downloading: 1559368/1559368
Extracting [13/13] scripts/updater.jsjsjs
MCL startup script has been updated.
Use ".\mcl" to start MCL.

Press Enter to exit.

```

## linux之ubuntu安装mcl

在ubuntu终端进入将要安装mirai的文件夹输入以下命令：

```

curl -LJO https://github.com/iTXTech/mcl-installer/releases/download/v1.0.2/mcl-installer-1.0.2-linux-amd64
sudo chmod +x mcl-installer-1.0.2-linux-amd64
sudo ./mcl-installer-1.0.2-linux-amd64

```

```

ubuntu@VM-0-6-ubuntu:~/mirai$ curl -LJO https://github.com/iTXTech/mcl-installer/releases/download/v1.0.2/mcl-installer-1.0.2-linux-amd64
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100  639  100  639    0     0   317      0  0:00:02  0:00:02 --:--:--   317
100 718k  100 718k    0     0  170k      0  0:00:04  0:00:04 --:--:--  669k
curl: Saved to filename 'mcl-installer-1.0.2-linux-amd64'
ubuntu@VM-0-6-ubuntu:~/mirai$ ls
mcl-installer-1.0.1-linux-amd64  mcl-installer-1.0.2-linux-amd64
ubuntu@VM-0-6-ubuntu:~/mirai$ sudo chmod +x mcl-installer-1.0.2-linux-amd64

```

再根据系统的环境选择对应的mirai安装包

```
ubuntu@VM-0-6-ubuntu:~/mirai$ ./mcl-installer-1.0.2-linux-amd64
iTXTech MCL Installer 1.0.2 [OS: linux]
Licensed under GNU AGPLv3.
https://github.com/iTXTech/mcl-installer

iTXTech MCL and Java will be downloaded to "/home/ubuntu/mirai"

Checking existing Java installation.
openjdk version "11.0.10" 2021-01-19
OpenJDK Runtime Environment (build 11.0.10+9-Ubuntu-0ubuntu1.18.04)
OpenJDK 64-Bit Server VM (build 11.0.10+9-Ubuntu-0ubuntu1.18.04, mixed mode, sha
ring)

Would you like to install Java? (Y/N, default: Y) N
Fetching iTXTech MCL Package Info from https://gitee.com/peratx/mirai-repo/raw/m
aster/org/itxtech/mcl/package.json
Mirai Console Loader 公告栏

The latest stable version of iTXTech MCL is 1.0.5
Would you like to download it? (Y/N, default: Y) Y
Start Downloading: https://github.com/iTXTech/mirai-console-loader/releases/down
load/v1.0.5/mcl-1.0.5.zip
Downloading: 1559368/1559368
Extracting [13/13] scripts/updater.jsjsjs
```

## 安装mirai-http-api

mirai安装完成后需要先运行一遍以生成“plugins”文件夹：

1.进入mirai的安装目录。

2.window系统需要在命令行中输入 `.\mcl` 运行mirai；

linux系统则要在命令行中输入以下命令行：

```
sudo chmod +x mcl
sudo ./mcl
```

3.运行mcl后，输入 `/autologin add bot的QQ号 bot的密码`，添加自动登陆的QQ号

4.接着安装mirai-api-http：退出mirai，将mirai文件夹中的“mirai-api-http-v1.9.8.mirai.jar”放入“/mirai安装文件夹/plugins”文件夹，再运行mcl就能使用mirai-http了

```
2021-02-23 18:19:33 I/main: Frontend Terminal: version 2.4.0, provided by Mamoe
Technologies
2021-02-23 18:19:33 I/plugin: Successfully loaded plugin net.mamoe.mirai-api-htt
p
2021-02-23 18:19:34 I/main: Prepared built-in commands: autoLogin, help, login,
permission, status, stop
2021-02-23 18:19:34 W/net.mamoe.mirai-api-http: USING INITIAL KEY, please edit t
he key
2021-02-23 18:19:34 I/net.mamoe.mirai-api-http: Starting Mirai HTTP Server in 0.
0.0.0:8080
2021-02-23 18:19:34 I/Mirai HTTP API: Http api server is running with authKey: I
NITKEYYrxSLfrJH
2021-02-23 18:19:34 I/net.mamoe.mirai-api-http: 心跳模块启用状态: false
2021-02-23 18:19:34 I/net.mamoe.mirai-api-http: 上报模块启用状态: false
2021-02-23 18:19:34 I/main: 1 plugin(s) enabled.
2021-02-23 18:19:34 I/main: Auto-login [redacted]
2021-02-23 18:19:35 I/Bot.1909239542: Logging in...
2021-02-23 18:19:36 I/Bot.1909239542: No OtherClient online.
2021-02-23 18:19:37 I/Bot.1909239542: Login successful
2021-02-23 18:19:37 I/main: mirai-console started successfully.
```

5.再对mirai-api-http进行配置,进入文件夹“/mirai的安装文件夹/config/net.mamoe.mirai-api-http/”，修改“setting.yml”配置文件中的authKey等信息

```

cors:
  - '*'
host: 0.0.0.0
port: 8080
authKey: 
cacheSize: 4096
enableWebsocket: false
report:
  enable: false
  groupMessage:
    report: true
  friendMessage:
    report: true
  tempMessage:
    report: true
  eventMessage:
    report: true
  destinations: []
  extraHeaders: {}

heartbeat:
  enable: false
  delay: 1000
  period: 15000
  destinations: []
  extraBody: {}

-- INSERT --
5,25 Top

```

## 运行main.js

1.根据mirai的配置，在main.js中对main.js的配置进行自定义。

```

const Mirai = require("node-mirai-sdk");
const { Plain, At } = Mirai.MessageComponent;
const repair = require("../scripts/repair.js");

/**
 * 服务端设置(*)
 * host: mirai-api-http 的地址和端口, 默认是 http://127.0.0.1:8080
 * authKey: mirai-api-http 的 authKey, 建议手动指定
 * qq: 当前 BOT 对应的 QQ 号
 * enableWebsocket: 是否开启 WebSocket, 需要和 mirai-api-http 的设置一致
 */
const bot = new Mirai({
  host: 'http://127.0.0.1:8080',
  authKey: ' ',
  qq: 1909239542,
  enableWebsocket: false,
});

// auth 认证(*)
bot.onSignal('authed', () => {
  console.log(`Authed with session key ${bot.sessionKey}`);
  bot.verify();
});

```

2.接着安装node项目的依赖项：

```
sudo npm install --registry=https://registry.npm.taobao.org --unsafe-perm
```

3.接着在命令行中输入 `node main.js` 即可运行bot，输入 `sudo nohup node main.js >dev/null 2>&1 &` 可以后台运行bot且不使用输出日志

## 其它

后台运行mcl：`sudo nohup ./mcl >dev/null 2>&1 &`，可以不使用输入日志