VRIJE
UNIVERSITEIT
BRUSSEL

SECURITY IN COMPUTING

Reinventing eIDs

Arno De Witte

Kwinten Pardon

May 29, 2017

Prof. Vincent Naessens

Faculty Science and Bio-Engineering

Todo list

Think about it, .. I have no clue at the moment 2
Find other vulnerabilities. 2
Arno? What is the authentication time? 4
lecture 5 5

1 Analysis of the Problem Statement and Protocols

The current way the Belgian Electronic identity card (eID nowadays works makes it hard to implement the use of the Belgian eID in commercial activity due to some privacy concerns. The Belgian eID releases the the user's name and national registry number at each authentication which makes you uniquely identifiable. This makes sense when we look at the eID from a government instances where you need them to be identifiable. Think about healthcare and the need to store your medical history as well as law enforcement when receiving a ticket for speeding.

However, when we take 3rd party applications into account, the release of these attributes becomes a clear issue. When releasing sensitive, uniquely identifiable, information about one self leaves them viable for identity theft. it also allows companies to create data stores where the record the purchase history of each of their customers. This might be useful to the companies to create personal advertisements based on their history. This information has become valuable as personalised advertisement has gain in popularity. However, This information is released at every authentication which causes the customers to be unable to opt out. If the use of the Belgian eID by 3rd party applications were to become widespread, they would be able to create detailed profiles of their customers. This information could cause harm when fallen in the wrong hands.

For this reason, an alternative system has been implemented with the focus on safeguarding the user's privacy. At the same time we try to answer the questions given to us and analyse our solution. By analysing the project, we hope to explore vulnerabilities and possible attacks our system may fall victim too. We also discuss solutions for the found vulnerabilities.

2 Evaluation

- **What elements are lacking in order to commercialize the system as an alternative for the current eID cards?**

- **How can you build a certificate chain? What is verified during authentication?**
  Every certificate is created by means of a key pair. One public key and a corresponding private key. The root of the chain is the central authority (CA). The CA signs other certificate using his private key. Since the public key is, as the name suggest, is public, everyone is able to check the integrity of the sign. The CA is trusted, therefore all certificates signed with his private key will be trusted as well.

  Each certificate has again his own public key, secret key pair with which they are able to sign certificates of their own. This way a certificate chain is created which eventually leads back to the CA. On authentication, the whole chain is processed until we reach a certificate signed by the CA

Think about it, .. I have no clue at the moment

- **Storing a common private key on each eID card implies a substantial risk. What risk? How can you mitigate the risk?**
  When the private key is stolen in an attack, one may create an eID card for any identity he likes. This is because only the private key is required to determine the validity of the card and the private key is shared among all citizens. this poses a threat to identity theft or the creation of fake identities.

- **How is the authentication of the SP to the card realized in the protocol?**
  Each service provider (SP) has its own certificate signed by the central authority. The public key of the central authority is know and has been hard coded on the tamper resistant part of the card. The card is therefore able to verify certificates whether or not it has been signed by the central authority.

  The Service provider sends his certificate to the card for authentication. If the card is able to verify that the certificate has been signed by the central authority, we start the second phase of the authentication by means of a challenge. The cards creates a symmetric key K_S and a challenge. the symmetric key is encrypted with the public key of the service provider PK_{SP} while the challenge and the name of the service provider are encrypted with K_S and send back to the service provider. The service provider first decrypts K_S with his private key PK_{SP}. After having retrieved K_S he decrypts the challenge and generates a response. This response is encrypted with K_S and send back to the card who will validate the response. The service provider is authenticated if the final response is validated.

Find other vulnerabilities.

- **How can stolen eID cards be revoked? How can the server check the revocation status of eID cards?**
  By changing the key pair (SK_{CO} - PK_{CO}) and issuing the new public keu PK_{CO} to the *Service Providers*, All eID cards would be revoked. This is because all eIDs share the same (SK_{CO} - PK_{CO}) key pair. The authentication of the card is accomplished by the *Service Provider* sending a challenge using the previously established symmetric key. The card decrypts the challenge and signs it with SK_{CO} The sign and $Cert_{CO}$ are then encrypted using the same symmetric key and send back to the server. The server will decrypt the message again and check the sign using PK_{CO}. If the key pair has changed, the validation will fail because the card used the old SK_{CO} to sign the challenge. This method is however only a defensible solution when SK_{CO} has been stolen. revoking all cards over the theft of one seems to be overkill.

  The government could hold a revocation list with the uniquely identifiable information of the revoked identity cards and the date of revocation. If we change the way the time stamp is requested by demanding the uniquely identifiable information when a new time stamp is requested, the government server would be able to check whether or not he's present in the revocation list. If we add the date when the card has been issued on the card, we are able to check if the card is the old or stolen revoked card, or the new card issued after the previous card has been revoked.

- **How can server certificates be revoked? How can the revocation status of server certificates be checked on the card?**
  We could expand the Government Server to include a certificate revocation list storing the certificates that have been revoked. When the card is in the process of authenticating the certificate, he first send the certificate to the Government Server who will look up the certificate in the certificate revocation list. If present, the authentication of the *Service Provider* will fail.

- **How can you transfer the data of the current Belgian eID card securely to the new eID card? Can the government mediate in this process?**
  The Belgian eID expires every five year, forcing the owner to renew his identity card. The way the renewal if your eID works is that your local city hall informs you when your new eID is available at to be retrieved. This means that the government already transfers the data to a new card at five year intervals. When a new eID systems would be taken into use, the government could mediate this process by simply handing over the new type eID upon renewal. This process could be sped up by forcing the population

to renew their eID more quickly.

- **Give a short performance analysis. What is the authentication time of the new eID? Is this acceptable? What are possible bottlenecks?**

Arno? What is the authentication time?

- **Assume that no secure random generator is available on the smart card. How to set up a secure communication channel?**
  The *Service Provider* authenticates to the card by sending its certificate. The card will initiate the handshake in order to create a secure communication channel when the certificate is valid. If the card would not have a any way of creating a random number, the card could reply with a message to the Service provider to inform him that he is allowed to initiate the handshake. Alternatively, the *Service Provider* could initiate the handshake and send this information along with the certificate upon authenticating. When the card successfully authenticates the *Service Provider* he solves the challenge and replies with the new challenge to complete the handshake.

- **How to prevent that malicious service providers can steal data on the card?**
  We can encrypt the data using the common public key of the eIDs PK_{CO}. Only the eIDs, containing the corresponding private key SK_{CO} are able to decrypt the data. The card only decrypts the data to which the service provider has access. This is a theoretical solution which has not been implemented in the current project.

- **Are Man-in-the Middle attacks possible? Why (not)? Make a clear analysis.**
  We start off by analysing all communications between the card and other servers to see if any are prone to man-in-the-middle attacks. We will follow the steps given in the document as our solution has been implemented this way.

  The first step is to contact the government server to get a time stamp. The time stamp is encrypted with the private key of the government server. Everyone is therefore able to decrypt the message using the know public key but it is impossible to fake the response as the private key remains unknown to all but the government server.

  The second step is to authenticate the service provider. This also consists of multiple phases. First of all, the service provider send his certificate which has been signed by the Central authority. This certificate is send without

encryption. Since the certificate is publicly available, this is not an issue. In the second phase, a challenge and symmetric key are created and send back to the service provider. The challenge is encrypted by the symmetric key while the symmetric key is encrypted with the public key of the service provider. Only the service provider is able to decrypt the symmetric key with his private key. Therefore only the service provider is able to decrypt the challenge. All further communication is encrypted with the symmetric key. Since the service provider was the only one able to send decrypt the symmetric key, all communication is safe.

All communication in steps three and four are encrypted with the symmetric transferred in step 2. Since we already concluded that the symmetric key could not be intercepted by anyone else but the service provider, we deem the communication channel to be safe.

We therefore conclude man-in-the-middle attacks to be impossible.

- **Compare the Belgian eID card to the alternative eID card. Give a table with differences related to security and privacy.**

lecture 5

3 Possible Attacks

3.1 Stolen Private Key

Central Authority

If the private key of the Central Authority were to be stolen, one would be able to create their own service and appoint them rights to all the information on the card. Since the certificate of this malicious service provider would be signed by the private key of the central authority, they would be trusted. This way, private information could be stolen by an attacker.

Common eID Private Key

If the common private key of the eIDs were to be stolen, one would be able to create eIDs at will. This would enable the attacker to steal identities without anyone being aware of the theft. They would also be able to create new identities at will.

3.2 Service Provider

If the private key of the service provider would be stolen, a man-in-the-middle attack would be possible due to the ability to intercept the communication in phase 2 of the service provider authentication step.

4 Future Work

4.1 Age Calculation

At the moment the age is a static filed on the card. Since the card also knows the date of birth of the user, age can be calculated using the current date. In order to this, the threshold of the last validation time may not exceed 24 hours. An other work around is to make the card contact the government server regardless the last validation time if the last validation time was less then threshold time away from the users birthday.