

# 嵌入式系統建置報告

---

FTP SERVER — vsftpd

組員：李承謙、詹博丞、許儒怡、黃煜翔、陳燮涵

# vsftpd介紹

## 1.簡介

vsftpd 是"very secure FTP daemon"的縮寫，安全性是它的一個最大的特點。

vsftpd 是一個 UNIX 類操作系統上運行的服務器的名字，它可以運行在諸如 Linux、BSD、Solaris、 HP-UNIX等系統上面，是一個完全免費的、開放源代碼的ftp服務器軟體，支持很多其他的 FTP 服務器所不支持的特徵。

比如：非常高的安全性需求、帶寬限制、良好的可伸縮性、可創建虛擬用戶、支持IPv6、速率高等。

## 2.特點

①vsftpd 是以一般身份啟動服務，所以對於 Linux 系統的使用權限較低，對於Linux 系統的危害就相對的減低了。

此外， vsftpd 亦利用 chroot() 這個函式進行改換根目錄的動作，使得系統工具不會被vsftpd 這支服務所誤用。

②任何需要具有較高執行權限的 vsftpd 指令均以一支特殊的上層程式( parent process ) 所控制，該上層程式享有的較高執行權限功能已經被限制的相當的低，並以不影響Linux 本身的系統為準；

③所有來自 clients 端，想要使用這支上層程式所提供的較高執行權限之vsftpd 指令的需求，均被視為『不可信任的要求』來處理，必需要經過相當程度的身份確認後，方可利用該上層程式的功能。例如chown(), Login 的要求等等動作；

④此外，上面提到的上層程式中，依然使用 chroot() 的功能來限制使用者的執行權限

# 目標功能

由於vsftpd有著非常豐富和龐大的功能，例如匿名登錄等，還有著很多周邊的輔助構建，例如pam來進行一個login的管理等。所以我們沒有實現全部的功能，我們最終的實現結果是能讓我們自己假設的linux系統能夠運行vsftpd，然後能夠讓外部的device通過ftp用賬號密碼的形式來登錄。并且能夠瀏覽、上傳、下載文件。

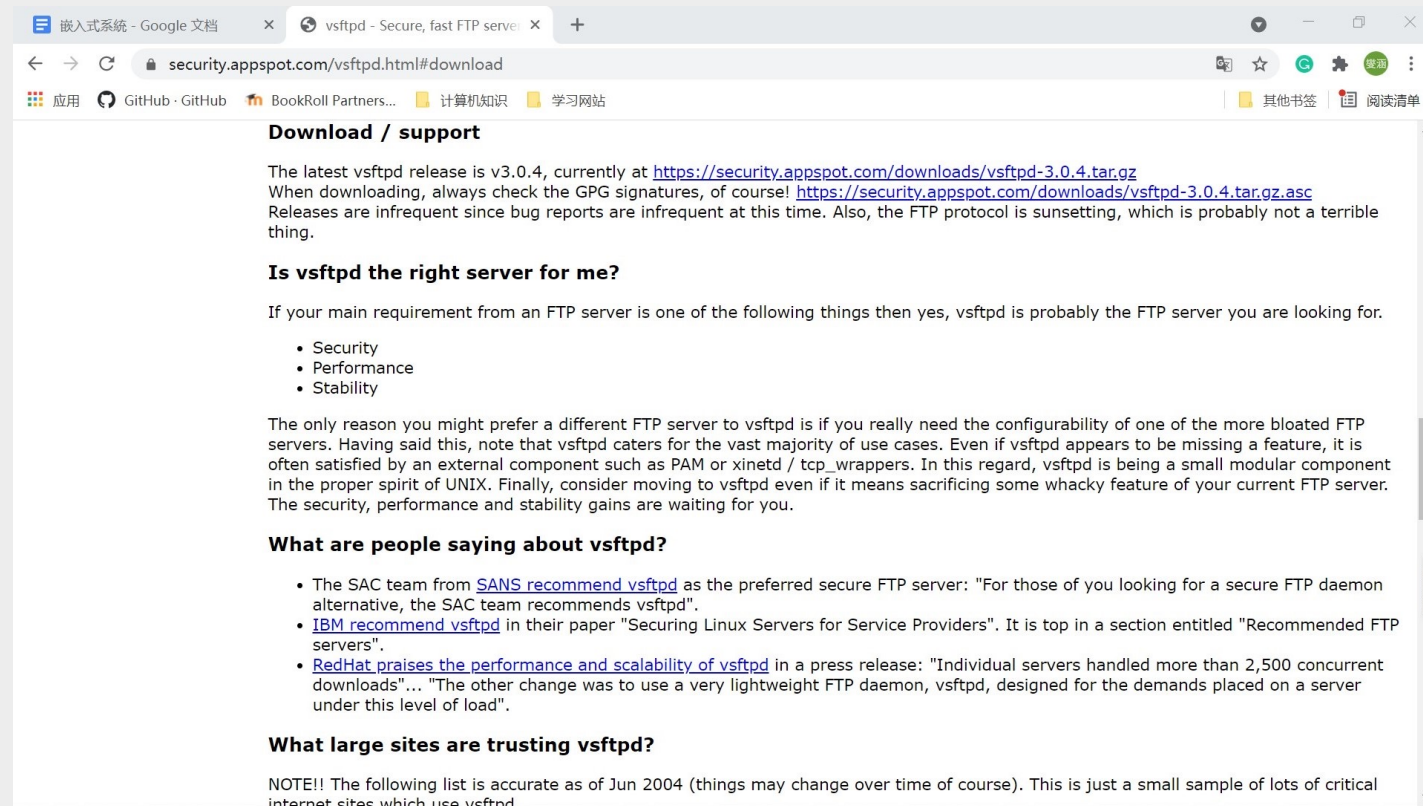
當然本次組建過程能夠成功的前提是能夠提前構建好自己的小型linux系統并且經過設置擁有自己的賬號密碼以及能夠對外聯網。

# 第一步：從網路上下載源碼

爲了在我們自己的linux系統上架設vsftp，我們首先需要從網路上下載vsftpd的源碼壓縮檔，放到我們的~/rootfs/src文件夾中。

其下載的網址為：

<https://security.appspot.com/vsftpd.html>



## 第二步：解壓縮下載好的檔案

現在我們已經下載好我們的源碼檔案：

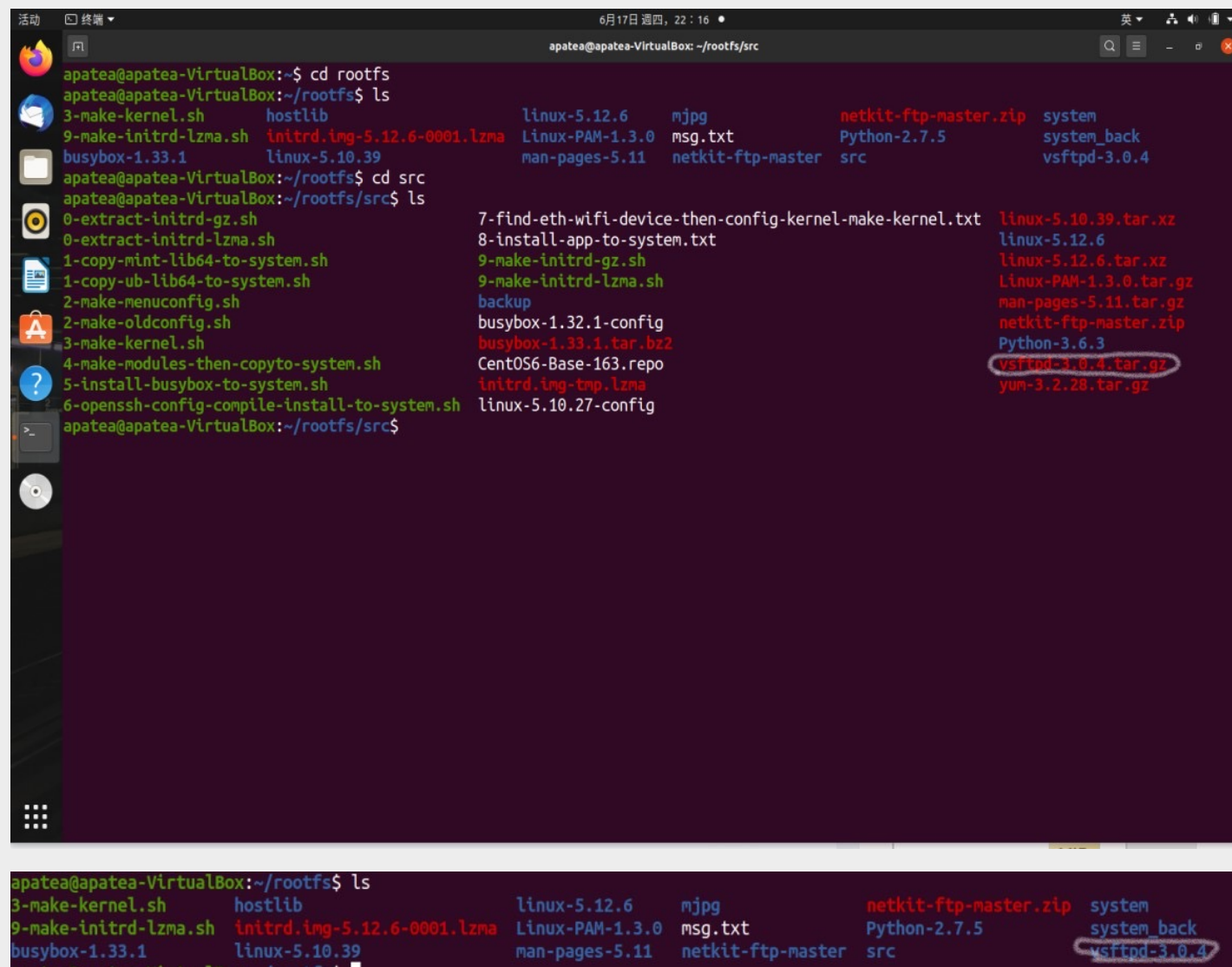
**vsftpd-3.0.4.tar.gz**

接下來我們使用

**tar -zxvf vsftpd-3.0.4.tar.gz**

的方式來對檔案進行解壓縮，解壓縮后的檔案夾如圖為：

**vsftpd-3.0.4**



```
apatea@apatea-VirtualBox:~$ cd rootfs
apatea@apatea-VirtualBox:~/rootfs$ ls
3-make-kernel.sh      hostlib               linux-5.12.6          mjpg                 netkit-ftp-master.zip  system
9-make-initrd-lzma.sh initrd.img-5.12.6-0001.lzma Linux-PAM-1.3.0      msg.txt              Python-2.7.5           system_back
busybox-1.33.1         linux-5.10.39         man-pages-5.11       netkit-ftp-master    src                     vsftpd-3.0.4
apatea@apatea-VirtualBox:~/rootfs$ cd src
apatea@apatea-VirtualBox:~/rootfs/src$ ls
0-extract-initrd-gz.sh  7-find-eth-wifi-device-then-config-kernel-make-kernel.txt  linux-5.10.39.tar.xz
0-extract-initrd-lzma.sh 8-install-app-to-system.txt                                linux-5.12.6
1-copy-mint-lib64-to-system.sh  9-make-initrd-gz.sh                                         linux-5.12.6.tar.xz
1-copy-ub-lib64-to-system.sh   9-make-initrd-lzma.sh                                       Linux-PAM-1.3.0.tar.gz
2-make-menuconfig.sh          backup                                                       man-pages-5.11.tar.gz
2-make-oldconfig.sh           busybox-1.32.1-config                                       netkit-ftp-master.zip
3-make-kernel.sh              busybox-1.33.1.tar.bz2                                     Python-3.6.3
4-make-modules-then-copyto-system.sh CentOS6-Base-163.repo                                       vsftpd-3.0.4.tar.gz
5-install-busybox-to-system.sh  initrd.img-tmp.lzma                                         yum-3.2.28.tar.gz
6-openssh-config-compile-install-to-system.sh linux-5.10.27-config
apatea@apatea-VirtualBox:~/rootfs/src$
```

```
apatea@apatea-VirtualBox:~/rootfs$ ls
3-make-kernel.sh      hostlib               linux-5.12.6          mjpg                 netkit-ftp-master.zip  system
9-make-initrd-lzma.sh initrd.img-5.12.6-0001.lzma Linux-PAM-1.3.0      msg.txt              Python-2.7.5           system_back
busybox-1.33.1         linux-5.10.39         man-pages-5.11       netkit-ftp-master    src                     vsftpd-3.0.4
```

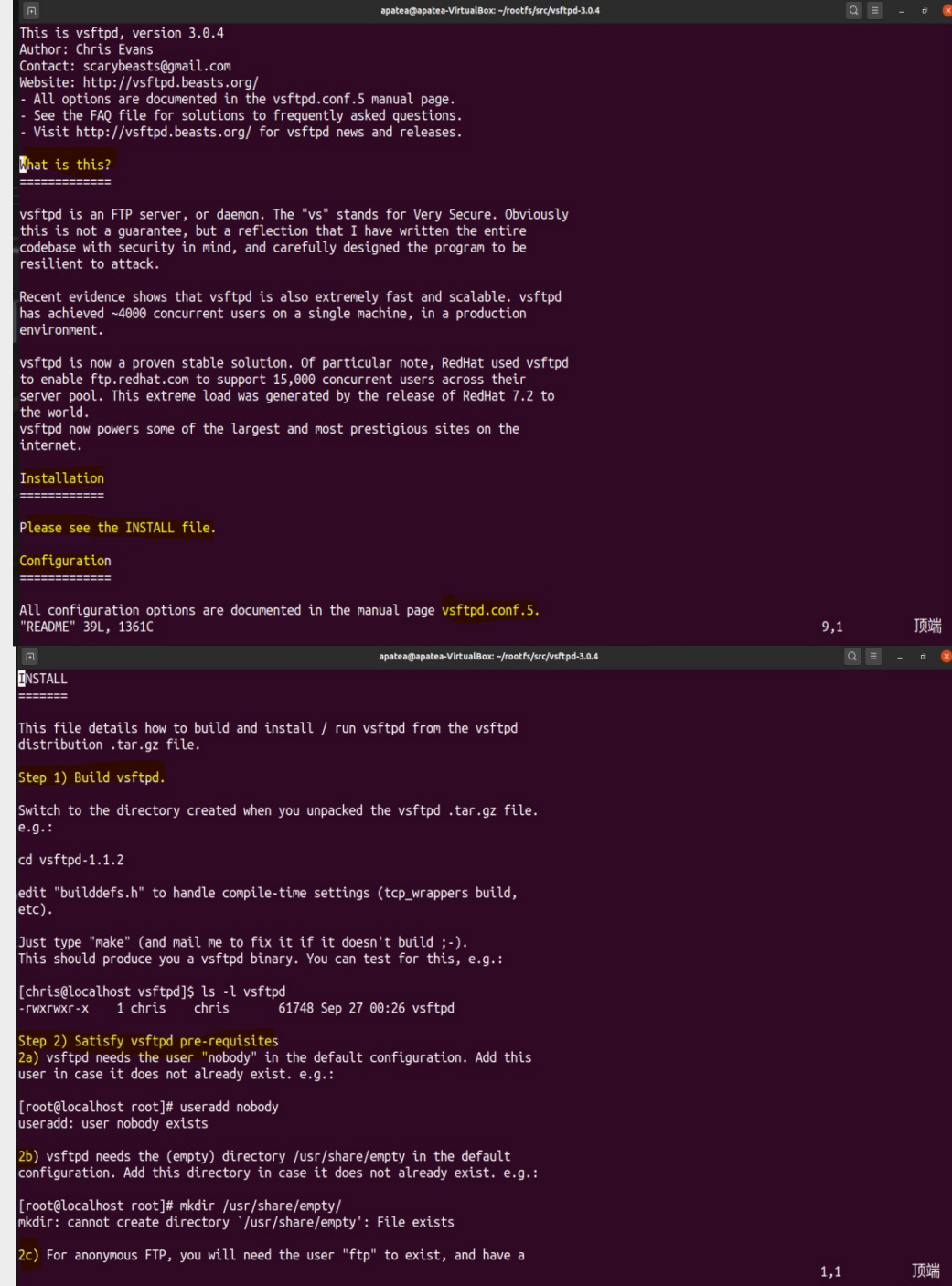
# 第三步：查看 README文件來確 認安裝流程和需要

其實如果你進行過很多次這種源碼下載後自己解壓縮和安裝的操作，你會很自然的去打開解壓縮后檔案夾中的README文件去瞭解這個構建的安裝和設定方式以及需求。

例如在vsftpd中，README指引我們去瀏覽INSTALL文件來獲取進一步的安裝下載信息。

在INSTALL文件中我們可以看到整個假設所需要的流程以及prerequisite（需求）。我們需要根據他的指示一步一步的來進行配置和安裝。

當然其中的內容是非常的豐富的，而且很多不同的設置和流程對應了不同的功能。然而由於我們只是要將其放到我們自己假設的小系統中，不需要那麼全面的功能，我就只執行了小部分的prerequisite，如果想要執行更多的功能建議仔細查閱並執行。



```
apatea@apatea-VirtualBox: ~/rootfs/src/vsftpd-3.0.4
This is vsftpd, version 3.0.4
Author: Chris Evans
Contact: scarybeasts@gmail.com
Website: http://vsftpd.beasts.org/
- All options are documented in the vsftpd.conf.5 manual page.
- See the FAQ file for solutions to frequently asked questions.
- Visit http://vsftpd.beasts.org/ for vsftpd news and releases.

What is this?
=====

vsftpd is an FTP server, or daemon. The "vs" stands for Very Secure. Obviously
this is not a guarantee, but a reflection that I have written the entire
codebase with security in mind, and carefully designed the program to be
resilient to attack.

Recent evidence shows that vsftpd is also extremely fast and scalable. vsftpd
has achieved ~4000 concurrent users on a single machine, in a production
environment.

vsftpd is now a proven stable solution. Of particular note, RedHat used vsftpd
to enable ftp.redhat.com to support 15,000 concurrent users across their
server pool. This extreme load was generated by the release of RedHat 7.2 to
the world.
vsftpd now powers some of the largest and most prestigious sites on the
internet.

Installation
=====

Please see the INSTALL file.

Configuration
=====

All configuration options are documented in the manual page vsftpd.conf.5.
"README" 39L, 1361C
9,1 顶端

apatea@apatea-VirtualBox: ~/rootfs/src/vsftpd-3.0.4
INSTALL
=====

This file details how to build and install / run vsftpd from the vsftpd
distribution .tar.gz file.

Step 1) Build vsftpd.

Switch to the directory created when you unpacked the vsftpd .tar.gz file.
e.g.:

cd vsftpd-1.1.2

edit "builddefs.h" to handle compile-time settings (tcp_wrappers build,
etc).

Just type "make" (and mail me to fix it if it doesn't build ;-).
This should produce you a vsftpd binary. You can test for this, e.g.:

[chris@localhost vsftpd]$ ls -l vsftpd
-rwxrwxr-x  1 chris  chris    61748 Sep 27 00:26 vsftpd

Step 2) Satisfy vsftpd pre-requisites
2a) vsftpd needs the user "nobody" in the default configuration. Add this
user in case it does not already exist. e.g.:

[root@localhost root]# useradd nobody
useradd: user nobody exists

2b) vsftpd needs the (empty) directory /usr/share/empty in the default
configuration. Add this directory in case it does not already exist. e.g.:

[root@localhost root]# mkdir /usr/share/empty/
mkdir: cannot create directory '/usr/share/empty': File exists

2c) For anonymous FTP, you will need the user "ftp" to exist, and have a
1,1 顶端
```



## 第四步：查閱 Makefile檔案並進 行修改

有一個tip，很多下載後加壓縮后的檔案夾進行各種make以及make install的行為都是根據一個叫做makefile的檔案決定的。有時候我們可以通過閱讀這個檔案中的script或者注釋來瞭解他的安裝過程以及修改安裝路徑。

我們此時打開并修改這個文件的目的是由於我們不想將他安裝在默認的我們正在使用的系統當中，而是安裝到我們自己架設的系統當中

我們主要聚焦於install的過程，其中分為三塊，第一塊是用於存放make后的指令，第二塊適用於存放這個指令的manual及操作手冊。第三部分則是在存在xinetd這個開源的超級服務器守護綫程的時候能夠利用這個組件來維護我們的連接

由於我們只是用一小部分功能，而且我們系統中並沒有裝man和xinetd所以我們可以將這兩個部分刪去，如果想要使用那就一樣需要從網上下載源碼，安裝。

我們最需要改動的是第一個存放源碼的部分，只要把路徑和判斷的路徑改成我們自己系統相應的路徑即可

```
install:
    if [ -x /usr/local/sbin ]; then \
        $(INSTALL) -m 755 vsftpd /usr/local/sbin/vsftpd; \
    else \
        $(INSTALL) -m 755 vsftpd /usr/sbin/vsftpd; fi
    if [ -x /usr/local/man ]; then \
        $(INSTALL) -m 644 vsftpd.8 /usr/local/man/man8/vsftpd.8; \
        $(INSTALL) -m 644 vsftpd.conf.5 /usr/local/man/man5/vsftpd.conf.5; \
    elif [ -x /usr/share/man ]; then \
        $(INSTALL) -m 644 vsftpd.8 /usr/share/man/man8/vsftpd.8; \
        $(INSTALL) -m 644 vsftpd.conf.5 /usr/share/man/man5/vsftpd.conf.5; \
    else \
        $(INSTALL) -m 644 vsftpd.8 /usr/man/man8/vsftpd.8; \
        $(INSTALL) -m 644 vsftpd.conf.5 /usr/man/man5/vsftpd.conf.5; fi
    if [ -x /etc/xinetd.d ]; then \
        $(INSTALL) -m 644 xinetd.d/vsftpd /etc/xinetd.d/vsftpd; fi
```

```
install:
    if [ -x /home/apatea/rootfs/system/usr/local/sbin ]; then \
        $(INSTALL) -m 755 vsftpd /home/apatea/rootfs/system/usr/local/sbin/vsftpd; \
    else \
        $(INSTALL) -m 755 vsftpd /home/rootfs/system/usr/sbin/vsftpd; fi
    if [ -x /home/apatea/rootfs/system/etc/xinetd.d ]; then \
        $(INSTALL) -m 644 xinetd.d/vsftpd /home/apatea/rootfs/system/etc/xinetd.d/vsftpd; fi
```



# 第五步：make， make install來進 行安裝

首先，我們執行make之後就可以從中獲取到最終的執行檔。我們從圖片中可以看到其他部分所需要的文件也出现在了當中

然後我們利用make install就可以根據我們的設定將其安裝到我們對應的系統中啦。當然如果你看懂了之前的Install邏輯，你也可以自己手動複製過去。

```
apatea@apatea-VirtualBox: ~/rootfs/vsftpd-3.0.4$ ls
access.c      FAQ           hash.h        oneprocess.c  privops.c     secbuf.h      standalone.h  tcpwrap.o
access.h      features.c    hash.o        oneprocess.h  privops.h     secbuf.o      standalone.o  TODO
access.o      features.h    INSTALL       oneprocess.o  privops.o     seccompandbox.c  str.c         tunables.c
ascii.c       features.o    ipaddrparse.c  opts.c        privsock.c    seccompandbox.h  str.h         tunables.h
ascii.h       fileize.h    ipaddrparse.h  opts.h        privsock.h    seccompandbox.o  strlist.c     tunables.o
ascii.o       filestr.c    ipaddrparse.o  opts.o        privsock.o    SECURITY       strlist.h     TUNING
AUDIT         filestr.h    LICENSE       parseconf.c   ptracesandbox.c  secutil.c     strlist.o     twoprocess.c
banner.c      filestr.o    logging.c     parseconf.h   ptracesandbox.h  secutil.h     str.o         twoprocess.h
banner.h      ftpcmdio.c  logging.h     parseconf.o   ptracesandbox.o  secutil.o     sysdeputil.c  twoprocess.o
banner.o      ftpcmdio.h  logging.o     port           ptracesandbox.o  session.h     sysdeputil.h  utility.c
BENCHMARKS   ftpcmdio.o  ls.c          postlogin.c   README         SIZE           sysdeputil.o  utility.h
BUGS          ftpcodes.h  ls.h          postlogin.h   README.security SPEED          sysstr.c      utility.o
builddefs.h  ftpdataio.c  ls.o          postlogin.o   README.ssl     ssl.c          sysstr.h      vsf_findlibs.sh
Changelog    ftpdataio.h  main.c        postprivparent.c  readwrite.c    ssl.h          sysstr.o      vsftpd
COPYING      ftpdataio.o  main.o        postprivparent.h  readwrite.o    ssl.o          sysutil.c     vsftpd.8
COPYRIGHT    ftppolicy.c  Makefile      postprivparent.o  readwrite.o    sslslave.c     sysutil.h     vsftpd.conf
defs.h       ftppolicy.h  netstr.c      prelogin.c     RedHat         sslslave.h     sysutil.o     vsftpd.conf.5
dummyinc     ftppolicy.o  netstr.h      prelogin.h     REWARD        sslslave.o     tcpwrap.c     vsftpver.h
EXAMPLE      hash.c       netstr.o      prelogin.o     secbuf.c       standalone.c   tcpwrap.h     xinetd.d
```





# 第六步：修改設定檔案vsftpd.conf

除了之前的獲取并且安裝vsftpd指令之外，還有一個最重要的操作就是對於vsftpd.conf的設定。

這個設定檔決定了整個vsftpd的各種功能和行爲，例如是否允許匿名用戶，是否允許連線的用戶上傳資料，是否允許設定用戶連線后的家目錄，是否設定超時等等許許多多複雜的設定。

在vsftpd.conf檔案中他幫你羅列出了一系列較為常用的設定，而更加完整的設定介紹則是在vsftpd.conf5中，之前的makefile文件中我們也可以看到在install的過程中他將其加入man中供使用者參考。所以如果想要更加細緻和完整的設定，則需要參考這個文件後再進行設定。

而我們此次主要的設定有這幾個：

- 1.write\_enable=YES 允許更改
- 2.dirmessage\_enable=YES允許查看目錄信息
- 3.local\_enable=YES允許localusr登錄
- 4.chroot\_local\_usr=YES允許指定特定的家目錄給localusr
- 5.（需要自己加）allow\_writable\_chroot=YES由於我們給local\_usr指定的家目錄不能是可寫目錄，所以爲了將我們系統中自己的賬號能夠通過ftp連線，需要將他設定成可以寫目錄也能作爲家目錄

```
apatea@apatea-VirtualBox: ~/rootfs/vsftpd-3.0.4
# Example config file /etc/vsftpd.conf
#
# The default compiled in settings are fairly paranoid. This sample file
# loosens things up a bit, to make the ftp daemon more usable.
# Please see vsftpd.conf.5 for all compiled in defaults.
#
# READ THIS: This example file is NOT an exhaustive list of vsftpd options.
# Please read the vsftpd.conf.5 manual page to get a full idea of vsftpd's
# capabilities.
#
# Allow anonymous FTP? (Beware - allowed by default if you comment this out).
anonymous_enable=YES
#
# Uncomment this to allow local users to log in.
#local_enable=YES
#
# Uncomment this to enable any form of FTP write command.
#write_enable=YES
#
# Default umask for local users is 077. You may wish to change this to 022,
# if your users expect that (022 is used by most other ftpd's)
#local_umask=022
#
# Uncomment this to allow the anonymous FTP user to upload files. This only
# has an effect if the above global write enable is activated. Also, you will
# obviously need to create a directory writable by the FTP user.
#anon_upload_enable=YES
#
# Uncomment this if you want the anonymous FTP user to be able to create
# new directories.
#anon_mkdir_write_enable=YES
#
# Activate directory messages - messages given to remote users when they
# go into a certain directory.
dirmessage_enable=YES
#
1,1 顶端
```

## 第七步：做最後的前置準備

現在我們還需要根據之前INSTALL的要求做一些相應的前置準備就可以啦！

1. 將我們設置好的vsftpd.conf檔案複製到我們的system/etc中。
2. 爲了能夠讓我們的vsftp能夠獲取到用戶信息，我們需要在system/etc中創建一個vsftpd.usr\_list文件，并在裏面放入用戶名（一行一個）
3. 在system/usr/share中創建一個empty的文件夾，這個是一個默認的要求，我們從INSTALL中的prerequisite的2b中可以看到這項要求。
4. 做完了前三個步驟之後，我們只需要回到rootfs，並用我們的system重新製作initrd文件就可以重啓啦！

```
apatea@apatea-VirtualBox:~/rootfs/system/etc$ ls
fstab      hostname  issue     motd       passwd     protocols  securetty  shells     sudoers.d  udev
group      hosts     ld.so.conf mtab       pcmcia     ps1        services  skel       sudoers.dist vsftpd.conf
gshadow    init.d    mke2fs.conf nsswitch.conf profile     resolv.conf shadow     sudo.conf  sudo_logsrvd.conf vsftpd.usr_list
host.conf  inittab   modprobe.conf os-release profile.d   rpc        shadow-    sudoers    sysconfig
```

```
apatea
~
```

```
apatea@apatea-VirtualBox:~/rootfs/system/usr/share$ ls
doc empty i18n kmap locale misc syslinux tabset terminfo udhpcp
```

## 第八步：重啓系統 並使用vsftpd

現在我們已經在我們自己的system中加入了vsftpd，並且將他編譯壓縮為了initrd檔案放入我們的/boot中，現在我們之要開機打開我們的mylinux就可以執行這個指令並連線啦。

注意由於我們的vsftpd是一個後臺程序（vsftpd daemon），所以如果你不想要他一直占據著你的console的話，你最好在後面加一個&來讓他作為後臺程序執行。

由於我用的是虛擬機來進行這一系列的操作，而我的虛擬機的網路模式是（網絡地址轉換NAT)模式，所以外部的機器無法網路連接到虛擬機，因此我自己又通過一系列的操作在內部安裝了一個netkit\_ftp這個ftp client application來連線到我們的系統。

```
ubuntu [正在运行] - Oracle VM VirtualBox
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-rw-r--  1 1000  1000          0 Jun 11 13:32 test
226 Directory send OK.
ftp> bye
221 Goodbye.
[root@MyLinux ~]# kill %1
[root@MyLinux ~]# jobs
[1]+  Terminated                  vsftpd
[root@MyLinux ~]# vsftpd &
[root@MyLinux ~]# netkit-ftp
ftp> open 10.0.2.15
Connected to 10.0.2.15.
220 Welcome to blah FTP service.
Name (10.0.2.15:root): apatea
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> _
```

# 第九步：利用ftp連接來瀏覽文件夾，上傳/下載文件

接下來，我們可以利用對應的ftp指令來瀏覽我們的文件夾，以及上傳，下載文件！

這樣我們就成功架設並且可以使用我們的vsftpd組件啦！

```
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-rw-r-- 1 1000 1000 0 Jun 11 13:32 test
-rw-r--r-- 1 1000 1000 0 Jun 17 15:41 up_file
226 Directory send OK.
```

```
[root@MyLinux ~]# ls
[root@MyLinux ~]# touch up_file
[root@MyLinux ~]# ls
up_file
[root@MyLinux ~]# netkit-ftp
ftp> open 10.0.2.15
Connected to 10.0.2.15.
220 Welcome to blah FTP service.
Name (10.0.2.15:root): apatea
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-rw-r-- 1 1000 1000 0 Jun 11 13:32 test
226 Directory send OK.
ftp> get test
local: test remote: test
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for test (0 bytes).
226 Transfer complete.
ftp>
```

```
ubuntu [正在运行] - Oracle VM VirtualBox
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-rw-r-- 1 1000 1000 0 Jun 11 13:32 test
226 Directory send OK.
ftp> get test
local: test remote: test
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for test (0 bytes).
226 Transfer complete.
ftp> send
(local-file) up_file
(remote-file) up_file
local: up_file remote: up_file
200 PORT command successful. Consider using PASV.
150 Ok to send data.
226 Transfer complete.
ftp> bye
221 Goodbye.
[root@MyLinux ~]# ls
test up_file
```

## 拓展：netkit-ftp的下載和架設

其實所有的組件的下載和安裝流程都是一樣的，我們從網站下載原始碼 —> 進行編譯 —> 通過README等文件來瞭解安裝流程 —> 通過Makeinstall來瞭解改變安裝位置的方法（有些直接改文件，有些需要加入參數，例如--prefix=..等） —> 進行安裝和配置等 —> 成功後再重新編譯壓縮生成initrd文件放入/boot中再重啓即可。

其中還有一些較為簡單的方法，例如我的ubuntu中就含有netkit-ftp這個指令，並且他沒有過多必須的關聯文件，因此我只需要將這個指令複製到我的system中，再將這個指令所需要的一些文件（/lib/x86\_64-linux-gnu中的libreadline.o）放入system的相應位置即可，然後再生成initrd文件再重啓即可使用。



# 後記

其實這次假設的vsftpd的複雜程度，無論是操作上還是功能上都比較的簡單，如果需要實現某些特定的功能，或是讓vsftpd的功能更強大，連線更安全還是需要更加深入的瞭解vsftpd，然後來進行相應的配置，例如：

- 1.按照INSTALL的指示來進行一些前置工作
- 2.按照vsftpd.conf5來對vsftpd.conf來進行相應的設定。
- 3.通過下載pam，man，xinetd等來加強安全性和功能性以及便捷性。

同時針對於虛擬機的問題，還需要學會設定網路，使用例如“橋接網卡”的方式來使得我們的虛擬機能夠真正擁有自己的實體ip和獨立的網關，從而能夠真正的實現ftp的遠距離通信功能。



謝謝