# How to Protect a Signature from Being Shown to a Third Party?

Marek Klonowski, Przemysław Kubiak,
Mirosław Kutyłowski, Anna Lauks

Wrocław University of Technology

Kraków, TrustBus 2006

# The Introduction

1. In most of digital signature schemes the recipient can prove having a valid signature
2. Some schemes allow to control the flow of signatures by enforcing cooperation with designated persons during the verification protocol
   - Undeniable signatures
   - Designated confirmer signatures
   - ...

# The Introduction

1. In most of digital signature schemes the recipient can prove having a valid signature
2. Some schemes allow to control the flow of signatures by enforcing cooperation with designated persons during the verification protocol
   - Undeniable signatures
   - Designated confirmer signatures
   - ...

# The Goal

To create a model in which:

1. The signer is partially protected
2. The recipient is able to show the signature to the third party
3. If the recipient presents the signature to the third party, he will be punished for that → he will do that only in a very special situations

# Dedicated Digital Signature

**Dedicated Digital Signature (dds)** of message $M$ is a special construction that:

1. Allows only a designated verifier to retrieve a standard signature of $M$ from the dds
2. Together with the standard signature of $M$ reveals depending on the protocol version:
   - the private key of designated verifier
   - designated verifier's signature of a particular message

# Dedicated Digital Signature

**Dedicated Digital Signature (dds)** of message $M$ is a special construction that:

1. Allows only a designated verifier to retrieve a standard signature of $M$ from the dds
2. Together with the standard signature of $M$ reveals depending on the protocol version:
   - the private key of designated verifier
   - designated verifier's signature of a particular message

# Dedicated Digital Signature

**Dedicated Digital Signature (dds)** of message $M$ is a special construction that:

1. Allows only a designated verifier to retrieve a standard signature of $M$ from the dds
2. Together with the standard signature of $M$ reveals depending on the protocol version:
   - the private key of designated verifier
   - designated verifier's signature of a particular message

# Dds Leaking the Verifier's Private Key

**Scenario:**

- Alice constructs a dds of a message $M$
- After getting the dds of $M$, Bob can transform it into a standard Alice's signature of $M$
- If Bob presents this signature to the third party that knows the dds of $M$, then Bob's private key can be computed

# Assumptions

- $g$ is the generator of a subgroup of $\mathbb{Z}_p^*$
- $\operatorname{ord} g$ has no small prime factors
- $\operatorname{ord} g = q$, where $q$ is some very large prime divisor of $p - 1$
- Alice and Bob use the same $p$ and $g$

| | Alice | Bob |
|---|---|---|
| private key | $x$ | $x_1$ |
| public key | $y = g^x$ | $y_1 = g^{x_1}$ |

# Creation of a Dedicated Signature

Alice:

1. Chooses $k \in \{1, 2, \ldots, q-1\}$ uniformly at random
2. Computes:

$$\begin{aligned} a &:= y_1^k \bmod p \\ b &:= k^{-1}\left(H(M) - ax\right) \bmod q \end{aligned}$$

where $H$ is a hash function

$(a, b)$ – the dds of $M$ given to Bob

# Creation of a Dedicated Signature

Alice:

1. Chooses $k \in \{1, 2, \ldots, q-1\}$ uniformly at random
2. Computes:

$$\begin{aligned} a &:= y_1^k \bmod p \\ b &:= k^{-1}\,(H(M) - ax) \bmod q \end{aligned}$$

where $H$ is a hash function

$(a, b)$ – the dds of $M$ given to Bob

# Transformation of a Dedicated Signature

- Bob computes:

$$\hat{a} := a$$
$$\hat{b} := x_1^{-1} \cdot b \bmod q$$

- $(\hat{a}, \hat{b})$ is an ElGamal signature of Alice

$$\hat{a} = g^{(x_1 \cdot k)}$$
$$\hat{b} = (x_1 \cdot k)^{-1}(H(M) - ax)$$

  - it is valid $\iff$ $\hat{a}^{\hat{b}} \cdot y^{\hat{a}} = g^{H(M)}$

# Presenting a Signature to Other Parties

- Bob shows the signature $(\hat{a}, \hat{b})$
- Anybody who has access to the dds $(a, b)$ can retrieve Bob's private key $x_1$ from equality:

$$\hat{b} = b \cdot x_1^{-1} \bmod q$$

From where the third party can get the parameter $b$?

- Alice can publish it
- The protocol can be easily improved so that Bob will have to give this value

# Presenting a Signature to Other Parties

- Bob shows the signature $(\hat{a}, \hat{b})$
- Anybody who has access to the dds $(a, b)$ can retrieve Bob's private key $x_1$ from equality:

$$\hat{b} = b \cdot x_1^{-1} \bmod q$$

From where the third party can get the parameter $b$?

- Alice can publish it
- The protocol can be easily improved so that Bob will have to give this value

# Dds Revealing the Verifier's Signature

Properties:

- If designated verifier shows the signature of $M$, his signature of some message $M_1$ can be revealed
- Construction is similar, a little bit more sophisticated
- The potential loss of designated verifier is bigger

# Extensions

## Multi-key scheme

- $n$ different designated verifiers $V_1, \ldots, V_n$
- Each designated verifier receive one dedicated signature
- Designated verifiers have to cooperate in order to transform dds-es into standard signatures
- Private keys $x_1, \ldots, x_n$ will be revealed only after all verifiers $V_1, \ldots, V_n$ show signatures corresponding to the appropriate dedicated signatures

# Extensions

**Example application of multi-key scheme:** business negotiations

- Alice + two negotiators
- Alice gives the negotiators two different signed documents
- If they try to use both, their private keys will be revealed

# Extensions

## Threshold scheme

- Signer sends $n$ dedicated signatures to the verifier
- Designated verifier is allowed to use $k-1$ regular ElGamal signatures corresponding to $k-1$ out of $n$ dds
- If designated verifier uses more then $k-1$ signatures his private key will be revealed

**Example application:** business representative

- The representative receives some number of dds-signed messages
- He can use only a part of signatures

# Extensions

## Threshold scheme

- Signer sends $n$ dedicated signatures to the verifier
- Designated verifier is allowed to use $k-1$ regular ElGamal signatures corresponding to $k-1$ out of $n$ dds
- If designated verifier uses more then $k-1$ signatures his private key will be revealed

**Example application:** business representative

- The representative receives some number of dds-signed messages
- He can use only a part of signatures

# Conclusions

- Dds – a kind of box in which a standard signature is hidden
- Construction of a dds based on ElGamal scheme is relatively straightforward
- After transformation designated verifier receives a regular ElGamal signature
- The private key of designated verifier can not be revealed until he presents a signature retrieved from dds

# Open Problem

Designated verifier may try to **avoid** a punishment:

The dedicated verifier may provide zero-knowledge proof that he has a certain ElGamal signature.

How to design dds scheme so that it would not be possible?

# Thank you for attention!