DC3 Technical Advisory Signal Chat App Decryption

Department of Defense Cyber Crime Center (DC3) sent this bulletin at 05/26/2021 03:21 PM EDT

TECHNICAL-ADVISORY-2021-005
SIGNAL CHAT APPLICATION DECRYPTION

SUMMARY

Signal is a messaging app that supports one-on-one chats, group chats, and file sharing. On Android and iOS, the app must be registered using a telephone number, and messages can be synchronized with the Signal desktop application on Windows, macOS, and Linux. On all platforms, the Signal app stores data in an encrypted SQLite database. Generally, the encryption key can be obtained and the database can be decrypted. In addition to encrypted data, some remnants of Signal usage may be retained in unencrypted form.

TECHNICAL CHALLENGE

The primary challenge is to decrypt the database that Signal uses to store message content. For mobile devices, there is the added technical challenge of recovering the encryption key from protected containers.

DETAILS

Decrypting the Signal “db.sqlite” SQLite database on Windows, macOS, and Linux is straightforward. The encryption key is stored in plaintext in a file named config.json, which can used with DB Browser for SQLCipher to decrypt the file and view the messages in the messages table (see Figure below). This table contains message text, timestamp, whether it was ingoing or outgoing, and a column called json that contains a JSON representation of any file attachments contained in the message. Another table of interest is the conversations table which contains the names of participants in the chat and the type of chat such as private, public, or group chat.

Signal for iOS stores its encryption key in the iOS keychain, which certain commercial tools can extract in its entirety. The extracted encryption key, which is stored in the keychain in Base64 format, can be used to decrypt the database /var/mobile/Containers/Shared/AppGroup/[UUID]/grdb/signal.sqlite. In addition, attachments sent and received can be stored in /var/mobile/Containers/Shared/AppGroup/[UUID]/Attachments, and snapshots of Signal messaging screens, with visible text can be stored in /var/mobile/Containers/Data/Application/[UUID]/Library/SplashBoard/Snapshots.

Signal for Android stores its encrypted SQLCipher file at /data/data/org.thoughtcrime.securesms/databases/signal.db. This file is encrypted using a key from the Android Keystore, which presents a challenge from a forensic perspective.

SYNOPSIS

Decryption of the Signal app on different platforms ranges from straightforward on desktops, to challenging on iOS and Android, particularly when hardware security protects the keys. Once the key is recovered, the SQLite database is decrypted and all stored Signal content can be examined.

Contributors Acknowledgements

Technical Advisories are a five-minute read to raise awareness among investigators, forensic practitioners, attorneys, and judges about an emerging trend in Digital/Multimedia Forensics.

- Eric Robertson (DC3/TSD Developer) and Kevin Westerman (DC3/CFL Examiner)
- DC3 Editorial Board (Curation, Review, & Publication)

For additional information and forensic analysis capabilities please reach out to DC3 at hub@dc3.mil.

Relevant Forensic Artifacts

DC3 is building a crowdsourced catalog of forensic artifacts, created by practitioners for practitioners, to curate expertise across the digital forensic community, making it available as a user friendly, online knowledge management platform. In this context, a digital artifact is defined as *a singular unit of interpretable data that can be extracted from a given data source*.

Click to edit this heading.

Container	OS	Artifact
C:/Users/[USER]/AppData/Roaming/Signal/config.json	Windows	Decryption Key
C:/Users/[USER]/AppData/Roaming/Signal/sql/db.sqlite	Windows	SQLCipher File
/Users/[USER]/Library/Application Support/Signal/config.json	macOS	Decryption Key
/Users/[USER]/Library/Application Support/Signal/sql/db.sqlite	macOS	SQLCipher File
/home/[USER]/.config/Signal/config.json	Linux	Decryption Key
/home/[USER]/.config/Signal/sql/db.sqlite	Linux	SQLCipher File
/var/mobile/Containers/Shared/AppGroup/UUID/grdb/signal.sqlite	iOS	SQLCipher File
GRDBDatabaseCipherKeySpec for GRDBKeyChainService	iOS	Signal iOS Keychain (Base64 encoded)
/var/mobile/Containers/Shared/AppGroup/UUID/Attachments	iOS	Message attachments (sent and received)
/var/mobile/Containers/Data/Application/UUID/Library/SplashBoard/Snapshots	iOS	Screenshots of the messaging screens
/data/data/org.thoughtcrime.securesms/databases/signal.db	Android	SQLCipher File
Android Keystore	Android	Signal encryption key
org.thoughtcrime.securesms\shared_prefs\org.thoughtcrime.securesms_preferences.xml	Android	Telephone number registered with Signal account
org.thoughtcrime.securesms\app_parts\	Android	Message attachments (encrypted)

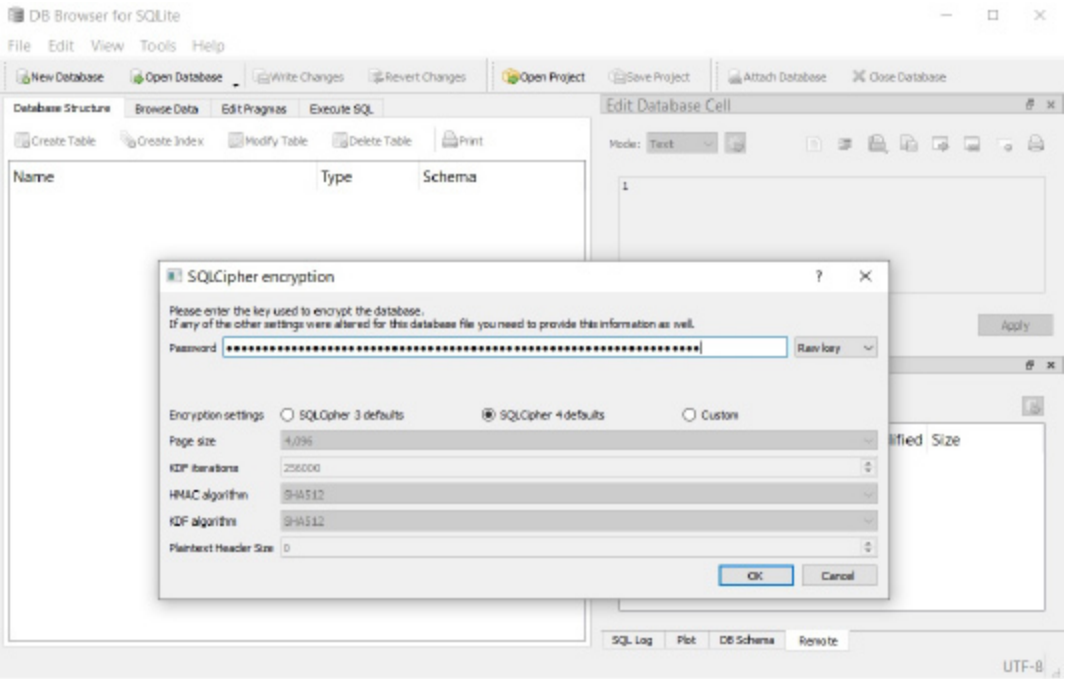

Figure: Input raw key from config.json to decrypt the SQLCipher encrypted file