Table of Contents

Introduction

this is alick's note book

email: alick97@outlook.com

github: https://github.com/alick97/note

online: https://alick97.github.io/note

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

something about linux

note software in linux, like how to setup and use

gitbook

created: 20221001 updated: 20230305 authors: alick97

> note: gitbook-cli is legacy, suggest using honkit which is gitbook alternatives

preview markdown file in cli

in ubuntu16.04

1. 安装 node.js

```
download node.js from https://nodejs.org/en/
wget https://nodejs.org/dist/v8.9.4/node-v8.9.4-linux-x64.tar.xz
xz -d node-v8.9.4-linux-x64.tar.xz
tar -xvf node-v8.9.4-linux-x64.tar

移动到指定的位置，创建软链接后，方便下次设置path环境变量后直接启动
mv node-v8.9.4-linux-x64 ~/.bin
创建软连接
ln -s ~/.bin/node-v8.9.3-linux-x64/bin/npm ~/.bin/npm
ln -s ~/.bin/node-v8.9.3-linux-x64/bin/node ~/.bin/node
ln -s ~/node-v8.9.3-linux-x64/bin/gitbook ~/.bin/gitbook

ls
lrwxrwxrwx  1 alick alick    50 1月   3 02:16 gitbook -> /home/alick/.bin/node-v8.9.
lrwxrwxrwx  1 alick alick    47 1月   3 02:13 node -> /home/alick/.bin/node-v8.9.3-l
drwxrwxr-x  7 alick alick  4096 1月   3 02:14 node-v8.9.3-linux-x64/
lrwxrwxrwx  1 alick alick    46 1月   3 02:11 npm -> /home/alick/.bin/node-v8.9.3-li
设置 path 方便shell中直接使用
在 ~/.bashrc 中添加
export PATH=/home/alick/.bin:$PATH
```

2. 安装gitbook-cli和使用

> 官网链接

1. install GitBook:

   ```
   $ npm install gitbook-cli -g
   ```

   gitbook-cli is an utility to install and use multiple versions of GitBook on the same system. It will automatically install the required version of GitBook to build a book. Create a book

2. GitBook can setup a boilerplate book:

```
$ gitbook init
```

If you wish to create the book into a new directory, you can do so by running gitbook init ./directory

3. Preview and serve your book using:

```
$ gitbook serve
```

4. Or build the static website using:

```
$ gitbook build
```

Install pre-releases

5. gitbook-cli makes it easy to download and install other versions of GitBook to test with your book:

```
$ gitbook fetch beta
```

Use gitbook ls-remote to list remote versions available for install.

6. Debugging You can use the options --log=debug and --debug to get better error messages (with stack trace). For example:

```
$ gitbook build ./ --log=debug --debug
```

export to pdf

prepartion in ubuntu

```
sudo apt install calibre
```

when use chinese, you need one chinese font at least for export to pdf

You can simply run cmd below:

```
sudo apt install ttf-wqy-microhei
```

or download font files to `~/.fonts` and run `fc-cache -f -v`

run cmd

```
gitbook pdf ./ t.pdf
```

honkit gitbook alternatives

created: 20231223 updated: 20231223 authors: alick

HonKit is building beautiful books using Markdown - Fork of GitBook

git-config-ssh-connection-through-proxy

created: 20221217 updated: 20221217 authors: alick97

---

issue: cannot connect to github.com port 22 through ssh directly some times in china.

solved methods:

> https://stackoverflow.com/questions/19161960/connect-with-ssh-through-a-proxy

config ssh command in file ~/.ssh/config

> change 127.0.0.1:1089 to your proxy address
>
> ```
> Host github.com
> ProxyCommand nc -x 127.0.0.1:1089 %h %p
> ```

audiorelay-use-phone-mic-as-pc-mic

created: 20230322 updated: 20230322 authors: alick97

- official link
- use-your-phone-as-a-mic-for-a-linux-pc

android terminal emulator termux

created: 20230325 updated: 20230325 authors: alick97

---

termux official link

run nc server for pc connect to android

On android ‘server’ side:

```
$ rm -f /tmp/f; mkfifo /tmp/f
$ cat /tmp/f | /bin/sh -i 2>&1 | nc -l 0.0.0.0 4444 > /tmp/f
```

On pc ‘client’ side:

```
$ nc [phone-ip] 4444
```

run sshd for pc connect to android

On android

```
pkg upgrade
pkg install openssh
sshd -p 4444
```

> see remote acess by ssh

run x86 docker in qemu vm without rooting android

> Install Docker on Termux, if link missing, you can search keyword to find same tutoral. link
>
> alpinelinux install alpine in qemu

Cross-Compiled Linux From Scratch

- link

home page about clfs

http://clfs.org/view/clfs-embedded/arm/index.html

start about linux(language chinese)

http://users.cecs.anu.edu.au/~okeefe/p2b/chinese/power2bash.html

fat32 file system

created: 20240503 updated: 20240503 authors: alick97

fat32 format explore

create fat32 dev file

```
dd if=/dev/zero of=./test_fat32_file bs=1M count=10
mkfs.vfat -F 32 -v ./test_fat32_file
```

mount file

```
mkdir t_dir
sudo mount -t vfat -o rw,user,uid=1000,umask=007,exec ./test_fat32_file ./t_dir
```

create file and dir for exployer fat32 how to origanize struct for dir and file

```
cd t_dir
mkdir -p dir_bbb/{dir_ccc,dir_ddd}
echo "eee" > dir_bbb/file_eee.txt
echo "aaa" > file_aaa.txt
```

first 512 bytes

```
Hex View  00 01 02 03 04 05 06 07  08 09 0A 0B 0C 0D 0E 0F

00000000  EB 58 90 6D 6B 66 73 2E  66 61 74 00 02 01 20 00  .X.mkfs.fat... .
00000010  02 00 00 00 50 F8 00 00  20 00 02 00 00 00 00 00  ....P... .......
00000020  00 00 00 00 9E 00 00 00  00 00 00 00 02 00 00 00  ................
00000030  01 00 06 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00000040  80 01 29 20 5B 7A 86 4E  4F 20 4E 41 4D 45 20 20  ..) [z.NO NAME  
00000050  20 20 46 41 54 33 32 20  20 20 0E 1F BE 77 7C AC    FAT32   ...w|.
00000060  22 C0 74 0B 56 B4 0E BB  07 00 CD 10 5E EB F0 32  ".t.V.......^..2
00000070  E4 CD 16 CD 19 EB FE 54  68 69 73 20 69 73 20 6E  .......This is n
00000080  6F 74 20 61 20 62 6F 6F  74 61 62 6C 65 20 64 69  ot a bootable di
00000090  73 6B 2E 20 20 50 6C 65  61 73 65 20 69 6E 73 65  sk.  Please inse
000000A0  72 74 20 61 20 62 6F 6F  74 61 62 6C 65 20 66 6C  rt a bootable fl
000000B0  6F 70 70 79 20 61 6E 64  0D 0A 70 72 65 73 73 20  oppy and..press 
000000C0  61 6E 79 20 6B 65 79 20  74 6F 20 74 72 79 20 61  any key to try a
000000D0  67 61 69 6E 20 2E 2E 2E  20 0D 0A 00 00 00 00 00  gain ... .......
000000E0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
000000F0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00000100  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00000110  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00000120  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00000130  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00000140  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00000150  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00000160  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00000170  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00000180  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00000190  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
000001A0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
000001B0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
000001C0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
000001D0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
000001E0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
000001F0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 55 AA  ..............U.
```

find fat32 table #1 address

- sector size in 0x0b(2).
- find align 0x0e(2) value is 32, so align is 32 * 512 = 0x4000 512 is sector size.
- value

```
Hex View  00 01 02 03 04 05 06 07  08 09 0A 0B 0C 0D 0E 0F

00004000  F8 FF FF 0F FF FF FF 0F  F8 FF FF 0F FF FF FF 0F  ................
00004010  FF FF FF 0F FF FF FF 0F  FF FF FF 0F FF FF FF 0F  ................
00004020  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
```

- fat32 table 4 bytes is one cluster data address or flag. cluster index is 1,2,3,4... one cluster address is 4 bytes.

- cluster data size is 0x0d(1) = 1 sector = 512bytes

find root dir address

- has 2 fat table 0x10(1)=2, one table size 0x24(4) = 158 sector, 158 * 512 = 0x13c00
- 0x4000 + (0x13c00 * 2) = 0x2b800
- value

```
Hex View  00 01 02 03 04 05 06 07  08 09 0A 0B 0C 0D 0E 0F

0002B800  41 64 00 69 00 72 00 5F  00 62 00 0F 00 18 62 00  Ad.i.r._.b....b.
0002B810  62 00 00 00 FF FF FF FF  FF FF 00 00 FF FF FF FF  b...............
0002B820  44 49 52 5F 42 42 42 20  20 20 20 10 00 84 16 27  DIR_BBB    ....'
0002B830  A3 58 A3 58 00 00 1A 27  A3 58 03 00 00 00 00 00  .X.X...'.X......
0002B840  41 66 00 69 00 6C 00 65  00 5F 00 0F 00 E5 61 00  Af.i.l.e._....a.
0002B850  61 00 61 00 2E 00 74 00  78 00 00 00 74 00 00 00  a.a...t.x...t...
0002B860  46 49 4C 45 5F 41 41 41  54 58 54 20 00 61 20 27  FILE_AAATXT .a '
0002B870  A3 58 A3 58 00 00 20 27  A3 58 07 00 04 00 00 00  .X.X.. '.X......
0002B880  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
```

- file_aaa.txt cluster is (0x2b870, 04)0 (0x2b870, 0a)07 = 7, first 2 cluster is reserved partition. so cluster index is 7 - 2 = 5
- file_aaa.txt data address is 5 * 0x200 + 0x2b800 = 0x2c200, 0x200=512bytes

find data address, in fat32 data address is same as root dir address

- which is 0x2b800

read file_aaa.txt example code, run as root

- read_file_aaa.c

```c
#include <sys/types.h>
#include <unistd.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <stdio.h>
#include <errno.h>
#include <string.h>


int main() {
    int dev_f = open("/dev/loop20", O_RDONLY);
    const int sector_size = 0x200;
    unsigned char buf[0x200] = {0};
    int fat_table_1_pos = 32 * sector_size;
    int data_pos = fat_table_1_pos + (158 * sector_size * 2);
    // find aaa cluster index, change this value in 0x0a column
    int file_aaa_cluster_index = 7;
    int cluster_size = sector_size * 1;
    printf("data pos: %8x\n", data_pos);
    lseek(dev_f, data_pos, SEEK_SET);
    int n = read(dev_f, buf, sizeof(buf));
    if (n < 0) {
        printf("err: %s\n", strerror(errno));
        return n;
    }
    int i, j;
    printf("----: ");
    for (j=0; j<0x10; j++) {
        printf("%02x ", j);
    }
    printf("\n");
    for (i=0; i< 0x20; i++) {
        printf("%04x: ", i);
        for (j=0; j < 0x10; j++) {
            printf("%02x ", buf[i*0x10+j]);
        }
        printf("\n");
    }
    unsigned long new_pos = lseek(dev_f, cluster_size * (file_aaa_cluster_index - 2
    printf("new pos: %08lx\n", new_pos);

    n = read(dev_f, buf, sizeof(buf));
    // expect aaa
    printf("file content: %s\n", buf);
    return 0;
}
```

- result

```
data pos:    2b800
----: 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f
0000: 41 64 00 69 00 72 00 5f 00 62 00 0f 00 18 62 00
0001: 62 00 00 00 ff ff ff ff ff ff 00 00 ff ff ff ff
0002: 44 49 52 5f 42 42 42 20 20 20 20 10 00 84 16 27
0003: a3 58 a3 58 00 00 1a 27 a3 58 03 00 00 00 00 00
0004: 41 66 00 69 00 6c 00 65 00 5f 00 0f 00 e5 61 00
0005: 61 00 61 00 2e 00 74 00 78 00 00 00 74 00 00 00
0006: 46 49 4c 45 5f 41 41 41 54 58 54 20 00 61 20 27
0007: a3 58 a3 58 00 00 20 27 a3 58 07 00 04 00 00 00
0008: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0009: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000a: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000b: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000c: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000d: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000e: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000f: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0010: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0011: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0012: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0013: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0014: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0015: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0016: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0017: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0018: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0019: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
001a: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
001b: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
001c: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
001d: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
001e: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
001f: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
new pos: 0002c200
file content: aaa
```

reference

- https://en.wikipedia.org/wiki/File_Allocation_Table
- cn link https://zh.wikipedia.org/wiki/%E6%AA%94%E6%A1%88%E9%85%8D%E7%BD%AE%E8%A1%A8#%E4%B8%BB%E7%A3%81%E7%9B%98%E7%BB%93%E6%9E%84
- tools ImHex

vpn setting and vpn software

ubuntu 设置过程记录

1. 安装 shadowsocket

   ```
   sudo apt-get install python-pip
   sudo pip install shadowsocks
   ```

2. 配置客户 local_port 自己随便配置， 其他的参数和服务器端的保持一致

   ```
   alick@alick-ubuntu:~/workspace/vpn$ cat shadowsocks_g.json
   {
        "server":"144.48.141.168",
        "server_port":443,
        "local_address": "127.0.0.1",
        "local_port":1080,
        "password":"xxx",
        "timeout":600,
        "method":"aes-256-cfb"
   }
   ```

3. 启动客户端 -d start 选项是让在后台运行， 调试查看log 可以先不加这个参数

   ```
   alick@alick-ubuntu:~/workspace/vpn$ sudo sslocal -c shadowsocks_g.json -d start
   ```

   到这里，已经支持socks5协议的代理， http的不支持

检验方式:

- curl

有代理的环境变量http_proxy， 端口1080是前边设置的本地端口， 可以设置其他，但是要和前面的保持一直

```
alick@alick-ubuntu:~$ http_proxy=socks5://localhost:1080 curl ip.gs
Current IP / 当前 IP: 144.48.141.168
ISP / 运营商:  serverrun.com
City / 城市:  Hong Kong
Country / 国家: China
IP.GS is now IP.SB, please visit https://ip.sb/ for more information. / IP.GS 已更改
Please join Telegram group https://t.me/sbfans if you have any issues. / 如有问题，请

  /\_/\
=( °w° )=
  )   (  //
 (__ __)//
```

- google-chrome

google-chrome的代理参数[目前不起作用]

```
google-chrome --proxy-server="socks5://localhost:1080"
```

chromium[可以]

```
chromium --proxy-server="socks5://localhost:1080"
```

到这里已经满足基本的功能

1. 支持http协议的代理，使用http代理服务，这里使用polipo
2. 安装软件

   ```
   $ sudo apt-get install polipo
   ```

3. 修改配置， 其中socksParentProxy 是之前shadowsocker代理的本地端口1080，这里对这个socks代理端口再代理， 转换成http协议代理端口， polipo本地代理端口默认为8123

```
alick@alick-ubuntu:~$ cat /etc/polipo/config
# This file only needs to list configuration variables that deviate
# from the default values.  See /usr/share/doc/polipo/examples/config.sample
# and "polipo -v" for variables you can tweak and further information.

logSyslog = true
logFile = /var/log/polipo/polipo.log

# alick add beg
socksParentProxy = 127.0.0.1:1080
socksProxyType = socks5
logLevel=4
# alick add end
```

- 启动服务

  ```
  sudo service  polipo start
  ```

  ## 检验

- curl

```
alick@alick-ubuntu:~$ http_proxy=http://localhost:8123 curl ip.gs
Current IP / 当前 IP: 144.48.141.168
ISP / 运营商:  serverrun.com
City / 城市:  Hong Kong
Country / 国家: China
IP.GS is now IP.SB, please visit https://ip.sb/ for more information. / IP.GS 已更改
Please join Telegram group https://t.me/sbfans if you have any issues. / 如有问题，请

  /\_/\
=( °w° )=
  )   (  //
 (__ __)//
```

- firfox

```
http_proxy=http://127.0.0.1:8123 firefox
```

一般的shell环境都用 http_proxy 环境变量

- google-chrome[设置不起作用， 先用chromiu]

```
chromium --proxy-server="http://localhost:8123"
```

1. 设置ubuntu的全局[不推荐] 在网络设置中，设置代理 填入http代理http://localhost:8123

自己的启动脚本

```
#!/bin/bash -e

function show_help() {
    echo "my vpn service ctrl\

          usage: $0 [stop|start|-h|--help]
"
}

while getopts ":h-:" opt;do
    case "$opt" in
        h)
            show_help
            exit 0
            ;;
        -)
            opt=$OPTARG
            case $opt in
                start)
                     sudo sslocal -c /home/alick/workspace/vpn/shadowsocks_g.json -d
                     sudo service polipo restart
                    ;;
                stop)
                        sudo service polipo stop
                        sudo sslocal -c /home/alick/workspace/vpn/shadowsocks_g.json
                    ;;
                help)
                        show_help && exit 0;;
                *) echo "param error: $opt" && show_help && exit 1
                    ;;
            esac
                    ;;
        *)
            echo "error param" && show_help && exit 1
            ;;
        esac
done

shift $[$OPTIND -1]
if [ "$*" != "" ];then
    echo "param error: $@" && exit 1
fi
```

start chrome

```
alick@alick-ubuntu:~/.local/bin$ cat my_chrome
#!/bin/bash -e
if [[ "`ps -ef | grep sslocal`" == "" ]] || [[ "`ps -e | grep polipo`" == "" ]];then
    my_proxy --stop
    my_proxy --start
fi
chromium --proxy-server="localhost:8123"
```

other record

https://blog.csdn.net/jdliyao/article/details/80609742 https://www.linuxbabe.com/desktop-linux/how-to-install-and-use-shadowsocks-command-line-client

报错处理

> ShadowSocks启动报错undefined symbol EVP_CIPHER_CTX_cleanup
> https://kionf.com/2016/12/15/errornote-ss/

```
ubuntu18.04中报错解决方法 :
1. vim打开文件openssl.py
vim /usr/local/lib/python3.5/dist-packages/shadowsocks/crypto/openssl.py

路径不同根据报错路径而定

2. 修改libcrypto.EVP_CIPHER_CTX_cleanup.argtypes
:%s/cleanup/reset/

:x

以上两条为VIM命令, 替换文中libcrypto.EVP_CIPHER_CTX_cleanup.argtypes 为libcrypto.EVP_CI

3. 运行Shadowsocks ok
```

原因

```
这个问题是由于在openssl1.1.0版本中，废弃了EVP_CIPHER_CTX_cleanup函数，如官网中所说：

EVP_CIPHER_CTX was made opaque in OpenSSL 1.1.0. As a result, EVP_CIPHER_CTX_reset(

EVP_CIPHER_CTX_reset函数替代了EVP_CIPHER_CTX_cleanup函数

EVP_CIPHER_CTX_reset函数说明：

EVP_CIPHER_CTX_reset() clears all information from a cipher context and free up any

EVP_CIPHER_CTX_cleanup函数说明：

EVP_CIPHER_CTX_cleanup() clears all information from a cipher context and free up a

可以看出，二者功能基本上相同，都是释放内存，只是应该调用的时机稍有不同，所以用reset代替cleanup问题
```

查看温度

第一次要更新检测配置

```
sudo sensors-detect
```

查看温度

```
sensors
```

输出

```
Core 0:        +49.0°C  (high = +100.0°C, crit = +100.0°C)
Core 1:        +49.0°C  (high = +100.0°C, crit = +100.0°C)
Core 2:        +49.0°C  (high = +100.0°C, crit = +100.0°C)
Core 3:        +49.0°C  (high = +100.0°C, crit = +100.0°C)
Core 4:        +51.0°C  (high = +100.0°C, crit = +100.0°C)
Core 5:        +48.0°C  (high = +100.0°C, crit = +100.0°C)

BAT0-acpi-0
Adapter: ACPI interface
in0:          14.32 V
curr1:         1.79 A

iwlwifi_1-virtual-0
Adapter: Virtual device
temp1:        +38.0°C

pch_cannonlake-virtual-0
Adapter: Virtual device
temp1:        +69.0°C

acpitz-acpi-0
Adapter: ACPI interface
temp1:        +51.0°C  (crit = +120.0°C
```

安装电池管理服务

> https://linrunner.de/tlp/
>
> ```
> sudo add-apt-repository ppa:linrunner/tlp
> sudo apt update
> sudo apt install tlp tlp-rdw
> sudo tlp start
> ```

安装ui工具

> https://github.com/d4nj1/TLPUI
>
> ```
> sudo add-apt-repository -y ppa:linuxuprising/apps
> sudo apt update
> sudo apt install tlpui
> ```

中文输入法

created: 20221001 updated: 20221130 authors: alick97

安装ibus-pinyin

```
sudo apt install ibus-pinyin
```

配置

- ibus setup add input method, run cmd `ibus-setup`. In setting panel, Input Method --> Add --> Chinese --> Pinyin
- Add input sources, Global Settings --> Keyboards --> Input Sources +， 选择Chinese --> Chinese(Pinyin) --> Add，那么在Settings界面的Input Sources里面就会多出一个Chinese(Pinyin)

english input

created: 20250502 updated: 20250502 authors: alick97

---

ibus-typing-booster

> https://mike-fabian.github.io/ibus-typing-booster/

- has english auto suggestion.

wifi wpa2 attack

created: 20220709 updated: 20221210 authors: alick97

- https://www.mzbky.com/1791.html
- https://zh.wikihow.com/%E4%BD%BF%E7%94%A8Kali-Linux%E7%A0%B4%E8%A7%A3WPA%E6%88%96WPA2%E6%97%A0%E7%BA%BF%E5%B1%80%E5%9F%9F%E7%BD%91
- [for mac os] https://martinsjean256.wordpress.com/2018/02/12/hacking-aircrack-ng-on-mac-cracking-wi-fi-without-kali-in-parallels/
- [for mac os] https://louisabraham.github.io/articles/WPA-wifi-cracking-MBP.html

packages install in ubuntu

```
sudo apt install aircrack-ng hashcat

wget https://github.com/hashcat/hashcat-utils/archive/master.zip
unzip master.zip
cd hashcat-utils-master/src
make
# get cap2hccapx.bin
```

#

一些常见命令

常用指令

```
# 启动监听 会创建新的net interface 后缀为 mon
airmon-ng start wlan0
# 停止监听 net interface
airmon-ng stop wlan0
# 关闭相关没开启mon之前用到interface的service
airmon-ng start check --kill


# 监听所有
airodump-ng wlan0mon

# 发起断开连接
sudo aireplay-ng -0 10 -a 24:CF:24:32:F2:4F -c 60:AB:67:E8:C6:44  wlo1mon
# 支持搜索5g 默认2.4g
sudo airodump-ng --band abg --output-format pcap -w t -c 10 --bssid 24:CF:24:32:F2:4

# hashcat相关
cap2hccapx capture.cap capture.hccapx
hashcat -m 2500 capture.hccapx wordlist.txt
hashcat -m 2500 -a3 capture.hccapx ?d?d?d?d?d?d?d?d
sudo hashcat --deprecated-check-disable -m 2500 t2.hccapx ../dict/rockyou.txt
```

ubuntu关闭mon interface后 重启network

```
sudo systemctl restart NetworkManager
```

proxy frp

created: 20221004 updated: 20221004 authors: alick97

---

docs

- examples

docker image

- images

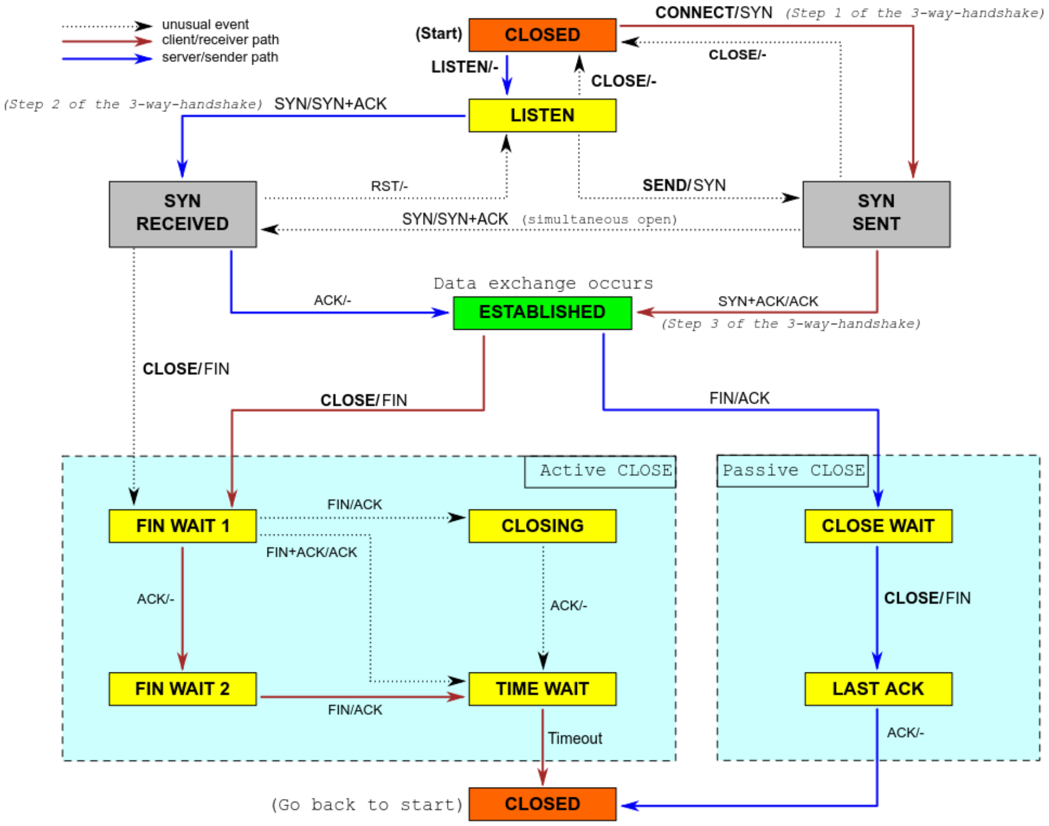
> free server
>
> - https://freefrp.net/docs

net-state

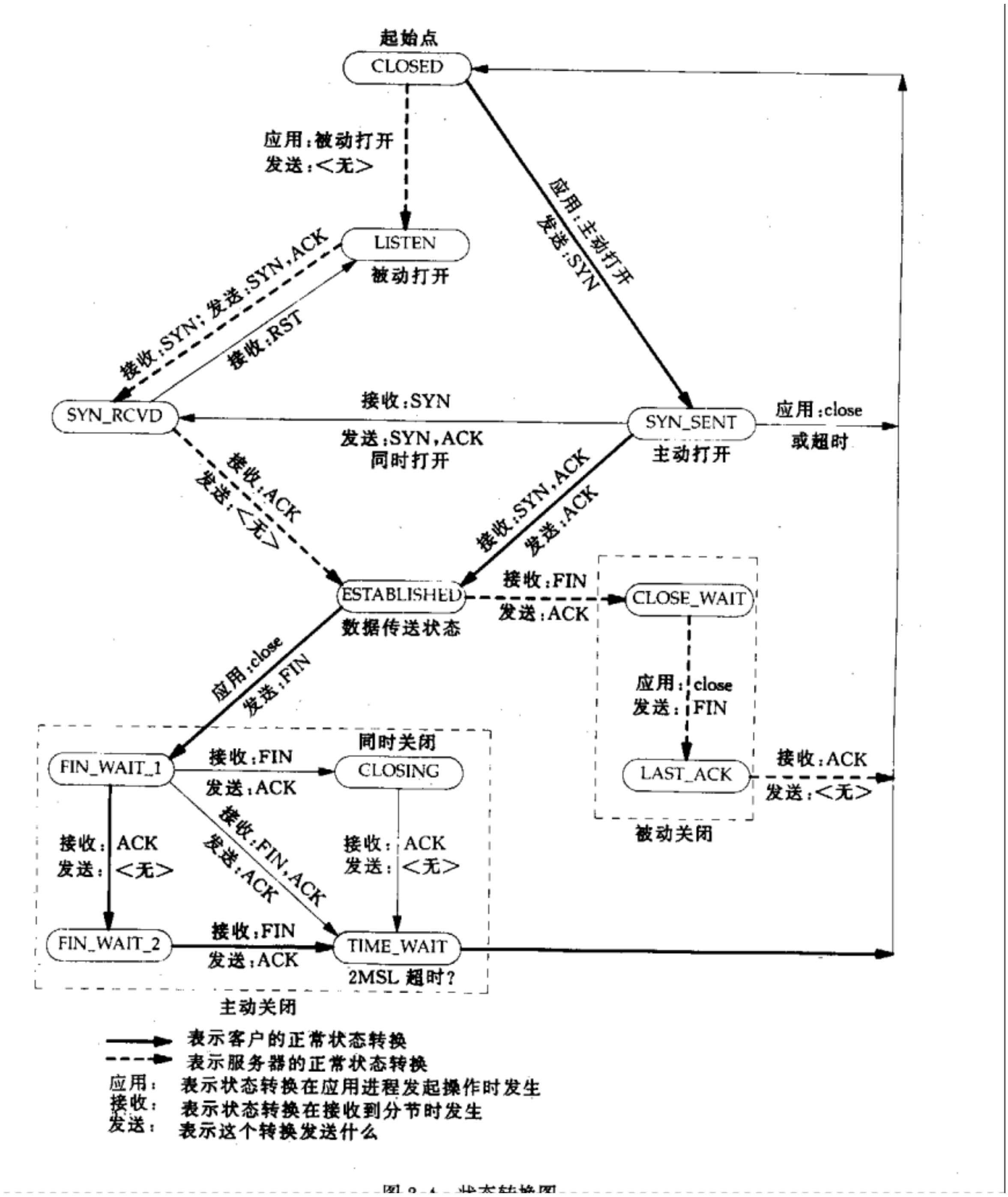
created: 20230827 updated: 20230827 authors: alick97

net state

- net state wiki



- net state picture language in cn



iptables use linux as windows gateway and forward target connect to linux local port

created: 20240425 updated: 20240425 authors: alick97

background

- use linux as gateway
- window10 pc config gateway to linux
- window10 connect to public ip port 112.60.14.252:6666, linux not send this connect, redirect it to local
- router ip 192.168.31.1
- linux ip 192.168.31.158
- window10 ip 192.168.31.172
- window10 and linux connect to router

in win, config window10 gateway

- control panel -> internat -> connect -> property -> TCP/IPv4 -> property

  ```
  ip 192.168.31.172
  netmask 255.255.255.0
  gateway 192.168.31.158   (linux pc ip)
  ```

- dns server

  ```
  114.114.114.114
  8.8.8.8
  ```

in linux config forward and start http server

- config ip forward

  ```
  sysctl -w net.ipv4.ip_forward=1
  ```

- config gateway rule by iptables, wlo1 is net out interface

  ```
  sudo iptables -t nat -A POSTROUTING -s 192.168.31.172 -o wlo1  -j MASQUERADE
  ```

- config redirect rule by iptables for special windows10 connect

  ```
  sudo iptables -t nat -A PREROUTING -p tcp -d 112.60.14.252 --dport 6666 -j DNAT
  sudo service iptables restart
  ```

- start demo server in linux

  ```
  python3 -m http.server  7777
  ```

in win, access target url

- in browser access `http://112.60.14.252:6666/`
- all will be sended to linux 7777 port
- in wireshark see `ip.addr == 112.60.14.252`

more iptables

> http://cn.linux.vbird.org/linux_server/0250simple_firewall.php#nat_what
>
> https://www.karlrupp.net/en/computer/nat_tutorial

capture app https message simply

created: 20240719 updated: 20240719 authors: alick

---

background information

- I want to get a black box app to call which api.
- This app install in win10.
- I work in linux and run win10 in virtualbox.

use software reqable as proxy to do MITM[Man in the Middle]

> reqable Advanced API Debugging Proxy

setup in linux

- install software reqable in linux, you can see proxying on 192.168.x.x:9000, config this later in win as proxy.
- export root ca file, in software reqable, run Certificate -> Root Certificate Management -> Export Root Public Certificate. this will export reqable-ca.crt file.
- click reqable start button.

setup in win10

- install dummy root CA to win10.

  > import certificate to trusted root authorities
  >
  > example in chinese language link

- set win10 net proxy to software reqable

```
To set up a proxy manually:

Open Settings.
Click Network & Internet.
Click Proxy.
In the Manual Proxy Setup section, set the Use a Proxy Server switch to On.
In the Address field, type the IP address given to you by your network administrator
In the Port field, type the port given to you by your network administrator.
Click Save, then close the Settings window.
```

start capture

- run need capture app in win10
- see catpured message in reqable.

reference

- https
- other good mitm tools mitmproxy

sshd-auth-password-delay

created: 20221030 updated: 20221030 authors: alick97

config delay when password auth failed

check pam_faildelay.so installed

```
locate pam_faildelay.so
```

config pam sshd

```
sudo vim /etc/pam.d/sshd
```

add line to top

```
auth  required  pam_faildelay.so  delay=60000000
```

delay unit is us, 60000000 means 1min

Note

> https://stackoverflow.com/questions/69823279/why-is-pam-faildelay-so-not-taking-effect-if-user-clicks-cancel-button

ansible

created: 20221123 updated: 20221126 authors: alick97

Document

- ansible-core
- playbook

run ubuntu in qemu with kvm

created: 20240107 updated: 20240107 authors: alick

download ubuntu iso

```
aria2c -c -x 10 -j 10 http://cdimage.ubuntu.com/ubuntu/releases/18.04/release/ubuntu
```

create hda disk and load img to setup system

```
qemu-img create vdisk.img 10G
qemu-system-x86_64 -cpu host -enable-kvm -hda vdisk.img -cdrom ./ubuntu-18.04.3-serv
```

start installed system and just in console with ssh port

```
qemu-system-x86_64 -cpu host -enable-kvm -hda vdisk.img -boot c -m 4096  -nographic
    -nic user,hostfwd=tcp:127.0.0.1:4022-:22
```

> https://qemu-project.gitlab.io/qemu/system/images.html
>
> https://serverfault.com/questions/471719/how-to-start-qemu-directly-in-the-console-not-in-curses-or-sdl

```
You can update GRUB to pass the required options to the kernel. I'm using Ubuntu 18.

Update grub in the guest OS:

sudoedit /etc/default/grub

Change GRUB_CMDLINE_LINUX="" to GRUB_CMDLINE_LINUX="console=tty0 console=ttyS0,9600n
run sudo update-grub
```

linux capabilities

created: 20240113 updated: 20240113 authors: alick97

setcap to program for running program without root

example running nethogs without root

- not setting cap

  ```
  alick@alick-pc:/tmp$ nethogs
  To run nethogs without being root you need to enable capabilities on the program
  ```

- set cap to execute file. now run program without root

  ```
  alick@alick-pc:/tmp$ sudo setcap   'cap_net_admin+eip cap_net_raw+eip' /usr/sbin
  ```

- example for delete cap

  ```
  alick@alick-pc:/tmp$ sudo setcap   'cap_net_admin-eip cap_net_raw-eip' /usr/sbin
  ```

reference:

- man 8 capabilities
- man 8 setcap
- man 3 cap_from_text

create or resize linux swap

created: 20240211 updated: 20240211 authors: alick97

how to resize swap

create new swap file eg: 8GB

- create empty file

  ```
  sudo dd if=/dev/zero of=/swapfile.img bs=1024 count=8M
  sudo chmod 600 /swapfile.img
  ```

- change format to swap

  ```
  sudo mkswap /swapfile.img
  ```

- mount to filesystem

  ```
  sudo swapon /swapfile.img
  ```

- set for reboot

  ```
  # vim /etc/fstab
  # add line
  /swapfile.img                                 none            swap    sw
  ```

if exist old swap, stop it and delete it.

- check it

  ```
  cat /etc/fatab
  ```

- stop it, eg: it is /swapfile

  ```
  sudo swapoff /swapfile
  ```

- remove if from tab for reboot

  ```
  # vim /etc/fstab
  # find line /swapfile and remove line
  ```

- remove file

  ```
  sudo rm /swapfile
  ```

decompiler android app

created: 20240723 updated: 20240723 authors: alick97

- unzip app file
- find classes.dex file
- download https://github.com/pxb1988/dex2jar/releases
- unzip dex-tools-v2.4 and mv to /opt/
- `/opt/dex-tools-v2.4/d2j-dex2jar.sh -f classes.dex`
- find *.jar
- download https://github.com/java-decompiler/jd-gui/releases
- start jd-gui and mv xx.jar to it

android adb cheatsheat

created: 20250503 updated: 20250503 authors: alick97

resize window

```
# 1090 width, 2048 height
adb shell wm size 1090x2048
adb shell wm size reset
```

fix laptops 4 pin 3.5mm microphone is not detected issue

created: 20241117 updated: 20241117 authors: alick97

> see Correctly detect microphone plugged in a 4-pin 3.5mm xxx_jack) in alsa(Advanced Linux Sound Architecture) document.

fix my acer aspire E1-570G laptops step by step

- my linux use alsa sound system and can play sound but can not use microphone.
- list sound card and found use device ALC282

  ```
  aplay -l
  ```

  ```
  *** List of PLAYBACK Hardware Devices ****
  card 0: PCH [HDA Intel PCH], device 0: ALC282 Analog [ALC282 Analog]
    Subdevices: 1/1
    Subdevice #0: subdevice #0
  card 0: PCH [HDA Intel PCH], device 3: HDMI 0 [HDMI 0]
    Subdevices: 1/1
    Subdevice #0: subdevice #0
  ```

- find alc282 driver for pc acre aspire e1 visit https://docs.kernel.org/sound/hd-audio/models.html and find ALC22x/23x/25x/269/27x/28x/29x (and vendor-specific ALC3xxx models) title. This one acer-aspire-e1 (Acer Aspire E1 fixups) is for my pc.
- add line to /etc/modprobe.d/alsa-base.conf

  ```
  options snd-hda-intel model=acer-aspire-e1
  ```

- reboot

About program language

Python language.

Python library

- sqlalchemy

- 官方文档 python3.7 logging

示例代码

```
import logging

# log1 = logging.getLogger('l1')
# log1_1 = logging.getLogger('l1.1')
# log2 = logging.getLogger('l1')
#
# assert id(log1) == id(log2), 'log1 and log2 must be same instance.'

# # level = log2.getEffectiveLevel()
# is_enable_info = log2.isEnabledFor(logging.WARNING)
# print(is_enable_info)
# logger = log1.getChild('12')
# print(logger)
#
# try:
#     raise ValueError('xxx')
# except Exception:
#     log1.exception("=====")
#     t = sys.exc_info()
#     trace = t[2]
#     print(t)
#     pass
#

# FORMAT = '%(asctime)-15s %(clientip)s %(user)-8s %(message)s'
# logging.basicConfig(format=FORMAT)
# d = {'clientip': '192.168.0.100', 'user': 'fbloggs'}
# logger = logging.getLogger('tcpserver')
# logger.warning('Protocol problem: %s', 'connection reset', extra=d)

a_b1_log_path='a.b1.log'

log_config = {
    'version': 1,
    'disable_existing_loggers': False,
    'handlers': {
        'console': {
            'class': 'logging.StreamHandler',
            'formatter': 'verbose',
        },
        'a_b1_file': {
            'class': 'logging.handlers.RotatingFileHandler',
            'filename': a_b1_log_path,
            'maxBytes': 1024*1024*5,
            'backupCount': 5,
            'formatter': 'verbose',
        },
    },
    'formatters': {
        'verbose': {
            'format': 'verbose: {levelname} {asctime} {module}:{funcName} {process:
                      '{threadName}{thread:d} {name}:{lineno} {message}',
            'style': '{',
        },
        'simple': {
            'format': 'simple: {levelname} {message}',
            'style': '{',
        },
    },
    'loggers': {
        'a': {
            'handlers': ['console'],
            'level': 'ERROR',
            'formatter': 'simple',
        },
        'a.b1': {
            'handlers': ['a_b1_file'],
            'level': 'INFO',
            'formatter': 'verbose',
        },
        'a.b2': {
            'handlers': ['console'],
            'level': 'WARNING',
            'formatter': 'verbose',
        }
    }
}

import logging.config

logging.config.dictConfig(log_config)


# clean log
with open(a_b1_log_path, 'w') as f:
    f.write('')

logger_a = logging.getLogger('a')
logger_a_b1 = logging.getLogger('a.b1')
logger_a_b2 = logging.getLogger('a.b2')


logger_a.error('this message from logger_a')
logger_a_b1.info('this message from logger_a_b1')

logger_a_b2.info('this message from logger_a_b2')
logger_a_b2.error('this message from logger_a_b2')
```

输出情况

```
verbose: ERROR 2020-08-12 18:57:07,894 log:<module> 5147 MainThread139875926808320 a
verbose: INFO 2020-08-12 18:57:07,895 log:<module> 5147 MainThread139875926808320 a.
verbose: ERROR 2020-08-12 18:57:07,895 log:<module> 5147 MainThread139875926808320 a
verbose: ERROR 2020-08-12 18:57:07,895 log:<module> 5147 MainThread139875926808320 a
cat a.b1.log
verbose: INFO 2020-08-12 18:57:07,895 log:<module> 5147 MainThread139875926808320 a.
```

logging简单使用总结:

配置中几个主要用到的组件:

- handers 相当与最终写入设备，可以是console， 文件，网络等等.
- formatters, 输出的log的额外的字符应该是什么样式 如时间，进程号，代码行号，函数名等
- loggers， 用于专门记录log的instance.可以配置特定的一个或多个hander，可以配置formatter，以及log的最小级别
- filters 这里没有用到，用于特定的过滤.

关于loggers

- loggers有继承关系，类似于模块命名空间,子空间的logger会流入父空间logger，如果没有父空间那么这个模块就是root. 例子:

  ```
  a
  a.b1
  a.b2
  ```

  a是a.b1与a.b2的父节点，a.b1与a.b2平级

- logger的level级别 父子节点有关联关系, 但是handler是独立的.

logger level父子节点的关系为:

1. 子节点没有筛出来的log不会出现在父节点, 如

   ```
   logger_a_b2.info('this message from logger_a_b2')
   ```

   不会被a.b2的logger筛出来，应为对应的logger的筛选级别为 WARNING.
2. 子节点筛选出来的log， 哪怕父节点的level比子节点的高， 也会出现在父亲节点中. 如 a的level 为ERROR， a.b1的level 为INFO，只要是a.b1 筛出来的log，都会在a最终筛选结果中出现. 如 ``` logger_a_b1.info('this message from logger_a_b1')

~~~
```

verbose: INFO 2020-08-12 18:57:07,895 log:<module> 5147 MainThread139875926808320 a.
~~~

所以正确的配置应该是 **子节点的level应该大于等于父节点.**

handler独立，是说log记录到对应的logger后，都会又对应logger配置的handler集合处理。这就出现一种现象， 如 a 和 a.b2 都配置了hander console，同时a.b2 level也比a的低，所以a.b2筛选出来的log都会在a出现，这样同一个log会在console出现两次,造成干扰. 如:

```
logger_a_b2.error('this message from logger_a_b2')
```

```
verbose: ERROR 2020-08-12 18:57:07,895 log:<module> 5147 MainThread139875926808320 a
verbose: ERROR 2020-08-12 18:57:07,895 log:<module> 5147 MainThread139875926808320 a
```

处理原则应该是 **子节点的log会流入父节点， 子节点不要和父节点共用同一个handler**

sqlalchemy bulk_save_object 使用中注意点记录

问题

> sqlalchemy 1.4.23

session创建的时候feature为False， 使用orm 对象外键指定的是对象 而不是id的时候 或者 不管session创建的时候feature指定False or True， 主动使用refobject.backref_collections.append(current_object) 都会导致使用 bulk_save_object 有重复插入数据的问题

关于session feature为False的说明， 其中有一个行为就是 定义外键的时候，relationship.cascade_backrefs=True。这会导致你给对象指定外键的时候只用的是外键的对象而不是id指定的时候 如 EntityB_instance.a = EntityA_instance, 就会发生一个默认的行为 EntityA_instance.bs.append(EntityB_instance) 这个行为会将EntityB_instance添加到session中，就是上面说的第二个情况，然后bulk_save_object的时候就会再加一次， 发生两次insert，原本只要一次

避免的方式

使用core原生sql， 或者使用orm_bulk_save_object的时候对象所有外键只指定id不指定对象 同时不主动或者被动发生外键对象backref集合append当前对象的情况 官方建议使用core

相关参考

> https://docs.sqlalchemy.org/en/14/orm/persistence_techniques.html#bulk-operations-caveats
> https://github.com/sqlalchemy/sqlalchemy/discussions/5928
> https://github.com/sqlalchemy/sqlalchemy/issues/5833

这个例子会复现，出现两个B对象记录

> 如果调整EntityB **init** 为

```
# self.a = a
self.a_id = a.id
```

就会正常

```
import pathlib

import sqlalchemy
import sqlalchemy.orm
from sqlalchemy.ext.declarative import declarative_base


Base = declarative_base()
class EntityA(Base):
    __tablename__ = "entity_a"

    id = sqlalchemy.Column(sqlalchemy.Integer, primary_key=True)
    name = sqlalchemy.Column(sqlalchemy.String)

    # bs = sqlalchemy.orm.relationship("EntityB", back_populates="a", cascade_backre
    bs = sqlalchemy.orm.relationship("EntityB", back_populates="a")
    # bs = sqlalchemy.orm.relationship("EntityB", backref="a")

    def __init__(self, name):
        self.name = name

    def __repr__(self):
        return f"<EntityA {self.id} '{self.name}'>"


class EntityB(Base):
    __tablename__ = "entity_b"

    id = sqlalchemy.Column(sqlalchemy.Integer, primary_key=True)
    a_id = sqlalchemy.Column(sqlalchemy.Integer, sqlalchemy.ForeignKey('entity_a.id'

    # a = sqlalchemy.orm.relationship("EntityA", backref="bs")
    a = sqlalchemy.orm.relationship("EntityA", back_populates="bs")

    def __init__(self, a):
        self.a = a
        # self.a_id = a.id  # 如果按照id只指定这个 不使用self.a = a 就 bulk_save不会发生 

    def __repr__(self):
        return f"<EntityB {self.id} {self.a_id} {self.a!r}>"


db_file = pathlib.Path(__file__).resolve().with_suffix(".db")
db_file.unlink(missing_ok=True)
db_engine = sqlalchemy.create_engine(f"sqlite:///{db_file}", echo=True)
Base.metadata.create_all(db_engine)

db_session = sqlalchemy.orm.sessionmaker(db_engine, future=True)()

a = EntityA("foo")
```

```
db_session.add(a)
db_session.commit()

aa = db_session.query(EntityA).first()
b = EntityB(aa)
db_session.bulk_save_objects([b])
db_session.commit()

print("=====")
for obj in db_session.query(EntityB):
    print("[  ", repr(obj))
```

python3 singleton metaclass 实现以及 对象模型

created: 20200920 updated: 20200920 author: alick97

示例代码 运行python3.8中:

```
class Singleton(type):
    def __init__(cls, *args, **kwargs):
        print(f"singleton __init__: {cls}")
        cls.__a__instance = None
        # super(Singleton, cls).__init__(*args, **kwargs)
        super().__init__(*args, **kwargs)
        cls.show_a = cls.show

    def show(self):
        print('show')

    def ssss(self):
        print(f'sssssss {self}')

    def __new__(cls, name, bases, dct):
        # bases default is object class.
        print(f"singleton __new__: cls {cls}, name {name}, base {bases}")
        return super().__new__(cls, name, bases, dct)

    def __call__(self, *args, **kwargs):
        print(f"singleton __call__: {self}")
        if self.__a__instance is None:
            self.__a__instance = super().__call__(*args, **kwargs)
            return self.__a__instance
        else:
            print(f'singleton {self} has_instance, {self.__a__instance}')
            return self.__a__instance


class Foo(object, metaclass=Singleton):
    AAAA = 1
    BBBB = 2

    def showccc(self):
        pass

    # aclass__ = Singleton #  在代码执行到这里的时候，元类中的__new__方法和__init__方法其实
    def __new__(cls, *args, **kwargs):
        print(f"Foo __new__: {cls}")
        return super().__new__(cls, *args, **kwargs)

    def __init__(self):
        print(f'Foo __init__: {self}')


class BBB(object, metaclass=Singleton):
    def __init__(self):
        print("BBB self init: {}".format(self))
```

```
a = Foo()
b = Foo()
c = BBB()
cc = BBB()

Foo.ssss()
# a.ssss() #  error
a.show_a()
Foo.show_a()
```

运行结果:

```
singleton __new__: cls <class '__main__.Singleton'>, name Foo, base (<class 'object'
singleton __init__: <class '__main__.Foo'>
singleton __new__: cls <class '__main__.Singleton'>, name BBB, base (<class 'object'
singleton __init__: <class '__main__.BBB'>
singleton __call__: <class '__main__.Foo'>
Foo __new__: <class '__main__.Foo'>
Foo __init__: <__main__.Foo object at 0x7ff7f036ab80>
singleton __call__: <class '__main__.Foo'>
singleton <class '__main__.Foo'> has_instance, <__main__.Foo object at 0x7ff7f036ab8
singleton __call__: <class '__main__.BBB'>
BBB self init: <__main__.BBB object at 0x7ff7f036a670>
singleton __call__: <class '__main__.BBB'>
singleton <class '__main__.BBB'> has_instance, <__main__.BBB object at 0x7ff7f036a67
sssssss <class '__main__.Foo'>
show
show
```

上面代码的理解:

- 用Foo类型的metaclass Singleton创建Foo类型，调用singleton的**new**创建Foo， base默认为object， name为类型名，使用的就是父metaclass type的**new**.
- metaclass Singleton **init**初始化Foo类型, Foo类型创建完成.
- a = Foo() 调用metaclass Singleton的**call**方法， 来创建对应的Foo的实例对象。
- Singleton的**call** 会执行两个重要的动作， 调用对应Foo的**new**创建对应的的对象，然后调用对应的Foo的**init**初始化对应的Foo的实例，这样Foo得实例a就创建成功了。
- b = Foo() 由于Foo类型已经又对应的metaclass Singleton创建所以这里不会再创建，也就是Singleton的**new**和**init**不会执行。直接执行Singleton的**call**, 由于**call**中我们做的hook，之前存储的创建的Foo的实例已经存储直接返回， 就打到了Singleton的效果。
- c = BBB（ ） cc = BBB()逻辑同上.
- Foo.ssss() 可以调用是因为Foo作为类型是Singleton的实例， 对应的**class**指向Singleton，所以有singleton的方法ssss()。
- a.ssss() 报错，是因为Foo作为类型没有这个方法，Foo继承的是object, 导致a的方法映射不会出现ssss这个方法。 这里Foo有两面性， 一个是作为metaclass Singleton的实例，一个是作为a的父类.
- a有show*a()的方法是它的父类有showa()的方法， 这个方法是在Singleton在__init* 初始化Foo类型的时候 执行 cls.show_a = cls.show 赠予给Foo的.这里的cls都是Foo，右边是Foo作为Singleton的实

例， 左边也是Singleton这样做的目的 是 Foo建立作为a的父类能够将show_a -> show这样一个映射传递下去.

- 从上面 可以看到， 设计模式singleton在不同的语言的实现是有细节差异的
- metaclass这种定制类型的应用有内建的enum等， 也有应用级别如django中的orm model的应用.

about metaclass-type class class-instance

metaclass中的方法是**类型级别**的操作，不是class级别的.

python中类型就是我们所说的class也是一种metaclass的实例，大多数class的metaclass都是type. class的最初的base也就是所有class的起源都是从object这个class来的. class instance是对应的class实例化的.

类与类之间的关系是 is-kind-of的关系， 查看**bases**

类的实例和对应的类之间的关系 is-instance-of 查看**class**

三种类型的经典关系:

```
     metaclass       |      class            |     class instance
  |-- <type 'type'>  <------ <type 'object'>
  |     /\                    /\
  |---->|                     ||
        |                     ||
        ----------------- <class 'A'>    <-----------  a 'A instance'
```

其中:

---> 代表 is-instance-of

===> 代表 is-kind-of

class Foo的metaclass是Singleton，所以它是由这个metaclass类型创建的,是is-instance-of的关系， 查看它的**class**就是Singleton.

TODO: 关于为什么要搞出来一个type metaclass, 直接所有的class都从顶级的object发芽不行吗?

有很多原因， 如多个class共通的都抽到type中， 允许用户做些hook来定制class的行为，如enum staticmethod等等类型都是在metaclass的基础上搞出来的。 最主要的 可以动态创建类.

至于能否去掉，因为很多class的type都是type的实例，有待研究源码，目前源码是从class对应type给上搜索反向初始化的. 还有就是涉及研究 动态解释语言以及反射等。

关于类型创建, 可以参见type

```
class type(object)
class type(name, bases, dict)

class X:
    a = 1

X = type('X', (object,), dict(a=1))
```

> 参考
>
> - https://stackoverflow.com/questions/44178162/call-method-of-type-class
> - https://stackoverflow.com/questions/100003/what-are-metaclasses-in-python
> - https://realpython.com/python-metaclasses/
> - << Python 源码剖析 >> 作者 陈儒 章节 python虚拟机中的类机制
> - super 以及 多继承方法解析顺序 Method Resolution Order [MRO]

python timestamp with time_zone

created: 20240511 updated: 20240511 authors: alick

when system time zone is +08:00 Asia/Shanghai

example

```
from datetime import datetime, timezone
from zoneinfo import ZoneInfo

asia_shanghai_tz = ZoneInfo("Asia/Shanghai")
d = datetime(2024, 5, 11, 8, 0, 0)
d_with_tz = datetime(2024, 5, 11, 0, 0, 0, tzinfo=ZoneInfo("UTC"))
t = d.timestamp()
t2 = d_with_tz.timestamp()
print(f"d: {d} timestamp: {t}")
print(f"d_with_tz: {d_with_tz} timestamp: {t2}")
d1 = datetime.fromtimestamp(t)
d2 = datetime.fromtimestamp(t, timezone.utc)
d3 = datetime.fromtimestamp(t, asia_shanghai_tz)
print(f"from timestamp, datetime: {d1}, tzinfo: {d1.tzinfo}")
print(f"from utc tz timestamp, datetime: {d2}, tzinfo: {d2.tzinfo}")
print(f"from Asia/Shanghai tz timestamp, datetime: {d3}, tzinfo: {d3.tzinfo}")
```

result

```
d: 2024-05-11 08:00:00 timestamp: 1715385600.0
d_with_tz: 2024-05-11 00:00:00+00:00 timestamp: 1715385600.0
from timestamp, datetime: 2024-05-11 08:00:00, tzinfo: None
from utc tz timestamp, datetime: 2024-05-11 00:00:00+00:00, tzinfo: UTC
from Asia/Shanghai tz timestamp, datetime: 2024-05-11 08:00:00+08:00, tzinfo: Asia/S
```

conclusion

- note when tz is utc, fromtimestamp parse datetime is with tz and show as utc datetime
- when tz is Asia/Shanghai or is None (use default system time zone +08:00), it parse tiemstamp result is same as d `d = datetime(2024, 5, 11, 8, 0, 0)` . so when use datetime.fromtimestamp without param tz, is result is effect by system timezone
- but datetime.timestamp() is effect by system timezone when datetime is without tz, it use system tz. when datetime with tz, datetime.timestamp() is not effect by system tz.

python profile tools

created: 20230225 updated: 20230225 authors: alick97

some tools

> blog how-to-find-out-the-bottleneck-of-my-python-code
>
> - CProfile
>
>   ```
>   python -m cProfile scatter.py
>   python -m cProfile -o scatter.prof scatter.py
>   gprof2dot --colour-nodes-by-selftime -f pstats output.pstats | \
>     dot -Tpng -o output.png
>   ```
>
> - pyinstrument
> - py-spy

py-spy

test code

```
import time
CC = 1

def f1():
    global CC
    CC += 1
    time.sleep(1)

def f2():
    time.sleep(2)

def f3():
    global CC
    time.sleep(1)
    for i in range(100):
        CC += 1

def run():
    for i in range(2):
        f1()
        f2()
        f3()
    print(CC)

if __name__ == "__main__":
    run()
```

case for consider io, param with --idle

```
sudo py-spy record -o profile.svg --idle -- python3 t.py
```

case for just consider cpu, param without --idle

```
sudo py-spy record -o profile.svg   -- python3 t.py
```

python conda environment

created: 20250521 updated: 20250521 authors: alick97

---

cheatsheet

- exit

  ```
  conda deactivate
  ```

- find version

  ```
  conda env list
  conda run -n env1 python --version
  ```

- create env ``` conda create --name env2 python=3.10.4

```

- active env
```

conda activate env2

```

- has other err need
```

source $(conda info --base)/etc/profile.d/conda.sh

```
```

cxx language.

循环引用 泄露错误实例

```
#include <memory>
#include <string>
#include <iostream>

struct Foo {
    Foo(const char* s):name(s) { std::cout << name << " Foo...\n"; }
    ~Foo() { std::cout << name << " ~Foo...\n"; }

    std::shared_ptr<Foo> other;
    std::string name;
};


int main()
{
    std::shared_ptr<Foo> f1, f2;
    std::cout << "f1 count: " << f1.use_count() << std::endl;
    std::cout << "f2 count: " << f2.use_count() << std::endl;
    f1 = std::make_shared<Foo>("f1");
    // f2 = std::make_shared<Foo>("f2");
    f1->other = f1;
    // f2->other = f1;

    std::cout << "f1 count: " << f1.use_count() << std::endl;
    // std::cout << "f2 count: " << f2.use_count() << std::endl;
    return 0;
}
```

运行结果:

```
f1 count: 0
f2 count: 0
f1 Foo...
f1 count: 2
```

永远调用不到析构函数, 原因f1中other是自己， f1的总引用计数为2. main栈结束，f1引用计数为1，没变成0不会调用到真正的引用对象的析构函数。

解决方法，将结构体Foo中的other换成std::weak_ptr

```
struct Foo {
    Foo(const char* s):name(s) { std::cout << name << " Foo...\n"; }
    ~Foo() { std::cout << name << " ~Foo...\n"; }

    std::weak_ptr<Foo> other;
    std::string name;
};
```

weak_ptr 不增加引用计数， 不控制生命周期

其他例子：

> https://blog.csdn.net/jacketinsysu/article/details/53341370

vlc

created: 20220126 updated: 20220126 authors: alick97

official link

link

use vlc to capture desktop video

> version 3.0.16

- open vlc
- menu file -> open capture device
- in [file disc network capture] -> select [capture]
- input device -> select [screen]
- select [stream output]
- open [settings]
- file option input file path and select encapsulation method to MPEG 4 or other you like
- transcoding options click video select h256, bitrate 500kb/s or other you like
- select ok and click open, now it start to capture
- pause capture or close capture is to click pause or close button, then file will save.

> other link how-to-record-ubuntu-desktop-with-vlc

ffmpeg

created: 20220129 updated: 20220129 authors: alick97

convert audio

```
ffmpeg -i my.m4a -acodec mp3 -ac 2 -ab 192k my.mp3
```

video 压缩帧率

```
ffmpeg -i VID_20181222_204902.mp4 -b:v 2000k  VID_20181222_204902_1000k.mp4
```

- ffmpeg org link
- 使用gpu https://www.jianshu.com/p/59da3d350488

  ```
  ffmpeg -hwaccel cuvid -c:v h264_cuvid -noautorotate -i src.mp4 -c:v h264_nvenc -
  ```

  512k -> 300m film 1024k -> 600m film

plantuml

created: 20220131 updated: 20220202 authors: alick

---

a tool for uml in pure text

- url
- guide
- vscode plug: PlantUML
- inline into markdown code type: plantuml

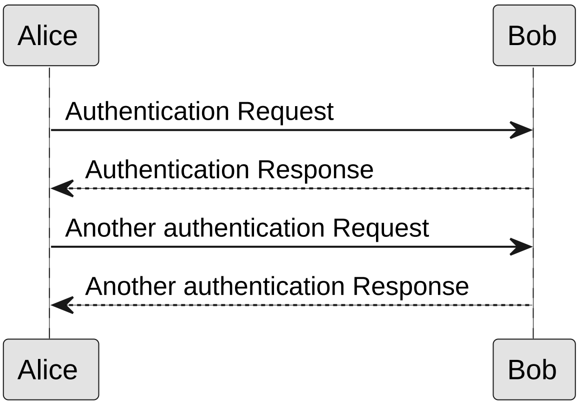
pre install

- some graph need install graphviz

gitbook plugin

> https://github.com/vowstar/gitbook-plugin-uml

demo


Alice
Bob
Authentication Request
Authentication Response
Another authentication Request
Another authentication Response
Alice
Bob

vpn server tailscale

created: 20230611 updated: 20230611 authors: alick97

Features

- Set up user private vpn server to all user machines.
- It is open source and has free plans.
- Easily setup and support amost all system.

link

official link

Shell directory jump tools [z]

created: 20220828 updated: 20220828 authors: alick97

- https://github.com/rupa/z

glow

created: 20230209 updated: 20230209 authors: alick97

---

preview markdown file in cli

link

goldendict

created: 20230225 updated: 20230225 authors: alick97

goldendict is a translation tools

- link
- Add offline dict, goldendict can use stardict files.

  Download link

  For chinese, i use these.

  ```
  /opt/stardict/dic/
  ├── stardict-cedict-gb-2.4.2
  ├── stardict-HanYuChengYuCiDian-new_colors-2.4.2
  ├── stardict-langdao-ce-gb-2.4.2
  ├── stardict-langdao-ec-gb-2.4.2
  ├── stardict-lazyworm-ec-2.4.2
  ├── stardict-oald-cn-2.4.2
  ├── stardict-oxford-gb-formated-2.4.2
  ├── stardict-powerword2011_1_901-2.4.2
  └── stardict-ProECCE-2.4.2
  ```

- Add plugin online translation translate-shell

In ubuntu, you can run below command:

```
sudo apt install translate-shell
```

Integration Config in Menu Edit -> Sources -> Dictionaries -> Programs

Type	Name	Command Line
Plain Text	trans-shell-zh	trans -x http://127.0.0.1:7890 -no-ansi -t zh %GDWORD%
Plain Text	trans-shell-en	trans -x http://127.0.0.1:7890 -no-ansi -t en %GDWORD%

- Or use in terminal, you can set alias

  ```
  alias s='_s() { goldendict "$1"; unset -f _s; }; _s'
  ```

- For searching word in other app, you can select word in other app and input `Ctrl + c + c`. Be sure of goldendict is opened firstly.

net diagnostic tool

created: 20231107 updated: 20231107 authors: alick

---

some net diagnostic tool

- mtr: combines of traceroute and ping
- nc
- ping
- traceroute
- dig: dns

net linux tunnel

created: 20240114 updated: 20240114 authors: alick97

---

link

tools link

liked tools

- frp
- Tailscale
- tunwg
- SirTunnel

most popular base component

- wiregruard

references

- https tls sni

cloudflare network tunnels

created: 20250602 updated: 20250602 authors: alick97

---

use case for export internal port to internet permanently

1. go to dash board https://dash.cloudflare.com/
2. click zerotrust -> networks -> tunnels create a tunnel
3. install cloudflared and run it.
4. config port mapping in dashboard.(you must have one public hostname)

eg:

```
# Add cloudflare gpg key
sudo mkdir -p --mode=0755 /usr/share/keyrings
curl -fsSL https://pkg.cloudflare.com/cloudflare-main.gpg | sudo tee /usr/share/keyr

# Add this repo to your apt repositories
echo 'deb [signed-by=/usr/share/keyrings/cloudflare-main.gpg] https://pkg.cloudflare

# install cloudflared
sudo apt-get update && sudo apt-get install cloudflared
```

After you have installed cloudflared on your machine, you can install a service to automatically run your tunnel whenever your machine starts:

```
sudo cloudflared service install [token]
```

OR run the tunnel manually in your current terminal session only:

```
cloudflared tunnel run --token [token]
```

use case for export internal port to internet temporarily

eg export port 8000

```
cloudflared tunnel --url http://localhost:8000
```

neovim lazyvim

created: 20240215 updated: 20240215 authors: alick97

---

easy to install for beginner

- lazyvim

database gui tools

created: 20240330 updated: 20240330 authors: alick

- for MySQL,MariaDB,PostgreSQL ... [win,mac,linux] https://dbeaver.io/download/

peek a gif recorder

created: 20240525 updated: 20240525 authors: alick97

- link

music

created: 20241116 updated: 20241116 authors: alick97

---

music free

- music free
- offcial plugin
- third plugin
- theme

midi tools

created: 20250120 updated: 20250120 authors: alick97

- LMMS is an open-source cross-platform digital audio workstation designed for music production

openssh

created: 20221023 updated: 20221023 authors: alick

---

in win7

基本操作

- link

  #### 修改配置

  连接拒绝 注释掉对应的配置 c:\ProgramData\ssh\sshd_config

  ```
  Match User
  Match Group
  ```

  ## 使用git bash

- need install git in windows before
- login cmd

  ```
  ssh alick@192.168.1.12 'C:\alick\software\Git\bin\bash.exe -i'
  ```

windows-config-hosts

created: 20240411 updated: 20240411 authors: alick

---

env win10

- run edit as admin, edit C:\Windows\System32\drivers\etc\hosts file
- add host config eg:

  ```
  192.168.31.158 a.local
  ```

- run in cmd, flush dns

  ```
  ipconfig/flushdns
  ```

web csrf and cookie

created: 20240214 updated: 20240214 authors: alick97

---

csrf

- csrf wiki

debug example code

file: ttt/t.html

```
<html>

<body>
    <iframe name="dummyframe" id="dummyframe"></iframe>
    <img src="http://1.local:8881/cc">
    <form id="myForm" action="http://1.local:8881/api/v1/messages" method="post">
        <ul>
          <li>
            <label for="name">Name:</label>
            <input type="text" id="name" name="user_name" />
          </li>
          <li>
            <label for="mail">Email:</label>
            <input type="email" id="mail" name="user_email" />
          </li>
          <li>
            <label for="msg">Message:</label>
            <textarea id="msg" name="user_message"></textarea>
          </li>
        </ul>
        <button type="submit">submit</button>
      </form>
</body>

<script defer>
function formSubmit(event) {
  var url = "http://1.local:8881/s1";
  var request = new XMLHttpRequest();
  request.open('POST', url, true);
  request.withCredentials = true;
  request.onload = function() {
    console.log(request.responseText);
  };
  request.onerror = function(e) {
    console.log(`err: ${e}`);
  };
  request.send(new FormData(event.target));
  event.preventDefault();
}

document.getElementById('myForm').addEventListener("submit", formSubmit);

</script>
</html>
```

run web site localhost:8888

```
python3 -m http.server --directory ttt 8888
```

config /etc/hosts

```
# local example 1
127.0.0.1 1.local
```

run web site 1.local:8881

```
 while :; do echo -e "HTTP/1.1 200 OK\r\nContent-Length:5\r\nAccess-Control-Allow-Or
```

conclusion

- Browser do more to prevent send with cookies of site B from site A to site B for preventing attack of csrf. Set cookie safe fields like SameSite.
- Most csrf attack need cookie. but csrf is defined that it is a action can without cookie. So if api method is GET but do things is modify and it is not design by rest style or auth without cookie, it will by attacked easily.
- For preventing csrf, add csrf token to your api or other methods(TODO).

reference

- https://www.ducktypelabs.com/csrf-tool/
- cookie SameSite 1
- mdn cookie SameSite

web certificate authority (CA)

created: 20240502 updated: 20240502 authors: alick97

---

get url ca chain

```
echo quit | openssl  s_client -showcerts -connect qq.com:443|less
```

extract info from pem

```
x509 -in DigiCert_Global_Root_CA.crt -pubkey  -text|less
```

get issue public key from pem

```
 x509 -pubkey -noout -in DigiCert_Global_Root_CA.crt |sed '1d;$d'| base64 -d| openss
SHA256(stdin)= aff988906dde12955d9bebbf928fdcc31cce328d5b9384f21c8941ca26e20391
```

Reference

> https://en.wikipedia.org/wiki/X.509
>
> https://en.wikipedia.org/wiki/Public_key_certificate
>
> https://developer.mozilla.org/en-US/docs/Web/Security/Transport_Layer_Security
>
> https://daniel.haxx.se/blog/2018/11/07/get-the-ca-cert-for-curl/

mysql timestamp field with time_zone

created: 20240511 updated: 20240511 authors: alick

---

get mysql server timezone

- sql

  ```
  SELECT @@session.time_zone;
  ```

- result

  ```
  SYSTEM
  ```

- sql

  ```
  select timediff(now(),convert_tz(now(),@@session.time_zone,'+00:00'));
  ```

- result mean it's timezone is utc

  ```
  00:00:00
  ```

create example table

```

create table table_1


CREATE TABLE `table_1` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `name` varchar(20) NOT NULL,
  `create_time` timestamp NULL DEFAULT current_timestamp(),
  PRIMARY KEY (`id`)
)

insert into table_1 (id, name) values
(1, 'n1'),
(2, 'n2')

select * from table_1
```

result

```
"id","name","create_time"
1,n1,2024-05-10 16:39:39.000
2,n2,2024-05-10 16:39:39.000
```

timezone effect timestamp result read

- change timezone to +08:00

```
set time_zone = '+08:00';
select * from table_1
```

- result

```
"id","name","create_time"
1,n1,2024-05-11 00:39:39.000
2,n2,2024-05-11 00:39:39.000
```

- conclusion

```
result is effect by timezone, it parse timestamp to related timezone
```

timezone effect timestamp field write

- write datetime when tz is +08:00

```
set time_zone = '+08:00';
update table_1 set create_time = '2024-05-10 16:39:39' where id = 1
select * from table_1
```

- result

  ```
  "id","name","create_time"
  1,n1,2024-05-10 16:39:39.000
  2,n2,2024-05-11 00:39:39.000
  ```

- see result is 16:39:39 +08:00, so when insert time '2024-05-10 16:39:39' when session timezone is '+08:00', it actually mean use insert iso datetime '2024-05-10T16:39:39.000+08:00', when change timezone to '+00:00', it is not you need 16:39:39, it is wrong time 08:39.39, less than 8 hours.

```
set time_zone = '+00:00';
select * from table_1
```

```
"id","name","create_time"
1,n1,2024-05-10 08:39:39.000
2,n2,2024-05-10 16:39:39.000
```

timezone effect timestamp field filter， if you want filter by iso format str

- use +8:00 time 2024-05-11 00:39:39.000 filter when db time zone is utc

  ```
  set time_zone = '+00:00';
  select * from table_1 where create_time = CONVERT_TZ('2024-05-11 00:39:39.000','
  ```

- result

  ```
  "id","name","create_time"
  2,n2,2024-05-10 16:39:39.000
  ```

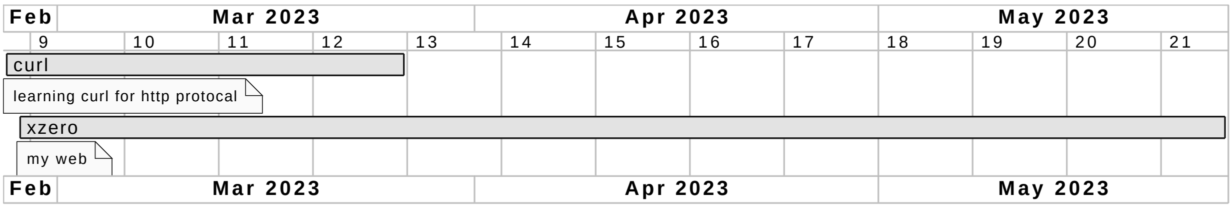
more in python

- when use pymysql is parse timestamp field data as str like '2024-05-10 16:39:39.000' when write or read with session time_zone.
- when session time zone is not same as program time zone, it may cause issue. eg: db time_zone is +00： 00， python program timezone is +08:00. read record 2, it get create_time is '2024-05-10 16:39:39.000' -> python `d1=datetime(2024, 05, 10, 16, 39, 39, tzinfo=None)` , when use d1.timestamp(), it effect python program timezone +08:00, this cause wrong result.
- so it is best when timezone of python program and timezone of db are same.
- timestamp in python link

plan-2023

created: 20230225 updated: 20230225 authors: alick97

plan


Feb
Mar 2023
Apr 2023
May 2023
curl
learning curl for http protocal
xzero
my web
Feb
Mar 2023
Apr 2023
May 2023