Secret Sharing Environment for Remote Cryptographic Services

Degree programme: BSc in Computer Science | Specialisation: IT-Security
Thesis advisors: Prof. Dr. Rolf Haenni, Prof. Gerhard Hassenstein
Expert: Dr. Igoz Metz (Glue Software Engineering AG)

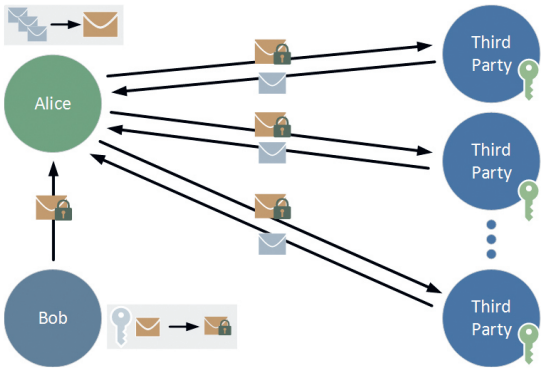
We present a proof of concept that allows outsourcing keys for decryption and signing securely to a third party. The user does not have to trust a single provider because keys are shared over several parties. A single service provider is unable to misuse an outsourced key since it only holds one partial key.

There is a new trend called remote signing. Cryptographic keys are no longer stored on a device that the user controls himself but outsourced on a service. The European Telecommunications Standards Institute (ETSI) designed and standardized a blueprint for such services.

However, this standard has serious drawbacks when it comes to user control. The design requires that a user fully trusts the service provider. There is no verification process described which would allow a user to identify misuse of his cryptographic keys. Additionally, we have analyzed the same issue for remote decryption. Outsourcing of signing and decryption operations is called remote cryptographic services.

We address these issues in our thesis. Today's cryptography enables the implementation of robust and secure infrastructure for the use case of remote cryptographic services. Our proof of concept demonstrates that the required algorithms perform well on current hardware. We achieved a good level of performance even in a web browser, although the operations are quite elaborated.

Our proof of concept is a web browser extension that performs cryptographic operations for OpenPGP messages. While signature verification and encryption happen locally in the browser, a trust service provider handles decryption and signature creation.



High level overview of a remote decryption

The extension verifies and reassembles the received results. A user can pick trust service providers of his choice at the time of key generation. These providers are then involved in the future cryptographic operations.

Our work mainly focuses on encryption. It is achievable to sign as well, but with simpler cryptography. We use threshold ElGamal for encryption and decryption. This is a combination of the ElGamal cryptosystem with special key setup where the secret key is shared using a threshold secret sharing scheme. Each provider only performs a partial decryption, which does not leak information about the plaintext. Moreover, only a defined subset of providers needs to deliver a valid partial decryption because we use a threshold scheme. This improves the schemes resistance against denial of service by a single provider. We use cryptographic proofs to verify if a provider acts honestly.

We have not implemented a threshold secret sharing scheme for signing. Instead, we implemented a kind of multi-signature. Each provider holds a typical signing key. A signature is valid, if enough honest providers sign the message.

Our web browser extension allows to encrypt or sign text using OpenPGP for instance in a webmail client or in a web-based chat client. The extension is fully compatible to other OpenPGP implementations such as GnuPG regarding encryption and decryption. The signing scheme offers basic compatibility but comes with some features that are exclusive to our software.

Roger Andrea Ellenberger

Tobias Flühmann