pwn入门

1.安装vmware-station

2.安装ubuntu

参见网址：

1. VMware Workstation Pro v16.2.0 官方完整版(附永久激活密钥) | USB迷 (usbmi.com)

   内附16.0版本的激活密匙：

   VM16：ZF3R0-FHED2-M80TY-8QYGC-NPKYF

2. https://blog.csdn.net/qq_42566274/article/details/112272767

   具体关于上面两个的安装教程

3.安装pip

参见网址：

1. 技术|如何在 Ubuntu 上安装 pip (linux.cn)
2. 如何在 Ubuntu 20.04 上安装 Python Pip - 知乎 (zhihu.com)
3. 详见B站上的网课

4.安装pwntool

5.安装pwndbg

需要先安装git，输入如下命令：

```
sudo apt install git
```

而后在github上搜索pwndbg，按照下面的指令安装即可

6.安装one_gadget

下载one_gadget首先需要下载配置ruby，gem，用github上的命令即可:

下载ruby，gem时注意翻墙，否则会导致无法下载完全；同时注意如果还是无法下载完全，用sudo高权限：sudo apt-get update升级

后使用：sudo gem-install one_gadget命令安装即可

7.安装libcsearcher

目前不保证其正确性。。。

也是在github上查找libcsearcher，然后按照命令操作即可

8.安装main_arena_offset

同样在github上查找main_arena_offset并按照命令操作即可

9.安装wine

> ## 最简单的方法
>
> https://wiki.winehq.org/Ubuntu_zhcn
> 按照上述网站的步骤安装即可，很简单的呜呜呜。。。

在Ubuntu系统下，你可以使用官方的PPA方便的安装最新的Wine开发版本。打开一个终端并使用sudo权限执行下列命令。

复制代码

代码如下:

```
$ sudo add-apt-repository ppa:ubuntu-wine/ppa
```

但是在上述过程中出现如下报错：

```
E: 仓库 “http://ppa.launchpad.net/ubuntu-wine/ppa/ubuntu focal Release” 没有 Release 文件。
N: 无法安全地用该源进行更新，所以默认禁用该源。
N: 参见 apt-secure(8) 手册以了解仓库创建和用户配置方面的细节。
```

执行第二句 sudo-apt-get update报下面的错误：

仓库 “http://ppa.launchpad.net/chris-lea/node.js/ubuntu focal Release” 没有 Release 文件

解决办法，删除chris-lea/node.js 这个ppa文件，删除命令

```
sudo add-apt-repository --remove ppa:/chris-lea/node.js
```

然后重新执行 sudo apt-get update ，更新成功。

注意，“--remove ppa:/”后面跟提示错误的文件名，如果错误是 仓库 “http://ppa.launchpad.net/webupd8team/sublime-text-3/ubuntu focal Release” 没有 Release 文件，

那么删除命令要下面这样写：

```
sudo add-apt-repository --remove ppa:/webupd8team/sublime-text-3
```

需要根据出错的文件名来写命令。

```
$ sudo apt-get update
```

```
$ sudo apt-get install wine 1.7 winetricks
```

但是又出现有进程在占用导致无法进行

```
zrz@zrz-virtual-machine:~/py_main_arena_offset$ sudo apt-get install wine 1.7 winetricks
E: Could not get lock /var/lib/dpkg/lock-frontend. It is held by process 27171 (unattended-upgr)
N: Be aware that removing the lock file is not a solution and may break your system.
E: 无法获取 dpkg 前端锁 (/var/lib/dpkg/lock-frontend)，是否有其他进程正占用它？
```

采取如下操作：

查看进程：

```
ps -aux
```

使用sudo权限杀死妨碍继续进行的进程

```
sudo kill -9 (+进程编号27171)
```

然后在使用上述的命令即可

但是还有些无法下载，进而考虑用如下的方法：

1.安装前准备

安装必要的工具及deepin-wine依赖

```
sudo apt install wget g++ git      #如已安装可自行跳过
```

2.安装deepin-wine

```
git clone "https://gitee.com/wszqkzqk/deepin-wine-for-ubuntu.git"
cd deepin-wine                     #切换到下载目录
sudo ./install.sh                  #执行安装
```

deepin-wine容器安装完成，下面进行具体软件的安装。

(25条消息) ubuntu安装wine_Ubuntu 20.04 安装微信、QQ等_weixin_39812142的博客-CSDN博客

但是deepin-wine并没有很好的适配我的ubuntu，甚至是无法运行.exe文件（提示.EXE格式无效的报错）

所以采用以下方法安装wine3.0

https://blog.csdn.net/aicyo8644/article/details/102063895?spm=1001.2101.3001.6650.9&utm_medium=distribute.pc_relevant.none-task-blog-2%7Edefault%7ECTRLIST%7Edefault-9.pc_relevant_default&depth_1-utm_source=distribute.pc_relevant.none-task-blog-2%7Edefault%7ECTRLIST%7Edefault-9.pc_relevant_default&utm_relevant_index=15

https://blog.csdn.net/qq_26056015/article/details/83061737

https://blog.csdn.net/weixin_30381317/article/details/97330810

http://blog.sina.com.cn/s/blog_55465b470100sc40.html

1.一种获得Wine最新稳定版本（即现在的3.0版本）的方法是使用以下命令从源码包构建wine。

```
$ wget  https://dl.winehq.org/wine/source/3.0/wine-3.0.tar.xz
$ tar -xvf wine-3.0.tar.xz
$ cd wine-3.0/
$ sudo ./configure
$ sudo ./configure --enable-win64  [对于64位平台]
$ sudo make && sudo make install
```

2.除此之外还可安装稳定的wine6.0.2（推荐采用这个）

1.第一步是安装原生的Wine6.0.2稳定版的可执行程序。

```
1、首先在系统上启用32位支持，运行命令：sudo dpkg --add-architecture i386
2、添加正式的Wine仓库密钥：
wget -nc https://dl.winehq.org/wine-builds/winehq.key
sudo apt-key add winehq.key
3、添加Wine仓库本身，不同的ubuntu版本是不同的执行命令，可以直接到官网查看，下面是ubuntu 20.04及其衍生版的：
sudo add-apt-repository 'deb https://dl.winehq.org/wine-builds/ubuntu/ focal main'
sudo apt-get update
4、安装Wine 6.0稳定版（其他版本请查看官网的相应命令）：
sudo apt install --install-recommends winehq-stable
以上步骤执行完成后在/opt目录中就增加了wine-stable目录，在/usr/bin目录中wine可执行脚本指向这个目录中的可执行文件。
```

2.第二步是下载原生的Wine6.0.2稳定版的源代码并进行编译并打补丁。

打补丁参见：

tps://blog.csdn.net/ericden/article/details/121818332

补丁为：（注意：在打补丁时按照文中所说将下面的补丁替换到/opt/wine-stable里的相关路径就可以了，不需要单独make文中说的那一部分，否则后面整体make会报错！！！）

别打了，打了就寄了！！！

```
链接：https://pan.baidu.com/s/1NA_7oCEYe6xafNz9btdgGQ
提取码：p39i
```

```
1、在wine官网上下载源代码压缩包，下载地址为：
https://dl.winehq.org/wine/source/6.0/wine-6.0.2.tar.xz
2、将该压缩包解压至用户主目录中，所有文件位于wine-6.0.2中。
3、进入wine-6.0.2目录中，执行./congfigure
```

但是在进行configure操作时遇到文件缺失的问题

错误信息如下：

```
configure: error: X 64-bit development files not found. Wine will be built
without X support, which probably isn't what you want. You will need
to install 64-bit development packages of Xlib at the very least.
Use the --without-x option if you really want this.
```

缺少依赖，解决方法如下

```
sudo apt install xserver-xorg-dev
```

```
1、./configure时出现Cannot build a 32-bit program, you need to install 32-bit
development libraries.

解决：sudo apt-get install gcc-multilib g++-multilib

2、error: X 32-bit development files not found. Wine will be built
without X support, which probably isn't what you want. You will need
to install 32-bit development packages of Xlib at the very least.
Use the --without-x option if you really want this.

解决：sudo apt-get install libx11-dev:i386

3、configure: error: FreeType 32-bit development files not found

解决：sudo apt-get install libfreetype6-dev:i386 libfreetype6-dev

如果已安装，sudo ln -s /usr/include/freetype2 /usr/include/freetype

4、其他库

sudo apt-get install libxrender-dev:i386 libgnults-dev:i386
```

在依赖项全部安装后，系统提示make，但这时不要make，先执行第五条指令enable-win64

但是这里报如下错误：

https://blog.csdn.net/weixin_43242942/article/details/89563534

```
configure: error: X 64-bit development files not found. Wine will be built
without X support, which probably isn't what you want. You will need
to install 64-bit development packages of Xlib at the very least.
Use the --without-x option if you really want this.
```

、

我们需使用如下命令补全缺失文件：

```
sudo apt install xserver-xorg-dev
```

3.再次执行第五条指令enable-win64，发现会提示有很多依赖库未安装，故使用：

```
sudo apt-get install ***
```

具体的文件名称参见如下官方提示依赖库网址：

https://wiki.winehq.org/Building_Wine

Library name	Debian(***)	Fedora	Arch	Function	Notes
Generally necessary					
MinGW cross-compiler	gcc-mingw-w64	mingw32-gcc, mingw64-gcc	mingw-w64-gcc	PE format DLLs	
ALSA	libasound2-dev	alsa-devel	alsa-lib	Sound backend	At least one is necessary for sound.
PulseAudio	libpulse-dev	libpulse-devel	libpulse		
libdbus	libdbus-1-dev	dbus-libs	dbus	Dynamic device detection (specifically, mass storage)	Removable drives may be incorrectly detected otherwise.
libfontconfig	libfontconfig-dev	fontconfig-devel	fontconfig	Host font enumeration	Install if you want host fonts to be detected.
libfreetype	libfreetype-dev	freetype-devel	freetype2	FreeType font reading	
libgnutls	libgnutls28-dev	gnutls-devel	gnutls	Cryptography	
libinotify	N/A	N/A	N/A	File change notification	Only necessary for some platforms (Linux does not need this.)
libjpeg	libjpeg62-turbo-dev	libjpeg-turbo-devel	libjpeg-turbo	Image format decoding	
libpng	libpng-dev	libpng-devel	libpng		
libtiff	libtiff-dev	libtiff-devel	libtiff		
OpenGL	libgl-dev	mesa-libGL-devel	mesa	Hardware-accelerated/3D graphics	
libunwind	libunwind-dev	libunwind-devel	libunwind	Exception unwinding	Necessary for x86_64 and arm64, but not used on other platforms.
libX*				Window management	

Library name	Debian(***)	Fedora	Arch	Function	Notes
libxml, libxslt	libxml2-dev, libxslt1-dev	libxml2-devel, libxslt-devel	libxml2, libxslt	XML parsing	
Needed for many applications					
libfaudio	libfaudio-dev	libFAudio-devel	faudio	XAudio implementation	Needed for audio in some newer applications, especially games. (XAudio was initially released in 2008.)
libgstreamer	libgstreamer1.0-dev, libgstreamer-plugins-base1.0-dev	gstreamer1-devel, gstreamer1-plugins-base-devel	gstreamer, gst-plugins-base-libs	Multimedia playback	Generally necessary for games or applications that play back audio or video files.
libmpg123	libmpg123-dev	mpg123-devel	mpg123	mp3 decoding	Generally necessary for games or applications that play back audio files.
OSMesa	libosmesa6-dev	libOSMesa-devel	mesa	OpenGL bitmap support	
libSDL2	libsdl2-dev	SDL2-devel	sdl2	HID joystick support	Generally necessary for joystick or other HID support. Only one library is necessary, but they may exhibit different behaviour.
libudev	libudev-dev	?	systemd		
libvkd3d	libvkd3d-dev	libvkd3d-devel	vkd3d	Direct3D 12	Needed for some games. (Direct3D 12 was released in 2016.)

Library name	Debian(***)	Fedora	Arch	Function	Notes
Vulkan	libvulkan-dev	vulkan-headers, libvulkan-loader	vulkan-icd-loader, vulkan-headers	Hardware-accelerated/3D graphics	Necessary for some games; only supported by some video cards.
Rare or domain-specific					
libcapi20	libcapi20-dev	(none)	(none)	ISDN/telephony	Install only if you're using ISDN software.
liblcms	liblcms2-dev	lcms2-devel	lcms2	Color management	Rarely needed.
libcups	libcups2-dev	cups-devel	libcups	Printing	Install only if you need printer support.
libgphoto2	libgphoto2-dev	libgphoto2-devel	libgphoto2	Scanner/still image	Install only if you're using scanner/still image software.
libsane	libsane-dev	sane-backends-devel	sane		
libgsm	libgsm1-dev	gsm-devel	gsm	GSM audio codec	Very rarely needed, and generally only in older software.
Kerberos	libkrb5-dev	krb5-devel	krb5	Kerberos authentication	Install only if you're connecting via Kerberos.
LDAP	libldap2-dev	openldap-devel	libldap	LDAP remote directory protocol	Install only if you're using remote directories.
libnetapi	samba-dev	samba-devel	smbclient	Networking	Rarely needed.
OpenCL	ocl-icd-opencl-dev	ocl-icd-devel	ocl-icd	Parallel computing / GPGPU	Install if you're using parallel computing or GPGPU software.

Library name	Debian(***)	Fedora	Arch	Function	Notes
libpcap	libpcap-dev	libpcap-devel	libpcap	Packet capture	Install if you are using applications that require packet capture. (This replaces native wpcap.dll shipped by applications.)
libusb	libusb-1.0-0-dev	libusbx-devel	libusb	USB device support	Install only if you're using an application that accesses a USB device directly.
libv4l2	libv4l-dev	libv4l-devel	v4l-utils	Video capture	Install only if you're capturing video.
Never necessary					
libhal	(none)	(none)	(none)	Dynamic device detection	Obsolete; use libdbus instead.
OpenAL	libopenal-dev	openal-soft-devel	openal	Audio engine	Should never be needed. (This replaces native openal32 shipped by applications.)

注意可能会提示下面的错误：

```
configure: libxcomposite 64-bit development files not found, Xcomposite won't be supported.
```

执行下面的命令：

```
sudo apt-get install libxcomposite-dev
```

里面的x不是大写！！！

https://launchpad.net/ubuntu/+source/libxcomposite/1:0.4.5-1build1

4.但是按照上面将网站上提示的全部依赖项都安装后，仍会报如下的问题：

```
configure: libhal 64-bit development files not found, no legacy dynamic device support.
configure: OSS sound system found but too old (OSSv4 needed), OSS won't be supported.
configure: libFAudio 64-bit development files not found, XAudio2 won't be supported.

configure: Finished.  Do 'make' to compile Wine.
```

用下面网站的提示完成这些项的修复：

第一项：

From the same WineHQ Wiki page:
"libhal Dynamic device detection Obsolete; use libdbus instead."
So you can skip that one

所以可以不管；

第二，三项：

均与声卡相关，在这里和我们想要的东西没太大关系，所以查阅以下网站可以略过。

https://forum.winehq.org/viewtopic.php?f=8&p=130303

https://www.cnblogs.com/itholidaycn/p/6259798.html

https://blog.csdn.net/chujiu_/article/details/112767390

5.而后再make

10.安装IDA

注意在使用ubuntu时，解压rar文件一定要检查linux系统中是否sudo apt-get install 了rar和unrar，否则解压会导致大量空间占用！！！！！！！

这里用到了常用的命令去查看空间占用情况：

```
df -ha 或 df -h
```

下面的这个命令可以图形化地查看各个磁盘空间占用的情况：

```
sudo baobab
```

11.linux的.desktop文件入门

(26条消息) Linux下Desktop文件入门*YiferHuang 的博客-CSDN博客*desktop文件

在Linux下为软件程序添加“快捷方式”

> Desktop Entry文件是Linux桌面系统中用于描述程序启动配置信息的文件，它以.desktop为后缀名，相当于Windows系统下的桌面快捷方式。通常一个二进制可执行程序是一个没有后缀没有图标的文件，不可以随意移动。

因此很多Linux发行版都提供了启动器，便于集中管理应用程序。启动器本质是一个位于/usr/share/applications/路径下的目录。启动器目录中存放着很多.desktop文件，每个.desktop文件都是一个应用程序的入口，并且.desktop文件可以显示图标，对用户更加友好。

desktop文件基本模板

以demo.desktop为例

1. [Desktop Entry]
2. Name=<应用程序名>
3. Type=Application
4. Exec=<应用程序完整路径>
5. Icon=<应用程序图标的完整路径>

Name: desktop 文件最终显示的名称（一定要注意和 desktop 文件名的区别）
Type: 用于指定 desktop 文件的类型（包括 3 种类型：Application、Link、Directory)
Exec: 用于指定二进制可执行程序的完整路径
Icon: 指定应用程序图标的完整路径(可以省略后缀名)。图标支持 png 格式、svg 格式等，图标的推荐尺寸为 128x128。

拓展：针对linux系统上用wine启动的windows可执行程序的快捷方式

其格式如下：

1. [Desktop Entry]
2. Name=**<自己设定的应用程序名>**
3. Exec=env WINEPREFIX=**<"本机wine的地址" `$2`，eg："/home/zrz/.wine">** wine **<直接要wine运行的windows可执行文件，eg：ida.exe>**
4. Icon=**<应用程序图标的完整路径 `$1` >**
5. Type=Application
6. StartupNotify=true
7. Path=**<上面你要创建快捷方式的可执行文件的上一层目录的完整路径 `$1`，eg：/home/zrz/pwn_software/IDA>**
8. StartupWMClass=ida.exe

- **$1：那么如何知道其完整路径?**

  可鼠标点击至你想得知其完整路径的目录，用终端（命令行）打开并使用pwd即可！

- **$2：wine的地址怎么找?**

  利用如下命令：

  ```
  sudo nautilus ~/.local/share/applications
  ```

找到这些.desktop文件的位置，可以进入wine的目录查看用wine执行的程序使用wine的地址，便可以照猫画虎的知道我们需要wine的地址。