Preliminary security report for Vanish

MLC Consulting

2017 September 2

Services Performed

The source was preliminarily analyzed to determine compilation and usage characteristics. A local testing environment was set up with dummy keys and internet addresses. The back-end (server) was pen-tested (mostly via fuzzing) without any further knowledge of source. The front-end (chat client) and back-end was tested in concert, with packet capture and subsequent analysis to determine any leaks of unencrypted information.

Then, a cursory source code analysis was performed to (1) determine promising targets for pen-testing and (2) provide preliminary suggestions. Further pen-testing informed by this analysis was performed.

Fuzzing report

Server

Random fuzzing with stock tools is insufficient to recover secret information.

After a source review, targeted fuzzing was similarly unsuccessful in recovering secret information.

In this case, secret information means plaintext (binary), hex-encoded, or base64-encoded data at the packet level.

Notably, metadata exfiltration was not analyzed. This requires a protocol review to determine which metadata is intended to be transmitted as-is over wire.

Crashes and hangs were observed on multiple fuzzed requests.

iOS

Similarly, fuzzing the iOS client over network is insufficient to recover secret information. Same caveat about metadata.

Crashes and hangs were observed on multiple fuzzed requests.

Packet analysis

No plaintext messages were intercepted over the wire during normal operation of the application. No base64 or hex-encoded messages were intercepted.

Further pentesting

Informed by source code analysis, more advanced fuzzing (protocol-consistent) and connection failure modes were tested on client and server. No message data (plain or encoded) was recovered.