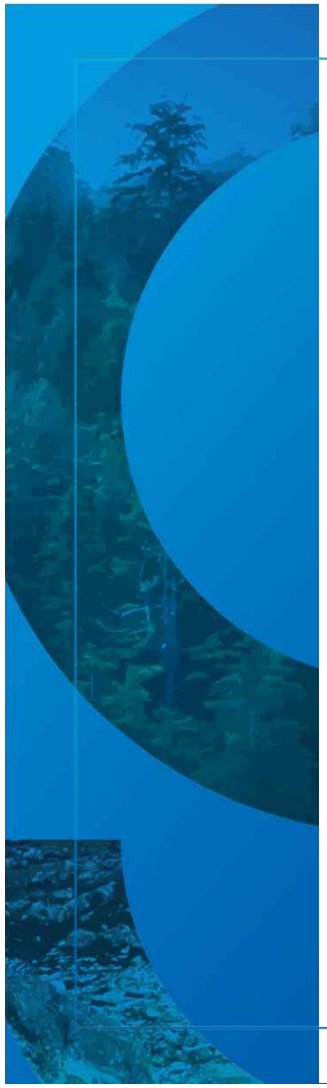

THE LINUX FOUNDATION
OPEN SOURCE SUMMIT
An Empirical Study of
an Advanced Kernel Tailoring
Framework
Junghwan Kang / ultract@nsr.re.kr
@ultractt
THE
LINUX
FOUNDATION

Contents

- Introduction
- Review
  - My Previous Work @ OSSummit NA 2017
- Advanced Features
- Demo
- Evaluation
- Discussion
- Conclusion



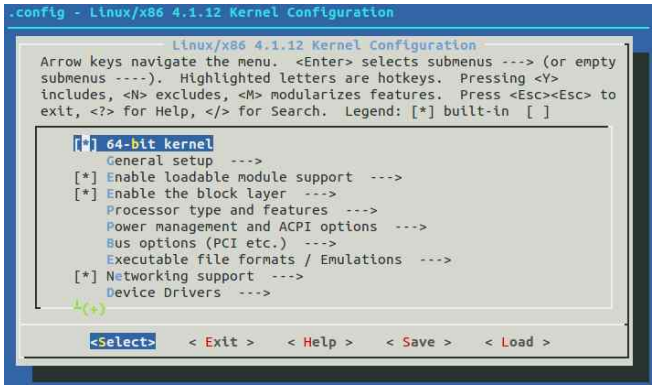
Introduction

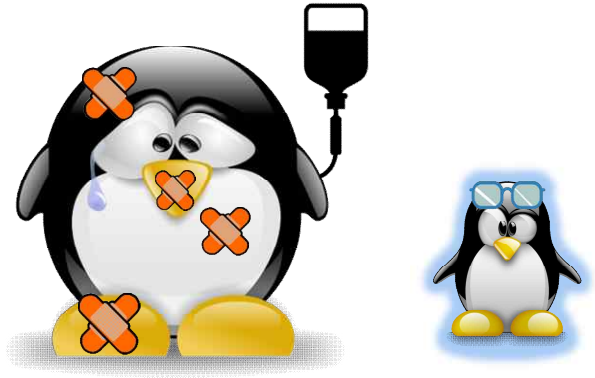
THE LINUX FOUNDATION

Introduction

- Motivations of my work
  - Minimize the attack surface of the Linux kernel
  - Automate the kernel configuration
  - Produce a stable tailored Linux kernel

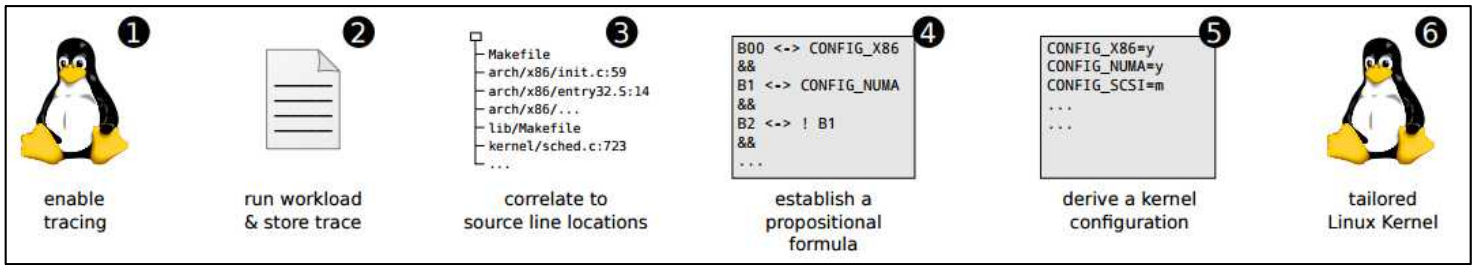
More than
12,000 Options
(Has Prompts)





Introduction

- (Previous work Ⅰ) Undertaker-tailor
  - Uses ftrace (Kernel function tracer)
  - Formulates dependency relationships among kernel configuration options
  - Uses SAT solver



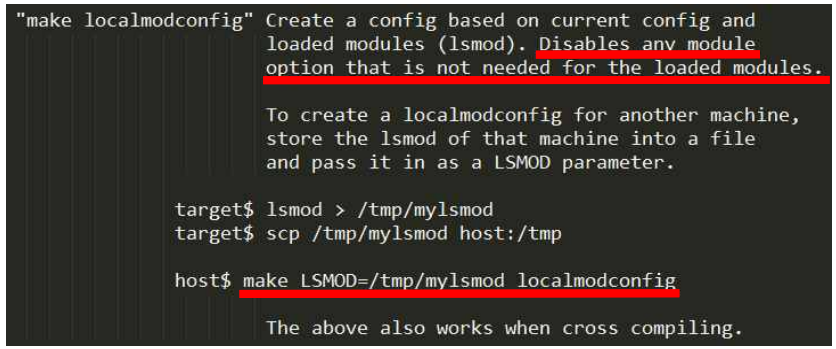
Workflow of Undertaker-tailor

Introduction

- (Previous work Ⅰ) Undertaker-tailor
  - Great! However, the tailored kernels failed to boot up…
  - Some bugs need to be fixed

Introduction

- (Previous work Ⅱ) Localmodconfig
  - Command for configuring the kernel
  - Very useful to reduce the # of kernel modules
    - Many of kernel modules removed
    - There are still unnecessary configuration options…



Introduction

- (Previous work Ⅲ) Kernel tailoring framework
  - Uses the undertaker-tailor with some fixes
  - Automates all workflow of the kernel tailoring
  - Makes candidates of the kernel configuration options to find the missing configuration options at the 1st tailored kernel configuration
  - Groups the candidates of the kernel configuration options to reduce the time for the kernel tailoring.
  - Finds out the missing configuration options among the candidate groups by looking into
    - Boot-up state, system logs, kernel modules and etc

Introduction

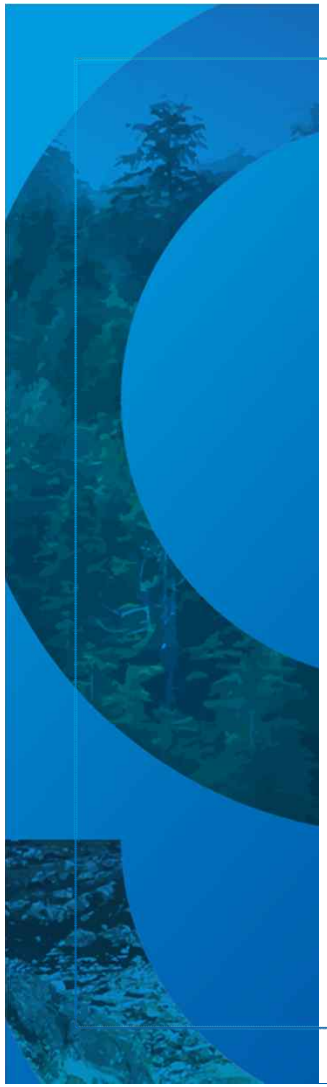
- (Previous work Ⅲ) Kernel tailoring framework
  - **I got the working tailored kernel!**
    - It was a little poor for supporting several applications and services
  - It sometimes failed to derive a tailored kernel
    - Caused by the dependency of the kernel configuration

Introduction

- (This work) **Advanced kernel tailoring framework**
  - Improves a stability
    - With fine-grained configuration options (Not Grouping)
    - Includes more various conditions to verify tailored kernels
      - Shows relationships with kernel configuration options

Introduction

- (This work) **Advanced kernel tailoring framework**
  - Supports for other Linux distributions
    - Debian
    - Ubuntu
    - …
  - Measures the performance between a tailored and original kernel
    - Lmbench (Micro-benchmark for Linux/UNIX/POSIX)
    - Phoronix-test-suite (Benchmark for Linux & Other Operating Systems)



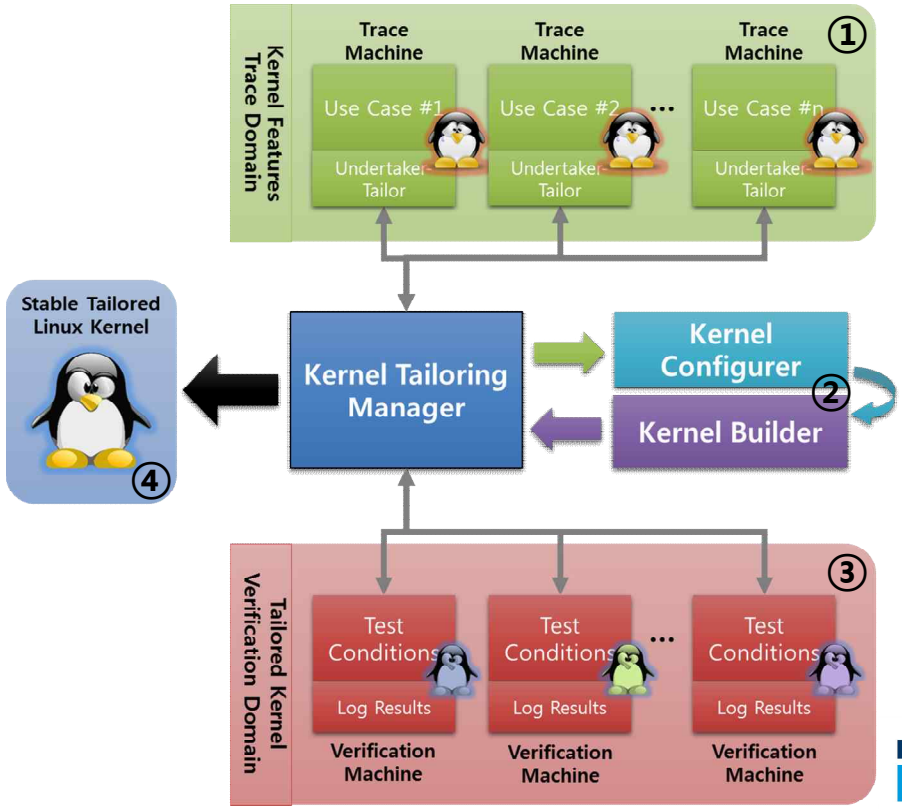
Review - My Previous Work

※ Details of My Previous Work are in a Presentation File at OSSummit NA 2017 ☺
(http://sched.co/BCsG)

THE LINUX FOUNDATION

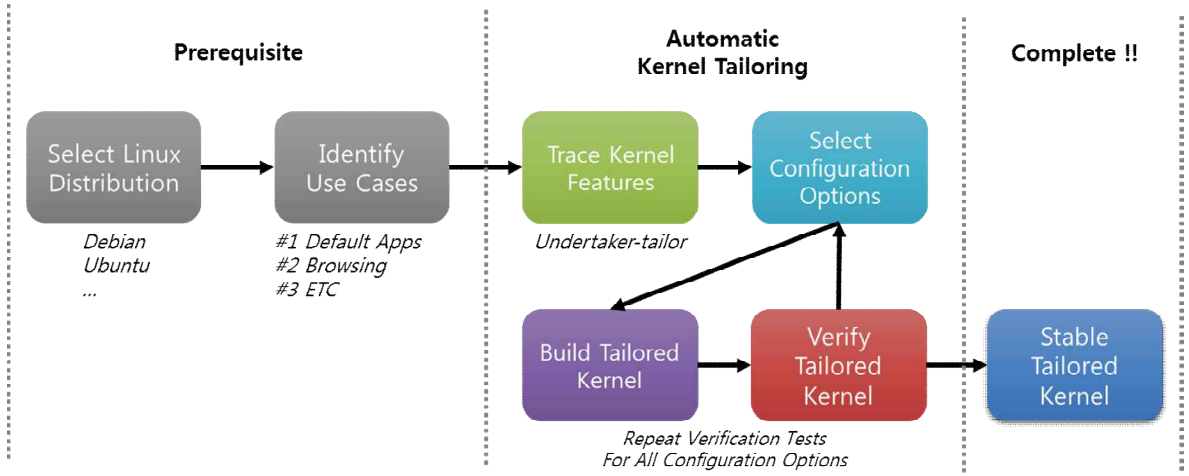
Review – My Previous Work

- Design
  - Architecture



Review – My Previous Work

- Design
  - Workflow

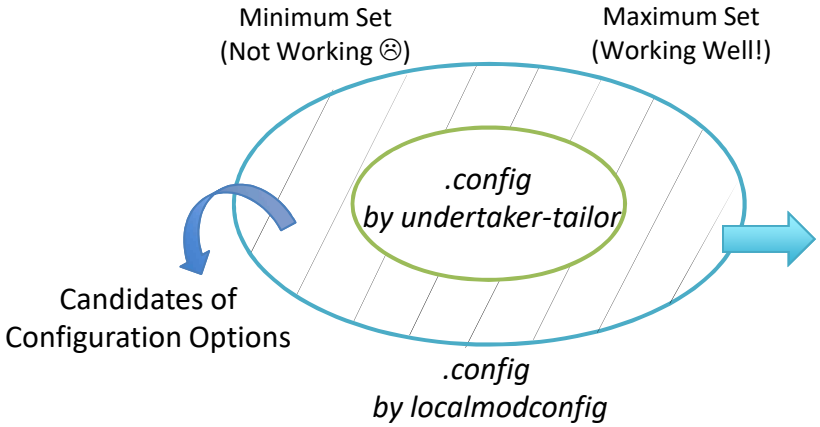


Review – My Previous Work

- Design
  - Kernel configurer
    - Makes the candidates of the kernel configuration options
      - To find the missing configuration options at the 1st kernel configuration by the undertaker-tailor
    - Groups the candidates for the time for the kernel tailoring
      - Reduces the number of the configuration options to test
        ※ test: Configure → Build → Verify a Tailored Kernel

Review – My Previous Work

- Design
  - Kernel configurer

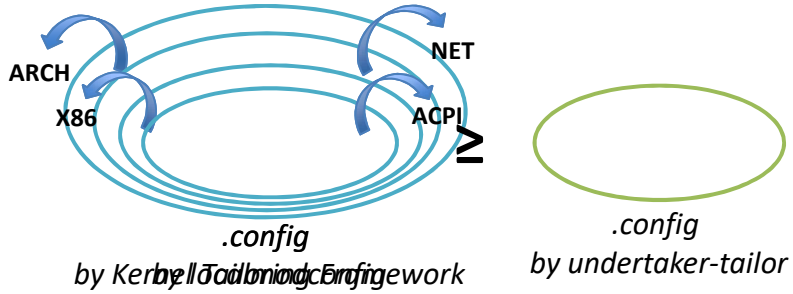


Select & Group the Configuration Options

```
# Grouping example :)
# Merge & Sort each groups of configuration

CONFIG_ARCH_MMAP_RND_COMPAT_BITS_MIN=8    # ARCH group
CONFIG_ARCH_MMAP_RND_COMPAT_BITS_MAX=16   #

CONFIG_NEED_DMA_MAP_STATE=y               # NEED group
CONFIG_NEED_SG_DMA_LENGTH=y               #

CONFIG_GENERIC_ISA_DMA=y                  # GENERIC Group
CONFIG_GENERIC_BUG=y                      #
```

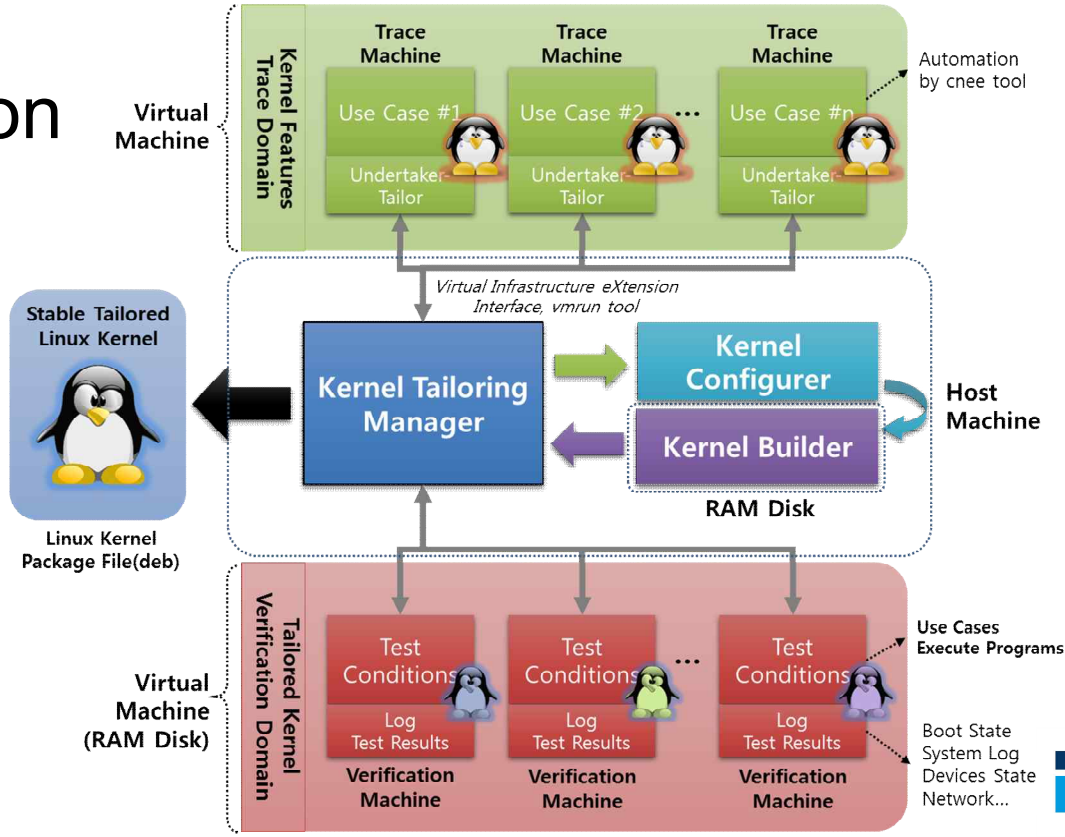
Review – My Previous Work

- Design
  - Kernel configurer



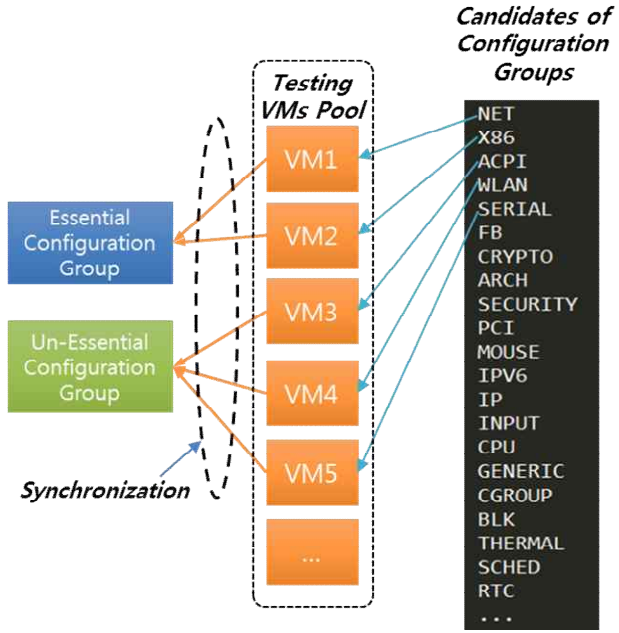
Review – My Previous Work

- Implementation



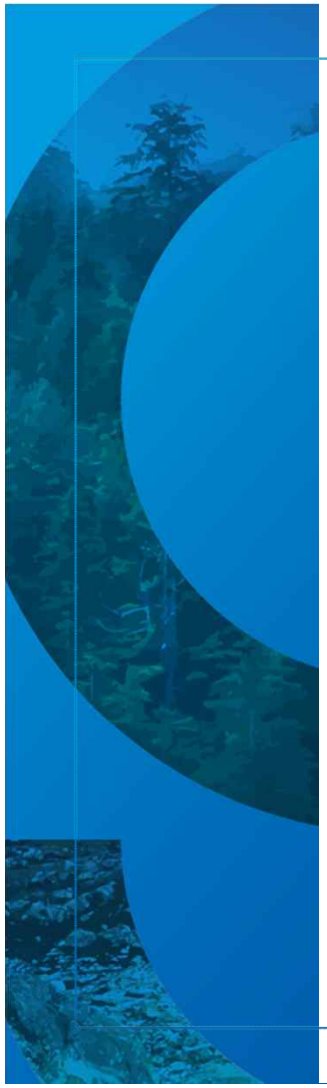
Review – My Previous Work

- Implementation
  - Multi-VMs for a verification
    - # of Maximum VMs: 5



Review – My Previous Work

- Evaluation
  - Elapsed time: about 5 hours (# of verification VMs: 5)
  - Kernel image size: about ½ ↓
  - # of kernel modules: 110/3269 ≒ 3.4 %
  - **I got a working tailored Linux kernel finally!**
    - But, I found out that the kernel is still unstable ☹
    - The boot up is the only thing it can
  - The kernel tailoring framework failed to get the working tailored kernel sometimes



Advanced Features

THE LINUX FOUNDATION

Advanced Features

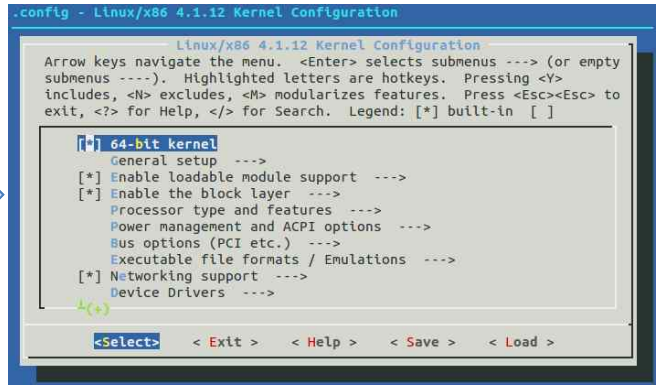
- Fine-grained kernel tailoring
  - Not grouping
    - Tailoring each kernel configuration option (one by one)
    - Relationship with conditions for the verification

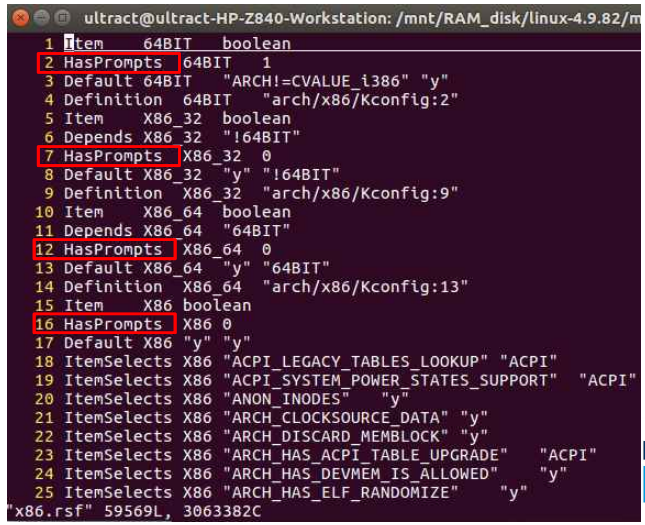
Candidates of Configuration Options
(# of Candidates: 650 For Gooroom)

Advanced Features

- Fine-grained kernel tailoring
  - Only selectable configuration options
    - Using a model file by the *undertaker-kconfigdump*
      - “HasPrompts”

Showing Selectable Configuration Options





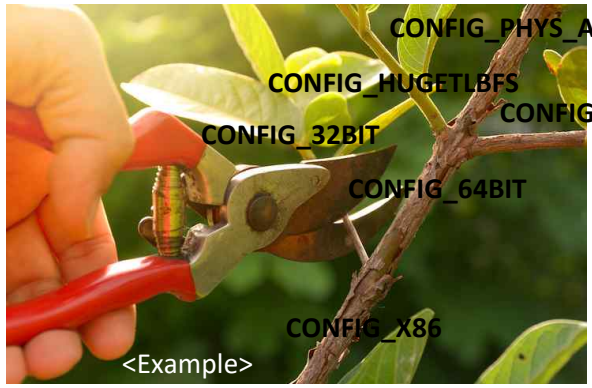
“x86.rsf” File by undertaker-kconfigdump

Advanced Features

- Fine-grained kernel tailoring
  - Dependency between configuration options
    - Counting how other configuration options “Depend on” a particular configuration option
    - Checking the configuration options from lowest to highest

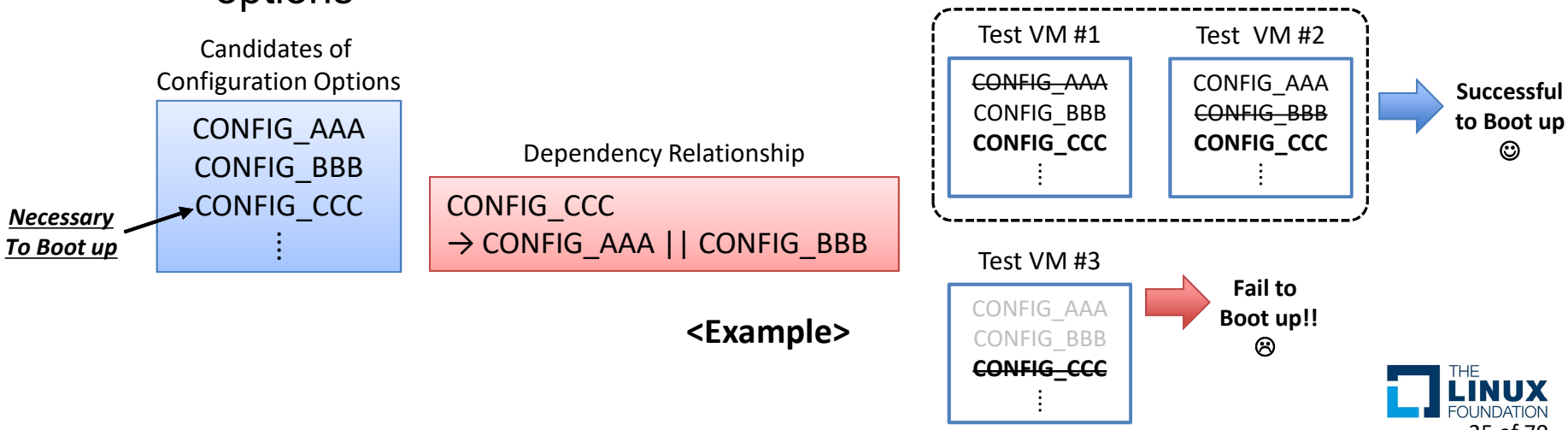
of the counted dependency

```
5       STACKTRACE
6       ACPI_APEI
6       AGP
6       AMD_IOMMU
6       DMI
6       MEMORY_HOTPLUG
6       PARPORT
6       PARTITION_ADVANCED
6       PCIEPORTBUS
6       SECURITY_TOMOYO
7       INTEL_IOMMU
7       PCI_MSI
8       AUDIT
8       IOMMU_SUPPORT
8       KALLSYMS
8       THERMAL
9       CPU_FREQ
```



Advanced Features

- Fine-grained kernel tailoring
  - Randomizing configuration options
    - Minimizes the dependency between candidates of configuration options



Advanced Features

- Various conditions considered for
  - Display
    - Resolution and Dimension
  - Network
  - Other peripherals
    - Keyboard and Mouse
  - Security
    - Kernel protection mechanisms
  - File systems
  - Etc
    - Power state
    - System logs (Journalctl)
    - Running applications

Advanced Features

- Various conditions considered for Display
  - Resolution & dimension

    Phoronix-test-suite system-info → Compare the before and after
    - Xdpyinfo or xrandr
      → Compare the before and after

```
ultract2@ultract: ~/phoronix-test-suite$ ./phoronix-test-suite system-info

Phoronix Test Suite v7.8.0
System Information

  PROCESSOR:
    Core Count:          Intel Xeon E5-2697 v3 @ 2.59GHz
    Extensions:          4
    Cache Size:          SSE 4.2 + AVX2 + AVX + RDRAND + FSGSBASE
    Microcode:           35840 KB
                         0x3c

  GRAPHICS:              LLVMpipe
    OpenGL:              3.3 Mesa 13.0.6 Gallium 0.4 (LLVM 3.9 256 bits)
    Display Driver:      modesetting 1.19.2
    Screen:              1440x900

  MOTHERBOARD:           Intel 440BX
    BIOS Version:        6.00

  MEMORY:                2048MB

  DISK:                  21GB VMware Virtual S
    File-System:         ext4
    Mount Options:       data=ordered errors=remount-ro relatime rw
    Disk Scheduler:      CFQ

  OPERATING SYSTEM:      Gooroom 1.0
    Kernel:              4.9.82 (x86_64)
    Desktop:             Xfce 4.12
    Compiler:            GCC 6.3.0 20170516
    System Layer:        VMware
    Security:            KPTI + __user pointer sanitization + Full generic retpoline Protection
```

Advanced Features

- Various conditions considered for Network
  - IPv4
    - /bin/ip a | grep "192.168."
  - IPv6
    - /bin/ip a | grep "inet6 [a-z0-9]\+::[a-z0-9:]\+"
    - dmesg and journalctl | grep "Failed to insert module 'ipv6'"
  - Ping the gateway

Advanced Features

- Various conditions considered for Peripherals
  - Keyboard & mouse device
    - /dev/input & udevadm(udev management tool) info
      - ID_INPUT_KEYBOARD, ID_INPUT_MOUSE
    - lsmod | grep 'psmouse‘

Advanced Features

- Various conditions considered for Security
  - Kernel protection mechanisms
    - Checksec → Compare the before and after
      - Checks kernel protection mechanisms.
        E.g. Restrict /dev/mem, ASLR, GCC stack protector support…
        (https://github.com/slimm609/checksec.sh)
    - Phoronix-test-suite info → Compare the before and after

Advanced Features

- Various conditions considered for File systems
  - Mount → Compare the before and after
    - Filters pluggable (Dynamic) file systems
      E.g. grep -v "binfmt_misc\|iso9660\|fusectl"

※ Verifiable by Other Conditions or Use-cases

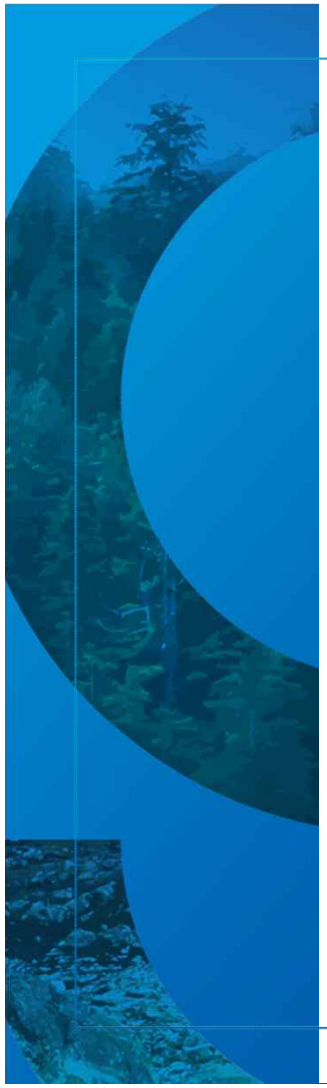
Advanced Features

- Various conditions consider for …
  - Etc
    - Power state (Suspend & hibernation)
      - grep "suspend" | /sys/power/disk
      - grep "disk" | /sys/power/state

        ※ https://www.kernel.org/doc/Documentation/power/
    - Journalctl → Compare the before and after
    - Phoronix-test-suite info → Compare the before and after
    - Running applications

Advanced Features

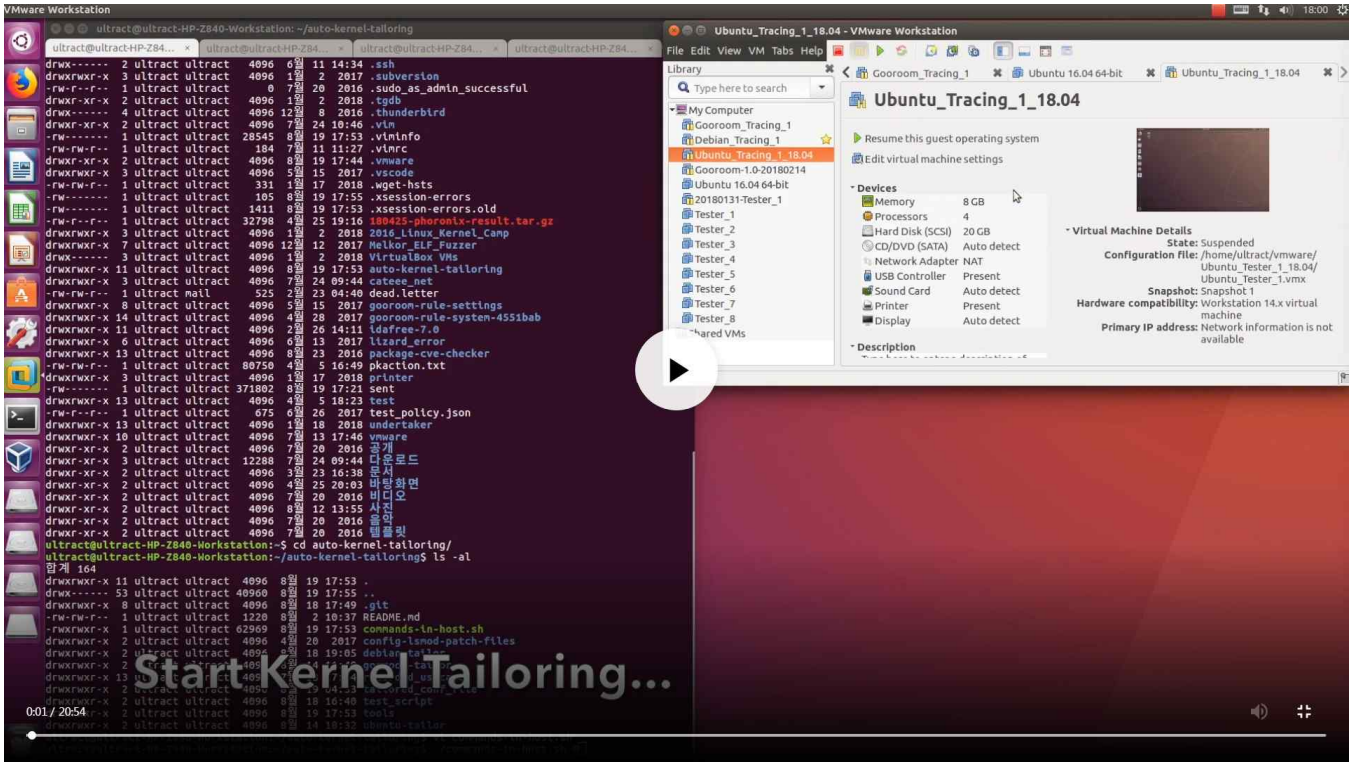
- Supports for other Linux distributions
  - Gooroom (Our custom desktop Linux ☺)
    - Beta 1.0 64bit, Kernel Ver 4.9
    - Xfce Desktop Environment, Lightdm
  - Debian
    - Stretch(9.4) 64bit Desktop, Kernel Ver 4.9
    - Gnome Desktop Environment, Lightdm
  - Ubuntu
    - Bionic Beaver(18.04) 64bit Desktop, Kernel Ver 4.15
    - Gnome Desktop Environment, Lightdm



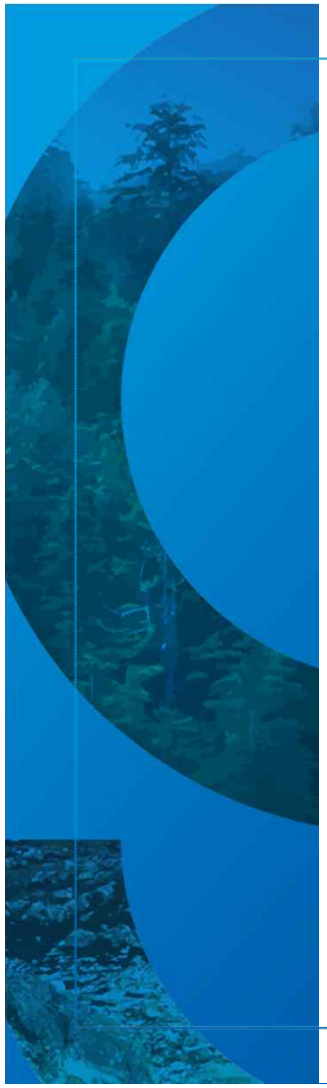
Demo

THE LINUX FOUNDATION

Demo



※ This Video: https://youtu.be/fHceA4asiXU
Previous Work : https://youtu.be/fnnCn-Bxjnw



Evaluation

THE LINUX FOUNDATION

Evaluation

- Total Elapsed Time, ※ Tested more than 5 times, Deviation : ± 1 hour
  - Gooroom Beta 1.0
    - About 7 hours
      - # of Verification VMs: 8
      - # of Candidates of Configuration Options: about 650
  - Debian 9.4
    - About 9 hours
      - # of Verification VMs: 8
      - # of Candidates of Configuration Options: about 630
  - Ubuntu 18.04
    - About 15 hours
      - # of Verification VMs: 8
      - # of Candidates of Configuration Options: about 1000

Evaluation

- The size of the kernel family
  - Gooroom beta 1.0
    - Kernel image size
      - Tailored : 14,399,796 bytes (≈ 72%)
      - Original : 20,090,752 bytes, ※ Decompressed by extract-vmlinux
    - Initial ramdisk size
      - Tailored : 6,672,465 bytes (≈ 20%)
      - Original : 34,078,719 bytes
    - The size of kernel modules
      - Tailored : 6,650,050 bytes (≈ 0.04%), # of .ko : 91 (≈ 0.03%)
      - Original : 186,697,093 bytes , # of .ko : 3,387

Evaluation

- The size of the kernel family
  - Debian 9.4
    - Kernel image size
      - Tailored : 12,289,612 bytes (≈ 61%)
      - Original : 20,161,244 bytes, ※ Decompressed by extract-vmlinux
    - Initial ramdisk size
      - Tailored : 5,910,123 bytes (≈ 30%)
      - Original : 19,582,713 bytes
    - The size of kernel modules
      - Tailored : 5,026,255 bytes (≈ 0.03%), # of .ko : 91 (≈ 0.03%)
      - Original : 189,458,941 bytes , # of .ko : 3,387

Evaluation

- The size of the kernel family
  - Ubuntu 18.04
    - Kernel image size
      - Tailored : 20,951,272 bytes (≈ 22%)
      - Original : 94,147,992 bytes, ※ Decompressed by extract-vmlinux
    - Initial ramdisk size
      - Tailored : 12,377,995 bytes (≈ 22%)
      - Original : 53,935,618 bytes
    - The size of kernel module
      - Tailored : 5,772,651 bytes (≈ 0.02%), # of .ko : 64 (≈ 0.01%)
      - Original : 236,401,113 bytes , # of .ko : 5,161

Evaluation

- Kernel configuration file
  - Gooroom beta 1.0

	Original .config	1st Tailored .config by Undertaker-Tailor	Localmodconfig .config	Final Tailored .config	
Enable (=y)	1785	359	1194	565	1785 → 565 (≈ 32%)
Module (=m)	3189	75	101	90	3189 → 90 (≈ 3%)
Disable (not set)	1601	1377	2329	1608	
Etc (String, Number)	139	47	83	65	
Total (Enable + Module + Etc)	5113	481	1378	720	5113 → 720 (≈ 14%)

Evaluation

- Kernel configuration file
  - Gooroom beta 1.0

```
BCH_CONST_M     "drivers/mtd/devices/Kconfig:218"
BCH_CONST_M     "lib/Kconfig:316"
PCI_MMCONFIG    "arch/x86/Kconfig:2480"
PCI_MMCONFIG    "arch/x86/Kconfig:2497"
```

?

Sub-Directory of Linux Kernel	Original .config	1st Tailored .config by Undertaker-Tailor	Localmodconfig .config	Final Tailored .config	
arch	271	149	256	189	
block	32	8	32	12	
crypto	130	35	54	47	
drivers	3109	85	473	140	3019 → 140 (≈ 5%)
fs	261	22	58	44	
init	126	48	125	85	
kernel	93	47	89	57	
lib	127	40	99	62	
mm	52	18	47	26	
net	640	16	73	29	640 → 29 (≈ 5%)
security	52	8	52	19	
sound	214	14	25	19	214 → 19 (≈ 9%)
usr	7	0	7	2	
virt	14	1	1	1	
Total	5128	491	1391	732	

Evaluation

- Kernel configuration file
  - Debian 9.4

	Original .config	1st Tailored .config by Undertaker-Tailor	Localmodconfig .config	Final Tailored .config	
Enable (=y)	1761	364	1170	565	1761 → 565 (≈ 32%)
Module (=m)	3202	75	103	94	3202 → 94 (≈ 3%)
Disable (not set)	1602	1391	2335	1605	
Etc (String, Number)	139	47	83	65	
Total (Enable + Module + Etc)	5102	486	1356	724	5102 → 724 (≈ 14%)

Evaluation

- Kernel configuration file
  - Debian 9.4

Sub-Directory of Linux Kernel	Original .config	1st Tailored .config by Undertaker-Tailor	Localmodconfig .config	Final Tailored .config	
arch	273	149	258	190	
block	32	8	32	12	
crypto	127	35	49	47	
drivers	3111	92	474	147	3111 → 147 (≈ 5%)
fs	261	21	55	44	
init	126	48	124	84	
kernel	93	47	89	55	
lib	126	40	94	63	
mm	52	18	47	26	
net	639	16	72	28	639 → 28 (≈ 4%)
security	42	8	42	17	
sound	214	14	25	19	214 → 19 (≈ 9%)
usr	7	0	7	3	
virt	14	1	1	1	
Total	5117	497	1369	736	

Evaluation

- Kernel configuration file
  - Ubuntu 18.04

	Original .config	1st Tailored .config by Undertaker-Tailor	Localmodconfig .config	Final Tailored .config	
Enable (=y)	2381	338	1596	634	2381 → 634 (≈ 23%)
Module (=m)	4937	45	74	55	4937 → 55 (≈ 1%)
Disable (not set)	749	1423	2630	1620	
Etc (String, Number)	173	45	105	69	7491 → 758 (≈ 10%)
Total (Enable + Module + Etc)	7491	428	1775	758	

Evaluation

- Kernel configuration file
  - Ubuntu 18.04

Sub-Directory of Linux Kernel	Original .config	1st Tailored .config by Undertaker-Tailor	Localmodconfig .config	Final Tailored .config	
arch	315	157	304	211	
block	45	8	40	15	
cert	10	0	10	2	
crypto	151	4	65	53	
drivers	5085	74	681	133	5085 → 133 (≈ 3%)
fs	280	7	75	41	
init	141	39	140	83	
kernel	118	55	111	66	
lib	156	38	116	66	
mm	67	21	64	28	
net	679	18	84	33	679 → 33 (≈ 5%)
security	67	7	65	20	
sound	377	12	30	19	377 → 19 (≈ 5%)
ubuntu	1	0	0	0	
usr	7	0	7	2	
virt	14	1	1	1	
Total	7513	441	1793	773	

New Directories

Evaluation

- Verification log - Gooroom beta 1.0

※ https://github.com/ultract/linux-kernel-tailoring-framework/tree/master/tailoring_log

[Boot Up]

BINFMT_SCRIPT
DEVTMPFS
EPOLL
FILE_LOCKING
FUTEX
INOTIFY_USER
MULTIUSER
RD_GZIP
SERIAL_8250
SHMEM
SIGNALFD
SYSFS
TIMERFD
TMPFS
TTY
UNIX
UNIX98_PTYS
VT

[Phoronix-test-suite]

DMI → Motherboard & BIOS Information
DMIID → Motherboard & BIOS Information
DRM_LEGACY → Graphics
IOSCHED_CFQ → Disk Scheduler - CFQ(Before), NOOP(After)
PACKET → No Internet Connectivity
PAGE_TABLE_ISOLATION → Security - KPTI
RETPOLINE → Security - Full generic retpoline Protection

[Journalctl Log]

ECRYPT_FS → Failed to find module 'ecryptfs'
IPV6 → device (enp2s1): addrconf6: failed to start neighbor discovery ...
NAMESPACES → Failed to start Hostname Service ...
PACKET → (Socket Filtering) are enabled in your kernel ...
PARPORT → Failed to find module 'lp', 'parport_pc', 'ppdev'
PRINTER → Failed to find module 'lp'
RETPOLINE → Spectre V2 : kernel not compiled with retpoline;
TMPFS_POSIX_ACL → Failed to apply ACL on /dev/dri/card0: Operation not supported ...

Evaluation

- Verification log - Gooroom beta 1.0

※ https://github.com/ultract/linux-kernel-tailoring-framework/tree/master/tailoring_log

[Checksec]

AUDIT → SELinux Enable
CC_STACKPROTECTOR_STRONG → GCC stack protector support
RANDOMIZE_BASE → Address space layout randomization
RELOCATABLE → Address space layout randomization
SECURITY → SELinux Enable
SECURITY_SELINUX → SELinux Enable
STRICT_DEVMEM → Restrict /dev/mem access

[File Systems]

DEFAULT_SECURITY_SMACK → smackfs
NAMESPACES → hugetlbfs
SECURITY → smackfs
SECURITY_SMACK → smackfs

[Peripherals]

INPUT_KEYBOARD
INPUT_MOUSE
KEYBOARD_ATKBD
MOUSE_PS2

[Network]

IPV6 → IPv6 Address Not Set
NAMESPACES → IPv4 Address Not Set
PACKET → IPv4 Address Not Set, Ping to Gateway Failed

[Power State]

HIBERNATION → /sys/power/disk, /sys/power/state
SUSPEND → /sys/power/disk
SWAP → /sys/power/disk, /sys/power/state

[Kernel Module]

MODULE_UNLOAD → Kernel Module Loading Failed

[Applications]

ADVISE_SYSCALLS → Browser Not Working - Fatal Error
NAMESPACES → Pulse Audio Not Working

Evaluation

- Verification log - Debian 9.4

※ https://github.com/ultract/linux-kernel-tailoring-framework/tree/master/tailoring_log

[Boot Up]

BINFMT_SCRIPT
DEVTMPFS
EPOLL
EXT4_USE_FOR_EXT2
FILE_LOCKING
FUTEX
INOTIFY_USER
MULTIUSER
RD_GZIP
SHMEM
SIGNALFD
SYSFS
TIMERFD
TMPFS
TTY
UNIX
UNIX98_PTYS
VT

[Phoronix-test-suite]

DMI → Motherboard & BIOS Information
DMIID → Motherboard & BIOS Information
IOSCHED_CFQ → Disk Scheduler - CFQ(Before), NOOP(After)
NET_VENDOR_REALTEK → No Internet Connectivity
PACKET → No Internet Connectivity
PAGE_TABLE_ISOLATION → Security - KPTI
RD_LZ4 → No Internet Connectivity
RETPOLINE → Security - Full generic retpoline Protection

[Journalctl Log]

IPV6 → device (enp2s1): addrconf6: failed to start neighbor discovery ...
NAMESPACES → Failed to start Hostname Service ...
NET_VENDOR_REALTEK → setsockopt(udp, IP_ADD_MEMBERSHIP)(0.0.0.0): No such device
PACKET → (Socket Filtering) are enabled in your kernel ...
PARPORT → Failed to find module 'lp', 'parport_pc', 'ppdev'
PRINTER → Failed to find module 'lp'
RD_LZ4 → setsockopt(udp, IP_ADD_MEMBERSHIP)(0.0.0.0): No such device
RETPOLINE → Spectre V2 : kernel not compiled with retpoline; no mitigation available!
SERIAL_8250 → bad device "/dev/ttyS0" given
TMPFS_POSIX_ACL → Failed to apply ACL on /dev/dri/card0: Operation not supported ...
VT_CONSOLE → /dev/ttyS0: not a tty

Evaluation

- Verification log - Debian 9.4

※ https://github.com/ultract/linux-kernel-tailoring-framework/tree/master/tailoring_log

[Checksec]

AUDIT → SELinux Enable
CC_STACKPROTECTOR_STRONG → GCC stack protector support
RANDOMIZE_BASE → Address space layout randomization
RELOCATABLE → Address space layout randomization
SECURITY → SELinux Enable
SECURITY_SELINUX → SELinux Enable
SLAB_FREELIST_RANDOM SLAB freelist randomization
STRICT_DEVMEM → Restrict /dev/mem access
VMAP_STACK Virtually-mapped kernel stack

[File Systems]

NAMESPACES → hugetlbfs

[Peripherals]

INPUT_KEYBOARD
INPUT_MOUSE
KEYBOARD_ATKBD
MOUSE_PS2

[Network]

IPV6 → IPv6 Address Not Set
NAMESPACES → IPv4 Address Not Set
PACKET → IPv4 Address Not Set, Ping to Gateway Failed

[Power State]

HIBERNATION → /sys/power/disk, /sys/power/state
SWAP → /sys/power/disk, /sys/power/state

[Kernel Module]

MODULE_UNLOAD → Kernel Module Loading Failed

[Applications]

NAMESPACES → Pulse Audio Not Working

Evaluation

- Verification log - Ubuntu 18.04 ※ https://github.com/ultract/linux-kernel-tailoring-framework/tree/master/tailoring_log

[Boot Up]

BINFMT_SCRIPT
DEVTMPFS
EPOLL
EXT4_FS
FUTEX
INOTIFY_USER
MULTIUSER
RD_GZIP
SERIAL_8250
SERIAL_8250_CONSOLE
SHMEM
SIGNALFD
TIMERFD
TMPFS
UNIX
UNIX98_PTYS
VT

[Phoronix-test-suite]

DMI → Motherboard & BIOS Information
DMIID → Motherboard & BIOS Information
IOSCHED_CFQ → Disk Scheduler - CFQ(Before), NOOP(After)
PACKETT → No Internet Connectivity
PAGE_TABLE_ISOLATION → Security - KPTI
RETPOLINEE → Security - Full generic retpoline Protection
VIRTIO_BALLOON → No Internet Connectivity

[Journalctl Log]

FILE_LOCKING → [autospawn] core-util.c: lock: Permission denied ...
FUSE_FS → Failed to find module 'fuse'
INPUT_EVDEV → cannot open input layer
IPV6 → device (enp2s1): addrconf6: failed to start neighbor discovery ...
OSF_PARTITION → Failed to mount Mount unit for core, revision 5145
PACKET → (Socket Filtering) are enabled in your kernel ...
PARPORT → Failed to find module 'lp', 'parport_pc', 'ppdev'
PARPORT_PC Failed to find module 'parport_pc'
POSIX_TIMERS Failed to call clock_adjtime(): Function not implemented
PRINTER → Failed to find module 'lp'
PRINTK → activation of module imklog failed
RETPOLINE → Spectre V2 : kernel not compiled with retpoline; no mitigation available!
SQUASHFS_XZ → squashfs: SQUASHFS error: Filesystem uses "xz" compression
TMPFS_POSIX_ACL → Failed to apply ACL on /dev/dri/card0: Operation not supported ...

Evaluation

- Verification log - Ubuntu 18.04

※ https://github.com/ultract/linux-kernel-tailoring-framework/tree/master/tailoring_log

[Checksec]

VMAP_STACK → Virtually-mapped kernel stack
HARDENED_USERCOPY → Hardened Usercopy
SLAB_FREELIST_RANDOM → SLAB freelist randomization
CC_STACKPROTECTOR_STRONG → GCC stack protector support
RANDOMIZE_BASE → Address space layout randomization
AUDIT → SELinux Enable
SECURITY_SELINUX → SELinux Enable
SECURITY → SELinux Enable

[File Systems]

SQUASHFS → squashfs
SQUASHFS_XZ → squashfs
CONFIGFS_FS → configfs
FUSE_FS → fuse.gvfsd-fuse
MISC_FILESYSTEMS → pstore

[Peripherals]

INPUT_KEYBOARD
INPUT_MOUSE
KEYBOARD_ATKBD
MOUSE_PS2

[Network]

PACKET → IPv4 Address Not Set, Ping to Gateway Failed
IPV6 → IPv6 Address Not Set

[Power State]

HIBERNATION → /sys/power/disk, /sys/power/state
SUSPEND → /sys/power/disk
SWAP → /sys/power/disk, /sys/power/state

[Kernel Module]

MODULE_UNLOAD → Kernel Module Loading Failed

[Applications]

FILE_LOCKING → Pulse Audio Not Working

Evaluation

- Boot up time - Gooroom beta 1.0
  - Tailored kernel image

※ By system-analyze

    - Startup finished in **1.577s** (kernel) + 2.930s (userspace) = **4.507s**
    - Startup finished in **1.410s** (kernel) + 2.928s (userspace) = **4.338s**
    - Startup finished in **1.523s** (kernel) + 3.241s (userspace) = **4.764s**
  - Original kernel image
    - Startup finished in **2.695s** (kernel) + 3.324s (userspace) = **6.020s**
    - Startup finished in **2.839s** (kernel) + 3.502s (userspace) = **6.341s**
    - Startup finished in **2.836s** (kernel) + 3.082s (userspace) = **5.918s**

Evaluation

- Boot up time - Debian 9.4
  - Tailored kernel image

※ By system-analyze

    - Startup finished in **1.416s** (kernel) + 6.751s (userspace) = **8.167s**
    - Startup finished in **1.450s** (kernel) + 6.649s (userspace) = **8.100s**
    - Startup finished in **1.442s** (kernel) + 6.598s (userspace) = **8.041s**
  - Original kernel image
    - Startup finished in **1.845s** (kernel) + 7.243s (userspace) = **9.089s**
    - Startup finished in **1.800s** (kernel) + 7.228s (userspace) = **9.029s**
    - Startup finished in **2.053s** (kernel) + 6.992s (userspace) = **9.046s**

Evaluation

- Boot up time - Ubuntu 18.04
  - Tailored kernel image

※ By system-analyze

    - Startup finished in **1.724s** (kernel) + 5.912s (userspace) = **7.636s**
    - Startup finished in **1.662s** (kernel) + 4.319s (userspace) = **5.982s**
    - Startup finished in **1.737s** (kernel) + 5.660s (userspace) = **7.397s**
  - Original kernel image
    - Startup finished in **3.931s** (kernel) + 5.752s (userspace) = **9.683s**
    - Startup finished in **3.980s** (kernel) + 4.162s (userspace) = **8.143s**
    - Startup finished in **3.894s** (kernel) + 3.793s (userspace) = **7.688s**

Evaluation

- Performance – Lmbench on Gooroom
  - Most of the test results are similar, except some test items below

Processor, Processes - times in microseconds - smaller is better				
	fork proc	exec proc	sh proc	
Tailored	353.29	1321.86	2677.57	
Original	393.29	1454.14	2919.43	
※ *Variation*	-40.00	-132.29	-241.86	

Context switching - times in microseconds - smaller is better			
	8p/16K ctxsw	16p/64K ctxsw	
Tailored	43.49	53.14	
Original	54.66	60.79	
※ *Variation*	-11.17	-7.64	

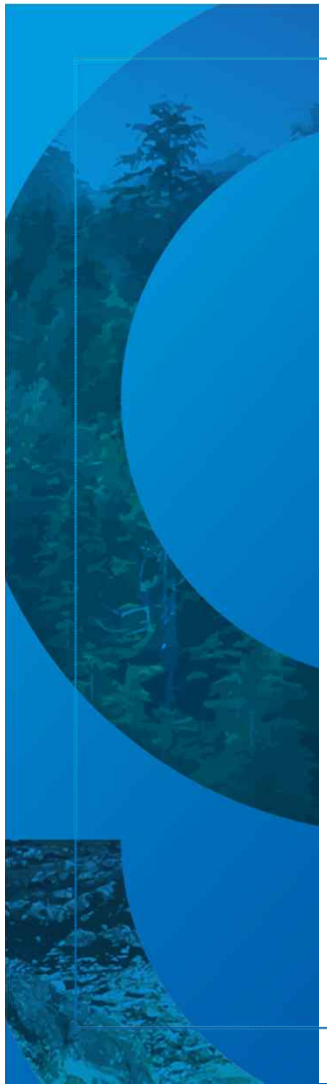
Local Communication bandwidths in MB/s - bigger is better							
	TCP	File reread	Mmap reread	Bcopy(libc)	Bcopy(hand)	Mem read	Mem write
Tailored	2301.14	3944.31	5484.96	4640.73	2479.63	5567.86	3444.14
Original	2196.57	3427.71	5348.34	4141.60	1784.83	5054.29	2547.57
※ *Variation*	104.57	516.60	136.61	499.13	694.80	513.57	896.57

※ https://github.com/ultract/linux-kernel-tailoring-framework/tree/master/performance_test

Evaluation

- Performance - Phoronix-test-suite on Gooroom
  - The original results

    ※ https://github.com/ultract/linux-kernel-tailoring-framework/tree/master/performance_test



Discussion

THE LINUX FOUNDATION

Discussion

- The fine-grained kernel tailoring
  - Considers the dependency & randomizes the configuration options
    - Reducing a failure rate of the kernel tailoring empirically
  - With the various conditions for a verification
    - The tailored kernel is more stable than before
    - We can make a whitelist for the kernel tailoring based on the verification log

Discussion

- The fine-grained kernel tailoring
  - Reduces the candidates of the configuration options by the selectable options (“HasPrompts”)
  - Takes longer than the previous tailoring framework
    - More than 2 hours in case of Gooroom

Discussion

- The performance of the tailored kernel
  - A little better than the original kernel
    - To understand the reason, I need to look into the results more…

Discussion

- The performance of the tailored kernel
  - It is impossible to trace the configuration options related to the performance by the undertaker-tailor and the tailoring framework
  - The configuration options need to be added by hand
    - I refer to some Linux performance and tuning guidelines
    - I added it as a whitelist for the performance

Discussion

- The conditions for a verification
  - Making the conditions is a difficult work.
    - Too many H/W Spec, drivers, modules, applications, etc
    - By trial and error…
    - By comparing the before and after…
  - It need to be formalized and organized later
  - *The more conditions are added, the more configuration options are gathered, and then the tailored kernel will be heavier*

Discussion

- Desktop manager issues for the verification
  - Xfce and Lightdm are better than Gnome and Gdm
    - The virtual machine using the gnome is slow to be revert and play
    - Gdm service doesn't work to restart properly for the use-cases and the verification during the kernel tailoring
  - xfce4-terminal and gnome-terminal
    - They have different options to execute commands for the use-cases and the verification scripts

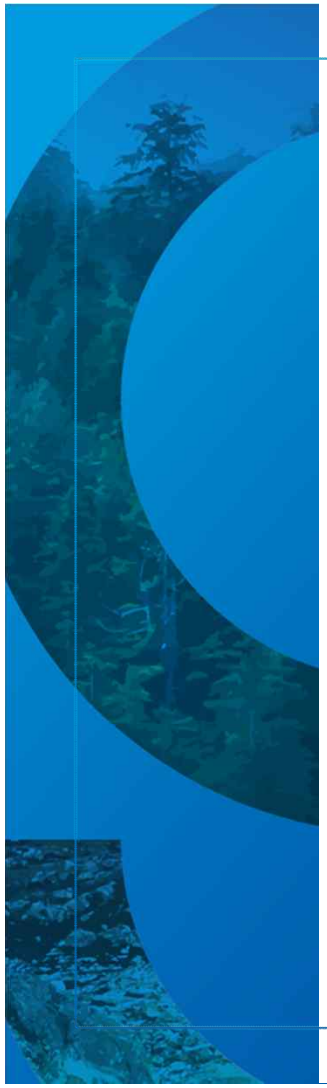
Discussion

- The error for making Kconfig model files on the Ubuntu
  - The undertaker-kconfigdump can’t handle “imply” attribute of the Kconfig
    - “imply”(weak select) → “select”

      ※ https://www.kernel.org/doc/Documentation/kbuild/kconfig-language.txt

Discussion

- The limitation of the Localmodconfig
  - It can only include configuration options of inserted modules via the insmod command
  - The necessary kernel module should be loaded beforehand
- The kernel tailoring only works on virtual machines
  - I need another a new approach for a physical machine
    - How to implement the kernel tailoring framework for a physical machine?
    - The automation of tracing kernel features and the verification tailored kernels like on the virtual machines?

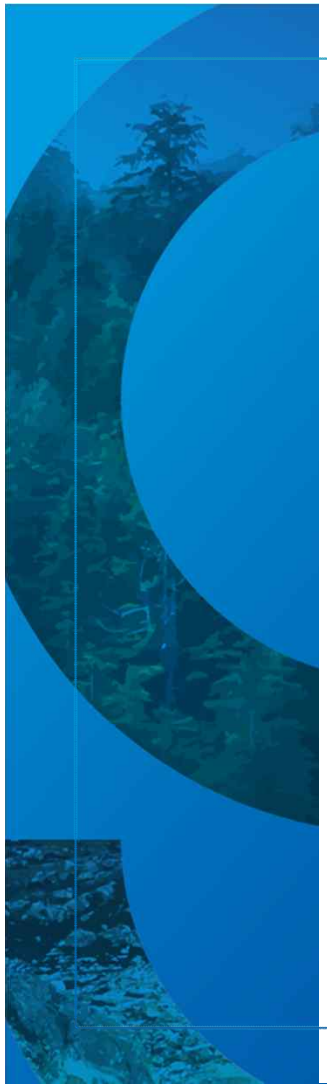


Conclusion

THE LINUX FOUNDATION

Conclusion

- We looked into several approaches for the kernel tailoring
  - Undertaker-tailor
  - Localmodconfig
  - Kernel tailoring framework
- Advanced features for the kernel tailoring framework
  - Fine-grained kernel tailoring
    - Enhanced Stability of a Tailored Kernel
    - Relation between Configuration Options & Various Verification Conditions
  - Supported for other Linux distributions
    - Debian, Ubuntu
- A little performance benefit
- Future work
  - Formalizing or organizing the conditions for a verification
  - Kernel tailoring toward a physical machine ☺



Questions?

(https://github.com/ultract/linux-kernel-tailoring-framework)

THE LINUX FOUNDATION


THE LINUX FOUNDATION
OPEN SOURCE SUMMIT
THE LINUX FOUNDATION