

PEARSON IT

CYBERSECURITY CURRICULUM



SECOND EDITION

A PRACTICAL GUIDE TO DIGITAL FORENSICS INVESTIGATIONS

DR. DARREN R. HAYES

Table of Contents

Introduction	xxxvii
Chapter 1: The Scope of Digital Forensics	2
Popular Myths about Computer Forensics	3
Myth 1: Computer Forensics Is the Same As Computer Security	3
Myth 2: Computer Forensics Is about Investigating Computers	3
Myth 3: Computer Forensics Is about Investigating Computer Crime	3
Myth 4: Computer Forensics Is Really Used to Resurrect Deleted Files....	4
Types of Digital Forensic Evidence Recovered	5
Electronic Mail (Email).....	5
Images	7
Video	8
Websites Visited and Internet Searches	9
Cellphone Forensics	10
IoT Forensics.....	10
What Skills Must a Digital Forensics Investigator Possess?	10
Computer Science Knowledge	10
Legal Expertise	11
Communication Skills	11
Linguistic Abilities	12
Continuous Learning.....	12
Programming	12
An Appreciation for Confidentiality	12
The Importance of Digital Forensics	12
Job Opportunities	13
A History of Digital Forensics	14
1980s: The Advent of the Personal Computer.....	15
1990s: The Impact of the Internet.....	15
2000s: Virtual Currencies, IoT, Encryption, and the Edward Snowden Effect	20

Training and Education	21
Law Enforcement Training	21
High School Training.....	22
University Training.....	22
Professional Certifications	22
Summary	27
Key Terms	28
Assessment	30
Chapter 2: Windows Operating and File Systems	34
Physical and Logical Storage	36
File Storage.....	36
Paging	39
File Conversion and Numbering Formats	42
Conversion of Binary to Decimal	42
Hexadecimal Numbering	42
Conversion of Hexadecimal to Decimal.....	43
Conversion of Hexadecimal to ASCII	44
Using Hex to Identify a File Type	47
Unicode.....	47
Operating Systems	47
The Boot Process	48
Windows File Systems.....	49
Windows Registry	59
Registry Data Types.....	61
FTK Registry Viewer.....	62
Microsoft Office	62
Microsoft Windows Features	63
Windows Vista	63
Windows 7	68

Windows 8.1	79
Windows 10	82
Microsoft Office 365.....	83
Summary	84
Key Terms	85
Assessment	88
Chapter 3: Handling Computer Hardware	92
Hard Disk Drives	93
Small Computer System Interface (SCSI).....	93
Integrated Drive Electronics (IDE)	94
Serial ATA (SATA).....	95
Cloning a PATA or SATA Hard Disk	97
Cloning Devices	98
Removable Memory	105
FireWire	105
USB Flash Drives	106
External Hard Drives	107
MultiMediaCards (MMCs)	108
Summary	120
Key Terms	120
Assessment	122
Reference	125
Chapter 4: Acquiring Evidence in a Computer Forensics Lab	126
Lab Requirements	127
American Society of Crime Laboratory Directors (ASCLD).....	127
American Society of Crime Laboratory Directors/Lab Accreditation Board (ASCLD/LAB)	127
ASCLD/LAB Guidelines for Forensic Laboratory Management Practices.....	127

ISO/IEC 17025:2017	129
Scientific Working Group on Digital Evidence (SWGDE)	129
Private-Sector Computer Forensics Laboratories	130
Evidence Acquisition Laboratory	131
Email Preparation Laboratory	131
Inventory Control	131
Laboratory Information Management Systems	131
Web Hosting	132
Computer Forensics Laboratory Requirements	132
Laboratory Layout.....	132
Laboratory Management.....	154
Laboratory Access	155
Extracting Evidence from a Device	157
Using the dd Utility	157
Using Global Regular Expressions Print (GREP)	158
Skimmers	166
Steganography	168
Summary	170
Key Terms	170
Assessment	172
Chapter 5: Online Investigations	176
Working Undercover	177
Generating an Identity.....	178
Generating an Email Account	179
Masking Your Identity	181
Dark Web Investigations	184
OSINT Framework	184
Tor	184

Invisible Internet Project	186
Freenet	186
SecureDrop	186
Dark Web Marketplaces	186
Virtual Currencies	188
Bitcoin	188
Venmo and Vicemo	189
Website Evidence	189
Website Archives	189
Website Statistics	190
Background Searches on a Suspect	191
Finding Personal Information	192
Personal Interests and User Groups	195
Searching for Stolen Property	196
Online Crime	209
Identity Theft	210
Credit Cards for Sale	210
Electronic Medical Records	210
Counterfeit and Counter-proliferation Investigations (CPI)	211
Cyberbullying	211
Social Networking	211
Capturing Online Communications	212
Using Screen Captures	212
Using Video	213
Viewing Cookies	214
Using Windows Registry	215
Edge Web Browser	215
Summary	216
Key Terms	216
Assessment	218

Chapter 6: Documenting the Investigation	224
Obtaining Evidence from a Service Provider	224
Documenting a Crime Scene	226
Seizing Evidence	227
Crime Scene Examinations.....	227
Crime Scene Investigator Equipment	228
Documenting the Evidence.	229
Completing a Chain of Custody Form	229
Completing a Computer Worksheet	230
Completing a Hard Disk Drive Worksheet	232
Completing a Server Worksheet	233
Using Tools to Document an Investigation	234
FragView	234
Helpful Mobile Applications (Apps).....	235
Writing Reports	236
Time Zones and Daylight Saving Time (DST).....	236
Creating a Comprehensive Report	238
Using Expert Witnesses at Trial	242
The Expert Witness.....	242
The Goals of the Expert Witness	242
Preparing an Expert Witness for Trial.....	243
Summary	245
Key Terms	246
Assessment	246
Chapter 7: Admissibility of Digital Evidence	252
History and Structure of the United States Legal System	253
Origins of the U.S. Legal System.....	254
Overview of the U.S. Court System.....	254
In the Courtroom.....	259

Evidence Admissibility	262
Constitutional Law.	262
First Amendment.....	262
First Amendment and the Internet.....	263
Fourth Amendment.....	265
Fifth Amendment.....	279
Sixth Amendment.....	280
Congressional Legislation.....	281
CLOUD (Clarifying Lawful Overseas Use of Data) Act.....	288
Rules for Evidence Admissibility.....	288
Criminal Defense.....	293
California Consumer Privacy Act (CCPA).....	294
NYS DFS Rule 23 NYCRR 500.....	294
Canada Personal Information Protection and Electronic Documents Act (PIPEDA).....	295
When Computer Forensics Goes Wrong.	296
Pornography in the Classroom.....	296
Structure of the Legal System in the European Union (E.U.).	296
Origins of European Law.....	297
Structure of European Union Law.....	297
Privacy Legislation in Asia	303
China.....	304
India.....	304
Summary	305
Key Terms	306
Assessment	309
Chapter 8: Network Forensics and Incident Response	314
The Tools of the Trade.	315
Networking Devices	316

Proxy Servers.....	317
Web Servers.....	317
DHCP Servers.....	321
DHCP Logs.....	323
Hub.....	324
Switch.....	324
SMTP Servers.....	324
DNS Servers.....	326
The Hosts File.....	327
DNS Protocol.....	328
Internet Corporation for Assigned Names and Numbers (ICANN).....	328
Traceroute.....	328
Routers.....	328
IDS.....	338
Firewalls.....	339
Ports.....	340
Understanding the OSI Model.....	341
The Physical Layer.....	341
The Data Link Layer.....	342
The Network Layer.....	342
The Transport Layer.....	343
The Session Layer.....	344
The Presentation Layer.....	344
The Application Layer.....	345
Introduction to VoIP.....	346
Voice over Internet Protocol (VoIP).....	346
Disadvantages of VoIP.....	346
PBX (Private Branch Exchange).....	346

Session Initiation Protocol (SIP).....	348
STUN (Simple Traversal of UDP Through NATs (Network Address Translation)).....	348
Incident Response (IR)	348
STIX, TAXII, and Cybox	349
Advanced Persistent Threats	349
APT10	350
Cyber Kill Chain	350
Indicators of Compromise (IOC)	354
Investigating a Network Attack	357
Random Access Memory (RAM).....	357
AmCache	357
ShimCache.....	358
ShellBags.....	358
Volume Shadow Copy	358
Endpoint Detection and Response (EDR).....	359
Kibana.....	359
Log2Timeline/Plaso	359
SANS SIFT Workstation	360
Windows Registry	361
Summary	364
Key Terms	365
Assessment	367
Chapter 9: Mobile Forensics	372
The Cellular Network	374
Base Transceiver Station	374
Mobile Station.....	378
Cellular Network Types	383
SIM Card Forensics	385
Types of Evidence.....	388

Handset Specifications	389
Memory and Processing	389
Battery.....	390
Other Hardware.....	390
Mobile Operating Systems	391
Android OS	391
Symbian OS.....	400
BlackBerry 10.....	400
Windows Phone	400
Standard Operating Procedures for Handling Handset Evidence.	401
National Institute of Standards and Technology (NIST)	401
Handset Forensics.	406
Cellphone Forensics Tools.....	406
Logical Versus Physical Examination.....	408
Manual Cellphone Examinations	408
Flasher Box.....	409
Global Satellite Service Providers	410
Satellite Communication Services	410
Legal Considerations	410
National Crime Information Center (NCIC).....	411
Other Mobile Devices	413
Tablets.....	413
GPS Tracking	414
Documenting the Investigation.	415
Summary	416
Key Terms	416
Assessment	421

Chapter 10: Mobile App Investigations	426
Static Versus Dynamic Analysis	427
Static Analysis.....	427
Dynamic Analysis.....	431
Introduction to Debookee	433
Dating Apps	441
Tinder	442
Grindr	445
Rideshare Apps	450
Uber	451
Communication Apps	453
Skype	453
Summary	457
Key Terms	457
Assessment	458
Chapter 11: Photograph Forensics	460
National Center for Missing and Exploited Children (NCMEC)	462
Project VIC	463
Case Studies	463
Facebook Selfie	463
To Catch a Predator	463
Extortion.....	464
Understanding Digital Photography.	464
File Systems.....	464
Digital Photography Applications and Services.....	465
Examining Picture Files.	466
Exchangeable Image File Format (EXIF)	467
Evidence Admissibility	470

Federal Rules of Evidence (FRE).....	470
Analog vs. Digital Photographs.....	470
Case Studies	471
Worldwide Manhunt.....	471
NYPD Facial Recognition Unit.....	473
Summary	474
Key Terms	474
Assessment	475
Chapter 12: Mac Forensics	480
A Brief History	480
Macintosh.....	481
Mac mini with OS X Server.....	481
iPod.....	482
iPhone.....	483
iPad.....	485
iPad Pro.....	485
Apple Watch.....	485
Apple Wi-Fi Devices	487
Apple TV.....	487
AirPort Express.....	488
AirPort Extreme.....	488
AirPort Time Capsule.....	488
Macintosh File Systems	489
Hierarchical File System (HFS).....	489
HFS+.....	489
APFS.....	490
Forensic Examinations on a Mac.....	494
Epoch Time.....	496
DMG.....	498

PList Files.....	499
SQLite Databases	501
Email Files.....	501
Hibernation File.....	501
Macintosh Operating Systems	502
macOS Catalina.....	502
File Vault.....	503
Disk Utility	503
macOS Keychain	503
iCloud Keychain.....	504
Multiple Displays.....	504
Notifications	504
Tags.....	504
Safari.....	504
Target Disk Mode and Device Cloning.....	506
Apple Mobile Devices	507
iOS	508
Enterprise Deployment of Apple Devices	526
Battery.....	527
Performing a Mac Forensics Examination.	527
Case Studies	529
Find My iPhone.....	529
Wanted Hactivist.....	529
Michael Jackson	529
Stolen iPhone.....	529
Drug Bust.....	530
Murder Trial	530
Summary	531
Key Terms	531
Assessment	535

Chapter 13: Case Studies	538
Silk Road	538
Genesis of the Silk Road.....	539
Death Threat	542
Silk Road Takedown	542
The Takedown of Ulbricht	543
Ross Ulbricht Pre-trial.....	544
Ross Ulbricht on Trial.....	546
Laptop Evidence.....	546
Trial Verdict.....	549
Las Vegas Massacre	549
Zacharias Moussaoui	551
Background.....	551
Digital Evidence	552
Standby Counsel Objections	553
Prosecution Affidavit.....	554
Exhibits	554
BTK (Bind Torture Kill) Serial Killer	555
Profile of a Killer	555
Evidence	556
Cyberbullying	557
Federal Anti-harassment Legislation	557
State Anti-harassment Legislation.....	557
Warning Signs of Cyberbullying.....	557
What Is Cyberbullying?.....	558
Phoebe Prince.....	558
Ryan Halligan	559
Megan Meier	559
Tyler Clementi	559

Sports	561
Summary	563
Key Terms	563
Assessment	564
Assignment	570
Chapter 14: Internet of Things (IoT) Forensics and Emergent Technologies	572
5G	573
Wi-Fi 6	575
Wi-Fi Mesh Networks	576
Shodan	576
Mirai Botnet	577
Cryptocurrency Mining	577
Alexa	578
Micro-Chipping	579
Fitness Trackers	579
Apple Watch	581
Action Cameras	583
Police Safety	583
Police Vehicles	585
Vehicle Forensics	585
Low-Tech Solution for High-Tech Seizures	586
Summary	588
Key Terms	588
Assessment	590
Answer Key	594
Index	606

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
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.

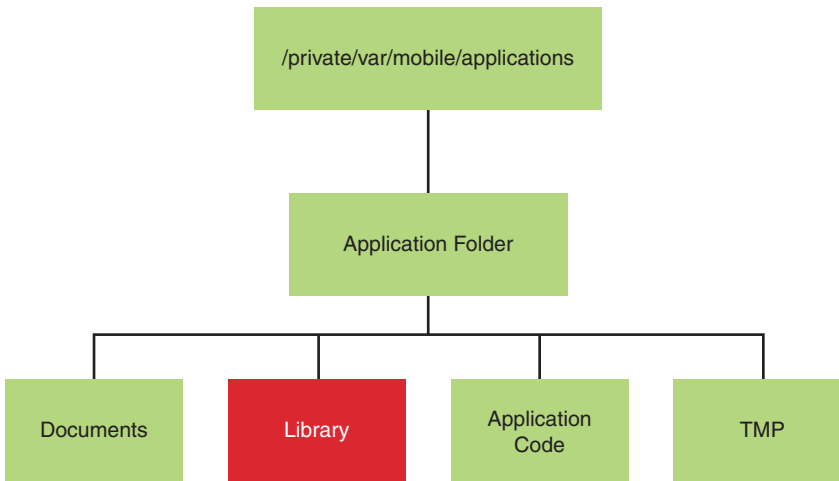


FIGURE 10.2 Application storage on iOS

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.

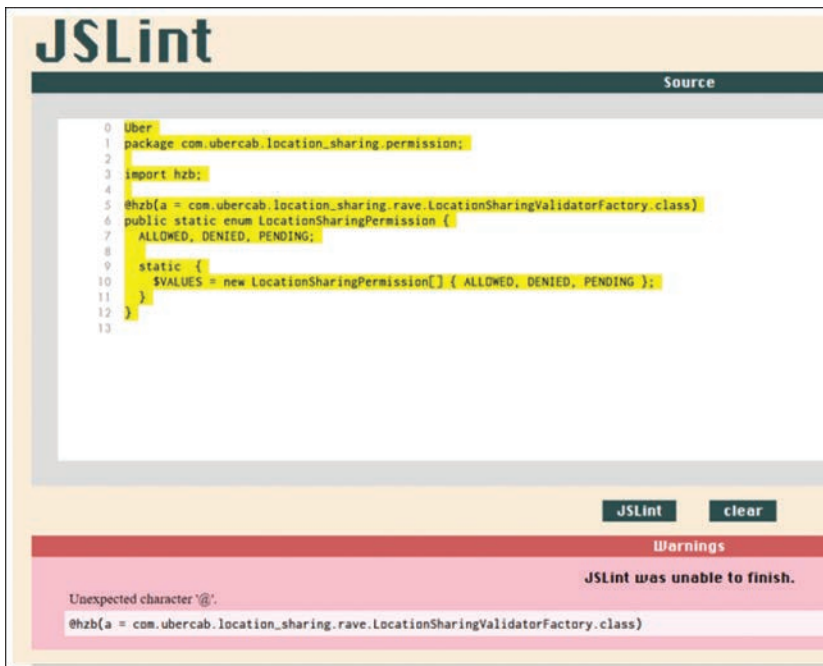


FIGURE 10.5 JSLint user interface

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.

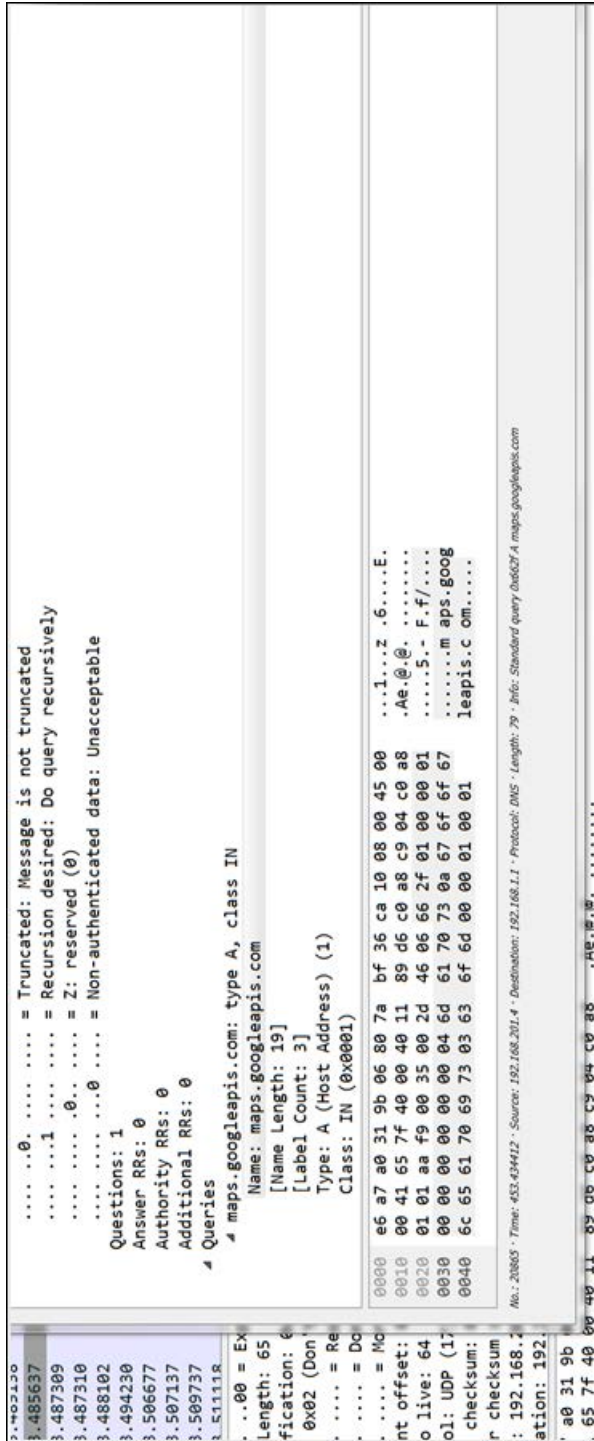


FIGURE 10.6 Google Maps API identified in a PCAP captured by Wireshark

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.



FIGURE 10.7 Debookee home page

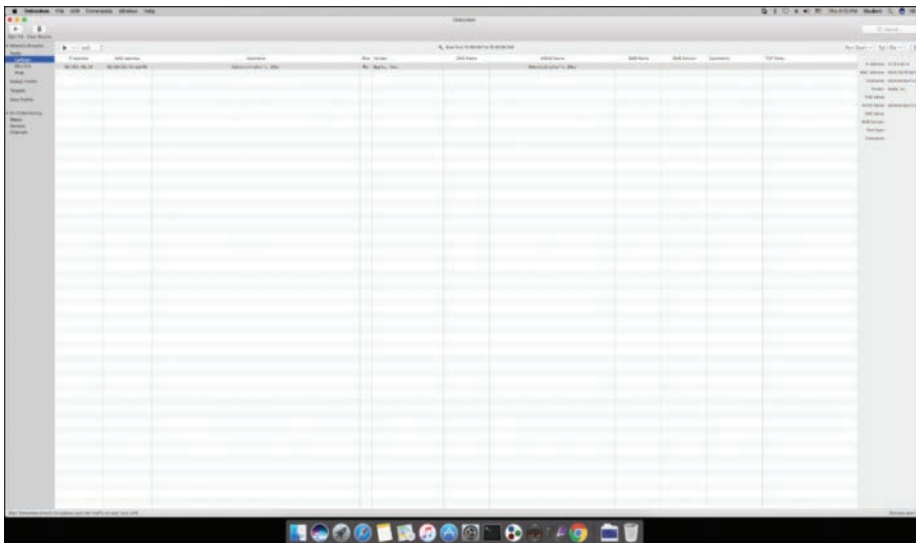


FIGURE 10.8 Debookee user interface

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.

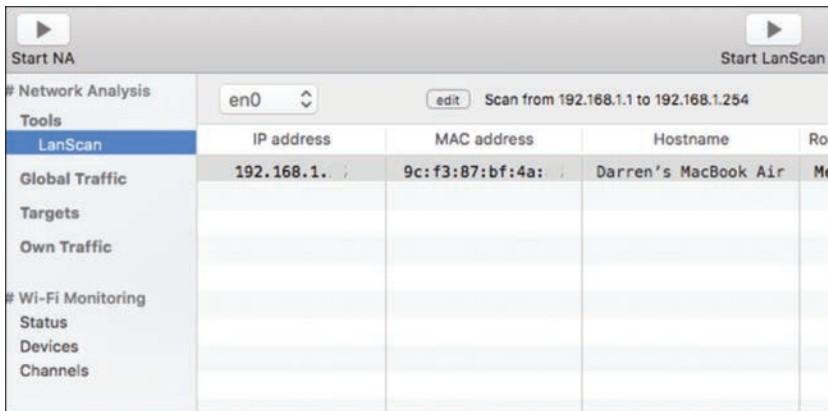


FIGURE 10.9 Debookee user interface with host computer information displayed

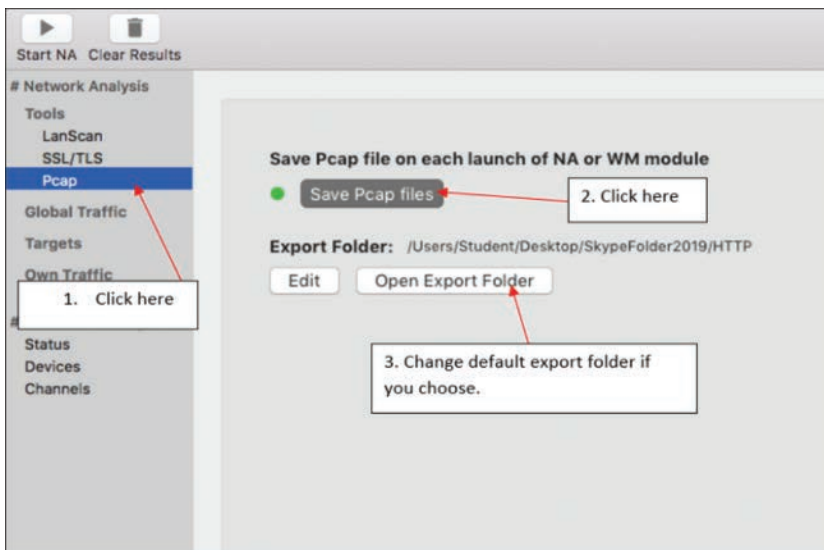


FIGURE 10.10 Save Pcap files option in Debookee

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.

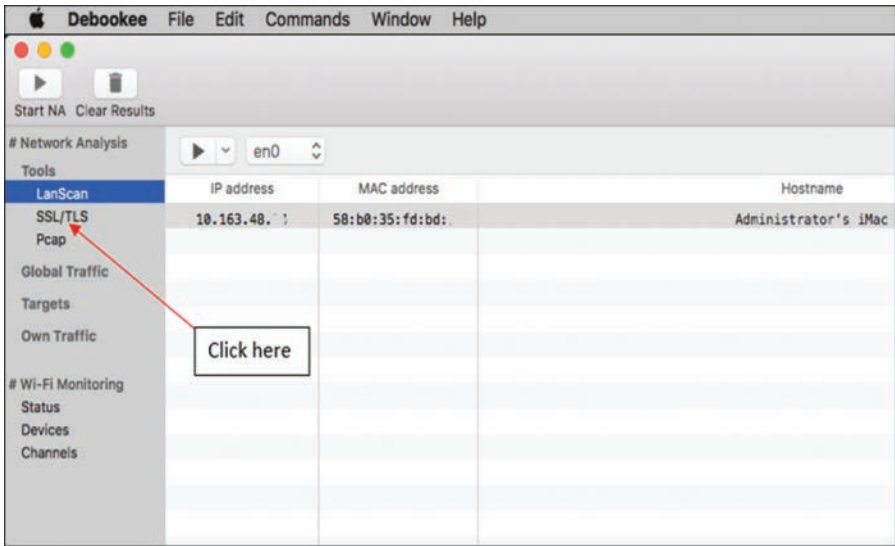


FIGURE 10.11 SSL/TLS decryption option in Debookee

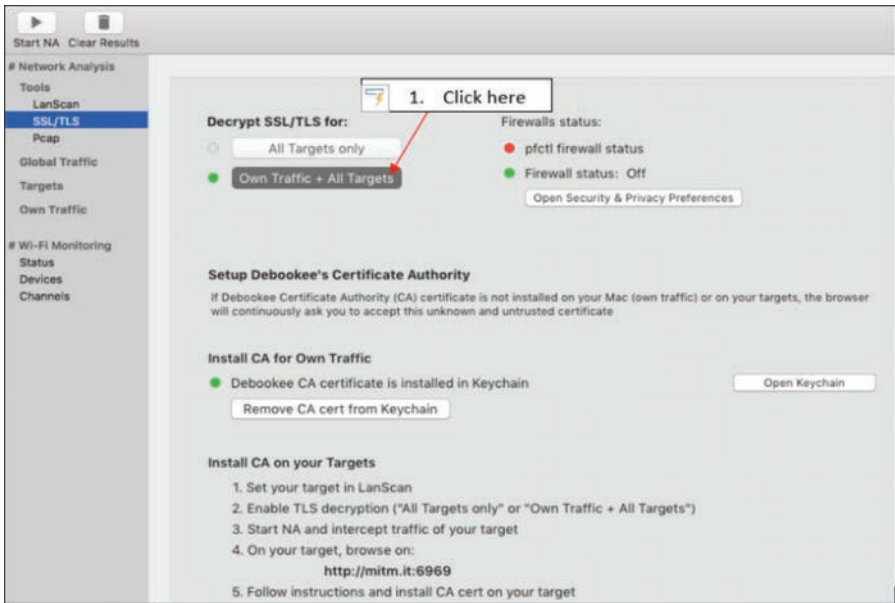


FIGURE 10.12 Decryption option in Debookee

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.

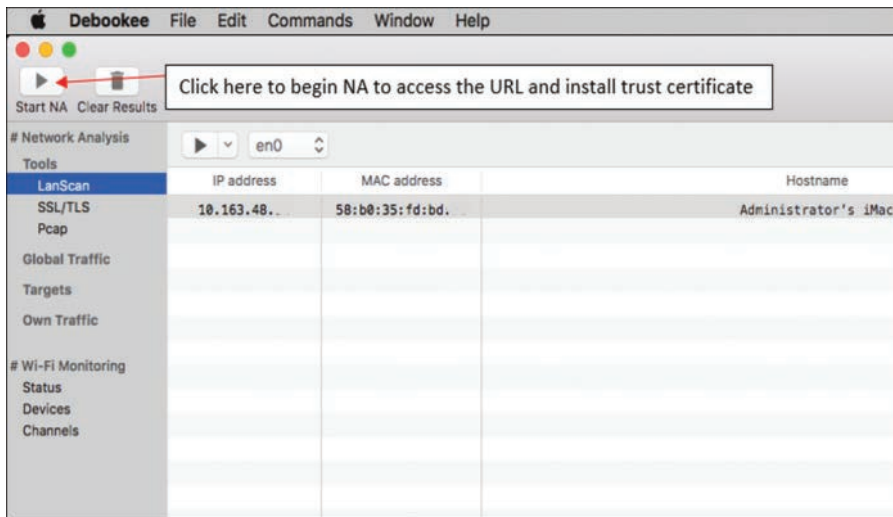


FIGURE 10.13 Start NA option in Debookee

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.robtext.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.

# Network Analysis	Time	Hostname	IP
Tools	16:19:12	lh3.googleusercontent.com	172.217.12.129
LanScan	16:19:12	www.gstatic.com	172.217.12.163
SSL/TLS	16:19:12	lh5.googleusercontent.com	172.217.12.129
Pcap	16:19:12	encrypted-tbn3.gstatic.com	172.217.12.206
Global Traffic	16:19:13	www.facebook.com	31.13.71.36
Targets	16:19:13	static.xx.fbcdn.net	31.13.71.7
Own Traffic	16:19:13	scontent-lga3-1.xx.fbcdn.net	31.13.71.7
DNS	16:19:14	googleads.g.doubleclick.net	172.217.12.194
HTTP	16:19:37	www.instagram.com	31.13.71.174
	16:19:39	connect.facebook.net	31.13.71.7
# Wi-Fi Monitoring	16:19:41	clients1.google.com	172.217.6.206
Status	16:19:41	graph.instagram.com	31.13.71.52

FIGURE 10.14 DNS connections captured

# Network Analysis	Time	URL	Method	Status
Tools	13:05:06	http://v16.muscdn.com/d2f41c7f2580685ec5398488659d1ac2/Se2791bd/video/tos/maliva/tos-maliva-v-0068/3ff13...	GET	206
LanScan	13:05:06	http://v16.muscdn.com/fcc95ee17db0475430c9c5193291a5bc/Se2791ba/video/tos/useast2a/tos-useast2a-ve-0068...	GET	206
	13:05:07	http://v16.muscdn.com/f0456678af5159bb248719715a689e79/Se2791b7/ideo/tos/maliva/tos-maliva-v-0068/1ddc...	GET	206
	13:05:08	http://v16.muscdn.com/df98407dbd3d4afd1d80bb7f8c1129a0/Se2791bb/video/tos/maliva/tos-maliva-v-0068/ff341...	GET	206
Targets	13:05:14	http://p16.muscdn.com/img/tos-maliva-p-0068/c0ac737d1b2d4beb8aacdf1db04d110_1574604937-tpiv-tiktok-shrL...	GET	200
192.168.2.3	13:05:14	http://p16.muscdn.com/img/tos-maliva-p-0068/bdcca8b537744038987158e027fdb33_1574555997-tpiv-tiktok-shrL...	GET	200
HTTP	13:05:14	http://p16.muscdn.com/img/tos-maliva-p-0068/3c816ae3779943c0a3c6d19da0c8204c_1574020710-tpiv-tiktok-shrL...	GET	200
DNS	13:05:14	http://p16.muscdn.com/img/tos-maliva-p-0068/0f94d0e67b6e4f0b9ed338670eca4e6b_1577566314-tpiv-tiktok-shrL...	GET	200
Own Traffic	13:05:14	https://p16.muscdn.com/img/musically-maliva-obj/e6252bedf8f919da07b42c203db009c-tpiv-tiktok-shrink:750:...	GET	200


```

Request
GET http://p16.muscdn.com/img/tos-maliva-p-0068/9d5bfecce020437a9e978b0e11f921a3_1579305427-tpiv-tiktok-shrink:248:330.webp?i1og=
HTTP/1.1

Response
Host: p16.muscdn.com
Connection: keep-alive
Accept-Encoding: gzip, deflate
User-Agent: TikTok/166800 CFNetwork/811.5.4 Darwin/16.7.0
Accept-Language: en-us
Accept: image/webp, image/*;q=0.8
  
```

FIGURE 10.15 Decrypted TikTok packet (pcap)

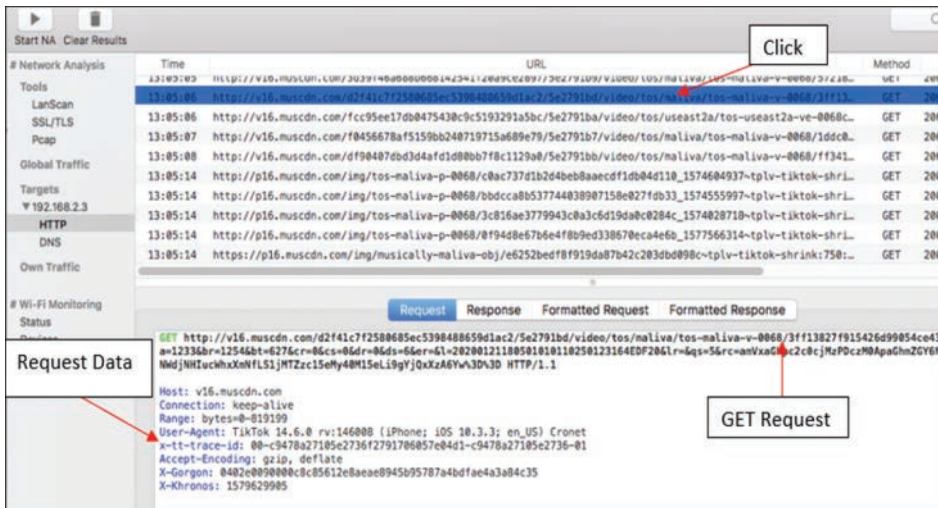


FIGURE 10.16 GET request data displayed

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.

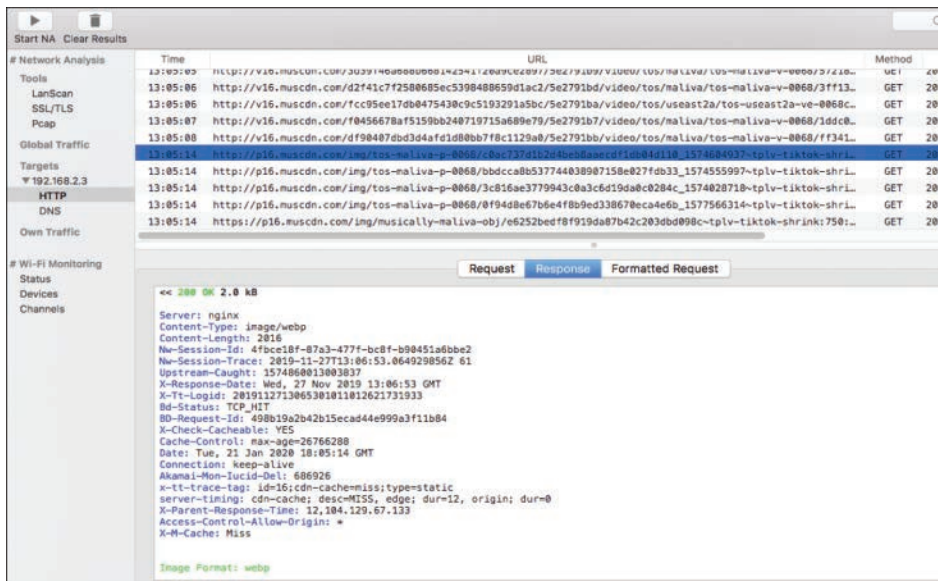


FIGURE 10.17 Response results

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.