A Study on Android Emulator Detection Using Build Properties

Author : Jae-do Lim, Il-kyu Kim, Namsu Kim, BooJoong Kang, Seong-je Cho

Presentator : Il-kyu Kim

Affilation : Dankook Univ.

E-mail : ik.kim@dankook.ac.kr

INDEX

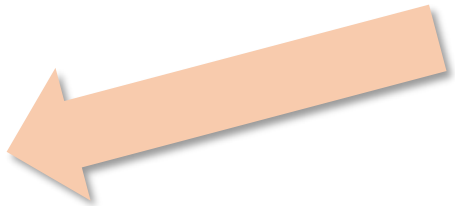
01 Introduction

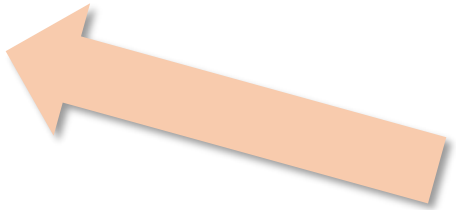
Introduction

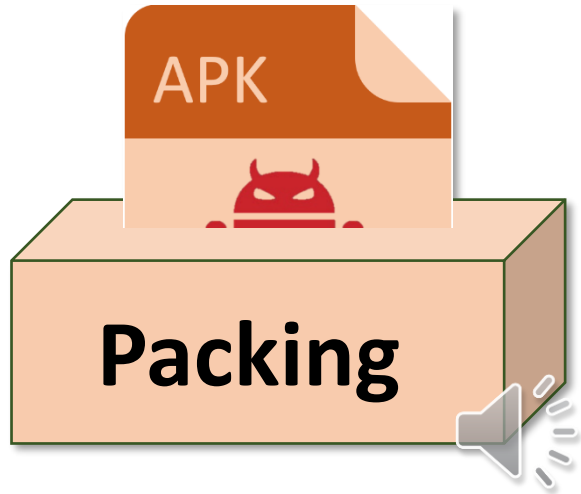
Static Analysis

Dynamic Analysis

Introduction







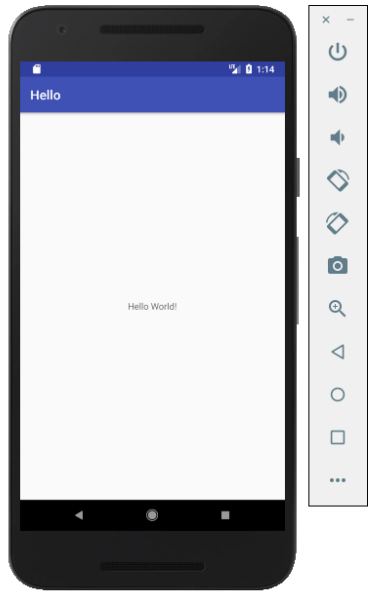
Introduction

Introduction

Real Device

MEmu

nox



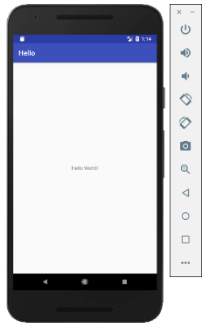
BlueStacks
Play Bigger

Emulator

Introduction

MEmu

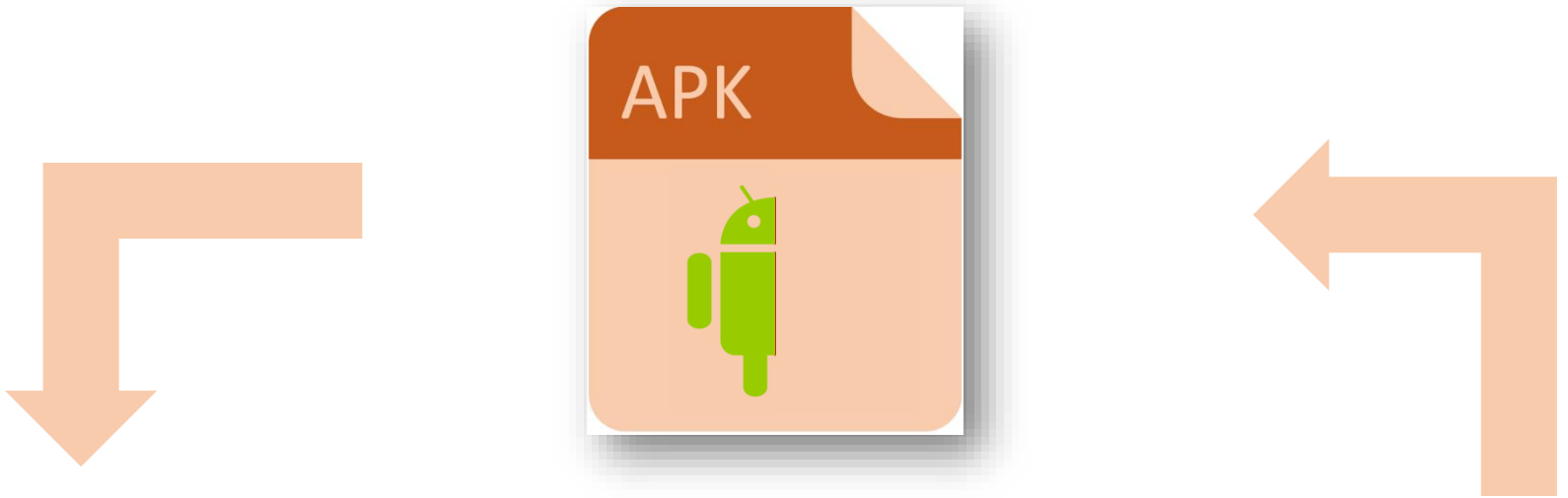
nox

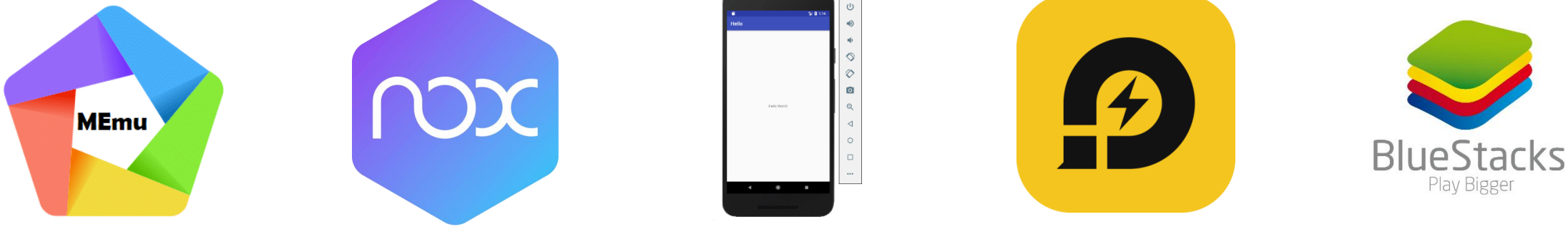


BlueStacks
Play Bigger

Emulator

Introduction





Emulator

Introduction

/system/build.prop

- It contains the build properties and settings.

android.os.Build class

- The class keeps information about the soft -ware build properties related to the SDK b uild process.

02 Related Work

Related Work

❖ [2], [3] proposed the techniques to detect emulator through system properties.

Morpheus: Automatically Generating Heuristics to Detect Android Emulators

Yiming Jing†, Ziming Zhao†, Gail-Joon Ahn†, and Hongxin Hu‡
†Arizona State University ‡Clemson University
{ymjing,zmzhao,gahn}@asu.edu, hongxih@clemson.edu

Survey of Dynamic Anti-Analysis Schemes for Mobile Malware

Jongsu Lim, Yonggu Shin, Sunjun Lee, Kyuho Kim, and Jeong Hyun Yi*
School of Software, Soongsil University, 06978, Republic of Korea
{jongsu253, tls09611, starj1024, krbgh205760}@gmail.com, jhyi@ssu.ac.kr

❖ [4] proposed the emulator detecting methods via field of android.os.Build class, however, they found that th ese field values easily modifying.

Evading Android Runtime Analysis via Sandbox Detection

Timothy Vidas
Carnegie Mellon University
tvidas@cmu.edu

Nicolas Christin
Carnegie Mellon University
nicolasc@andrew.cmu.edu

❖ [5] checked the way malicious app detect the emulator using Build.prop file, IMEI, Network, and Sensor.

A Robust Dynamic Analysis System Preventing SandBox Detection by Android Malware

Jyoti Gajrani*
MNIT, Jaipur, India
2014rcp9542@mnit.ac.in

Jitendra Sarswat†
MNIT, Jaipur, India
2011uit1582@mnit.ac.in

Meenakshi Tripathi‡
MNIT, Jaipur, India
mtripathi.cse@mnit.ac.in

Vijay Laxmi§
MNIT, Jaipur, India
vlaxmi@mnit.ac.in

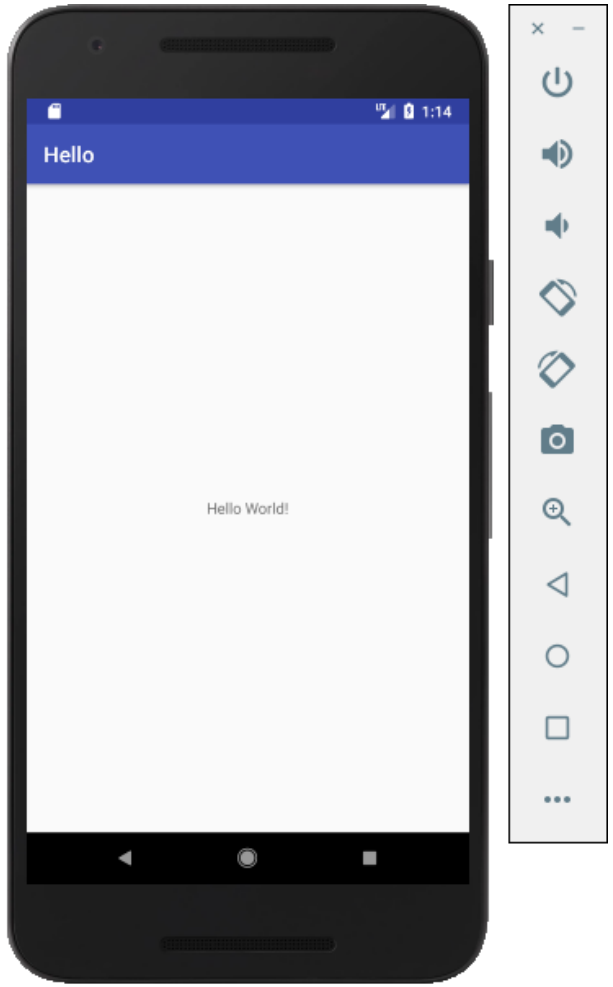
M.S. Gaur¶
MNIT, Jaipur, India
gaurms@mnit.ac.in

Mauro Conti
University of Padua, Italy
conti@math.unipd.it

03 Emulator Detection Using the Build Properties

A Emulators and Experimental Environments

Emulators and Experimental Environments



❖The Android Emulator simulates Android devices on a user's computer to test apps on various devices virtually and Android API Levels.

❖Each instance of the Android Emulator uses an Android Virtual Device(AVD) to designate the Android version a nd hardware properties of the simulated device.

❖To test an Android app, it needs to create an AVD that models each device on where the app is supposed to run.

Emulators and Experimental Environments

❖NoxPlayer, an Android emulator developed by MoreTech Inc. in China, emphasizes high performance and ultimate compatibility to play mobile game on PC.

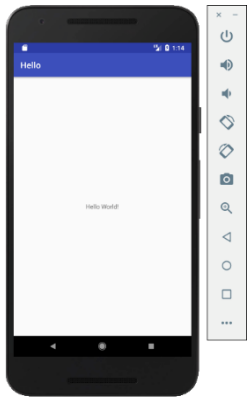
❖The BlueStacks App Player, one of the earliest Android emulators developed by BlueStacks Systems, Inc. in USA, is one of the most popular and most extensively used emulators.

Emulators and Experimental Environments

	AVD	Nox	AVD	BlueStacks
Devices	Google Pixel 2		Google Pixel 2 XL	
SDK	Version 25		Version 25	
Android OS	7.1.1	7.1.2	7.1.1	7.1.2

OS : Microsoft Windows 11 pro

Processor : Intel® CORE™ i7-8565U



B Analysis Methods

Analysis Methods

/system/build.prop

- It contains the build properties and settings.

android.os.Build class

- The class keeps information about the software build properties related to the SDK build process.

system/build.prop

```
# begin build properties
# autogenerated by buildinfo.sh
ro.build.id=NYC
ro.build.display.id=sdk_google_phone_arm64-userdebug 7.1.1 NYC 8695018 test-keys
ro.build.version.incremental=8695018
ro.build.version.sdk=25
ro.build.version.preview_sdk=0
ro.build.version.codename=REL
ro.build.version.all_codenames=REL
ro.build.version.release=7.1.1
ro.build.version.security_patch=2018-01-01
ro.build.version.base_os=
ro.build.date=Wed Jun  8 02:25:04 UTC 2022
ro.build.date.utc=1654655104
ro.build.type=userdebug
ro.build.user=android-build
ro.build.host=abfarm400
ro.build.tags=test-keys
ro.build.flavor=sdk_google_phone_arm64-userdebug
ro.product.model=Android SDK built for arm64
ro.product.brand=google
ro.product.name=sdk_google_phone_arm64
ro.product.device=generic_arm64
ro.product.board=
# ro.product.cpu.abi and ro.product.cpu.abi2 are obsolete,
# use ro.product.cpu.abilist instead.
ro.product.cpu.abi=arm64-v8a
ro.product.cpu.abilist=arm64-v8a
ro.product.cpu.abilist32=
ro.product.cpu.abilist64=arm64-v8a
ro.product.manufacturer=Google
ro.product.locale=en-US
ro.wifi.channels=
ro.board.platform=
# ro.build.product is obsolete; use ro.product.device
ro.build.product=generic_arm64
# Do not try to parse description, fingerprint, or thumbprint
ro.build.description=sdk_google_phone_arm64-userdebug 7.1.1 NYC 8695018 test-keys
ro.build.fingerprint=google/sdk_google_phone_arm64/generic_arm64:7.1.1/NYC/8695018:userdebug/test-keys
ro.build.characteristics=emulator
# end build properties
#
# from build/target/board/generic_arm64/system.prop
#
#
# system.prop for generic arm64 sdk
#

rild.libpath=/system/lib/libreference-ril.so
rild.libargs=-d /dev/ttyS0

#
# ADDITIONAL_BUILD_PROPERTIES
#
ro.config.notification_sound=OnTheHunt.ogg
ro.config.alarm_alert=Alarm_Classic.ogg
ro.ril.hsxpa=1
ro.ril.gprsclass=10
ro.adb.qemud=1
dalvik.vm.heapstartsize=5m
dalvik.vm.heapgrowthlimit=48m
dalvik.vm.heapsize=256m
dalvik.vm.heaptargetutilization=0.75
dalvik.vm.heapminfree=512k
dalvik.vm.heapmaxfree=2m
persist.sys.dalvik.vm.lib.2=libart.so
dalvik.vm.isa.arm64.variant=generic
dalvik.vm.isa.arm64.features=default
dalvik.vm.lockprof.threshold=500
xmpp.auto-presence=true
ro.config.nocheckin=yes
net.bt.name=Android
dalvik.vm.stack-trace-file=/data/anr/traces.txt
```

android.os.Build

```
Class<?> buildClass = Class.forName("android.os.Build");
Filed[] fileds = buildClass.getDeclaredFields();
```

- ❖ Above code snippet used to get the android.os.Build class.
- ❖ The android.os.Build class keeps information about the software build properties related to the SDK build process.
- ❖ It contains followed fields.

BOARD	BRAND	CPU_ABI	DEVICE	DISPLAY
HOST	ID	MANUFACTURER	MODEL	PRODUCT
TYPE	USER	HARDWARE	IS_EMULATOR	SERIAL

C Comparison of an AVD and NoxPlayer

Comparison of an AVD and NoxPlayer

	Property of /system/ build.prop	AVD	NoxPlayer	Field of android.os.Build	AVD	NoxPlayer
1	ro.product.board	-	msm8998	BOARD	*Unknown*	Walleye
2	ro.product.brand	*google*	samsung	BRAND	*google*	Google
3	ro.product.cpu.abi	*x86*	x86	CPU_ABI	*x86*	x86
4	ro.product.device	***generic**_x86*	dream2qltechn	DEVICE	***generic**_x86*	x86
5	ro.build.display.id	*NYC*	N2G48H.G9550 ZHU1AQEE	DISPLAY	*NYC*	google Pixel 2-user 7.1.2 LMY47I 700210909 release-keys
6	ro.build.host	*wprg10.hot.corp.google.com*	SWHD7308	HOST	*wprg10.hot.corp.google.com*	**ubuntu**
7	ro.build.id	*NYC*	N2G48H	ID	*NYC*	LMY47I
8	ro.product. manufacturer	*Google*	samsung	MANUFACTURER	*Google*	Google
9	ro.product.model	*Android **SDK** built for x 86*	SM-G9550	MODEL	*Android **SDK** built for x86*	google Pixel 2
10	ro.build.product	***generic**_x86*	dream2qltechn	PRODUCT	***sdk**_google_phone_x86*	google Pixel 2
11	ro.build.type	*user*	user	TYPE	*user*	user
12	ro.build.user	*android-build*	dpi	USER	*android-build*	user
13	ro.build. characteristics	***emulator***	phone			
14				HARDWARE	***ranchu***	android_x86
15				IS_EMULATOR	***true***	false
16				SERIAL	***EMULATOR**31X3X10X0*	91481a824d469ceb

D

Comparison of an AVD and Bluestacks 5

Comparison of an AVD and Bluestacks5

	Property of /system/ build.prop	AVD	BlueStacks 5	Field of android.os.Build	AVD	BlueStacks5
1	ro.product.board	-	-	BOARD	*unknown*	taimen
2	ro.product.brand	*google*	**BlueStacks**	BRAND	*google*	google
3	ro.product.cpu.abi	*x86*	x86	CPU_ABI	*x86*	x86
4	ro.product.device	***generic**_x86*	**BlueStacks**	DEVICE	***generic**_x86*	taimen
5	ro.build.display.id	***sdk**_google_phone_x86-userdebug 7.1.1 NYC 6695155 **test-keys***	N2G47H.7.8.23	DISPLAY	***sdk**_google_phone_x86-userdebug 7.1.1 NYC 6695155 **test-keys***	NOF26V
6	ro.build.host	***abfarm626***	Build2	HOST	*abfarm626*	Build2
7	ro.build.id	*NYC*	N2G47H	ID	*NYC*	NOF26V
8	ro.product. manufacturer	*Google*	samsung	MANUFACTURER	*Google*	Google
9	ro.product.model	*Android **SDK** built for x86*	**BlueStacks**	MODEL	*Android **SDK** built for x86*	PIXEL 2 XL
10	ro.build.product	***generic**_x86*	**AppPlayer**	PRODUCT	*sdk_google_phone_x86*	taimen
11	ro.build.type	***userdebug***	user	TYPE	*userdebug*	user
12	ro.build.user	*android-build*	build	USER	*android-build*	build
13	ro.build. characteristics	***emulator***	phone			
14				HARDWARE	***ranchu***	taimen
15				IS_EMULATOR	***true***	false
16				SERIAL	***EMULATOR**31X3X10X0*	4c5cb6d2cdd4

04 Conclusion and Future Work

Conclusion and Future Work

❖Conclusion

- We proposed the emulator detecting method using /system/build.prop file and android.os.Build class.
- Via our method, we could check specific key-value that discriminate the specific emulator.
  - In the case of the AVD, “generic”, “SDK”, “emulator”, “test-key”, “abfarm”, “userdebug”, “ranchu” can be used for detection.
  - “BlueStacks”, “AppPlayer” are special keys for BlueStacks.
- However, unlike AVD and BlueStacks, the NoxPlayer only has a special value : HOST of android.os.Build.
  - This value could be changed and the shown value “ubuntu” on our table, also appeared when we setting the device Galaxy S10, Galaxy S10 5G and it can be modified.

Conclusion and Future Work

❖Limitation

- In the real device, only root can read /system/build.prop file, so the android applications that only have normal user permission are hard to read this file.
- Some emulator like NoxPlayer, they have the value as same as real device, so it is hard to detect them using our methods.

❖Future Work

- Through further study, we checked another emulator's specific values to show their characteristics.
- We are going to verify the effective method for detecting the emulator.

Q&A

Email: ik.kim@dankook.ac.kr