



## Sumário

<b>Breve comentário.....</b>	<b>2</b>
<b>Pré-requisitos.....</b>	<b>2</b>
<b>Procedimentos.....</b>	<b>2</b>
1)Criando um arquivo de swap.....	2
3)Baixando o ISO.....	3
4)Ponto de Montagem.....	3
5)Montando a imagem ISO.....	4
<b>Customizando seu (K/X/ED)Ubuntu.....</b>	<b>5</b>
1) Acessando a imagem a ser customizada.....	5
2) Abrindo uma interface gráfica dentro do Xnest.....	5
3) Instalando e removendo programas.....	6
<b>Fechando programas e preparando para gerar o ISO.....</b>	<b>7</b>
1) Hora da faxina e liberação de espaço no Live-CD.....	7
2)Desmontando.....	7
<b>Fechando a Imagem.....</b>	<b>7</b>
1) Gerando o arquivo squashfs.....	7
2) Recriação do MD5.....	8
<b>Testando sua imagem sem queimar o CD.....</b>	<b>9</b>
<b>Referência Bibliográfica:.....</b>	<b>9</b>
<b>Glossário.....</b>	<b>10</b>
<b>Sobre o autor.....</b>	<b>10</b>



## **Breve comentário**

Essa documentação é fruto de um trabalho acadêmico sobre geração de Live-CD. É comum vermos algum cliente ou amigo, com dificuldade na instalação do flash, mp3, divx, DVD, rádio pela internet, sem contar que o acesso a banda larga para realizar download não é realidade para todos. Com esse intuito de descobrir os mistérios de uma customização comecei minha pesquisa. Procurei a forma mais prática para customizar o (K/X/ED)Ubuntu, sendo o Kubuntu minha distribuição de uso diário.

Um ponto favorável, lembrado pelo meu professor Leocádio é a utilização desse tutorial por empresas ou faculdades que usam sistemas de distribuição de imagens. Mesmo tendo os computadores do fabricante, nem sempre eles são do mesmo modelo, o que gera retrabalho as equipes de suporte. Outro ponto importante de grande ajuda do meu professor foi na questão do licenciamento Ubuntu. A mesma não proíbe a customização desde que você siga o que ela denomina de "License Policy". O bacana é que ela autoriza inclusive a cobrar pela customização da distribuição. Mais detalhes pode ser visto na url abaixo.

<http://www.ubuntu.com/ubuntu/licensing?highlight=%28license%29> (consultada Fev/2007)

## **Pré-requisitos**

- Uma computador Linux com suporte à Squashfs no kernel;
- ISO da distribuição (Ubuntu, Kubuntu, Xubuntu, Edubuntu);
- Instalação dos pacotes squashfstools, mkisofs , Xnest;
- 4 pontos de montagem;
- Acesso a Internet (será necessário para instalar novos programas);
- 5 GB de espaço Livre em disco (mínimo);
- 512 para memória RAM (recomendado);
- 1 GB de SWAP (O swap irá conter a maior parte da imagem do CD sem a compactação, por isso precisamos aumentar. Pode ser necessário 6 GB de espaço livre em seu computador, caso não possua 1 GB de SWAP nativo).

## **Procedimentos**

**Todos os procedimentos abaixo deverão ser executados em um terminal com o usuário "root". Com o comando `su` no terminal é possível passar para o usuário "root" (recomendado) ou então utilize o comando `sudo` antes de cada comando.**

### ***1) Criando um arquivo de swap.***

Caso não tenha tanto swap em seu linux (1 GB), saiba que é possível criar um "arquivo" para ser usado como swap, sem a necessidade de uma partição de disco para essa função. Execute o comando abaixo para criar o arquivo swap.

```
# dd if=/dev/zero of=/tmp/swap count=1 bs=1000000000
```

O comando `dd` é bem conhecido dos usuários de linux, o parâmetro `if=/dev/zero` significa que serão copiados a partir do dispositivo especial `/dev/zero`, para preencher apenas com zeros. O `of=/tmp/swap` indica o arquivo que será utilizado como swap. O parâmetro `count` é o numero de blocos e o parâmetro `bs` o numero de bytes que serão copiados para o arquivo (1 GB).

# Customizando o (K/X/ED)Ubuntu Linux Live-CD

Devemos agora formatar o arquivo no formato de swap e habilita-lo on-line, sem a necessidade de configurações adicionais na fstab.

```
#mkswap /tmp/swap (formata o arquivo swap criado)
#swapon /tmp/swap (habilita o arquivo swap para o uso)
```

Para conferir se funcionou use o comando "free" na contagem de swap deve ter 1GB a mais. Exemplo:

## Depois de habilitar o swap ficou com 1.256 MB

	total	used	free	shared	buffers	cached
Mem:	775632	761680	13952	0	4996	468012
-/+ buffers/cache:		288672	487420			
Swap:	<b>1286144</b>	248	1285896			

## Antes de habilitar o swap possuía 256 MB

	total	used	free	shared	buffers	cached
Mem:	775632	761492	14140	0	5016	468264
-/+ buffers/cache:		288212	487420			
Swap:	<b>262114</b>	248	261896			

## 2) Instalação de programas necessários

Resolvido o problema do swap vamos instalar os programas necessários. Isso é um procedimento simples no ubuntu (ou variantes de debian) através do comando apt-get. Execute os comandos abaixo.

```
#apt-get update
#apt-get install squashfs-tools xnest mkisofs
```

## 3)Baixando o ISO

Primeiro é preciso obter uma cópia da imagem ISO da distribuição favorita (Ubuntu, Kubuntu, Xubuntu, Edubuntu) Uma bom local para encontrar é no site [www.ubuntu.com](http://www.ubuntu.com). Essa imagem ISO será necessária no processo de customização.

Observação: O seu kernel precisa ter suporte ao squashfs para realizar essa customização. O kernel instalado no ubuntu já possui essa opção como padrão.

## 4)Ponto de Montagem

Crie os 4 pontos de montagem que serão utilizados durante todo o processo. No nosso exemplo iremos montar 3 no diretório /mnt para centralizar e um em outro disco. Execute os comandos abaixo para criar os pontos de montagem.

```
#cd /mnt
#mkdir teste teste2 projeto
```

Posso criar o quarto ponto de montagem no diretório /mnt/teste3, porém no nosso exemplo irei usar outro disco que tenho formatado com Ext3, montado em /media/hdb1. Se quiser criar todos diretórios em outro disco fique a vontade.

# Customizando o (K/X/ED)Ubuntu Linux Live-CD



## Funções dos pontos de montagem

- 1º) /mnt/teste – montagem do arquivo ISO original, gasta em média 700 MB;
- 2º) /mnt/teste2 – Montagem do arquivo ISO descompactado, gasta em média 1,5 GB;
- 3º) /media/hdb1 – Local de alteração dos programas e configurações , gasta em média 2 GB, por isso usei outro disco nesse tutorial;
- 4º) /mnt/projeto – Necessário para gerar o arquivo squashfs e o ISO customizados, gasta em média 700 MB.

## 5)Montando a imagem ISO

De posse da imagem ISO da distribuição favorita, montaremos a mesma no 1º ponto de montagem criado acima, utilize o comando descrito abaixo.

```
#mount -o loop /(path_da_ISO)/kubuntu-6.10-desktop-i386.iso /mnt/teste
```

O próximo passo é a montagem do arquivo /mnt/teste/casper/filesystem.squashfs no 2º ponto de montagem. Montaremos com os comandos abaixo.

```
#cd /mnt/teste/casper/ (é necessário entrar nesse diretório)
#mount -o loop -t squashfs filesystem.squashfs /mnt/teste2
```

Dentro do diretório /mnt/teste2 terá disponível todo o filesystem da distribuição descompactado, no qual teremos que copiar para o 3º ponto de montagem.

No nosso exemplo utilizamos o diretório /media/hdb1 (sistema de arquivos ext3). A formatação do sistema de arquivos pode ser qualquer uma nativa no Linux, ou seja não serve FAT, NTFS. Copie todo os arquivos para /media/hdb1 com o comando.

```
#cd /mnt/teste2
#cp -pRf * /media/hdb1 (este procedimento demora um pouco)
```

## 6)Últimos detalhes na montagem

Copie o arquivo resolv.conf e hosts da sua distribuição corrente para o 3º ponto de montagem. Isso possibilitará o acesso a internet e conseqüentemente o uso do apt-get. Se preferir você configurar manualmente em /media/hdb1/etc/resolv.conf e /media/hdb1/etc/hosts. Para copiar o seu atual resolv.conf e hosts execute os comandos abaixo.

```
#cp /etc/resolv.conf /media/hdb1/etc
#cp /etc/hosts /media/hdb1/etc
```

Usaremos o comando chroot para apontar o diretório /media/hdb1 como se fosse a raiz de nossa distribuição. Precisamos também montar os diretórios dinâmicos para o funcionamento do sistema em chroot, execute os comandos abaixo.

```
#chroot /media/hdb1
#mount -t proc none /proc
#mount -t sysfs none /sys
#cd /dev
#MAKEDEV pty (o comando MAKEDEV é escrito em maiúscula)
```

# Customizando o (K/X/ED)Ubuntu Linux Live-CD

Agora você já está com um sistema funcional possível de usar o apt-get ou qualquer gerenciador de pacotes para instalar e remover programas.

## **Customizando seu (K/X/ED)Ubuntu**

**(Os procedimentos aqui descritos, devem ser executadas em uma interface gráfica como KDE, GNOME, XFCE ou qualquer outro de sua preferência.)**

### ***1) Acessando a imagem a ser customizada***

Para acessar a distribuição a ser customizada é necessário usar o Xnest. O Xnest é uma janela dentro do seu servidor X, que vai atender requisição de alguns programas que forem direcionados para ele.

Abra outro terminal para seu usuário comum, e não feche o **terminal** em que estamos com o chroot. No terminal onde está logado com "seu usuário comum" digite.

```
$Xnest :1 -ac &
```



(atenção execute esse comando com o seu usuário comum e não com o root)

O Xnest será aberto na sua tela usando o display 1 (a sua atual interface gráfica usará o display 0), se o display 1 já estiver em uso, troque o comando acima para :2 e assim por diante.

Volte a janela de terminal onde estávamos trabalhando com o chroot, nela vamos configurar uma variável de ambiente para que todo aplicativo gráfico procure o Xnest.

```
#export DISPLAY=localhost:1
```

### ***2) Abrindo uma interface gráfica dentro do Xnest***

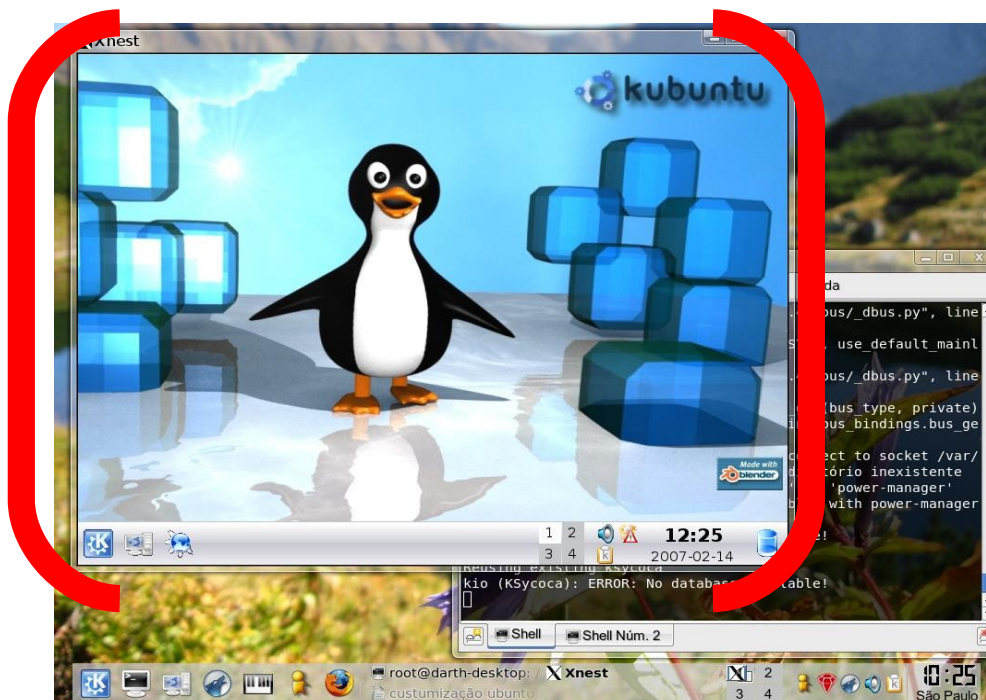
O Xnest está no display 1, qualquer programa gráfico que abrirmos na tela do terminal que está com chroot será aberta dentro do Xnest. Vamos invocar um gerenciador de janelas de sua preferência.

Para usar o KDE, execute esse comando no terminal que está com chroot.

```
#startkde &
```

Para o Gnome, , execute esse comando no terminal que está com chroot.

```
#gnome-session &
```



O gerenciador de janelas será inicializado na janela do Xnest. Você já pode começar sua customização, pela janela do Xnest. Caso o gerenciador de janelas não abra, certifique que os arquivos `/etc/hosts` e `/etc/resolv.conf` foram copiados.

### ***3) Instalando e removendo programas***

O resto do procedimento de customização vai ser provavelmente o acréscimo de outros repositórios no arquivo `/etc/apt/source.list` para poder baixar mais programas e fazer sua instalação. Utilize o prompt e use a sua ferramenta de gerenciamento de pacote favorita para essa função (`synaptic`, `adept`, `aptitude`, `apt`, `smart`, etc).

### **Faça sua customização!**

As alterações feitas no KDE (no usuário `root`), devem ser copiadas para o diretório `/usr/share/kubuntu-default-settings/kde-profile/default`. Copie sua customização feita no KDE com os comandos abaixo.

```
#cd /root/.kde/share
# cp -pRf * /usr/share/kubuntu-default-settings/kde-profile/default/share
```

Feito o procedimento acima, toda vez que criar um usuário novo ele vai pegar essas configurações. Poderíamos também copiar para `/etc/skel`, mas temos que evitar a redundância, pois devemos economizar espaço ao máximo se quisermos montar um Live-CD.

Em meus testes a personalização do Gnome deve ser copiada para o `/etc/skel`. Leve no mínimo os arquivos `.gnome2*` e `.gconf*`



## **Fechando programas e preparando para gerar o ISO**

Após a sua customização ser finalizada é hora de começar a desfazer o que construímos até esse ponto. Se você navegou pela janela do Xnest com seu browser favorito, lembre de limpar o cache. Se possível faça isso agora.

Feche a sua interface gráfica (KDE, Gnome, XFCE) aberta no Xnest (logout). Lembre de utilizar apenas o terminal que esta com o chroot.

### **1) Hora da faxina e liberação de espaço no Live-CD**

a) Vamos limpar todos aqueles arquivos desnecessários para poder ter mais espaço no Live-CD. Primeiro vamos limpar o cache do apt-get. Digite o seguinte comando.  
`#apt-get clean`

b) Depois vamos remover arquivos temporários.  
`#cd /tmp`  
`#rm -rf *`

c) Os arquivos de log também são desnecessários no Live-CD.  
`#cd /var/log`  
`#rm -rf *`

d) Apague os arquivos `/etc/resolv.conf` e `/etc/hosts` com o comando abaixo. Ambos não existem no Live-CD e foram copiados por nos anteriormente.  
`#rm -rf /etc/resolv.conf`  
`#rm -rf /etc/hosts`

e) Apague os arquivos de configuração do desktop que foram criados no `/root`.  
`#cd /root`  
`#rm -rf .bash_history .kde (se for gnome é .gnome2*, .gconf*)`

### **2)Desmontando**

Feche o gerenciador de janelas Xnest. Em seguida desmonte o `/proc` , o `/sys` e saia do chroot. Digite os comandos abaixo para executar esse procedimento.

```
#umount /proc
#umount /sys
#exit <-- Esse comando sai do modo chroot.
```

## **Fechando a Imagem**

### **1) Gerando o arquivo squashfs**

Todos os comandos abaixo serão executados com o usuário "**root**" do seu Linux.

É preciso fechar a imagem em um novo arquivo squashfs de novo para coloca-lo no CD. Para execução desse processo precisaremos do squashfstools (instalado anteriormente). Primeiro vamos copiar o conteúdo do CD montado em `/mnt/teste` para um diretório onde é possível escrever. Digite o comando abaixo.

# Customizando o (K/X/ED)Ubuntu Linux Live-CD



```
#cp -Rf /mnt/teste/* /mnt/projeto/
```

Neste ponto será necessário substituir o arquivo filesystem.squashfs original por uma imagem da nossa customização. Primeiro iremos apagar o original e em seguida iremos criar o nosso arquivo filesystem.squashfs resultado da nossa customização. Para criar o sistema de arquivos em Squashfs, utilize os comandos abaixo.

```
#rm /mnt/projeto/casper/filesystem.squashfs (apagando o original)
#mksquashfs /mnt/hdb1 /mnt/projeto/casper/filesystem.squashfs
```

O comando mksquashfs cria a partir do conteúdo de /mnt/hdb1 um arquivo filesystem.squashfs com seu conteúdo no formato squashfs. **Esse procedimento demora um pouco.**

Observação: Cuidado o conteúdo do CD não pode passar de 700MB, caso o contrário terá de se gravar em um DVD. Para verificar o tamanho do CD use o comando abaixo dentro do diretório do projeto (após o uso do comando acima).

```
#cd /mnt/projeto
#du -sh
```

O CD do Ubuntu e derivados possui uma série de programas para windows que ocupam em torno de 70MB. Apagando irá liberar mais espaço no CD. Execute os comandos abaixo para limpar esses arquivos .

```
#rm -rf /mnt/projeto/bin
#rm -rf /mnt/projeto/programs
#rm -rf /mnt/projeto/autorun.inf
```

## 2) Recriação do MD5

Recrie o md5sum para poder testar a imagem. Isso é feito na gravação do CD.

```
#rm /mnt/projeto/md5sum.txt
#cd /mnt/projeto/ && find . -type f -print0 | xargs -0 md5sum > md5sum.txt
```

## 3) Gerando a nova imagem ISO

Gere a imagem ISO. Entre dentro do diretório do projeto e **digite em uma única linha o comando mkisofs.**

```
#cd /mnt/projeto/
#mkisofs -r -V "kubuntu-6.10" -cache-inodes -J -l -b isolinux/isolinux.bin -c
isolinux/boot.cat -no-emul-boot -boot-load-size 4 -boot-info-table -o
/path_para_salvar_ISO/kubuntu-6.10.1-desktop-i386-custom.iso .
```

Observação: Fique atento e lembre de substituir o texto acima "(local onde deseja gerar o iso)" por uma partição com espaço suficiente para gerar o ISO. O título do CD está entre as aspas, ele vai criar a partir do conteúdo do diretório corrente, por isso o ponto "." no final do comando, não esqueça dele.



# Customizando o (K/X/ED)Ubuntu Linux Live-CD

Em poucos instantes você terá sua imagem que pode ser gravada em um CD. Com o CD pronto, agora você pode desmontar o swap e apagar o arquivo criado no principio desse tutorial com o comando.

```
#swapoff /tmp/swap  
#rm /tmp/swap (isso irá limpar o arquivo gerado para servir de swap extra, 1 GB)
```

Desmonte as imagens montadas (loop) em /mnt/teste e /mnt/teste2

```
#umount /mnt/teste2  
#umount /mnt/teste
```

Com o arquivo ISO criado você pode queimar o CD e fazer a instalação do sistema. Um pequeno bug foi encontrado em nossos testes. Mesmo que você instale todos os pacotes de internacionalização (pacotes que traduzem os programas para diversas línguas), na hora que você for instalar o CD, no final do processo de instalação o programa instalador do ubuntu vai tentar baixar esses pacotes. Esse problema se resolve facilmente, cancelando o processo ou simplesmente desconectando da internet.

## **Testando sua imagem sem queimar o CD**

Para testar sua customização é necessário ter instalado o QEMU. Caso necessite instalar o Qemu use o comando abaixo.

```
#apt-get install qemu
```

Agora para testar sua imagem use o comando abaixo.

```
#cd /path_onde_salvou_ISO/  
#qemu -cdrom kubuntu-6.10.1-desktop-i386-custom.iso -boot d
```

## **Referência Bibliográfica:**

- Eduardo Bacchi  
Tutorial sobre Xnest  
<http://www.noticiaslinux.com.br/eduardo/xnest.php>
- Ednei Alves dos Santos e Felipe Brunelli de Andrade  
Remasterizando o KNOFIS  
<http://basalto.ifsc.usp.br/KNOFIS/remaster.html>
- Jeremy Visser  
LiveCDCustomization  
<https://help.ubuntu.com/community/LiveCDCustomization/6%2e06>



## Glossário

(informações retiradas dos sites <http://www.guiadohardware.net/termos>, e <http://planeta.ubuntu.com.br>)

**UBUNTU** - Ubuntu é um sistema operacional completo, baseado em Linux, distribuído livremente para uso em desktops, uso corporativo e servidores.

**ISO** - Este é o sistema de arquivos padrão para CD-ROMs, que permite que o CD seja lido em PCs rodando qualquer sistema operacional.

**SQUASHFS** - É um sistema de compressão e leitura que permite uma compressão de todo sistema de arquivos ou diretórios, e você pode utilizá-lo para montar estes arquivos comprimidos em devices, diretamente, ou utilizando um loop.

**KERNEL** - É o núcleo, encarregado de controlar o acesso à memória de demais componentes de Hardware, gerenciar os programas abertos, dividir o tempo de processador entre eles, etc. É a base, sobre a qual rodam as demais partes do sistema operacional, drives de dispositivo e programas.

**RAM** - Random Access Memory. Memória de acesso aleatório, é a memória mais usada não apenas em micros PCs, mas na maioria dos computadores. É a tecnologia de memória mais barata, mas tem a desvantagem de ser volátil.

**SWAP** - Arquivo de troca. É um arquivo, criado no disco rígido, usado pelo sistema operacional para simular memória RAM, sempre que a memória física se esgota. Veja também: memória virtual

**FILESYSTEM** - um conjunto de estruturas lógicas e de rotinas, que permitem ao sistema operacional controlar o acesso ao disco rígido. Diferentes sistemas operacionais usam diferentes sistemas de arquivos.

**MD5** - É um algoritmo que gera uma "assinatura" de um arquivo qualquer, um código de 32 bits obtido a partir da soma de todos os bits contidos no arquivo.

**CHROOT** - é um programa que transfere a uma partição montada com o chroot a característica de arquivo raiz.

Este tutorial está sob Licença GPL 2.

## Sobre o autor

Alexandre da Silva Costa - Técnico em Informática e Profissional em Linux, aluno do curso de Tecnologia da Informação do Centro Universitário de Belo Horizonte - UNI-BH

Sugestões e críticas podem ser enviadas para [anakinpendragon@gmail.com](mailto:anakinpendragon@gmail.com)

Revisado por Antônio Ricardo Leocádio Gomes, Prof.  
Centro Universitário de Belo Horizonte - UNI-BH  
Belo Horizonte, Minas Gerais