

Vytvorte si svoju vlastnú distribúciu

Napísal intense

Sobota, 17 Marec 2012 18:55 - Posledná zmena Sobota, 17 Marec 2012 19:07

Po čase používania Linuxu pravdepodobne každého geeka napadne upraviť si nejaký Live obraz distra pre svoje potreby. Problém môže byť, kde začať? Ohľadom upravovania live obrazov nie je veľa dokumentácie takže začať s úpravou môže byť trochu problémové. Existujú samozrejme nejaké tie klikátka alebo web služby typu SUSEStudio a pod. To je ale také pre BFU podľa môjho názoru, človek sa tam toho veľa nenaučí že ako to funguje. Ukážeme si teraz radšej ako si upraviť distro za použitia štandardných linuxových nástrojov a terminálu. ⁸⁸

Upozornenie, nasledujúci postup vyžaduje priemerné znalosti Linuxového systému a používania terminálu ⁸⁸

Ako prvé si musíme najprv správne pripraviť prostredie v ktorom budeme obraz upravovať. Uistite sa že máte nainštalované tieto programy:

- **mksquashfs**
- **rsync**
- **chroot**
- **mkisofs**
- **libisofs**

Ak nemáte, nájdite si vo vašom repozitári príslušné balíky, ktoré ich obsahujú. Balík cdrtools by

Vytvorte si svoju vlastnú distribúciu

Napísal intense

Sobota, 17 Marec 2012 18:55 - Posledná zmena Sobota, 17 Marec 2012 19:07

mal obsahovať mkisofs (alebo genisoimage, záleží od distribúcie) a squashfs-tools zase mksquashfs. Pre upravovanie potrebujeme mať ale ešte podporu v kerneli pre SquashFS. To či ju máte môžete zistiť v súbore /proc/filesystems alebo príkazom:

```
cat /proc/filesystems |grep squash
```

Aj keď to z hľadiska bezpečnosti nie je dobré, robte všetky popísané príkazy ako root. Aj tak takmer všetky z nich budú potrebovať root práva aj keď samozrejme sa nájdu také čo môžete vykonať ako bežný user.

Presuňte sa v termináli do priečinka v ktorom budete upravovať váš live obraz a taktiež si tam nakopírujte pokusné *.iso. Ja použijem Debian live 6.0.3 i386 xfce, odporúčam vám tiež použiť debian alebo nejaké debian based distro keďže postup úpravy nie je práve veľmi univerzálny pre distribúcie aj keď po malej úprave môžete kludne upravovať aj iné distribúcie. Postup by nemal byť závislý od distra na ktorom obraz upravujete aj keď túto závislosť som nejak neskúmal, každopádne môj hosťovský systém na ktorom budem vykonávať úpravu je plne updatnuté Gentoo. Fajn, takže teraz si vytvorme základné priečinky:

Singularity blog # mkdir root

Singularity blog # mkdir mounter

Singularity blog # mkdir remaster

Vytvorte si svoju vlastnú distribúciu

Napísal intense

Sobota, 17 Marec 2012 18:55 - Posledná zmena Sobota, 17 Marec 2012 19:07

Priečinok root bude obsahovať kompletný filesystem distribúcie, remaster bude slúžiť na uchovanie rozbaleného iso súboru a mounter budeme používať ako dočasný mountpoint. Takto by to teraz u vás malo vyzeráť:

Singularity blog # ls

```
debian.iso mounter remaster root
```

Mountneme si náš originálny iso obraz a skopírujeme z neho všetky súbory, keď budeme mountovať nejaké fs tak warningy o tom že to je len na čítanie kludne ignorujte, aj tak stamadiaľ budeme vždy len kopírovať:

Singularity blog # mount -o loop debian.iso mounter/

```
mount: warning: mounter/ seems to be mounted read-only.
```

Singularity blog # rsync -ad mounter/ remaster/

Vytvorte si svoju vlastnú distribúciu

Napísal intense

Sobota, 17 Marec 2012 18:55 - Posledná zmena Sobota, 17 Marec 2012 19:07

Singularity blog # umount mounter/

Priečinkom remaster nám teraz obsahuje vnútornosti iso obrazu. Veľa nám to ale ešte nepomôže keďže pre úpravy sa musíme dostať do vnútra upravovaného systému.

Kompletný obraz systému sa nachádza v súbore filesystem.squashfs. Tento súbor sa nemusí nachádzať na nejakom presne definovanom mieste, to závisí od distribúcie, ktorú upravujete. V Debiane sa tento súbor nachádza v priečinku live/ , kdežto v ubuntu based distrách v priečinku casper/ . Cestu k tomuto súboru si prispôbte podľa vášho distra. SquashFS si môžete predstaviť ako nejaký iso obraz, je to svojím spôsobom archív (normálne len read-only) ktorý obsahuje zapackovaný celý súborový systém distribúcie. Tento obraz je navyše komprimovaný pomocou lzma, bzip2 alebo gzip algoritmu aby sa na CD/DVD zmestilo toho viac. Mountneme si teraz tento súbor a skopírujeme si z neho kompletný FS distribúcie v ktorom budeme robiť úpravy. Táto operácia môže trvať pomerne dlho a zaberie pár GB diskového priestoru, pretože ako som už spomínal, súborový systém vnútri squashfs je komprimovaný takže po rozbalení narastie jeho veľkosť a rozbaľovanie jednak trvá pomerne dlho keďže je použitá kompresia. Kto sa v Linuxe vyzná tak môže použiť pre rozbalenie nástroj unsquashfs, my ale použijeme teraz mount s rsyncom rovnako ako pri rozbaľovaní iso obrazu:

Singularity blog # mount -t squashfs -o loop remaster/live/filesystem.squashfs mounter/

mount: warning: mounter/ seems to be mounted read-only.

Singularity blog # rsync -ad mounter/ root/

Vytvorte si svoju vlastnú distribúciu

Napísal intense

Sobota, 17 Marec 2012 18:55 - Posledná zmena Sobota, 17 Marec 2012 19:07

```
Singularity blog # umount mounter/
```

Super, teraz máme už unpacknutý celý FS (File System) distribúcie. Musíme ale ešte spraviť pár úprav pred tým než sa tam chrootneme. Aby nám fungovala sieť tak skopírujeme hosts a resolv.conf z nášho hosťovského systému a následne sa chrootneme do vnútra systému:

```
Singularity blog # cp /etc/hosts root/etc/
```

```
Singularity blog # cp /etc/resolv.conf root/etc/
```

```
Singularity blog # mount --bind /dev root/dev
```

```
Singularity blog # chroot root/ /bin/bash
```

```
root@Singularity:/#
```

Fajn, teraz už sme vo vnútri nášho systému, zrejme vás to už napína upravovať ale ešte stále musíme upraviť nastaviť vecí :-D a to konkrétne:

Vytvorte si svoju vlastnú distribúciu

Napísal intense

Sobota, 17 Marec 2012 18:55 - Posledná zmena Sobota, 17 Marec 2012 19:07

```
root@Singularity:/# mount -t proc none /proc
```

```
warning: can't open /etc/fstab: No such file or directory
```

```
root@Singularity:/# mount -t sysfs none /sys
```

```
root@Singularity:/# mount -t devpts none /dev/pts
```

```
root@Singularity:/# export HOME=/root
```

```
root@Singularity:/# export LC_ALL=C
```

```
root@Singularity:/# dbus-uuidgen > /var/lib/dbus/machine-id
```

```
root@Singularity:/# dpkg-divert --local --rename --add /sbin/initctl
```

```
Adding 'local diversion of /sbin/initctl to /sbin/initctl.distrib'
```

```
root@Singularity:/# ln -s /bin/true /sbin/initctl
```

Vytvorte si svoju vlastnú distribúciu

Napísal intense

Sobota, 17 Marec 2012 18:55 - Posledná zmena Sobota, 17 Marec 2012 19:07

Takže, gentoo useri pravdepodobne vedia že o čo sa jednalo v týchto krokoch. Pre ten zvyšok to teraz tak zbežne vysvetlím. Najprv sme mountli pár default pseudo-filesystémov pre prípad že by k nim nejaký proces pristupoval aby sa nenarobila neplecha. Toto je úplne bežný postup ak sa chrootujete do nejakého systému. Následne sme exportovali home premennú a locale pretože ak budeme inštalovať nejaký software alebo používať niektoré špecifické programy tak tieto premenné používajú a bez nich by veľmi nefungovali. Následne sme vygenerovali unikátne uuid, ktoré bude slúžiť ako id pre našu virtuálnu mašinu. Potom sme vytvorili diversion na /sbin/initctl pretože vo vnútri chrootu nemôžu štartovať upstart joby, ktoré by zlyhali a mohli narobiť bordel pretože by nemohli komunikovať s prostredím mimo chrootu. Ak by aj predsa len chcel nejaký takýto proces raziť hlavou stenu tak v poslednom kroku sme nalinkovali /bin/true na pôvodný initctl aby sme ho mierne obľbli.

V tomto štádiu už si teraz môžeme chlípať kávičku a robiť ľubovoľné zmeny v systéme. Fajnšmekri si dokonca môžu pomocou Xephyr otvoriť Xka do tohto systému a využívať tak grafické prostredie rovnako ako keby ten disk práve nabootovali. Proste si tam teraz doinštalujte čo potrebujete, upravujte si nejaké konfiguráky, odstráňte nejaké zbytočnosti čo tam mať nechcete a podobne. Pár tipov pri úprave:

Ak odstraňujete balíky, používajte voľbu purge!

Upravením /etc/skel môžete upraviť nastavenie všetkých používateľov naraz, všetko čo je v tomto priečinku sa automaticky použije ako kostra pri vytvorení nového používateľa.

Dávajte bacha že čo upravujete, ľahko to môže dopadnúť tak že vám to ani nenabootuje, v tých miernych prípadoch to bude trpieť schizofréniou ;-)

Ak robíte pokročilé úpravy, uistite sa že ste tam nevytvorili žiaden súbor s UID > 999, inak vám váš systém nenabootuje.

Vytvorte si svoju vlastnú distribúciu

Napísal intense

Sobota, 17 Marec 2012 18:55 - Posledná zmena Sobota, 17 Marec 2012 19:07

Takže, teraz keď ste si už systém poupravovali tak je na čase ho zbaliť späť do iso obrazu aby sme ho mohli normálne naboovať. Ak ste niečo inštalovali tak vyčistite cache apt a ten blast čo sa popri tom narobil:

```
root@Singularity:/# aptitude clean
```

```
root@Singularity:/# rm -rf /tmp/*
```

Upozornenie!!!: Ak ste manipulovali s kernelom alebo ste ho nejak zmenili, vygenerujte nový initrd!!! (napr. pomocou mkinitframs)

Navrátíme naše chroot úpravy do pôvodného stavu:

```
root@Singularity:/# rm /etc/hosts
```


Vytvorte si svoju vlastnú distribúciu

Napísal intense

Sobota, 17 Marec 2012 18:55 - Posledná zmena Sobota, 17 Marec 2012 19:07

```
root@Singularity:/# rm /etc/resolv.conf
```

```
root@Singularity:/# rm /var/lib/dbus/machine-id
```

```
root@Singularity:/# rm /sbin/initctl
```

```
root@Singularity:/# dpkg-divert --rename --remove /sbin/initctl
```

```
Removing 'local diversion of /sbin/initctl to /sbin/initctl.distrib'
```

```
root@Singularity:/# umount /proc || umount -lf /proc
```

```
root@Singularity:/# umount /sys
```

```
root@Singularity:/# umount /dev/pts
```

```
root@Singularity:/# umount /dev
```

```
root@Singularity:/# exit
```

Singularity blog #

Vytvorte si svoju vlastnú distribúciu

Napísal intense

Sobota, 17 Marec 2012 18:55 - Posledná zmena Sobota, 17 Marec 2012 19:07

Následne prejdeme k procesu znovu zostavenia iso obrazu, filesystem.package môže byť v iných distribúciach ako je napr. ubuntu nahradené za filesystem.manifest!:

```
Singularity blog # chmod +w remaster/live/filesystem.packages
```

```
Singularity blog # chroot root dpkg-query -W --showformat='${Package} ${Version}\n' >  
remaster/live/filesystem.packages
```

Vybudujeme nový filesystem.squashfs, dajte si caffè/vodku/, lebo toto potrvá dosť dlhú dobu:

```
Singularity blog # rm remaster/live/filesystem.squashfs
```

```
Singularity blog # mksquashfs root remaster/live/filesystem.squashfs
```

Ak chcete tak môžete pre mksquashfs použiť switch -b a nastaviť väčšiu veľkosť blokov (napr. 1048576) takže výsledný súbor bude menší za cenu dlhšej komprimácie. Môžete taktiež meniť

Vytvorte si svoju vlastnú distribúciu

Napísal intense

Sobota, 17 Marec 2012 18:55 - Posledná zmena Sobota, 17 Marec 2012 19:07

spôsob použitej kompresie, default je zvyčajne lzma kompresia. Ak ale zmeníte použitú kompresiu tak sa uistite že má podporu v kerneli daného systému inak vám bude výsledná aj keď menšia veľkosť na figu keď nenabootujete. ;-)

Ten kto si pozorne všimol že čo sme teraz vlastne za kroky spravili možno postrehol že podobným spôsobom sa dá zabaliť nainštalovaný systém a vytvoriť z neho live distro. To je už ale mierne komplikovanejšie a samozrejme si to vyžaduje pomerne dosť také zásahy do systému. Ako som už vyššie spomínal, ak ste robili niečo s kernelom tak teraz nahradte initrd, prípadne aj vmlinuz novým čo ste si vygenerovali inak vám nemusí vôbec systém bootnúť, debian má originál v priečinku live/ tak ich tam nahradte. Niektoré distribúcie obsahujú súbor filesystem.size, tento súbor obsahuje veľkosť SquashFS systému, ak máte takýto súbor (napr. ubuntu based distrá) tak aktualizujte túto veľkosť následovne:

```
printf $(du -sx --block-size=1 root | cut -f1) > remaster/live/filesystem.size
```

Keďže sme upravili live systém, zmenili sme taktiež kontrolné súčty jeho súborov, tie musíme teraz aktualizovať inak overovanie poškodenia disku zlyhá a bude hlásiť chyby na disku:

```
Singularity blog # cd remaster/
```

```
Singularity remaster # rm md5sum.txt
```

```
Singularity remaster # find -type f -print0|xargs -0 md5sum|grep -v isolinux/boot.cat |tee md5sum.txt
```

Vytvorte si svoju vlastnú distribúciu

Napísal intense

Sobota, 17 Marec 2012 18:55 - Posledná zmena Sobota, 17 Marec 2012 19:07

Teraz nám už len zostáva to všetko späť zabaliť do iso súboru, **bacha! nezabudnite tú bodku na konci!** :

```
Singularity remaster # mkisofs -D -r -V "SuperDistro" -cache-inodes -J -l -b isolinux/isolinux.bin  
-c isolinux/boot.cat -no-emul-boot -boot-load-size 4 -boot-info-table -o ../superdistro.iso .
```

```
Singularity remaster # cd ..
```

```
Singularity blog # ls
```

```
debian.iso mounter remaster root superdistro.iso
```

Vidíme že tam máme náš superdistro.iso, ktorý už nedočkavo čaká aby sme ho nabootovali cez qemu, virtualbox alebo dačo iné. Ak ste nič v systéme nepokašľali alebo máte miesto toho priazeň veľkého TUXa tak by ste mali bez problémov nabootovať do vášho novoupraveného systému. :-) Nebolo to ani veľmi ťažké že? Takýmto spôsobom si môžete vytvoriť vlastnú distribúciu alebo napríklad nejaké servisné CD čo bude obsahovať všelijaké užitočné nástroje. Prajem veľa šťastia s úpravou a tvorbou vašich distribúcií. 🍀