https://doi.org/10.2352/EI.2024.36.4.MWSF-343

Sovereign document verification by smartphones

Waldemar Berchtold, Julian Heeger, Simon Bugert, Lukas Biermann, Huajian Liu; Fraunhofer Institute for Secure Information Technology SIT / ATHENE; Darmstadt, Hesse/Germany

Abstract

In this paper, we present a method to secure sovereign documents. It is based on technical guidelines from the International Civil Aviation Organization. A public key infrastructure is used to secure the document information. Therefore, the personal data of the sovereign document are used such as metadata and facial image. The data is digitally signed and stored in a JAB Code, a polychrome barcode, and printed on the sovereign document. With this procedure, security papers can be completely omitted for sovereign documents and the verification of integrity and authenticity can be done by any citizen using his smartphone. The evaluation of the implementation was performed on a generalized concept for sovereign documents together with the German Federal Office for Information Security.

Introduction

Sovereign documents, such as passports, driver's licenses, visas and ID cards are issued by states to citizens and visitors and frequently targeted by criminals for fraudulent activities. Rights and obligations are attached to the documents. To enforce these rights and obligations, it is important that the security, namely integrity and authenticity, of the documents can be verified when they are in use. Currently, the security of these documents typically relies on secret features on the physical medium like e.g. printing technology. They can be forged or altered to facilitate identity theft, financial fraud, human trafficking, and other illegal activities. The average damage caused by a forged sovereign document is estimated to be around 50,000€. According to sumsub[1], more than 200 million sovereign documents were forged worldwide in 2022. Vietnam alone had to fight more than 10 million fake ID cards in 2022. Many sovereign documents contain sensitive personal information, such as biometrics and personal identification numbers (PINs). Ensuring the security of these documents is essential to protect individuals' privacy.

In the case of sovereign documents, individuals are highly sensitive when it comes to security and the protection of privacy. This also creates an area of controversy, as anyone entering into a contract with an unknown person should be able to verify their identity to prevent criminal offenses. At the same time, it is not possible to identify good ID forgeries, as the security features cannot be fully verified by a citizen. Most of the features are subject to strict secrecy. There is no provision for simple data comparison. Transmitting the data of a contractual partner to a central office would reveal a lot of information in addition to private data. Therefore, a citizen usually has no choice but to trust that an ID card is not forged if it does not look suspicious.

The goal is a simple and secure verification process for sovereign documents. To this end, we want to eliminate the currently used security features. Every citizen should be able to use his smartphone to check the integrity and authenticity of a document and likewise verify that the document belongs to a person. Sovereign documents always contain personal and sensitive data, which must be handled accordingly. Access to databases must be seen critically, as they are a popular target for attackers.

Our work uses the technical guidelines for the visa documents and the according cryptographic requirements for the Public Key Infrastructure (PKI) of the International Civil Aviation Organization (ICAO) Doc.9303 Part 12 and Part 13. With this work, we want to give everybody the opportunity to use the implementation of the technical standards within a web app and smartphone app where the security covers the state of the art and maximal user experience.

Requirements

The three main IT-Security goals are Confidentiality, Integrity and Availability. In applications that deal primarily with sensitive personal data, privacy is typically included as a further IT-Security goal.

- Confidentiality: Ensuring that sensitive information is accessible only to those who are authorized to view it.
- Integrity: Maintaining the accuracy and reliability of data and systems to prevent unauthorized alterations.
- Availability: Ensuring that information and resources are available and accessible when needed, preventing disruptions or downtime.
- Data Privacy: Protecting personal and sensitive data from unauthorized access or disclosure.

Digital signatures play a crucial role in ensuring the authenticity, integrity, and non-repudiation of electronic documents and communications.

- Authentication: Verify the identity of the sender or signatory, ensuring that the digital signature is associated with the correct individual or entity.
- Integrity: Confirm that the content of the digitally signed document has not been altered or tampered with since the signature was applied.
- Non-Repudiation: Prevent the signer from denying their involvement in the creation or approval of the document. Digital signatures provide evidence that the signer knowingly endorsed the content.

The security goals for sovereign documents are subset of the goals for IT-Security and digital signature. There are four important requirements for sovereign documents, namely integrity, authenticity, privacy and availability. The solution shall guarantee the integrity of the document, i.e. ensure that a document cannot be

[1] https://sumsub.com/blog/most-forged-ids-2022/

changed without anyone finding out about it, and authenticity, i.e. that everybody can trust who it comes from. The data need to kept private which is the ability to verify the document without needing to exchange data from the document with a third party. To verify the integrity and authenticity all necessary data need to be available at that moment.

Method

The approach shown in Figure 1 is used to guarantee the integrity, authenticity, availability and privacy of digitally signed sovereign documents. The user data is extracted, digitally signed then encoded in a matrix code, the JAB code, and printed on the official document. As mentioned above, the digital signature can be used to verify authenticity and integrity. Storing the data in a matrix code ensures availability and privacy protection. To verify, the matrix code must be read accordingly and the integrity and authenticity can then be checked. The huge advantage of this is that the documents can also be verified using a smartphone. The concept is evaluated in previous work as well [4, 5, 7].

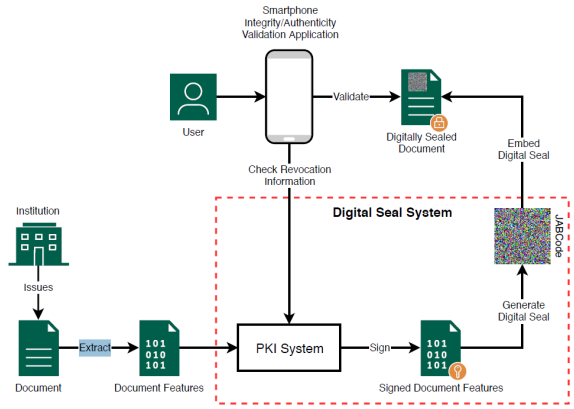

Figure 1: Digital seal generation, integrity and authenticity validation system concept.

For the data structure we use a technical guideline for travel documents from ICAO. The metadata for a visa document described there is extended in this work to include a passport photograph. This allows the document to be visually associated with a person. We digitally sign the metadata as well as image data formed. The signature is stored together with the data in a barcode, which is printed together with the passport image and the metadata. This set of information forms the sovereign document that we consider in this work.

We use JAB Code as barcode in this work, because the required payload is between 1kB and 1.24kB which needs a barcode with higher data capacity and density. With our image processing pipeline the photograph results in roughly 900 bytes. It is the best compromise we could achieve between visual quality and file size. The metadata can be up to 256 bytes. The signature length depends on the key length used, either 64 bytes or 96 bytes for a key length of 256 bits or 384 bits respectively.

We implemented the technical guideline [1] in a web app and usable for smartphones accessible for everyone on https://jabcode.org/fxQcfl4czH1y/jabpro as shown in Figure 2. The implementation will be published as open source on GitHub in April 2024. It covers eight different sovereign documents as they are specified. In this work, we only use "Test profile for aliens law" as this covers the highest data capacity. In the web app one can select the example certificates with the according keys as well as example metadata. After the JAB code is generated, it can be downloaded.

Figure 2: Output screen after verifying a generated JAB Code by the webapp.

Image Processing Pipeline

Since the requirement for availability and privacy protection is solved by using a matrix code that is placed on the official document and the available space on the documents is very limited, it is essential to keep the matrix code as small as possible. At the same time, a signature with a large key length and a high-resolution passport image represent a significant improvement on the current status. However, the latter means a high storage density of the matrix code. The JAB code already provides a multiplication of the storage density. However, this is not yet sufficient to fully meet the requirement. This gives us the trade-off between the optimal solution and the available resources.

We are therefore forced to save storage capacity in every respect. For the passport photo, we carried out a number of tests and weighed up visual quality against file size. The compression algorithm, the display size of the image and the compression level were the result of an evaluation by a team of five. Although the JPEG 2000 standard is not widely used and is considered outdated

(a) Original (b) Grayscale (c) Scaled

Figure 3: Passport photo of ”Erika Musterfrau”. Original (left image). Scaled to 413:531 px, converted to grayscale image, and compressed by 120 to jpeg2000 (center image, word can not show jpeg2000 compressed images, that's why we inserted a png here). Scaled the center image to 100:129px as presented to the viewer (right image).

by many in the image processing industry, it has the best properties for this application.

Our compression pipeline for passport photos first resizes the input photo to 413:531 px, convert it to a grayscale image and use a compression rate of 120 with the libopenjpeg library to a jpeg2000 image. By doing so, we have a huge advantage, namely the image size results in between 880 and 910 bytes. The application provides the image in a size of 100:129 px for a human viewer as shown in Figure 3. We tested several different image compression pipelines and found this to be the best tradeoff between visual quality and image size.

Digital Signature

The digital signature process is based on a public key infrastructure proposed in the technical guideline of ICAO, technical guideline Doc9303 Part 12[2]. Elliptic Curve Digital Signature Algorithm (ECDSA) with Brainpool-r1 (regular curve) and 256 bit as well as 384 bit is used as cryptographic algorithm. The signature is used as raw data, and we force the output length of the signature to 64 bytes and 96 bytes for a key length of 256 bits and 384 bits respectively. To do so padding is used. On the web app we simulate the certificate structure and created our own root certificate and signing certificates with the open-source library OpenSSL. We simulate the PKI structure for demo purposes and want to point out, that this demo is not for a commercial use.

JAB Code

The JAB code has many advantages. Firstly, by using 8 colors, it has three times the data density compared to other matrix codes such as QR Code or DataMatrix. It is also more flexible in form than these. In addition to square codes, rectangular codes and other shapes, such as L, C, D or U, can be easily generated. This has advantages when space is limited.

We would like to point out to the reader that even though the JAB code is an ISO standard, it is a relatively new development and unfortunately cannot yet be read by all commercial readers. This is due to the monochrome cameras used there.

Implementation

We implemented the data structure and encoding in a C library. The JAB Code generation is an existing library written in C and published under https://github.com/jabcode/jabcode [3, 6]. For the cryptographic steps in the signing and verification processes, e.g. encryption and decryption, we use the OpenSSL library. The interaction with the libraries is implemented as progressive web app.

The whole program is cached when visiting the webpage https://jabcode.org/fxQcfl4czH1y/jabpro. All the input data will be processed on the client side and not uploaded to the server. By visiting the web app the version is checked and reloaded if a new version is available. In the verification process of the demo app, the web app prompts the user for the certificate to verify the signature. In a real scenario, a PKI may be used. No other server communication takes place and thus give the maximal security and privacy level.

Result

In this section, we are going to evaluate the robustness of the solution. The test results directly correlate with the open-source JAB code reader and the smartphone camera and image post-processing of the manufacturer.

Test Setup

We printed the JAB Codes generated with the web app https://jabcode.org/fxQcfl4czH1y/jabpro/create with the default metadata and the photo from Figure 3. We used the example private key and certificate provided for 256 bit and 384 bit. Our profile “Test profile for aliens law” was selected. For the module size we used the settings such that we got square modules of size 1.016 mm and in the second test 0.423 mm. The resulting JAB Codes are printed by an inkjet and a laser printer.

Shots were taken under different light temperatures and daylight using an Iphone 12 and OnePlus 9. 3000K, 4000K, 5000K and 6500K were used in a darkened room. The images were taken freehand at a distance of about 14 cm and 8 cm for the codes with 1 mm and 0.4 mm module side size respectively. A successful reading result is shown in Figure 2.

Test Results

In the demo the barcode reading we counted the number of shots required to read the JAB Code successfully. If more than 3 shots where necessary, we counted it as not detected. Figures 4 and 5 show the results of the Inkjet and figures 6 and 7 the results with the Laser printer. We achieved better results with the Laser printer compared to the Inkjet. We further analyzed the reason and found that the color representation of the Inkjet is not as good as with the laser. Especially the color blue and cyan are much closer to each other as expected. We also had some difficulties to read the JAB Codes with 1 mm module side size as the smartphone had a higher distance and the resolution and quality of the captured images were not so good.

Conclusion and Future work

We implemented a technical guideline for sovereign documents and added a passport photo. To handle the trade-off between the matrix code size and the required payload we focused to reduce the payload for the photo. For this, we performed photo post-processing, which was the best tradeoff between visual quality and file size. To our knowledge, there is currently no known solution for conveniently incorporating images and metadata into sovereign documents, thus making their review accessible to all. The digital signature was generated at the state-of-the-art security

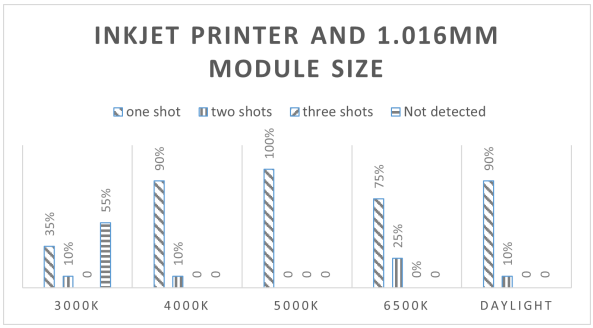

Figure 4: Results for Inkjet prints with 1.016mm module size under different light temperature, 3000K, 4000K, 5000K and 6500K.

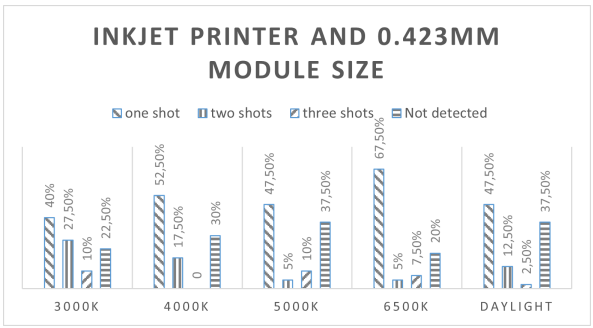

Figure 5: Results for Inkjet prints with 0.423mm module size under different light temperature, 3000K, 4000K, 5000K and 6500K.

level. The implementation is open source, and the web app is already accessible to all. The implementation offers the maximum security and privacy standard as a progressive web app. With the implementation, anyone with a smartphone can verify the authenticity and integrity of a sovereign document. States can experiment with it and thus weigh up which advantages it offers and whether there are any disadvantages for them before they introduce it. With this solution, security paper and security printing can be omitted. The reason why we postulate to eliminate the security printing is that in the most cases the security features are not known and not verifiable by a usual person. If the features cannot be easily verifiable by everybody and the technology need to be obscured, the security is questionable. This would be a huge step for the design and this sector, but we would like to discuss this step. We did the work together with the German Federal Office for Information Security and others are welcome to evaluate the solution as well.

In our further research we want to expand the tests with more printers and many more smartphones. Further, we want to use the different cameras from the phones and see if the wide-angle cameras give better results.

Acknowledgments

This work has been funded by the German Federal Office for Information Security (BSI) within the support of the National Research Center for Applied Cybersecurity ATHENE.

References

[1] BSI TR-03137 Part 1: Optically Verifiable Cryptographic Protection of non-electronic Documents (Digital Seal).

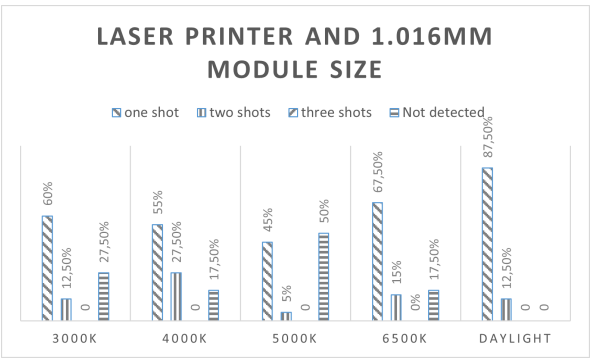

Figure 6: Results for Laser prints with 1.016mm module size under different light temperature, 3000K, 4000K, 5000K and 6500K.

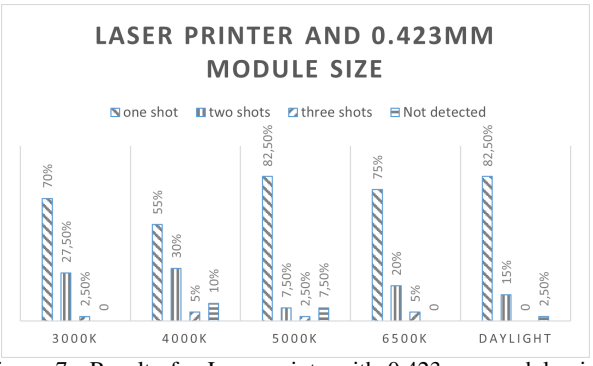

Figure 7: Results for Laser prints with 0.423mm module size under different light temperature, 3000K, 4000K, 5000K and 6500K.

[2] INTERNATIONAL CIVIL AVIATION ORGANIZATION (ICAO) Doc9303 Machine Readable Travel Documents Eighth Edition, 2021, Part 12: Public Key Infrastructure for MRTDs BSI TR-03137 Part 1: Optically Verifiable Cryptographic Protection of non-electronic Documents (Digital Seal).

[3] ISO/IEC 23634:2022 (2022). Information technology — Automatic identification and data capture techniques — JAB Code polychrome bar code symbology specification. Standard, International Organization for Standardization, Geneva, CH.

[4] Waldemar Berchtold, Dani El-soufi, Martin Steinebach, ”Smartphone-supported integrity verification of printed documents” in Proc. IST Int’l. Symp. on Electronic Imaging: Media Watermarking, Security, and Forensics, 2022, pp 325-1 - 325-5, https://doi.org/10.2352/EI.2022.34.4.MWSF-325.

[5] C. Winter, W. Berchtold, J. N. Hollenbeck, Securing Physical Documents with Digital Signatures, In: 2019 IEEE 21st International Workshop on Multimedia Signal Processing (MMSP). Vol. 09. 2019, pp. 1–6.

[6] Berchtold Waldemar, Liu Huajian, Steinebach Martin, Klein Dominik, Senger Tobias, Thenee Nicolas. (2020). JAB Code - A Versatile Polychrome 2D Barcode. Electronic Imaging. 2020. 207-1. 10.2352/ISSN.2470-1173.2020.3.MOBMU-207.

[7] Bugert Simon, Heeger Julian, Berchtold Waldemar. (2023). Integrity and authenticity verification of printed documents by smartphones. Electronic Imaging. 2023. 10.2352/EI.2023.35.3.MOBMU-352.