



Mobile Phone Security



Dr. Digvijaysinh Rathod

Associate Professor & Associate Dean

School of Cyber Security and Digital Forensics

National Forensic Sciences University with status of Institution of National Importance

digvijay.rathod@gfsu.edu.in

Frida

**Dynamic instrumentation toolkit for
developers, reverse-engineers, and
security researchers**

What is frida ?

- ✓ Frida's core is written in C
- ✓ it's a dynamic code instrumentation toolkit.
- ✓ **Scriptable:**
 - ✓ It lets you inject snippets of JavaScript or your own library into native apps on Windows, macOS, GNU/Linux, iOS, Android, and QNX.
 - ✓ Frida also provides you with some simple tools built on top of the Frida API.
 - ✓ These can be used as-is, tweaked to your needs, or serve as examples of how to use the API.

Ref: <https://frida.re/docs/home/>

What is frida ?

✓ **Portable:**

- ✓ Works on Windows, macOS, GNU/Linux, iOS, Android, and QNX.
- ✓ Install the Node.js bindings from npm, grab a Python package from PyPI, or use Frida through its Swift bindings, .NET bindings, Qt/Qml bindings, or C API.

✓ **Free:**

- ✓ Frida is and will always be free software

Ref: <https://frida.re/docs/home/>

What is frida ?

- ✓ **Battle-tested:**
 - ✓ Frida has a comprehensive test-suite and has gone through years of rigorous testing across a broad range of use-cases.

Ref: <https://frida.re/docs/home/>

Why a Python API, but JavaScript debugging logic?

- ✓ Frida's core is written in C and injects QuickJS into the target processes, where your JS gets executed with full access to memory, hooking functions and even calling native functions inside the process.
- ✓ There's a bi-directional communication channel that is used to talk between your app and the JS running inside the target process.

Ref: <https://frida.re/docs/home/>

Why a Python API, but JavaScript debugging logic?

- ✓ Using Python and JS allows for quick development with a risk-free API. Frida can help you easily catch errors in JS and provide you an exception rather than crashing.
- ✓ You can use Frida from C directly, and on top of this C core there are multiple language bindings, e.g. Node.js, Python, Swift, .NET, Qml, etc. It is very easy to build additional bindings for other languages and environments.

Ref: <https://frida.re/docs/home/>

Why a Python API, but JavaScript debugging logic?

Common purpose of Frida,

- ✓ Spy on Crypto APIs
- ✓ Modify function's output
- ✓ Bypass AES encryption
- ✓ Bypass SSLPinning and Root detection
- ✓ Trace private application code
- ✓ Bypass various software sided locks (like applock)

Ref: <https://frida.re/docs/home/>

Installation – frida server in Genymotion

- ✓ Python – latest 3.x is highly recommended
- ✓ Create Genymotion android device with API > 7.0 otherwise you face frida installation issues.
- ✓ Ping Genymotion AVD from Kalu
- ✓ Server installation in the Genymotion:
- ✓ Find the android device CPU architecture
- ✓ Query all configuration information of Android devices: `adb shell getprop`.
 - ✓ `>adb shell getprop`
 - ✓ `>adb shell getprop | grep abi`
 - ✓ `>adb shell getprop ro.product.cpu.abi`
- ✓ Genymotion Emulator shows its X86 CPU architecture

Installation – frida server in Genymotion

- ✓ Visit : <https://github.com/frida/frida/releases>
- ✓ Search for frida-server key word in the webpage

The screenshot shows a web browser window displaying the GitHub releases page for the repository `frida/frida`. The address bar shows the URL `github.com/frida/frida/releases`. A search bar in the top right corner contains the text `frida-server`, and a dropdown menu shows the search results, with `frida-server` selected. The main content area displays a list of release assets, each with a download icon, the asset name, and its size. The assets are organized into two groups: `frida-portal` and `frida-qml`, each with sub-groups for different operating systems and architectures. The `frida-server` assets are listed at the bottom of the page, with the `frida-server-15.1.14-android-arm64.xz` asset highlighted in yellow.

Asset Name	Size
<code>frida-portal-15.1.14-macos-arm64.xz</code>	
<code>frida-portal-15.1.14-macos-x86_64.xz</code>	1.97 MB
<code>frida-portal-15.1.14-windows-x86.exe.xz</code>	14.7 MB
<code>frida-portal-15.1.14-windows-x86_64.exe.xz</code>	15.5 MB
<code>frida-qml-15.1.14-linux-x86_64.tar.xz</code>	16.2 MB
<code>frida-qml-15.1.14-macos-x86_64.tar.xz</code>	9.39 MB
<code>frida-qml-15.1.14-windows-x86.exe</code>	13.4 MB
<code>frida-qml-15.1.14-windows-x86_64.exe</code>	14 MB
<code>frida-server-15.1.14-android-arm.xz</code>	6.54 MB
<code>frida-server-15.1.14-android-arm64.xz</code>	14 MB
<code>frida-server-15.1.14-android-x86.xz</code>	13.8 MB
<code>frida-server-15.1.14-android-x86_64.xz</code>	28 MB
<code>frida-server-15.1.14-linux-arm64.xz</code>	7 MB
<code>frida-server-15.1.14-linux-armhf.xz</code>	6.66 MB
<code>frida-server-15.1.14-linux-x86.xz</code>	8.15 MB
<code>frida-server-15.1.14-linux-x86_64.xz</code>	15.7 MB
<code>frida-server-15.1.14-macos-arm64.xz</code>	12.2 MB

Installation – frida server in Genymotion

- ✓ Download suitable version of Frida-server. In this case as we have X86 CPU architecture so I downloaded following version,

 frida-server-15.1.14-android-arm.xz	6.54 MB
 frida-server-15.1.14-android-arm64.xz	14 MB
 frida-server-15.1.14-android-x86.xz	13.8 MB
 frida-server-15.1.14-android-x86_64.xz	28 MB
 frida-server-15.1.14-linux-arm64.xz	7 MB
 frida-server-15.1.14-linux-armhf.xz	6.66 MB
 frida-server-15.1.14-linux-x86.xz	8.15 MB
 frida-server-15.1.14-linux-x86_64.xz	15.7 MB
 frida-server-15.1.14-macos-arm64.xz	12.2 MB

Installation – frida server in Genymotion

- ✓ Unzip the frida-server-15-14-1-android.xz
- ✓ Unzip the folder and rename the file as frida-server
- ✓ `> adb connect ip-address-of-genymotion-AVD`
- ✓ `>adb devices`
- ✓ `>adb push frida-server /data/local/tmp`
- ✓ `# check the location of Frida-server file`
- ✓ `>adb shell`
- ✓ `root-android > cd /data/local/tmp`
- ✓ `# check that frida-server file is there or not`

Installation – frida server in Genymotion

- ✓ `root-android/data/local/tmp> chmod 755 frida-server`
- ✓ `root-android/data/local/tmp> ./frida-server`
- ✓ frida server will start and waiting for frida client to connect.
- ✓ If you receive any error such as
- ✓ `reloc_library [1311] : 1319 cannot locate “getauxval”.....`

CANNOT LINK EXECUTABLE

It means you are using ancient you are using ancient versions of Android

Ref: <https://frida.re/docs/android/>

Installation – frida in the kali or santoku

- ✓ > sudo pip install frida or
- ✓ >sudo pip install frida-tools
- ✓ > sudo pip3 install frida
- ✓ # once it's install successful then test the Friday
- ✓ >frida-ps -U
- ✓ >frida-ps this will shows the module.
- ✓ # if you find list of module it shows that frida is working.



Mobile Phone Security



Dr. Digvijaysinh Rathod

Associate Professor & Associate Dean

School of Cyber Security and Digital Forensics

National Forensic Sciences University with status of Institution of National Importance

digvijay.rathod@gfsu.edu.in