

Trust-Rated Authentication for Domain-Structured Distributed Systems

R. Holz, H. Niedermayer, P. Hauck, G. Carle

Wilhelm-Schickard-Institut für Informatik
Eberhard Karls Universität Tübingen, Germany

`lastname@informatik.uni-tuebingen.de`

EuroPKI 2008
June 16, 2008



Table of Contents

1. Introduction (PKIs for P2P)
2. Cross-Domain Authentication
3. Trust-Rated Cross-Domain Authentication
4. Security Verification
5. Conclusions

Part I

Introduction

Authentication in Decentralized Settings

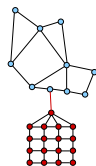
Theorem (Secure Channels for Authentication – Need for TTPs)

'It is not possible to establish an authenticated session key without existing secure channels already being available.' (Boyd, 1993)

Unsolvable problem for fully decentralized scenarios – there is no a priori context.

Commonly found approaches

- ▶ Hierarchical PKIs with CAs (e.g.X.509)
 - ▶ Centralized
- ▶ 'Flat' PKIs (Webs of Trust)
 - ▶ Sybil-like infiltration (Douceur, 2002)



Do these actually meet our requirements?

A CA/TTP is associated with a purpose.

- ▶ Can the same global CA/TTP be used for all purposes?

Applying global CAs to P2P and spontaneous networks...

- ▶ “It is centralized... (live with it?)”
- ▶ Do I *want* to authenticate with my global ID – even for a local context?
 - ▶ “I am a member in this P2P network”? (→ privacy)
- ▶ Anyway, what does this mean: “A CA has verified my identity...”?
 - ▶ It means the CA must ‘know’ every entity

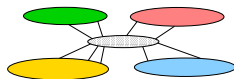
Introducing Domains

It is a question of scale.

- ▶ Huge P2P networks difficult to control
- ▶ Can be easier for smaller networks
 - ▶ May also reflect social or trust relations (cf. 'dark-nets')

We consider domain-based P2P a good compromise.

- ▶ Make establishing TTPs easier
- ▶ 'Semi-federation': together a global network of domains ('semi-federation')



Part II

Scheme for Cross-Domain Authentication

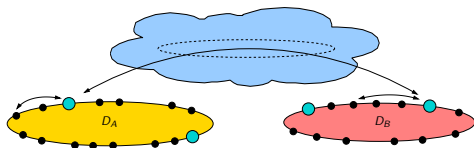
A Scheme for Cross-Domain Authentication

Domain Authentication Servers (DAS)

- ▶ $DAS = \text{Trusted Third Parties} \rightarrow$ 4-party authentication

We can distinguish two cases:

- ▶ Channel between DAS may be *secure* or *not secure*
- ▶ If not secure, we'll need to do something about this – Trust-Rating
- ▶ We'll get back to this later
- ▶ First discuss case of secure channel



We have the following protocol goals:

- ▶ Authentication as Fresh Agreement (cf. G. Lowe, 1997)
- ▶ Key Establishment
- ▶ Freshness of the Session Key
- ▶ Mutual Belief in Key

Notation and Assumptions

Principal tokens

- ▶ **Keys established** between DAS \leftrightarrow peers
- ▶ **Public Key Tokens:** $Sig_{S_B}(h(B, PK_B))$
- ▶ **Credentials:** $Sig_{S_B}(h(B, A, PK_B, N_{S_B}))$
- ▶ **Nonces:** secure each two-party communication

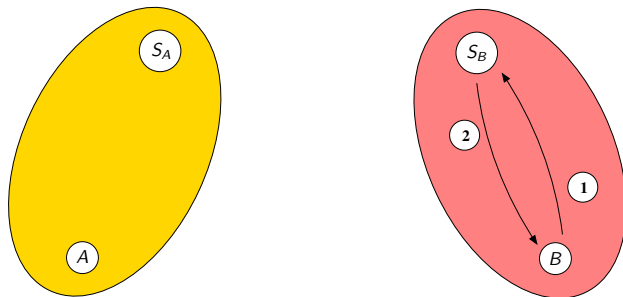
Exchange of keys & Public Key Token between peers

$B \rightarrow A : (B, A, PK_B, Sig_{S_B}(h(B, PK_B)))$ (Key Exchange Query)

$A \rightarrow B : (A, B, PK_A, Sig_{S_A}(h(A, PK_A)))$ (Key Exchange Reply)

Protocol Steps 1 and 2

Credential Request and Credential Grant

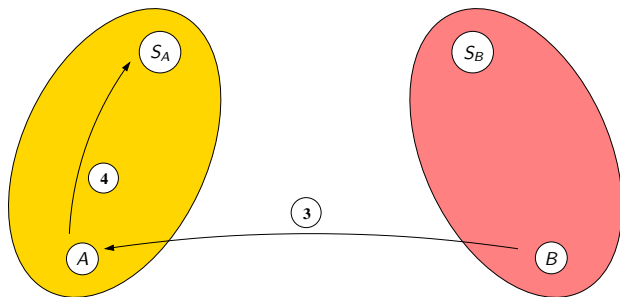


$$B \rightarrow S_B : \{B, S_B, A, PK_A, \text{Sig}_{S_A}(h(A, PK_A)), N_B\}_{K_{BSB}} \quad (1)$$

$$S_B \rightarrow B : \{S_B, B, \text{Sig}_{S_B}(h(B, A, PK_B, N_{S_B})), N_B, N_{S_B}\}_{K_{BSB}} \quad (2)$$

Protocol Steps 3 and 4

Credential Forwards

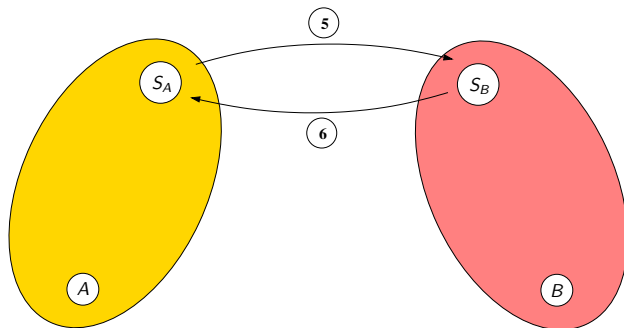


$$B \rightarrow A : E_A(B, A, S_B, \text{Sig}_{S_B}(h(B, A, PK_B, N_{S_B})), N_B, N_{S_B}) \quad (3)$$

$$A \rightarrow S_A : \{A, S_A, B, S_B, PK_B, \text{Sig}_{S_B}(h(B, A, PK_B, N_{S_B})), N_{S_B}, N_A\}_{K_{BS_B}} \quad (4)$$

Protocol Steps 5 and 6

Freshness Verification and Authentication Token for A

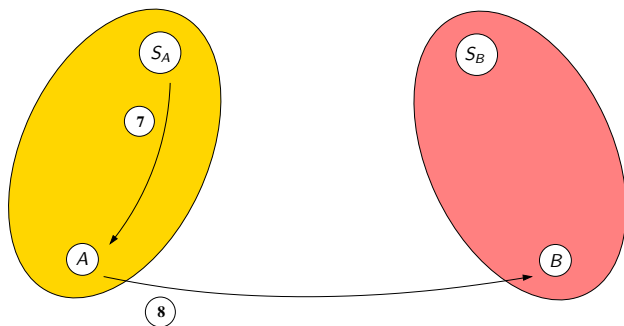


$$S_A \rightarrow S_B : E_{S_B}(S_A, S_B, B, A, N_{S_B}, N_{S_A}) \quad (5)$$

$$S_B \rightarrow S_A : E_{S_A}(S_B, S_A, \text{Sig}_{S_B}(h(A, N_{S_B})), N_{S_A}) \quad (6)$$

Protocol Steps 7 and 8

Authentication Decision and Completion



$$S_A \rightarrow A : \{S_A, A, \text{Sig}_{S_B}(h(A, N_{S_B})), N_A\}_{K_{BS_B}} \quad (7)$$

$$A \rightarrow B : E_B(A, B, \text{Sig}_{S_B}(h(A, N_{S_B})), N_B, g_A, p, DH_A) \quad (8)$$

$$B \rightarrow A : E_A(B, A, N_B, DH_B) \quad (9)$$

Part III

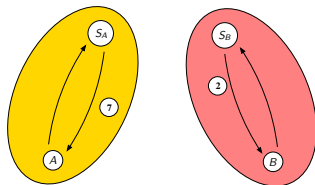
Extension: Trust-Rated Authentication

Trust Tokens for Trust-Rated Authentication

What if there is no secure channel between domains?

- ▶ Quite a common case in P2P and Ad-hoc, less in industry

Idea: DAS include Trust Token for their clients



$$S_B \rightarrow B : \{S_B, B, \text{Sig}_{S_B}(h(B, A, PK_B, N_{S_B})), N_B, N_{S_B}, \text{trust}_A\}_{K_{BS_B}} \quad (2)$$

$$S_A \rightarrow A : \{S_A, A, \text{Sig}_{S_B}(h(A, N_{S_B})), N_A, \text{trust}_B\}_{K_{BS_B}} \quad (7)$$

Trust Token: standardized set of properties describing DAS, domain, peer

- ▶ E.g. secure channel (yes/no), a priori knowledge, prior contacts
- ▶ Use this information to support authentication decision: *trust-rated*

Clients evaluate the Trust Token

- ▶ Check the information in the Trust Token – does it match my security requirements?
- ▶ Does not create a secure channel – but provides a ‘risk assessment’

Trust Token (Example)

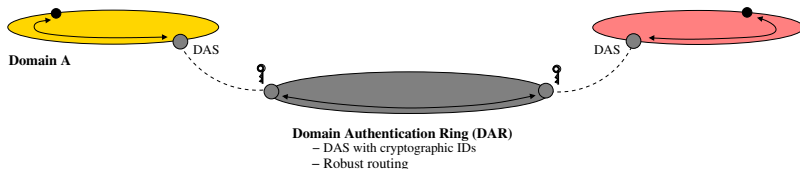
SecureChannelToDAS KeyExchangeWithDAS KeyExchangeWithDAS	No MultiplePathLookup RedundantRouting
OtherDAS OtherDAS	SelfCertifyingID: yes WellKnownIPRange: yes
OtherDomain OtherDomain OtherDomain	PriorContacts: yes(10) HumanFeedback: yes(3)/no(0) KnownFrauds: no(0)
OtherPeer OtherPeer OtherPeer OtherPeer	SelfCertifyingID: no PriorContacts: no(0) KnownFrauds: no(0) HumanFeedback: yes(0)/no(0)

Note: This way, DAS can build up trust over time.

Adding a few ideas...

Strengthening inter-domain channels

- ▶ Multiple-path routing (limits chances of MITM)
- ▶ Self-certifying IDs for DAS/domains
- ▶ Incorporate user feedback to DAS



Part IV

Security Evaluation

We only investigate the case of a secure channel between the DAS.

- ▶ Everything else is pointless as no authenticated session can be achieved.

We modelled with the AVISPA Model Checker.

- ▶ Up to three parallel sessions (max acceptable time)
- ▶ Models with just two sessions actually found all attacks

→ **Protocol found safe**

Part V

Conclusions

Conclusions

We presented a concept for trust-rated authentication between domains

- ▶ Meant for P2P and ad-hoc use case
- ▶ But inner working of a domain actually not relevant

General Properties

- ▶ Concept positions itself between PKIs and Webs of Trust
- ▶ Better suited to P2P and spontaneous networks
- ▶ 'Decoupling' domains helps with scalability

Cryptographic Properties

- ▶ Authentication, Key Establishment, Perfect Forward Secrecy
- ▶ Key Revocation and Update

Thank you!

Questions?