Security Analysis of $\mathcal{HABS}$

Nesrine Kaaniche and Maryline Laurent

This document provides a detailed security analysis of the attribute based credential system $\mathcal{HABS}$, in Section 1 and proofs of the homomorphism and correctness to support multiple issuers in Section 2.

1 Security of the Main Scheme

In this section, we prove that our attribute based credential system $\mathcal{HABS}$ provides the security requirements defined in Section 4.2.

1.1 Correctness

The proof of Theorem 1 relies on the correctness of the following three Equations:

$$\hat{e}(C_1, g_2) \overset{?}{=} X_{is} \cdot \hat{e}(X_u^{\mathcal{H}(\mathcal{S})^{-1}}, g_2) \cdot \hat{e}(h_1, C_3) \tag{1}$$

$$\hat{e}(\sigma_1, g_2) \overset{?}{=} X_{is}\hat{e}(A_R, A)\hat{e}(C_2', h_2) \prod_{i=1}^{l} \hat{e}(u_{\rho(i)} {h_1}^{\tau_i}, \omega_i)\hat{e}(\sigma_2, {g_2}^{m}) \tag{2}$$

$$\hat{e}(\varpi^*, g_2) \cdot {X_{is}}^{-1} \overset{?}{=} \hat{e}({X_u}^{-\mathcal{H}(\mathcal{S})}, g_2) \tag{3}$$

where ϖ^* is such that $\varpi^* = C_1' {C_2'}^{sk_{ins}}$.

First, the correctness of Equation 1 guarantees the correctness of the obtained credential. It is easy to check using the bilinearity property of pairing functions as follows:

$$\begin{aligned} \hat{e}(C_1, g_2) &= \hat{e}(x_{is} \cdot [{X_u}^{\mathcal{H}(\mathcal{S})^{-1}}] \cdot {h_1}^{r}, g_2) \\ &= \hat{e}({g_1}^{s_{is}}, g_2) \cdot \hat{e}({X_u}^{\mathcal{H}(\mathcal{S})^{-1}}, g_2) \cdot \hat{e}({h_1}^{r}, g_2) \\ &= \hat{e}(g_1, g_2)^{s_{is}} \cdot \hat{e}({X_u}^{\mathcal{H}(\mathcal{S})^{-1}}, g_2) \cdot \hat{e}(h_1, {g_2}^{r}) \\ &= X_{is} \cdot \hat{e}({X_u}^{\mathcal{H}(\mathcal{S})^{-1}}, g_2) \cdot \hat{e}(h_1, C_3) \end{aligned}$$

Second, for the correctness of the presentation token, the verifier checks if the received token $\Sigma = (\Omega, \sigma_1, \sigma_2, C_1', C_2', A, \mathcal{S}_R)$ is a valid signature of the message m, based on the predicate Υ (corresponding to $(M_{l\times k}, \rho)$). As such, the verifier first checks the set of revealed attributes $\mathcal{S}_R$. Note that the verification process has to be stopped if the verification of $\mathcal{S}_R$ was rejected. Otherwise, the verifier computes an accumulator A_R of the revealed attributes' values, using σ_2, such

as $A_R = {\sigma_2}^{\mathcal{H}(\mathcal{S}_R)^{-1}}$, where $\mathcal{H}(\mathcal{S}_R) = \prod_{a_i \in \mathcal{S}_R} \mathcal{H}(a_i)^{-1}$.
To prove the correctness of Equation 2, we first express σ_1 as follows:

$$\begin{aligned}
\sigma_1 &= C'_1 \cdot B \cdot {g_1}^{r_m m} \\
&= C'_1 \cdot \prod_{i=1}^{l} (u'_{\rho(i)})^{v_i} \cdot {g_1}^{r_m m} \\
&= x_{is} \cdot {X_u}^{-\mathcal{H}(\mathcal{S})} \cdot {h_1}^{r+r'} \cdot \prod_{i=1}^{l} (u_{\rho(i)})^{(r+r')v_i} \cdot {g_1}^{r_m m}
\end{aligned}$$

Now, we provide the proof of correctness of the presentation token verification. In the following proof, we denote $(r + r')$ by R, and the first side of Equation 2 by Ⓢ.

$$\begin{aligned}
\text{Ⓢ} &= \hat{e}(x_{is} \cdot {X_u}^{\mathcal{H}(\mathcal{S})^{-1}} \cdot {h_1}^{r+r'} \cdot \prod_{i=1}^{l} (u_{\rho(i)})^{Rv_i} \cdot {g_1}^{r_m m}, g_2) \\
&= \hat{e}(x_{is}, g_2) \cdot \hat{e}({X_u}^{\mathcal{H}(\mathcal{S})^{-1}}, g_2) \cdot \hat{e}({h_1}^{R}, g_2) \cdot \hat{e}({g_1}^{r_m m}, g_2) \cdot \hat{e}(\prod_{i=1}^{l} {u_{\rho(i)}}^{Rv_i}, g_2) \\
&= \hat{e}(g_1, g_2)^{s_{is}} \cdot \hat{e}({X_u}^{\mathcal{H}(\mathcal{S}_R \cup \mathcal{S}_H)^{-1}}, g_2) \cdot \hat{e}({h_1}^{R}, g_2) \cdot \hat{e}(\sigma_2, {g_2}^{m}) \cdot \prod_{i=1}^{l} \hat{e}({u_{\rho(i)}}^{Rv_i}, g_2) \\
&= X_{is} \cdot \hat{e}([{g_1}^{sk_u}]^{\mathcal{H}(\mathcal{S}_R)^{-1}\mathcal{H}(\mathcal{S}_H)^{-1}}, g_2) \cdot \hat{e}({g_1}^{-R}, h_2) \cdot \hat{e}(\sigma_2, {g_2}^{m}) \cdot \prod_{i=1 \cdot {X_{is}}^{-1}}^{l} \hat{e}(u_{\rho(i)}, {g_2}^{Rv_i}) \\
&= X_{is} \cdot \hat{e}({g_1}^{\mathcal{H}(\mathcal{S}_R)^{-1}}, [{g_2}^{sk_u}]^{\mathcal{H}(\mathcal{S}_H)^{-1}}) \cdot \hat{e}(C'_2, h_2) \cdot \hat{e}(\sigma_2, {g_2}^{m}) \cdot \prod_{i=1}^{l} \hat{e}(u_{\rho(i)}, \omega_i) \\
&= X_{is} \cdot \hat{e}(A_R, A) \cdot \hat{e}(C'_2, h_2) \cdot \prod_{i=1}^{l} \cdot \hat{e}(u_{\rho(i)} {h_1}^{\tau_i}, \omega_i) \cdot \hat{e}(\sigma_2, {g_2}^{m})
\end{aligned}$$

We note that $\tau_i = \sum_{i=1}^{k} \mu_j M_{i,j}$, the last equality is derived by Definition 8, such as

$$\sum_{i=1}^{l} \tau_i (v_i R) = R \sum_{i=1}^{l} \tau_i v_i = R \cdot 1 = R$$

As such, the term $\hat{e}({h_1}^{R}, g_2)$ can be represented as $\hat{e}({h_1}^{R}, g_2) = \prod_{i=1}^{l} \hat{e}({h_1}^{R\tau_i}, {g_2}^{Rv_i})$.

Finally, for the correctness of our `judge` algorithm, we consider the proof of validity of the inspection procedure proving that the El-Gamal decryption algorithm has been correctly done, using the knowledge of sk_{ins}, as presented in Equation 3. The correctness of Equation 3 is as follows:

$$\begin{aligned}
\hat{e}(\varpi, g_2) \cdot {X_{is}}^{-1} &= \hat{e}(C'_1 {C'_2}^{sk_{ins}}, g_2) \cdot {X_{is}}^{-1} \\
&= \hat{e}([x_{is} \cdot {X_u}^{\mathcal{H}(\mathcal{S})^{-1}} \cdot {h_1}^{R}] \cdot {g_1}^{-R^{sk_{ins}}}, g_2) \cdot {X_{is}}^{-1} \\
&= \hat{e}(x_{is}, g_2) \cdot \hat{e}({X_u}^{\mathcal{H}(\mathcal{S})^{-1}} \cdot {g_1}^{\alpha R} \cdot {g_1}^{-R^{\alpha}}, g_2) \cdot {X_{is}}^{-1} \\
&= \hat{e}({X_u}^{\mathcal{H}(\mathcal{S})^{-1}}, g_2)
\end{aligned}$$

1.2 Unforgeability

Sketch of proof. We prove that our credential system $\mathcal{HABS}$ satisfies the unforgeability requirement using an *absurdum* reasoning. We suppose that an attacker $\mathcal{A}$ can violate the statements of the Theorem 2 by reaching the advantage $Pr[\mathbf{Exp}_{\mathcal{A}}{}^{unforg}(1^{\xi}) = 1] \geq \epsilon(\xi)$.

Let us first start by the *MC-game*. Given the public-private key of the user (pk_u, sk_u), $\mathcal{A}$ tries to forge a credential C, while relying on several sessions. Obviously, $\mathcal{A}$ tries a forgery attack against the CDH assumption, considering that the credential element C_1 is a product of an accumulator over the set of user attributes, the secret key of the issuer x_{is} and a randomization of the public key of the inspector h_1. Knowing that this randomization is required for deriving the remaining credential elements, $\mathcal{A}$ is led to break the CDH assumption. The *MC-game* is then considered with respect to the CDH-assumption. Recall that the complexity of the CDH assumption has been studied in [3] and it is demonstrated to be hard to solve; i.e. a (t, ϵ) CDH group is a group for which the $\mathbf{Adv}(\mathcal{A}, t) \leq \epsilon$ for every PPT adversary running in a time t.

Now, we suppose that the adversary $\mathcal{A}$ can violate the CDH assumption by reaching an advantage $\mathbf{Adv}(\mathcal{A}, t) \geq \epsilon$ and show the existence of an attacker $\mathcal{B}$ that can reach an advantage $\mathbf{Adv}(\mathcal{B}, t') \geq \epsilon'$.

Intuitively, $\mathcal{B}$ relies on the capabilities of $\mathcal{A}$ to forge credentials C obtained from interactions with $\mathcal{C}$ in the *MC-Game*.

Since $\mathcal{A}$ and $\mathcal{B}$ algorithms are based on coin tosses, the first condition for $\mathcal{B}$ to succeed is that it does not abort the *MC-game* before $\mathcal{A}$. In [1], this probability has been shown to be $\frac{1}{e}$ if the probability for the coin flipping to be 0 is $\frac{1}{\xi_c+1}$, where ξ_c is the number of credential queries. The other condition of the attacker is to be able to identify the value of r or to extract the private key x_{is} of the issuer, for which the credential has been forged by the $\mathcal{A}$. After a time t', this probability is equal to $\frac{1}{\xi_c+1}$. This shows that the attacker $\mathcal{B}$ can violate the CDH-assumption with a probability equal to $\frac{\epsilon}{e(\xi_c+1)}$ which conflicts the fact that $\mathbb{G}_1$ is a (t, ϵ)-CDH group.

Another desirable property of our $\mathcal{HABS}$ construction is the presentation token unforgeability, which is based on *MU-Game* and *Col-Game*. The proof directly goes from the unforgeability property of the ABS scheme and the security of the commitment algorithm, required for proving the possession of all non-revealed attributes $\mathcal{S}_H$ with respect to the presented credential C.

We thus prove that our construction is unforgeable under the selective predicate attack (i,e; *MU-Game*, *Col-Game*), assuming that the q-DHE holds in $\mathbb{G}_1$. On one side, for the *MU-Game*, $\mathcal{A}$ relies on several *Show-Query* sessions to conduct to forgery attack against the unforgeability property of the ABS signature, referred to as presentation token. Note that our construction inherits the unforgeability property from Waters' CP-ABE scheme [5], which is proven secure under the assumption of the decisional q-Bilinear Diffie Hellman Exponent (q-BDHE) problem, formalized by Boneh et al. in [2]. Thus, based on [5, 4], the advantage of an algorithm $\mathcal{B}$, against the q-DHE assumption, is equal to $\mathbf{Adv}(\mathcal{B}, t) = \mathcal{O}(\frac{1}{\xi_s+1})$, where ξ_s is the number of showing queries the adversary $\mathcal{A}$ can make. Similarly, $\mathcal{B}$ can violate the q-DHE problem with a probability $\epsilon' \geq \epsilon \cdot \mathcal{O}(\frac{1}{\xi_s+1})$, which contradicts the q-DHE security assumption.

Consequently, we can prove the resistance of $\mathcal{HABS}$ against a collusion attack between two malicious users, considering the *Col-Game*. That is, this property

is ensured as it is considered as sub-case of the unforgeability requirement of an ABS scheme.
On the other side, the security of our commitment scheme, considered in Show algorithm for proving the possession of all attributes certified by the issuer, stems from the hardness of the DLP assumption. That is, it can be considered as the Pederson commitment scheme, which is unconditionally hiding and has been proven secure under the DLP assumption. Additionally, we have to note that $\mathcal{HABS}$ is resistant against replay attacks, thanks to the randomness appended by the challenger, for each request.

As such, our $\mathcal{HABS}$ construction satisfies the unforgeability requirement, under the q-DHE, CDH and DLP assumptions, with respect to *MC-Game*, *MU-Game* and *Col-Game*.

1.3 Privacy

Sketch of proof. Theorem 3 relies on three security games, namely *PP-Game*, *MS-Game* and *IS-Game*. That is, the attacker $\mathcal{A}$ tries to distinguish between two honestly derived presentation tokens for different settings with respect to every security game. As such, for the *PP-Game*, since a new presentation token for the same message m and the same access predicate Υ is computed from randoms, generated by $\mathcal{C}$, both presentation tokens are identically distributed in both cases. Thus, we can easily show that an ABS signature (presentation token) created by using $\mathcal{S}_1$ can be also generated using $\mathcal{S}_2$. As such, it follows the probability of predicting b is $\frac{1}{2}$.
Similarly, the *MS-Game* relies also on a *left-or-right* oracle, where an attacker $\mathcal{A}$ cannot distinguish the oracle's outputs better than a flipping coin. In fact, both presentation tokens for the same message m and the same access predicate Υ sent to different users, such as $\Upsilon(\mathcal{S}_{u_1}) = \Upsilon(\mathcal{S}_{u_2}) = 1$, are statistically indistiguishable. As such, it follows the probability of predicting b is $\frac{1}{2}$.
Then, an attacker $\mathcal{A}$, against the issue-show property, has an access to the *Issue* oracle for generating users' credentials. The *IS-Game* assumes that the attacker also knows the public keys of the requesting user. But, since an honest user produces a different presentation token for each presentation session $\mathcal{HABS}$.Show, thanks to the randomness introduced by the user while generating the ABS signature. As such, the $\mathcal{A}$ cannot distinguish two different presentations tokens with a probability such $\mathbf{Adv}(\mathcal{A}, t) \neq \frac{1}{2} + \epsilon$.
Therefore, our scheme is unlinkable, satisfying as well the privacy property. The reason is that the different entities, namely, issuers, users and verifiers, have to generate randomness for each procedure of the $\mathcal{HABS}$ construction.

1.4 Anonymity Removal

Sketch of proof. Let $\mathcal{A}$ be a successful attacker against the inspection property, with respect to the *IA-Game*.
First, if the inspector is able to conclude, then a valid presentation token has been produced during the attack on a new u^*, which contradicts the unforgeability property of our $\mathcal{HABS}$ construction. More precisely, we have to extract the underlying user u^*, and then as we know the private key sk_{u^*}, we extract a valid presentation token Σ on u^* and win the unforgeability game.
Second, we prove the resistance of our construction against such an attacker,

based on the security of El-Gamal encryption scheme which is proven to be computational-hiding. As such, the probability of success for $\mathcal{A}$ is negligible, such as $\mathbf{Adv}(\mathcal{A}, t) \leq \epsilon$. Thus, our scheme satisfies the inspection feature.

2 Homomorphism to Support multiple Issuers

This section provides the proof of homomorphism and correctness of the extension of $\mathcal{HABS}$ to support multiple issuers.

Theorem 5 defines the aggregation algorithm agg. The proof of correctness and homomorphism of Theorem 5 comes directly from the following Lemma 2.1 which expresses $\mathcal{H}(S_i \cup S_j)$ based on $\mathcal{H}(S_i)$ and $\mathcal{H}(S_j)$ in order to write $C_{\{1,S_i \cup S_j\}}$ with respect to ${C_1}^{(i)}$ and ${C_1}^{(j)}$.

Lemma 2.1 *Given the hash function $\mathcal{H}$ and for every sets of attributes S_i and S_j, there exist two integers a and b, such that $\mathcal{H}(S_i \cup S_j)^{-1} = a\mathcal{H}(S_i)^{-1} + b\mathcal{H}(S_j)^{-1}$.*

Proof 2.2 *Referring to the Bezout's lemma, the gcd satisfies the following property:*

$$gcd(\mathcal{H}(S_i), \mathcal{H}(S_j)) = b\mathcal{H}(S_i) + a\mathcal{H}(S_j) \tag{4}$$

where a and b are two non zero integers (a and b are called Bezout coefficients). In addition, the gcd and lcm satisfy Equation 5 such that

$$gcd(\mathcal{H}(S_i), \mathcal{H}(S_j)) * lcm(\mathcal{H}(S_i), \mathcal{H}(S_j)) = \mathcal{H}(S_i)\mathcal{H}(S_j) \tag{5}$$

As such, using Equation 5, we have:

$$lcm(\mathcal{H}(S_i), \mathcal{H}(S_j))^{-1} = \frac{gcd(\mathcal{H}(S_i), \mathcal{H}(S_j))}{\mathcal{H}(S_i)\mathcal{H}(S_j)} = \frac{b\mathcal{H}(S_i) + a\mathcal{H}(S_j)}{\mathcal{H}(S_i)\mathcal{H}(S_j)} = b\mathcal{H}(S_j)^{-1} + a\mathcal{H}(S_i)^{-1} \tag{6}$$

On the other side, we write $\mathcal{H}(S_i \cup S_j)$ as follows:

$$\mathcal{H}(S_i \cup S_j) = \prod_{a_k \in S_i \cup S_j} \mathcal{H}(a_k) = lcm(\prod_{a_k \in S_i} \mathcal{H}(a_k), \prod_{a_k \in S_j} \mathcal{H}(a_k)) = lcm(\mathcal{H}(S_i), \mathcal{H}(S_j)) \tag{7}$$

2.1 Proof of Homomorphism

In order to prove the homomorphism property with respect to the union operator, we first express $[{C_1}^{(i)}]^a \cdot [{C_1}^{(j)}]^b$, denoted by RS, as a function of $\mathcal{S}_i \cup \mathcal{S}_j$, sk_{is_i} and sk_{is_j}, as follows:

$$\begin{aligned} RS &= [x_{is_i} \cdot [{X_u}^{\mathcal{H}(\mathcal{S}_i)^{-1}}] \cdot {h_1}^{r_i}]^a \cdot [x_{is_j} \cdot [{X_u}^{\mathcal{H}(\mathcal{S}_j)^{-1}}] \cdot {h_1}^{r_j}]^b \\ &= {g_1}^{a.s_{is_i} + b.s_{is_j}} \cdot [{X_u}^{a\mathcal{H}(\mathcal{S}_i)^{-1} + b\mathcal{H}(\mathcal{S}_j)^{-1}}] \cdot {h_1}^{a.r_i + b.r_j} \\ &= {g_1}^{a.s_{is_i} + b.s_{is_j}} \cdot [{X_u}^{\mathcal{H}(\mathcal{S}_i \cup \mathcal{S}_j)^{-1}}] \cdot {h_1}^{a.r_i + b.r_j} \end{aligned}$$

Similarly, we can write the elements of the resulting credential C_R, such that $C_R = (C_{1,\mathcal{S}_i \cup \mathcal{S}_j}, C_{2,\mathcal{S}_i \cup \mathcal{S}_j}, C_{3,\mathcal{S}_i \cup \mathcal{S}_j}, \{C_{l,4,\mathcal{S}_i \cup \mathcal{S}_j}\}_{l \in [1,N]})$, where $C_{1,\mathcal{S}_i \cup \mathcal{S}_j} = [{C_1}^{(i)}]^a \cdot [{C_1}^{(j)}]^b = {x_{is_i}}^a \cdot {x_{is_j}}^b \cdot [{X_u}^{\mathcal{H}(\mathcal{S}_i \cup \mathcal{S}_j)}] \cdot {h_1}^{a.r_i + b.r_j}$, $C_{2,\mathcal{S}_i \cup \mathcal{S}_j} = [{C_2}^{(i)}]^a \cdot$

$[{C_2}^{(j)}]^b = {g_1}^{-(a.r_i+b.r_j)}$ $C_{3,\mathcal{S}_i \cup \mathcal{S}_j} = [{C_3}^{(i)}]^a \cdot [{C_3}^{(j)}]^b = {g_2}^{a.r_i+b.r_j}$ and $\{C_{l,4,\mathcal{S}_i \cup \mathcal{S}_j}\}_{l \in [1,N]} = \{{u_l}^{a.r_i+b.r_j}$, (i.e; N is the maximum number of attributes).

The form of the aggregated credential $C_{1,\mathcal{S}_i \cup \mathcal{S}_j}, C_{2,\mathcal{S}_i \cup \mathcal{S}_j}, C_{3,\mathcal{S}_i \cup \mathcal{S}_j}, \{C_{l,4,\mathcal{S}_i \cup \mathcal{S}_j}\}_{l \in [1,N]}$ is similar to the individual credentials like C_i, thus leading to the aggregated presentation token Σ_R by applying exactly the same $\mathcal{HABS}$Show algorithm. The obtained Σ_R is as follows: $\Sigma_R = (\Omega_R, \sigma_{1,R}, \sigma_{2,R}, C'_{1,R}, C'_{2,R}, A, \mathcal{S}_R)$.

2.2 Proof of Correctness

We show how the verifier can rely on the aggregated presentation token Σ_R, to authenticate the user (u), with respect to his access policy Υ, such as $\Upsilon(\mathcal{S}_i \cup \mathcal{S}_j) = 1$, where $\mathcal{S}_k$ presents the set of attributes certified by the issuer IS_k, $k \in \{i, j\}$. Using the properties of the pairing function $\hat{e}$, we can easily prove the correctness of Equation 8:

$$\hat{e}(\sigma_{1,R}, g_2) \overset{?}{=} {X_{is_i}}^a {X_{is_j}}^b \hat{e}(A_R, A) \hat{e}(C'_{2,R}, h_2) \prod_{i=1}^{l} \hat{e}(u_{\rho(i)} {h_1}^{\tau_i}, \omega_i) \hat{e}(\sigma_{2,R}, {g_2}^m) \tag{8}$$

where a and b are two integers as defined in Lemma 2.1.
By equivalence to Equation 2, we can consider that $D = a.r_i + b.r_j + r'$ presents the quantity $R = r + r'$.

$$\begin{aligned}
\hat{e}(\sigma_{1,R}, g_2) &= \hat{e}({x_{is_i}}^a {x_{is_j}}^b \cdot {X_u}^{\mathcal{H}(\mathcal{S}_i \cup \mathcal{S}_j)} \cdot {h_1}^D \cdot \prod_{i=1}^{l} (u_{\rho(i)})^{Dv_i} \cdot {g_1}^{r_m m}, g_2) \\
&= \hat{e}(x_{is_i}, g_2)^a \cdot \hat{e}(x_{is_j}, g_2)^b \cdot \hat{e}({X_u}^{\mathcal{H}(\mathcal{S}_i \cup \mathcal{S}_j)}, g_2) \cdot \hat{e}({h_1}^D, g_2) \cdot \hat{e}({g_1}^{r_m m}, g_2) \cdot \hat{e}(\prod_{i=1}^{l} {u_{\rho(i)}}^{Dv_i}, g_2) \\
&= \hat{e}(g_1, g_2)^{a.s_{is_i}} \cdot \hat{e}(g_1, g_2)^{b.s_{is_j}} \cdot \hat{e}({X_u}^{\mathcal{H}(\mathcal{S}_R \cup \mathcal{S}_H)^{-1}}, g_2) \cdot \hat{e}({h_1}^D, g_2) \cdot \hat{e}(\sigma_2, {g_2}^m) \cdot \prod_{i=1}^{l} \hat{e}({u_{\rho(i)}}^{Dv_i}, g_2) \\
&= {X_{is_i}}^a \cdot {X_{is_j}}^b \cdot \hat{e}({g_1}^{\mathcal{H}(\mathcal{S}_R)^{-1}}, [{g_2}^{sk_u}]^{\mathcal{H}(\mathcal{S}_H)^{-1}}) \cdot \hat{e}(C'_{2,R}, h_2) \cdot \hat{e}(\sigma_2, {g_2}^m) \cdot \prod_{i=1}^{l} \hat{e}(u_{\rho(i)}, \omega_i) \\
&= {X_{is_i}}^a {X_{is_j}}^b \hat{e}(A_R, A) \hat{e}(C'_{2,R}, h_2) \prod_{i=1}^{l} \hat{e}(u_{\rho(i)} {h_1}^{\tau_i}, \omega_i) \hat{e}(\sigma_{2,R}, {g_2}^m)
\end{aligned}$$

This proves the correctness of our $\mathcal{HABS}$.Verify, while considering a multi-issuers setting according to the agg algorithm.

References

[1] J. H. Ahn, D. Boneh, J. Camenisch, S. Hohenberger, A. Shelat, and B. Waters. Computing on authenticated data. In *Proc. of TCC*, LNCS, 2012.

[2] D. Boneh, X. Boyen, and E.-J. Goh. Hierarchical identity based encryption with constant size ciphertext. EUROCRYPT'05, 2005.

[3] D. Boneh, B. Lynn, and H. Shacham. Short signatures from the weil pairing. In *Proceedings of the 7th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology*, ASIACRYPT '01, 2001.

[4] S. Schage and J. Schwenk. A cdh-based ring signature scheme with short signatures and public keys. FC'10, 2010.

[5] B. Waters. Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization. PKC'11, 2011.