Zero-knowledge proofs of identity using electronic passports

[

282155110-514ae671-3c02-434f-ac6a-31ce20eec24d

1792×401 181 KB

](https://ethresear.ch/uploads/default/original/2X/a/ae85fbbbb938d295752b4f8fcb38f3fcab695885.jpeg)

Many applications need to verify their user's identity online, whether it is nationality, age, or simply uniqueness. Today, this is hard. They are stuck between shady heuristics like tracking IP addresses and technologies like Worldcoin that need to deploy their infrastructure widely.

Fortunately, UN countries in association with the International Civil Aviation Organization have built a great tool for us to piggyback on: electronic passports. They are issued by more than 172 countries and include an NFC chip with a signature of the person's information, including name, date of birth, nationality and gender. Issuing countries make their public keys accessible in online registries, enabling the verification of signatures.

A circuit for passport verification

For someone to prove their identity using a passport, they will have to do two things. First, read the content of their passport's chip. This can be done easily with any NFC-enabled phone. Then, show a verifier that their passport has been correctly signed. Instead of sending all of their personal data for the verification to happen, they can generate a zero-knowledge proof that redacts some of their inputs.

Our circuit will have to checks two things:

- The disclosed attributes have been signed correctly
- The corresponding public key is part of the public key registry of UN countries

A simple circuit compliant with the electronic passport specs would look something like this:

[

B1_4A2ZxC.png

2483×1669 96.6 KB

](https://ethresear.ch/uploads/default/original/2X/e/e55186294a00f20a4e0fd0cdc8da1172a89d3e3e.png)

Here is roughly what happens:

- Each datagroup stored in the passport contains some of the person's information. The datagroups we are most interested in are the first one (nationality, age, etc) and the second one (photo). The circuit takes them as inputs along with the signing public key.
- Datagroups are hashed, concatenated and hashed again.
- The final result is formatted, hashed and signed by the country authority. We can use the public key to check this signature.

This makes the following attributes disclosable: name, passport number, nationality, issuing state, date of birth, gender, expiry date, photo.

Some countries also provide additional data like place of birth, address, phone number, profession and a person to notify. Biometrics like fingerprint and iris are sometimes included but can't be retrieved, as they require a special access key.

In practice, we want our circuit to have a few other features:

- Instead of passing the country's public key directly, we want the user to prove that the public key that signed their passport is part of the registry published by the ICAO. This can be done by passing a merkle proof of inclusion and having only the merkle root as a public input.
- To allow for selective disclosure of any attribute, we pass a bitmap as a public input that will redact some of the attributes.

- We want specific modules for age disclosure and nationality list inclusion. A range check can guarantee someone is above a certain age without disclosing the precise age, and an inclusion check can be done over a set of countries to prove someone is or is not a citizen of any country in a list.
- For applications like minting an SBT or voting, we want to check that the passport is not expired. This can be done by passing the current date and doing a range check over the date in the circuit. We can then check that the current date is correct using the block timestamp in a smart contract or server-side in offchain verification.
- For applications that need sybil-resistance, we want to store a nullifier that prevents using the same passport twice. The simplest approach involves storing a hash of the government's signature, though this does not render the individual anonymous from the government's perspective. There are other approaches, see here for a discussion of the tradeoffs.

A map of a more complete circuit can be foundhere.

One of the challenges is thenumber of signature algorithms used. Most countries use common ones like RSA with SHA256, but the ICAO specifications are quite permissive and some countries chose to use hash functions like SHA512 or unusual padding formats. We currently support the most common one and we are working on adding support for more.

Applications

Applications roughly fall into three categories: proof of humanity, selective disclosure and authentication.

Proof of humanity can be used in general for sybil resistance. This includes voting, fair airdrops, quadratic funding and helping social media fight bots. If passports can't be construed as a general solution today, they can be integrated into wider systems like Gitcoin Passport or Zupass.

Selective disclosure has applications like privacy preserving age check. Some countries restrict buying alcohol, drugs or entering casinos for minors, and zk could help bringing better privacy to those controls.

Another example of selective disclosure is proving one is not a citizen of any country in a set of forbidden countries. This could help creating an intermediate level of compliance between KYC-gated traditional finance and fully permissionless DeFi.

Using passport signatures for authentication, one can build a ERC-4337 recovery module that asks for a proof from a specific passport as one of the conditions for recovery. Some passports also support Active Authentication, meaning they have their own private key and the ability to sign data. This would make them suitable for direct transaction signing, either for small transactions or in a multisig setup with other signers.

Limitations

The most obvious limitations of using passport signatures are the following:

- The passport does not do any kind of biometric check when the chip is read. Therefore there is no straightforward way to know if the passport has not been borrowed or stolen.
- Most of the world population does not have a passport. Even in the US, only around 50% of the population owns a passport.
- Issuing authorities can create an arbitrary number of passports and cheat in systems that require passports for sybil resistance.
- Passports can be lost or revoked. Some countries allow citizen to keep their previous passport when they are issued a new one. Some people have dual citizenship. All those cases are hard to mitigate, as the signatures stay valid.

Those limitations are all quite fundamental to the way passports work today. They can be addressed by aggregating attestations from multiple sources, which will be covered in a future post.

Current state

Proof of Passport is fully open source, from mobile app to circuits. If you are interested in contributing, please checkopen issues.

While performance would have been a bottleneck a few years ago, work from teams like Polygon ID, arkworks and mopro have made client-side proving on smartphones quite fast. Generating a proof with the current circuit takes ~4 seconds on a recent iPhone.

We are currently focused on shipping the mobile app for the first integrations. It allows users to mint an Soulbound Token disclosing only specific attributes they chose, or none at all other than the validity of their passport.Contact us to try out the

beta release.

Thanks to Rémi, Andy, Aayush, Youssef and Vivek for contributing ideas and helping build this technology!