BISHOPFOX

REVERSE ENGINEERING MOBILE APPS
NEVER PAY FOR TRANSIT…AGAIN

PRIYANK NIGAM
@Rev_Octo

ABOUT ME

Senior Security Engineer

- 4+ Exp with AppSec/MobileSec/NetworkSec/RE/IoT etc.

Motivations

- Appreciate app-specific attack vectors and manual RE
- If you learn stuff from this, appreciate the mobile use cases, and giggle how ridiculous this is.

ROADMAP

01 DEMO

02 INTRO TO MOBILE SECURITY

03 MOBILE APP STATIC ANALYSIS

04 RUNTIME INSTRUMENTATION

05 SERVER-SIDE VULNS

06 VENDOR DISCLOSURES

07 MITIGATIONS/CONCLUSION

01

02

INTRO TO MOBILE SECURITY

INTRO TO MOBILE SECURITY

Why mobile === thick client

- Typically provide rich functionality independent of the central server
- Still requires periodic connection to the network

Crucial Decisions

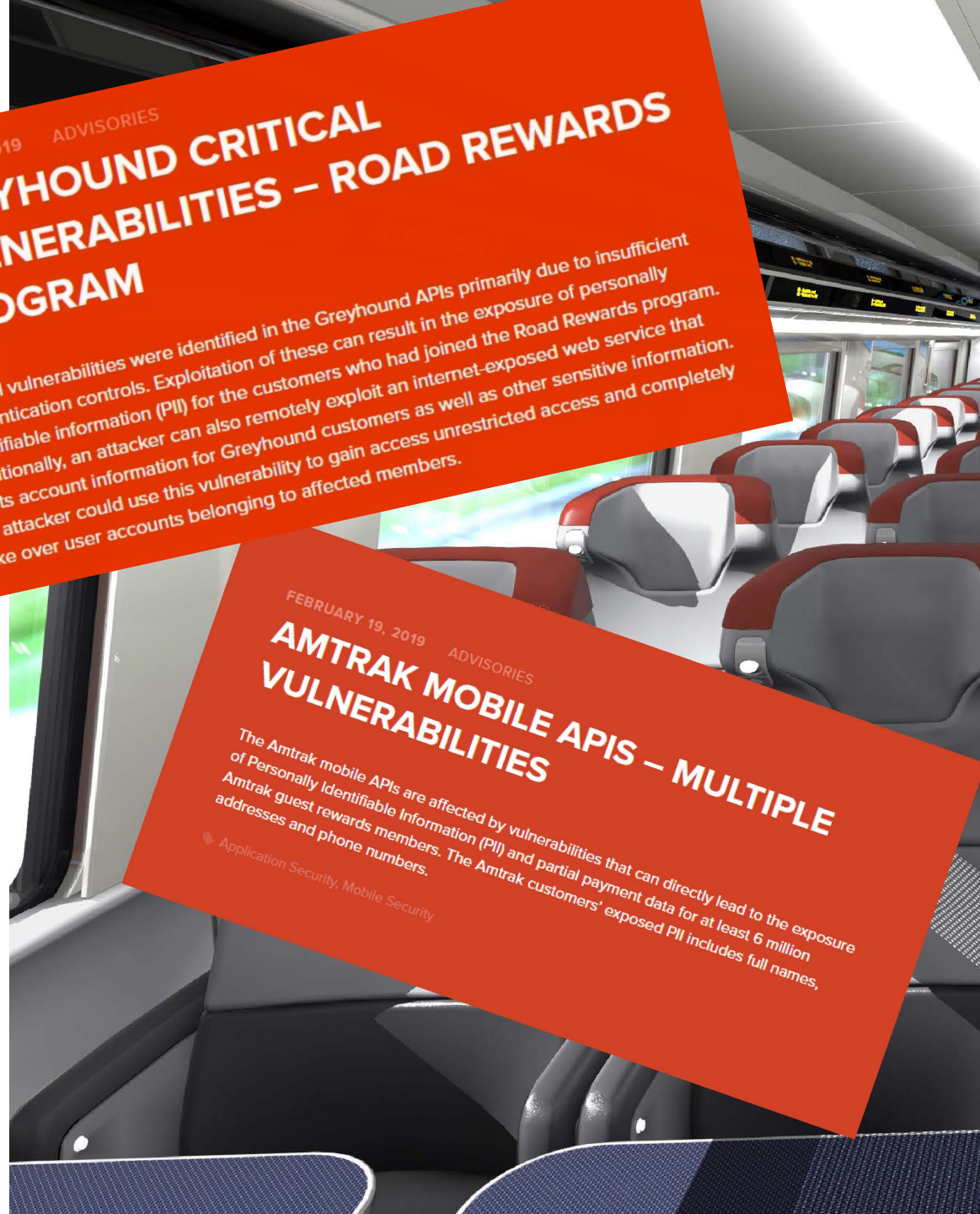
- Which tasks executes on the server and which on the client
- Fail safe or fail open when network is non-existent

02

TARGETS

- City Mass Transit Apps
  - $CITY Transit Authorities
- Inter-city Transit Apps
  - Amtrak
  - Greyhound

- Any platform which handles mobile ticketing



03
STATIC ANALYSIS

STATIC ANALYSIS

- Analyze the app package
  - Unzip the apk/decrypt and dump IPA
    - https://github.com/Alonemonkey/frida-ios-dump
  - Dex2jar/JD-GUI – Decompile Java classes
  - ClassDump - Examine ObjC runtime info from mach-O file
  - Hopper/IDA/Ghidra - Disassemble, decompile, debug

04
RUNTIME INSTRUMENTATION

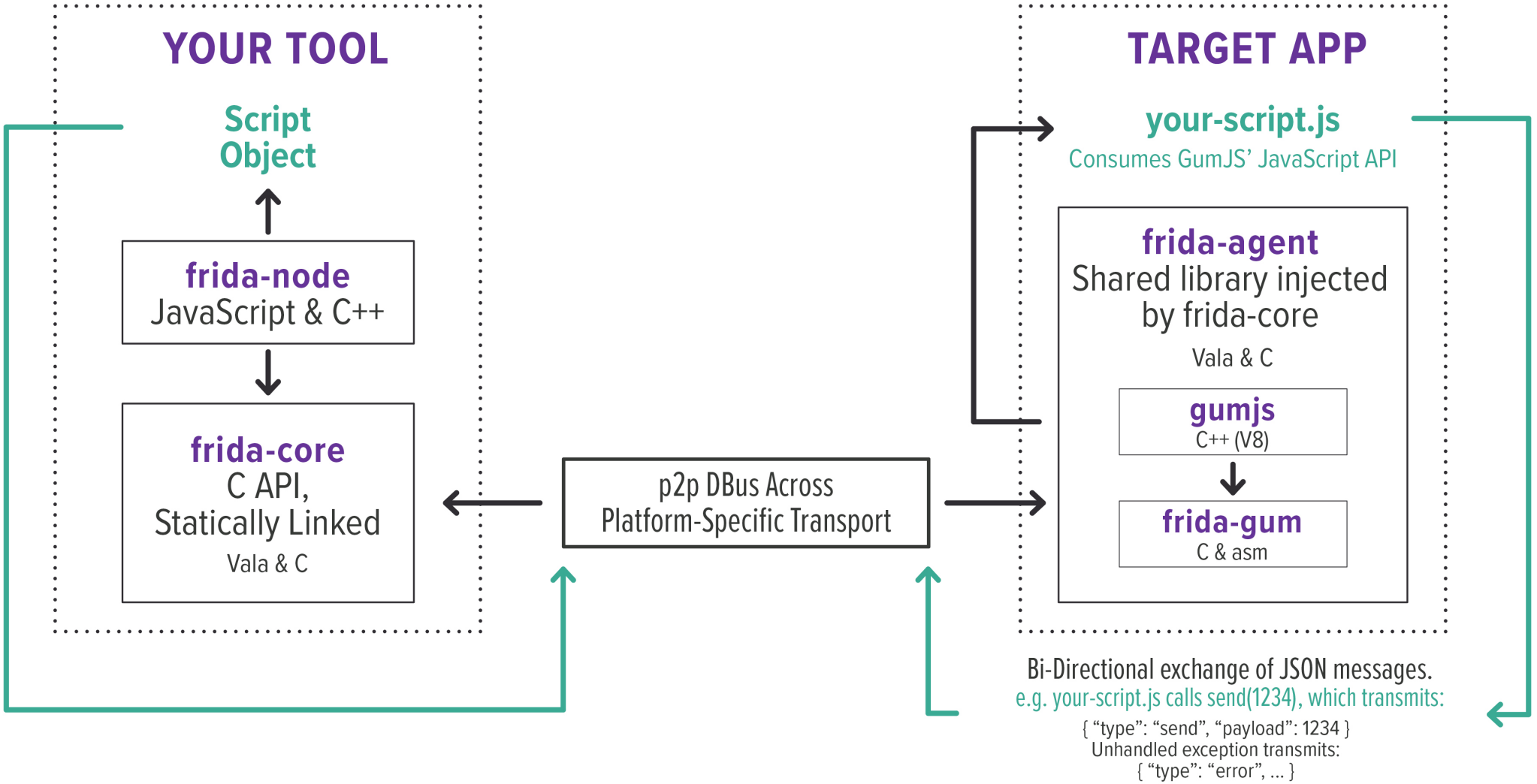
04 RUNTIME INSTRUMENTATION

DYNAMIC ANALYSIS

- Load the app, Observe Network Traffic
- Inspect Device Storage
- Install Frida
  - Configure virtualenv
  - pip install Frida
  - pip install Frida-tools

FRIDA
BASIC OPERATIONS

FRIDA - ARCHITECTURE



FRIDA – BASIC SCRIPT

```
var res = new ApiResolver(type); //create a new resolver of the given type
 var matches = res.enumerateMatchesSync(pattern); // perform resolver specific
”query” string returns name and address as a NativePointer



Interceptor.attach(matches.address, {
onEnter: function(args) {
       // print/change the args
       },
onLeave: function(retval){
       // Do stuff with retval
}
```

Awesome Open Source Scripts: https://github.com/0xdea/frida-scripts/ & https://codeshare.frida.re/

FRIDA – CLASS/METHOD ENUMERATION

Encryption method names –

BYPASS ENCRYPTION IN STORAGE/TRANSIT

- Method hooking and parameter/retval enumeration
- Alter/Nullify primitive types
- Alter non-primitive types:
  - `$className`: string containing the class name of this object

DECRYPT ON-THE-FLY

```
Interceptor.attach(aesEnc.implementation, {
        onEnter: function(args) {
            console.error("Hooking implementation .. " );
            var message = ObjC.Object(args[3]);
            console.error("Original arg[3] type : " + ObjC.Object(args[3]).$className );

            var argObj=JSON.parse(ObjC.Object(args[3]).toString().replace("{","[").replace("}","]"));
            var str = ''
            for (var i = 0; i < argObj.length; i++) {
                try{
                   str += String.fromCharCode(Math.abs(argObj[i])) || Math.abs(argObj[i]).toString()
                    }
                catch(error){ console.log('hmm some issue')  }
                }
              var str_json = JSON.parse(str);
              console.warn("Parsed JSON " + JSON.stringify(str_json));
```

TAMPER THE RET VALUES

ObjC.classes: an object mapping class names to ObjC.Object JavaScript bindings for each of the currently registered classes. You can interact with objects by using dot notation and replacing colons with underscores, i.e.:

```
[NSString stringWithString:@"Hello World"]
```

is now

```
var NSString = ObjC.classes.NSString; NSString.stringWithString_("Hello
World");
```

*Note the underscore after the method name

TAMPER THE RET VALUES

```
var ptrMsg = Memory.alloc(inputByte.length)

Memory.writeByteArray(ptrMsg, inputByte)

var newret =
ObjC.classes.MAPrimitiveByteArray.alloc().initWithBytes_length_(ptrMsg,input
Byte.length);
```

BYPASS “ENCRYPTED” SQLITE DB

- The sensitive info (ticket objects, etc) are in encrypted DB.
- Where is the key ?
- Hook `[ENC_EncPLSqliteDatabase initWithPathandPassword: ]`
- Alternatively dump the keychain or wherever it is stored
- Tamper the database, see if the app enforces any integrity checks?

BYPASS CUSTOM MOBILE SDK-BASED ENCRYPTION

```
trace("*[XXPassSDK passes*]");
trace("*[XXPass objectFromJSON*]");

while ((key = enumerator.nextObject()) !== null) {
                    var value = dict.objectForKey_(key);
                    count++;
                    if (count === 1){
                    var arr1 = ObjC.classes.NSMutableArray.arrayWithObject_(key);
                    var arr2 = ObjC.classes.NSMutableArray.arrayWithObject_(value);
                                                        }
[…]
                              if(key == "expiration"){
                                                  value = "2019-07-15T07:43:01.000Z"
                                                  console.error("Expiration: " + value);
                              }
```

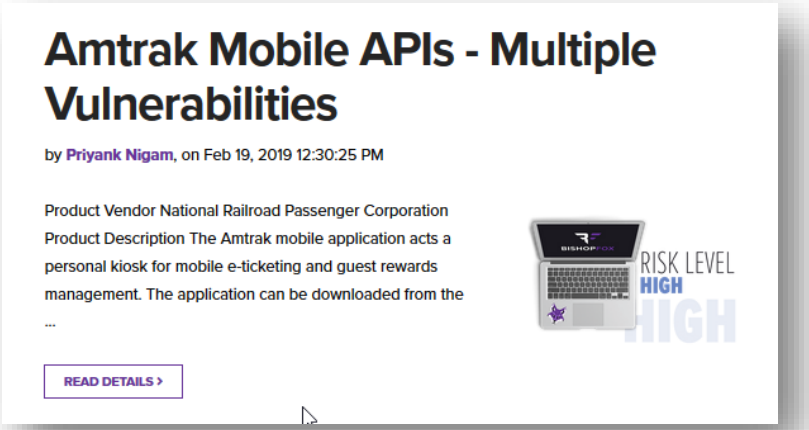
05
SERVER SIDE ISSUES

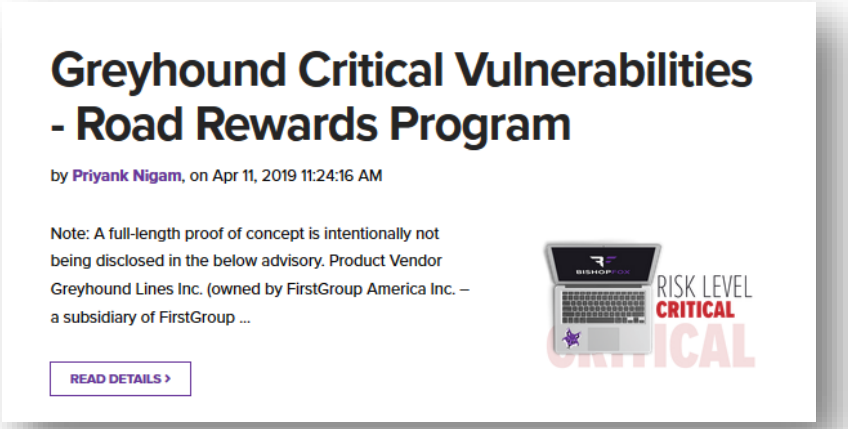
SERVER-SIDE VULNS

- After the parameters can be decrypted and re-encrypted, common server side issues can be uncovered – AuthN, AuthZ etc.

- Public Disclosures:
  - Amtrak Authentication Bypass/Account Takeover (https://www.bishopfox.com/news/2019/02/amtrak-mobile-apis-multiple-vulnerabilities/)

  - Greyhound Account Takeover (https://know.bishopfox.com/advisories/news/2019/04/greyhound-critical-vulnerabilities-road-rewards-program)





06
VENDOR DISCLOSURES

VENDOR RESPONSES

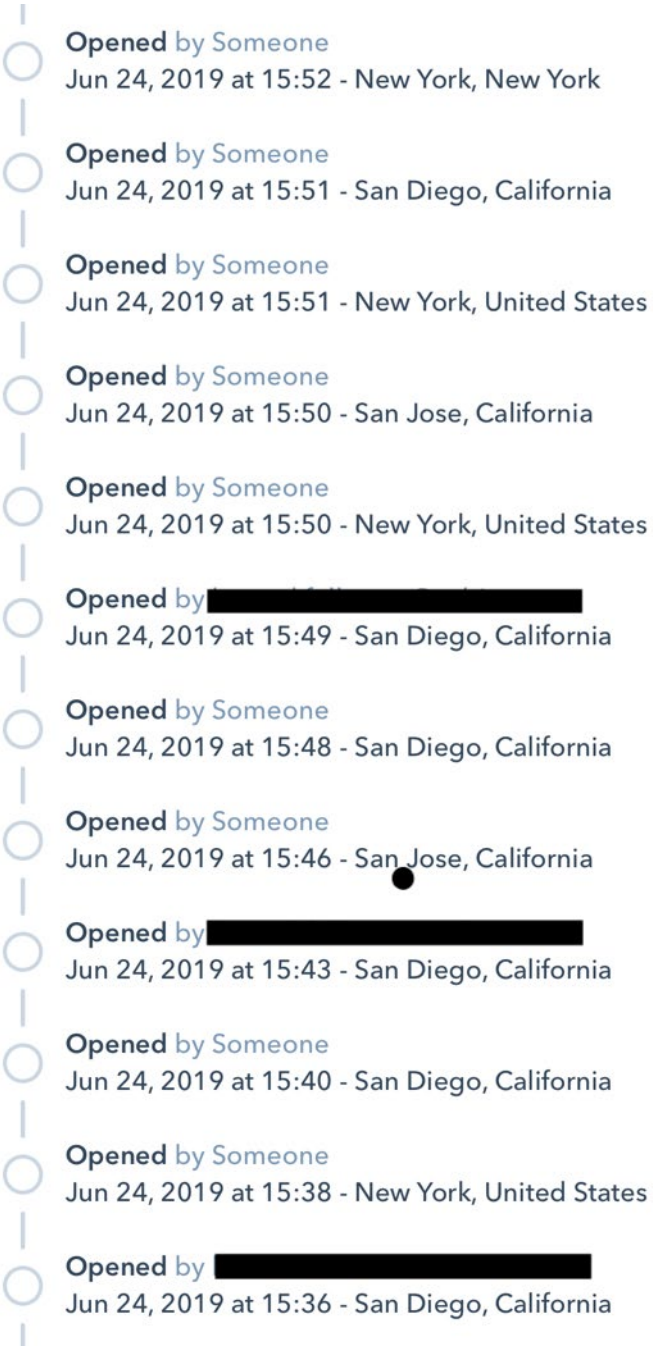
Hello Priyank,
Thank you for this information. Very interesting indeed.
Would this vulnerability be possible on a device that was not jailbroken?

Thank you,

- No, The answer is runtime integrity checks.
- Silence

VENDOR RESPONSES

These guys engaged at first, acknowledged the vulns, and then..



VENDOR RESPONSES

Yes, what on earth has happened here?

Who am I joining a call with and why?

Kind regards

[redacted]

[redacted]

Chief Information Security Officer

FirstGroup Plc

Actually the call did happen ~8 months ago, and then the void...

VENDOR RESPONSES

“Thank you for bringing this to our attention, we appreciate your professionalism in the way you raise and handle this type of concern.”

- The bottom-line is…

“With all mobile ticketing, from all suppliers, there are known trade-offs between security, infrastructure investment (in hardware validation) and passenger convenience, which must be compared to known risks in other forms of transit fare collection (e.g. metal tokens, paper tickets or smartcards) and that is a discussion we have with all transit agencies we work with.”

07
MITIGATIONS

MITIGATIONS

- Rethink mobile use-cases
- Enforce strong server-side protections
- Strong runtime integrity checks
  - Move sensitive logic to native layer
  - Strip the binary
  - Check # of dynamic libraries loaded at runtime
- Sensitive Data storage on device

THANK YOU
FOR YOUR TIME
ANY QUESTIONS

pnigam@bishopfox.com

@Rev_Octo

Interested in joining the Foxes? https://www.bishopfox.com/