Testing interoperability with closed-source software through scriptable diplomacy

Ole André Vadla Ravnås
Karl Trygve Kalleberg

Who are we?

Ole André Vadla Ravnås

- Author of Frida, CryptoShark, oSpy, libmimic...
- Developer, hacker and reverse engineer
- Currently working at NowSecure
- Doing R+D on mobile platforms

@oleavr

Karl Trygve Kalleberg

- Trusty sidekick
- Sporadic contributor to Frida, NixOS, Spoofax, Stratego/XT, Gentoo *(way back)* , ...
- Developer, hacker, forward engineer
- Working at KolibriFX and Sensonomic
- Doing all-round backend development

@karltk

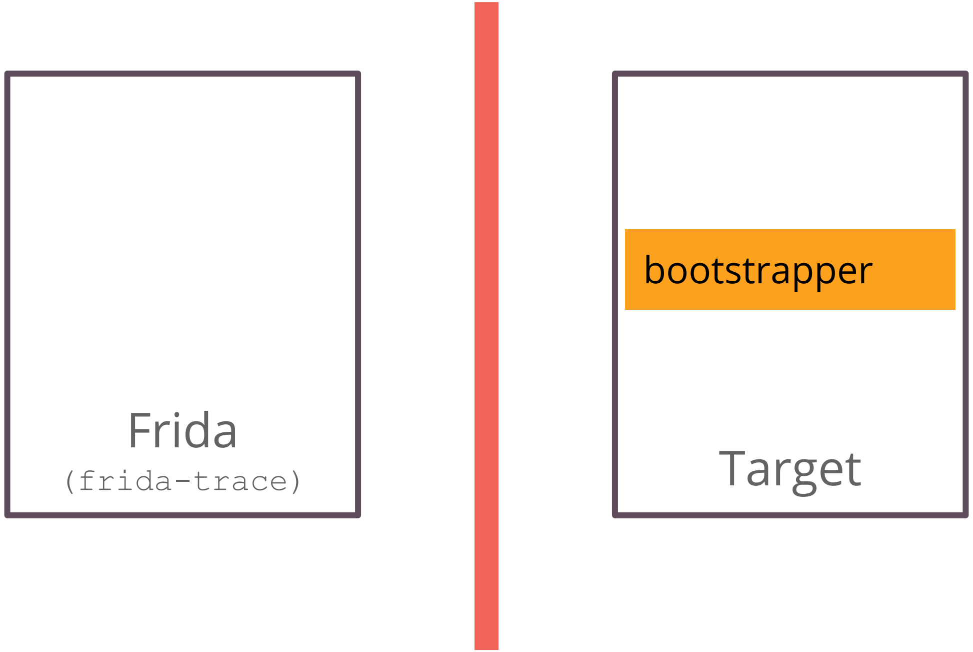
What is Frida?

- Dynamic instrumentation toolkit
  - Inspect and instrument live processes
  - Execute instrumentation scripts inside other processes
  - Scripts are
    - written in JavaScript
    - executed on a JS interpreter running inside the inspected process
- Multi-platform
  - Windows, Mac, Linux, iOS, Android, QNX
- Open-source
  - wxWindows Library Licence, Version 3.1

Demo

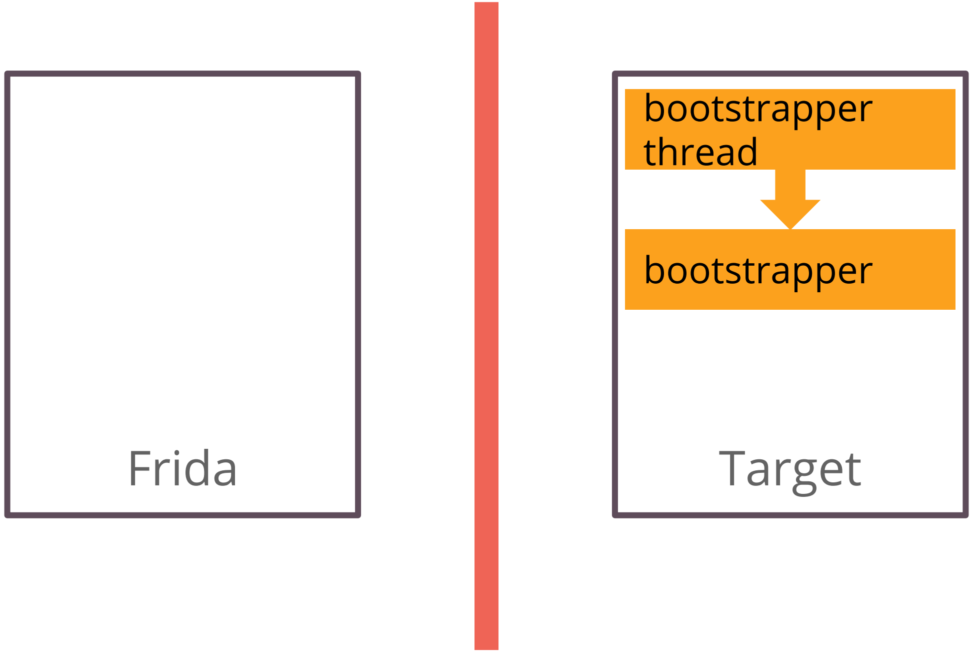
`frida-trace`

How does Frida work?



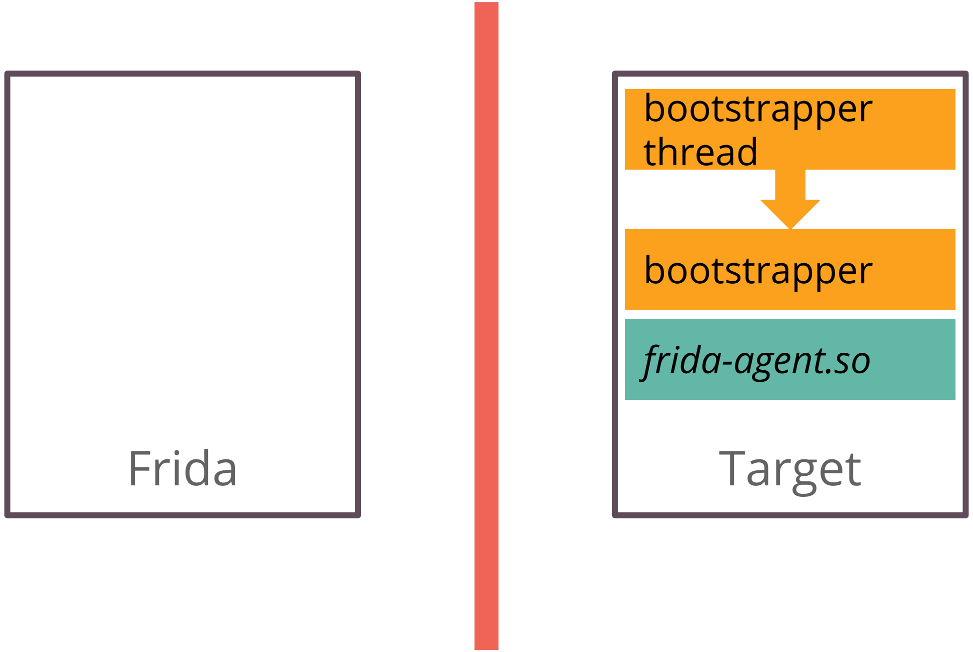
Frida process writes *bootstrapper* code into memory of *Target* process

How does Frida work?



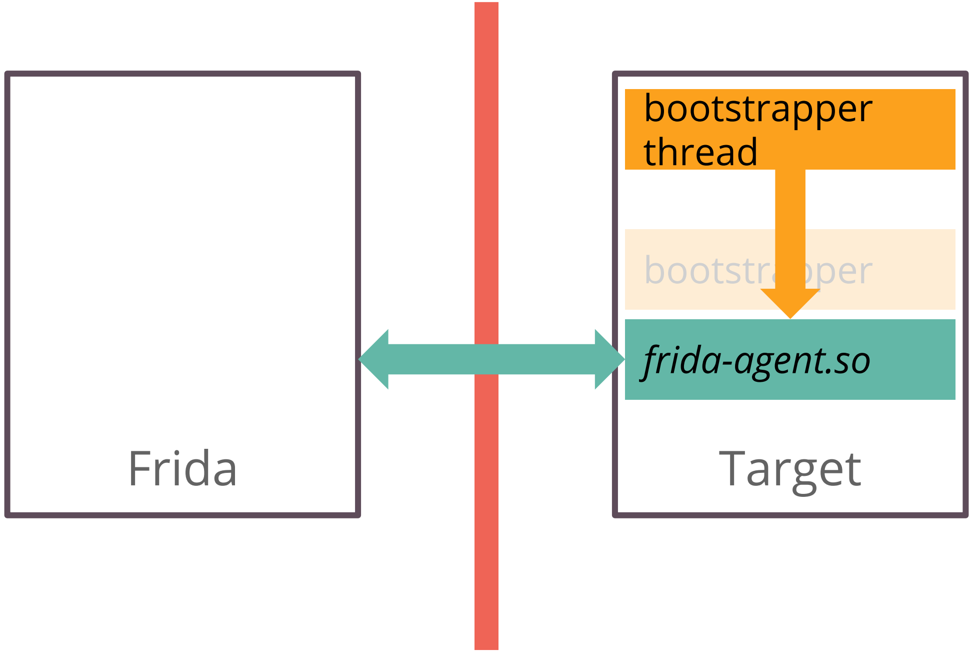
Frida hijacks an existing thread in *Target* and has it execute *bootstrapper*

How does Frida work?



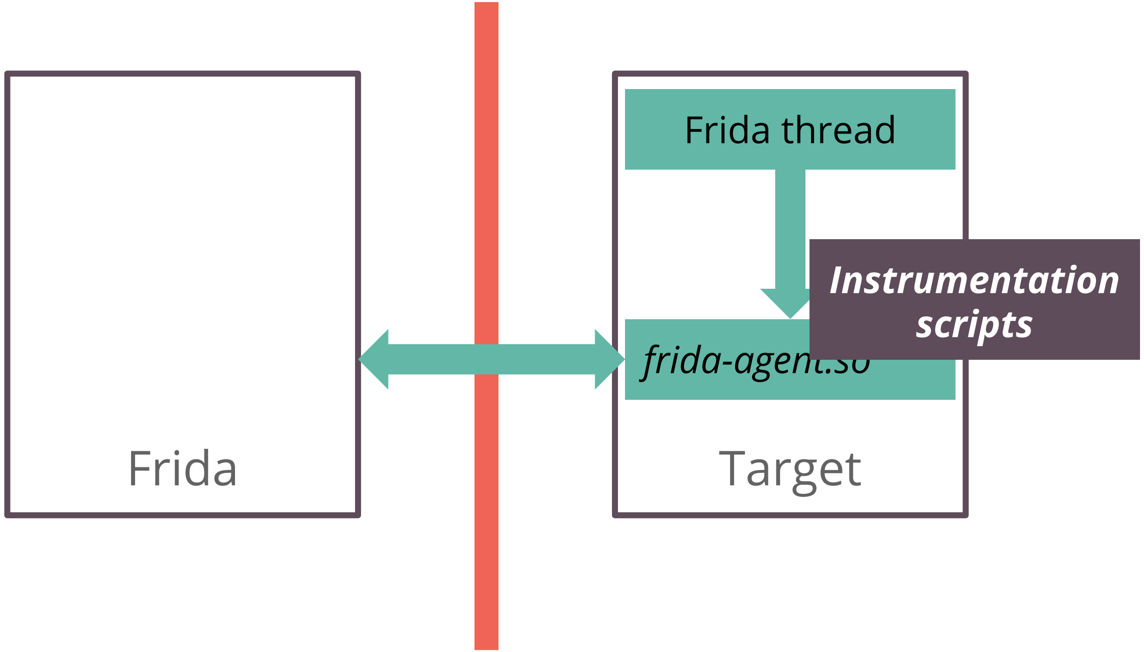
Bootstrapper loads *frida-agent.so* into *Target*'s memory space

How does Frida work?



Frida-agent.so opens a bidirectional channel between *Frida* and *Target*

How does Frida work?



Frida-agent.so sets up its own *thread*, and accepts instrumentation scripts from *Frida*

Why use Frida for testing?

- Reach internal, closed-source functionality
  - Lift logic out of closed frameworks into your tests
  - Modify behaviour of closed frameworks to improve testing
  - *Theme*: black box → grey box testing
- Caveats apply
  - Warnings as for invasive software composition, especially
    - *Brittle*: framework internals may change
    - *Time-consuming*: Reverse-engineering becomes necessary
  - Your test suite may become quite complex quite quickly

Running example: ConferenceBeats

- Open-source application for iOS
  - (Almost) available on GitHub
- Plays material from the Spotify record collection
  - When you recompile it, you can change the list - open source, yeah!
- For demo purposes only
  - Open-source application on a closed OS, dependent on closed online services + support libraries
  - (= The new world order?)

#1: Fill in Spotify login automatically

- Keyword: UI automation
- Challenges
  - On closed-source iOS
  - Login form is a web form, inside a UIWebView
  - The UIWebView is fully controlled by closed-source Spotify.Framework (*abbrev S.F*)
- Solution
  - Inject JavaScript into UIWebView with Frida

#2a: S.F must always use HTTPS

- Keyword: Property-based testing
- Challenges
    - Want to write an assertion over the stream of network calls
    - No control over calls from Spotify.Framework into CFNetwork
- Solution
    - Use Frida’s tracing features to inspect all calls to CFNetwork

#2b: S.F must use specific servers

- Keyword: Property-based testing
- Challenges
  - Want to write an assertion over the stream of network calls
  - No control over calls from Spotify.Framework into CFNetwork
- Solution
  - Use Frida’s tracing features to inspect all calls to CFNetwork

#3: Simulating flaky networks

- Keyword: Regression testing
- Challenge
  - Want to ensure 3rd party library gracefully handles flaky network
  - (Current S.F version does not)
- Solution
  - Hook network calls—simulate lost connection
  - Check for non-empty login popup

What are other applications for Frida?

- Networking
  - Emulate captive gateway
  - Apply test properties only for 3rd party libraries, based on stack trace
- Predictable data
  - Random/unpredictable data sources in framework → deterministic values
    - E.g., for camera, microphone, motion sensors
- Cross-framework workflows
  - Simulate SMS-based auth
- Resource starvation
  - Insufficient heap space
  - Insufficient disk space
  - Failure to open camera/mic
- Time
  - Simulate different passing of time
    - Faster/slower progression
    - “Reverse” (e.g., tz adjust)
  - Will my app work in 2020?
  - Is my video conference still in sync after 2 days?

Take home messages

- Frida is applicable to certain kinds of tests
  - Especially regression and integration
- Succinct test code is possible
  - … even for complicated test scenarios
- Use sparingly
  - Prefer vendor-provided testing frameworks that are maintained
- Beware the brittleness
  - Be mindful of any reverse engineering necessary

Thank you!

frida.re | #frida on freenode.net | @fridadotre