Using Pets for Age Verification (and more)

Olivier Blazy

Bruxelles

2023

Blazy Olivier

Professor at Ecole Polytechnique

Olivier.Blazy@polytechnique.edu

PhD: Zero-Knowledge proofs and applications

Habilitation: Implicit Hash Proofs

PoC: Double Anonymous Online Age Verification with CNIL (French DPA)

(Round 4 candidates for the PQC competition)

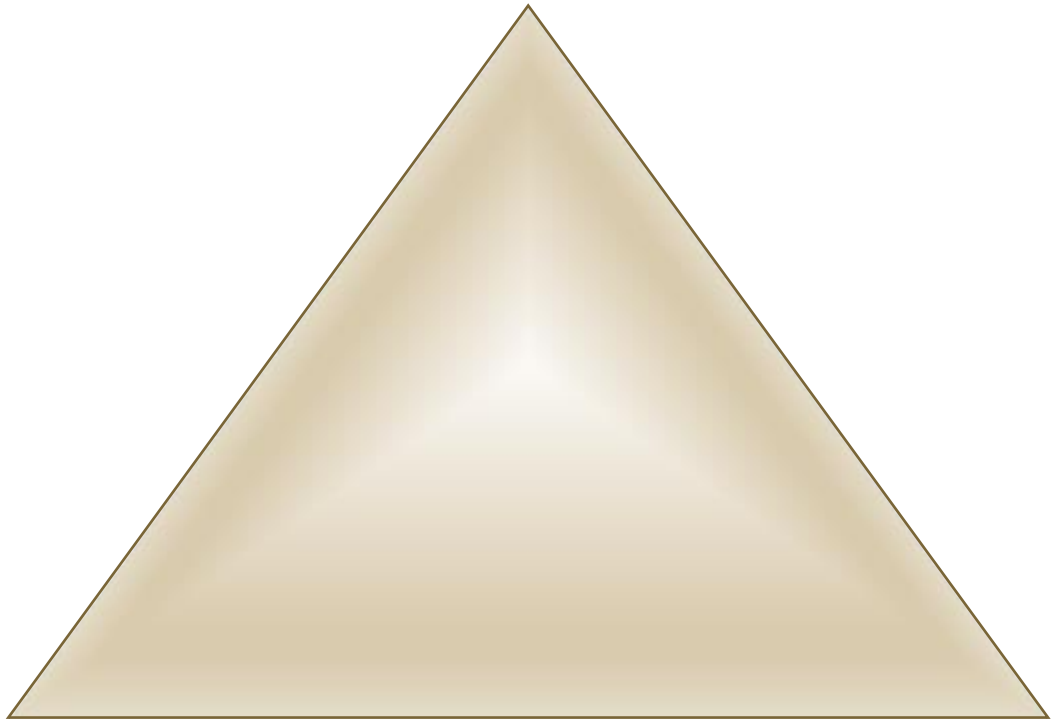
Age Verification

More and more pressing issue

- Internet should not be a place outside the law
  - What is limited in the physical world should have the same restrictions online
  - What is permitted in the real world should be allowed online
- Many issues
  - Who is behind the screen?
    - 1 user = 1 login = 1 person?
  - What kind of guarantees do we want?
    - Block all underage? Let all legit users access? Both?
  - Do we take the physical world as a guideline? Should we do better?
    - Should we leak someone name when checking their age? To whom?

A delicate Tripod

Unforgeability



Protection against the Verifier

Protection against websites

Unforgeability

It should not be possible to:

- produce a valid token, if we are underage
  - Use of a digital signature mechanism
    Signature are not anonymous - pseudonymous

- replay a token
  - Interaction is necessary / Token should have restricted use
    There should exist a nonce to avoid replay by other users

Protection against the website

The website should not learn:

- Our identity

  Difference with the physical world: Selective Credential Disclosure

- Our age (Knowing whether someone is above 18, is not knowing he is 37)

  ➡ The answer should be a bit, and not contain the age. (GDPR: Minimisation)

- Learn which age check authority certified the age

  ➡ Signature should be anonymous (aka group signature)

PETs for Authentication: Group Signatures

Signatures are a well-known primitive: An entity authenticates a message so that anyone can check that the message has been approved by said entity.

Problem: If a bank authenticates an ID, then people learn the ID is a client of the bank.

Group Signature allows to sign “anonymously” in a group. Meaning, that anyone knows that the ID has been certified by a member of the group but without knowing which.

Special authority (Opener) can revoke this property in case of an authority misbehaving.

Group Signatures: Security

Experiment $\mathsf{Exp}^{\mathsf{anon}-b}_{\mathsf{GS},\mathcal{A}}(\mathfrak{K})$

1. $(\mathsf{pk}, \mathsf{msk}, \mathsf{skO}) \leftarrow \mathsf{Setup}(1^{\mathfrak{K}})$
2. $(m, i_0, i_1) \leftarrow \mathcal{A}(\texttt{FIND}, \mathsf{pk}, \mathsf{msk} : \mathsf{joinP}, \mathsf{corrupt}, \mathsf{sign})$
3. $\sigma \leftarrow \mathsf{Sign}(\mathsf{pk}, i_b, m, \mathsf{sk}[i])$
4. $b' \leftarrow \mathcal{A}(\texttt{GUESS}, \sigma : \mathsf{joinP}, \mathsf{corrupt}, \mathsf{sign})$
5. $\texttt{IF}\ \ i_0 \notin \mathsf{HU}\ \texttt{OR}\ \ i_1 \notin \mathsf{HU}\ \texttt{RETURN}\ 0$
6. $\texttt{RETURN}\ \ b'$

Experiment:

1) Generate all the keys
2) The adversary is the authority (msk) and can corrupt users. After a while, it picks two (honest) providers, and a target message.
3) We pick one of the providers, and sign in their name and send it to the adversary
4) The adversary guess which one?
5) If the provider is not corrupted
6) We consider the adversary guess.

Protection against the verifier

The verifier should not learn:

- Which age is expected (SN > 15 vs X-rated site > 18)
  - ➡ Three main thresholds exist, so answer to all 3

- Learn for which website the query is
  - ➡ The query should not contain information about the website
  - ➡ No direct communication between the website and the vérifier

Physical world: When emitting your ID card, nobody knows what you are going to do with it

Very high overview of the PoC with CNIL and PEReN

A meta-authority certifies the verifiers (ERGA or its national equivalent)

→ Allow verifiers to prove they operate within some legislative framework

When a user access a website, they receive a challenge

→ Separation + Unicity

The user forwards this challenge to a verifier of their choice

→ Separation + Choice

The verifier signs (for the meta-authority) the challenge

→ Anonymity of the verifier + Unforgeability

The user forwards the signature

→ Separation + Anonymity of the user

General evolutions

The API is compatible with every age verification system

- No technical limitation, the legislator can pick those he finds suitable

The code is open, online for nearly a year

- Public audits are good, suggestions / evolutions are welcome
- Proposing a digital common is important in a digital context

A modular tool

- Possibility to integrate a mechanism to bill the verification to the platforms. (2 lines to tweak)
- Anonymity can be reinforced (blind signature, randomisation [already here])
- Can add an additional layer, so that User can check they are not targeted by their verifier
- Various trust level can exist for verifiers depending on the context

PETs for Anonymous Billing: Batch Threshold Opening

Age provider want to get money for their services. As directly billing the user is not acceptable, they want to bill the service to websites.

Problem: We build a gap so that none knows the identity of the other

A Tax Service could get the generated token for a website when doing key refreshing and do a batch threshold opening. Without handling individual token, they could get a total tally of the form, N tokens were generated by A, M by B, L by C…
Batch + Threshold decrypting is often done for ballot opening in e-voting.

Canari tokens can be generated to check whether a website “hides” some token to avoid fees…

PETs for Strengthen Anonymity: Blind (Group) Signature

As is, the authority learns which nonce it is signing.

Problem: While the nonce itself does not contain information about the website, a collusion between an authority and a website could leak information

Blind Signature allows an entity to sign a message without learning its content. This would require a little more work on the user/client side to transform the blind signature into a regular signature.

Small caveat: Users not using a VPN already create this “link” by having the IP address on both interactions.

Blind Signatures: Security

$\mathsf{Exp}^{\mathsf{bl}-b}_{\mathcal{BS},\mathcal{S}^*}(\mathfrak{K})$

1. $\mathsf{param} \leftarrow \mathsf{BSSetup}(1^{\mathfrak{K}})$
2. $(\mathsf{vk}, M_0, M_1) \leftarrow \mathcal{A}(\texttt{FIND} : \mathsf{param})$
3. $\sigma_b \leftarrow \mathsf{BSProtocol}\langle \mathcal{A}, \mathcal{U}(\mathsf{vk}, M_b)\rangle$
4. $\sigma_{1-b} \leftarrow \mathsf{BSProtocol}\langle \mathcal{A}, \mathcal{U}(\mathsf{vk}, M_{1-b})\rangle$
5. $b^* \leftarrow \mathcal{S}^*(\texttt{GUESS} : M_0, M_1)$;
6. RETURN $b^* = b$.

Experiment:

1) Generate all the keys
2) The adversary is the authority (sk) and can corrupt users. After a while, it picks two messages.

3) We pick one of the message, and ask a blind signing on one,
4) Then on the other
5) The adversary guess which one was first
6) He wins if it's the right one

PETs against SubGroup: Steppable Group Signature

As is, the user gets a signature without learning who generated it.

Problem: If an evil authority has two certified keys, they can use one for all users but one. This way, when someone opens the tokens, they could detect those generated for Alice.

Steppable Signature allows an authority to voluntary prove to the user, that they use the secret associated with the general public value: increasing trust at a very negligible cost

A posteriori Group Signature: One could also conceive signatures that are not anonymous by default, and where users can then add the extra layer of anonymity before sending them to the website

A trade-off: The first solution only ask user simple computation to check something, the other requires “more” cryptographical computations, as they need to add protection, and so they need a device with enough entropy.

Pitfalls: Moving from Cryptography to the Real World

What about digital fracture?

- People accessing a website already have some tech available
- Might be an issue for tourists, so need for some age verifier not based on local IDs.

What about VPN?

- Independant from the solution… If the website don’t know they should do an age check
- Need for a global solution

Anonymity?

- In term of GDPR it’s pseudonymity. IP is not hidden.
- But the API does not weaken « more », the loss in privacy due to this

Beyond Age Verification

Age verification is a start.

It is a very sensitive topic due to its application.

Deployment is far from easy:

- Need for standardization
- **Public** audit
- Create trust…
- Find an application for **Good**

eIDAS is coming!

Requires work for compatibility

Leads to extensions:

- General attributes management
- Selective disclosure
- Proofs of Attribute

PETs for for Beyond: Anonymous Credentials

As is, we now have a proof of majority.

Problem: Modern world has complexed system. What if I need to prove that I am a French citizen, that worked in Germany, and I have a bus pass in the UK?

Anonymous Credentials allow much more complex access policy. As such, every useful characteristic can be certified by one/multiple authority so that users/citizens gain full control of their online access.

A point of Warning: Credential Reuse

For proof of majority, the problem was limited

Problem: What if I prove to someone I can access a document and then they (maliciously) want to reuse this proof?

In the real world, this (sometimes) happen when sending scan IDs.

Digitally, we should modify the credentials for the target website. This way, once used, they cannot be replayed somewhere else.

Merci

Olivier.blazy@polytechnique.edu

https://github.com/LINCnil/SigGroup

ÉCOLE POLYTECHNIQUE

INSTITUT POLYTECHNIQUE DE PARIS