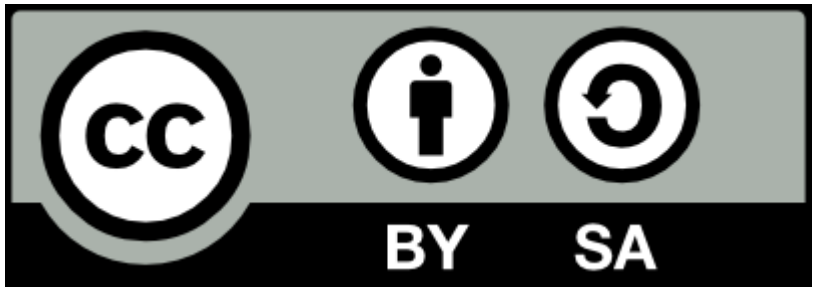
Android Security, From the Ground Up

ELCE 2014

Karim Yaghmour
@karimyaghmour



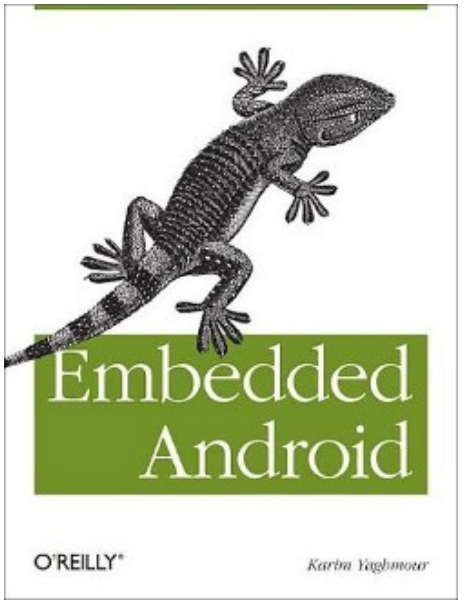
These slides created by: Karim Yaghmour

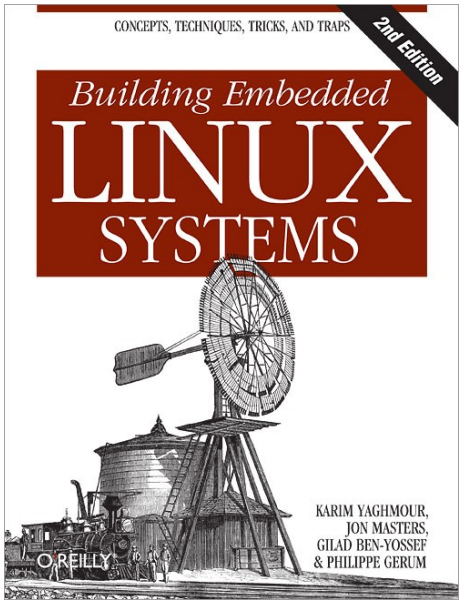
Originals at: www.opersys.com/community/docs

Delivered and/or customized by

About

- Author of:





- Introduced Linux Trace Toolkit in 1999
- Originated Adeos and relayfs (kernel/relay.c)
- Training, Custom Dev, Consulting, ...

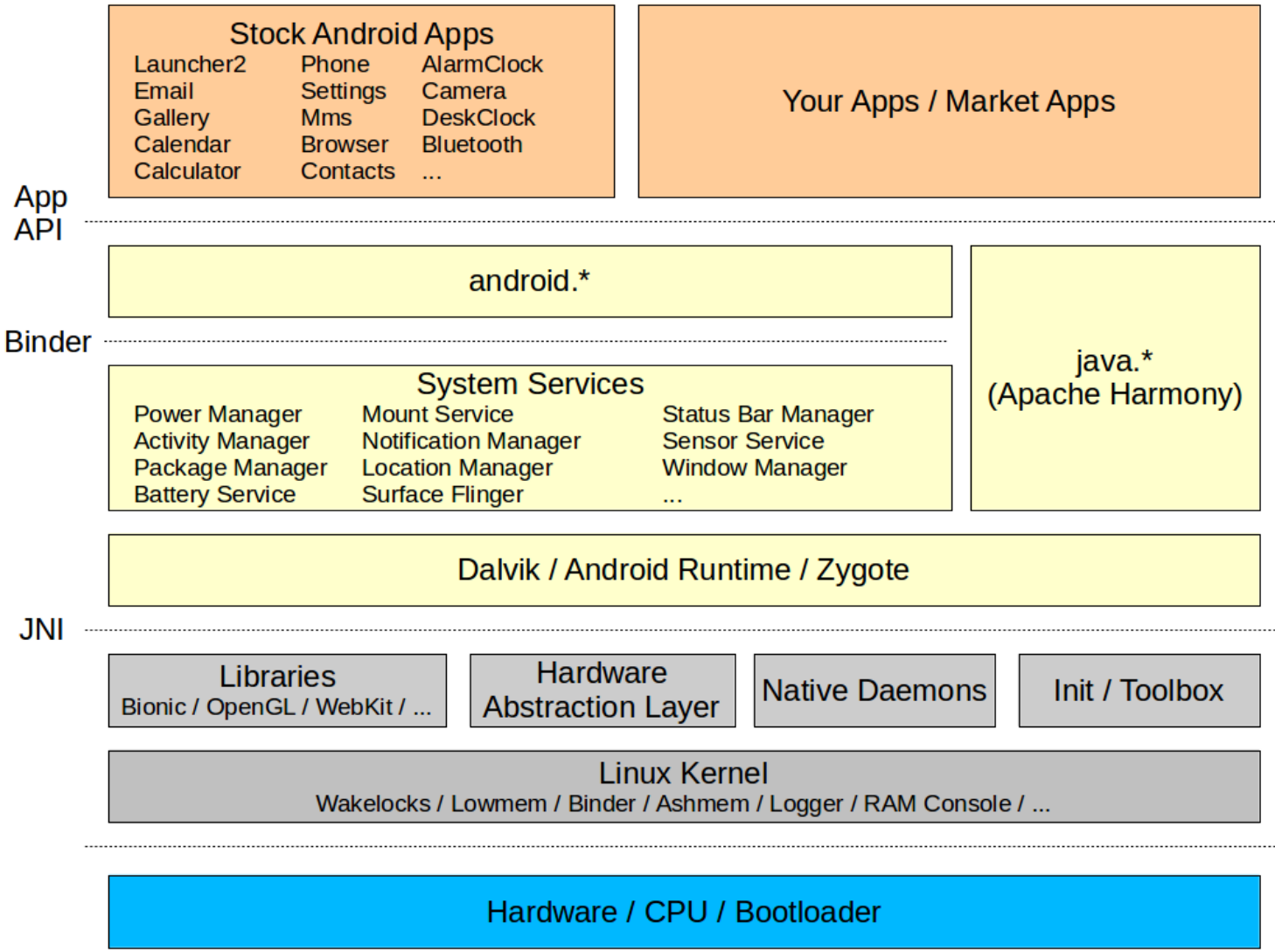
Android Security, From the Ground Up

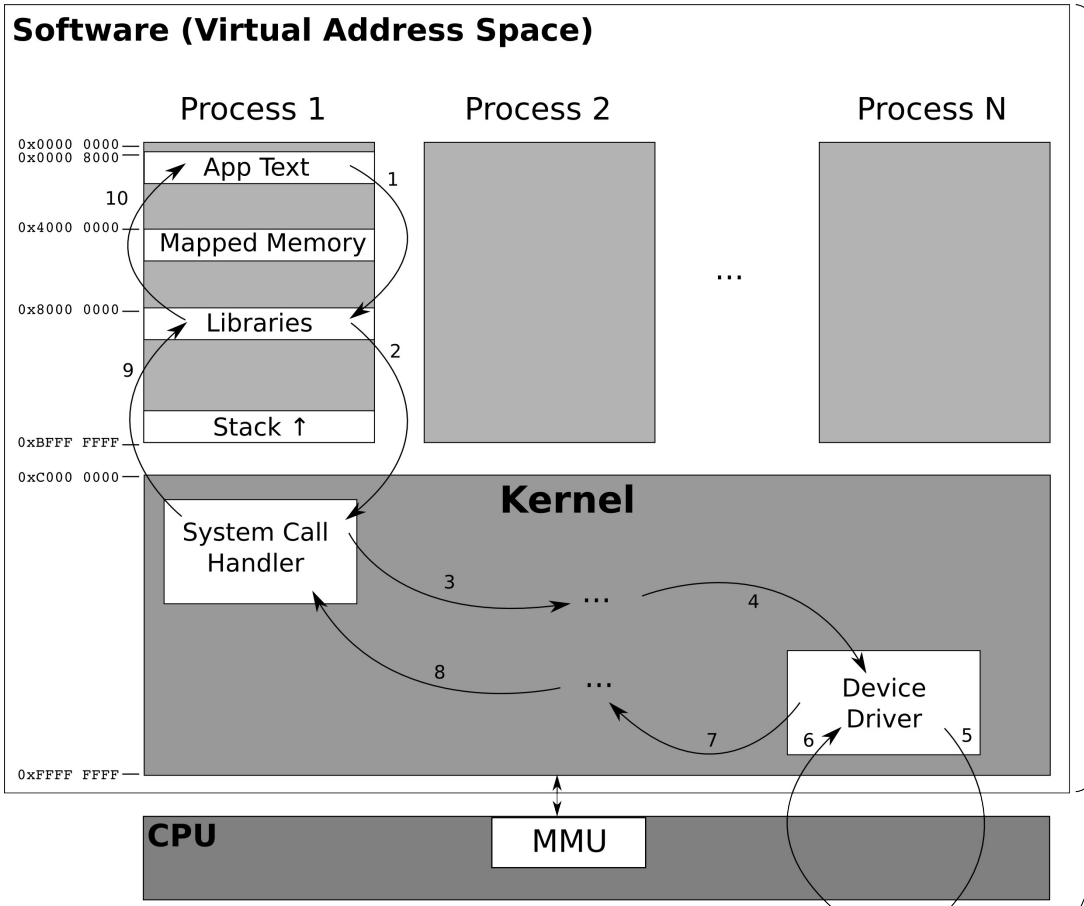
1. Goals and Features
2. Layers involved
3. CPU
4. Bootloader
5. Kernel
6. Native user-space
7. Framework
8. Updates
9. AppOpps
10. Bottom line

1. Goals and Features

- Goals:
  - Protect user data
  - Protect system resources (including the network)
  - Provide application isolation
- Key Features:
  - Robust security at the OS level through the Linux kernel
  - Mandatory application sandbox for all applications
  - Secure interprocess communication
  - Application signing
  - Application-defined and user-granted permissions

2. Layers involved





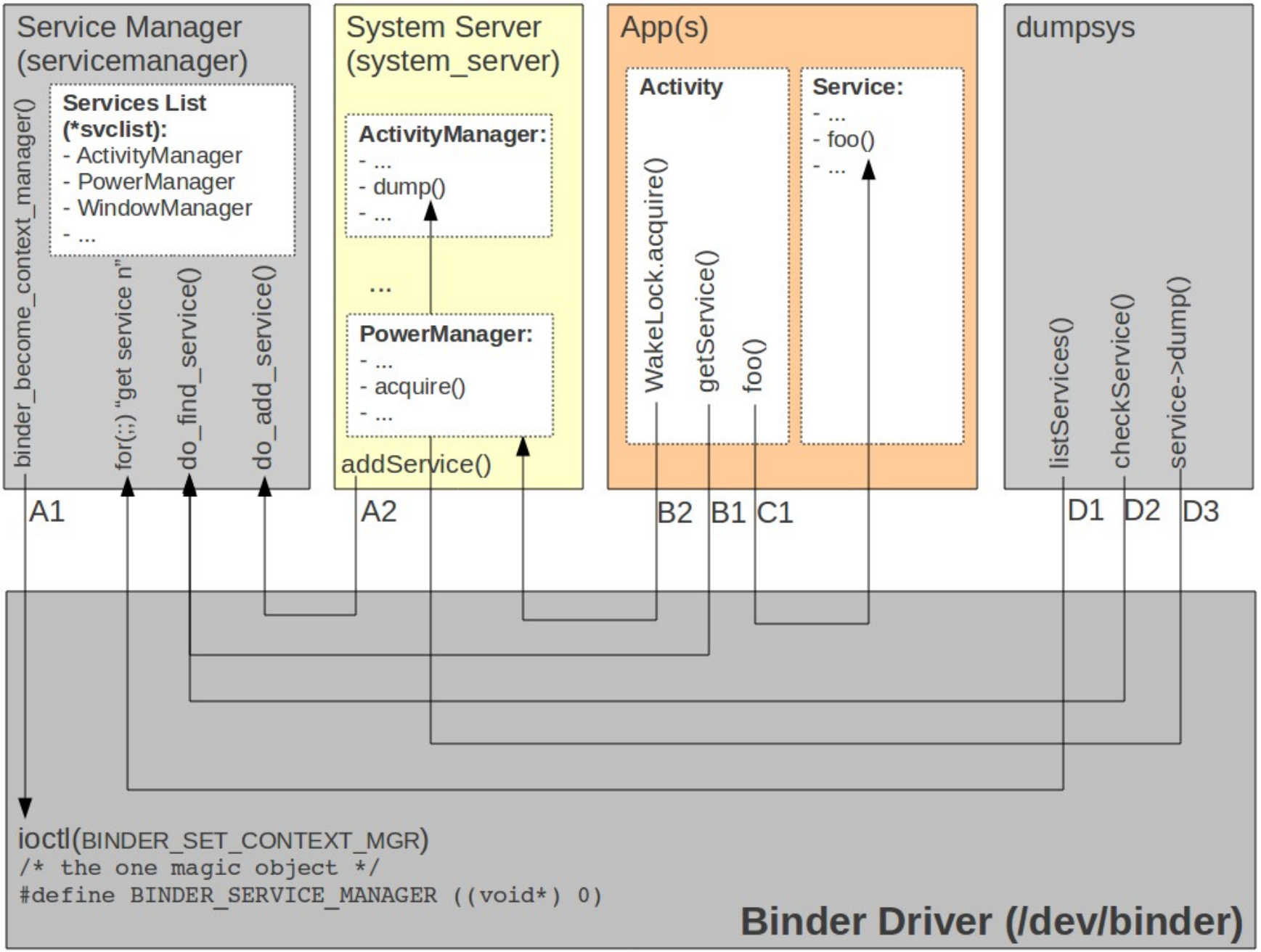
Hardware (Physical Address Space)

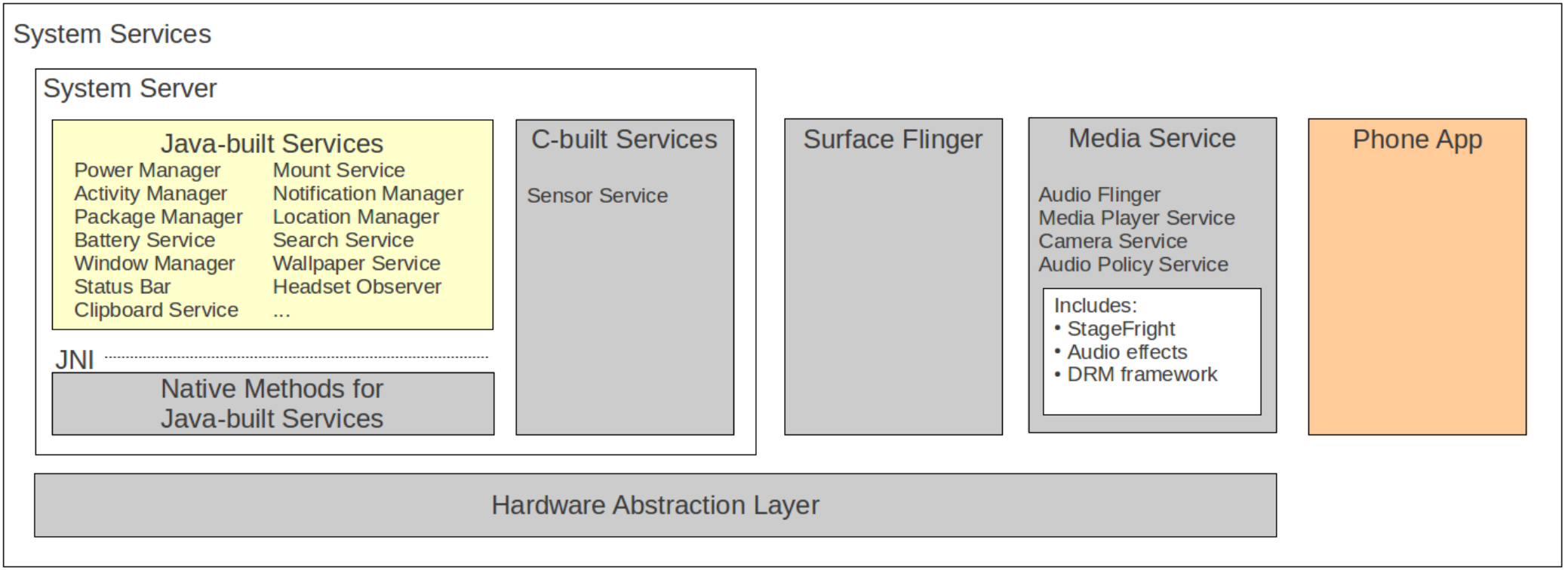
Goldfish (emulator)

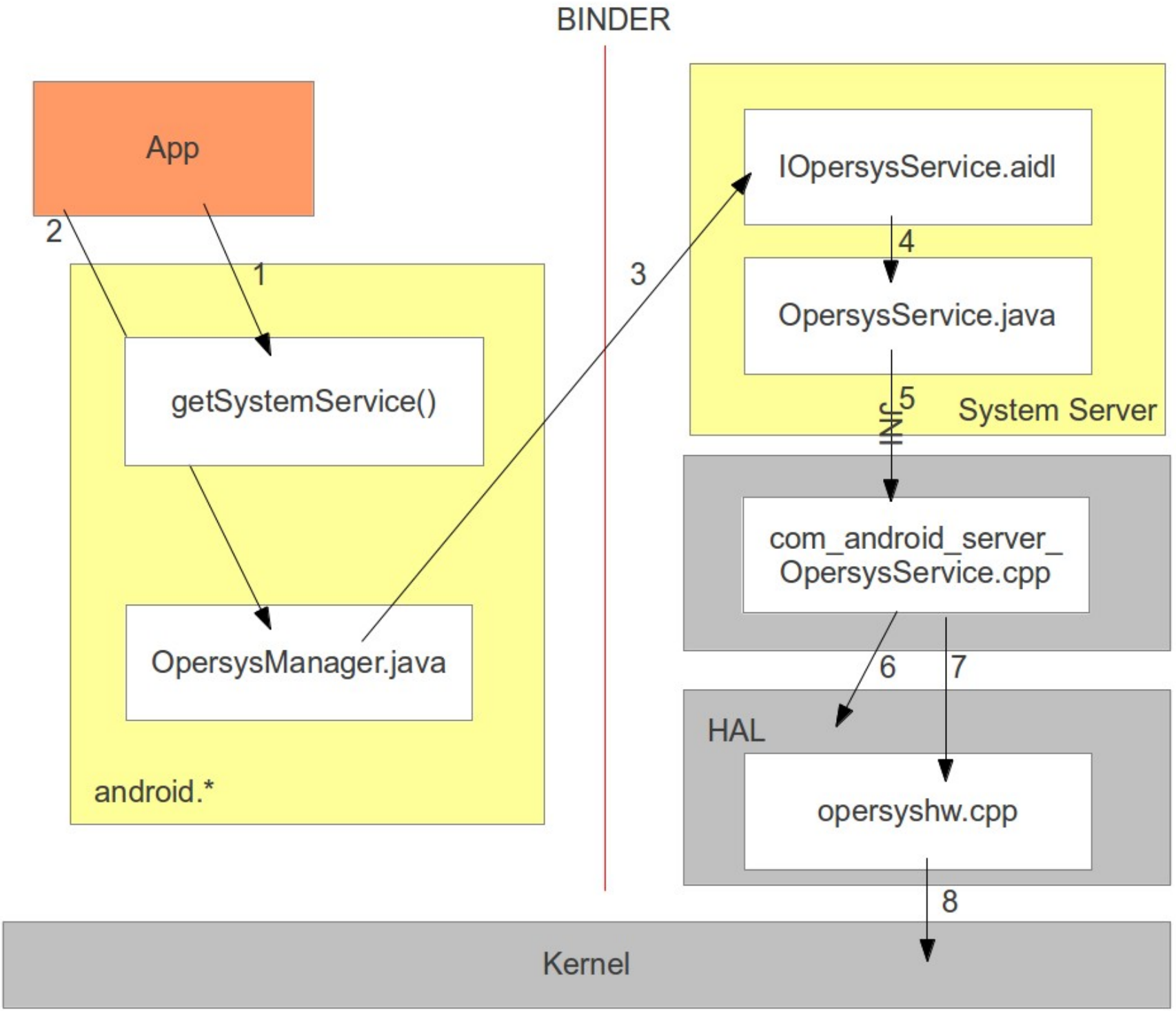
```
00000000-05ffffff : System RAM
ff000000-ff000fff : goldfish_interrupt_
ff001000-ff001fff : goldfish_device_bus
ff002000-ff002fff : goldfish_tty.0
ff003000-ff003fff : goldfish_timer
ff004000-ff004fff : goldfish_audio.0
ff006000-ff006fff : goldfish_memlog.0
ff010000-ff010fff : goldfish_rtc
ff011000-ff011fff : goldfish_tty.1
ff012000-ff012fff : smc91x.0
ff013000-ff013fff : goldfish_fb.0
ff014000-ff014fff : goldfish-battery.0
ff015000-ff015fff : goldfish_events.0
ff016000-ff016fff : goldfish_nand.0
ff017000-ff017fff : goldfish-switch.0
ff018000-ff018fff : goldfish-switch.1
```

BeagleBone Rev. A3

```
00000000-00000000 : omap_hsmmc.0
44e05000-44e053ff : omap_hsmmc.0
44e07000-44e07fff : omap_gpio.3
44e09000-44e0afff : omap_hsmmc.0
44e0b000-44e0bfff : omap_i2c.1
44e31000-44e313ff : omap_timer.1
44e35000-44e35fff : omap_wdt
44e3e000-44e3efff : omap_rtc
47400000-47400fff : usbss
47401000-474017ff : musb0
47401800-47401fff : musb1
47810100-478200ff : omap_hsmmc.0
48022000-48023fff : omap_hsmmc.0
48024000-48025fff : omap_hsmmc.0
4802a000-4802afff : omap_i2c.1
48030100-480304ff : omap2_mcspi.1
48040000-480403ff : omap_timer.2
...
4a101200-4a1012ff : cpsw.0
80000000-8fffffff : System RAM
```


Service Manager
(servicemanager)
Services List
(*svclist):
- ActivityManager
- PowerManager
- WindowManager
- ...
binder_become_context_manager()
for(;;) "get service n"
do_find_service()
do_add_service()
System Server
(system_server)
ActivityManager:
- ...
- dump()
- ...
...
PowerManager:
- ...
- acquire()
- ...
addService()
App(s)
Activity
WakeLock.acquire()
getService()
foo()
Service:
- ...
- foo()
- ...
dumpsys
listServices()
checkService()
service->dump()
A1
A2
B2
B1
C1
D1
D2
D3
ioctl(BINDER_SET_CONTEXT_MGR)
/* the one magic object */
#define BINDER_SERVICE_MANAGER ((void*) 0)
Binder Driver (/dev/binder)


System Services
System Server
Java-built Services
Power Manager
Activity Manager
Package Manager
Battery Service
Window Manager
Status Bar
Clipboard Service
Mount Service
Notification Manager
Location Manager
Search Service
Wallpaper Service
Headset Observer
...
JNI
Native Methods for Java-built Services
C-built Services
Sensor Service
Surface Flinger
Media Service
Audio Flinger
Media Player Service
Camera Service
Audio Policy Service
Includes:
• StageFright
• Audio effects
• DRM framework
Phone App
Hardware Abstraction Layer


BINDER
App
2
1
getSystemService()
OpersysManager.java
android.*
3
IOpersysService.aidl
4
OpersysService.java
5
System Server
com_android_server_
OpersysService.cpp
6
7
HAL
opersyshw.cpp
8
Kernel

3. CPU

- Paging
- Memory segmentation
- Privileged instructions
- Security:
  - Secure boot
  - Crypto acceleration (ARM v8)
  - TrustZone

4. Bootloader

- Locked vs. unlocked
- Signed vs. unsigned images
- Very bootloader dependent

5. Kernel

- Process isolation
- UID/GID
- Capabilities
- SELinux
- Misc. additions and features

5.1. Process isolation

- Each process gets its own address space
- Processes can't see each others' memory
- Processes can't access the kernel's memory
- MUST use system calls to talk to kernel

5.2. UID/GID

- Each process has a UID / GID
- Privileges granted to processes sharing UIDs and GIDs
  - Filesystem access
  - Signals
  - Tracing
  - etc.

5.3. Capabilities

- Root has a lot of power
- Sometimes only part of root privileged needed
- Use “man capabilities” to find out more
- Used by installd to drop out of root and keep privileges.

5.4. SELinux

- Linux has Discretionary Access Controls (DAC) by default.
- SELinux adds Mandatory Access Controls (MAC).
- Requires all process operations to be explicitly mapped out.
- Unlisted operations are forbidden, even if you're root.
- Provides safe firewalling in case of privilege escalation.

5.5. Misc. additions and features

- Paranoid networking
- CONFIG_STRICT_MEMORY_RWX
- ...

6. Native user-space

- Filesystem partitions
- init.rc permission settings
- /dev/*
- /dev/socket/*
- Native daemons
- installd

6.1. Filesystem partitions

- Each partition has different mount options
  - RAM disk => Read-Only
  - System image => Read-Only (unless update)
  - Data image => Read-Write (specific user permissions needed)
  - Cache => Read-Write
  - Recovery => Not mounted by default
  - Virtual filesystems (proc, sysfs, etc.)
  - “sdcard” => Read-Write (world readable/writable)
- Directories and files have specific rights, see system/core/include/private/android_filesystem_config.h

6.2. init.rc permission settings

- mount
- mkdir
- chown
- chmod

6.3. /dev/*

- All devices accessed through device nodes
- Devices nodes have regular file permissions
- Entries created by ueventd
- App do NOT have access to most entries

6.4. /dev/socket/*

- Unix domain sockets used native daemons and services:

```
srw-rw---- system   system            2014-07-24 18:45 adbd
srw-rw---- root     inet              2014-07-24 18:45 dnsproxyd
srw------- system   system            2014-07-24 18:45 installd
srw-rw---- root     system            2014-07-24 18:45 mdns
srw-rw---- root     system            2014-07-24 18:45 netd
srw-rw-rw- root     root              2014-07-24 18:45 property_service
srw-rw-rw- root     root              2014-07-24 18:45 qemud
srw-rw---- root     radio             2014-07-24 18:45 rild
srw-rw---- radio    system            2014-07-24 18:45 rild-debug
srw-rw---- root     mount             2014-07-24 18:45 vold
srw-rw---- root     system            2014-07-24 18:45 zygote
```

6.5. Native daemons

- Some native daemons authenticate the requests they get:
  - sevicemanager
  - init property service
- Some daemons shadow key system services:
  - vold
  - netd
  - rild
  - keystore

6.6. installd

- Package Manager's “shadow”
- Starts as root
- Notifies kernel that it will drop out of root but wants to keep capabilities.
- Changes UID to “install” user
- Sets caps kept as:
  - DAC_OVERRIDE
  - CHOWN
  - SETUID
  - SETGID

7. Framework

- Framework permissions
- App signatures
- Multi-human support
- Device administration
- SEAndroid

7.1. Framework permissions

- Since apps can't access /dev/* entries, they must talk to system services through Binder.
- Binder doesn't enforce security
- System services check for permissions on a call-by-call basis.
- Package Manager is solicited to check permissions.
- Ex: checkCallingOrSelfPermission()

7.2. App signatures

- Apps must be signed by publisher
- Publishers are NOT authenticated
- There is NO certificate authority in this model
- Based on Java “keytool”

7.3. Multi-human support

- Each user gets a region of UIDs
- Each gets entries in:
  - /data/user
    - Per-app data directories
  - /data/system/users
    - Per-user accounts DB

7.4. Device administration

- API for BYOD
- Very limited
- Only good for password-strength enforcement
- Does not provide:
  - Provisioning of apps
  - Bulk configuration

7.5. SEAndroid

- Mandatory access controls for Android
- Enforcing/non-enforcing: setenforce
- Not merged:
  - Middle-ware MAC

8. Updates

- OTA certs:
  - platform: Phone, SystemUI, framework, etc.
  - shared: Launcher2, Contacts, LatinIME, etc.
  - media: Gallery, MediaProvider, etc.
  - testkey: default key
- Default keys in: build/target/product/security
- Use of development/tools/make_key to generate new keys.

9. AppOps

- AppOps system service
- Introduced and the removed
- frameworks/base/core/java/com/android/internal/app/IAppOppsService.aidl
- packages/apps/Settings/AndroidManifest.xml

10. Bottom line

- Strong built-in mechanisms
- but ...

- Dude, where's my “sudo apt-get update” / “sudo apt-get upgrade”?
- No updates = No security

References and Pointers

- http://source.android.com/devices/tech/security/index.html
- http://seandroid.bitbucket.org

Thank you ...

karim.yaghmour@opersys.com