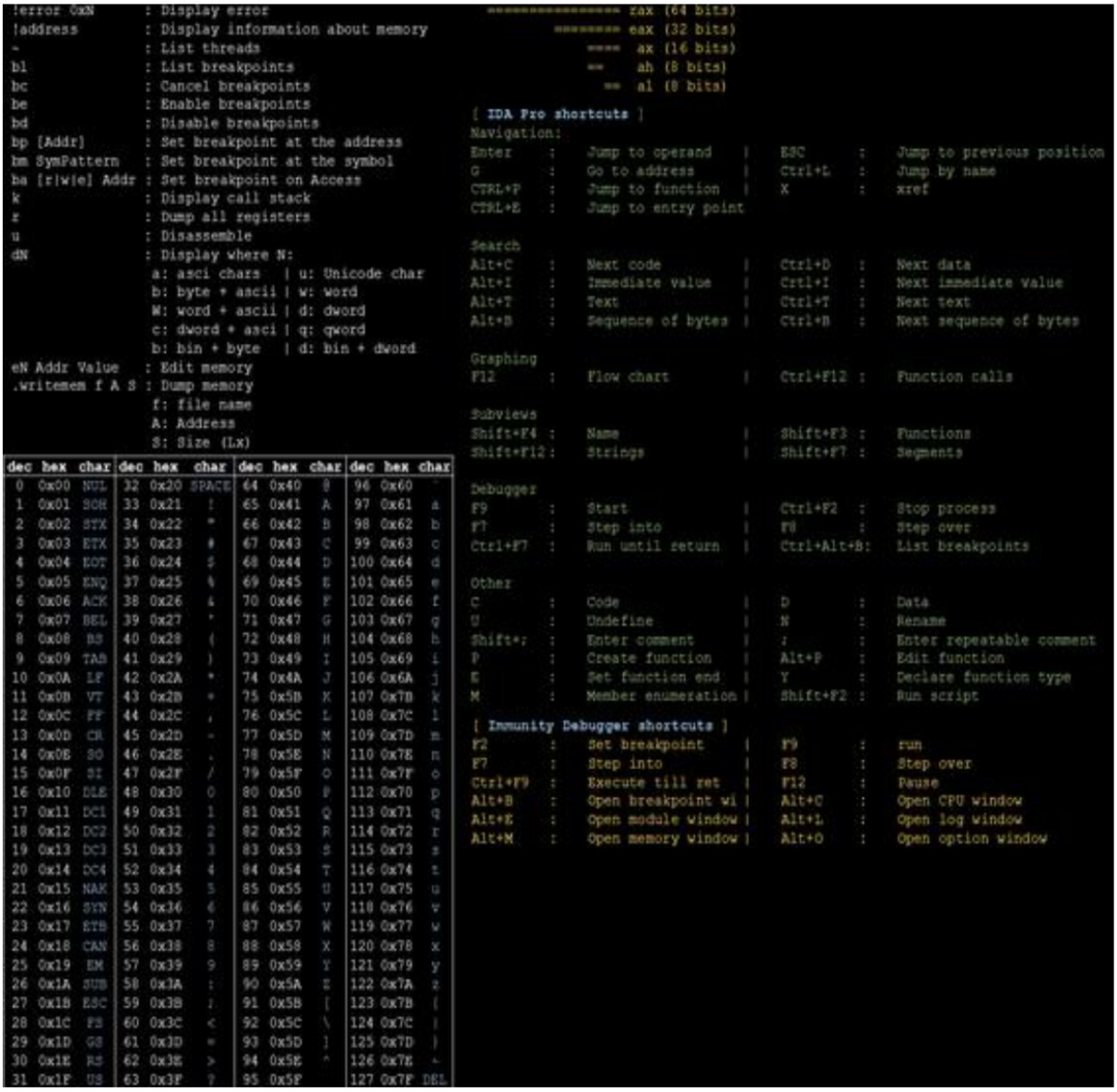


```
!error 0xN       : Display error
!address         : Display information about memory
~                : List threads
bl               : List breakpoints
bc               : Cancel breakpoints
be               : Enable breakpoints
bd               : Disable breakpoints
bp [Addr]        : Set breakpoint at the address
bm SymPattern    : Set breakpoint at the symbol
ba [r|w|e] Addr  : Set breakpoint on Access
k                : Display call stack
r                : Dump all registers
u                : Disassemble
dN               : Display where N:
                    a: asci chars   | u: Unicode char
                    b: byte + ascii | w: word
                    W: word + ascii | d: dword
                    c: dword + asci | q: qword
                    b: bin + byte   | d: bin + dword
eN Addr Value    : Edit memory
.writemem f A S  : Dump memory
                    f: file name
                    A: Address
                    S: Size (Lx)
```

dec	hex	char	dec	hex	char	dec	hex	char	dec	hex	char
0	0x00	NUL	32	0x20	SPACE	64	0x40	@	96	0x60	`
1	0x01	SOH	33	0x21	!	65	0x41	A	97	0x61	a
2	0x02	STX	34	0x22	"	66	0x42	B	98	0x62	b
3	0x03	ETX	35	0x23	#	67	0x43	C	99	0x63	c
4	0x04	EOT	36	0x24	$	68	0x44	D	100	0x64	d
5	0x05	ENQ	37	0x25	%	69	0x45	E	101	0x65	e
6	0x06	ACK	38	0x26	&	70	0x46	F	102	0x66	f
7	0x07	BEL	39	0x27	'	71	0x47	G	103	0x67	g
8	0x08	BS	40	0x28	(	72	0x48	H	104	0x68	h
9	0x09	TAB	41	0x29	)	73	0x49	I	105	0x69	i
10	0x0A	LF	42	0x2A	*	74	0x4A	J	106	0x6A	j
11	0x0B	VT	43	0x2B	+	75	0x5B	K	107	0x7B	k
12	0x0C	FF	44	0x2C	,	76	0x5C	L	108	0x7C	l
13	0x0D	CR	45	0x2D	-	77	0x5D	M	109	0x7D	m
14	0x0E	SO	46	0x2E	.	78	0x5E	N	110	0x7E	n
15	0x0F	SI	47	0x2F	/	79	0x5F	O	111	0x7F	o
16	0x10	DLE	48	0x30	0	80	0x50	P	112	0x70	p
17	0x11	DC1	49	0x31	1	81	0x51	Q	113	0x71	q
18	0x12	DC2	50	0x32	2	82	0x52	R	114	0x72	r
19	0x13	DC3	51	0x33	3	83	0x53	S	115	0x73	s
20	0x14	DC4	52	0x34	4	84	0x54	T	116	0x74	t
21	0x15	NAK	53	0x35	5	85	0x55	U	117	0x75	u
22	0x16	SYN	54	0x36	6	86	0x56	V	118	0x76	v
23	0x17	ETB	55	0x37	7	87	0x57	W	119	0x77	w
24	0x18	CAN	56	0x38	8	88	0x58	X	120	0x78	x
25	0x19	EM	57	0x39	9	89	0x59	Y	121	0x79	y
26	0x1A	SUB	58	0x3A	:	90	0x5A	Z	122	0x7A	z
27	0x1B	ESC	59	0x3B	;	91	0x5B	[	123	0x7B	{
28	0x1C	FS	60	0x3C	<	92	0x5C	\	124	0x7C	|
29	0x1D	GS	61	0x3D	=	93	0x5D	]	125	0x7D	}
30	0x1E	RS	62	0x3E	>	94	0x5E	^	126	0x7E	~
31	0x1F	US	63	0x3F	?	95	0x5F	_	127	0x7F	DEL

```
================ rax (64 bits)
        ======== eax (32 bits)
            ====  ax (16 bits)
            ==    ah (8 bits)
              ==  al (8 bits)
```

[IDA Pro shortcuts]

Navigation:

Enter	:	Jump to operand	ESC	:	Jump to previous position
G	:	Go to address	Ctrl+L	:	Jump by name
CTRL+P	:	Jump to function	X	:	xref
CTRL+E	:	Jump to entry point			

Search

Alt+C	:	Next code	Ctrl+D	:	Next data
Alt+I	:	Immediate value	Ctrl+I	:	Next immediate value
Alt+T	:	Text	Ctrl+T	:	Next text
Alt+B	:	Sequence of bytes	Ctrl+B	:	Next sequence of bytes

Graphing

F12	:	Flow chart	Ctrl+F12	:	Function calls

Subviews

Shift+F4	:	Name	Shift+F3	:	Functions
Shift+F12	:	Strings	Shift+F7	:	Segments

Debugger

F9	:	Start	Ctrl+F2	:	Stop process
F7	:	Step into	F8	:	Step over
Ctrl+F7	:	Run until return	Ctrl+Alt+B	:	List breakpoints

Other

C	:	Code	D	:	Data
U	:	Undefine	N	:	Rename
Shift+;	:	Enter comment	;	:	Enter repeatable comment
P	:	Create function	Alt+P	:	Edit function
E	:	Set function end	Y	:	Declare function type
M	:	Member enumeration	Shift+F2	:	Run script

[Immunity Debugger shortcuts]

F2	:	Set breakpoint	F9	:	run
F7	:	Step into	F8	:	Step over
Ctrl+F9	:	Execute till ret	F12	:	Pause
Alt+B	:	Open breakpoint wi	Alt+C	:	Open CPU window
Alt+E	:	Open module window	Alt+L	:	Open log window
Alt+M	:	Open memory window	Alt+O	:	Open option window

Máster en Análisis de Malware, Reversing y Bug Hunting

ENIIT INNOVA IT BUSINESS SCHOOL

excellence innova Consultoría y Formación S.L.

UCAM UNIVERSIDAD CATÓLICA DE MURCIA

Campus Internacional CIBERSEGURIDAD

Ramon Gonzalez Gaztelupe

1.- Instalación de entorno de análisis Android/IOS

Instalaremos el entorno de análisis Mobile Security Framework en MacOS, para ello tendremos que satisfacer unos requisitos previos antes de empezar con la instalación:

- **GIT**
- **Python 3.8-3.9**
- **JDK +8**
- **xcode-select**
- **wkhtmltopdf**

- **virtualbox como HyperVisor**
- **adb para comunicarse con el dispoitivo virtual por consola**

Nuestro sistema tenia instalado la versión python 3.11.3:

```
rajgon@RajKit.local ~
  python --version
Python 3.11.3
```

Instalamos la versión 3.9.16 con pyenv para funcionar con diferentes versiones:

- **pyenv install 3.9.16**

```
rajgon@RajKit.local ~/Mobile-Security-Framework-MobSF  <master>
  pyenv versions
  system
  2.7.18
  3.9.16
  3.10.11
* 3.11.3 (set by /Users/rajgon/.pyenv/version)
```

Descargamos el repositorio MobSF:

- **git clone https://github.com/MobSF/Mobile-Security-Framework-MobSF.git**

Accedemos al directorio y elegimos localmente la versión de python 3.9.16, de tal forma que en ese directorio se use esa versión de Python sin afectar al resto del equipo

- **cd Mobile-Security-Framework-MobSF**
- **pyenv local 3.9.16**

Después creamos un entorno virtual para cuidar las dependencias con virtualenv y ejecutamos el script de instalación:

- **Virtualenv mobile-reversing**
- **source mobile-reversing/bin/activate**
- **./setup.sh**

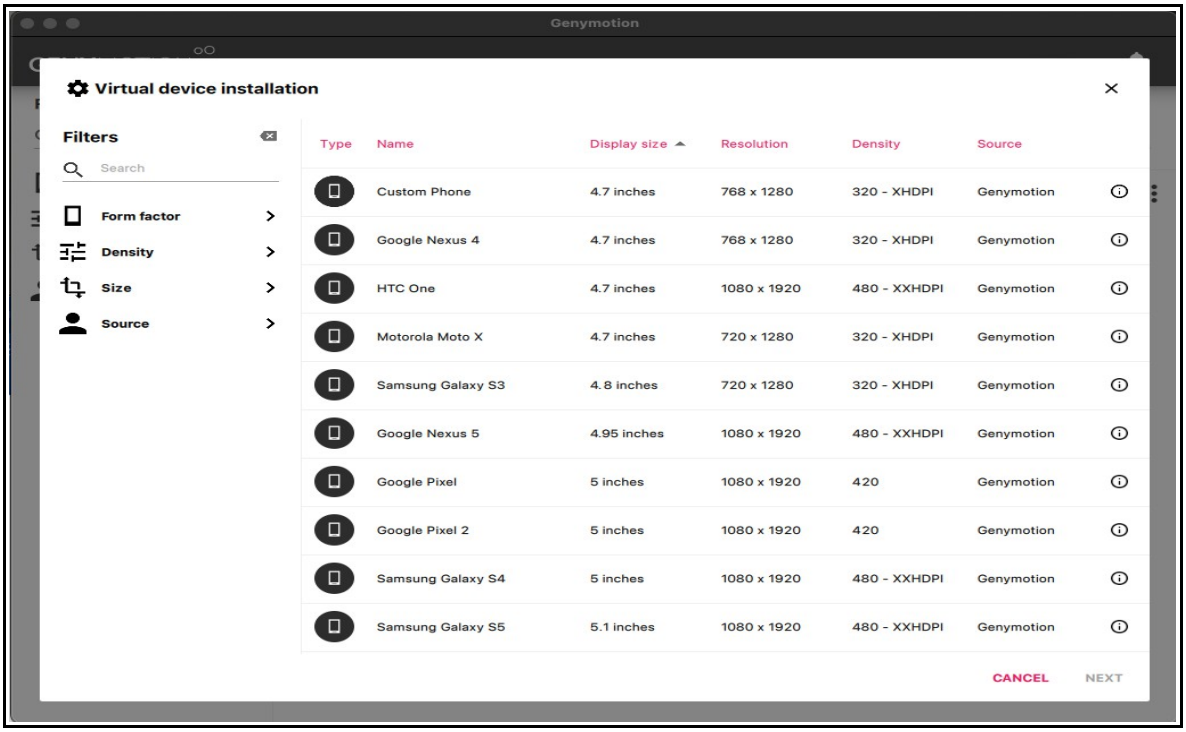
```
rajgon@RajKit.local ~/Mobile-Security-Framework-MobSF  <master*>
  virtualenv mobile-reversing                                                    127 ↵
created virtual environment CPython2.7.18.final.0-64 in 476ms
  creator CPython2macOsFramework(dest=/Users/rajgon/Mobile-Security-Framework-MobSF/mobile-reversing, clear=False, no_vcs_ignore=False, global=False)
  seeder FromAppData(download=False, pip=bundle, wheel=bundle, setuptools=bundle, via=copy, app_data_dir=/Users/rajgon/Library/Application Support/virtualenv)
    added seed packages: pip==20.3.4, setuptools==44.1.1, wheel==0.37.1
  activators NushellActivator,PythonActivator,FishActivator,CShellActivator,PowerShellActivator,BashActivator
rajgon@RajKit.local ~/Mobile-Security-Framework-MobSF  <master*>
  source mobile-reversing/bin/activate
(mobile-reversing) rajgon@RajKit.local ~/Mobile-Security-Framework-MobSF  <master*>
  ./setup.sh
```

```
  __  __       _    ____  _____       _____  __
 |  \/  | ___ | |__/ ___||  ___|_   _|___ / / /_
 | |\/| |/ _ \| '_ \___ \| |_  \ \ / / |_ \| '_ \
 | |  | | (_) | |_) |__) |  _|  \ V / ___) | (_) |
 |_|  |_|\___/|_.__/____/|_|     \_/ |____(_)___/

[INFO] 10/May/2023 13:16:53 - Mobile Security Framework v3.6.6 Beta
REST API Key: f4380c8c2aa90bc758998b954fee1b19e5556212472c6489662340bb08ea9cd7
[INFO] 10/May/2023 13:16:53 - OS: Darwin
[INFO] 10/May/2023 13:16:53 - Platform: macOS-13.1-x86_64-i386-64bit
[INFO] 10/May/2023 13:16:53 - Dist: darwin 22.2.0
[INFO] 10/May/2023 13:16:53 - MobSF Basic Environment Check
[WARNING] 10/May/2023 13:16:53 - Dynamic Analysis related functions will not work.
Make sure a Genymotion Android VM/Android Studio Emulator is running before performing Dynamic Analysis.
Operations to perform:
  Apply all migrations: StaticAnalyzer, auth, contenttypes, sessions
Running migrations:
  No migrations to apply.
[INFO] 10/May/2023 13:16:53 - Checking for Update.
[INFO] 10/May/2023 13:16:53 - No updates available.
wkhtmltopdf 0.12.6 (with patched qt)
[INSTALL] Installation Complete
```

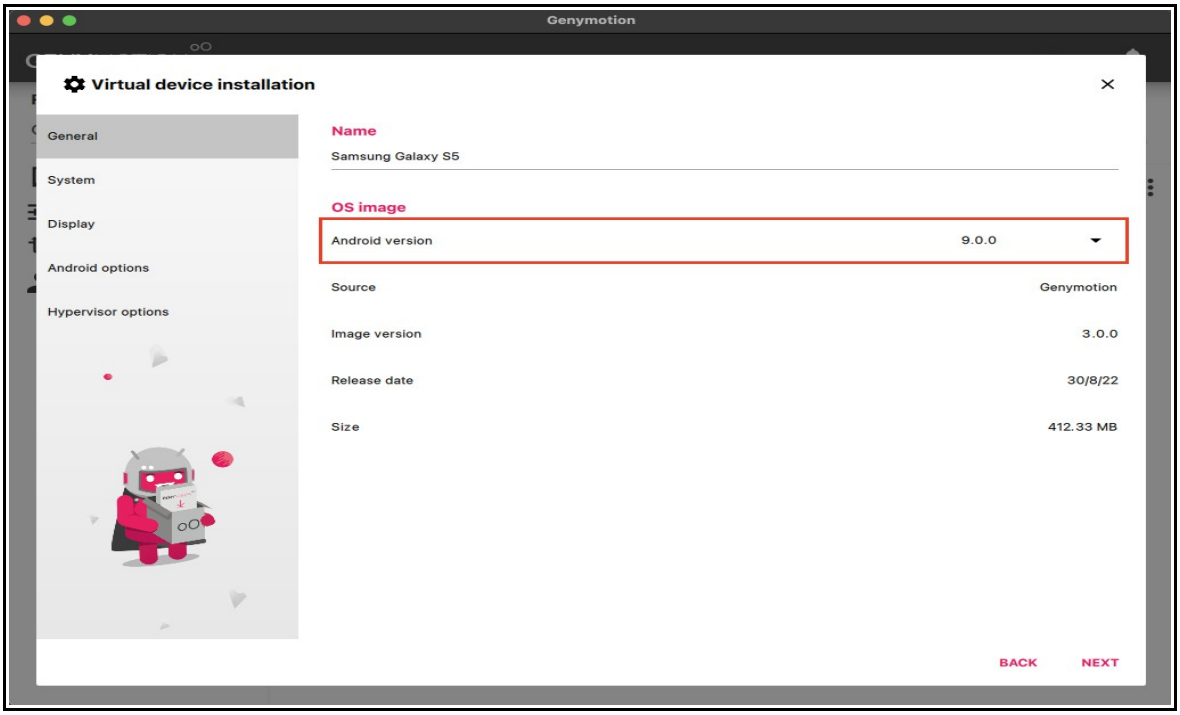
Como emulador usaremos GenyMotion que podemos descargarlo desde su pagina y utilizar previo registro:

Una vez instalado tenemos que crear y configurar un dispositivo virtual:

- Elegimos el dispositivo:

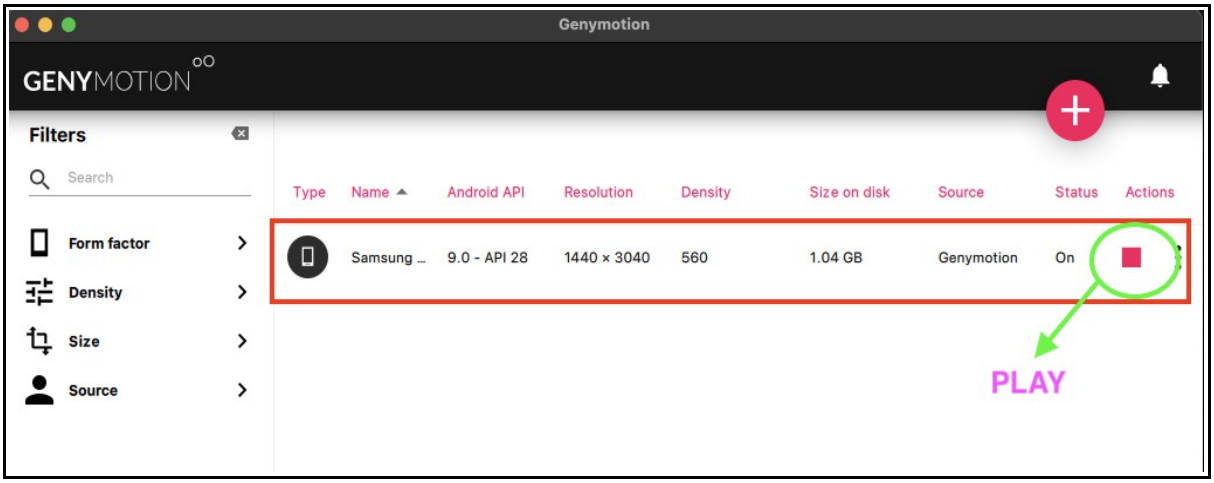


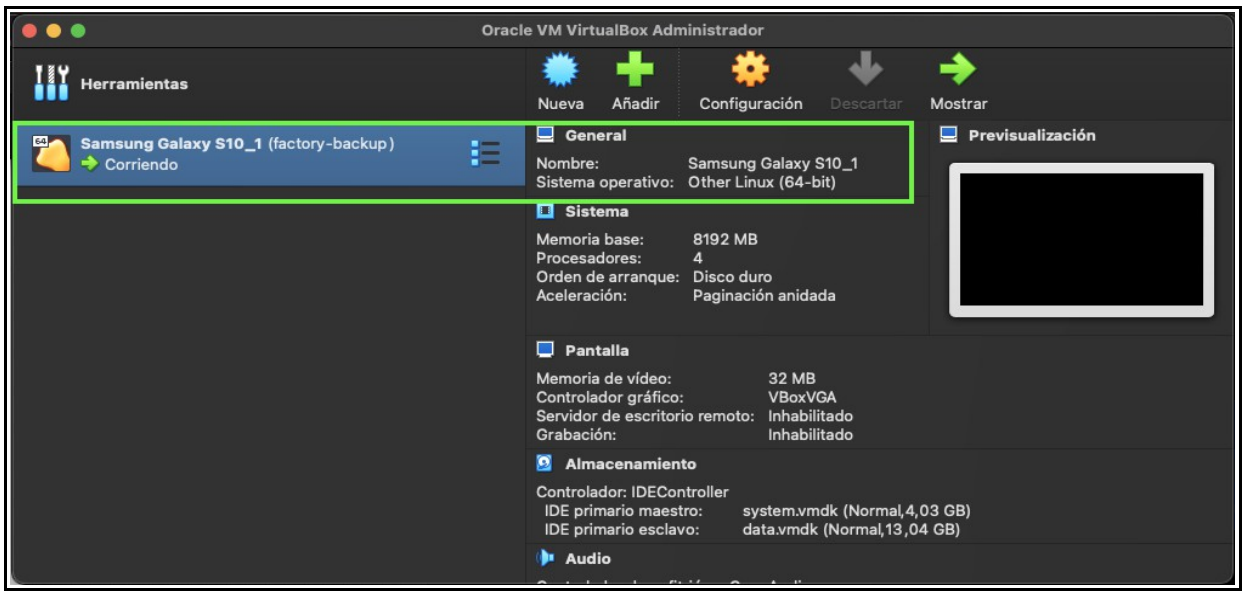
- La imagen del sistema en este caso android y para que sea apto para el análisis dinámico hasta API LEVEL 29, lo que implica una versión de Android no superior a la 10:



- Continuamos los siguientes pasos hasta que cree el dispositivo virtual y finalmente lo arrancamos.

GenyMotion utiliza VirtualBox para virtualizar el dispositivo utilizando su HyperV:



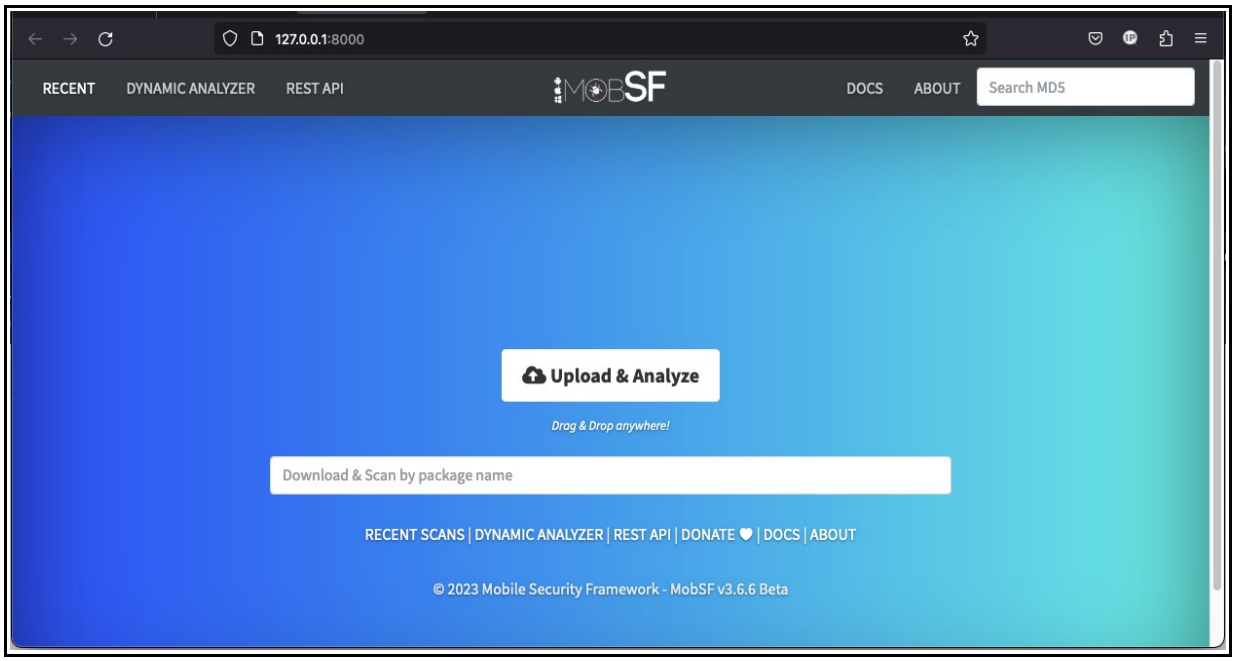


Una vez instalado MobSF y arrancado en dispositivo virtual android, volvemos al entorno virtual mobile-reversing y arrancamos MobSF que nos creara un servidor local con la interfaz gráfica web para el análisis tanto dinámico como estático:

- **/run.sh**

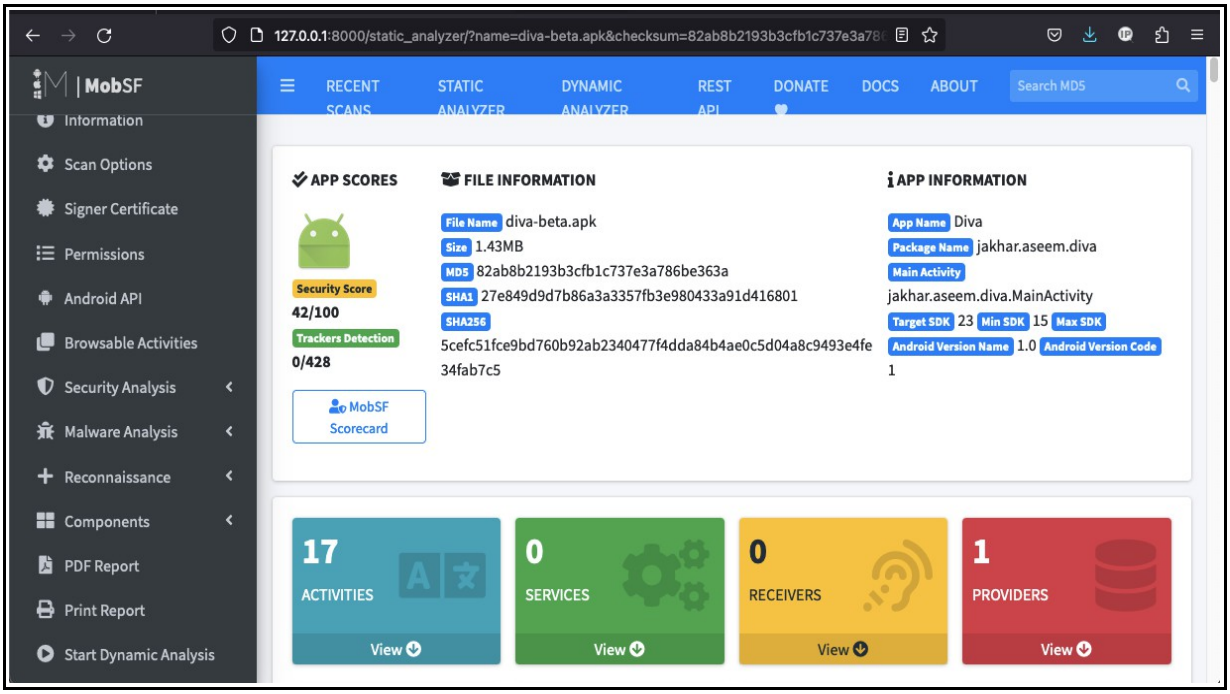
```
(mobile-reversing) ┌─rajgon@RajKit.local ~/Mobile-Security-Framework-MobSF ‹master*›
└─➤  ./run.sh
[2023-05-10 15:22:25 +0200] [43668] [INFO] Starting gunicorn 20.1.0
[2023-05-10 15:22:25 +0200] [43668] [INFO] Listening at: http://[::]:8000 (43668)
[2023-05-10 15:22:25 +0200] [43668] [INFO] Using worker: gthread
[2023-05-10 15:22:25 +0200] [43672] [INFO] Booting worker with pid: 43672
[INFO] 10/May/2023 13:22:42 -
  __  __       _     ____  _____       _____  __
 |  \/  | ___ | |__/ ___||  ___|_   _|___ / / /_
 | |\/| |/ _ \| '_ \___ \| |_  \ \ / / |_ \| '_ \
 | |  | | (_) | |_) |__) |  _|  \ V / ___) | (_) |
 |_|  |_|\___/|_.__/____/|_|     \_/ |____(_)___/

[INFO] 10/May/2023 13:22:42 - Mobile Security Framework v3.6.6 Beta
REST API Key: f4380c8c2aa90bc758998b954fee1b19e5556212472c6489662340bb08ea9cd7
[INFO] 10/May/2023 13:22:42 - OS: Darwin
[INFO] 10/May/2023 13:22:42 - Platform: macOS-13.1-x86_64-i386-64bit
[INFO] 10/May/2023 13:22:42 - Dist: darwin 22.2.0
[INFO] 10/May/2023 13:22:42 - MobSF Basic Environment Check
[INFO] 10/May/2023 13:22:42 - Checking for Update.
[INFO] 10/May/2023 13:22:42 - No updates available.
[INFO] 10/May/2023 13:22:58 - MobSFying Android instance
[INFO] 10/May/2023 13:23:01 - ADB Restarted
[INFO] 10/May/2023 13:23:01 - Waiting for 2 seconds...
[INFO] 10/May/2023 13:23:03 - Connecting to Android 192.168.56.103:5555
[INFO] 10/May/2023 13:23:03 - Waiting for 2 seconds...
[INFO] 10/May/2023 13:23:05 - Restarting ADB Daemon as root
[INFO] 10/May/2023 13:23:05 - Waiting for 2 seconds...
[INFO] 10/May/2023 13:23:07 - Reconnecting to Android Device
[INFO] 10/May/2023 13:23:07 - Waiting for 2 seconds...
[INFO] 10/May/2023 13:23:09 - Found Genymotion x86 Android VM
[INFO] 10/May/2023 13:23:09 - Remounting
[INFO] 10/May/2023 13:23:09 - Performing System check
[INFO] 10/May/2023 13:23:09 - Android API Level identified as 28
[INFO] 10/May/2023 13:23:09 - Android Version identified as 9.0
[INFO] 10/May/2023 13:23:09 - Android OS architecture identified as x86
[INFO] 10/May/2023 13:23:10 - Downloading binary frida-server-15.2.2-android-x86
[INFO] 10/May/2023 13:23:13 - Copying frida server for x86
[INFO] 10/May/2023 13:23:14 - Installing MobSF RootCA
[INFO] 10/May/2023 13:23:14 - Installing MobSF Clipboard Dumper
[INFO] 10/May/2023 13:23:16 - MobSFying Completed!
```

El servidor por defecto arranca en el puerto 8000, accedemos a la interfaz web:



Antes de realizar un análisis, descargamos el código de al APP vulnerable DIVA y la compilamos en Android Studio.

Para realizar el análisis arrastramos la .APK al escaner y nos devolverá un análisis estático:



Si queremos realizar un análisis estático y usar FRIDA, arrancaremos Dynamic Anlysis: