PEARSON IT CYBERSECURITY CURRICULUM

SECOND EDITION

A PRACTICAL GUIDE TO DIGITAL FORENSICS INVESTIGATIONS

DR. DARREN R. HAYES

Table of Contents

Chapter 10

Mobile App Investigations

Learning Outcomes

After reading this chapter, you will be able to understand the following:

- The importance of mobile apps in investigations;
- How to perform a static and dynamic analysis;
- The digital evidence available from dating, rideshare, and other popular apps;
- The value of deep-linking in investigations; and
- Analyzing SQLite databases.

Mobile applications (apps) are extremely important today in investigations for a variety of reasons. Interestingly, the databases associated with many apps, are unencrypted and are not too difficult to analyze. Furthermore, if a mobile device is locked or inaccessible, there are many other options available, which may include analyzing a linked desktop version of the app or sending a subpoena, or court order, to a third-party provider to obtain a suspect's data. Third-party companies collect, and store, a tremendous amount of data on their customers. Finally, many users opt to back up their data to cloud storage. For example, WhatsApp has the option for Apple iPhone/iPad users to back up their chats to iCloud, and that backup can be requested from Apple. Nevertheless, organized criminals and terrorist groups largely use mobile apps that utilize strong encryption or proprietary encryption, which can seriously hamper the work of law enforcement. Compounding these concerns is the fact that many apps maintain their servers in countries like Russia, which is beyond the reach of law enforcement in the West. Popular communication apps that use strong encryption include Telegram, Signal, Wickr, and Threema to name but a few. Nevertheless, zero-day exploits are frequently found in mobile apps, including Telegram, which can help investigators to gain access to an encrypted app. A **zero-day exploit** is a security vulnerability that is a threat on the day that it is discovered because a software patch, to fix the exploit, does not yet exist.

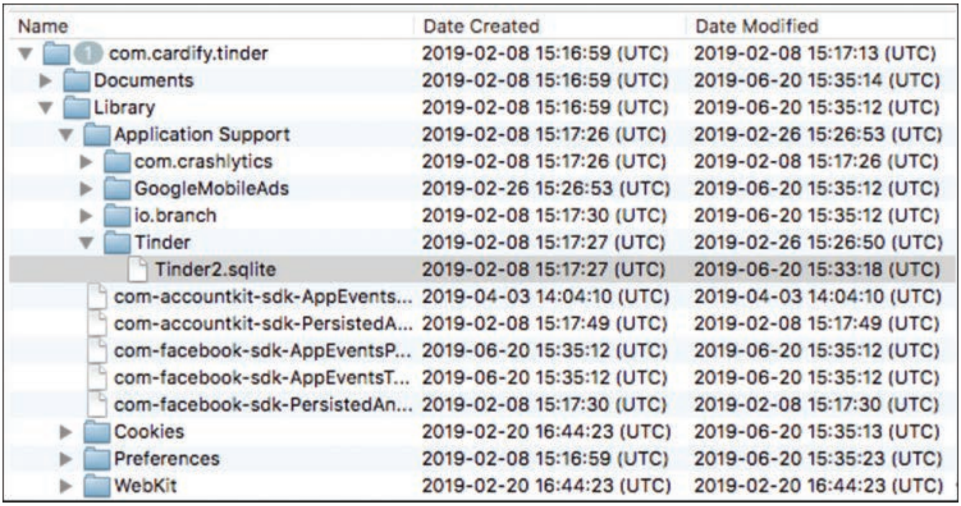
Static Versus Dynamic Analysis

During app installation, typically a SQLite database will be installed on the user device. This is a relational database that is comprised of tables. The data stored in these tables may or may not be encrypted. A table may contain a user's contacts, while a related table may store communications with contacts, for example. It is important to understand that these databases contain an extraordinary amount of personal information and, when unencrypted, can put an individual at risk for social engineering. Additionally, we should always consider the possibility to subpoena a third-party service provider for evidence.

When analyzing mobile apps, there are several approaches that an investigator can take, in order to examine the user data. A static analysis includes an examination of the SQLite database associated with that app. A dynamic analysis of the app is an analysis of the behavior of the application once it has been executed (or run). The sections that follow examine static analysis and dynamic analysis in more detail.

Static Analysis

A SQLite database is a relational database that is the preferred storage for data associated with mobile apps. SQLite is a C-language library that is responsible for the SQL database. SQLite source code is source code that resides in the public domain. Forensic tools, like BlackLight, enable the user to easily browse through application SQLite databases but there are other standalone tools that can be used. One of these tools is SQLite Database Browser, which is freeware. Later in this chapter we shall detail the types of evidence available from a number of popular mobile apps. Figure 10.1 shows an example of a SQLite database for the Tinder app on an iPhone.



Name	Date Created	Date Modified
▼ 1 com.cardify.tinder	2019-02-08 15:16:59 (UTC)	2019-02-08 15:17:13 (UTC)
▶ Documents	2019-02-08 15:16:59 (UTC)	2019-06-20 15:35:14 (UTC)
▼ Library	2019-02-08 15:16:59 (UTC)	2019-06-20 15:35:12 (UTC)
▼ Application Support	2019-02-08 15:17:26 (UTC)	2019-02-26 15:26:53 (UTC)
▶ com.crashlytics	2019-02-08 15:17:26 (UTC)	2019-02-08 15:17:26 (UTC)
▶ GoogleMobileAds	2019-02-26 15:26:53 (UTC)	2019-06-20 15:35:12 (UTC)
▶ io.branch	2019-02-08 15:17:30 (UTC)	2019-06-20 15:35:12 (UTC)
▼ Tinder	2019-02-08 15:17:27 (UTC)	2019-02-26 15:26:50 (UTC)
Tinder2.sqlite	2019-02-08 15:17:27 (UTC)	2019-06-20 15:33:18 (UTC)
com-accountkit-sdk-AppEvents...	2019-04-03 14:04:10 (UTC)	2019-04-03 14:04:10 (UTC)
com-accountkit-sdk-PersistedA...	2019-02-08 15:17:49 (UTC)	2019-02-08 15:17:49 (UTC)
com-facebook-sdk-AppEventsP...	2019-06-20 15:35:12 (UTC)	2019-06-20 15:35:12 (UTC)
com-facebook-sdk-AppEventsT...	2019-06-20 15:35:12 (UTC)	2019-06-20 15:35:12 (UTC)
com-facebook-sdk-PersistedAn...	2019-02-08 15:17:30 (UTC)	2019-02-08 15:17:30 (UTC)
▶ Cookies	2019-02-20 16:44:23 (UTC)	2019-06-20 15:35:13 (UTC)
▶ Preferences	2019-02-08 15:16:59 (UTC)	2019-06-20 15:35:23 (UTC)
▶ WebKit	2019-02-20 16:44:23 (UTC)	2019-02-20 16:44:23 (UTC)

FIGURE 10.1 Tinder SQLite database on iOS (iPhone)

A cursory view of the information in Figure 10.1 shows that there are many folders and files associated with a mobile app SQLite database. Ultimately, the database could have five tables or could have 100 tables, which means that a thorough examination can be a painstaking process. Within each SQLite database (*.sqlite)* you will find databases, which will contain the file extension *.db*; for example, *google_analytics.db*. You will often find recognizable files, like *.jpg* (picture images), *.vcf* (or vCard for your contacts), or *.mp3* (sound file).

The chart in Figure 10.2 provides a general outline of how an iOS application is stored on an iPhone or iPad.

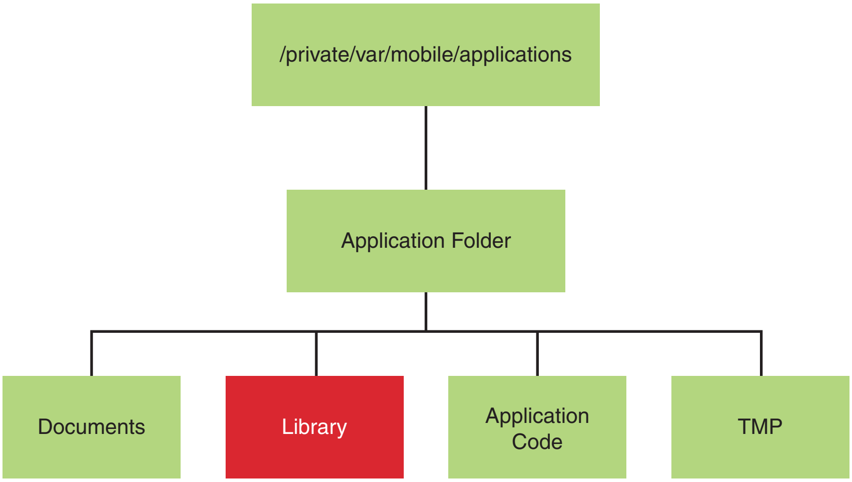
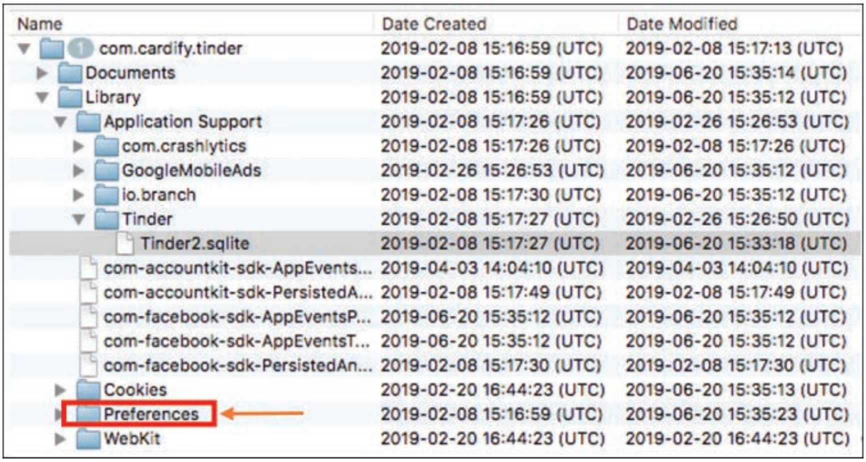


FIGURE 10.2 Application storage on iOS

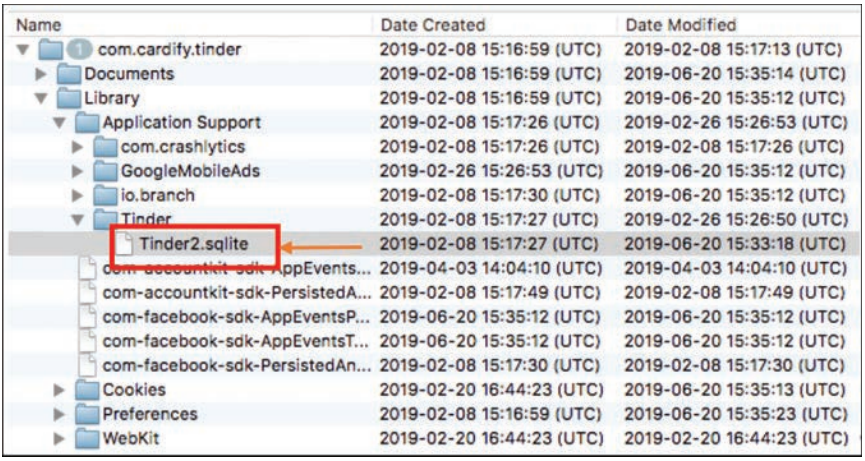
The `Library` folder, which is highlighted in Figure 10.2, is where you will find the all-important user data, including cache, cookies, and other personal information. In the `Preferences` folder, which is displayed and highlighted in Figure 10.3, you may actually discover usernames and passwords that are stored in plaintext.

In Figure 10.4, we can view the name *com.cardify.tinder* and this is referred to as a bundle ID. A **bundle ID** is a uniform type identifier, which is comprised of alphanumeric characters, that uniquely identifies a specific app. The bundle ID for Microsoft's iOS Outlook app is *com.microsoft.Office.Outlook*. Thus, the format for the bundle ID is generally com.<*YourCompany*>.<*AppName*>, which is referred to as a reverse-domain name style string. When you visit the Apple App Store and search for the Microsoft Outlook app for iOS, then you will arrive at this URL in your web browser: https://apps.apple.com/us/app/microsoft-outlook/id951937596. Notice the "id951937596", which identifies this app on the App Store. An iOS app also has a unique identifier known as an App ID. An **App ID** is a two-part string that identifies a development team (Team ID) and an application (bundle ID). The Team ID is created and assigned by Apple, while the bundle ID is generated by the app developer.



Name	Date Created	Date Modified
com.cardify.tinder	2019-02-08 15:16:59 (UTC)	2019-02-08 15:17:13 (UTC)
Documents	2019-02-08 15:16:59 (UTC)	2019-06-20 15:35:14 (UTC)
Library	2019-02-08 15:16:59 (UTC)	2019-06-20 15:35:12 (UTC)
Application Support	2019-02-08 15:17:26 (UTC)	2019-02-26 15:26:53 (UTC)
com.crashlytics	2019-02-08 15:17:26 (UTC)	2019-02-08 15:17:26 (UTC)
GoogleMobileAds	2019-02-26 15:26:53 (UTC)	2019-06-20 15:35:12 (UTC)
io.branch	2019-02-08 15:17:30 (UTC)	2019-06-20 15:35:12 (UTC)
Tinder	2019-02-08 15:17:27 (UTC)	2019-02-26 15:26:50 (UTC)
Tinder2.sqlite	2019-02-08 15:17:27 (UTC)	2019-06-20 15:33:18 (UTC)
com-accountkit-sdk-AppEvents...	2019-04-03 14:04:10 (UTC)	2019-04-03 14:04:10 (UTC)
com-accountkit-sdk-PersistedA...	2019-02-08 15:17:49 (UTC)	2019-02-08 15:17:49 (UTC)
com-facebook-sdk-AppEventsP...	2019-06-20 15:35:12 (UTC)	2019-06-20 15:35:12 (UTC)
com-facebook-sdk-AppEventsT...	2019-06-20 15:35:12 (UTC)	2019-06-20 15:35:12 (UTC)
com-facebook-sdk-PersistedAn...	2019-02-08 15:17:30 (UTC)	2019-02-08 15:17:30 (UTC)
Cookies	2019-02-20 16:44:23 (UTC)	2019-06-20 15:35:13 (UTC)
Preferences	2019-02-08 15:16:59 (UTC)	2019-06-20 15:35:23 (UTC)
WebKit	2019-02-20 16:44:23 (UTC)	2019-02-20 16:44:23 (UTC)

FIGURE 10.3 Tinder SQLite database on iOS



Name	Date Created	Date Modified
com.cardify.tinder	2019-02-08 15:16:59 (UTC)	2019-02-08 15:17:13 (UTC)
Documents	2019-02-08 15:16:59 (UTC)	2019-06-20 15:35:14 (UTC)
Library	2019-02-08 15:16:59 (UTC)	2019-06-20 15:35:12 (UTC)
Application Support	2019-02-08 15:17:26 (UTC)	2019-02-26 15:26:53 (UTC)
com.crashlytics	2019-02-08 15:17:26 (UTC)	2019-02-08 15:17:26 (UTC)
GoogleMobileAds	2019-02-26 15:26:53 (UTC)	2019-06-20 15:35:12 (UTC)
io.branch	2019-02-08 15:17:30 (UTC)	2019-06-20 15:35:12 (UTC)
Tinder	2019-02-08 15:17:27 (UTC)	2019-02-26 15:26:50 (UTC)
Tinder2.sqlite	2019-02-08 15:17:27 (UTC)	2019-06-20 15:33:18 (UTC)
com-accountkit-sdk-AppEvents...	2019-04-03 14:04:10 (UTC)	2019-04-03 14:04:10 (UTC)
com-accountkit-sdk-PersistedA...	2019-02-08 15:17:49 (UTC)	2019-02-08 15:17:49 (UTC)
com-facebook-sdk-AppEventsP...	2019-06-20 15:35:12 (UTC)	2019-06-20 15:35:12 (UTC)
com-facebook-sdk-AppEventsT...	2019-06-20 15:35:12 (UTC)	2019-06-20 15:35:12 (UTC)
com-facebook-sdk-PersistedAn...	2019-02-08 15:17:30 (UTC)	2019-02-08 15:17:30 (UTC)
Cookies	2019-02-20 16:44:23 (UTC)	2019-06-20 15:35:13 (UTC)
Preferences	2019-02-08 15:16:59 (UTC)	2019-06-20 15:35:23 (UTC)
WebKit	2019-02-20 16:44:23 (UTC)	2019-02-20 16:44:23 (UTC)

FIGURE 10.4 Tinder SQLite database on iOS

Static Analysis: Code Review

Another form of static analysis refers to performing a code review on a mobile app, which can help the investigator understand the type of evidence that is available. In terms of the evidence available for an Android app (.apk or Android Package) there is the manifest, which shows the permissions associated with a particular app. For example, the manifest may show that the app is collecting user location information ("COARSE_LOCATION" and/or "FINE_LOCATION"). ACCESS_COARSE_LOCATION is a permission that enables the app to access the approximate location of the user device, which is based on NETWORK_PROVIDER (cell sites, i.e. cell towers). ACCESS_FINE_LOCATION enables the app to determine the location of the user device based on NETWORK_PROVIDER and GPS (GPS_PROVIDER). An Android application contains a file at the root of the project source set, which is called *AndroidManifest.xml*. An **Android manifest file** contains the application's package name, its functionality, permissions, hardware, and software requirements for installation.

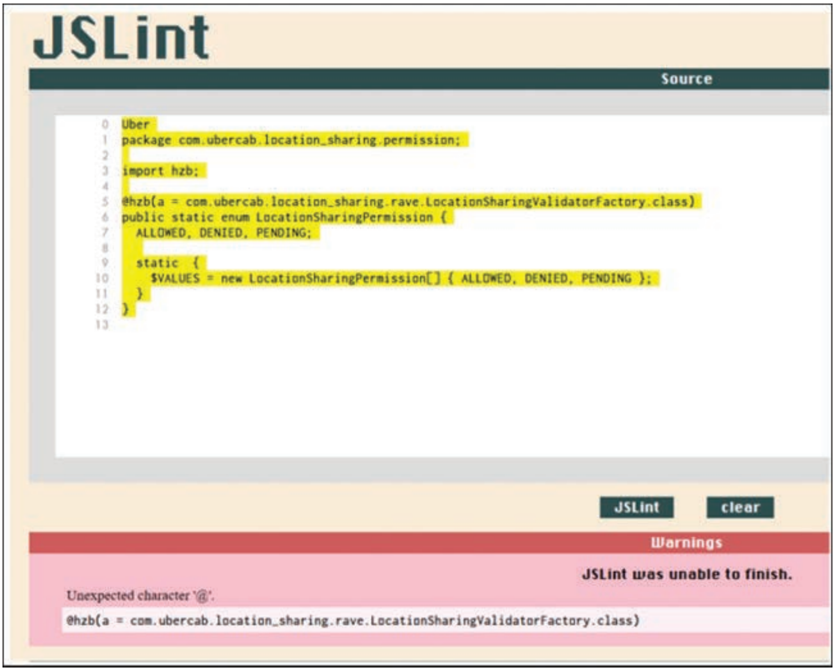
Understanding the permissions associated with an app allows the investigator to understand the type of evidence that can be requested from the provider and the type of evidence to look for when examining the SQLite database. The latter is important because examining one database can take many days, or even weeks, and therefore limiting the scope of your analysis is key. Example 10.1 shows a small extract from an Android manifest for WhatsApp.

EXAMPLE 10.1 Android Permissions Manifest for WhatsApp

```
<manifest xmlns:"http://schemas.android.com/apk/res/android"
android:versionCode="451048" android:versionName="2.12.550" package="com.whatsapp"
platformBuildVersionCode="23" platformBuildVersionName="6.0-2166767">
    <uses-sdk android:minSdkVersion="7" android:targetSdkVersion="23" />
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
    <uses-permission android:name="android.permission.AUTHENTICATE_ACCOUNTS" />
    <uses-permission android:name="android.permission.BLUETOOTH" />
    <uses-permission android:name="android.permission.BROADCAST_STICKY" />
    <uses-permission android:name="android.permission.CAMERA" />
    <uses-permission android:name="android.permission.CHANGE_WIFI_STATE" />
    <uses-permission android:name="android.permission.GET_ACCOUNTS" />
    <uses-permission android:name="android.permission.GET_TASKS" />
    <uses-permission android:name="android.permission.INSTALL_SHORTCUT" />
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.MANAGE_ACCOUNTS" />
    <uses-permission android:name="android.permission.MODIFY_AUDIO_SETTINGS" />
    <uses-permission android:name="android.permission.READ_CONTACTS" />
    <uses-permission android:name="android.permission.READ_PHONE_STATE" />
```

An understanding of the manifest is also important from a mobile security perspective. Many privacy policy statements are misleading or confusing and provide poor guidance about how trustworthy a mobile app is. The Federal Trade Commission (FTC), for example, investigated a popular free app for Android, called the Brightest Flashlight, after it was discovered that the app requested many more permissions from the user's device beyond the light function on the device. Therefore, some app permissions are high risk, while other permissions are low risk.

A Web search for the "Uber APK file", or any other APK file, quickly identifies where the application package can be downloaded. Once the APK has been downloaded, there are a number of applications that can be used to review the code and manifest for the APK. One tool for reviewing the APK developer code is dex2jar (dex compiler), which can be downloaded from SourceForge. Another application for viewing the APK is FileViewer Plus. One preferred tool is an online Java APK decompiler application, which is available from www.javadecompilers.com/apk. With this tool, you can decompile your APK in a web browser without downloading an APK decompiler to your computer. Therefore, you do not need to worry whether the application that you are downloading is from a trusted source because the application is being run from their web server and not from your computer. There are numerous other source code analytical tools that an investigator can use, including SourceMeter, JSLint, and FindBugs. Figure 10.5 shows the JSLint user interface.



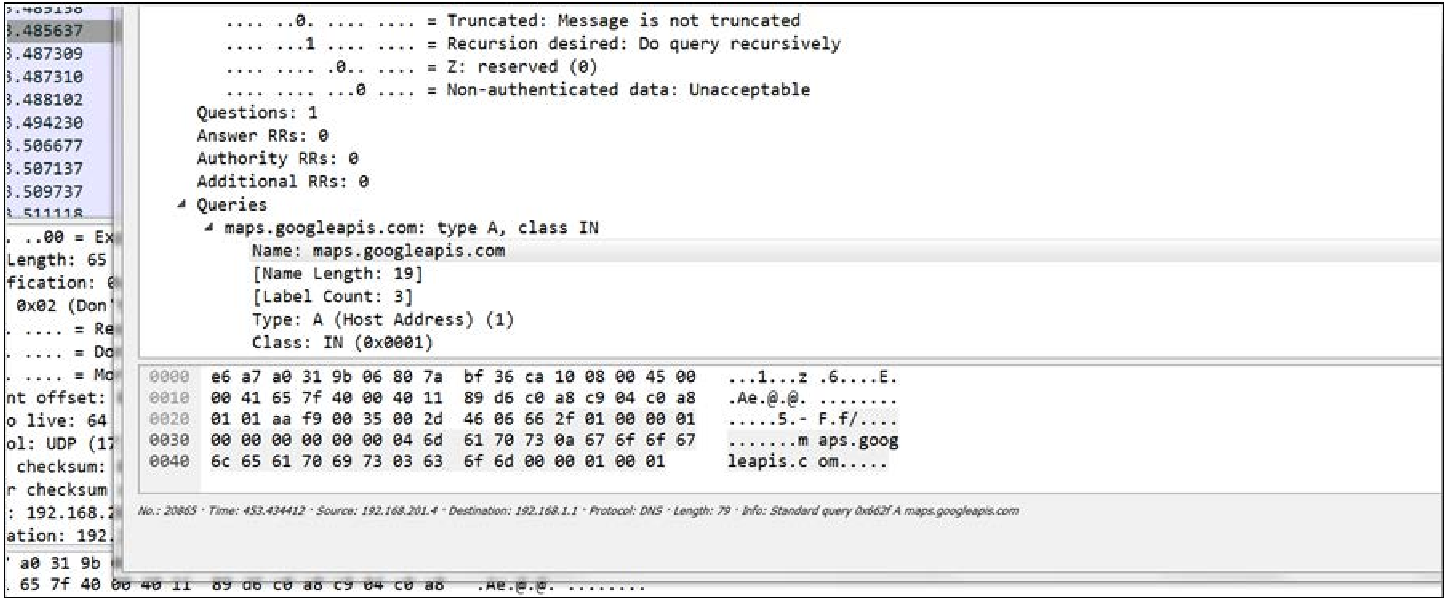
```
Uber
package com.ubercab.location_sharing.permission;

import hzb;

@hzb(a = com.ubercab.location_sharing.rave.LocationSharingValidatorFactory.class)
public static enum LocationSharingPermission {
  ALLOWED, DENIED, PENDING;

  static  {
    $VALUES = new LocationSharingPermission[] { ALLOWED, DENIED, PENDING };
  }
}
```

```
JSLint was unable to finish.
Unexpected character '@'.
@hzb(a = com.ubercab.location_sharing.rave.LocationSharingValidatorFactory.class)
```

FIGURE 10.5 JSLint user interface

Dynamic Analysis

A dynamic analysis of the app is an analysis of the behavior of the application once it has been executed (or run). An **Android emulator** is an application that simulates, or runs, the Android operating system in a virtual machine. These applications are generally developed for use with a personal computer and run as a virtual machine. App developers use an emulator to analyze how their apps will run before making them available to the public. However, an emulator can also benefit investigators who are interested in viewing the behavior of an app—especially if an app potentially contains malware. This is the benefit of using an emulator that operates as a virtual machine. An investigator may also be interested in monitoring the permissions and DNS connections associated with an executed mobile app. In terms of monitoring DNS connections (connections to servers), there is Wireshark (Windows) and Debookee (macOS), which are very effective at monitoring these connections over a wireless network. Figure 10.6 shows a screenshot of a pcap (packet capture) file from Wireshark. A **pcap file** is a wireless packet that contains user data and network data related to the sender and receiver of that data.



```
3.485637
3.487309
3.487310
3.488102
3.494230
3.506677
3.507137
3.509737
. ..00 = Ex
Length: 65
fication: 0
 0x02 (Don'
. .... = Re
. .... = Do
. .... = Mo
nt offset:
o live: 64
ol: UDP (17
 checksum:
r checksum
: 192.168.2
ation: 192.
' a0 31 9b
. 65 7f 40 00 40 11  89 d6 c0 a8 c9 04 c0 a8   .Ae.@.@. ........
```

```
    .... ..0. .... .... = Truncated: Message is not truncated
    .... ...1 .... .... = Recursion desired: Do query recursively
    .... .... .0.. .... = Z: reserved (0)
    .... .... ...0 .... = Non-authenticated data: Unacceptable
  Questions: 1
  Answer RRs: 0
  Authority RRs: 0
  Additional RRs: 0
◢ Queries
  ◢ maps.googleapis.com: type A, class IN
      Name: maps.googleapis.com
      [Name Length: 19]
      [Label Count: 3]
      Type: A (Host Address) (1)
      Class: IN (0x0001)

0000  e6 a7 a0 31 9b 06 80 7a  bf 36 ca 10 08 00 45 00   ...1...z .6....E.
0010  00 41 65 7f 40 00 40 11  89 d6 c0 a8 c9 04 c0 a8   .Ae.@.@. ........
0020  01 01 aa f9 00 35 00 2d  46 06 66 2f 01 00 00 01   .....5.- F.f/....
0030  00 00 00 00 00 00 04 6d  61 70 73 0a 67 6f 6f 67   .......m aps.goog
0040  6c 65 61 70 69 73 03 63  6f 6d 00 00 01 00 01      leapis.c om.....

No.: 20865 · Time: 453.434412 · Source: 192.168.201.4 · Destination: 192.168.1.1 · Protocol: DNS · Length: 79 · Info: Standard query 0x662f A maps.googleapis.com
```

FIGURE 10.6 Google Maps API identified in a PCAP captured by Wireshark

Note

When performing any type of wireless monitoring, ensure that you have permission to be on a particular network and ensure that you are only monitoring your wireless traffic.

To remain safe and compliant, consider using a personal hotspot device, like a Verizon Jetpack, in a secure lab. A tool like Debookee also has the ability to encrypt some wireless traffic, which means that while app data may be encrypted on the device and on the server, often companies will implement poor encryption protocols, whereby the data in transmission can be intercepted and viewed in plaintext. Thus, tools like Debookee can also be used, by security professionals analyzing apps, to try to determine how secure apps are.

Introduction to Debookee

Debookee is a comprehensive wireless packet sniffer for macOS. The tool is not passive as it performs a man-in-the-middle (MITM) attack to intercept data from mobile and IoT devices. A **man-in-the-middle (MITM) attack** is an attempt to intercept electronic communications between two computing devices, with the intent to decipher encrypted messages. The tool also performs SSL/TLS decryption. Debookee supports numerous protocols, including HTTP, HTTPS, DNS, TCP, DHCP, SIP, and RTP (VoIP). The tool can be used to identify what data is being collected and shared by mobile apps. In other words, you can identify DNS connections to servers around the world and other companies that could be potentially subpoenaed for information. The data generated from one mobile app can be shared with fifty or more third-party companies, which are mostly analytics companies like Crashlytics, UXCam, Fabric, etc.

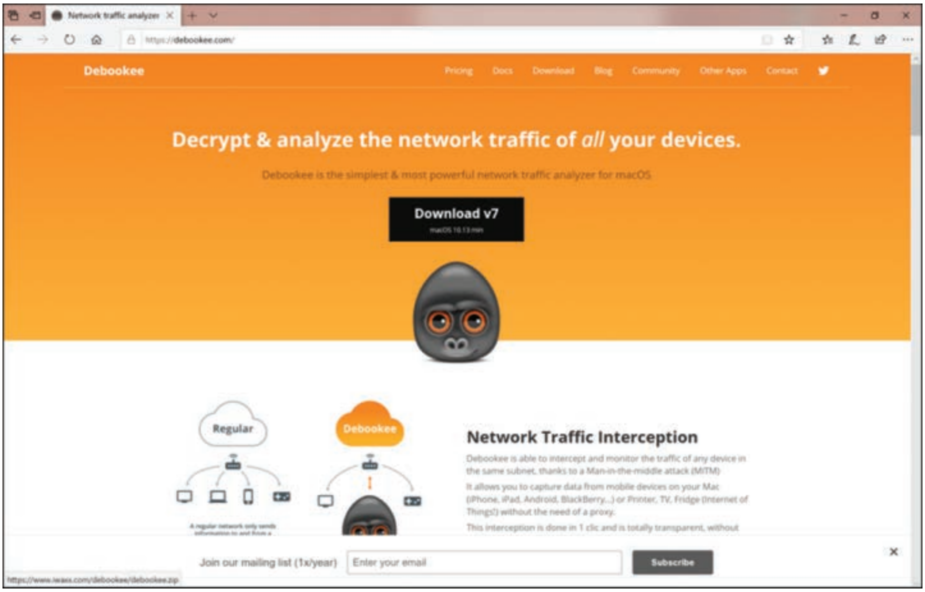
On the homepage of the Debookee website, click the **Download** button and install the software.

Note

You do not need to purchase the software but can begin by using the trial version. You may of course later decide to purchase the software, which is relatively inexpensive, and one license can be used on two different computers.

Once you install the software and start the program, you will see an interface, similar to Figure 10.8. The IP address, MAC address, and host name that are displayed provide information about your device.

Figure 10.9 shows a close-up of the information that we just discussed. Click the **Start LanScan** button as highlighted in Figure 10.9.



Decrypt & analyze the network traffic of *all* your devices.

Debookee is the simplest & most powerful network traffic analyzer for macOS

Network Traffic Interception

Debookee is able to intercept and monitor the traffic of any device in the same subnet, thanks to a Man-in-the-middle attack (MITM)

It allows you to capture data from mobile devices on your Mac (iPhone, iPad, Android, BlackBerry...) or Printer, TV, Fridge (Internet of Things!) without the need of a proxy.

This interception is done in 1 clic and is totally transparent, without

FIGURE 10.7 Debookee home page

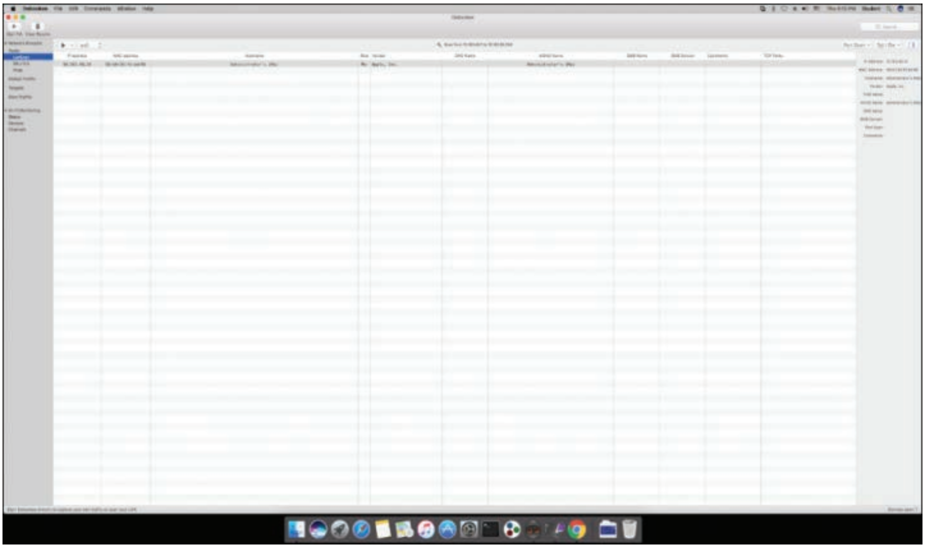
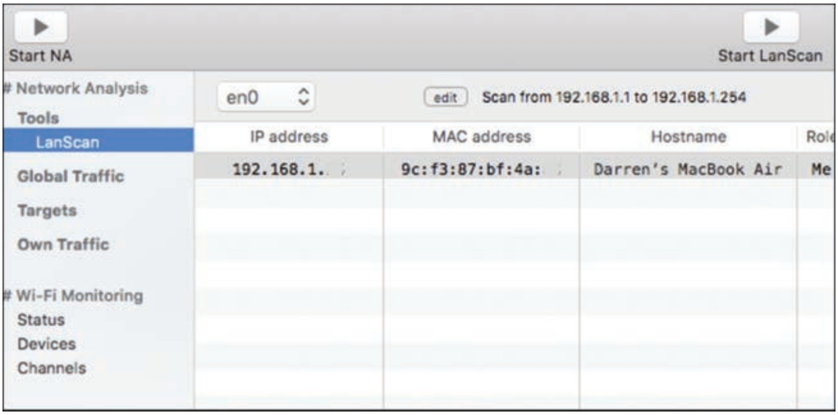


FIGURE 10.8 Debookee user interface

You will then see a list of all devices that are connected to the same wireless access point as your computer. Once you select your target device, click the **Pcap** option, on the upper left of your screen, and then click **Save Pcap files**, as shown in Figure 10.10.

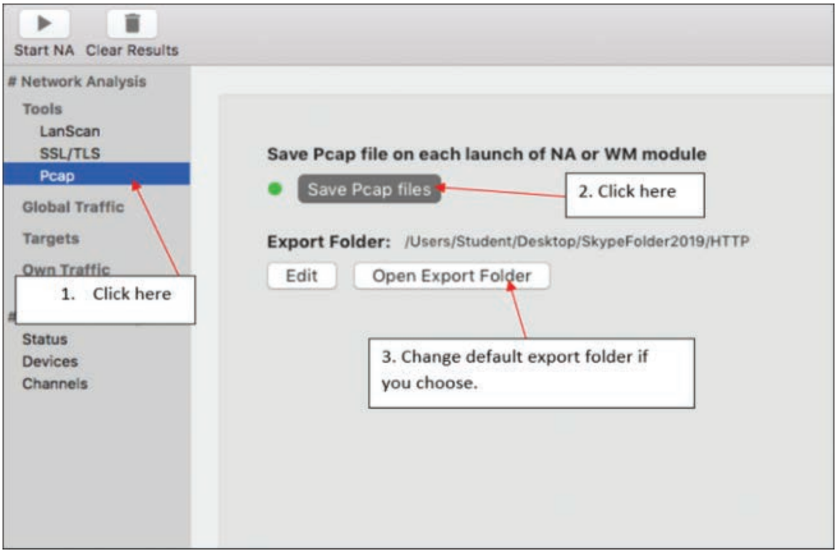


en0

Scan from 192.168.1.1 to 192.168.1.254

IP address	MAC address	Hostname	Rol
192.168.1.	9c:f3:87:bf:4a:	Darren's MacBook Air	Me

FIGURE 10.9 Debookee user interface with host computer information displayed



Save Pcap file on each launch of NA or WM module

Export Folder: /Users/Student/Desktop/SkypeFolder2019/HTTP

1. Click here

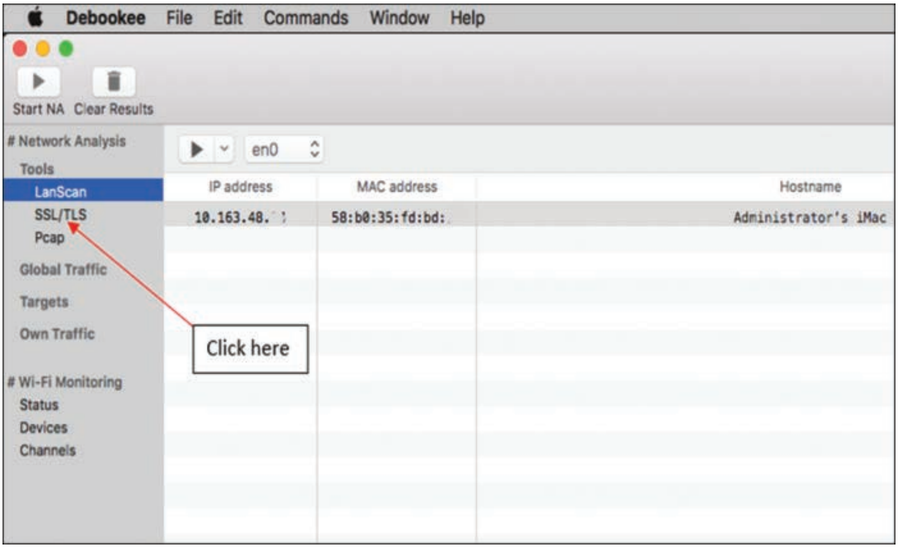
2. Click here

3. Change default export folder if you choose.

FIGURE 10.10 *Save Pcap files* option in Debookee

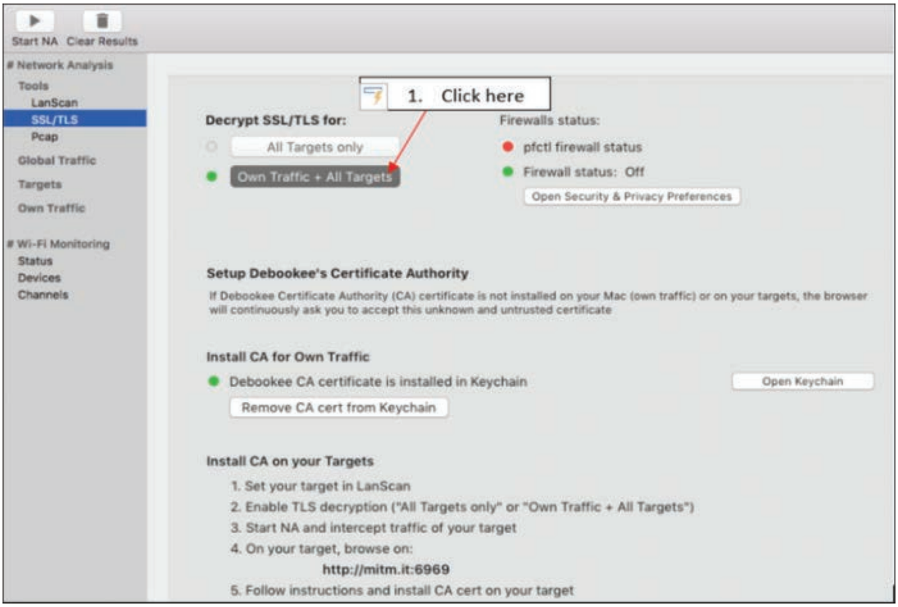
You can then click the **Open Export Folder** button to change the default export folder. There is an add-on tool in Debookee, which allows you to decrypt the contents of the pcap files. If you purchase this option, you can click the **SSL/TLS** button displayed in Figure 10.11.

The next step in the TLS decryption process is to install the certificate authority (CA) on the machine (see Figure 10-12). To start your NA, click the Play button ▶ in the very top left of your application screen (underneath it says, "Start NA"). Once the trust certificate has been installed, you should stop the NA (Network Analysis) by clicking the same button.



IP address	MAC address	Hostname
10.163.48.	58:b0:35:fd:bd:	Administrator's iMac

FIGURE 10.11 SSL/TLS decryption option in Debookee



Decrypt SSL/TLS for:

○ All Targets only

● Own Traffic + All Targets

Firewalls status:

● pfctl firewall status

● Firewall status: Off

Setup Debookee's Certificate Authority

If Debookee Certificate Authority (CA) certificate is not installed on your Mac (own traffic) or on your targets, the browser will continuously ask you to accept this unknown and untrusted certificate

Install CA for Own Traffic

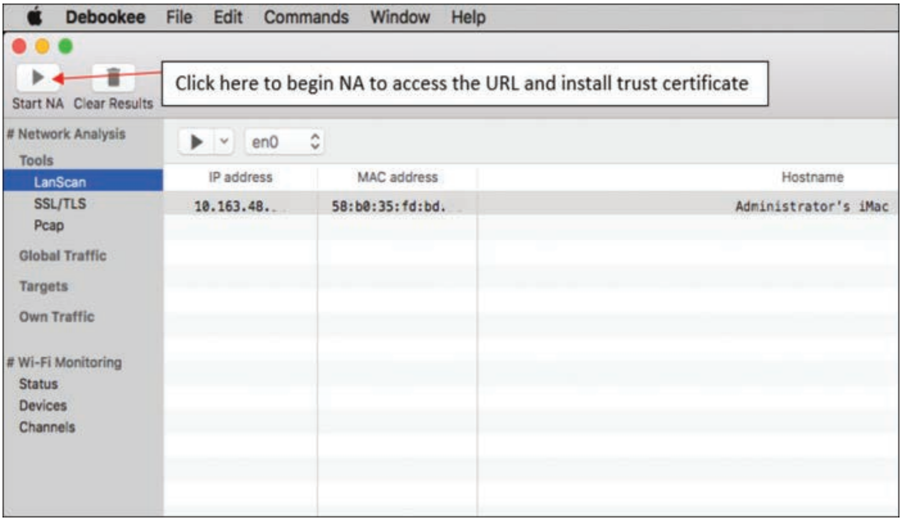
● Debookee CA certificate is installed in Keychain

Install CA on your Targets

1. Set your target in LanScan
2. Enable TLS decryption ("All Targets only" or "Own Traffic + All Targets")
3. Start NA and intercept traffic of your target
4. On your target, browse on:
   http://mitm.it:6969
5. Follow instructions and install CA cert on your target

FIGURE 10.12 Decryption option in Debookee

From the screen in Figure 10.13, click the **Start NA ▶** button again. Open the webpage, or application, you want to analyze (or the device that you wish to monitor), and begin generating data packets by opening and closing different functions, sending messages, or just using the application.



IP address	MAC address	Hostname
10.163.48..	58:b0:35:fd:bd.	Administrator's iMac

FIGURE 10.13 Start NA option in Debookee

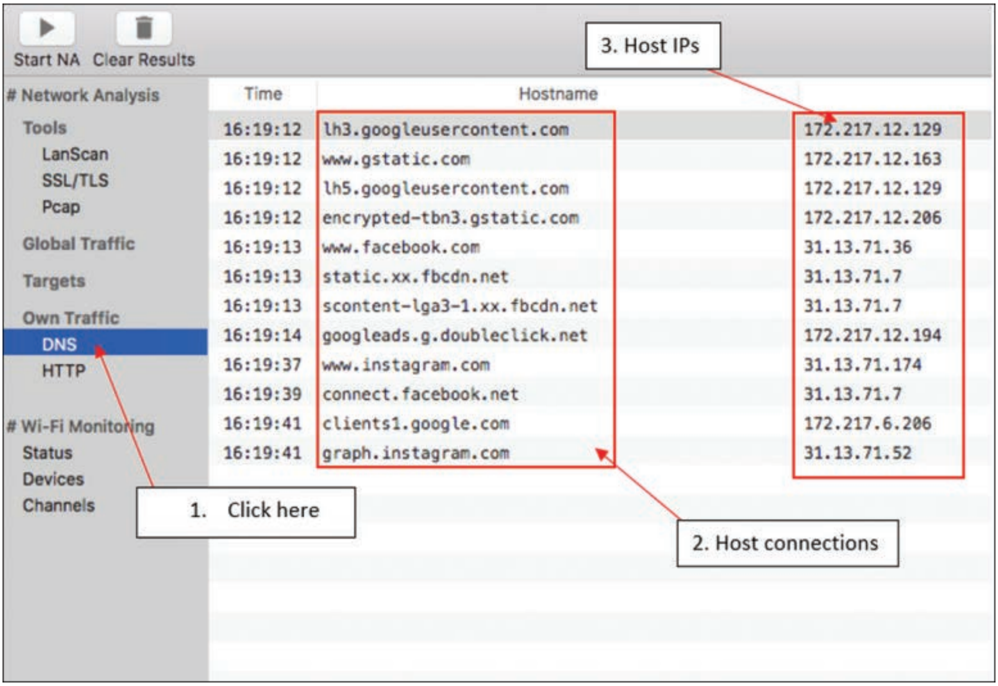
On the left column in Figure 10.14, under **Own Traffic**, you will see that **DNS** and **HTTP** have populated. The NA will run continuously until you terminate it. When you are satisfied with the data collected, press the stop button. Remember that your pcap files are automatically exported to the folder that you previously selected.

Click **DNS** in the left column and you will see all DNS connections made during the NA (timestamped) with the hostname and/or IP address. These are the IP addresses and hosts that you can analyze, in addition to the pcaps.

It is recommended that you click **File > Export** and save this list as a .doc or a .txt file. You can then use some open source DNS analysis tools, including www.robtex.com and www.dnsdumpster.com.

Clicking the **HTTP** button, as shown in Figure 10.15, will display an itemized list of every packet transmitted over HTTP, HTTPS, TCP, SIP, IMAP, and other protocols. If you did not purchase the SSL/TLS decrypt module, HTTPS packets (transmitted over port 443 using TLSv1.2) will display in red, and you will not be able to read the data until you decrypt the packets. Port 443 is the port number for secure HTTP communications—in other words, Web traffic. If you did purchase the SSL/TLS decrypt module, HTTPS packets will display in black, and when you click on them, the data will be displayed in plaintext in the data field.

Click on a packet that you wish to examine. In the data field you will see some text populate underneath the tab labeled **Request**. Upon further inspection of the data field, you will see the full GET request along with the packet parameters and data, as displayed in Figure 10.16. **GET** is an HTTP method used to request data from a specific resource, like a web server.



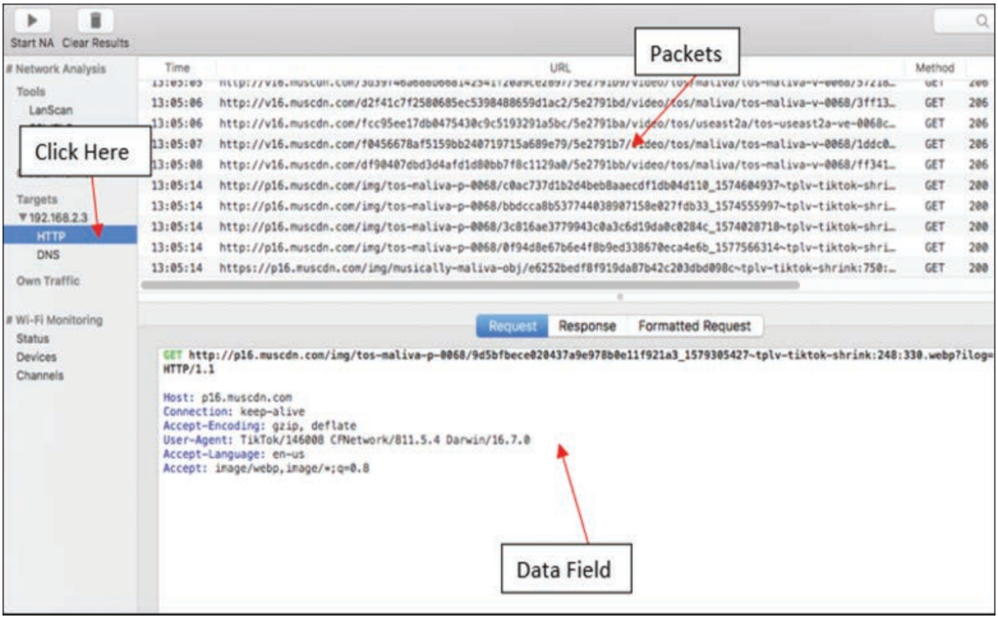
Time	Hostname	
16:19:12	lh3.googleusercontent.com	172.217.12.129
16:19:12	www.gstatic.com	172.217.12.163
16:19:12	lh5.googleusercontent.com	172.217.12.129
16:19:12	encrypted-tbn3.gstatic.com	172.217.12.206
16:19:13	www.facebook.com	31.13.71.36
16:19:13	static.xx.fbcdn.net	31.13.71.7
16:19:13	scontent-lga3-1.xx.fbcdn.net	31.13.71.7
16:19:14	googleads.g.doubleclick.net	172.217.12.194
16:19:37	www.instagram.com	31.13.71.174
16:19:39	connect.facebook.net	31.13.71.7
16:19:41	clients1.google.com	172.217.6.206
16:19:41	graph.instagram.com	31.13.71.52

3. Host IPs

1. Click here

2. Host connections

FIGURE 10.14 DNS connections captured



Time	URL	Method	
13:05:06	http://v16.muscdn.com/d2f41c7f2580685ec5398488659d1ac2/5e2791bd/video/tos/maliva/tos-maliva-v-0068/3ff13…	GET	206
13:05:06	http://v16.muscdn.com/fcc95ee17db0475430c9c5193291a5bc/5e2791ba/video/tos/useast2a/tos-useast2a-ve-0068c…	GET	206
13:05:07	http://v16.muscdn.com/f0456678af5159bb240719715a689e79/5e2791b7/video/tos/maliva/tos-maliva-v-0068/1ddc0…	GET	206
13:05:08	http://v16.muscdn.com/df90407dbd3d4afd1d80bb7f8c1129a0/5e2791bb/video/tos/maliva/tos-maliva-v-0068/ff341…	GET	206
13:05:14	http://p16.muscdn.com/img/tos-maliva-p-0068/c0ac737d1b2d4beb8aaecdf1db04d110_1574604937~tplv-tiktok-shri…	GET	200
13:05:14	http://p16.muscdn.com/img/tos-maliva-p-0068/bbdcca8b537744038907158e027fdb33_1574555997~tplv-tiktok-shri…	GET	200
13:05:14	http://p16.muscdn.com/img/tos-maliva-p-0068/3c816ae3779943c0a3c6d19da0c0284c_1574028718~tplv-tiktok-shri…	GET	200
13:05:14	http://p16.muscdn.com/img/tos-maliva-p-0068/0f94d8e67b6e4f8b9ed338670eca4e6b_1577566314~tplv-tiktok-shri…	GET	200
13:05:14	https://p16.muscdn.com/img/musically-maliva-obj/e6252bedf8f919da87b42c203dbd098c~tplv-tiktok-shrink:750:…	GET	200

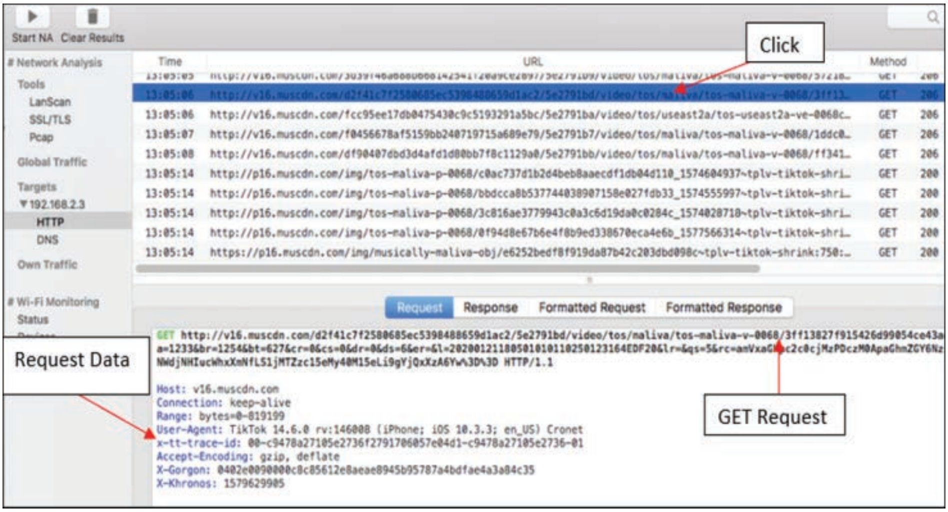
```
GET http://p16.muscdn.com/img/tos-maliva-p-0068/9d5bfbece020437a9e978b0e11f921a3_1579305427~tplv-tiktok-shrink:248:330.webp?ilog=
HTTP/1.1

Host: p16.muscdn.com
Connection: keep-alive
Accept-Encoding: gzip, deflate
User-Agent: TikTok/146008 CFNetwork/811.5.4 Darwin/16.7.0
Accept-Language: en-us
Accept: image/webp,image/*;q=0.8
```

Packets

Click Here

Data Field

FIGURE 10.15 Decrypted TikTok packet (pcap)

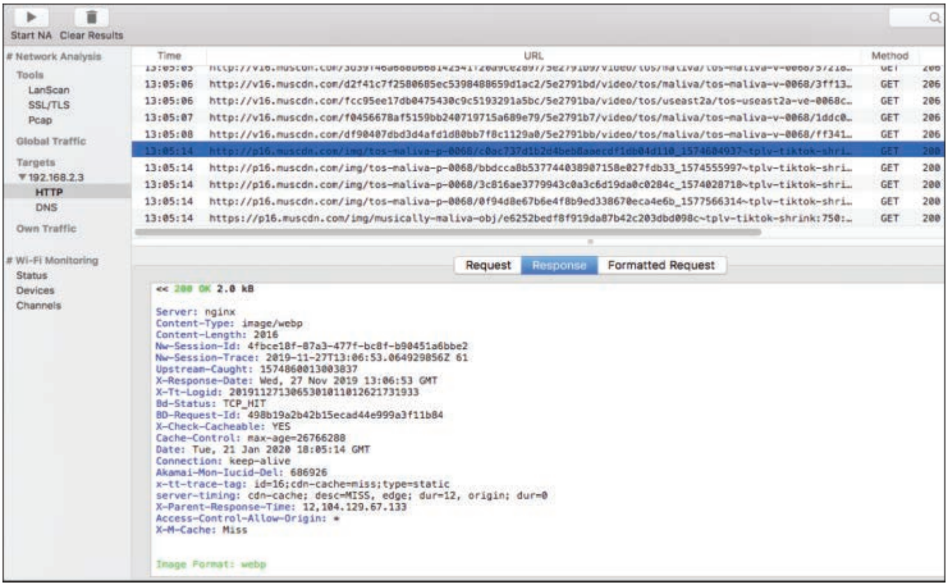


```
Time      URL                                                                                      Method
13:05:06  http://v16.muscdn.com/d2f41c7f2580685ec5398488659d1ac2/5e2791bd/video/tos/maliva/tos-maliva-v-0068/3ff13…  GET  206
13:05:06  http://v16.muscdn.com/fcc95ee17db0475430c9c5193291a5bc/5e2791ba/video/tos/useast2a/tos-useast2a-ve-0068c…  GET  206
13:05:07  http://v16.muscdn.com/f0456678af5159bb240719715a689e79/5e2791b7/video/tos/maliva/tos-maliva-v-0068/1ddc0…  GET  206
13:05:08  http://v16.muscdn.com/df90407dbd3d4afd1d80bb7f8c1129a0/5e2791bb/video/tos/maliva/tos-maliva-v-0068/ff341…  GET  206
13:05:14  http://p16.muscdn.com/img/tos-maliva-p-0068/c0ac737d1b2d4beb8aaecdf1db04d110_1574604937~tplv-tiktok-shri…  GET  200
13:05:14  http://p16.muscdn.com/img/tos-maliva-p-0068/bbdcca8b537744038907158e027fdb33_1574555997~tplv-tiktok-shri…  GET  200
13:05:14  http://p16.muscdn.com/img/tos-maliva-p-0068/3c816ae3779943c0a3c6d19da0c0284c_1574028710~tplv-tiktok-shri…  GET  200
13:05:14  http://p16.muscdn.com/img/tos-maliva-p-0068/0f94d8e67b6e4f8b9ed338670eca4e6b_1577566314~tplv-tiktok-shri…  GET  200
13:05:14  https://p16.muscdn.com/img/musically-maliva-obj/e6252bedf8f919da87b42c203dbd098c~tplv-tiktok-shrink:750:…  GET  200
```

```
GET http://v16.muscdn.com/d2f41c7f2580685ec5398488659d1ac2/5e2791bd/video/tos/maliva/tos-maliva-v-0068/3ff13827f915426d99054ce43a
a=1233&br=1254&bt=627&cr=0&cs=0&dr=0&ds=6&er=&l=20200121180501010110250123164EDF20&lr=&qs=5&rc=anVxaG[illegible]c2c0cjMzPDczM0ApaGhnZGY6Nz
NWdjNHIucWhxXmNfLS1jMTZzc15eMy40M15eLi9gYjQxXzA6Yw%3D%3D HTTP/1.1

Host: v16.muscdn.com
Connection: keep-alive
Range: bytes=0-819199
User-Agent: TikTok 14.6.0 rv:146008 (iPhone; iOS 10.3.3; en_US) Cronet
x-tt-trace-id: 00-c9478a27105e2736f2791706057e04d1-c9478a27105e2736-01
Accept-Encoding: gzip, deflate
X-Gorgon: 0402e0090000c8c85612e8aeae8945b95787a4bdfae4a3a84c35
X-Khronos: 1579629905
```

FIGURE 10.16 GET request data displayed

You may then click the **Response** tab to view the webpage or application response packet. Figure 10.17 displays a webpage response. Status code 200 means that it was successfully downloaded.



```
Time      URL                                                                                      Method
13:05:06  http://v16.muscdn.com/d2f41c7f2580685ec5398488659d1ac2/5e2791bd/video/tos/maliva/tos-maliva-v-0068/3ff13…  GET  206
13:05:06  http://v16.muscdn.com/fcc95ee17db0475430c9c5193291a5bc/5e2791ba/video/tos/useast2a/tos-useast2a-ve-0068c…  GET  206
13:05:07  http://v16.muscdn.com/f0456678af5159bb240719715a689e79/5e2791b7/video/tos/maliva/tos-maliva-v-0068/1ddc0…  GET  206
13:05:08  http://v16.muscdn.com/df90407dbd3d4afd1d80bb7f8c1129a0/5e2791bb/video/tos/maliva/tos-maliva-v-0068/ff341…  GET  206
13:05:14  http://p16.muscdn.com/img/tos-maliva-p-0068/c0ac737d1b2d4beb8aaecdf1db04d110_1574604937~tplv-tiktok-shri…  GET  200
13:05:14  http://p16.muscdn.com/img/tos-maliva-p-0068/bbdcca8b537744038907158e027fdb33_1574555997~tplv-tiktok-shri…  GET  200
13:05:14  http://p16.muscdn.com/img/tos-maliva-p-0068/3c816ae3779943c0a3c6d19da0c0284c_1574028710~tplv-tiktok-shri…  GET  200
13:05:14  http://p16.muscdn.com/img/tos-maliva-p-0068/0f94d8e67b6e4f8b9ed338670eca4e6b_1577566314~tplv-tiktok-shri…  GET  200
13:05:14  https://p16.muscdn.com/img/musically-maliva-obj/e6252bedf8f919da87b42c203dbd098c~tplv-tiktok-shrink:750:…  GET  200
```

```
<< 200 OK 2.0 kB

Server: nginx
Content-Type: image/webp
Content-Length: 2016
Nw-Session-Id: 4fbce18f-87a3-477f-bc8f-b90451a6bbe2
Nw-Session-Trace: 2019-11-27T13:06:53.064929856Z 61
Upstream-Caught: 1574860013003837
X-Response-Date: Wed, 27 Nov 2019 13:06:53 GMT
X-Tt-Logid: 20191127130653010110126217319 33
Bd-Status: TCP_HIT
BD-Request-Id: 498b19a2b42b15ecad44e999a3f11b84
X-Check-Cacheable: YES
Cache-Control: max-age=26766288
Date: Tue, 21 Jan 2020 18:05:14 GMT
Connection: keep-alive
Akamai-Mon-Iucid-Del: 686926
x-tt-trace-tag: id=16;cdn-cache=miss;type=static
server-timing: cdn-cache; desc=MISS, edge; dur=12, origin; dur=0
X-Parent-Response-Time: 12,104.129.67.133
Access-Control-Allow-Origin: *
X-M-Cache: Miss


Image Format: webp
```

FIGURE 10.17 Response results

You can choose to export your packets so that they can be analyzed later. You can select to view your packet data in a text file or in a Word document. Figure 10.18 displays the option to export the packet data.