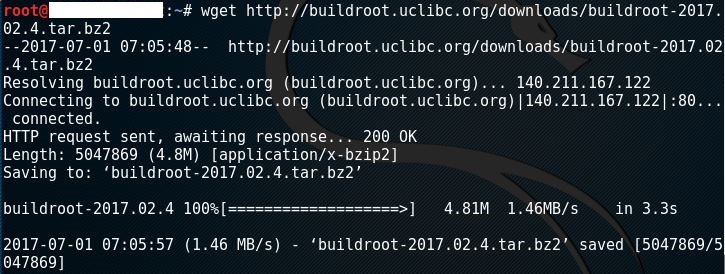
作者: 王駿逸 Simon Wang

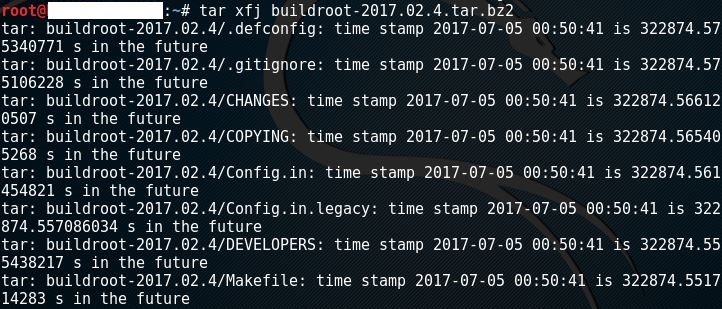
作業環境:

日期: 2017.9月

我們試著來安裝與設定 QEMU 這個虛擬機、做為我們的 Linux Kernel Debugging 的準備，在 “Qemu and the Kernel” 這篇文章中，先下載 Buildroot 這個工具的壓縮檔:



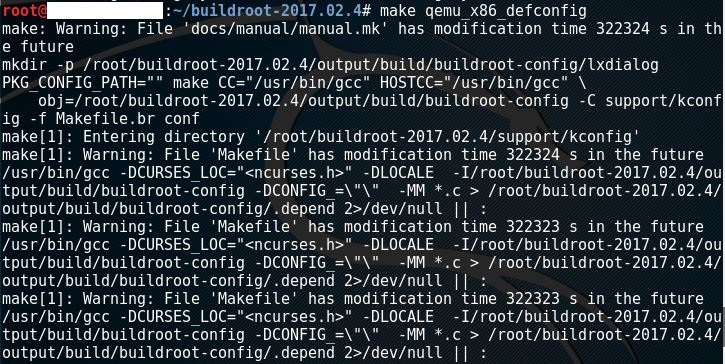
解開壓縮檔的時候、出現了一大串的這些東西(後面篇幅刪減):



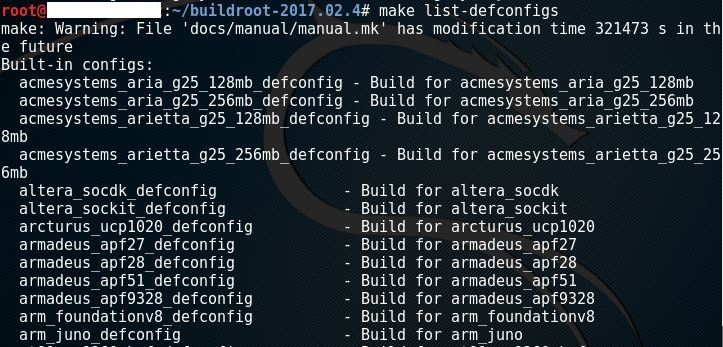
裡面有這些東西:



好像重複出現了大量的訊息:



一直重複下去都沒有停止、先按 Ctrl—z 來讓它停止。我們先用 make list-defconfigs 來確認有哪些預設的 OS config :



下面還有非常多、姑且省略，找到了與我們的系統環境相關的:

C:\Users\kdbot\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Qemu and the Kernel 7 找到與我們的系統環境相關的.jpg

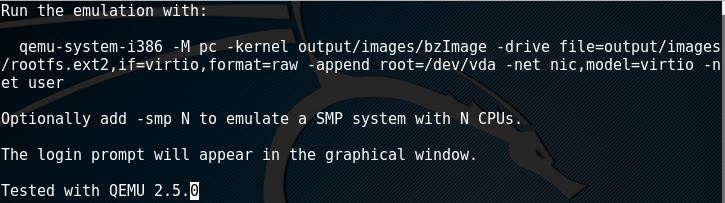
最後底下還有這樣的警示訊息:

C:\Users\kdbot\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Qemu and the Kernel 8 最後底下還有這樣的警示訊息.jpg

我們找到 buildroot/board/qemu 這個目錄下，確實有 x86 這個目錄:

C:\Users\kdbot\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Qemu and the Kernel 9 我們在 board qemu 這個目錄下 確實有 x86 這個目錄.jpg

打開這個 readme.txt 看看:

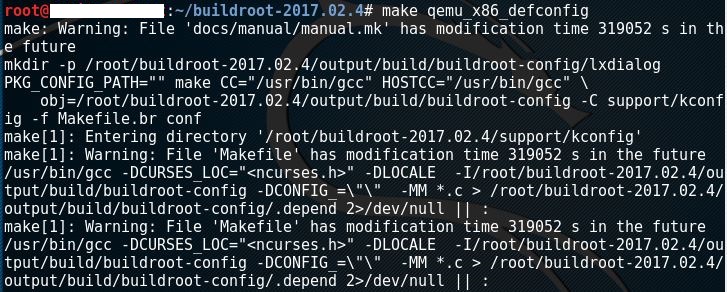


這是接下來要在 Buildroot 中啟動 QEMU 所模擬的 x86 系統時要輸入的指令。

從 “Compiling C++ on remote Linux machine - “clock skew detected” warning

“ 這篇討論看來、這個 make: Warning: File ‘docs/manual/manual.mk’ has modification time 322324 s in the future 錯誤訊息、應該是我們所下載的 Buildroot 檔案的修改日期、竟然是我們系統目前時間的未來式，所以我們要參照討論中的指示，用 touch \* 指令來校正檔案時間:

C:\Users\kdbot\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Qemu and the Kernel 11 使用 touch 星號 指令來校正檔案時間.jpg



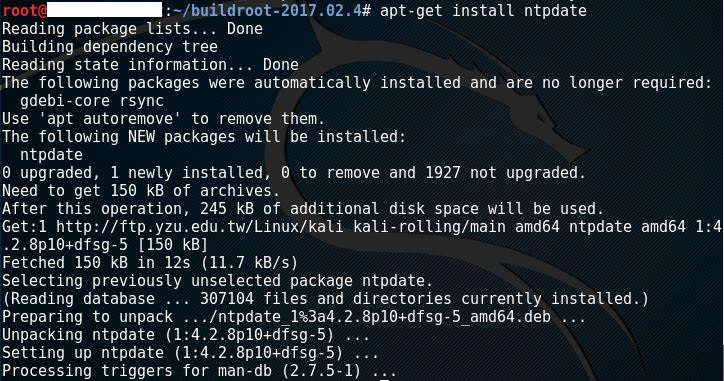
還是一樣的錯誤，看來我們需要的是 調整系統環境的時間到現時 ，由於我們個人的 Linux 系統並不是 NTP Server ，所以我們得向外找一個公正可信的 NTP Server 來校正我們的 Linux 系統時間、同時也寫入到我們的 BIOS 時間:

C:\Users\kdbot\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Qemu and the Kernel 13 使用 ntpdate 來和中華電信的伺服器來同步時間.jpg

居然找不到 ntpdate 指令?

C:\Users\kdbot\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Qemu and the Kernel 14 Kali Linux 預設果真是沒有 ntpdate 指令.jpg

Kali Linux 預設果真是沒有 ntpdate 指令，我們試著安裝看看:

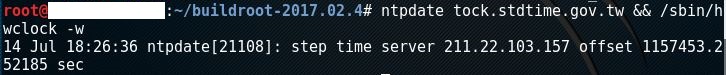


C:\Users\kdbot\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Qemu and the Kernel 16 有了 ntpdate.jpg

看一下我們的 Linux 系統目前的時間:

C:\Users\kdbot\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Qemu and the Kernel 17 看一下我們的 Linux 系統目前的時間.jpg

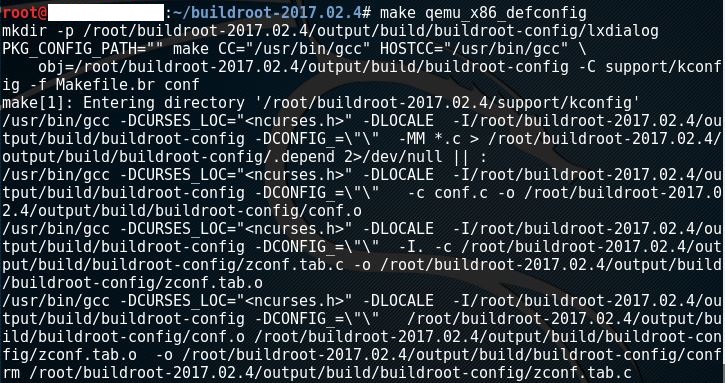
再一次同步時間:

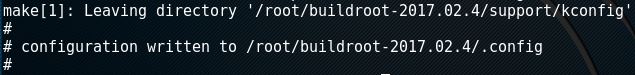


同步完成 顯示目前的時間:

C:\Users\kdbot\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Qemu and the Kernel 19 同步完成 顯示目前的時間.jpg

make qemu\_x86\_defconfig 成功:

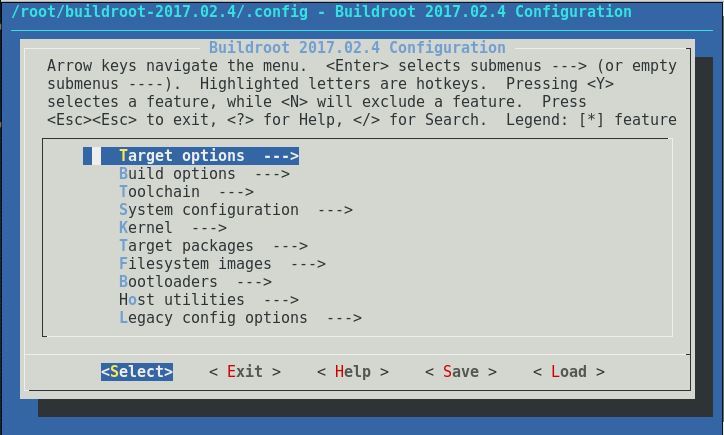


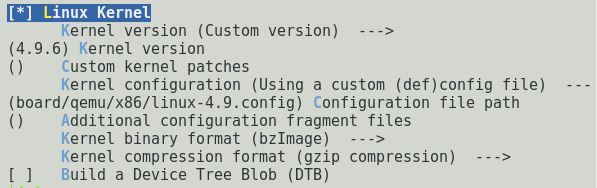


make menuconfig:

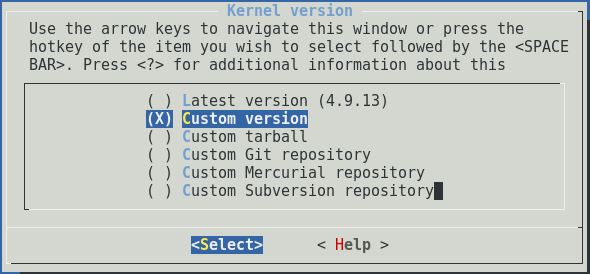
C:\Users\kdbot\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Qemu and the Kernel 21 make menuconfig 1.jpg

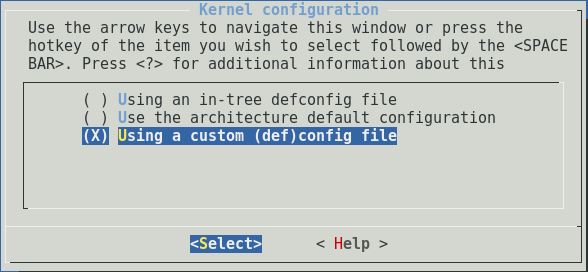
下面是我截圖的一些關於這個 menuconfig 的預設設定:



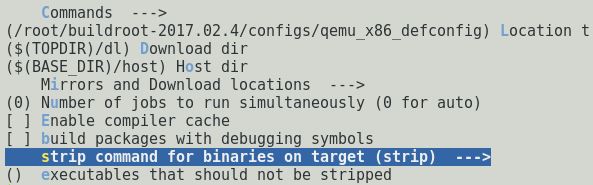


C:\Users\kdbot\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Qemu and the Kernel 24 make menuconfig 4.jpg

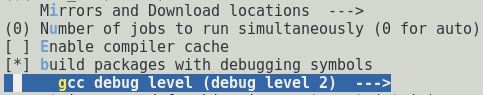




Enable compiler cache 不要打開:



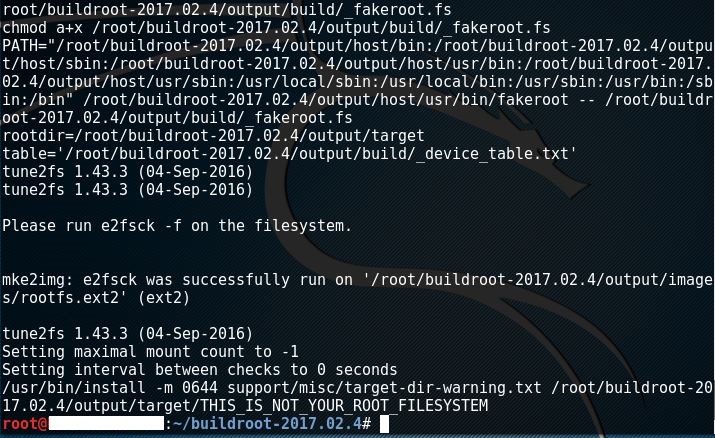
build packages with debugging symbols 可能是我們要的:



開始 make Compile:

C:\Users\kdbot\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Qemu and the Kernel 29 開始 make Compile.jpg

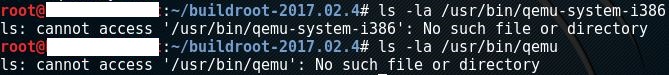
make 應該是完成了:



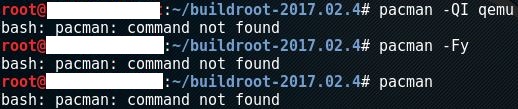
啟動 QEMU emulation 找不到指令:



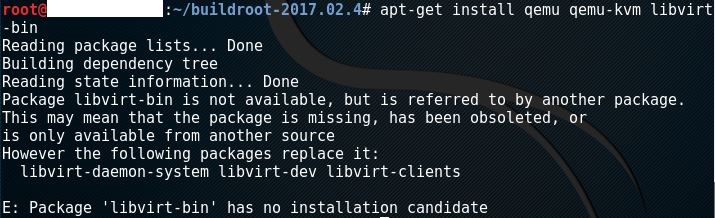
確實是沒有 qemu-system-i386 和 qemu 這兩個程式:



居然也沒有 pacman 這個指令:



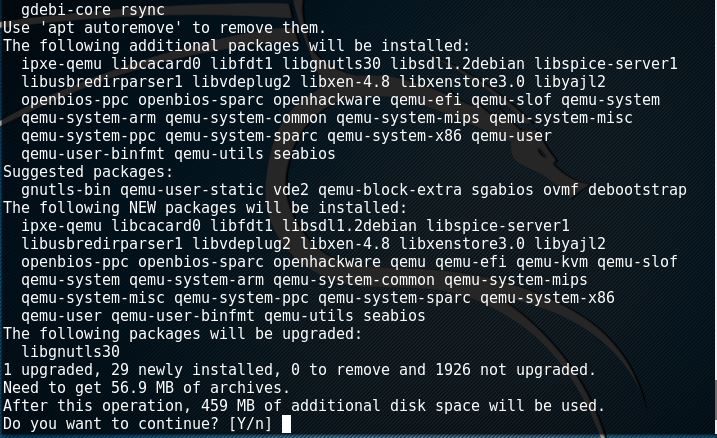
安裝 qemu 及其附屬套件:



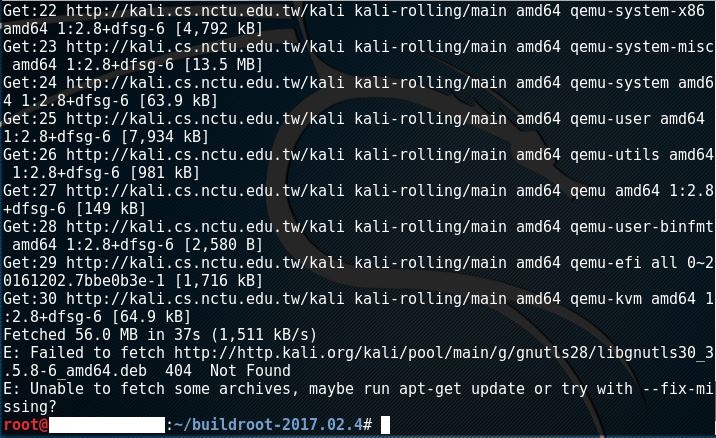
我們試著換另一種方式來安裝:

C:\Users\kdbot\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Qemu and the Kernel 35 安裝 qemu 及其附屬套件 我們試著換另一種方式來安裝 1.jpg

來到這裡:



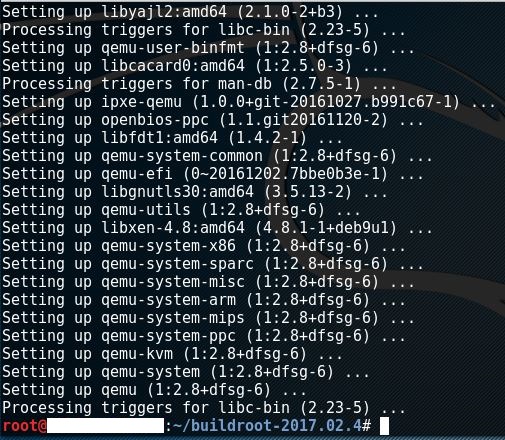
結果停在這裡:



我們試著 update Packages 來源看看:



應該是完成了:



如果我們想要我們的 KVM 虛擬環境有 GUI 介面 我們還要安裝 virt manager:

C:\Users\kdbot\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Qemu and the Kernel 40 安裝 qemu 及其附屬套件 我們試著換另一種方式來安裝 6 如果我們想要我們的 KVM 虛擬環境有 GUI 介面 我們還要安裝 virt manager 1.jpg