**Hướng dẫn tạo file ISO cấu hình tự động cho HĐH Ubuntu**

 Tham khảo chi tiết hơn tại link sau: <https://ubuntu.com/server/docs/install/autoinstall-reference>

**Phần 2: Hướng dẫn tạo ISO cấu hình tự động cho HĐH Ubuntu 22.04**

**Điểm khác biệt giữa cấu trúc ISO Ubuntu 20.04 và Ubuntu 22.04**

Kể từ phiên bản Ubuntu 22.04, ISOLINUX đã bị loại bỏ, và thay vào đó booloader sử dụng trên ISO của Ubuntu 22.04 là GRUB2. Ta có thể sử dụng lệnh fdisk -l các file ISO để nhìn chi tiết hơn vào cấu trúc bên trong của các file ISO

Với Ubuntu 20.04

|  |
| --- |
| monitor@localhost:/u01/data/iso$ fdisk -l ubuntu-20.04.3-live-server-**amd64.iso**  Disk ubuntu-20.04.3-live-server-amd64.iso: 1.18 GiB, 1261371392 bytes, 2463616 sectors  Units: sectors of 1 \* 512 = 512 bytes  Sector size (logical/physical): 512 bytes / 512 bytes  I/O size (minimum/optimal): 512 bytes / 512 bytes  Disklabel type: dos  Disk identifier: 0x7fcaeaa1  Device Boot Start End Sectors Size Id Type  ubuntu-20.04.3-live-server-amd64.iso1 \* 0 2463615 2463616 1.2G 0 Empty  ubuntu-20.04.3-live-server-amd64.iso2 656164 664163 8000 3.9M ef EFI (FAT-12/16/32) |

Với Ubuntu 22.04

|  |
| --- |
| Disk ubuntu-22.04.2-live-server-amd64.iso: 1.86 GiB, 1975971840 bytes, 3859320 sectors  Units: sectors of 1 \* 512 = 512 bytes  Sector size (logical/physical): 512 bytes / 512 bytes  I/O size (minimum/optimal): 512 bytes / 512 bytes  Disklabel type: gpt  Disk identifier: C3B09D5C-3F01-457B-AC5E-206E8818BFA0  Device Start End Sectors Size Type  ubuntu-22.04.2-live-server-amd64.iso1 64 3848587 3848524 1.9G Microsoft basic data  ubuntu-22.04.2-live-server-amd64.iso2 3848588 3858655 10068 4.9M EFI System  ubuntu-22.04.2-live-server-amd64.iso3 3858656 3859255 600 300K Microsoft basic data |

Ta có thể nhận thấy, với Ubuntu 20.04 cấu trúc file ISO vẫn theo chuẩn MBR cũ, trong khi đó từ Ubuntu 22.04 cấu trúc file ISO đã chuyển sang chuẩn GPT. Nếu trước đây, khi đóng gói lại ISO ta có thể bỏ phân vùng EFI đi được thì kể từ Ubuntu 22.04 khi đóng gói lại ISO bắt buộc phải giữ đủ cấu trúc 3 phân vùng có phân vùng EFI như mô tả ở trên, đồng thời việc gắn bootloader ISOLINUX cho Ubuntu 22.04 cũng không còn tác dụng nữa. Như vậy, việc giải nén và đóng gói lại file ISO cho Ubuntu 22.04 sẽ có sự thay đổi. Đối với các phần cấu hình khác liên quan đến cloud-init thì Ubuntu 22.04 tương đồng với Ubuntu 20.04, có thể sử dụng kịch bản cloud-init của Ubuntu 20.04 sang sử dụng cho Ubuntu 22.04.

Đồng thời khuyến cáo với phiên bản Ubuntu 22.04 (và có thể là cả các bản phát hành Ubuntu trở về sau), khi thực hiện cài đặt trên server vật lý thì nên chuyển boot mode của server về UEFI để tận dụng các tính năng của UEFI trên phần cứng mới, cũng như tránh các lỗi không tương thích giữa Ubuntu 22.04 và cơ chế boot BIOS truyền thống.

**Bước 1: Cài đặt 7zip** (vì 7zip cho phép extract all cả các file trong các phân vùng GRUB của file iso)

Giả sử trong môi trường là server Ubuntu, thực hiện cài 7zip như sau

sudo apt-get install p7zip-full

**Bước 2: Trích xuất thông tin từ file ISO gốc**

Thực hiện lệnh sau

xorriso -indev ubuntu-22.04.2-live-server-amd64.iso -report\_el\_torito as\_mkisofs

Kết quả có được là như sau

|  |
| --- |
| Drive current: -indev '/u01/data/iso/ubuntu-22.04.2-live-server-amd64.iso'  Media current: stdio file, overwriteable  Media status : is written , is appendable  Boot record : El Torito , MBR protective-msdos-label grub2-mbr cyl-align-off GPT  Media summary: 1 session, 964830 data blocks, 1884m data, 133g free  Volume id : 'Ubuntu-Server 22.04.2 LTS amd64'  -V 'Ubuntu-Server 22.04.2 LTS amd64'  --modification-date='2023021721571500'  --grub2-mbr --interval:local\_fs:0s-15s:zero\_mbrpt,zero\_gpt:'/u01/data/iso/ubuntu-22.04.2-live-server-amd64.iso'  --protective-msdos-label  -partition\_cyl\_align off  -partition\_offset 16  --mbr-force-bootable  -append\_partition 2 28732ac11ff8d211ba4b00a0c93ec93b --interval:local\_fs:3848588d-3858655d::'/u01/data/iso/ubuntu-22.04.2-live-server-amd64.iso'  -part\_like\_isohybrid  -iso\_mbr\_part\_type a2a0d0ebe5b9334487c068b6b72699c7  -c '/boot.catalog'  -b '/boot/grub/i386-pc/eltorito.img'  -no-emul-boot  -boot-load-size 4  -boot-info-table  --grub2-boot-info  -eltorito-alt-boot  -e '--interval:appended\_partition\_2\_start\_962147s\_size\_10068d:all::'  -no-emul-boot  -boot-load-size 10068  -isohybrid-gpt-basdat |

Chú ý ghi lại 2 thông tin này, vì nó sẽ được sử dụng cho phần đóng gói file ISO. Rất may, thông thường các ISO thuộc phiên bản Ubuntu 22.04 (kể cả bản gốc lẫn các bản nâng cấp sau đó) thì các thông số này không thay đổi.

|  |
| --- |
| -append\_partition 2 28732ac11ff8d211ba4b00a0c93ec93b  -iso\_mbr\_part\_type a2a0d0ebe5b9334487c068b6b72699c7 |

**Bước 3: Extract file iso ra thư mục**

Chạy lệnh sau

mkdir -p ubuntu\_2204/

7z -y x /u01/data/iso/ubuntu-22.04.2-live-server-amd64.iso -oubuntu\_2204

Chú ý: Dính ngay sau -o là tên thư mục muốn extract ISO ra (không để dấu cách)

**Bước 4: Tạo 1 thư mục chứa file cloud-init bên trong iso**

(giả sử thư mục cài là ubuntu\_2204)

Thực hiện tạo thư mục chứa các file cloud-init như sau

mkdir kickstart

touch ubuntu\_2204/kickstart/meta-data (file meta-data rỗng cũng được)

cp user-data ubuntu\_2204/kickstart/user-data (đẩy nội dung user-data cấu hình tự động vào đây)

**Bước 5: Đẩy boot bằng cloud-init lên trước**

Thực hiện sửa các file sau đây

Boot bằng UEFI: Sửa file <thư mục DVD>/boot/grub/grub.cfg

Ví dụ chèn đoạn sau lên phía trên các menu mặc định

|  |
| --- |
| menuentry "Kickstart cloud-init install Ubuntu Server" {  set gfxpayload=keep  linux /casper/vmlinuz quiet autoinstall "ds=nocloud;s=/cdrom/kickstart/" ---  initrd /casper/initrd  } |

**Bước 6 (tùy chọn): bỏ check md5**

Mặc định, quá trình cài đặt Ubuntu thực hiện check MD5 tất cả các file bên trong (điều này giải thích vì sao quá trình load bộ cài Ubuntu lại lâu hơn các HĐH khác), có thể bỏ qua việc này bằng cách như sau

echo "" > md5sum.txt

**Bước 7: Tạo iso**

Giả sử thực hiện lệnh tạo ISO như sau:

|  |
| --- |
| xorriso -as mkisofs -r -V 'Ubuntu 22.04 LTS' -o ./ubuntu\_2204\_liemmt.iso --grub2-mbr ./ubuntu\_2204/'[BOOT]'/1-Boot-NoEmul.img -partition\_offset 16 --mbr-force-bootable -append\_partition 2 28732ac11ff8d211ba4b00a0c93ec93b ./ubuntu\_2204/'[BOOT]'/2-Boot-NoEmul.img -appended\_part\_as\_gpt -iso\_mbr\_part\_type a2a0d0ebe5b9334487c068b6b72699c7 -c '/boot.catalog' -b '/boot/grub/i386-pc/eltorito.img' -no-emul-boot -boot-load-size 4 -boot-info-table --grub2-boot-info -eltorito-alt-boot -e '--interval:appended\_partition\_2:::' -no-emul-boot ubuntu\_2204/ |

Trong đó:

2 tham số này lấy dựa theo kết quả trích xuất đã chạy ở bước 2

|  |
| --- |
| -append\_partition 2 28732ac11ff8d211ba4b00a0c93ec93b  -iso\_mbr\_part\_type a2a0d0ebe5b9334487c068b6b72699c7 |

Thay đường dẫn trỏ đến thư mục đã extract bộ cài Ubuntu (mà ở trong ví dụ trên, thư mục đó đang là ./ubuntu\_2204) ở các tham số sau (lưu ý thư mục [BOOT] cho vào giữa 2 dấu '' để tránh ký tự đặc biệt)

|  |
| --- |
| --grub2-mbr ./ubuntu\_2204/'[BOOT]'/1-Boot-NoEmul.img  --mbr-force-bootable -append\_partition 2 28732ac11ff8d211ba4b00a0c93ec93b ./ubuntu\_2204/'[BOOT]'/2-Boot-NoEmul.img  -no-emul-boot ubuntu\_2204/ |

Lưu ý: Việc thực hiện tạo file iso nên được thực hiện trên 1 server chạy Ubuntu và riêng với Ubuntu 22.04 thì khuyến cáo đóng gói file ISO trên môi trường từ Ubuntu 20.04 trở lên. Nếu thực hiện trên server chạy OS khác như CentOS, SuSE v.v... có thể gặp lỗi không tương thích thư viện và phiên bản.

Phụ lục: File user-data tham khảo. Lưu ý, cấu hình phân vùng HDD file này chỉ dành cho server boot theo chế độ UEFI và format cả HDD /dev/sda

|  |
| --- |
| #cloud-config for cloud server  **autoinstall**:  **version**: 1  # default keyboard layout, locale and timezone  **keyboard**:  **layout**: us  **locale**: en\_US.UTF-8  **timezone**: Asia/Ho\_Chi\_Minh  # install ssh server  **ssh**:  **allow-pw**: **true**  **authorized-keys**: []  **install-server**: **true**  # default hostname, vt\_admin user  **identity**:  **realname**: viettel-server  **hostname**: viettel-server  **username**: vt\_admin  # password is 1  **password**: $6$I8Km57LbbFp01Krh$YlIykhuXYm/UiurphgWJ3CkwyDVYrQCM6jf.Siy90du32UMdMbDd629QkLYfMr3YXr1Q5WhO4dThRntcxLfMI0  # partition config, using for server uefi mode  **storage**:  **config**:  **- {ptable**: gpt, path: /dev/sda, wipe: superblock-recursive, preserve: false, name: '',  **grub\_device**: false, type: disk, id: disk-sda}  **- {device**: disk-sda, size: 512M, wipe: superblock, flag: boot, number: 1,  **preserve**: false, grub\_device: true, type: partition, id: partition-0}  **- {fstype**: fat32, volume: partition-0, preserve: false, type: format, id: format-0}  **- {device**: disk-sda, size: 1G, wipe: superblock, flag: '', number: 2,  **preserve**: false, grub\_device: false, type: partition, id: partition-1}  **- {fstype**: ext4, volume: partition-1, preserve: false, type: format, id: format-1}  **- {device**: disk-sda, size: -1, wipe: superblock, flag: '', number: 3,  **preserve**: false, grub\_device: false, type: partition, id: partition-3}  **- name**: vg-ubuntu  **devices**: [partition-3]  **preserve**: **false**  **type**: lvm\_volgroup  **id**: lvm\_volgroup-0  **- {name**: lv-ubuntu-swap, volgroup: lvm\_volgroup-0, size: 10G, wipe: superblock,  **preserve**: false, type: lvm\_partition, id: lvm\_partition-0}  **- {fstype**: swap, volume: lvm\_partition-0, preserve: false, type: format, id: format-3}  **- {path**: '', device: format-3, type: mount, id: mount-2}  **- {name**: lv-ubuntu-root, volgroup: lvm\_volgroup-0, size: -1, wipe: superblock,  **preserve**: false, type: lvm\_partition, id: lvm\_partition-1}  **- {fstype**: ext4, volume: lvm\_partition-1, preserve: false, type: format, id: format-4}  **- {path**: /, device: format-4, type: mount, id: mount-3}  **- {path**: /boot, device: format-1, type: mount, id: mount-1}  **- {path**: /boot/efi, device: format-0, type: mount, id: mount-0}  **swap**: {swap: 0}  # run command after install ubuntu  **late-commands**:  - cp -r /cdrom/postinstall/ /target/root/  - cd /target/root/postinstall/networkconf\_ubuntu/ ; bash ubuntu\_postinstall\_auto\_config\_network.sh |

Lưu ý khi thực hiện các lệnh cấu hình trong late-commands:

Toàn bộ filesystem của môi trường đích nằm dưới thư mục /target, nghĩa là ví dụ nếu muốn cấu hình network trong late-commands thì cần sửa cấu hình trong file /target/etc/netplan/00-installer-config.yaml.

So với môi trường cài đặt của CentOS/Redhat, môi trường cài đặt của Ubuntu mới và đầy đủ hơn, có hỗ trợ python3. Do đó có thể thực hiện chạy các script python trong môi trường cài đặt để thực hiện các thao tác cấp cao hơn, tuy nhiên cần lưu ý chỉ có các thư viện và module mặc định.

Có thể tham khảo cách sinh file cấu hình network tự động như sau:

Download script Python sinh file map cấu hình network tại đây: https://datasecurity.viettel.vn/s/V6SnyvtTEvYUdip/excel-network-config-file-generator

Từ 1 máy đã có Python 3, cài các Python package openpyxl, chạy script Python đã được biên soạn sẵn. Nếu không thêm tham số gì phía sau, mặc định đầu vào là file excel\_input.xlsx và file xuất ra là network\_servers.txt trong thư mục chạy. Do đó, cần tạo file đầu vào theo đúng cấu trúc đầu vào ví dụ như trong file excel\_input.xlsx.

|  |
| --- |
| python excel\_network\_config\_file\_generator.py |

Lưu ý: Nếu chạy script Python này trên Windows, file text kết quả xuất ra sẽ có định dạng end of line là Windows, cần convert về dạng Unix nếu muốn sử dụng trong môi trường Linux. Nếu sử dụng Notepad++, có thể thực hiện convert bằng cách chọn menu Edit -> EOL Conversion -> Unix (LF) rồi lưu lại file text.

Download các bash script cấu hình network tương ứng tại đây: https://datasecurity.viettel.vn/s/X4tdsU5O3cZurdc/networkconf-ubuntu

Trong thư mục chứa file ISO đã giải nén, ta nên tạo 1 thư mục mới để chứa các script cài đặt tương ứng. Lưu ý upload cả file map cấu hình network (network\_servers.txt) vào thư mục này.

|  |
| --- |
| mkdir /u01/laboratory/autokickstart\_dvd/ubuntu\_2004/postinstall |

Sử dụng các trình sftp hoặc ssh put file vào vị trí thư mục tương ứng.

Đưa các dòng sau vào late-commands để thực hiện gọi script cấu hình tự động sau khi cài xong Ubuntu:

|  |
| --- |
| - cp -r /cdrom/postinstall/ /target/root/  - cd /target/root/postinstall/networkconf\_ubuntu/ ; bash ubuntu\_postinstall\_auto\_config\_network.sh |