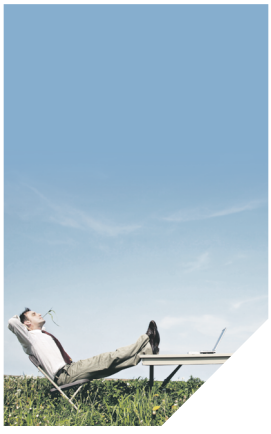
CRYPTOMATHIC

White Paper

Cloud Signing vs. Smartcard Signing





1. Introduction

What is driving successful Electronic Commerce and e-Government solutions? The answer is simple: useful applications and user-friendly solutions that can provide operational cost savings. Drivers should not be up-front legislation on digital signatures, smartcards or PKI for the sake of PKI, which was the approach we saw around the turn of the millennium. Some of the user experiences with PKI back then were so poor that for many years we could not even use the term PKI or Public Key Infrastructure. Smartcards used for digital signatures also never caught on in any significant volume for the mere fact that there are very few smartcard readers around, which is due to the costly and cumbersome nature of the readers.

But there is an alternative approach: Virtual smartcards, or signature servers as we call them – introduced by Cryptomathic in 2000 long before cloud solutions became fashionable – namely in an approach that allows transparent PKI, just as when you use your credit or debit card. This approach has been a tremendous success in countries like Denmark, Norway and Luxembourg and it is catching on outside Europe as well. In Denmark all companies and citizens must now, in principle, be able to communicate electronically with government agencies and in effect banks too.

This paper explains why a majority believe digital signatures in the cloud is the way forward and why it has been so successful.

2. What is the Goal?

There are obvious enormous savings for government, companies, individuals and the environment if we can communicate – and not least commit and been held liable – electronically rather than by paper.

But it is no good simply starting with building a PKI and then to legislate – you need to start with the applications that make it attractive to users. We have already seen how popular e-banking and internet shopping has become, and if you think of all the letters you get throughout the year from public authorities, e.g. on various taxes, registration for voting, pension etc, you start grasping the significance of making this digital. And to this you can add insurance, utilities, business contracts and much more.

Okay, so what do we need to get started? Secure reliable communication means more than anything. There is one and only one way to secure data in the Internet: by means of cryptography. For this, you in turn need keys, private keys for digital signatures and decryption and corresponding public keys for signature verification and encryption – and of course:

1. Access to your apps using your cryptographic keys
2. Secure storage of the keys
3. Secure generation of signatures

3. What are the Challenges?

3.1. WYSIWYS

The main challenges are around the generation of digital signatures. In 1998 we coined the phrase What You See is What You Sign (WYSIWYS) (see [1]). The point of this is that when you read a message or document off the screen of your tablet, workstation or whatever, and you want to commit to it by digitally signing it, how do you ascertain that it really is this message and this message only you are committing to? This by far is the hardest thing to achieve, as this is a significant challenge unless you have a Trusted GUI (Graphical User Interface), which you do not! The problem is that in the digital signing process, you digitally sign, e.g. you apply your private key, to what is known as a hash of the message that you see on the screen. Without being too technical a hash, in a way, is to a message what a DNA-molecule is to a human, in the sense that no two humans will have exactly the same DNA. When you sign, you actually have no clue per se that the hash is calculated from the very message you want to sign without a Trusted User Interface to guarantee this. For more on this, please see [1].

Storing the private key on a smartcard does not in any way address this challenge, and the only way to deal with this in case there is no Trusted GUI is to use two independent channels, as it is very unlikely both channels can be successfully attacked at the same time.

3.2. Secure Storage, Access and Generation

Regardless of the overall architecture, it is absolutely vital that the private keys are stored securely, and in a way which only the owner can access for signature generation and that the signature is generated within a protected environment.

In the late 90s, it was generally accepted that the only viable way forward was to use smartcards for this. Indeed, so did we in Cryptomathic, so we got involved in a number of pilots and ran into a number of shortcomings or challenges of which the most significant was purchasing card readers and installing them everywhere.

However, there are means for protecting private keys that are much more resilient to various attacks on the key: so-called Hardware Security Modules (HSMs). They are already used extensively by banks all over the world.

So we decided on a radically different approach, where we replaced smartcards with virtual smartcards, namely HSMs that can be accessed remotely but as securely as a smartcard in a smartcard reader.

4. What are the solutions

Assuming the reader is familiar with PKI, we briefly recall the characteristics. Each user has a public key pair for signing and possibly another for confidentiality. Users are registered with their public keys and a CA issues and maintain certificates on these.

The problem with a certificate is that although it pertains to say something about the future it really only says something about the past, such as when the certificate was generated. The certificate may have an expiry point of time at some stage in the future – but how do you know it has not been revoked prematurely? To cope with this, the traditional approach is to use blacklists and/or enable online verification, like OCSP.

Now, which consequences does this have if you use a Signature Server solution, rather than the originally intended smartcard approach?

5. Why the Signature Server Solutions are preferable

With a central signature server, also known as signatures in the cloud, the challenge as far as the individual user is concerned is to ensure that:

1. The user is properly authenticated before he can sign a message
2. Only the owner of a particular private key can initiate the signature calculation
3. WYSIWYS

To achieve this, the most secure approach is to have two independent servers, one for authentication and one for signing. They might even be physically situated in different environments with a secure channel connecting them. In fact, the signature server might be connected to a number of different authentication servers. Each server is supported by Hardware Security Modules (HSMs), and all secure calculations and verifications are carried out entirely by and in an HSM. This is very much like when debit- and credit-card transactions are being authorised by banks.

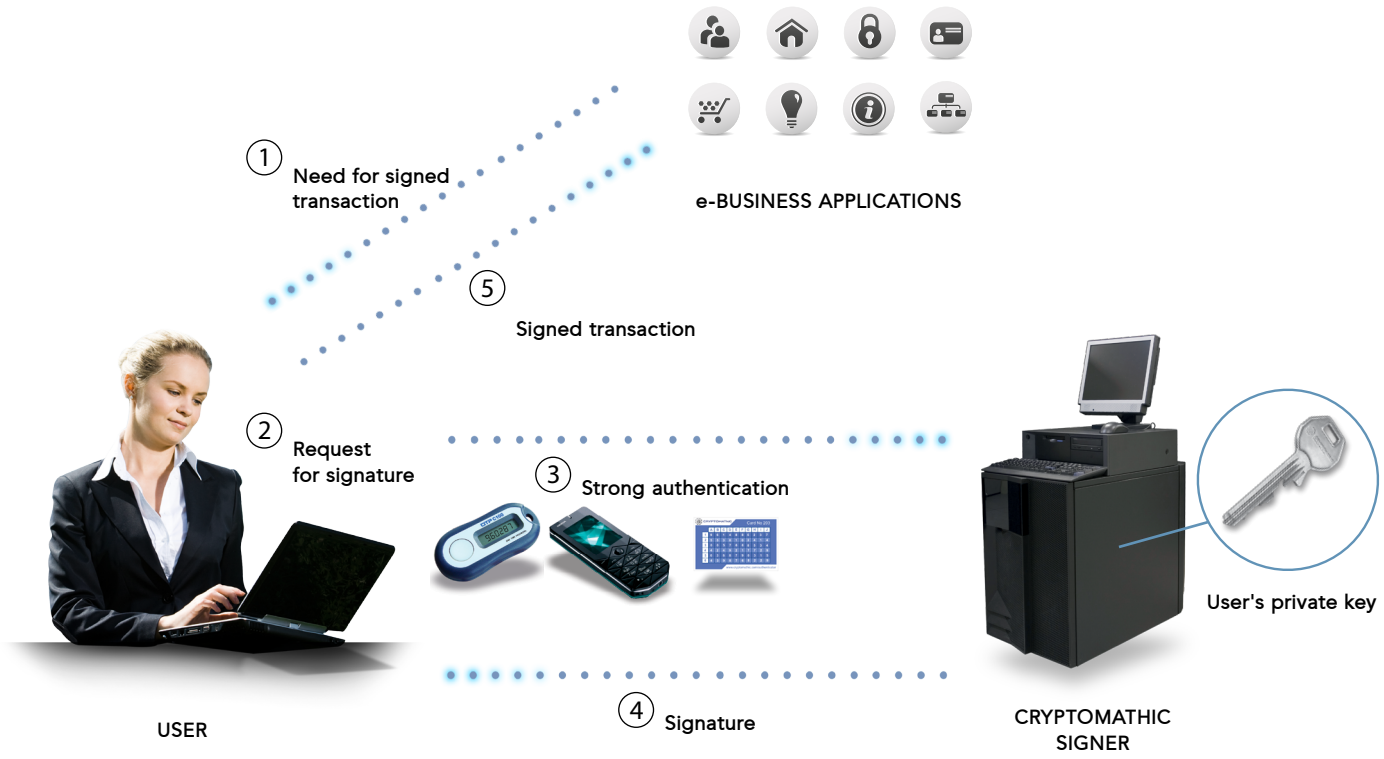
The advantage of this is that one would have a full range of various authentication mechanisms available. Below is a simplified usage example of this scenario:

The user generates a message to be signed on a tablet or work station and forwards a signing request to the authentication server together with a preferred strong authentication method to identify the user. The authentication server verifies the identity of the user and, if successful, forwards for signature generation. The signature server forwards a copy of the message to the user's mobile device together with an authorisation code of say 6 characters. If the user is satisfied s/he returns the code and the

signature is generated by the signature server through a secure TLS tunnel all the way into the authentication server.

If the Authentication and Signature Servers are in different environments this enables two independent loggings of when the signing was initiated by the user, so the system knows exactly when this took place, in contrast to the smartcard approach. WYSIWYS is thus achieved by the ability to compare the signed versions in each channel and observing that they are identical.

First of all, once a user is properly identified and registered, his private key pair is generated by the signature server, and the signature server can have a certificate issued at an assigned CA, which ONLY issues certificates for signature servers that are associated and adhere to the same kind of solution and security level.



Three-tier PKI ecosystem

5.1. Mobility and cost

The most striking advantage of signatures in the cloud is that the user is not confined to one particular workstation or laptop, which makes it a truly flexible solution. Another advantage is that if you already have the infrastructure and the applications, it is very cost-effective to add more users as you do not need any additional hardware.

5.2. PKI becomes transparent

Even better, this solution enables you to simplify the difficulties and shortcomings of a traditional PKI solution, where everybody in principle may communicate securely with everybody.

Thus all applications for which this infrastructure is used will in principle be able to recognise if a received signature from another user has been generated on a security server rather than a smartcard or even worse, in software. This obviously requires secure storage of the public key of the CA – but that will always be the case with PKI.

5.2.1. Certificate expiry/revocation no longer an issue

Certificates are like drivers' licenses: they pertain to say something about the future, but they only say something about the past. For example, It is relatively easy for a person to get an original copy of his driver's license, just go to the appropriate authority and explain you have lost it. Now you have two. If your

license is revoked, just return the latest. You can still rent a car all over the world with your original license as normally no check is carried out against a central database.

Likewise, if you receive a digital signature, you have to ask yourself: can I rely on it or could the private key used to generate it have been revoked either after or even before the signature was generated? The point is that all a certificate tells you as that at some point in time in the past the corresponding private key was valid. But to ascertain that the private key was still valid at the time when the signature allegedly was generated, you need to contact the CA e.g. through OCSP (Online Certificate Status Protocol) to inquire whether the key has been revoked meanwhile – or wait for a blacklist generated after receipt of the signature. But even this is not enough to prevent invalid certificates being used, as we shall see below.

With the central signer approach, this problem evaporates except for a very short clearing period. Indeed, with the central signer approach, the millisecond a certificate is revoked by the CA, the central signature server is informed by the CA, and requests from the user to sign with his private key are no longer honoured.

5.2.2. Blacklists are superfluous

Another major challenge of most PKI solutions is to handle revocation and blacklists. Blacklists are obsolete as soon as they have been generated and distributed.

So if the CA used in conjunction with a central signer solution issues certificate only on public keys where the private key is stored with an authorised central signature server, you no longer need blacklists. Indeed, when you receive a digital signature generated by the central signature server, you know the private key used to generate must have been valid at the time the signature was generated!

5.2.3. Independent Timestamping no longer required

Consider the scenario where you are a user of a traditional PKI solution where every user has his private key on a smartcard in his private possession, and assume you receive a digital signature from Alice on Dec 1st 2014 where she commits to paying you € 1000 on April 1st 2015. You then inquire, say on Dec 2nd, with the CA that her certificate has not been revoked. So assume in the following that it has not. April 1st arrives, but no money. You contact Alice, and she tells that Bob living next door has likely stolen her smartcard with her private key over New Year. In any event, Alice revoked her key on Jan 2nd, but there appears to be a constant stream of digital signatures generated with her private key even after, and there is nothing she can do about it. So Alice might claim that whoever the culprit is he must have generated said commitment after Dec 1st and simply back-dated the timestamp – which is trivial to do. You may of course know that Alice is lying, as you may know when you actually received the signature – but without additional measures, you cannot prove it!

This is why independent timestamping is required in a traditional PKI solution. You either need to send all received signatures to an independent 3rd party authority for independent timestamping – the most common approach proposed – or have other means of proving when the signature was received. Sadly, this is still not a well-understood risk.

5.3. EU legislation

The Electronic Identification and Trust Services (eIDAS) regulation offers a common legal framework for electronic signatures within the EU. It is intended to make it easier for citizens and businesses within EU member states to give e-transactions and other digitally signed documents the same legal status as those that are paper-based.

As a regulation, eIDAS covers the entire trust-chain including sealing, validation, time-stamping and central signing, making it more suited to the delivery of a mobile-friendly user experience.

The EU Commission has set standards around central server signing and smartcard signing to offer a clear legal framework for the roll-out of this technology. For an eIDAS compliant central signing implementation, there are a number of CEN and ETSI standards, such as TS 419 241:2014, which need to be observed. e-Signing technology must undergo an audit performed by a security assessor recognised by a supervision body if a business wants to be certified as delivering Advanced or Qualified Electronic Signatures.

Please contact Cryptomathic for more information on specific compliance requirements.

6. Conclusion

Signature Generation in the cloud is the most likely enabler of large-scale eGovernment and Electronic Commerce solutions as it is as safe or safer than the traditional smartcard approach, it enables WYSIWYS and last but not least, it makes the underlying PKI user-friendly.

	Signer	Chipcard
CARD READERS REQ	no	yes
MOBILITY	yes	no
WYSIWYS	yes	no
LOGGING POSSIBLE	yes	no
INSTANT REVOCATION	yes	no
BLACKLISTS REQUIRED	no	yes
IND. TIMESTMP REQ	no	yes
COST PER USER	very low	very high

Cloud signing vs. chip card signing

7. References

[1] WYSIWYS? – What you see is what you sign? Landrock, P. and what you sign? Landrock, P. and Pedersen, T., Information Security Technical Report Vol. 3, No. 2 (1998), 55-61, Elsevier Science Ltd.

[2] http://ec.europa.eu/dgs/connect/en/content/electronic-identification-and-trust-services-eidas-regulatory-environment-and-beyond

[3] http://ofti.org/wp-content/uploads/2013/09/feasibilityStudyonanElectronicIdentificationAuthenticationandSignaturePolicyIAS.pdf

Contact us:

Please contact your local Cryptomathic representative for more information on digital signatures in the cloud and virtual smartcards

OR

Email:
sales_enquiry@cryptomathic.com
technical_enquiry@cryptomathic.com