# A strategy for any DAA Issuer and an additional verification by a Host

Vadym Fedyukovych

June 15, 2008

**Abstract**

A successful strategy was identified for any Verifier colluding with any Issuer to distinguish honest Provers issuing DAA signatures. An additional verification equation was introduced for a Prover to detect 'tagged' credentials that may be issued while Join protocol. This verification can be done by the Host and do not affect TPM in any way.

## 1 Introduction

Direct Anonymous Authentication (DAA) [BCC04] is a promising pair of protocols that may be very useful for verifying statements regarding computer hardware. The major point of DAA is impossibility of linking signatures to credentials issued, stated as a design goal. This allows users of DAA to avoid profiling their online activities through analysis of their signatures.

At a glance, an *Issuer* produces *credentials* for a *Prover* while Join protocol, and a Prover produces a signature while Sign protocol to authenticate to a *Verifier* in return for a service. Prover is considered to be a *Host* (a general-purpose computer hardware and operating system) and a *TPM* (a tamper-resistant chip with a limited and strictly controlled functionality). We refer to original DAA specifications [BCC04] for all the details.

We say a party to a protocol is *honest* if he follows protocol specifications. We say a party is *honest but curious* if protocol transcript with such a party is indistinguishable from protocol transcript with a honest party. We say *any* party otherwise.

A game of *User-Controlled-Traceability* was introduced [BCL08] to facilitate formal analysis of DAA protocols. However, security model considered only include any Host and any TPM.

We analyse options available to any Issuer while Join protocol. We observe any Issuer may choose to produce credentials that allow any Verifier to decide whether a signature presented was produced by a Prover that was issued credentials in the specific instance of Join protocol. We conclude a strategy exists for a Verifier colluding with an Issuer to always win a variant of User-Controlled-Traceability game. We say such a credentials issued according to the strategy discovered are *tagged*. We introduce an additional verification equation to be tested by the Host to always reject tagged credentials.

# 2 A strategy for any Issuer

We observe an argument of knowledge protocol running by Issuer for Prover at the end of Join protocol do not imply validity of credentials issued (that is, validity of CL signature). A honest Prover is using credentials issued to produce signatures without verifying validity of credentials. We also observe a Verifier may accept signatures produced with invalid credentials. Even better, a Verifier may try an alternative equation for a failed signature using a list of tags issued while some instances of Join protocol.

In particular, credentials produced at step 7 of Join (using notations of [BCC04] are $(e, A, v'')$ and the tag is

$$\bar{Z} = A^e U S^{v''}, \qquad \bar{Z} \neq Z \tag{1}$$

where $U = R_0^{f_0} R_1^{f_1} S^{v'}$. We observe $\bar{Z}$ should be in the group generated by $h$ for the protocol of Issuer to be successful.

Prover sets $v = v'' + v'$ as part of his credentials. To produce a signature, Prover (step 2 of Sign protocol) chooses some $w$, produces $T_1 = Ah^w$. Prover produces a transcript of proof protocol to show that

$$Z = T_1^e R_0^{f_0} R_1^{f_1} S^v h^{-ew} \tag{2}$$

We observe that an alternative equation

$$\bar{Z} = T_1^e R_0^{f_0} R_1^{f_1} S^v h^{-ew} \tag{3}$$

actually holds for credentials tagged with $\bar{Z}$. We suggest that a Verifier presented with a signature that fails passing (2) may also try (3) with a watchlist

of $\bar{Z}$ values. In particular, Verifier produces

$$\bar{T}_1 = \bar{Z}^{-c} T_1^{s_e + c2^{l_e-1}} R_0^{s_{f0}} R_1^{s_{f1}} S^{s_v} h^{-s_{ew}} \tag{4}$$

and tests whether $c$ can be re-produced according to Fiat-Shamir with $\bar{T}_1$ in place of $\hat{T}_1$. We conclude such a Verifier can always recognise signatures produced with tagged credentials.

# 3 Additional verification equation

A Host running a Join with an Issuer may test whether (1) holds and reject for any alternative $\bar{Z} \neq Z$.

# 4 Acknowledgements

Author thanks Dr. Camenisch and Dr. Jiangtao for discussion.

# References

[BCC04] Ernie Brickell, Jan Camenisch, and Liqun Chen. Direct anonymous attestation. Cryptology ePrint Archive, Report 2004/205, 2004. `http://eprint.iacr.org/`.

[BCL08] Ernie Brickell, Liqun Chen, and Jiangtao Li. Simplified security notions of direct anonymous attestation and a concrete scheme from pairings. Cryptology ePrint Archive, Report 2008/104, 2008. `http://eprint.iacr.org/`.