Poster: Johnny, after 35 years, you will finally be able to use PGP!

Ronald Petrlic
Nuremberg Institute of Technology
ronald.petrlic@th-nuernberg.de

***Abstract*—*Pretty Good Privacy* (PGP) is a standard method for users to encrypt their e-mails. However, even after nearly 35 years of existence, PGP has not found wide-spread adoption in practice. This is mainly due to its (presumed) inconvenience and the chicken and egg problem that is still predominant: Who goes the extra mile to generate keys for e-mail encryption if there is no one else to securely communicate with?**

We are the first to show that the (relatively) new paradigm of *Self-Sovereign Identity* (SSI) can foster the usage of PGP in the (near) future. Thanks to new legislation such as the eIDAS 2.0 regulation in Europe, for example, SSI is rapidly finding its way into practice. By the end of 2026, Europeans will be equipped with digital wallets that enable the usage of SSI functionality. Many other countries worldwide (including the US) have paved the way for SSI as well. Users in possession of digital wallets have (nearly) everything they need to secure their e-mails. We show why the decentralized character of SSI is a perfect fit for PGP and why integration is relatively straightforward. Not only will users be in possession of key pairs needed for encryption of e-mails thanks to SSI, but they will also be in possession of certificates by different issuers—breathing life into the original idea of a "Web of Trust" (WoT)—and allowing for a truly flexible, context-sensitive, and fine-grained authentication of e-mails.

***Index Terms*—PGP, SSI, end-to-end encryption, authentication**

I. Introduction

Pretty Good Privacy (PGP) was invented by PHIL ZIMMERMANN in 1991, the approach being published in *RFC 1991* in 1996 and later being obsoleted by *RFC 4889*, standardizing the approach as *OpenPGP* in 2007. The most current version of OpenPGP was standardized in 2024 as *RFC 9580*. Despite its long history and ongoing further developments over the past decades, PGP has not really found a wide-spread adoption in practice. In 1999, the authors of the famous "Why Johnny Can't Encrypt? A Usability Evaluation of PGP 5.0" paper [1] concluded that PGP was not usable enough to provide effective security for most computer users—studies building on that followed, showing similar results. In a study from 2022, the authors found that only 0.06 % of 81 million e-mails sent by 37,000 users at a large German university over a period of 27 years were encrypted. [2] This low number is to be seen in light of German data protection authorities (DPAs) requiring e-mail end-to-end encryption (E2EE) with PGP or S/MIME for e-mails with sensitive personal data[1]—with a higher regional court determining that e-mails with attached invoices constitute such sensitive data in 2024[2] and, therefore, need to be E2E-encrypted. But how can controllers comply with the DPAs' requirements for E2EE–being derived from the GDPR—when data subjects still do not use PGP and do not provide a public key?

At this point, *Self-Sovereign Identity* (SSI) enters the stage. SSI allows users to create their "decentralized identifiers" (DIDs) on their own and get certain attributes certified by any (trustworthy) parties in the form of "verifiable credentials" (VCs). Both DIDs and VCs have been specified by the W3C. Users may freely choose which data they want to share with service providers. The underlying cryptography allows for a "Selective Disclosure": users may even share only some authenticated claims from VCs with service providers, further strengthening user's privacy. Digital wallets constitute one of the main building blocks. The users' full control over their digital identities, and, thus, their personal data, is managed via corresponding cryptographic keys stored within the wallets (besides all the VCs). According to the European eIDAS 2.0 regulation from 2024, member states need to make sure that European citizens have access to such digital wallets by the end of 2026. This legislation (and similar legislation in other countries) ensures that SSI finds its way into practice very soon, providing users with all kinds of certified attributes (claims) and cryptographic key material. These two ingredients have been missing for a wide-spread use of PGP so far.

II. Our Approach: Integration of SSI into PGP

We assume that the user has generated a DID key pair within his wallet and published a DID document (containing the public key) on a *Verifiable Data Registry* (VDR). The private key stays secured within the wallet—in the best case, even within the device's secure element.[3] We suggest to use the DID key as PGP key. This comes with one restriction: the supported key type for a DID key is Ed25519 and not RSA; however, this is no problem as PGP supports Ed25519 as well. Moreover, the user is in possession of VCs issued by trustworthy entities. The particularity here is that the VC contains the e-mail address of the user as a *claim*—e.g., the employer of the user could issue a VC with the user's professional e-mail as a claim.

[1] E-Mail Security Orientation Guideline by the German DPAs: https://www.datenschutzkonferenz-online.de/media/oh/20210616_orientierungshilfe_e_mail_verschluesselung.pdf

[2] Oberlandesgericht Schleswig-Holstein, Az. 12 U 9/24.

[3] E.g., the IDealWallet makes use of the secure element: https://www.kaprion.de/pages/idealwallet

The user can then publish a link (as a QR code) to his DID document (including the public key) on his website (like users publish their PGP keys on their websites today) or use any of the other forms of PGP key publishing (e.g., publish the DID public key as PGP key on a standard PGP keyserver, via ”Web Key Directory” (WKD)[4], or via ”DANE Bindings for OpenPGP” (RFC 7929)). Anyone can then retrieve the user's public key and send E2E-encrypted e-mails to him. This is all standard PGP and no wallet is needed for sending users.

1) Solving the Trust Issues of PGP: PGP makes use of the ”Web of Trust” (WoT) [3]. However, verifiers lack the possibility to truly check the authenticity of signing parties. With SSI, this is different: if signers sign others' public PGP keys with their DID key, verifiers can retrieve the public information contained in the corresponding signers' DID documents and, if needed, contact the signers' agents to retrieve further proof (in form of VCs) about their authenticity.

A. Interplay between PGP and Digital Wallets

Once a user retrieves a PGP-encrypted e-mail, the PGP software needs to contact the installed wallet and ask for decryption of the session key. The wallet will then prompt the user with the request for the usage of the private DID key (serving as the PGP decryption key) and the user confirms with his PIN. The session key is then decrypted within the wallet and sent to the PGP software, which can then decrypt the whole e-mail with the session key. This approach keeps the modifications small:

1) the PGP software only needs to be adapted with regard to the decryption of the session key—this adaption is analogue to the case where the PGP software retrieves the decryption from a smartcard, and, thus, this interface can be used for communication with the wallet (instead of the smartcard).

2) the wallet needs to be adapted so that it can retrieve requests from other services on the same device, asking for decryption. As eIDAS 2.0 requires that wallets support *qualified electronic signatures*, such a request handling is already implemented in wallets and can be used for our case.

Besides E2E encryption, PGP can be used for signing e-mails as well—and this is the scenario, where SSI can unfold its full potential. A user can sign his e-mails by using his VCs. It is not only possible to combine different VCs for this purpose but also to use only certain claims from a VC (”Selective Disclosure”). Thus, depending on the context, the user could decide to include claims from VCs that are necessary to prove that he inhibits a certain role, for example (e.g., being the manager of a company)—or, going even further, prove in a whistle-blowing case that he works for a certain company without revealing his real identity. From a technical perspective, a so-called *verifiable presentation* (VP) is generated that includes the claims from the VCs and which is signed with the private DID key, thereby showing to be the controller of that DID, and, thus the controller of a certain identity. In another context, other VCs (and even other e-mail addresses) can be used for the VP generation. The recipient of the signed e-mail, on the other hand, uses the sender's public DID key (retrieved from the DID document) to verify the authenticity of the e-mail signature and the authenticity of the shown VP. In contrast to standard PGP, users, thus, do not reveal all the entities that have signed their public key, as is the case with the WoT. Furthermore, the usage of SSI—given that the VCs are issued by *qualified trust service providers* according to eIDAS 2.0 and that the DID key is stored securely within the devices's secure element—can ensure that *qualified electronic signatures* (QES) of e-mails are possible with PGP. This is not the case with PGP today (in contrast to S/MIME, which already supports QES).

B. Implementation

GNU Privacy Guard (also referred to as *GnuPG* or *GPG*) is still the most widely used PGP software in practice, as it comes with most Linux distributions and is used for integrity protection of distributed software packages in Linux systems. And this despite numerous flaws that have been found in recent years in the underlying cryptography library *libgcrypt* and the fact that GPG does not support the current OpenPGP standard (RFC 9580) but rather aiming for the LibrePGP message format (”draft-koch-librepgp-03”). Some mail clients like KMail have built-in support for GPG and for other mail clients plug-ins exist (e.g., GPGol as part of Gpg4win for Outlook and GPGMail for Apple Mail). GPG is open-source software licensed under GNU GPL. Moreover, GPG supports the use of smartcards; instead of communicating with a smartcard, GPG will need to communicate with the wallet in our case. For these reasons, we have decided to choose GPG for our implementation of the SSI functionality for PGP.

III. Conclusion and Future Work

The integration of SSI to PGP can provide a tremendous step forward for PGP, as it solves a number of issues that PGP entails. We do not come up with a new protocol for e-mail security but rather suggest a solution that builds on top of a ”standard” protocol—providing for a full compatibility with existing systems. The next step is to implement our approach.

References

[1] Alma Whitten and J. D. Tygar. 1999. Why Johnny can't encrypt: a usability evaluation of PGP 5.0. In Proceedings of the 8th conference on USENIX Security Symposium - Volume 8 (SSYM'99). USENIX Association, USA, 14.

[2] C. Stransky, O. Wiese, V. Roth, Y. Acar and S. Fahl, ”27 Years and 81 Million Opportunities Later: Investigating the Use of Email Encryption for an Entire University,” 2022 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 2022, pp. 860-875, doi: 10.1109/SP46214.2022.9833755.

[3] A. Ulrich, R. Holz, P. Hauck and G. Carle, ”Investigating the OpenPGP Web of Trust,” In: Atluri, V., Diaz, C. (eds) Computer Security – ESORICS 2011. ESORICS 2011. Lecture Notes in Computer Science, vol 6879. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-23822-2_27.

[4] Internet Draft ”draft-koch-openpgpwebkey-service-19”

Technische
Hochschule
Nürnberg

46th IEEE Symposium on Security and Privacy
May 12-15 // San Francisco, CA, USA

Integrating SSI into PGP

Situation Today

"*No one*" uses PGP

- IEEE S&P '22 study: 0,06 % of 81 m. e-mails sent by 37,000 users in 27 years were E2E-encrypted [1]

Reasons?

- "*too complicated*" (look at the "*Why Johnny [[still,] still] can't encrypt*" studies)
- "*no one else uses it, why should I be the first?*" (chicken and egg problem)

But: data protection authorities and courts (in Germany) require E2E encryption for risky e-mails

…also Situation Today

Self-Sovereign Identity (SSI) is making fast progress into practice
e.g., *eIDAS 2.0* regulation (EU): member states need to provide digital wallets by end of 2026…

Idea & Goal

Why not use SSI to solve the issues of PGP?

- Users will have crypto. keys in their wallets → use them for E2E encryption
- User will have *verifiable credentials* (VCs), so use them
  → to fix the broken *Web of Trust* (WoT) [2]
  → for fine-grained authentication (get proof that the sender has a certain role in the company)
  → to get non-repudiation with PGP (*qualified electronic signature* (QES) is possible)

Our Approach

- Use *Decentralized Identity* (DID) key as PGP key (private key stays within wallet)
- As for the WoT: users can check the authenticity of signers (by requesting their VCs)

- PGP Software and Wallet need to communicate with each other
  - on reception of an E2E-encrypted e-mail: wallet is asked to decrypt session key with DID key, PGP software then uses session key to decrypt the message
  - signing an e-mail: hash value is signed with DID key within wallet
    * Claims from VCs (using *Selective Disc.*) can be used depending on context

Implementation

- Implementation for *GNU Privacy Guard* (GPG)
- GPG supports smartcards → we use this interface to communicate with wallet instead…
- There's a "perfect match" between SSI and PGP: Wallets already come with the required crypto. algorithms for PGP → implementation should be straight-forward…

Conclusion

- We are the first to propose integrating SSI into PGP
- We do not invent a new protocol → that's good! PGP is already there, and the integration of SSI can help to increase number of users
- Users do not need to generate PGP keys any longer, they already got the necessary keys in their wallets

Prof. Dr. Ronald Petrlic
ronald.petrlic@th-nuernberg.de

References
[1] Stransky, Wiese, Roth, Acar, Fahl: "27 Years and 81 Million Opportunities Later: Investigating the Use of Email Encrypiton for an Entire University", 2022 IEEE S&P
[2] Ulrich, Holz, Hauck, Carle: "Investigating the OpenPGP Web of Trust", 2011 ESORICS