About

Smoke is an Android communications research project. The software is composed of a single multi-threaded application. A companion application, SmokeStack, provides mobile server services.

Software sources are available at https://github.com/textbrowser/smoke and https://github.com/textbrowser/smokestack.

Android

Smoke has been successfully tested on Android 4.4, 5.0, 6.0, and 7.0. Android 4.4 and 5.0 are not officially supported.

According to https://developer.android.com/about/dashboards/index.html, Smoke supports 89.1% of all Android versions.

Database Containers

Most of the database fields contain authentically-encrypted values. Some fields contain keyed digests, including keyed digests of binary values.

Discovery via Cryptography

Cryptographic discovery is a mechanism which allows servers to lighten the computational and data responsibilities of mobile devices.

Shortly after a Smoke instance connects to a SmokeStack service, the Smoke instance shares some non-private material. The material allows a SmokeStack server to transfer messages to their correct destinations. SmokeStack instances routinely distribute gathered non-expired material to other SmokeStack services, thus creating a network of cooperative SmokeStacks. Cryptographic Discovery assumes a trustworthy network.

To mitigate replay attacks, Smoke offers SmokeStack instances random identity streams during message-retrieval requests. The identity streams self-expire.

Inflate

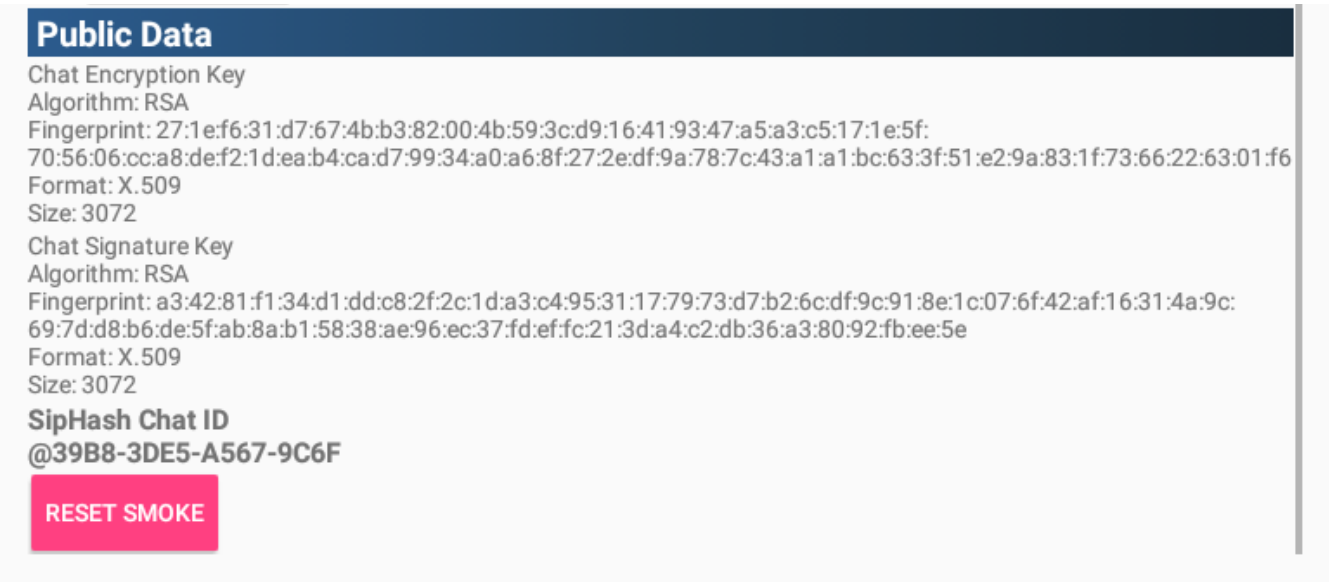
Smoke expands text-messaging data to 8192 bytes. If the provided data exceeds 8192 bytes, Smoke expands the provided data by 1024 + mod(data length, 2) bytes.

Ozone Address

An Ozone address is a pseudo-private string which identifies a virtual entity. Smoke and SmokeStack utilize Ozones as a means of retrieving and storing offline messages. Smoke supports one Ozone while SmokeStack supports infinitely many. Ozone addresses must be shared manually.

SipHash Identities

Exchanging public key pairs is often an involved process. Smoke implements the pseudo-random function SipHash so as to simplify the process. The SipHash function generates outputs of 8 bytes (16 characters hexadecimal). These short strings are easily memorized and/or distributed via other communications applications.



The transport keys which are generated from SipHash identities may be used for exchanging public-key data via the Echo Public Key Share (EPKS) protocol.

It is impossible to avoid collisions as there are infinitely-many inputs and a limited number of outputs.

TCP, UDP Protocols

Smoke supports both the TCP and UDP network protocols. Multicast and unicast UDP varieties are provided. Multiple clients may be defined via Settings. A limit on the number of clients is not imposed. When defining neighbors, one may define SmokeStack and/or Spot-On neighbors. SmokeStack, the companion application of Smoke, offers mobile server services as well as message storage.