PassQueen
a.k.a dumpsqueen.com

GIAC Advanced Smartphone Forensics

GIAC GASF

Version Demo

Total Demo Questions: 10

Total Premium Questions: 103

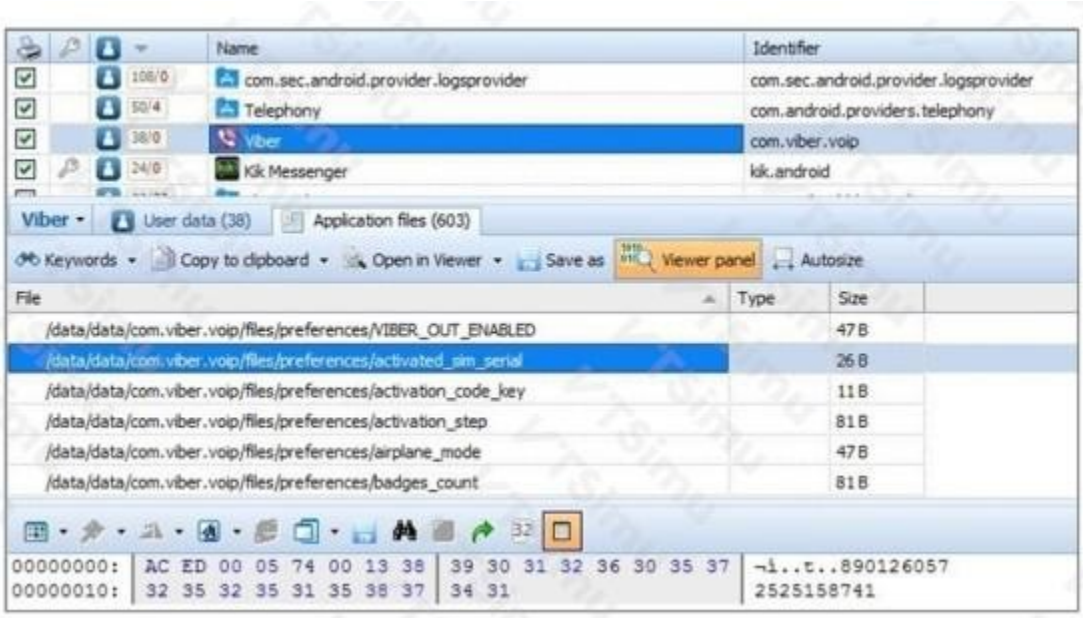
Buy Premium PDF

https://passqueen.com

support@passqueen.com

QUESTION NO: 1

Review the information contained within the Viber application running on an Android device. Which of the following can be determined?



A. A message containing the string 8901260572525158741 was sent using the Viber application.

B. The Viber account used to send/receive messages can be tied to the user in possession of the SIM card with an IMSI of 8901260572525158741

C. The user account for Viber is 8901260572525158741

D. The Viber account used to send/receive messages can be tied to the user in possession of the SIM card with an ICCID of 8901260572525158741

ANSWER: D

Explanation:

The Viber account used to send/receive messages can be tied to the user in possession of the SIM card with an ICCID of 8901260572525158741. The displayed value is a SIM-card serial identifier: an Integrated Circuit Card Identifier, commonly called an ICCID. ICCIDs are generally long numeric identifiers and frequently begin with digits identifying the telecommunications industry and issuing network. The value shown has the form and length expected for an ICCID, allowing an examiner to associate the Viber application's recorded SIM information with the particular physical SIM card installed in the device when the information was captured.

This is useful forensic attribution evidence because subscriber or carrier records may link that ICCID to the account holder or person assigned the SIM during a relevant time period. The conclusion should be phrased carefully as a tie to the person in possession of, or subscriber associated with, that SIM, rather than an absolute identity claim. Android's telephony framework describes the SIM serial number returned by its relevant API as the ICCID, supporting interpretation of this artifact as a SIM-card identifier. See the Android TelephonyManager documentation and the ETSI smart-card specification for ICCID-related technical context.

QUESTION NO: 2

Which Android partition commonly contains the boot image that includes the kernel and ramdisk used during device startup?

A. boot

B. userdata

C. cache

D. metadata

ANSWER: A

Explanation:

boot is correct. The Android boot partition commonly stores the boot image used to start the operating system. Depending on the device generation and partition scheme, the image contains the Linux kernel and ramdisk or related boot components needed early in the boot process.

The boot partition is distinct from the user-data partition, which contains user and application data, and from partitions such as system or vendor that contain operating-system components. An examiner must identify the actual partition layout for the particular device because modern Android devices can use dynamic partitions and may separate boot-related components across additional partitions. Android documents boot images and partition architecture in the Android boot partitions documentation.

QUESTION NO: 3

When dealing with mobile devices and flash memory, and the fact that data in memory constantly changes

even when the device is simply powered on. It is best practice to:

A. Only acquire from devices in an OFF state

B. Document those changes that were made to the device during the forensic process

C. Always use a write-blocker when dealing with mobile devices

D. Always remove the battery from a device before acquisition

ANSWER: B

Explanation:

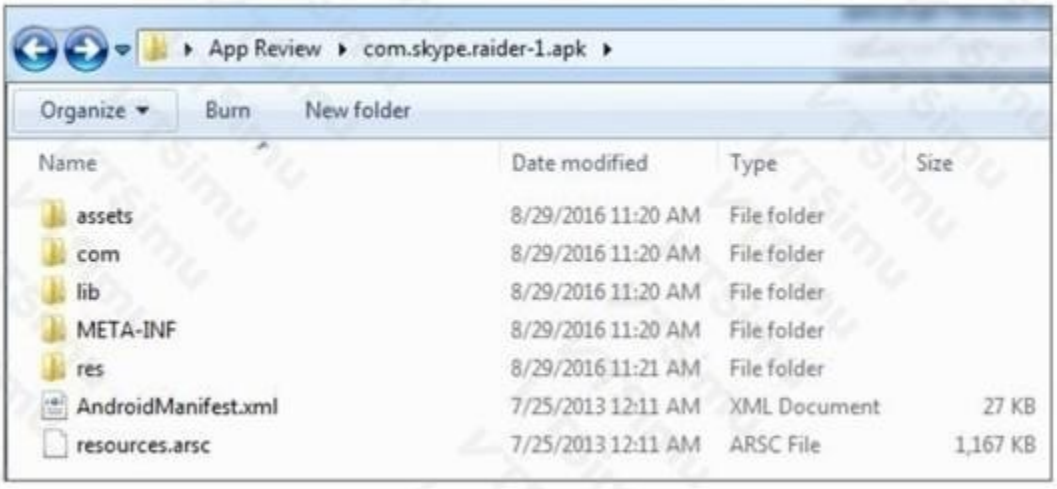
Document those changes that were made to the device during the forensic process is correct because mobile devices are inherently dynamic evidence sources. When powered on, the operating system, applications, radios, synchronization services, and background processes can alter logs, memory, databases, timestamps, network state, and other artifacts. Flash storage also uses controller-managed techniques such as wear leveling and garbage collection, which make traditional assumptions about static media and conventional write blocking unreliable for many mobile acquisition methods.

Sound mobile-device forensic practice is therefore to minimize interaction with the device, use an acquisition approach appropriate to its state and security configuration, and maintain detailed contemporaneous records of every action taken. Documentation should include the device condition when received, power and lock state, isolation measures, tools and versions used, commands or settings applied, acquisition times, observed changes, and chain-of-custody details. This record supports repeatability, transparency, and the ability to explain any alteration that may have occurred while preserving and collecting evidence. NIST's mobile-device forensic guidance emphasizes documenting procedures and maintaining evidence integrity throughout examination; see NIST SP 800-101 Rev. 1. Additional practical guidance on handling and preserving mobile devices is available from SWGDE.

QUESTION NO: 4

Examine the unpacked Android application below. Which important file, resident in most

Android applications, is missing?



A. dalvik-cache

B. classes.dex

C. com.skype.raider-1.apk

D. classes-dex2jar.jar

ANSWER: B

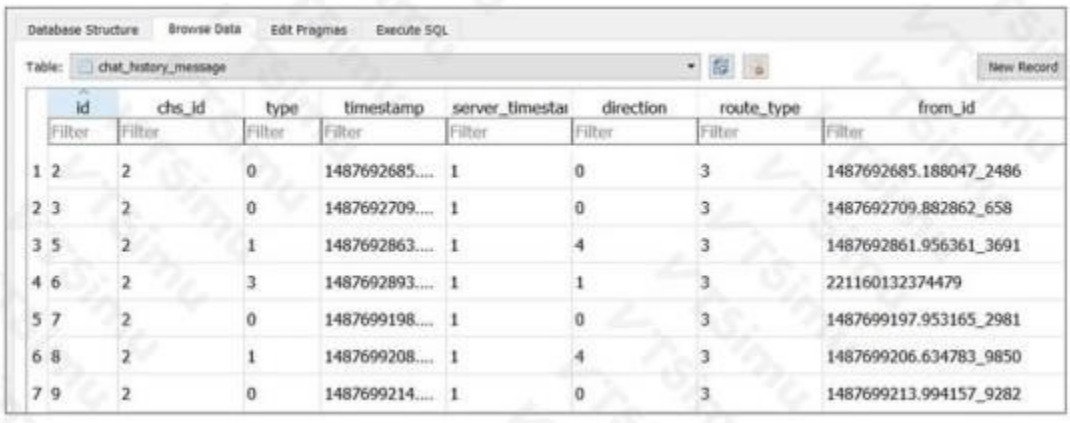
Explanation:

`classes.dex` is the important missing file. In a conventional Android application package, application code is compiled into Dalvik Executable (DEX) bytecode and stored in `classes.dex` at the root of the APK. Android Runtime uses this DEX content to load and execute the app's Java or Kotlin-derived code. When an investigator unpacks an APK, locating `classes.dex` is therefore a fundamental step before performing static-code review, decompilation, string analysis, or examination of application behavior.

An APK is a ZIP-based package that normally contains the manifest, resources, native libraries when applicable, and one or more DEX files. Modern multidex applications may additionally contain files such as `classes2.dex`, but the primary DEX file remains named `classes.dex`. Its absence from an otherwise normally structured unpacked application is significant because it indicates that the primary executable bytecode is not present in the displayed contents. Android's APK documentation identifies DEX files as the compiled code component of an APK; see Android Developers: App manifest overview and Android Open Source Project: Dalvik bytecode.

QUESTION NO: 5

Review the message database below.



What can be determined based on the information provided?

A. There are no deleted messages in this database

B. The interrupted sequence in the id column could indicate deleted messages

C. The messages in this table are “outgoing”

D. The messages contain attachments

E. The messages in this table are only “incoming”

ANSWER: B

Explanation:

The interrupted sequence in the id column could indicate deleted messages is the supported conclusion. In a message-database table, an identifier column commonly holds an integer primary-key value that increases as records are created. If the displayed records show a gap in that sequence, it is consistent with one or more records having existed between the surrounding identifiers and subsequently being deleted. This is a valuable forensic lead because deleted SQLite records may remain recoverable in unallocated space, freelist pages, write-ahead log files, or rollback journals, depending on database configuration and subsequent activity.

A gap is appropriately described as an indicator rather than definitive proof: database identifiers are not necessarily gapless, and application behavior or transaction handling can also affect numbering. Nevertheless, the observed interruption identifies a reasonable basis for further examination of SQLite artifacts and deleted-record recovery. SQLite documents that automatically assigned row identifiers are not guaranteed to be sequential without gaps, while its file-format documentation describes freelist structures that can retain pages no longer in active use. See SQLite AUTOINCREMENT documentation and SQLite database file-format documentation.

QUESTION NO: 6

Which of the following files contains details regarding the encryption state of an iTunes backup file?

A. Keychain-backup.plist

B. Manifest.mbdb

C. Manifest.plist

D. Status.plist

ANSWER: C

Explanation:

Manifest.plist is the backup-level property-list file that records whether an iTunes (now Finder/Apple Devices) backup is encrypted. In legacy iOS backup structures, investigators can examine this plist for the `IsEncrypted` key and its Boolean value before attempting to parse or decrypt backup content. This is operationally important because an encrypted local backup requires the backup password and uses a different protection scheme for the backup data; determining the encryption state early helps establish the appropriate forensic workflow and whether password recovery or decryption is necessary.

The file serves as metadata for the backup as a whole rather than as an inventory of individual backed-up files. Apple describes encrypted local backups as protected with a user-defined password and notes that encryption can include additional sensitive data, reinforcing why identifying that state is essential during acquisition and analysis. Apple’s platform-security documentation also describes the cryptographic protection applied to encrypted backups. See Apple Support: About encrypted backups and Apple Platform Security: Secure backup.

QUESTION NO: 7

What will happen when a user creates an encrypted backup of their personal iPhone on their work computer then later performs a backup to their personal computer at home?

A. The user will not be able to create the backup until they have deleted the original on their work computer

B. The user will be required to enter a new backup password on their personal computer

C. The user will be required to enter their backup password to start the backup process

D. The user will not need to enter a password on the personal computer

ANSWER: C

Explanation:

The user will be required to enter their backup password to start the backup process. Local iPhone backups protected with backup encryption use a password to protect the backup's encryption keys and the sensitive data included in the backup. When the same device is subsequently backed up through a different computer, that computer does not inherently possess or establish a separate replacement password merely because it is a new backup location. The existing encrypted-backup protection must be unlocked by supplying the backup password before the encrypted backup operation can proceed. This preserves continuity of the encryption state and ensures that only someone who knows the password can create and later access protected local backup data. Apple notes that encrypted local backups require a password and that the password is not recoverable by Apple; if it is forgotten, the user must reset the device's settings before creating a new encrypted backup. This demonstrates the password's role as a required control for the encrypted-backup process rather than a computer-specific convenience. See Apple's guidance on encrypted backups for iPhone, iPad, and iPod touch and the Apple Platform Security discussion of backup encryption.

QUESTION NO: 8

What is the essential piece of information is most often required in order to decrypt the contents of BlackBerry OS 10 handsets?

A. BlackBerry Blend username/pin

B. BlackBerry Balance username/password

C. BlackBerry Link ID/password

D. BBM pin

E. BlackBerry device password

ANSWER: E

Explanation:

The **BlackBerry device password** is the essential item most often needed to decrypt protected user data on a BlackBerry 10 handset. BlackBerry 10 uses the password supplied on the device as part of its cryptographic protection for data-at-rest. Consequently, a forensic examiner who has acquired handset data may still be unable to access encrypted content unless the device password is known, recovered through a lawful process, or the device is already in an unlocked state that permits an appropriate acquisition. This is a device-level credential, not merely an identifier for a messaging, synchronization, or companion service.

Forensic handling should preserve the handset's current power and lock state, document how the password was obtained, and avoid actions that could trigger locking, key eviction, or data loss. The password should be treated as sensitive case evidence and handled under the examination authority and evidence-management procedures. NIST's mobile-device-forensics guidance identifies encryption and authentication credentials as central considerations when acquiring and examining mobile-device evidence: NIST SP 800-101 Rev. 1. BlackBerry's security material also describes the platform's use of device-password-based protections for data stored on BlackBerry 10 devices: BlackBerry 10 Security Technical Overview.

QUESTION NO: 9

What is being shown in the image below?

A. An outgoing call that was not answered

B. A call that was answered but immediately hung up

C. A missed Skype message on an android device

D. A call that was answered and lasted 5 seconds

ANSWER: A

Explanation:

An outgoing call that was not answered is correct because the call-log indicator shown identifies the event as an outbound dialing attempt, while the associated call information indicates that no connected conversation occurred. In mobile call records, call direction and call outcome are separate forensic facts: an outgoing entry establishes that the device attempted to place the call, and a zero-duration or non-connected result supports the conclusion that the remote party did not answer. This is materially different from a completed call, for which the call-log record would reflect a connected duration.

Android exposes call-history entries through its CallLog provider. Its documented call-type values include outgoing calls, and its call-duration field records the duration in seconds; these are the key metadata elements used to interpret this kind of artifact during examination. See the Android CallLog.Calls reference and the Android Open Source Project's CallLog source documentation. In a forensic report, this artifact should be described as an attempted outgoing call with no evidence of an answered, connected conversation.

QUESTION NO: 10

Which of the following is one potential risk of using the ALWAYS OFF rule for handling cell phones?

A. Overwriting data

B. Engaging password or PIN protection mechanism

C. Destruction of call logs and cell tower information

D. Improper handling by the user

ANSWER: B

Explanation:

Engaging password or PIN protection mechanism is a potential risk of an ALWAYS OFF handling approach because powering down a mobile device commonly causes its screen-lock credentials to be required again when the device is restarted. This can immediately restrict examiner access to the device and, on modern devices, may leave protected data unavailable until the correct PIN, password, or biometric fallback credential is supplied. The concern is especially significant where encryption keys are available only after the device has been unlocked following a reboot. Consequently, an examiner must document the device's state and make a deliberate preservation decision based on the circumstances, rather than

assuming that powering it off is always harmless. NIST's mobile-device forensic guidance identifies the device's power state and lock state as key considerations during preservation and acquisition, since those states affect available evidence and access methods. See NIST SP 800-101 Rev. 1, Guidelines on Mobile Device Forensics. NIST also explains that mobile-device security controls include device locking and encryption protections that govern access to stored data: Guidelines for Managing the Security of Mobile Devices in the Enterprise.