Some ZK security proofs for Belenios

Pierrick Gaudry

CNRS, INRIA, Université de Lorraine

January 30, 2017

The purpose of this document is to justify the use of ZK proofs in Belenios. Most of them are exactly the same as in Helios, and we detail them only for completeness. But in Belenios, there is also a variant to explicitly allow blank votes which was not present in Helios; hence we find it necessary to detail the security proof in that case. All the proofs use standard arguments about Σ-protocols and are mostly based on [1].

1 Background: Σ-protocols

Definition

There are two players in a Σ-protocol: a prover **P** and a verifier **V**. The prover **P** knows a secret. After the protocol, V is convinced that **P** knows the secret but does not learn any other information.

A Σ-protocol goes in three rounds:

$$\begin{array}{ll} \mathbf{P} \longrightarrow \mathbf{V} & \text{commitment} \\ \mathbf{V} \longrightarrow \mathbf{P} & \text{challenge} \\ \mathbf{P} \longrightarrow \mathbf{V} & \text{response} \end{array}$$

and in the end, **V** accepts or rejects the proof.

Properties

The correctness and security properties that are expected from a Σ-protocol are informally the following ones:

- **Completeness**. A Σ-protocol is complete when **V** always accepts in a situation where **P** knows the secret and follows the protocol.
- **Zero-knowledge**. A Σ-protocol is ZK if **V** learns nothing from the protocol except that **P** knows the secret.
- **Soundness**. A Σ-protocol is sound when **P** not knowing the secret implies that **V** rejects at the end of the protocol (with high probability).

Proving the security of a Σ-protocol amounts to proving these three properties. In general, completeness follows from the formulae. The ZK property is typically proved by simulation: one shows that **V** (or anybody) can produce a valid transcript that is indistinguishable from a real one. This indistinguishability property is called "honest-verifier zero-knowledge" in the literature. For soundness, it is often convenient to show a "special-soundness" property that works intuitively as follows: assume we are given two distinct valid transcripts with the same commitment, then we prove that we can deduce the secret from those two transcripts.

We refer for instance to [3] for a precise definition of a Σ-protocol and of the corresponding properties in modern cryptographic language. In what follows, we only provide the main arguments of the proofs.

In Belenios, all the ZK proofs are Σ-protocols transformed with the Fiat-Shamir technique in order to obtain a non-interactive proof. This has been proved to be secure in the Random Oracle model [1].

2 Basic proofs of knowledge

We start with two basic Σ-protocols that are very standard. They are used in Belenios by the trustees to prove that they follow the protocol.

Everywhere in the descriptions and in the proofs, operations between scalars are always to be understood modulo q, the order of the group, even if not explicitly mentioned. A division makes sense if and only if we divide by an element that is non-zero modulo q (which is assumed to be prime).

Furthermore, we use the expression "an element picked at random" as a shorthand for a random choice according to a uniform distribution in the finite set in which the element is taken.

2.1 Knowledge a discrete logarithm

Proof of knowledge of a discrete logarithm

Public context: Group $G = \langle g \rangle$, with $\#G = q$.
Setting: Prover **P** has a public key $h = g^x$, where x is his secret key. He wants to prove to **V** that he knows x. (**V** knows h).

Commitment P $\longrightarrow$**V**. Prover **P** picks k at random in $\mathbb{Z}_q$, computes $r = g^k$ and sends r.
Challenge V $\longrightarrow$**P**. Verifier **V** picks e at random in $\mathbb{Z}_q$ and sends e.
Response P $\longrightarrow$**V**. Prover computes $s = k + xe \mod q$ and sends s.

Result. **V** accepts if and only if $r = g^s h^{-e}$.

Completeness. If the protocol goes as expected, we have $g^s h^{-e} = g^{k+xe} g^{-xe} = g^k = r$. Therefore **V** will accept at the end of the protocol.

ZK property. Anyone can create a valid transcript as follows: pick a random e, pick a random s, and compute $r = g^s h^{-e}$. The transcript (r, e, s) is valid by construction (the final equation is verified). Furthermore, it is indistinguishable from a transcript coming from a real instance of the

Σ-protocol (r and e are uniform random in G and $\mathbb{Z}_q$ respectively, and s is uniquely determined by them).

Special-soundness. Let (r, e, s) and (r, e', s') be two distinct valid transcripts with the same commitment. Then $r = g^s h^{-e} = g^{s'} h^{-e'}$, and therefore $h^{e'-e} = g^{s'-s}$. Since the transcripts are distinct, then $e \neq e'$ (otherwise we would also have $s = s'$), and we deduce $x = (s' - s)/(e' - e) \mod q$.

This proof of knowledge is used when generating trustees keys (Section 4.4 of Belenios specification).

2.2 Proof of correct decryption

Proof of correct decryption

Public context: Group $G = \langle g \rangle$, with $\#G = q$.
Setting: Prover **P** has a public key $h = g^x$, where x is his secret key. Two group elements C and M are made public, and **P** wants to prove to **V** that he knows x such that $M = C^x$.

Commitment P $\longrightarrow$**V**. Prover **P** picks k at random in $\mathbb{Z}_q$, computes $A = g^k$ and $B = C^k$ and sends (A, B).
Challenge V $\longrightarrow$**P**. Verifier **V** picks e at random in $\mathbb{Z}_q$ and sends e.
Response P $\longrightarrow$**V**. Prover **P** computes $s = k + xe \mod q$ and sends s.

Result. **V** accepts if and only if $A = g^s h^{-e}$ and $B = C^s M^{-e}$.

Completeness. If the protocol goes as expected, we have $g^s h^{-e} = g^{k+xe} g^{-xe} = g^k = A$ and $C^s M^{-e} = C^{k+xe} C^{-ex} = C^k = B$. Therefore **V** will accept at the end of the protocol.

ZK property. Anyone can create a valid transcript as follows: pick a random e, pick a random s, and compute $A = g^s h^{-e}$ and $B = C^s M^{-e}$. The produced quadruple (A, B, e, s) has the same distribution than a genuine transcript, so we get the ZK property.

Special-soundness. Let (A, B, e, s) and (A, B, e', s') be two distinct valid transcripts with the same commitment. Then $A = g^s h^{-e} = g^{s'} h^{-e'}$, and therefore $h^{e'-e} = g^{s'-s}$ and we deduce $x = (s' - s)/(e' - e) \mod q$. Furthermore, $B = C^s M^{-e} = C^{s'} M^{-e'}$, and therefore $M^{e'-e} = C^{s'-s}$ and we deduce $M = C^x$ for the same x as just computed.

This ZK-proof is used by the trustees to prove that they have correctly decrypted the tally (Section 4.12 of Belenios specification).

3 Proofs that ballots are well-formed

3.1 Proof of set membership

Proof that a discrete log belongs to a finite set

Public context: Group $G = \langle g \rangle$, with $\#G = q$. A public encryption key h is known.
Setting: Prover **P** has made a ciphertext (α, β) public. He wants to convince **V** that it has the form $(g^r, h^r g^m)$, where m belongs to a finite set $\{M_0, \ldots, M_k\}$. In other words, he proves that he knows r such that the property is verified.

Commitment P $\longrightarrow$V.
Let $i \in [0, k]$ be such that $m = M_i$. For all $j \neq i$, the prover **P** picks random (σ_j, ρ_j) in $\mathbb{Z}_q$ and computes $(A_j, B_j) = (g^{\rho_j}\alpha^{-\sigma_j}, h^{\rho_j}(\beta/g^{M_j})^{-\sigma_j})$.
He also picks a random element w in $\mathbb{Z}_q$ and computes $(A_i, B_i) = (g^w, h^w)$.
The prover **P** sends all the (A_j, B_j) for $j \in [0, k]$.
Challenge V $\longrightarrow$P. Verifier **V** picks e at random in $\mathbb{Z}_q$ and sends e.
Response P $\longrightarrow$V. Prover **P** computes $\sigma_i = e - \sum_{j \neq i} \sigma_j \mod q$ and $\rho_i = w + r\sigma_i \mod q$. He sends all the pairs (σ_j, ρ_j) for $j \in [0, k]$.

Result. **V** checks the following equalities and accepts if and only if all of them hold. First, she checks that $\sum \sigma_j = e$, and then for each $j \in [0, k]$ that $A_j = g^{\rho_j}\alpha^{-\sigma_j}$ and $B_j = h^{\rho_j}(\beta/g^{M_j})^{-\sigma_j}$.

Completeness. By construction of the response, the first equality holds and by construction of the commitment the equalities for A_j and B_j hold for all $j \neq i$. Finally, we have $g^{\rho_i}\alpha^{-\sigma_i} = g^{w+r\sigma_i}\alpha^{-\sigma_i} = g^w\alpha^{\sigma_i}\alpha^{-\sigma_i} = g^w = A_i$ and $h^{\rho_i}(\beta/g^{M_i})^{-\sigma_i} = h^{w+r\sigma_i}\beta^{-\sigma_i}g^{\sigma_i M_i} = h^w h^{r\sigma_i}(h^r g^m)^{-\sigma_i}g^{\sigma_i M_i} = h^w g^{-m\sigma_i}g^{\sigma_i M_i} = h^w$.

ZK property. Anyone can create a valid transcript as follows: pick a random e, pick random pairs (σ_j, ρ_j), for all $j \in [0, k]$, except for σ_0 that is computed as $\sigma_0 = e - \sum_{j \in [1,k]} \sigma_j$. Then the A_j and B_j are just computed from the formulae that **V** uses for the verification. This valid transcript has the same probability distribution than a genuine transcript and is therefore indistinguishable.

Special-soundness. Let $((A_j, B_j)_{j\in[0,k]}, e, (\sigma_j, \rho_j)_{j\in[0,k]})$ and $((A_j, B_j)_{j\in[0,k]}, e', (\sigma'_j, \rho'_j)_{j\in[0,k]})$ be two distinct valid transcripts with the same commitment. Assume that for all j we have $\sigma_j = \sigma'_j$. It then follows that $\rho_j = \rho'_j$ and $e = \sum \sigma_j = \sum \sigma'_j = e'$, which contradicts that the transcripts are distinct. Therefore, there exists j_0 such that $\sigma_{j_0} \neq \sigma'_{j_0}$. We have $A_{j_0} = g^{\rho_{j_0}}/\alpha^{\sigma_{j_0}} = g^{\rho'_{j_0}}/\alpha^{\sigma'_{j_0}}$, so that $g^{\rho_{j_0}-\rho'_{j_0}} = \alpha^{\sigma_{j_0}-\sigma'_{j_0}}$ and it follows that we can compute the value $r = (\rho_{j_0} - \rho'_{j_0})/(\sigma_{j_0} - \sigma'_{j_0}) \mod q$ where the division is well defined since $\sigma_{j_0} \neq \sigma'_{j_0}$. Then, we do the same kind of computation by equating the two expressions we know for B_{j_0}. We obtain $h^{\rho_{j_0}}(\beta/g^{M_{j_0}})^{-\sigma_{j_0}} = h^{\rho'_{j_0}}(\beta/g^{M_{j_0}})^{-\sigma'_{j_0}}$ which can be rewritten as $h^{\rho_{j_0}-\rho'_{j_0}}g^{M_{j_0}(\sigma_{j_0}-\sigma'_{j_0})} = \beta^{\sigma_{j_0}-\sigma'_{j_0}}$. Raising to the power $1/(\sigma_{j_0} - \sigma'_{j_0}) \mod q$, we deduce that $h^r g^{M_{j_0}} = h^r g^m$, and therefore $m = M_{j_0}$. Therefore, we have constructed an index j_0 and a corresponding random r such that m is equal to M_{j_0}.

This ZK-proof is used by the voters to prove that their encrypted answers are within the prescribed range. More precisely, when the blank vote is not allowed, there is a first block of proofs (called individual proofs) that provides a proof that each encrypted bit is indeed a 0 or a 1, using

membership in a set of two elements. An additional proof (called overall proof) proves that the sum of those 0 or 1 belongs to a prescribed set of values. See Section 4.7 and 4.8 of Belenios specification.

It can be noticed that, in Belenios, the finite sets under consideration are integer intervals and we could use more compact proofs for membership (see for instance [2] for recent work on the topic).

3.2 Proof of possibly-blank vote

This subsection is specific to Belenios, and corresponds to Section 4.9 in the specification document, except for a minor different sign convention.

In Belenios, an encrypted possibly-blank vote takes the form of a vector $((\alpha_0, \beta_0), (\alpha_1, \beta_1), \ldots, (\alpha_n, \beta_n))$, where (α_0, β_0) is the encryption of a bit m_0 that tells whether the voter has chosen "blank vote", and the other (α_i, β_i)'s encrypt n bits m_i, each of them answering a yes/no question. Therefore, for each $i \in [0, n]$, there exists $r_i \in \mathbb{Z}_q$ such that $(\alpha_i, \beta_i) = (g^{r_i}, h^{r_i} g^{m_i})$; the value r_i is the random nonce used by the voter to encrypt the i-th bit.

By the homomorphic property, the products of the (α_i, β_i) for $i \in [1, n]$ is a valid message $(\alpha_\Sigma, \beta_\Sigma)$ that is the encryption of the sum $m_\Sigma = \sum_{i \in [1,n]} m_i$, and the corresponding random is $r_\Sigma = \sum_{i \in [1,n]} r_i$.

We assume that the prover **P** (i.e. the voter) has already given proofs that each m_i belongs to the set $\{0, 1\}$ using the technique of the previous subsection.

What remains to be proven is two-fold:

- If the "blank vote" bit m_0 is set to 1, then all the other bits must be 0. This can be rewritten as $m_0 = 0 \vee m_\Sigma = 0$ and is called `blank_proof` in Belenios.
- If the 'blank vote" bit m_0 is set to 0, then the number of the other bits that are set to 1 is within the prescribed values. This is rewritten as $m_0 = 1 \vee m_\Sigma \in \{M_0, \ldots, M_k\}$ and is called `overall_proof` in Belenios.

Those two proofs belong to the same family: we need a Σ-protocol for proving a disjunction of formulae of the form $m = v$, where only the ElGamal encryption of the m's are given.

This is essentially the same construction as for set membership, which can be seen as a particular case of this general form, where we take all the m's to be the same.

Proof of a disjunction of equalities

Public context: Group $G = \langle g \rangle$, with $\#G = q$. A public encryption key h is known.
Setting: Prover **P** has made public a set (α_i, β_i) that corresponds to encrypted integers m_i: there exists r_i such that $(\alpha_i, \beta_i) = (g^{r_i}, h^{r_i} g^{m_i})$.
He wants to prove that at least one equality in a given set of k equalities $(m_{i_j} = v_j)_{j \in [1,k]}$ holds.

Commitment P $\longrightarrow$V.
Let j_0 be the index of one equality that is true. For all $j \neq j_0$, the prover **P** picks random (σ_j, ρ_j) and computes $A_j = g^{\rho_j} \alpha_{i_j}^{-\sigma_j}$ and $B_j = h^{\rho_j} (\beta_{i_j} / g^{v_j})^{-\sigma_j}$. He also picks a random element w in $\mathbb{Z}_q$ and computes $(A_{j_0}, B_{j_0}) = (g^w, h^w)$.
The prover **P** sends all the (A_j, B_j) for $j \in [1, k]$.
Challenge V $\longrightarrow$P. Verifier **V** picks e at random in $\mathbb{Z}_q$ and sends e.
Response P $\longrightarrow$V. Prover **P** computes $\sigma_{j_0} = e - \sum_{j \neq j_0} \sigma_j \mod q$ and $\rho_{j_0} = w + r_{i_{j_0}} \sigma_{j_0} \mod q$. He sends all the pairs (σ_j, ρ_j) for $j \in [1, k]$.

Result. **V** checks the following equalities and accepts if and only if all of them hold. First, she checks that $\sum \sigma_j = e$, and then for each $j \in [1, k]$ that $A_j = g^{\rho_j} \alpha_{i_j}^{-\sigma_j}$ and $B_j = h^{\rho_j} (\beta_{i_j} / g^{v_j})^{-\sigma_j}$.

The security analysis is essentially the same as for set membership.

Completeness. By construction of the response, the first equality holds and by construction of the commitment the equalities for A_j and B_j hold for all $j \neq j_0$. Finally, we have $g^{\rho_{j_0}} \alpha_{i_{j_0}}^{-\sigma_{j_0}} = g^{w + r_{i_{j_0}} \sigma_{j_0}} \alpha_{i_{j_0}}^{-\sigma_{j_0}} = g^w \alpha_{i_{j_0}}^{\sigma_{j_0}} \alpha_{i_{j_0}}^{-\sigma_{j_0}} = g^w = A_{j_0}$ and $h^{\rho_{j_0}} (\beta_{i_{j_0}} / g^{v_{j_0}})^{-\sigma_{j_0}} = h^{w + r_{i_{j_0}} \sigma_{j_0}} \beta_{i_{j_0}}^{-\sigma_{j_0}} g^{\sigma_{j_0} v_{j_0}} = h^w h^{r_{i_{j_0}} \sigma_{j_0}} h^{-r_{i_{j_0}} \sigma_{j_0}} g^{-m_{i_{j_0}} \sigma_{j_0}} g^{\sigma_{j_0} v_{j_0}} = h^w = B_{j_0}$, since $m_{i_{j_0}} = v_{j_0}$.

ZK property. Anyone can create a valid transcript as follows: pick a random e, pick random pairs (σ_j, ρ_j), for all $j \in [2, k]$, and compute $\sigma_1 = e - \sum_{j \in [2,k]} \sigma_j$. Then the A_j and B_j are just computed from the formulae that **V** uses for the verification. This valid transcript has the same probability distribution than a genuine transcript and is therefore indistinguishable.

Special-soundness. Let $((A_j, B_j)_{j \in [1,k]}, e, (\sigma_j, \rho_j)_{j \in [1,k]})$ and $((A_j, B_j)_{j \in [1,k]}, e', (\sigma'_j, \rho'_j)_{j \in [1,k]})$ be two distinct valid transcripts with the same commitment. Assume that for all j we have $\sigma_j = \sigma'_j$. It then follows that $\rho_j = \rho'_j$ and $e = \sum \sigma_j = \sum \sigma'_j = e'$, which contradicts that the transcripts are distinct. Therefore, there exists j_0 such that $\sigma_{j_0} \neq \sigma'_{j_0}$. We have $A_{j_0} = g^{\rho_{j_0}} / \alpha_{i_{j_0}}^{\sigma_{j_0}} = g^{\rho'_{j_0}} / \alpha_{i_{j_0}}^{\sigma'_{j_0}}$, so that $g^{\rho_{j_0} - \rho'_{j_0}} = \alpha_{i_{j_0}}^{\sigma_{j_0} - \sigma'_{j_0}}$ and it follows that we can compute the value $r_{i_{j_0}} = (\rho_{j_0} - \rho'_{j_0}) / (\sigma_{j_0} - \sigma'_{j_0}) \mod q$ where the division is well defined since $\sigma_{j_0} \neq \sigma'_{j_0}$.

We can then use similarly the two known expressions for B_{j_0}: we have $B_{j_0} = h^{\rho_{j_0}} (\beta_{i_{j_0}} / g^{v_{j_0}})^{-\sigma_{j_0}} = h^{\rho'_{j_0}} (\beta_{i_{j_0}} / g^{v_{j_0}})^{-\sigma'_{j_0}}$, so that $h^{\rho_{j_0} - \rho'_{j_0}} g^{v_{j_0} (\sigma_{j_0} - \sigma'_{j_0})} = \beta_{i_{j_0}}^{\sigma_{j_0} - \sigma'_{j_0}}$. Raising to the power $1/(\sigma_{j_0} - \sigma'_{j_0}) \mod q$, we obtain $h^{r_{i_{j_0}}} g^{v_{j_0}} = \beta_{i_{j_0}} = h^{r_{i_{j_0}}} g^{m_{i_{j_0}}}$. We have therefore computed an index j_0, and the corresponding random $r_{i_{j_0}}$ so that $(\alpha_{i_{j_0}}, \beta_{i_{j_0}}) = (g^{r_{i_{j_0}}}, h^{r_{i_{j_0}}} g^{m_{i_{j_0}}})$, and the equation $m_{i_{j_0}} = v_{j_0}$ holds.

References

[1] Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *ACM-CCS '93*, pages 62–73. ACM, 1993.

[2] Sébastien Canard, Iwen Coisel, Amandine Jambert, and Jacques Traoré. New results for the practical use of range proofs. In Sokratis Katsikas and Isaac Agudo, editors, *EuroPKI 2013*, volume 8341 of *LNCS*, pages 47–64. Springer, 2014.

[3] Sebastian Faust, Markulf Kohlweiss, Giorgia Azzurra Marson, and Daniele Venturi. On the non-malleability of the Fiat-Shamir transform. In Steven Galbraith and Mridul Nandi, editors, *INDOCRYPT 2012*, volume 7668 of *LNCS*, pages 60–79. Springer, 2012.