KUDELSKI SECURITY

SILENTDATA SECURITY ASSESSMENT

Applied Blockchain have designed a service, SILENTDATA, to make assertions about data without exposing the data itself. Applied Blockchain requested Kudelski Security to perform an assessment to independently validate the claims and test the security of the platform.

Applied Blockchain has authored a whitepaper which elucidates the theory and background of the SILENTDATA platform. In this whitepaper they make a number of claims about the security of the product. Kudelski Security performed an assessment of the architecture and source code, as well as a penetration test on the deployed environment. This assesment was based on a in-development version of the SilentData platform and focuses on its first service implementation, Plaid, a banking intermediate service. Below we list the claims and provide our professional opinion.

SILENTDATA

Services performed

- Source code audit (Nov 2020, re-review Jan 2021)
- Architecture review (Nov 2020)
- Penetration test (Jan-Feb 2021, re-review Apr 2021)

Claim	Kudelski Security comments
Claims about the SILENTDATA enclave and proofs	
1. If the statement is true a verifier can be convinced of this fact by the SILENTDATA proof (“Completeness”)	The source code as audited will produce the correct attestation, assuming there is no fault with the integrated system (ie Plaid in the examined case) It should also be noted that Applied Blockchain personnel cannot influence the result as it is guaranteed by the enclave.
2. No SILENTDATA proofs can be created for false statements, eg. a minimum balance proof proving a minimum balance greater than the sum of the account balances (“Soundness”)	The source code as audited relies on the integrated Plaid service to produce correct attestations. It is not possible from the SILENTDATA source code to create a false statement, unless the logic within the Plaid service fails.

Claim	Kudelski Security comments
3. If a workflow involves requests to external web services, these are made from inside the enclave only, there is no leakage of any information about the content of those communications	The cryptography primitives are used correctly and there is no leakage of sensitive information (e.g. tokens) outside of the Enclave. As the Enclave communicates directly with Plaid using HTTPS/TLS and with the Client using the encrypted channel, the SILENTDATA server cannot directly access the Link, Public or Access tokens, nor the user's bank account information or credentials.
4. In all connections the identities of external web services are checked using TLS certificates. The enclave "knows" for a fact which server it connected to	The enclave embeds the Plaid public certificate and enforces certificate pinning when establishing the HTTPS connection.
5. Both the communication between the enclave and external web services and the communication between a suitable user interface code are secure. Neither authentication information (to make requests on the user's behalf outside the enclave) nor any information that is not part of the plaintext information in the proof anyway can be extracted from those communications by SILENTDATA or anybody else. Therefore, for a correct and secure execution of the functionality provided by the SILENTDATA enclave, other parts of the SILENTDATA service, in particular the web application, do not need to be trusted	The user's trust in the confidentiality and authenticity of his data depends on the trust in the Enclave's security, the Client's security and the encryption keys being used. As correctly stated by Applied Blockchain, ultimately the user must trust SILENTDATA for the correct Client behavior, regardless of the strong security features of the Enclave. It should be noted that SILENTDATA could not access information without knowledge of the end user.

Claim	Kudelski Security comments
6. Data subjects interacting with the enclave can be convinced they are communicating with a secure Intel SGX enclave	The Client frontend communicates with the Enclave through a secure channel established as follows: 1. The Client requests a public RSA3072 key from the Enclave. 2. This key is signed with the Enclave's remote attestation quote report and is verified by the Client using a certificate chain with the Intel IAS certificate. 3. The Client generates and returns an AES128 session key encrypted with the Enclave's public key. 4. This symmetric key is used by the Enclave to report back to the Client. The remote attestation performed by the client with Intel IAS guarantees that the signed quote received from the enclave is indeed produced inside a genuine Intel SGX enclave, and that, given an authorized MRENCLAVE, the channel is secure.
7. A SILENTDATA proof can be verified (by a verifier or anybody else) offline at a later time with no connection to the SILENTDATA platform or the original data source	The client code correctly verifies the enclave's unique signature (MRENCLAVE) to prove its authenticity. Although this value is embedded in the client code, and therefore links back the trust to the SILENTDATA frontend, users can build the enclave from source to verify its correctness. Applied Blockchain have published past and present MRENCLAVE values on their website and on GitHub which allows users to easily compare them with both the client and the reproducible builds of the enclave.