LINUX FROM SCRATCH

Author: Katumbi Georges
Summary: Make your own linux distribution

Table of Contents

Chapter I: Forewords

Linus Torvalds Quotes

> No-one has ever called me a cool dude. I'm somewhere between geek and normal.

> I'm perfectly happy complaining, because it's cathartic, and I'm perfectly happy arguing with people on the Internet because arguing is my favourite pastime - not programming.

> Hey, I'm a good software engineer, but I'm not exactly known for my fashion sense. White socks and sandals don't translate to 'good design sense

> The Linux philosophy is Laugh in the face of danger. Oops. Wrong One. Do it yourself. Yes, that's it.

> I actually think that I'm a rather optimistic and happy person; it's just that I'm not a very positive person, if you see the difference.

> See, you not only have to be a good coder to create a system like Linux, you have to be a sneaky bastard too.

> I do get my pizzas paid for by Linux indirectly.

> To be a nemesis, you have to actively try to destroy something, don't you? Really, I'm not out to destroy Microsoft. That will just be a completely unintentional side effect.

See? Be a good Linus-like guy.

Chapter II: Introduction

Welcome to linux from scratch . In this subject, you have to build a basic, but functional, linux distribution.

This subject is not about Kernel programming, but it's highly related.

This distro will be the base for all your kernel projects, because all your kernel-code will be executed here, on your distro.

Try to implement what you want/need to. This is your userspace, take care of it!

Chapter III: Goals

- Build a Linux Kernel
- Install some binaries (See the list below)
- Implement a filesystem hierarchy compliant with the standards
- Connect to the Internet

Chapter IV: General Instructions

The Links

- The Bible
- How to build a Kernel
- Autotools

Instructions

- For this subject, you must use a virtual machine, like VirtualBox or VMWare.
- Though it is not REQUIRED, you SHOULD read this and that right now. Keep those standards in mind. You won't be graded on your compliance with them, but still, it would be good practice.
- You must use a kernel version >= 4.0. Stable or not, as long as it's a 4.0 >= version.
- The kernel sources must be in `/usr/src/kernel-$(version)`
- You must use at least 3 different partitions:
  - root
  - /boot
  - swap partition
- Your distro must implement a kernel_module loader, like udev.
- The kernel version must contain your student login in it. Example: 'Linux kernel 4.1.2-<student_login>'
- The distribution hostname must be your student login
- You're free to choose between a 32 or 64-bit system.
- You must use a software for central management and configuration, like SysV or SystemD.
- Your distro must boot with a bootloader, like LILO or GRUB.
- The kernel binary located in /boot must be named: `vmlinuz-<linux_version>-<student_login>

Chapter V: Mandatory Part

Packages to Install

Note: The following versions are known to work together correctly. However, you are free to use the versions you want. Some packages below (vim, bash, grub, udev) are examples. Feel free to change them b

Base System

- Acl (2.2.52)
- Attr (2.4.47)
- Autoconf (2.69)
- Automake (1.15)
- Bash (4.3.30)

- Bc (1.06.95)
- Binutils (2.25.1)

Development Tools
- Bison (3.0.4)
- GCC (5.2.0)
- Make (4.1)
- Patch (2.7.5)
- Perl (5.22.0)

System Utilities
- Bzip2 (1.0.6)
- Coreutils (8.24)
- E2fsprogs (1.42.13)
- Findutils (4.4.2)
- Grep (2.21)
- Gzip (1.6)

Additional Components

- GRUB (2.02 beta2)
- Vim (7.4)
- Util-linux (2.27)
- Sysklogd (1.5.1)
- Shadow (4.2.1)

Complete package list continues...

Chapter VI: Bonus Part

You have a stable system? Nice. Now let's have some fun! Install whatever you want.
Any software, GUI, ANYTHING.

Make this system yours, with your touch.

Special points for an X Server, and window managers / desktop environments, like GNOME / LXDE / KDE / i3 / dwm

Chapter VII: Turn-in and Peer-evaluation

Turn your work in using your GiT repository, as usual. Only work present on your repository will be graded in defense.

For obvious reasons, you will not push your entire virtual machine but a checksum of your disk image instead.

That can be done with something like:

```
    bash
georges < disk.vdi
```

Keep your disk image somewhere for the peer-evaluation.

End of Document