



Linux Kernel

Let's contribute!

Presented by
Levente Kurusa
Fedora Project Hungary
<levex@linux.com>

Topics

1. Quick introduction
2. Random configurations
3. Kernel testing
 - a. Stable kernels
 - b. RC kernels
4. Checkpatch patches

Quick introduction

Why?

- Linux powers the world!
 - Servers! The cloud! Desktop? Maybe this year?
- Open source contribution is always a big plus when it comes to job search!
- Linux is cool!
- Why not?

Some statistics

- 86% say Linux advanced their career
- 93% of HR will recruit a Linux pro next year
- 75% already received a call this year!
- Interesting projects!
- Greater pay check! (+10% this year)

First steps

- Fetch the Linux kernel source code
- For random configurations: linux-next
 - <http://git.kernel.org/pub/scm/linux/kernel/git/next/linux-next.git>
- For stable testing: linux-stable
 - <http://git.kernel.org/pub/scm/linux/kernel/git/stable/linux-stable.git>
- For Rcs, simply clone down Linus' tree
 - <http://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git>

Eudryptula Challenge

- <http://eudryptula-challenge.org/>
- Series of programming tasks
- Hard, but fun!

Linux-next

- The current state of the kernel
- Merged together by Stephen Rothwell
- An excellent opportunity to start
- Default configurations are tested

Linux-stable

- Collection of important patches to old kernel release
- Backports needed!
- Maintained by Greg KH and others
- `Documentation/stable_kernel_rules.txt`

Random configurations

What are randconfigs?

- Randomized configurations of the kernel
- Often have a lot of problems
 - GCC warnings
 - GCC errors
 - kconfig errors/warnings
- They sometimes don't even boot
- Extremely useful!

Your first randconfig



```
$ make randconfig
```

- Completely nonsense config!

- Try to build it:

```
$ make -j`nproc`
```

Possible errors

- Undeclared functions
- Kconfig circular dependencies
- Boot failure
Test in emulator!

Kernel testing

Run the latest kernel

- Quite easy!

```
$ make defconfig
```

```
$ make localmodconfig
```

```
$ make -j`nproc`
```

```
# make headers_install
```

```
# make install
```

```
# reboot
```

Actual testing

- If you test linux-stable
 - ... and things break: Report it!
 - ... if nothing breaks: Break it!
 - (repeat)
- If you test linux-next
 - ... it's risky to try on your computer

Actual testing

- I use my own tiny project.

LATTE – Linux Automatic Testing Tool for Expert
(<https://github.com/levex/latte>)

- Emulators are great.
- Benchmarks are also fun.
- perf

perf



perf list

perf record <command>

perf report

perf stat -e <event>

Sending your first patch

- Also known as: Checkpatch patches

Checkpatch?

- `scripts/checkpatch.pl`
- Checks for common errors in the patch
... and files with the `--file` switch
- Not just coding style, but bugs as well!

Getting maintainers

- `scripts/get_maintainer.pl`
- Gives you a list of maintainers for that particular patch or file
- ... or you can manually browse through the MAINTAINERS file

Creating the patch



```
$ git reset --hard HEAD
```

```
[work...]
```

```
$ git diff
```

```
[review work...]
```

```
$ git commit -as
```

```
$ git format-patch -1
```

... and then the hardest part

Sending it



- 9 out of 10 will fail this step
- `git send-email`
- Thunderbird and such can do it as well, but it's hard to configure them properly.

Summary

- People on LKML are grumpy!
- Don't be afraid to ask
- RTFM
-

Any questions?



Contact:
<levex@linux.com>
@ilevex