FRIDA All The Things

PiXeLS Camp

INTEGRITY

Herman Duarte
@hdontwit

whoami

- Lead Security Consultant @ INTEGRITY S.A.
- Penetration testing:
  - Mobile Apps
  - Web Apps
  - Infrastructure / Wireless
  - Hardware hacking
- BSc in Information Systems and Computer Engineering
- OSCP, CISSP Associate, ISO27001LA, CCNA

PiXELS Camp INTEGRITY

ls -l

- What is this talk about ?
- Frida
- Interacting with Frida
- Frida Scripts
- Frameworks based on Frida
- Resources & References

cd “What is this talk about ?”

- Frida for the beginner
- We're going from someone that has never seen Frida, to being able to use it to dynamically instrument an iOS application
- We will be using iOS as the example platform because its what lead me to Frida
- We will touch in a few iOS concepts along the way (for us to better understand the examples)
- The examples can be applied to the other platforms with some changes

cd Frida

- Open source tool used to explore black box apps, by injecting JavaScript code
- Scriptable, Portable
- Developed in C (Frida core), and has a Python API to inject JavaScript code :)
- Developed by @oleavr
- It has a growing and vibrant community

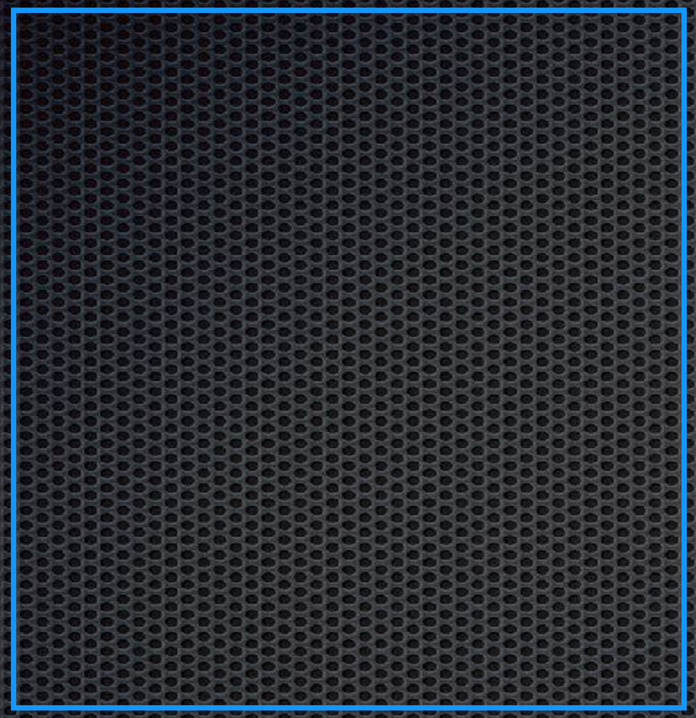
cd Frida

- Hook functions
- Enumerate modules (targets loaded dynamic/shared libs)
  - Enumerate their imported and exported functions
- Read and Write memory and scan it for patterns
- .. and much more

cd Frida

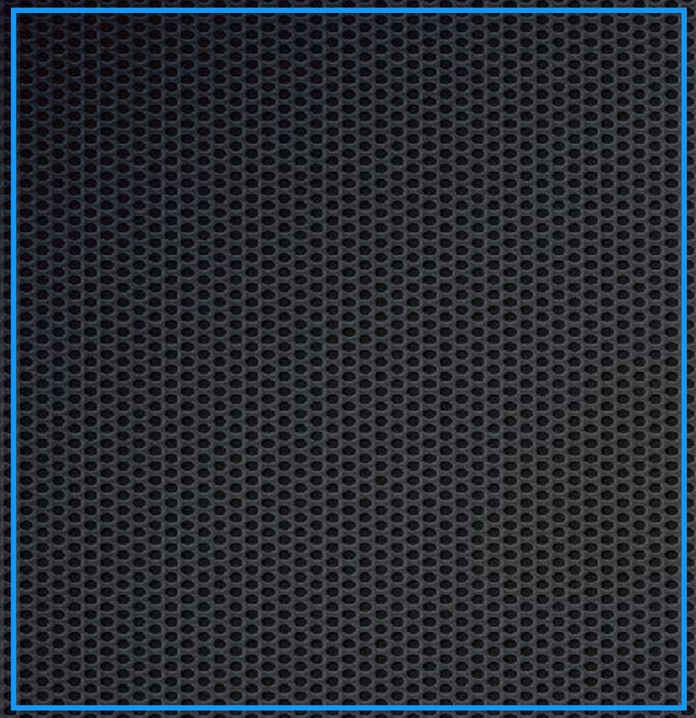
cd Frida

Frida based client



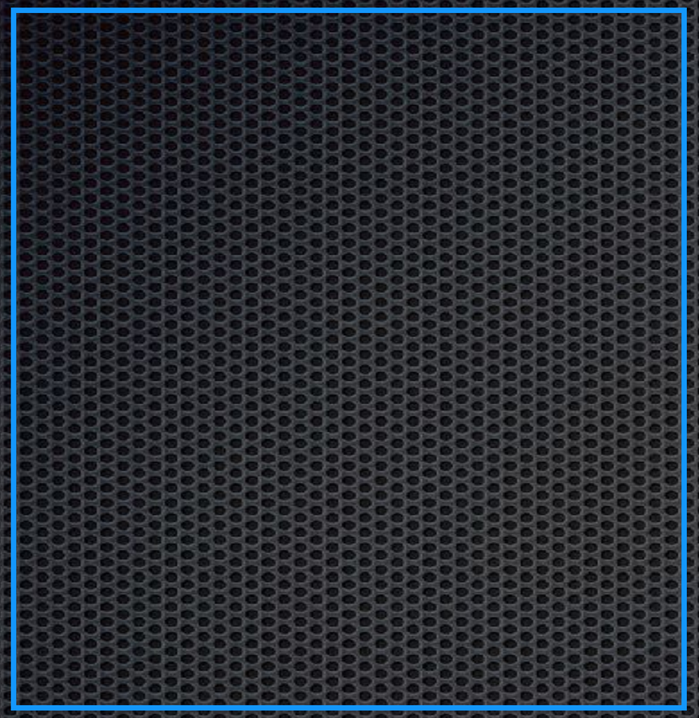
cd Frida

Frida based client

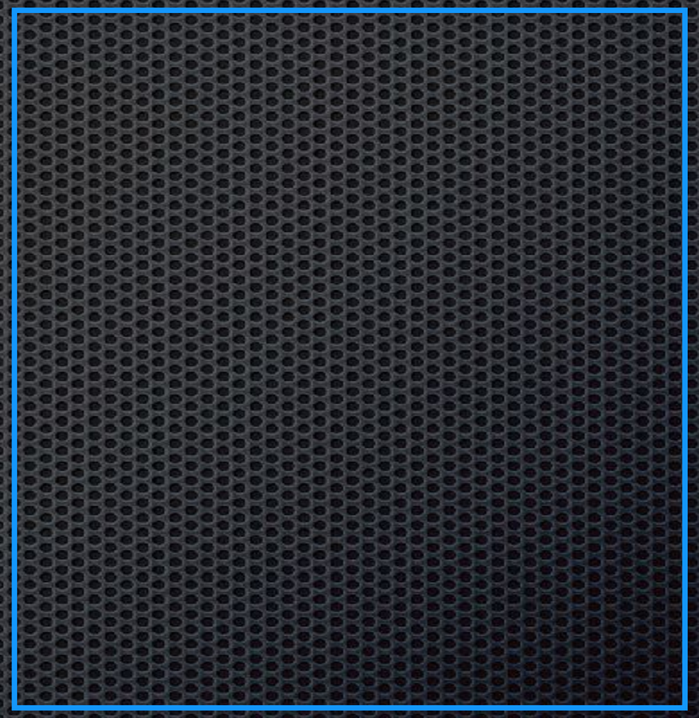


cd Frida

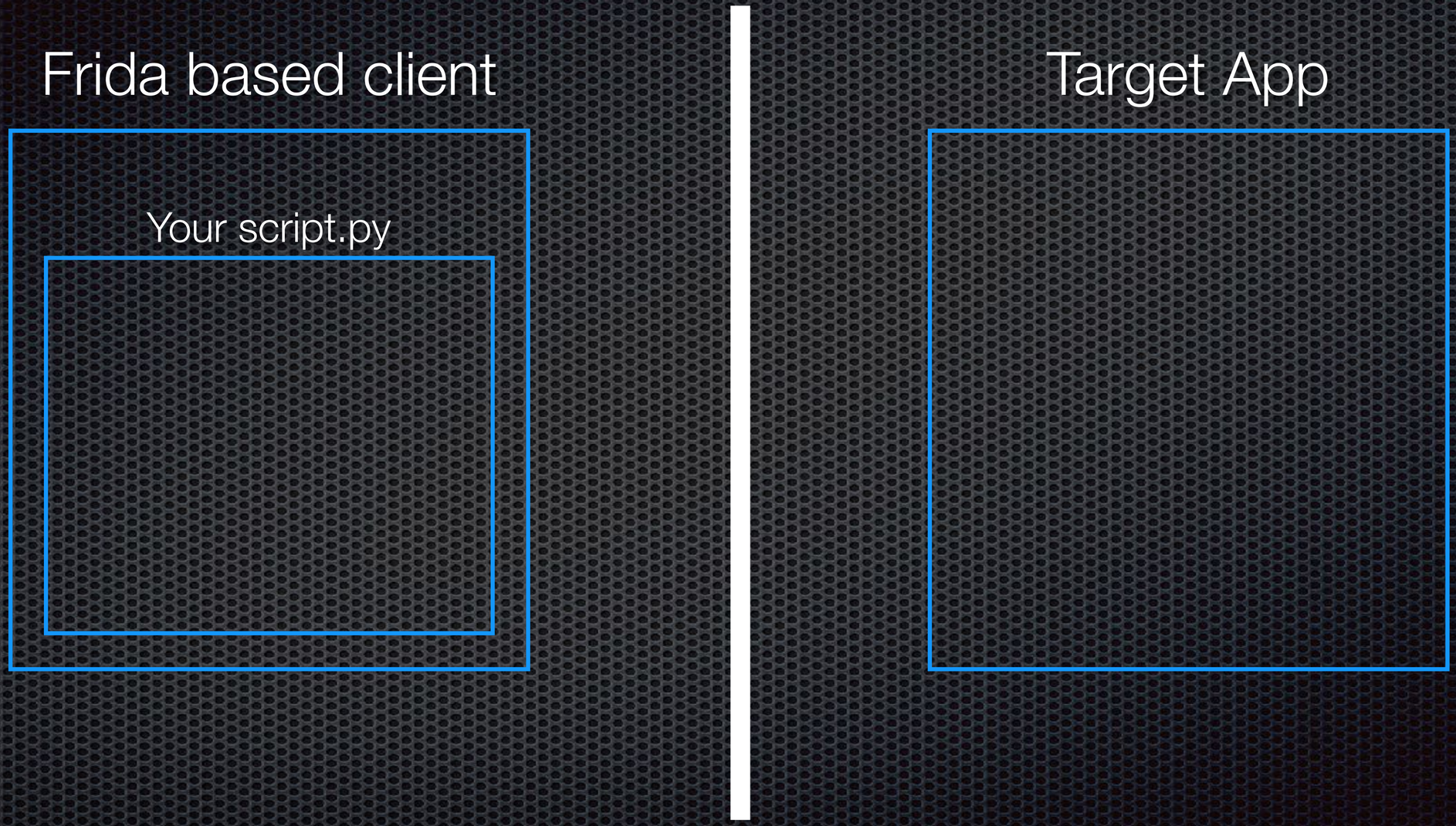
Frida based client



Target App

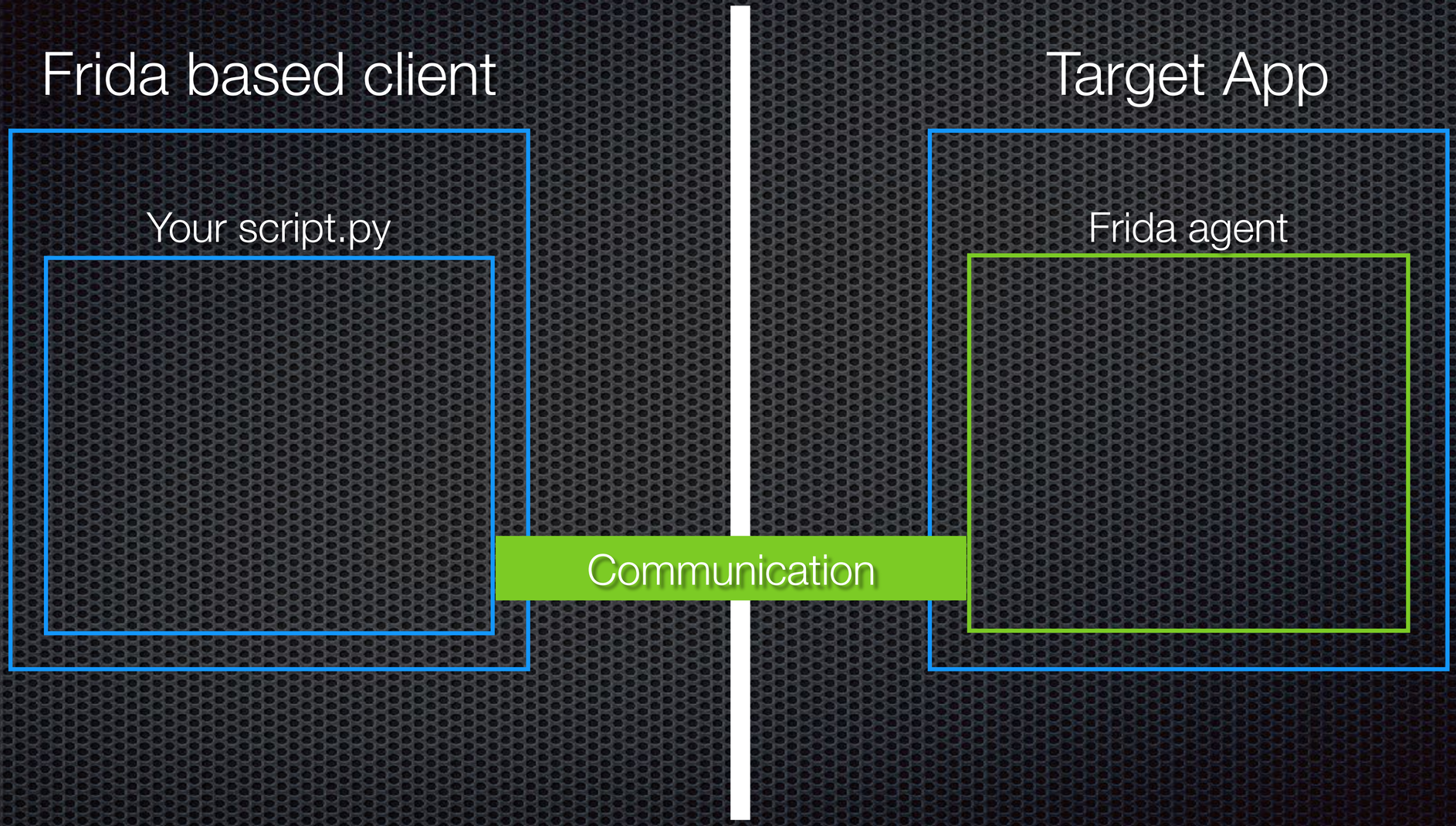


cd Frida

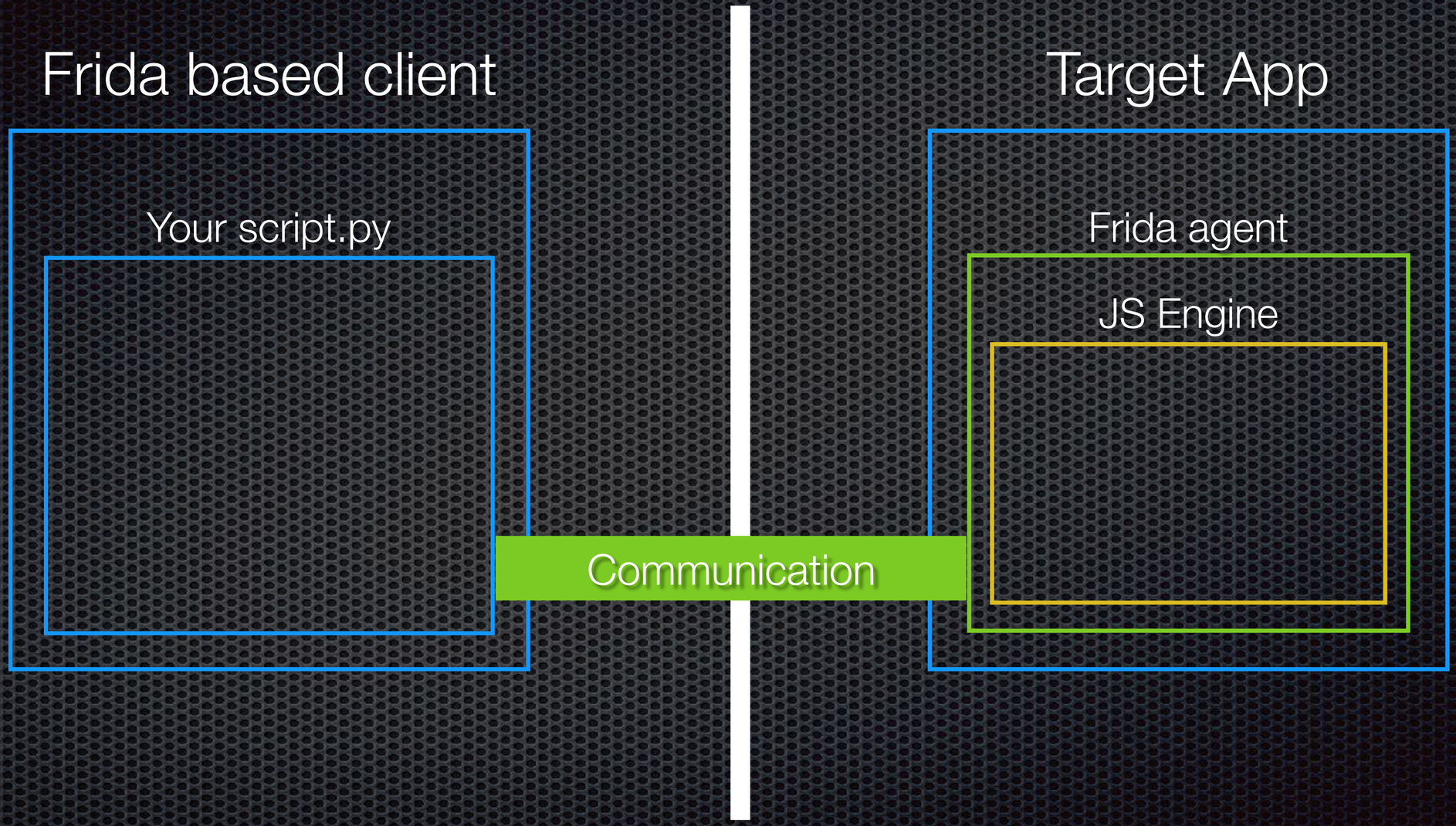


cd Frida

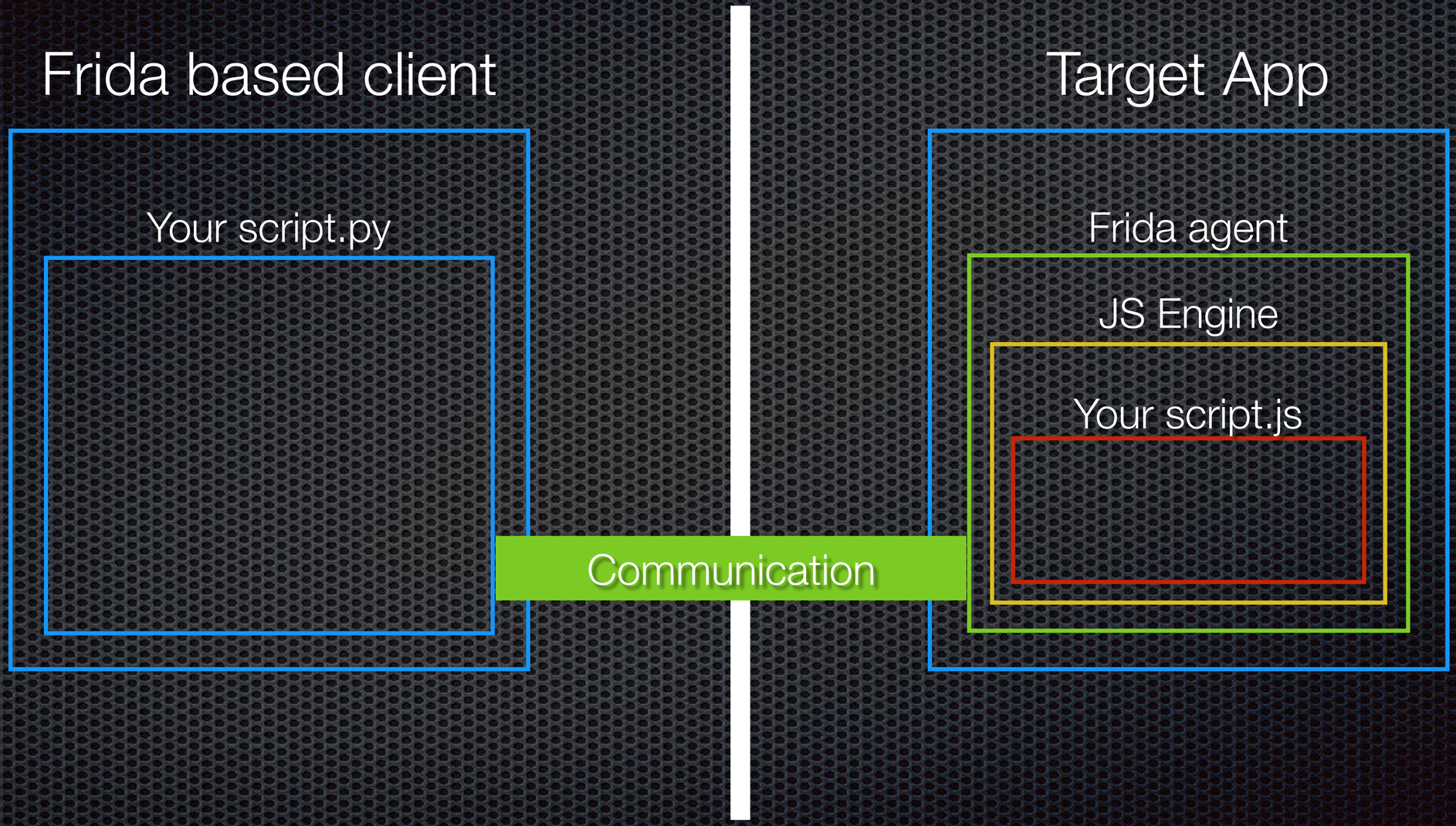
cd Frida


Frida based client
Target App
Your script.py
Frida agent
Communication

cd Frida


Frida based client
Target App
Your script.py
Frida agent
JS Engine
Communication

cd Frida


Frida based client
Target App
Your script.py
Frida agent
JS Engine
Your script.js
Communication

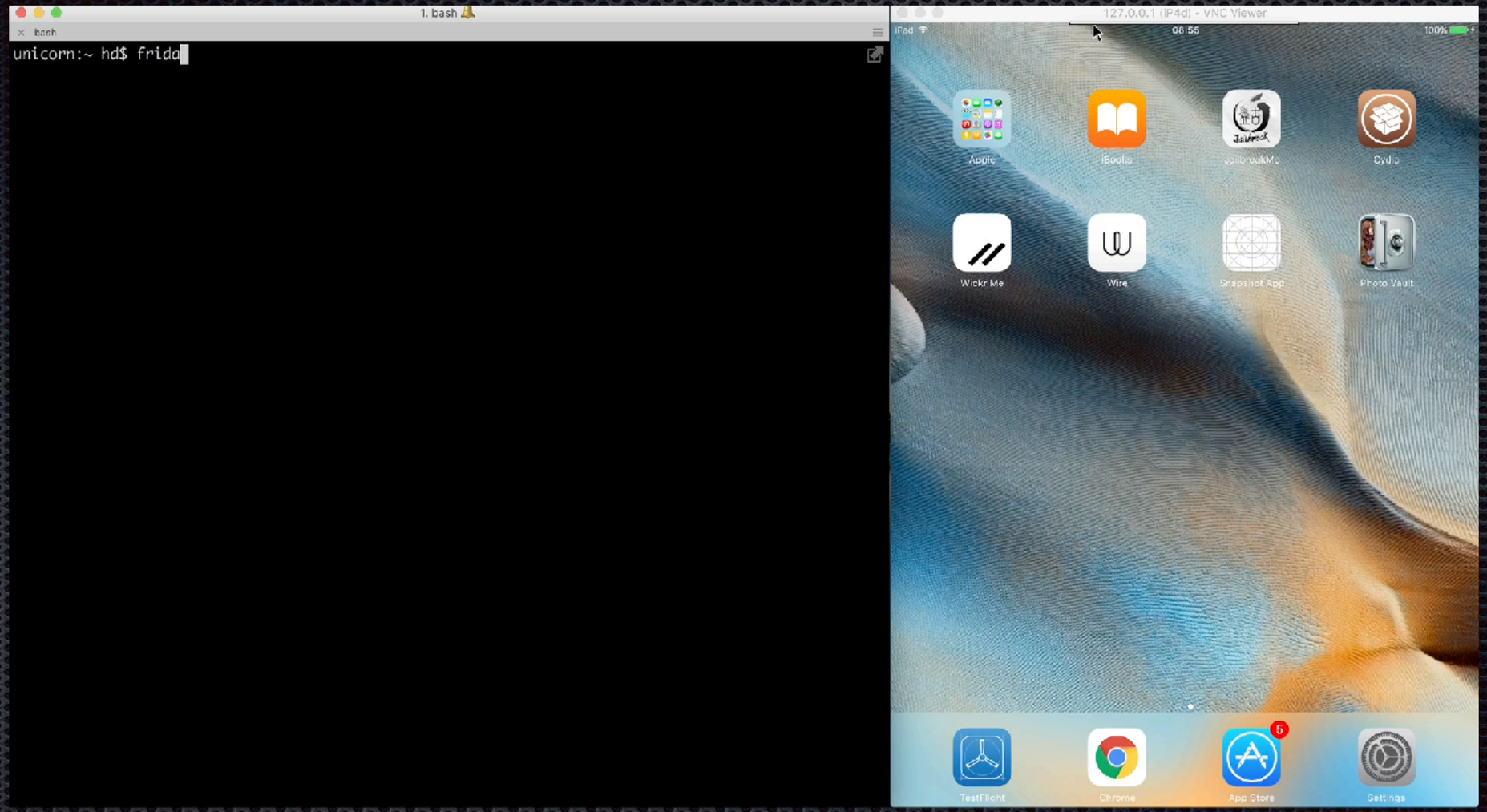
cd “Interacting with Frida”

- pip install --user frida - to install the client tools
- https://build.frida.re/ - you can find pre compiled binaries of frida server for the supported platforms
  - iOS, Android, Mac OS, Windows, Linux, QNX
  - x86, x64, arm, arm64, armhf, mips

cd “Interacting with Frida”

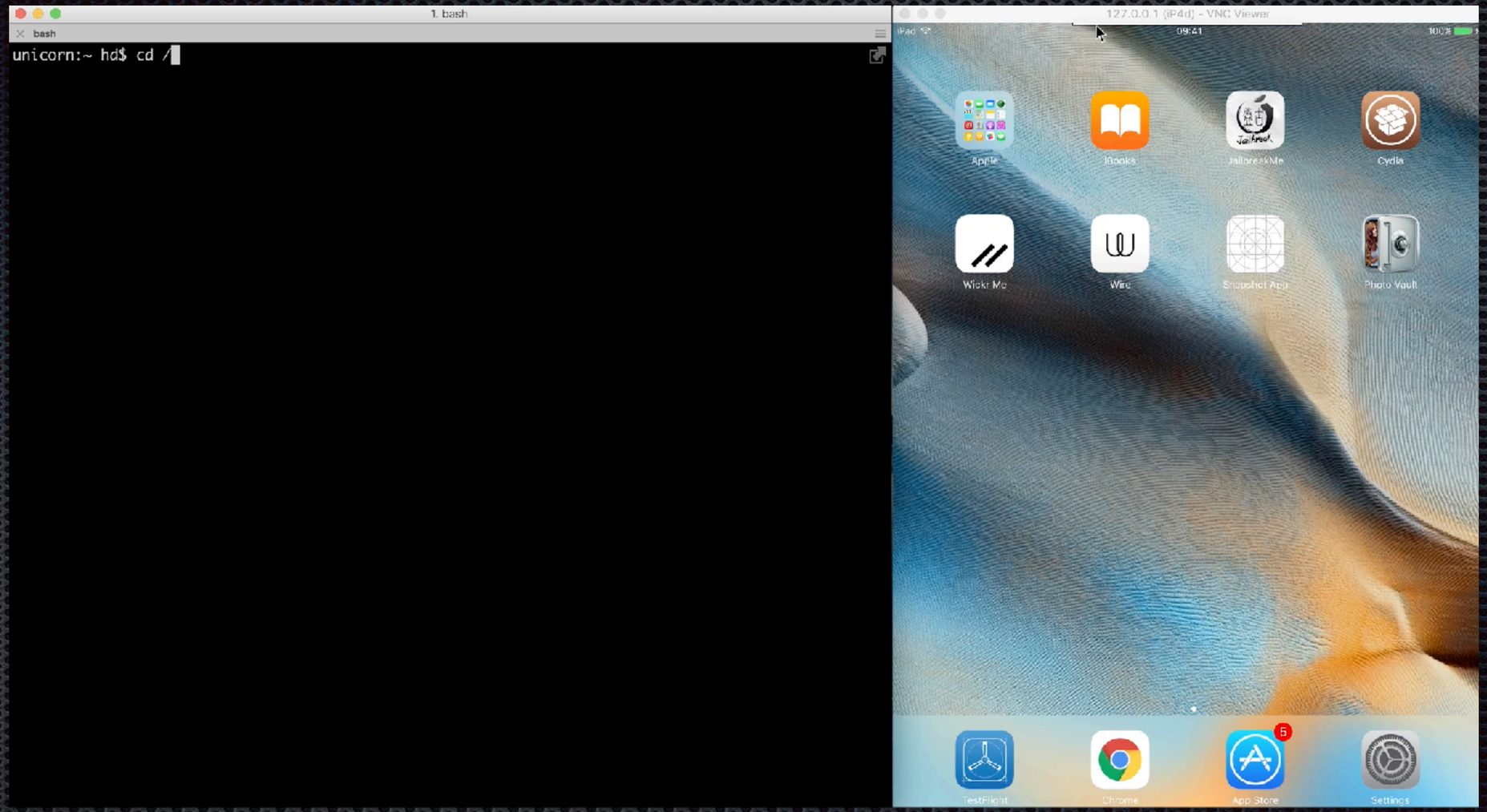
- Frida comes with a few command line tools written in Python:
  - frida-ps, frida-ls-devices, frida-kill
  - frida-discover
  - frida-trace
  - frida
- One can use it directly from C or from it’s bindings in Python, Swift, .NET, Node.js, QML, etc

./frida-discover.py



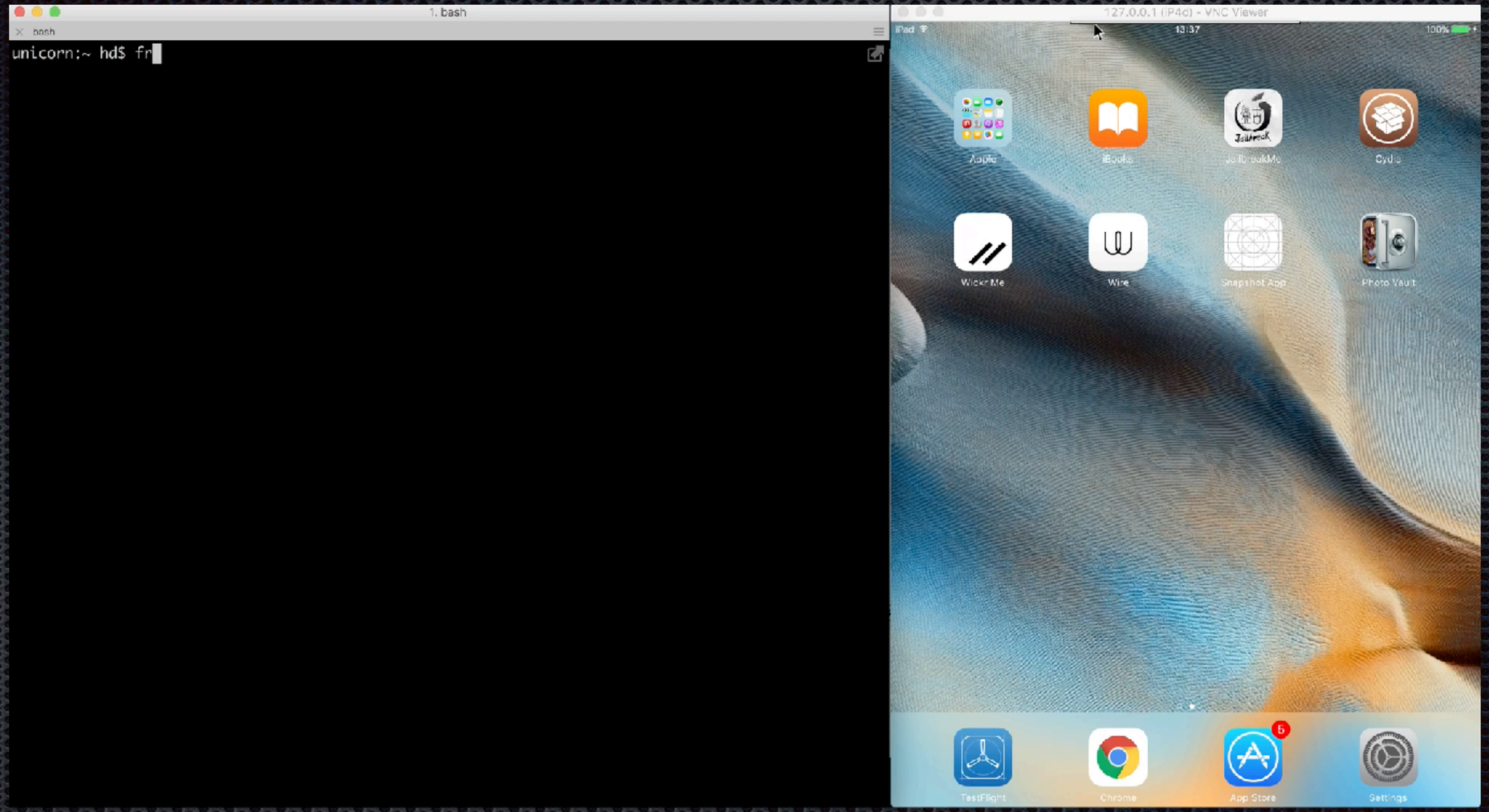
PiXELS Camp INTEGRITY

./frida-trace.py



PIXELS Camp INTEGRITY

./frida.py



PiXeLS Camp INTEGRITY

cd “Frida Scripts”

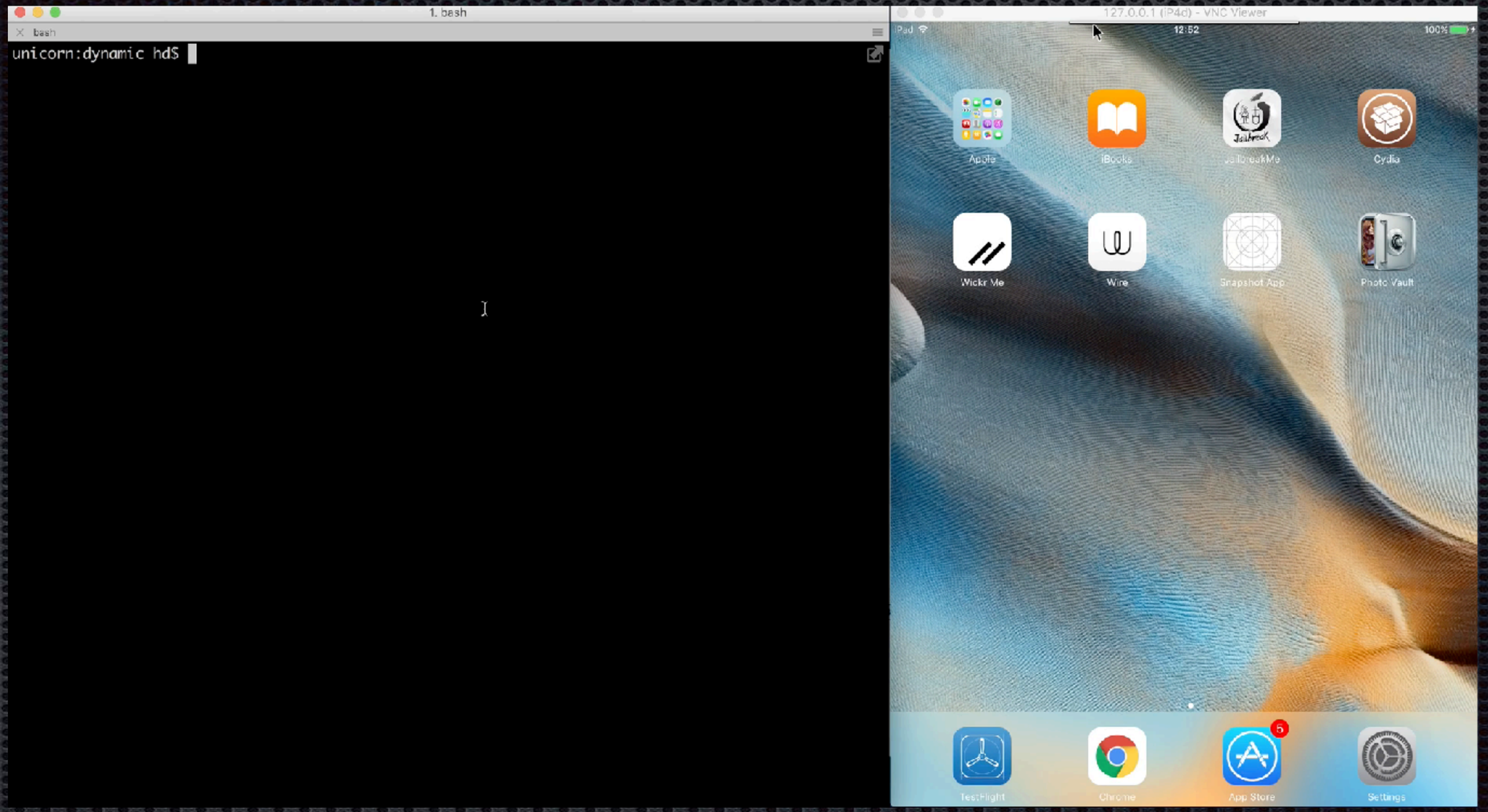
- iOS class dumper
- iOS hook skeleton
- iOS TLS pinning bypass

./classdumper.py

- Current iOS class dumper, class-dump, doesn’t support Swift binaries, only Objective-C ones
- Frida can dump all methods of **all classes** or class specific methods for **all classes**

- With classdumper.py we’re using Frida to find all class specific methods **that belong to the application binary** and/or bundled libraries (frameworks).

./classdumper.py



PiXELS Camp INTEGRITY

cat classdumper.py

```python
"""
Frida script to dump the classes specific to the app binary

# (c) 2017 INTEGRITY S.A.
# By: Herman Duarte <hd@integrity.pt>

"""

import frida
from time import sleep
import sys

script_test = '''
'use strict';

rpc.exports = {
    classes: function () {
        if (ObjC.available) { return ObjC.classes; }
    },
    getClassOwnMethods: function (className) {
        return ObjC.classes[className].$ownMethods;
    }
};
'''

#appToLaunch = 'com.example.ios.app'
appToLaunch = None

if appToLaunch == None:
    if len(sys.argv) > 1:
        appToLaunch = sys.argv[1]
        print(appToLaunch)
    else:
        print("usage: python3 " + sys.argv[0] + " <app identifier>")
        print("       python3 " + sys.argv[0] + " com.example.ios.app")
        sys.exit (0)

```

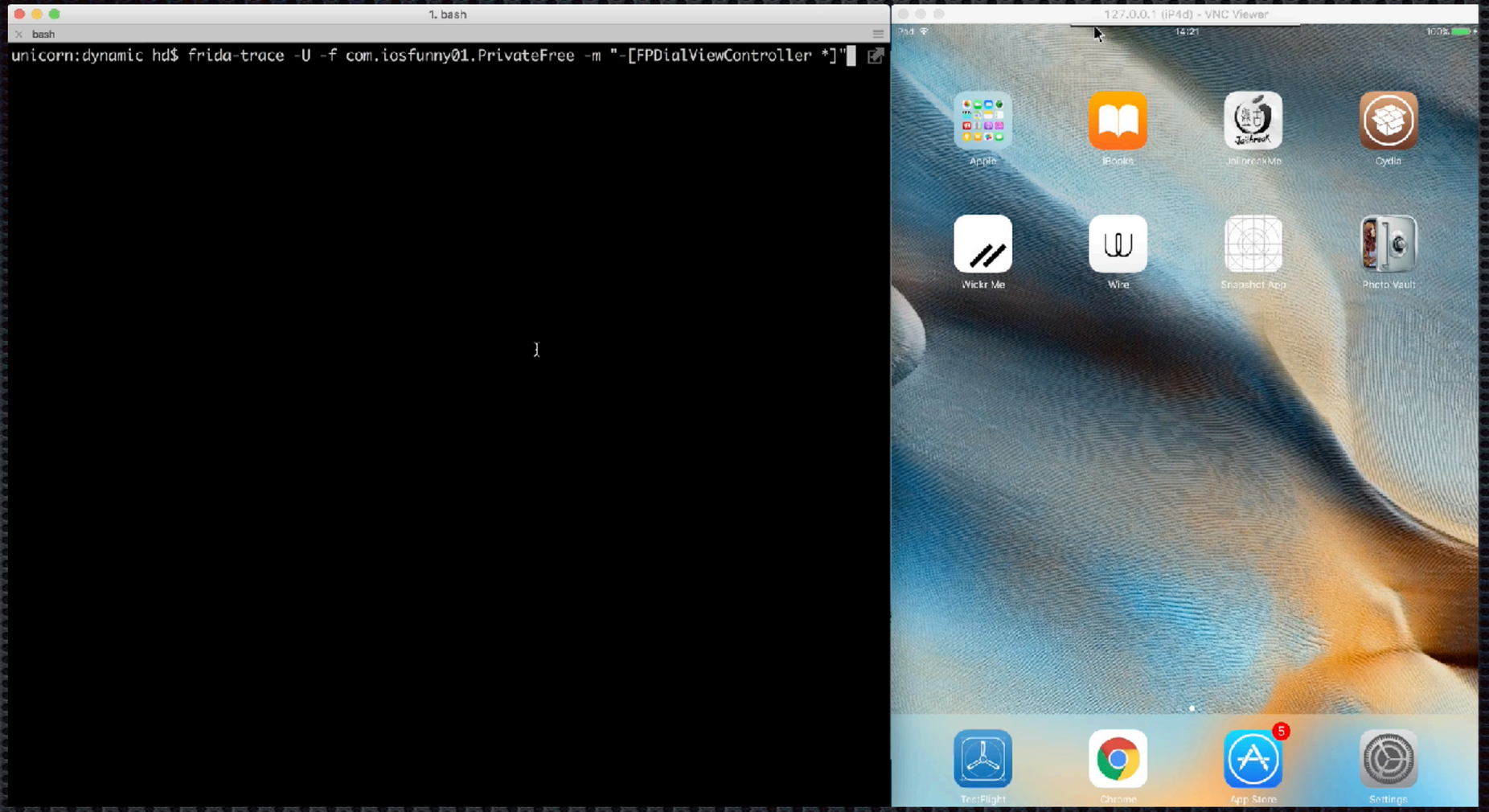
PIXELS Camp INTEGRITY

cat hookskeleton.js

```
/*
 * iOS Frida Hooking Skeleton Script
 *
 * (c) 2017 INTEGRITY S.A.
 * By: Herman Duarte <hd@integrity.pt>
 */

// lets check if the env is available
if (ObjC.available)
{
    // class or classes that we want to hook methods
    var classes = ['FPDialViewController'];
    var methodsWhiteList = ['- verifyPasscode'];

    for (var className in ObjC.classes)
    {
        if (ObjC.classes.hasOwnProperty(className))
        {
            if (classes.indexOf(className) > -1)
            {
                console.log('[*] Start: Hooking into "' + className + '" methods');

                var methods = ObjC.classes[className].$ownMethods;
                for (var i = 0; i < methods.length; i++)
                {
                    // If the method is in the whitelist then we can intercepted it
                    if (methodsWhiteList.indexOf(methods[i]) > -1)
                    {
                        try
                        {
                            var _className = "" + className;
                            var _methodName = "" + methods[i];

                            var method = ObjC.classes[_className][_methodName];

                            console.log('Hooking: ' + _methodName);

                            Interceptor.attach(method.implementation, {
                                onEnter: function (args) {
```

PIXELS Camp INTEGRITY

frida -l hookskeleton.js



PiXeLS Camp INTEGRITY

cat tls_pinning_bypass.js

```
'use strict';

var kSSLSessionOptionBreakOnServerAuth = 0;
var errSecSuccess = 0;
var noErr = 0;
var errSSLServerAuthCompleted = -9841;


function disablePinning()
{
    console.log('Disabling Pinning');
    internalDisablePinningIOS10();
    internalDisablePinningIOS9();
    console.log('... done');
}

function enablePinning()
{
    var tls_helper_create_peer_trust = Module.findExportByName('libcoretls_cfhelpers.dylib', 'tls_helper_create_peer_trust');
    if (tls_helper_create_peer_trust != null)
    {
        console.log('tls_helper_create_peer_trust PTR: ' + tls_helper_create_peer_trust);
        Interceptor.revert(tls_helper_create_peer_trust);
        console.log('tls_helper_create_peer_trust restored');
    }

    var SSLSetSessionOption = Module.findExportByName('Security', 'SSLSetSessionOption');
    if (SSLSetSessionOption != null)
    {
        console.log('SSLSetSessionOption PTR: ' + SSLSetSessionOption);
        Interceptor.revert(SSLSetSessionOption);
        console.log('SSLSetSessionOption restored');
    }

    var SSLCreateContext = Module.findExportByName('Security', 'SSLCreateContext');
    if (SSLCreateContext != null)
    {
        console.log('SSLCreateContext PTR: ' + SSLCreateContext);
        Interceptor.revert(SSLCreateContext);
        console.log('SSLCreateContext restored');
    }
```

cd “Frameworks based on Frida”

- r2frida by NowSecure
- objection by SensePost
- Needle by MWR
- SSL Logger by Google
- AppMon by @dpnishant
- …

cd “Resources & References”

- Documentation:
  - https://www.frida.re/docs/home/ especially the JavaScript API section
  - Source code - https://github.com/frida
- Help:
  - Twitter: @fridadotre, @oleavr
  - Github: https://github.com/frida/frida
  - #frida on FreeNode / fridadotree telegram group

cd “Resources & References”

- Resources:
  - JavaScript API - https://www.frida.re/docs/javascript-api/
  - Awesome Frida - https://github.com/dweinstein/awesome-frida
  - Frida Codeshare - https://codeshare.frida.re

Q&A

PiXELS Camp

INTEGRITY

Herman Duarte
@hdontwit

Thank You!

PiXeLS Camp

INTEGRITY

Herman Duarte
@hdontwit