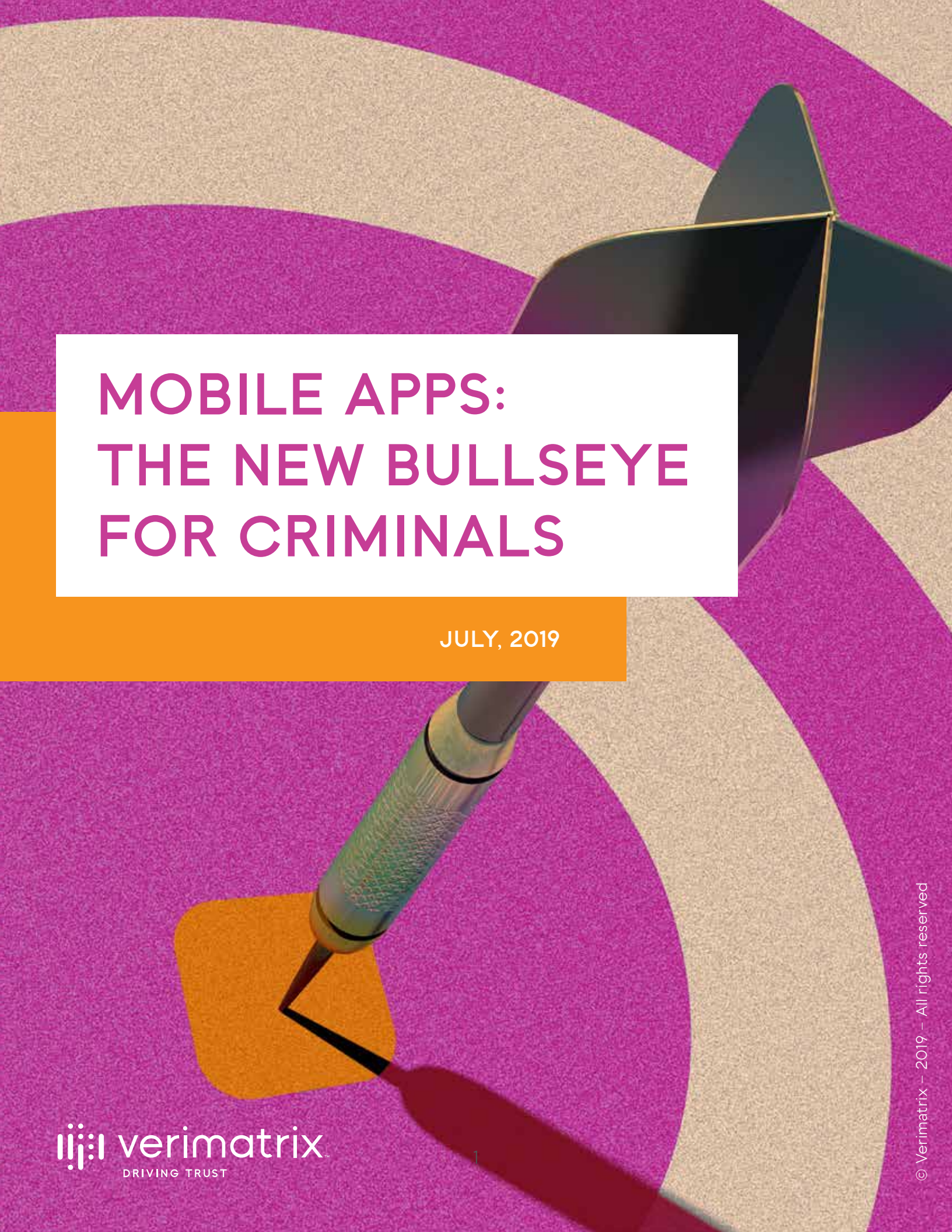


MOBILE APPS: THE NEW BULLSEYE FOR CRIMINALS

JULY, 2019

verimatrix
DRIVING TRUST

TABLE OF CONTENTS

IT'S A DANGEROUS WORLD

Once a mobile app leaves the relatively safe confines of the development environment, it is released into the wild. Most creatures your app meets will be friendly and just want to play. But it will also likely encounter some ferocious animals that want to tear it apart. These animals are not dinosaurs, but highly evolved and intelligent cybercriminals.

It is increasingly evident that cybercriminals are targeting mobile apps. They know that mobile apps are soft targets and understand that all apps have a "guaranteed vulnerability." In other words, no code is perfect.

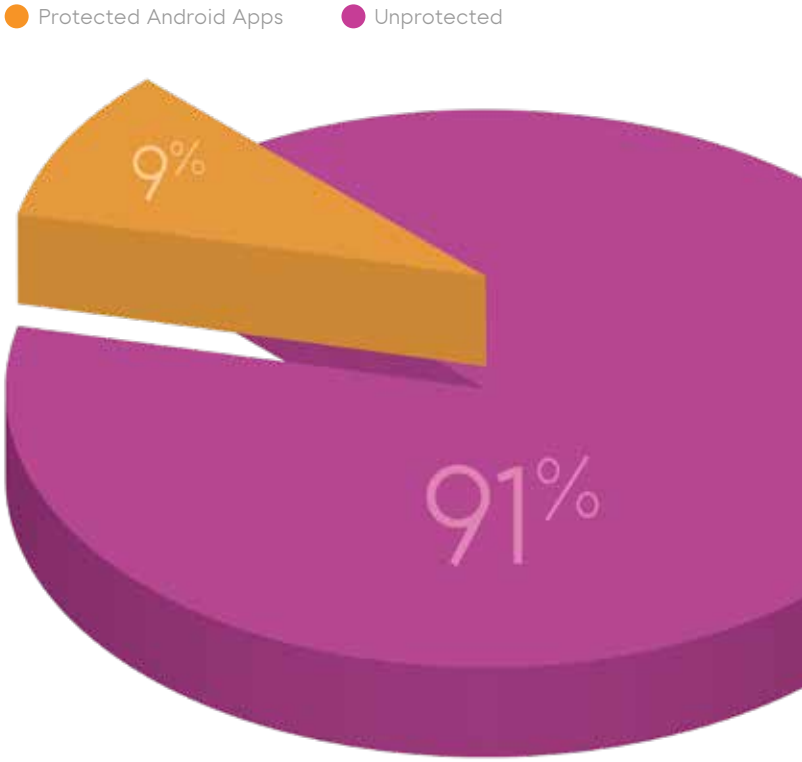
The generally accepted industry norm is for there to be between 15 and 50 errors per 1,000 lines of delivered code[1]. That is the doorway criminals need – and it is what the research teams at Verimatrix are seeing exploited on an ever-increasing basis in 2019.

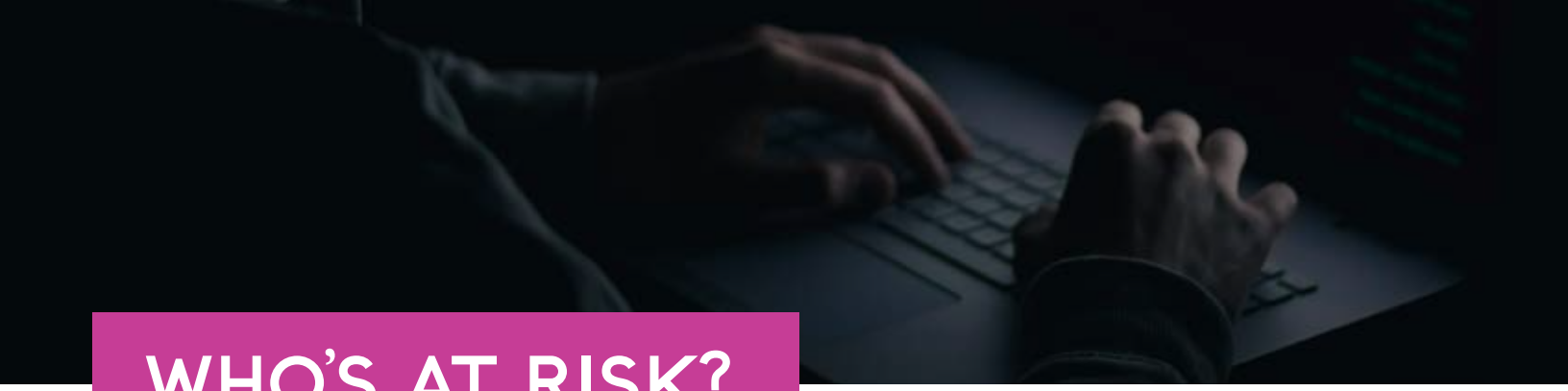
Focusing on mobile, the complexity and number of apps increases on a daily basis. That means an increasing number of vulnerabilities ready for exploitation.

[1] https://labs.sogeti.com/how-many-defects-are-too-many/

Verimatrix recommends that app developers take a pragmatic approach to protecting their apps by using code protection technologies.

Unfortunately, Verimatrix's own research shows that only 9% of Android apps use such protection... leaving the remainder nearly defenseless against cybercriminals.

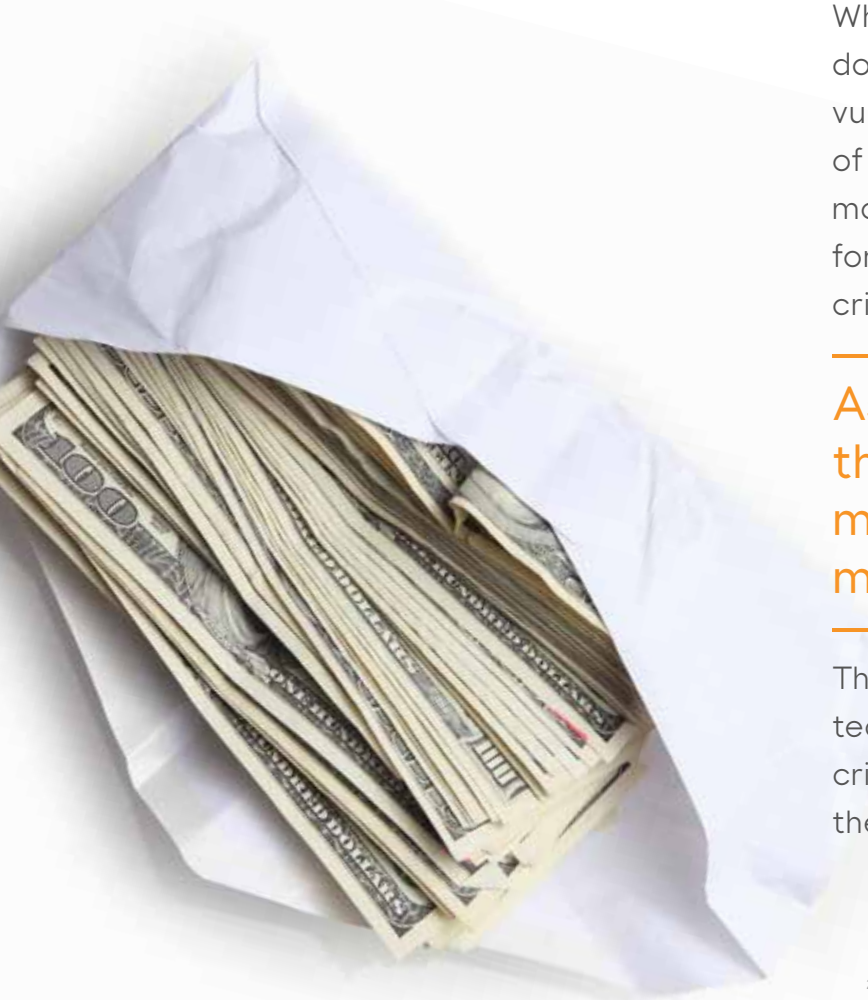




WHO'S AT RISK?

If your app is valuable to you and your customers, it is safe to assume it is also valuable to cybercriminals.

They either steal data that enables them to commit fraud by selling it, or alternatively, they lift intellectual property (IP) or extort via a form of ransomware.



Unfortunately, many organizations pay the ransom. That does not help the cause, despite the short-term convenience and cost-savings that often accompany the decision to simply give in and pay. Long term, it would be better if more organizations followed Radiohead's example[2] and stood up to the bullies.

While every app is at risk, it is those apps that do not take mitigation steps that are most vulnerable. Most attackers, with the exception of state-sponsored hackers, want to make money. As such, the attackers are looking for a return on their investment – it may be crime, but it is still business.

As with any business decision, the criminals are looking to maximize their return while minimizing their investment.

Through the use of code protection technology, the app is protected, the criminal's investment is greatly increased, and the risk mitigated.

[2] https://www.bbc.co.uk/news/entertainment-arts-48597096

Every app is at risk.

Whether it controls a connected device, a cloud service, or online banking system, any type of app is vulnerable.

THERE ARE TWO TYPES OF APPS IN TODAY'S MARKET:

Apps that Serve as Gateways to a Service

These are apps that serve as gateways to a service that is in the cloud. The app allows users to work with the service (banking, airlines, Facebook, WhatsApp, Tinder, etc.) They all essentially serve as a user interface for that bigger service.

Apps That Conduct Computational Functions

These are apps that are doing the computational functioning inside your mobile phone (gaming and medical apps, etc.) Using medical apps as an example, they take information from another device and the data manipulation is done on the actual phone. For instance, the latest insulin pumps simply provide data to the phone and let it process the data.

WHAT'S AT RISK?

It's common to start any risk mitigation discussion by identifying what exactly is at risk. It's a good question, but it's better to ask developers what is valuable to them and the app's users.

WHAT'S AT RISK?

TRUST

Your mobile app represents you to your customers. It's how your customers interact with your products and services. If customers cannot trust that interaction point, they will not trust your business.

If you are Air Canada, you sell tickets, not an app. The app is just an interface. If you are Mercedes, you make your money selling cars, not an app.

This gateway that you built to empower your business can also severely impact it. Once trust is lost, you sell less tickets, less cars - the financial consequences can be massive.

The mobile app sits in an exclusive and privileged position: it has access to your customer and to your service. There could be no more strategic point in the business process where criminals can hunker down. CRIMINALS KNOW IT. Going through the app is not only easier for the criminals, but it also serves as a doorway to the assets.

WHAT'S AT RISK?

INTELLECTUAL PROPERTY

Many apps contain functional components, such as games, medical apps or movie players. Anything that is designed to work offline has to have the processing contained in the app. As a developer, you will have put time and effort into developing the algorithms for that processing.

The last thing you want is for someone to take that hard work and use it for their own devices without your permission.

If you spend years developing your game and you sell it for $20.00 and then someone steals that gaming app and sells it for $1.00, you've lost a huge chunk of revenue.

WHAT'S AT RISK?

REVENUE STREAM

Many apps provide an ongoing revenue stream to their developers. It could be through advertisements or in-app purchases.

It is common for attackers to "repackage" an app, particularly video apps, to remove advertisements. The motivation is not financial in this case, as it aims to allow users to watch videos without being interrupted. For the developers; when the advertisements go, so does their revenue stream.

Equally, consider the video game market. As games move to free-to-play, the revenue protection battle morphs. The biggest threat is no longer piracy. Instead, it becomes all about cheating.

Once players work out how to cheat a game, the fun evaporates for the player and they decide to leave. With their departures, revenue streams leave as well.

Would you have considered the concept of fair play being a valuable software asset?

A REAL DANGER:

EXAMPLES IN 2019

7-Eleven

7-Eleven has found itself at the center of lots of negative publicity recently[3]. By failing to implement strong authentication, it was easy for attackers to steal $500,000 from users of its mobile payment app in Japan. It resulted in 7-Eleven having to shut down the services only days after it launched.

For any app developer, it is important to secure the login process and the code that manages it.

That helps prevent passwords from leaking and also stops attackers from launching an attack against the app or the process.

There is a second lesson to be learnt from the 7-Eleven example. When asked about the attack, the CEO give the impression that his organization did not know what good practice should have been followed. This created a second wave of bad publicity.

[3] https://www.theverge.com/2019/7/6/20684386/7-eleven-japan-shut-mobile-payments-app-7pay-security-flaw-cybersecurity

A REAL DANGER:

EXAMPLES IN 2019

WhatsApp

It was recently revealed that WhatsApp had been exploited to infect phones with spyware[4]. The attackers discovered a bug in the software that allows them to initiate a buffer overflow attack. This allows code to be injected into the app without the users knowledge.

In the WhatsApp case, the code was hidden in a series of specially crafted voice chat data packs.

WhatsApp even had a comprehensive bounty program and hackers still found their way in! The lesson in that case is to take steps to make it difficult for the attacker to find the bugs that will inevitably be in the app.

[4] https://arstechnica.com/information-technology/2019/05/whatsapp-vulnerability-exploited-to-infect-phones-with-israeli-spyware/

ARMING THE CYBERCRIMINAL

Cybercriminals are well resourced. This allows them to hire the brightest minds and give them the best tools. These means they are well equipped to exploit the guaranteed vulnerabilities found in most apps.

An attacker will go through several different stages to uncover the secrets they are looking for:

Static analysis
When the attacker will "read" the application code. Many freely available tools help the attacker crack open an app and then convert the "machine" code into something closer to the original code.

Dynamic analysis
Through using an open device, the attacker is able to garner a deeper knowledge of the app by observing it executing. They are able to set up known states and to force the execution in such a way as to deepen their knowledge.

Fuzzing
When an attacker automates an investigation. Taking the learning from static and dynamic analysis, an attack will try many different combinations of inputs against the various components that make up an app. The attacker is looking for combinations of inputs that either reveal data or a bug in the software.

Side channel attacks
By observing how an app interacts with the rest of the mobile environment, it is possible to glean insights into the internals of the app. The attacker will collect vast quantities of data from the apps interactions and then use powerful mathematics to analyze it.

Compromised Devices

Android and iOS build in many security defenses into the operation systems. These are designed to isolate each application, stopping another app from attacking it. These defenses do a decent job, but they only provide protection from a subset of attacks.

Jailbreaking (or "rooting," to use the Android terminology) is when a device's defenses are deliberately broken down. This may be benign – many tech savvy users jailbreak their phones to allow them more control over their devices. It could also be malicious – an attacker looking to gain more access to an executing app.

Attackers will also compromise devices in more intrusive ways by using hooking frameworks (such as Frida[5]), debuggers and emulators to gain a deeper insight into an application.

[5] https://www.frida.re

What is Frida?

After compromising the device, one of the first tools most attackers will reach for is Frida.

Frida is a dynamic instrumentation toolkit for developers, reverse-engineers, and security researchers.

It is a popular tool with attackers going after Android and iOS mobile apps. Allowing them to analyze running processes. It uses a host of different methods, including code injection and module loading. It allows the attacker to trace the application, viewing real-time function calls; and it allows the attacker to hook function call, changing the execution and behavior of the application in real-time.

FRIDA

Quick-start Instructions

```
~ $ pip install frida-tools
~ $ frida-trace -i "recv*" Twitter
recvfrom: Auto-generated handler: …/recvfrom.js
Started tracing 21 functions.
1442 ms     recvfrom()
# Live-edit recvfrom.js and watch the magic!
5374 ms     recvfrom(socket=67, buffer=0x252a618
length=65536, flags=0, address=0xb0420bd8,
address_len=16)
```

Reverse Engineering

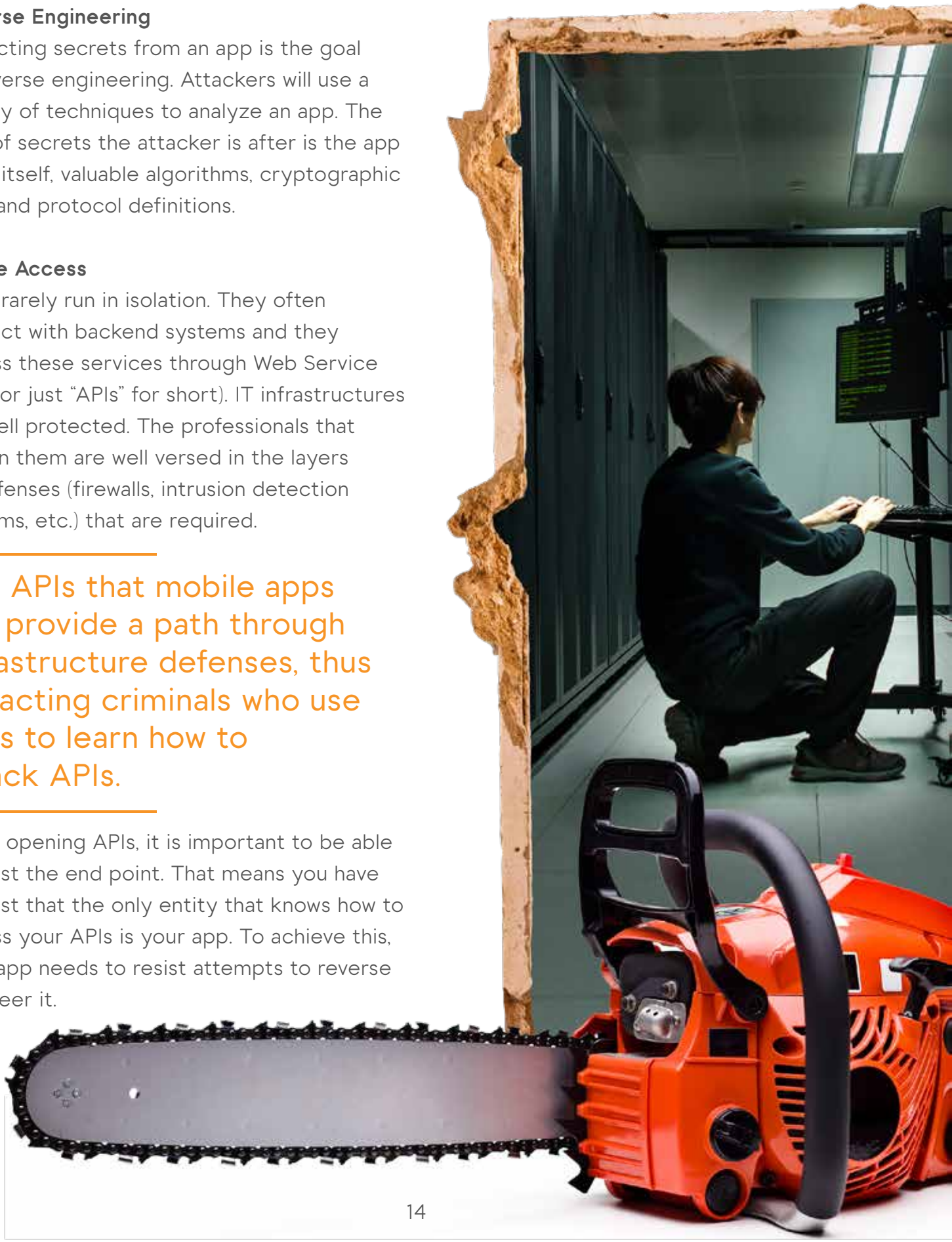
Extracting secrets from an app is the goal of reverse engineering. Attackers will use a variety of techniques to analyze an app. The sort of secrets the attacker is after is the app code itself, valuable algorithms, cryptographic keys and protocol definitions.

Rogue Access

Apps rarely run in isolation. They often interact with backend systems and they access these services through Web Service APIs (or just "APIs" for short). IT infrastructures are well protected. The professionals that design them are well versed in the layers of defenses (firewalls, intrusion detection systems, etc.) that are required.

The APIs that mobile apps use provide a path through infrastructure defenses, thus attracting criminals who use apps to learn how to attack APIs.

When opening APIs, it is important to be able to trust the end point. That means you have to trust that the only entity that knows how to access your APIs is your app. To achieve this, your app needs to resist attempts to reverse engineer it.



TAMING THE LION

Protecting yourself from these intelligent animals involves taking a pragmatic approach. With millions of lines of code in an app, there are going to be bugs. It is simply not going to be possible to remove every single bug and vulnerability. Even if you could, the app would still be open to other attack vectors.

The practical solution is to harden the app. Code should be as solid as it can be. The way to solidify your code is to use code protection technologies (also known as app shielding). This locks downs attack vectors, makes it difficult to identify exploitable bugs, and even if they can find the bugs, makes it difficult, at best, to understand how to exploit them.

With ProtectMyApp, Verimatrix brings much-needed app protection to the masses.

Verimatrix is the first in the industry to offer cost-effective yet time-tested mobile app security to any developer – removing the financial barriers and need for dedicated security expertise that plague today's app developers in both large and small organizations. It is a "send it and secure it" subscription service that revolutionizes app security just as criminals are starting to see their full potential.

ProtectMyApp
POWERED BY VERIMATRIX

PROTECT

Uses automated protection that does not require manually setting protection points. Intelligent optimization provides the best balance between security and performance while unparalleled self-protection capabilities are independent of the device/operating system security level.

DETECT

Combines client security with powerful cloud intelligence to detect disasters before they occur. App protection data is sent to ProtectMyApp that uses powerful analytics technology to provide any needed security alerts to developers.

ACT

Arms developers with the ability to terminate app execution when an imminent risk is detected. Developers can also easily patch vulnerabilities before they become exploits, and damage is done; deny access to service for apps running on emulators or debuggers; and block cloned or repackaged apps access to the service.

ProtectMyApp is there when it is needed most – when hackers have your app in their sights. Built on a technology platform proven in the wild for more than 10 years, but re-envisioned to provide easy one-step protection.

For more information, visit ProtectMyApp.com today.