brightsight®

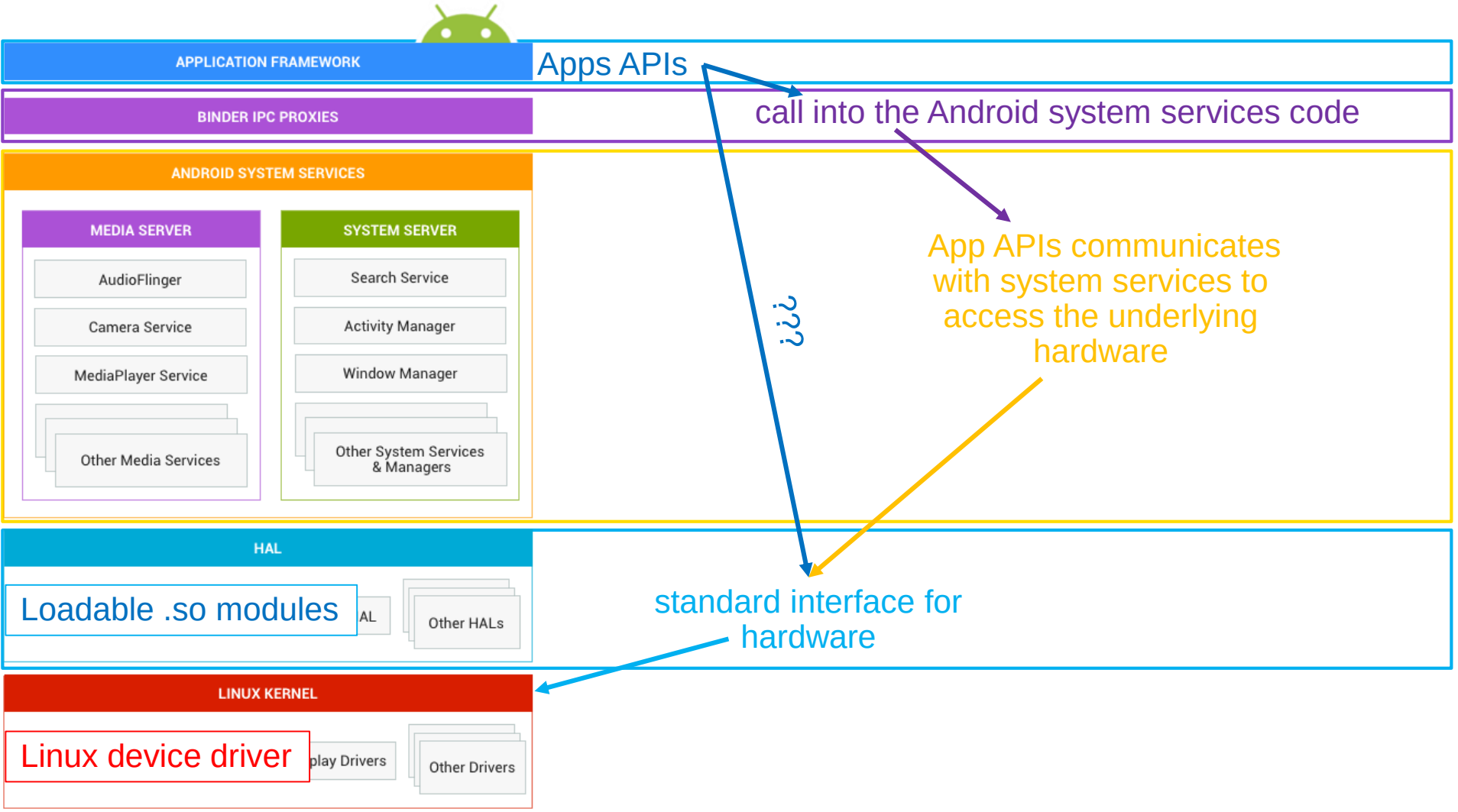
the number one
security lab
in the world

Android Security

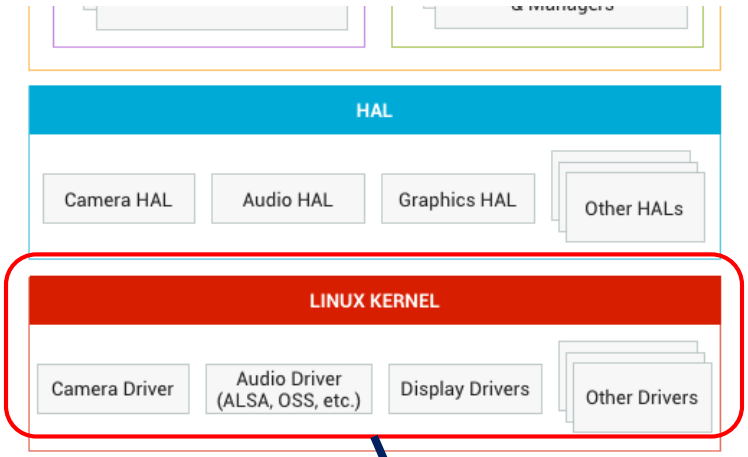
Security Peripheral Management (B2/B17)

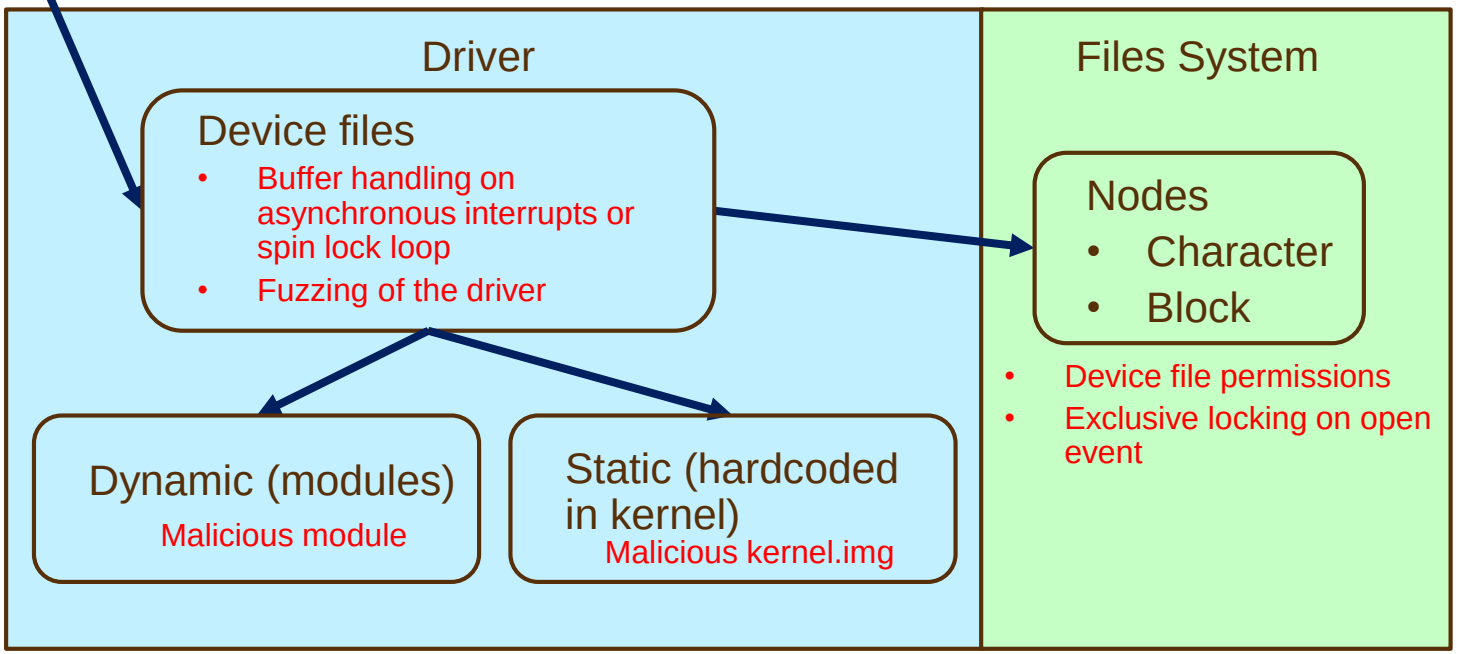
Shiqi Li / Brightsight

Device usage in Android



Security Concerns for Device Drivers





Evaluation Concerns

#	Description	Security risk	Evaluations bullets
1	Device driver security – loadable modules (.so)	Malicious module in Root File System (e.g in “boot.img” file)	Check whether modules signed and verified on: • Loading • Booting • Every 24hrs
2	Device driver security - static linked drivers	Malicious kernel image (e.g. “kernel.img”)	Check authentication procedure of the kernel-image by underlying boot layer - Each boot stage stage has to be authenticated by the underlaying boot-layer
3	Access privileges for device files	Wrong privileges bits may grant unnecessary access	Ask vendor to provide privilege settings list for device nodes (static & dynamic). Check that device node file privilege in Root File System is correct: • SUID 1 bit,Owner/Group/Other 9 bits
4	Access privileges by the android service for device node	Wrong privileges bits may grant unnecessary access	Check on whether an application access rights are verified within the android secure service • Access right should match access privilege of node devices

Evaluation Concerns (2)

#	Description	Security risk	Evaluations bullets
5	Exclusive device access	Block simultaneous access of more than 1 service/application to avoid memory leakage	1st layer locks Lock in the kernel/driver. Ask vendor to provide lock diagram for the kernel or the driver and do a sample check.
6	Direct apps APIs' mapping to the HAL layer	1. Multiple application access (see #6) 2. Asynchronous access	How exclusive access to the HAL library is managed • See #5
7	Exclusively handle application requests by android service for device	1. Block simultaneous access of more than 1 application to avoid memory leakage 2. For an transaction with multiple application requests, malicious commands could be inserted.	2nd layer locks 1. There is locks in the service 2. Check if there dependency between the a sequence of application requests. If yes, make sure this sequence cannot be interrupted/stopped by other application. 3. A lock design document of locks for the service (usually binder) has to be provided by vendor

Evaluation Concerns (3)

#	Description	Security risk	Evaluations bullets
8	Buffer handling	1. Incorrect handling of the buffer during I/O(HW interrupts or spin lock loop) may lead to a memory leak 2. Buffer overflow	• Check source code for buffer overflow, especially for API handling input data
9	Driver fuzzing	Like an ICCR	ICCR and USB already in PCI requirements • Like a pipe or processes data – define what you fuzz

Questions?