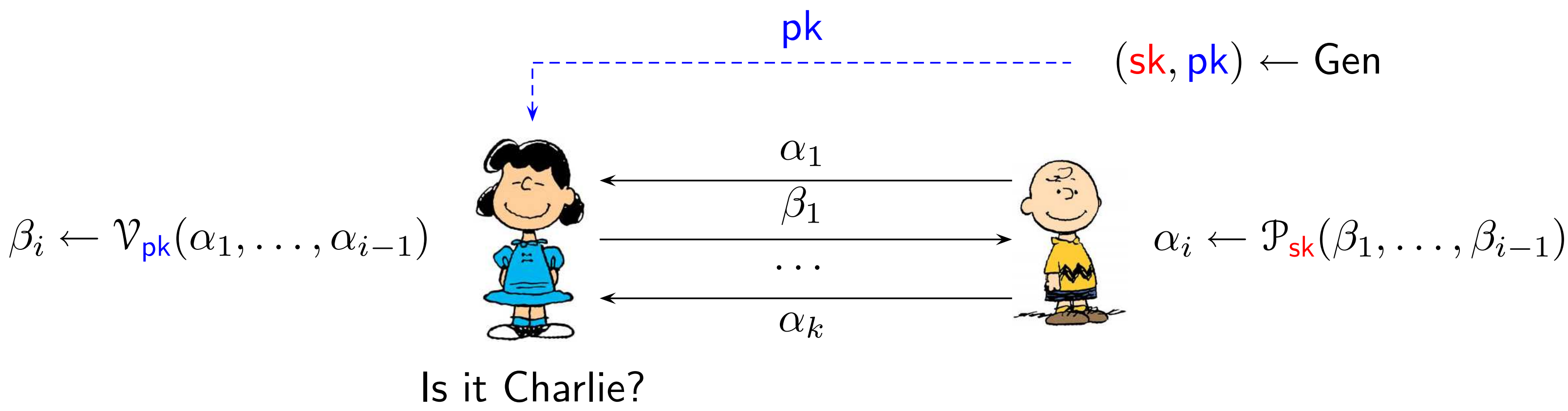
MTAT.07.003 Cryptology II

Entity Authentication

Sven Laur
University of Tartu

Formal Syntax

Entity authentication



- The communication between the prover and verifier must be authentic.
- To establish electronic identity, Charlie must generate $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}$ and convinces others that the public information pk represents him.
- The entity authentication protocol must convince the verifier that his or her opponent possesses the secret sk.
- An entity authentication protocol is *functional* if an honest verifier $\mathcal{V}_{\mathsf{pk}}$ always accepts an honest prover $\mathcal{P}_{\mathsf{sk}}$.

Classical impossibility results

Inherent limitations. Entity authentication is impossible

(i) **if** authenticated communication is unaffordable in the setup phase;

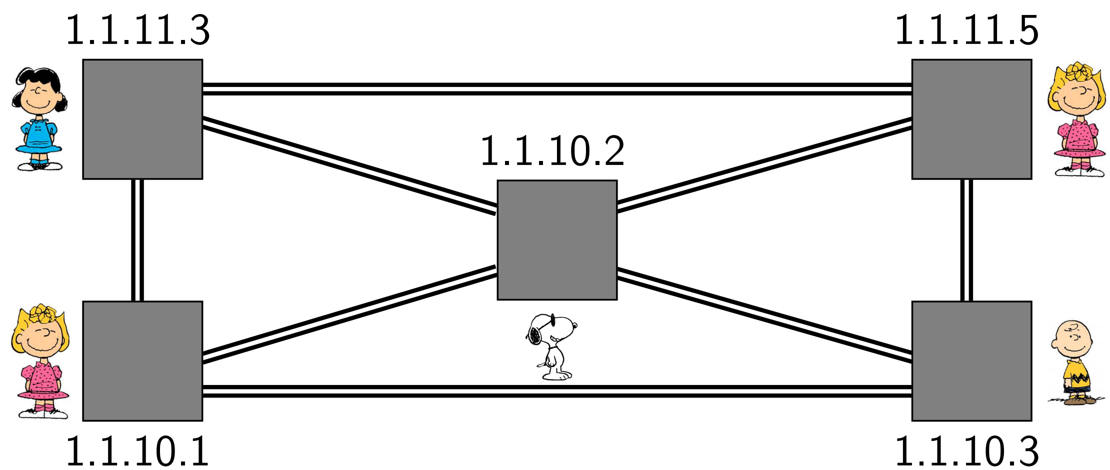
(ii) **if** authenticated communication is unaffordable in the second phase.

Proof. Man-in-the-middle attacks. Chess-master attacks.

Conclusions

- ▷ It is impossible to establish legal identity without physical measures.
- ▷ Any smart card is susceptible to physical attacks regardless of the cryptographic countermeasures used to authenticate transactions.
- ▷ Secure e-banking is impossible if the user does not have full control over the computing environment (secure e-banking is practically impossible).

Physical and legal identities



- ▷ Entity authentication is possible only if all participants have set up a network with authenticated communication links.
- ▷ A role of a entity authentication protocol is to establish a convincing bound between physical network address and legal identities.
- ▷ A same legal identity can be in many physical locations and move from one physical node to another node.

Challenge-Response Paradigm

Salted hashing

Global setup:

Authentication server $\mathcal{V}$ outputs a description of a hash function h.

Entity creation:

A party $\mathcal{P}$ chooses a password sk $\xleftarrow[u]{} \{0,1\}^\ell$ and a nonce $r \xleftarrow[u]{} \{0,1\}^k$. The public authentication information is pk $= (r, c)$ where $c \leftarrow h(\mathsf{sk}, r)$.

Entity authentication:

To authenticate him- or herself, $\mathcal{P}$ releases sk to the server $\mathcal{V}$ who verifies that the hash value is correctly computed, i.e., $c = h(\mathsf{sk}, r)$.

Theorem. If h is (t, ε)-secure one-way function, then no t-time adversary $\mathcal{A}$ without sk can succeed in the protocol with probability more than ε.

- ▷ There are no secure one-way functions for practical sizes of sk.
- ▷ A malicious server can completely break the security.

RSA based entity authentication

Global setup:
Authentication server $\mathcal{V}$ fixes the minimal size of RSA keys.

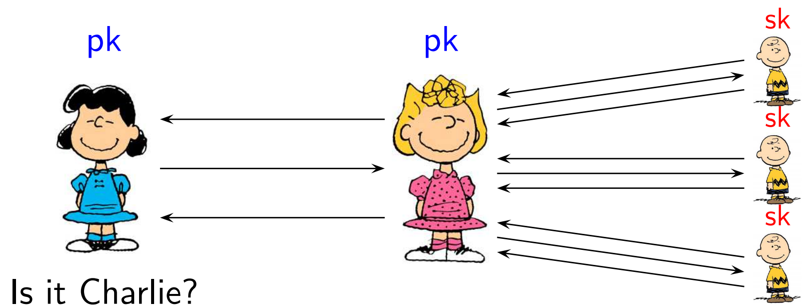
Entity creation:
A party $\mathcal{P}$ runs a RSA key generation algorithm $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}_{\mathrm{rsa}}$ and outputs the public key pk as the authenticating information.

Entity authentication:

1. $\mathcal{V}$ creates a challenge $c \leftarrow \mathsf{Enc}_{\mathsf{pk}}(m)$ for $m \xleftarrow{u} \mathcal{M}$ and sends c to $\mathcal{P}$.
2. $\mathcal{P}$ sends back $\overline{m} \leftarrow \mathsf{Dec}_{\mathsf{sk}}(c)$.
3. $\mathcal{V}$ accepts the proof if $m = \overline{m}$.

This protocol can be generalised for any public key cryptosystem.
The general form of this protocol is known as *challenge-response protocol*.
This mechanism provides explicit security guarantees in the TLS protocol.

The most powerful attack model



Consider a setting, where an adversary $\mathcal{A}$ can impersonate verifier $\mathcal{V}$

- ▷ The adversary $\mathcal{A}$ can execute several protocol instances with the honest prover $\mathcal{P}$ in parallel to spoof the challenge protocol.
- ▷ The adversary $\mathcal{A}$ may use protocol messages arbitrarily as long as $\mathcal{A}$ does not conduct the crossmaster attack.

Let us denote the corresponding success probability by

$$\mathsf{Adv}^{\mathsf{ent\text{-}auth}}(\mathcal{A}) = \Pr\left[(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen} : \mathcal{V}^{\mathcal{A}} = 1\right] \enspace .$$

Corresponding security guarantees

Theorem. If a cryptosystem used in the challenge-response protocol is (t, ε)-IND-CCA2 secure, then for any t-time adversary $\mathcal{A}$ the corresponding success probability $\mathsf{Adv}^{\mathsf{ent\text{-}auth}}(\mathcal{A}) \leq \frac{1}{|\mathcal{M}|} + \varepsilon$.

Proof. A honest prover acts as a decryption oracle.

The nature of the protocol

- ▷ The protocol proves only that the prover has access to the decryption oracle and therefore the prover must *possess* the secret key sk.
- ▷ The possession of the secret key sk does not imply the *knowledge* of it. For example, the secret key sk might be hardwired into a smart card.
- ▷ Usually, the inability to decrypt is a strictly stronger security requirement than the ability to find the secret key.
- ▷ *Knowledge* is permanent whereas *possession* can be temporal.

Proofs of knowledge

Schnorr identification protocol

$y = g^{x}$ $\qquad\qquad$ $x \in \mathbb{Z}_q$

$\beta \xleftarrow[u]{} \mathbb{Z}_q$  $\xleftarrow{\alpha = g^{k}}$ $\xrightarrow{\beta}$ $\xleftarrow{\gamma = k + \beta x}$  $k \xleftarrow[u]{} \mathbb{Z}_q$

$$g^{\gamma} = g^{k} g^{\beta x} \stackrel{?}{=} \alpha y^{\beta}$$

The group $\mathbb{G} = \langle g \rangle$ must be a DL group with a prime cardinality q.

- ▷ The secret key x is the discrete logarithm of y.
- ▷ The verifier $\mathcal{V}$ is assumed to be semi-honest.
- ▷ The prover $\mathcal{P}$ is assumed to be potentially malicious.
- ▷ We consider only security in the standalone setting.

Zero-knowledge principle

sk

$$\xrightarrow{\alpha(\mathsf{sk})}$$

$$\xleftarrow{\beta}$$

$$\xrightarrow{\gamma(\mathsf{sk},\alpha,\beta)}$$

pk

Even if Lucy is *honest*

▷ she might learn something about the secret sk.

since

▷ messages α and γ depend on the secret sk.

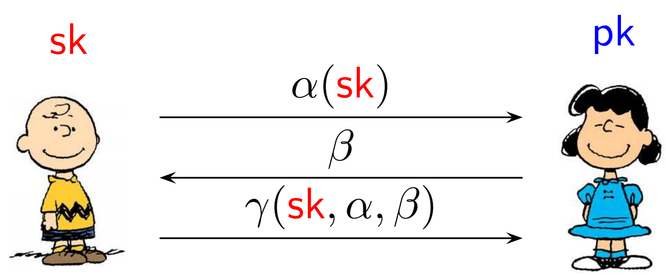
sk

pk

If Lucy *does not* interact with Charlie then nothing about the secret sk is revealed

Lucy should be equally *successful* in both experiments.

Simulation principle

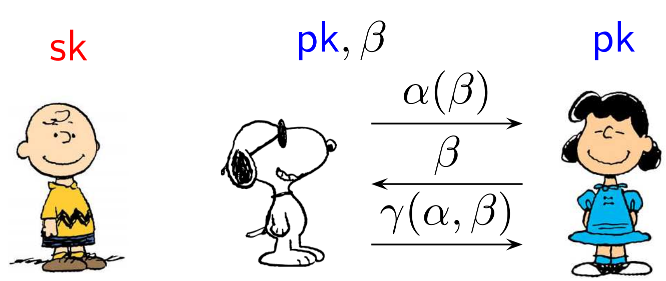


Even if Lucy is *honest*

- ▷ she might learn something about the secret sk.

since

- ▷ messages α and γ depend on the secret sk.



Since Lucy is *honest* the value of β is known by her before the protocol and Snoopy can use pk and β to simulate the other messages.

Lucy should not be able to distinguish between these two experiments.

Zero-knowledge property

Theorem. If a t-time verifier $\mathcal{V}_*$ is semi-honest in the Schnorr identification protocol, then there exists $t + \mathrm{O}(1)$-algorithm $\mathcal{V}_\circ$ that has the same output distribution as $\mathcal{V}_*$ but do not interact with the prover $\mathcal{P}$.

Proof.
Consider a code wrapper $\mathcal{S}$ that chooses $\beta \xleftarrow[u]{} \mathbb{Z}_q$ and $\gamma \xleftarrow[u]{} \mathbb{Z}_q$ and computes $\alpha \leftarrow g^\gamma \cdot y^{-\beta}$ and outputs whatever $\mathcal{V}_*$ outputs on the transcript (α, β, γ).

▷ If $x \neq 0$, then $\gamma = \beta + xk$ has indeed a uniform distribution.

▷ For fixed β and γ, there exist only a single consistent value of α.

□

Rationale: Semi-honest verifier learns nothing from the interaction with the prover. The latter is known as *zero-knowledge* property.

Knowledge-extraction lemma

Given two runs with a coinciding prefix α

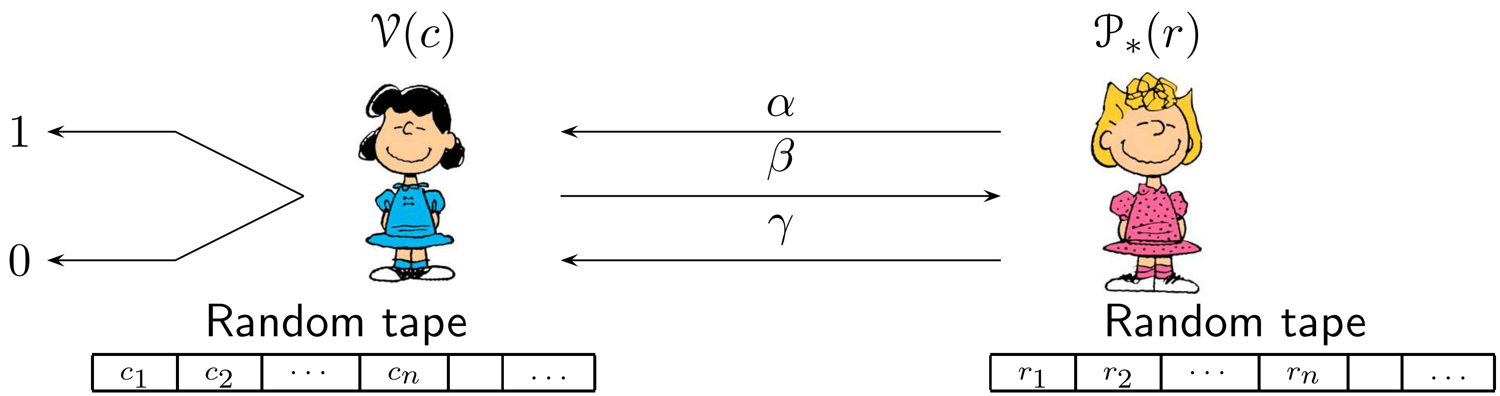
$$\begin{tikzcd}[column sep=small] & \alpha = g^{k} \arrow[dl, "\beta"'] \arrow[dr, "\beta'"] & \\ \gamma = k + \beta x & & \gamma' = k + \beta' x \end{tikzcd}$$

We can extract the secret key $x = \frac{\gamma - \gamma'}{\beta - \beta'}$.

This property is known as *special-soundness.*

- ▷ If adversary $\mathcal{A}$ succeeds with probability 1, then we can extract the secret key x by rewinding $\mathcal{A}$ to get two runs with a coinciding prefix α.
- ▷ If adversary $\mathcal{A}$ succeeds with a non-zero probability ε, then we must use more advanced knowledge-extraction techniques.

Find two ones in a row



Let $A(r, c)$ be the output of the honest verifier $\mathcal{V}(c)$ that interacts with a potentially malicious prover $\mathcal{P}_*(r)$.

- ▷ Then all matrix elements in the same row $A(r, \cdot)$ lead to same α value.
- ▷ To extract the secret key sk, we must find two ones in the same row.
- ▷ We can compute the entries of the matrix on the fly.

We derive the corresponding security guarantees a *bit later*.

Modified Fiat-Shamir identification protocol

$v = s^2$ $\qquad\qquad\qquad\qquad$ $s \in \mathbb{Z}_n^*$

$\beta \xleftarrow[u]{} \{0,1\}$

$\xleftarrow{\alpha = r^2}$

$\xrightarrow{\beta}$

$\xleftarrow{\gamma = rs^{\beta}}$

$r \xleftarrow[u]{} \mathbb{Z}_n^*$

Halt if $\gamma \notin \mathbb{Z}_n^*$

$\gamma^2 = r^2 s^{2\beta} \stackrel{?}{=} \alpha v^{\beta}$

All computations are done in $\mathbb{Z}_n$, where n is an RSA modulus.

- ▷ The secret key s is a square root of v.
- ▷ The verifier $\mathcal{V}$ is assumed to be semi-honest.
- ▷ The prover $\mathcal{P}$ is assumed to be potentially malicious.
- ▷ We consider only security in the standalone setting.

Zero-knowledge property

Theorem. If a t-time verifier $\mathcal{V}_*$ is semi-honest in the modified Fiat-Shamir identification protocol, then there exists $t + \mathrm{O}(1)$-algorithm $\mathcal{V}_\circ$ that has the same output distribution as $\mathcal{V}_*$ but do not interact with the prover $\mathcal{P}$.

Proof.
Consider a code wrapper $\mathcal{S}$ that chooses $\beta \xleftarrow[u]{} \{0,1\}$, $\gamma \xleftarrow[u]{} \mathbb{Z}_n^*$, computes $\alpha \leftarrow v^{-\beta} \cdot \gamma^2$ and outputs whatever $\mathcal{V}_*$ outputs on the transcript (α, β, γ).

- ▷ Since s is invertible, we can prove that $s \cdot \mathbb{Z}_n^* = \mathbb{Z}_n^*$ and $s^2 \cdot \mathbb{Z}_n^* = \mathbb{Z}_n^*$. As a result, γ is independent of β and has indeed a uniform distribution.
- ▷ For fixed β and γ, there exist only a single consistent value of α.

□

Knowledge-extraction lemma

Theorem. The Fiat-Shamir protocol is specially sound.

Proof. Assume that a prover $\mathcal{P}_*$ succeeds for both challenges $\beta \in \{0, 1\}$:

$$\gamma_0^2 = \alpha, \quad \gamma_1^2 = \alpha v \qquad \Longrightarrow \qquad \frac{\gamma_1}{\gamma_0} = \sqrt{v} \ .$$

The corresponding extractor construction $\mathcal{K}$:

- ▷ Choose random coins r for $\mathcal{P}_*$.
- ▷ Run the protocol with $\beta = 0$ and record γ_0
- ▷ Run the protocol with $\beta = 1$ and record γ_1
- ▷ Return $\zeta = \frac{\gamma_1}{\gamma_0}$

Bound on success probability

Theorem. Let v and n be fixed. If a potentially malicious prover $\mathcal{P}_*$ succeeds in the modified Fiat-Shamir protocol with probability $\varepsilon > \frac{1}{2}$, then the knowledge extractor $\mathcal{K}^{\mathcal{P}_*}$ returns $\sqrt{v}$ with probability $\varepsilon - \frac{1}{2}$.

Proof. Consider the success matrix $A(r, c)$ as before. Let p_1 denote the fraction rows that contain only single one and p_2 the fraction of rows that contain two ones. Then evidently $p_1 + p_2 \leq 1$ and $\frac{p_1}{2} + p_2 \geq \varepsilon$ and thus we can establish $p_2 \geq \varepsilon - \frac{1}{2}$. $\square$

Rationale: The knowledge extraction succeeds in general only if the success probability of $\mathcal{P}_*$ is above $\frac{1}{2}$. The value $\kappa = \frac{1}{2}$ is known as *knowledge error*.

Matrix Games

Classical algorithm

Task: Find two ones in a same row.

Rewind:

1. Probe random entries $A(r, c)$ until $A(r, c) = 1$.
2. Store the matrix location (r, c).
3. Probe random entries $A(r, \overline{c})$ in the same row until $A(r, \overline{c}) = 1$.
4. Output the location triple $(r, c, \overline{c})$.

Rewind-Exp:

1. Repeat the procedure Rewind until $c \neq \overline{c}$.
2. Use the knowledge-extraction lemma to extract sk.

Average-case running time

Theorem. If a $m \times n$ zero-one matrix A contains ε-fraction of nonzero entries, then the Rewind and Rewind-Exp algorithm make on average

$$\mathbf{E}[\text{probes}|\text{Rewind}] = \frac{2}{\varepsilon}$$

$$\mathbf{E}[\text{probes}|\text{Rewind-Exp}] = \frac{2}{\varepsilon - \kappa}$$

probes where $\kappa = \frac{1}{n}$ is a *knowledge error*.

Proof. We prove this theorem in another lecture.

Strict time bounds

Markov's inequality assures that for a non-negative random variable probes

$$\Pr[\text{probes} \geq \alpha] \leq \frac{\mathbf{E}[\text{probes}]}{\alpha}$$

and thus Rewind-Exp succeeds with probability at least $\frac{1}{2}$ after $\frac{4}{\varepsilon-\kappa}$ probes.

If we repeat the experiment ℓ times, we the failure probability goes to $2^{-\ell}$.

From Soundness to Security

Soundness and subjective security

Assume that we know a constructive proof:

If for fixed pk a potentially malicious t-time prover $\mathcal{P}_*$ succeeds with probability $\varepsilon > \kappa$, then a knowledge extractor $\mathcal{K}^{\mathcal{P}}$ that runs in time $\tau(\varepsilon) = \mathrm{O}\big(\frac{t}{\varepsilon-\kappa}\big)$ outputs sk with probability $1 - \varepsilon_2$.

and we *believe*:

No human can create a $\tau(\varepsilon_1)$-time algorithm that computes sk from pk with success probability at least $1 - \varepsilon_2$.

then it is *rational* to assume that:

No human without the knowledge of sk can create a algorithm $\mathcal{P}_*$ that succeeds in the proof of knowledge with probability at least ε_1.

Caveat: For each fixed pk, there exists a trivial algorithm that prints out sk. Hence, we cannot get objective security guarantees.

Soundness and objective security

Assume that we know a constructive proof:

If for a fixed pk a potentially malicious t-time prover $\mathcal{P}_*$ succeeds with probability $\varepsilon > \kappa$, then a knowledge extractor $\mathcal{K}^{\mathcal{P}}$ that runs in time $\tau(\varepsilon) = \mathrm{O}\left(\frac{t}{\varepsilon-\kappa}\right)$ outputs sk with probability $1 - \varepsilon_2$.

and know a mathematical fact that any $\tau(2\varepsilon_1)$-time algorithm $\mathcal{A}$

$$\Pr\left[(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen} : \mathcal{A}(\mathsf{pk}) = \mathsf{sk}\right] \leq \varepsilon_1(1 - \varepsilon_2)$$

then we can prove an average-case security guarantee:

For any t-time prover $\mathcal{P}_*$ that does not know the secret key

$$\mathsf{Adv}^{\mathsf{ent\text{-}auth}}(\mathcal{A}) = \Pr\left[(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen} : \mathcal{V}^{\mathcal{P}_*(\mathsf{pk})} = 1\right] \leq 2\varepsilon_1 \enspace .$$

Objective security guarantees

Schnorr identification scheme

If $\mathbb{G}$ is a DL group, then the Schnorr identification scheme is secure, where the success probability is averaged over all possible runs of the setup Gen.

Fiat-Shamir identification scheme

Assume that modulus n is chosen form a distribution $\mathcal{N}$ of RSA moduli such that on average factoring is hard over $\mathcal{N}$. Then the Fiat-Shamir identification scheme is secure, where the success probability is averaged over all possible runs of the setup Gen and over all choices of modulus n.