T-79.512 Cryptology: Special Topics / Pairing-based Cryptography:
Comments on "Aggregate Signatures from Bilinear Pairings"

Johan Wallén

March 25, 2003

1 Introduction

In these notes, we give a brief review of the survey [2] on aggregate signatures from bilinear pairings based on [1]. All comments refer to a draft of the survey dated 16.3.2003.

The survey starts with a short discussion of the computational co-Diffie-Hellman (co-CDH) problem and bilinear groups for co-CDH. This is followed by a description of the aggregate signature scheme. After that, security against aggregate forgery is defined and the signature scheme is proven to achieve this security definition. The survey closes with some practical applications of aggregate signature schemes.

The selection of topics in the survey is in general good, and the level of detail is suitable. It would, however, be nice with a short discussion about the verifiably encrypted signature schemes and ring signatures from [1]—at least they could be included in the section on applications.

2 More detailed comments

Section 1 (Introduction). The domain of the bilinear map $e\colon G_1 \times G_2 \to G_\mathrm{T}$ in the second paragraph should probably be $G_1 \times G_2$ and not G_1^2.

Section 2 (Required definitions and notation). The bilinear maps and isomorphism should probably be efficiently computable and not just computable. (On the other hand, the computational model is normalised so that both can be computed in one time unit.)

Note that the isomorphism $\psi\colon G_1 \to G_2$ is actually unique. Use group operation instead of group action in Definition 2.3 (to avoid confusion with a group acting on some set). The section heading ends with a period. This is probably not on purpose. The word 'problem' is missing from the computational and decisional Diffie-Hellman problems just before their definitions.

Section 3 (Bilinear aggregate signature scheme). The notation $\mathbb{U}$ and $\overline{1,k}$ should probably be explained, although it is quite clear from context.

Section 4 (Security definition). The definition of the advantage $\mathrm{adv}_\mathcal{A}$ is missing from Definition 4.1. It would also be good to use different notations for the advantage of solving co-CDH and the advantage of existential forgery in the aggregate chosen-key model.

One of the ts in Theorem 3.1 should probably be a t'. Why is the Theorem 3.1 numbered 3.1?

Section 5 (Security proof). It is claimed that $\mathcal{A}$ always will ask for the hash of a message before it asks for its signature. Mention that this holds by assumption, and that it is not a loss of generality, since any adversary can be modified to have this property (but how does this affect the advantage?). It is also assumed that $\mathcal{A}$ always asks for the hash of the messages it outputs as its forgery.

In the description of the bookkeeping of $\mathcal{C}$, note that the messages $M^{(i)}$ are unique. That is, when a message M arrives, $\mathcal{C}$ first checks if it is one of the previously received messages.

In the first paragraph of the second column on page 3, it is stated that $\mathcal{A}$ should have asked for the hash of every message in the aggregate. Note that this only holds due to the implicit assumption about the behaviour of $\mathcal{A}$. (See above.)

The description of how σ_1 is computed and why it has the desired value is a little bit messy. A clearer distinction between what was chosen by $\mathcal{C}$ before the simulation, what was computed by $\mathcal{C}$ after the simulation, and what follows from the fact that $\mathcal{A}$ succeeded in its forgery would make things easier to follow. Note that $e(g_1, \sigma_i) = e(v_i, h_i)$ only for $i \in \{2, \ldots, k\}$ (since σ_1 has not been computed yet).

There is a small mistake in the description of how the solution to the co-CDH instance is computed: σ_1 should be divided by $\psi(g_1^a)^{r_1}$ and not by $\psi(g_1)^{r_1}$.

When analysing the success probability of $\mathcal{C}$, explain why $\mathcal{A}$ behaves exactly as it would in a normal execution (provided that the simulation is not aborted).

The analysis of the running time is a bit confusing. Is t or t' the running time of $\mathcal{A}$? Is the bound $t' \leq t - 2c_{\mathcal{C}} \log_2 p(q_{\mathrm{H}} + q_{\mathrm{S}} + N + 2)$ correct? In the last row on page 3, '$\mathcal{C}$ may need to complete' should be '$\mathcal{C}$ can complete'. On page 4, the maximum number of exponentiations is missing from '$\mathcal{C}$ may need to perform such operations'. Also, 'group action' should be 'group operation'.

The security theorem is accidently called Theorem 3.2 in the proof.

Bibliography. Mention that [1] will be presented at Eurocrypt 2003. The are some typos in course code and name (T-79.512 Cryptography: Special Topics).

Language. Pay attention to the use of articles (the, a, an). For example, 'If the number of users' on row 5 of the first paragraph on page 1 and 'The adversary also has access to a signing oracle' on rows 5–6 in the third paragraph on page 1. There are also some typos (like 'pubic key' in the verification step on page 2, and 'Deffie-Hellman' and 'poor' in the third paragraph on page 1).

Displayed equations. Displayed equations (as produced by `\[...\]` or the `align*` environment) would make the formulas easier to read.

References

[1] Dan Boneh, Craig Gentry, Ben Lynn, and Hovav Shacham. Aggregate and verifiably encrypted signatures from bilinear maps. Cryptology ePrint Archive, Report 2002/175, 2002. http://eprint.iacr.org/.

[2] Tatiana Issaeva. Aggregate signatures from bilinear pairings. Survey for the seminar T-79.512: Cryptology: Special Topics / Pairing Based Cryptography at HUT, 16 March 2003.