Computer security
Identification

Olivier Markowitch

Identification

The aim is to allows a *verifier* to gain assurances about the identity of a *prover*

Such a process is needed for access controls (prevention) as well as for logging (detection and reaction)

Identification and authentication

Usually:

- *identification* is the process during which an entity claims its identity, and

- *authentication* is the process during which the entity proves the validity of its claimed identity

It may happen that the whole process (identity claim and proof) is called identification or authentication

Properties

Authentication protocols must be designed in order to prevent a verifier to use the authentication information received from a prover to *impersonalize* this prover (non-transferability of the identities)

Authentication protocols must be designed such that the probability that an opponent succeeds in proving the identity of another entity must be negligible

Identification and authentication

Identification and authentication can be based on:

- something that is secretly known by the prover
- something that is owned by the prover
- physical characteristics of the prover
- behaviour of the prover
- etc.

Weak authentication

based on:

- passwords
- one-time password: S/Key
- one-time password: Lamport authentication scheme
- message authentication code

Passwords

Risks:

- users choose often the same password to access to different resources

- remember a password $<>$ preventing to guess a password

Passwords

Attacks:

- online: fake login, social engineering
- offline: exhaustive search, dictionary search

Passwords

Protections:

- minimum length
- password format
- password automatic generation
- expiration date
- limited number of wrong passwords

Passwords

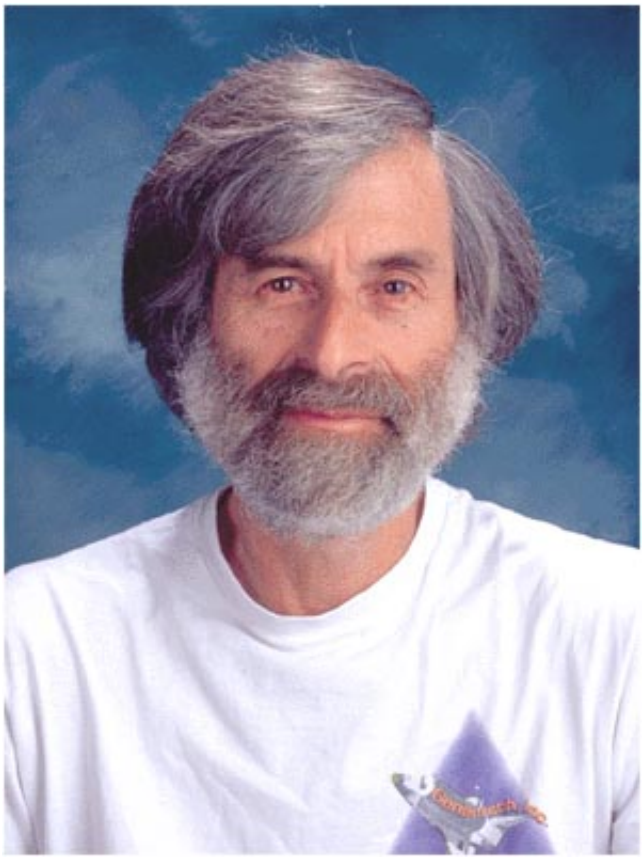
- display of information
- dictionary attack
- shadowing
- trusted path
- salting
- ...

One-time password: S-Key

Login and password may appear in clear while being transmitted

The S/Key one-time password system. Neil Haller. Symposium on Network and Distributed Systems Security, 1994

One-time password: Lamport



Leslie Lamport

One-time password: Lamport

Protocol Lamport's OWF-based one-time passwords

SUMMARY: A identifies itself to B using one-time passwords from a sequence.

1. *One-time setup*.
   (a) User A begins with a secret w. Let H be a one-way function.
   (b) A constant t is fixed (e.g., $t = 100$ or 1000), defining the number of identifications to be allowed. (The system is thereafter restarted with a new w, to avoid replay attacks.)
   (c) A transfers (the *initial shared secret*) $w_0 = H^t(w)$, in a manner guaranteeing its authenticity, to the system B. B initializes its counter for A to $i_A = 1$.
2. *Protocol messages*. The i^{th} identification, $1 \leq i \leq t$, proceeds as follows:

$$A \rightarrow B: \quad A,\ i,\ w_i\ (= H^{t-i}(w)) \quad (1)$$

   Here $A \rightarrow B$: X denotes A sending the message X to B.
3. *Protocol actions*. To identify itself for session i, A does the following.
   (a) A's equipment computes $w_i = H^{t-i}(w)$ (easily done either from w itself, or from an appropriate intermediate value saved during the computation of $H^t(w)$ initially), and transmits (1) to B.
   (b) B checks that $i = i_A$, and that the received password w_i satisfies: $H(w_i) = w_{i-1}$. If both checks succeed, B accepts the password, sets $i_A \leftarrow i_A + 1$, and saves w_i for the next session verification.

One-time password: MAC

$P \rightarrow V : r, h_k(r)$

where:

- k is a secret key shared by the prover and the verifier
- r is a value choosen randomly at each identification process

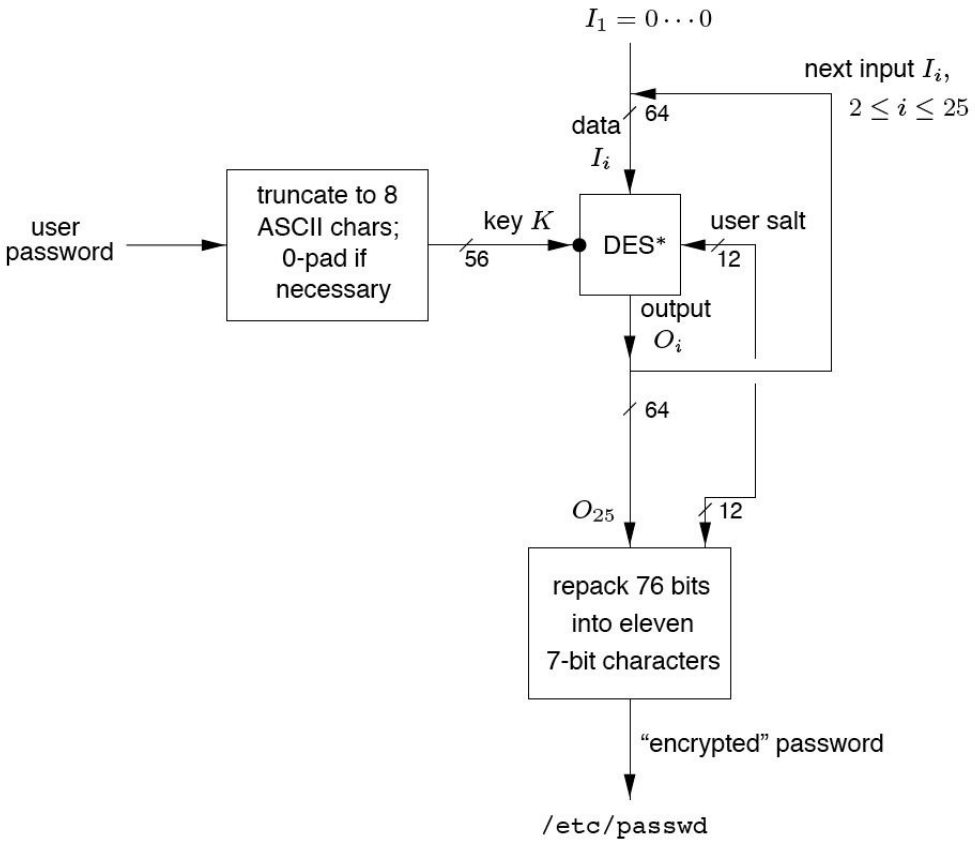


Figure 10.2: UNIX crypt *password mapping. DES* indicates DES with the expansion mapping* E *modified by a 12-bit salt.*

Strong authentication

Also called *challenge-response*

The prover proves the knowledge of its secret without revealing it to the verifier

Requires an interactivity between the prover and the verifier: at each session, the verifier asks a question (the challenge) to the prover and the prover can answer (the response) using it secret.

Strong authentication

Challenge-response can be based on :

- symmetric ciphers

Authentication: symmetric ciphers

- Unilateral authentication

Authentication: symmetric ciphers

- Unilateral authentication

  $V \rightarrow P : r_V$

Authentication: symmetric ciphers

- Unilateral authentication

  $V \rightarrow P$: r_V (challenge)

Authentication: symmetric ciphers

- Unilateral authentication

  $V \rightarrow P$: r_V (challenge)

  $P \rightarrow V$: $E_k(r_V)$

Authentication: symmetric ciphers

- Unilateral authentication

  $V \rightarrow P$: r_V (challenge)

  $P \rightarrow V$: $E_k(r_V)$ (response)

Authentication: symmetric ciphers

- Unilateral authentication

  $V \rightarrow P$: r_V (challenge)

  $P \rightarrow V$: $E_k(r_V)$ (response)

- Mutual authentication

Authentication: symmetric ciphers

- Unilateral authentication

  $V \rightarrow P$: r_V (challenge)

  $P \rightarrow V$: $E_k(r_V)$ (response)

- Mutual authentication

  $V \rightarrow P$: r_V (challenge 1)

Authentication: symmetric ciphers

- Unilateral authentication

  $V \rightarrow P$: r_V (challenge)

  $P \rightarrow V$: $E_k(r_V)$ (response)

- Mutual authentication

  $V \rightarrow P$: r_V (challenge 1)

  $P \rightarrow V$: $E_k(r_V, r_P)$ (response 1, challenge 2)

Authentication: symmetric ciphers

- Unilateral authentication

  $V \rightarrow P : r_V$ (challenge)

  $P \rightarrow V : E_k(r_V)$ (response)

- Mutual authentication

  $V \rightarrow P : r_V$ (challenge 1)

  $P \rightarrow V : E_k(r_V, r_P)$ (response 1, challenge 2)

  $V \rightarrow P : E_k(r_P, r_V)$ (response 2)

Strong authentication

Challenge-response can be based on :

- symmetric ciphers
- keyed hash functions

Authentication based on MAC

- Unilateral authentication

  $V \rightarrow P$: r_V (challenge)

  $P \rightarrow V$: $h_k(r_V)$ (response)

Authentication based on MAC

- Unilateral authentication

  $V \rightarrow P$: r_V (challenge)

  $P \rightarrow V$: $h_k(r_V)$ (response)

- Mutual authentication

  $V \rightarrow P$: r_V (challenge 1)

  $P \rightarrow V$: $r_P, h_k(r_V, r_P)$ (response 1, challenge 2)

  $V \rightarrow P$: $h_k(r_P, r_V)$ (response 2)

Identification sur base de MAC

- Unilateral authentication

  $V \rightarrow P$: r_V (challenge)

  $P \rightarrow V$: $h_k(r_V)$ (response)

- Mutual authentication: **SKID3**

  $V \rightarrow P$: r_V (challenge 1)

  $P \rightarrow V$: $r_P, h_k(r_V, r_P)$ (response 1, challenge 2)

  $V \rightarrow P$: $h_k(r_P, r_V)$ (response 2)

Strong authentication

Challenge-response can be based on :

- symmetric ciphers
- keyed hash functions
- asymmetric ciphers

Authentication: asymmetric ciphers

- Unilateral authentication

  $V \rightarrow P : E_{K_P}(r)$

Authentication: asymmetric ciphers

- Unilateral authentication

  $V \rightarrow P : E_{K_P}(r)$ (challenge)

Authentication: asymmetric ciphers

- Unilateral authentication

  $V \rightarrow P : E_{K_P}(r)$ (challenge)

  $P \rightarrow V : r$

Authentication: asymmetric ciphers

- Unilateral authentication

  $V \rightarrow P$: $E_{K_P}(r)$ (challenge)

  $P \rightarrow V$: r (response)

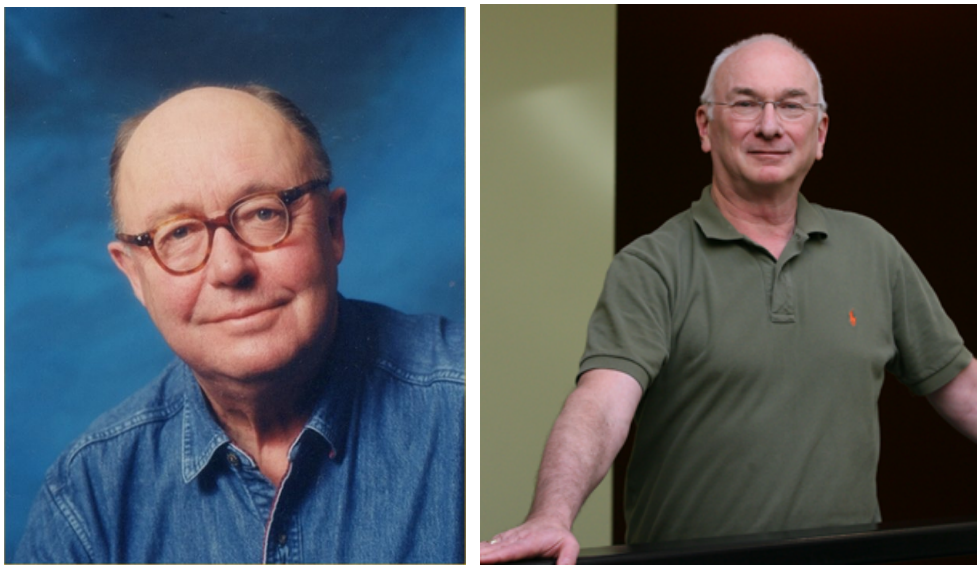
Authentication: asymmetric ciphers

- Unilateral authentication

  $V \rightarrow P : E_{K_P}(r)$ (challenge)

  $P \rightarrow V : r$ (response)

- Mutual authentication: **Needham-Schroeder**



Roger Needham and Michael Schroeder

Authentication: asymmetric ciphers

- Unilateral authentication

  $V \rightarrow P : E_{K_P}(r)$ (challenge)

  $P \rightarrow V : r$ (response)

- Mutual authentication: **Needham-Schroeder**

  $P \rightarrow V : E_{K_V}(r_1, P)$ (challenge 1)

Authentication: asymmetric ciphers

- Unilateral authentication

  $V \rightarrow P : E_{K_P}(r)$ (challenge)

  $P \rightarrow V : r$ (response)

- Mutual authentication: **Needham-Schroeder**

  $P \rightarrow V : E_{K_V}(r_1, P)$ (challenge 1)

  $V \rightarrow P : E_{K_P}(r_1, r_2)$ (response 1, challenge 2)

Authentication: asymmetric ciphers

- Unilateral authentication

  $V \rightarrow P$: $E_{K_P}(r)$ (challenge)

  $P \rightarrow V$: r (response)

- Mutual authentication: **Needham-Schroeder**

  $P \rightarrow V$: $E_{K_V}(r_1, P)$ (challenge 1)

  $V \rightarrow P$: $E_{K_P}(r_1, r_2)$ (response 1, challenge 2)

  $P \rightarrow V$: r_2 (response 2)

Needham-Schroeder

$P \rightarrow V : E_{K_V}(r_1, P)$

Needham-Schroeder

$$P \rightarrow V : E_{K_V}(r_1, P)$$

$$V \rightarrow V' : E_{K_{V'}}(r_1, P)$$

Needham-Schroeder

$$P \rightarrow V : E_{K_V}(r_1, P)$$

$$V \rightarrow V' : E_{K_{V'}}(r_1, P)$$

$$V' \rightarrow V : E_{K_P}(r_1, r_2)$$

Needham-Schroeder

$$P \rightarrow V : E_{K_V}(r_1, P)$$

$$V \rightarrow V' : E_{K_{V'}}(r_1, P)$$

$$V' \rightarrow V : E_{K_P}(r_1, r_2)$$

$$V \rightarrow P : E_{K_P}(r_1, r_2)$$

Needham-Schroeder

$P \rightarrow V : E_{K_V}(r_1, P)$

$V \rightarrow V' : E_{K_{V'}}(r_1, P)$

$V' \rightarrow V : E_{K_P}(r_1, r_2)$

$V \rightarrow P : E_{K_P}(r_1, r_2)$

$P \rightarrow V : r_2$

Needham-Schroeder

$P \rightarrow V : E_{K_V}(r_1, P)$

$V \rightarrow V' : E_{K_{V'}}(r_1, P)$

$V' \rightarrow V : E_{K_P}(r_1, r_2)$

$V \rightarrow P : E_{K_P}(r_1, r_2)$

$P \rightarrow V : r_2$

$V \rightarrow V' : r_2$

Needham-Schroeder fixed

$P \rightarrow V : E_{K_V}(r_1, P)$

$V \rightarrow P : E_{K_P}(\mathbf{V}, r_1, r_2)$

$P \rightarrow V : r_2$

Needham-Schroeder fixed

$P \rightarrow V : E_{K_V}(r_1, P)$

$V \rightarrow V' : E_{K_{V'}}(r_1, P)$

$V' \rightarrow V : E_{K_P}(V', r_1, r_2)$

$V \rightarrow P : E_{K_P}(V', r_1, r_2)$

$P \rightarrow V$: STOP

Zero-knowledge protocols

Interactive proof protocols specifically designed to achieve identification (using asymmetric techniques)

When an interactive proof protocol is complete and sound, the protocol is called a *proof of knowledge*

Zero-knowledge protocols

An interactive proof protocol is complete if, given an honest prover and an honest verifier, the verifier accepts the proof with a probability close to 1

An interactive proof protocol is sound if the probability that a dishonest prover (impersonating A) succeeds in convincing the verifier is negligible, otherwise the algorithm executed by the dishonest prover can be used to extract the secret of the genuine prover

Zero-knowledge protocols

A proof of knowledge protocol can respect the *zero-knowledge property*, the protocol is then said to be *simulatable*

A proof of knowledge protocol respects the zero-knowledge property if there exists a polynomial-time algorithm, called the simulator, which can produce, upon input of the assertion(s) to be proven but without interacting with the real prover, transcripts indistinguishable from those resulting from interaction with the real prover

How to Explain Zero-Knowledge Protocols to Your Children

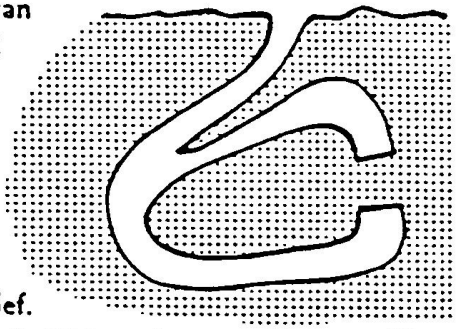
QUISQUATER Jean-Jacques[1], *Myriam, Muriel, Michaël*
GUILLOU Louis[2], *Marie Annick, Gaïd, Anna, Gwenolé, Soazig*
in collaboration with *Tom BERSON*[3] for the English version

[1] Philips Research Laboratory, Avenue Van Becelaere, 2, B–1170 Brussels, Belgium.
[2] CCETT/EPT, BP 59, F–35512 Cesson Sévigné, France.
[3] Anagram Laboratories, P.O. Box 791, Palo Alto CA 94301, USA.

The Strange Cave of Ali Baba

◇ Know, oh my children, that very long ago, in the Eastern city of Baghdad, there lived an old man named Ali Baba. Every day Ali Baba would go to the bazaar to buy or sell things. This is a story which is partly about Ali Baba, and partly also about a cave, a strange cave whose secret and wonder exist to this day. But I get ahead of myself ...

One day in the Baghdad bazaar a thief grabbed a purse from Ali Baba who right away started to run after him. The thief fled into a cave whose entryway forked into two dark winding passages: one to the left and the other to the right (The Entry of the Cave).



Ali Baba did not see which passage the thief ran into. Ali Baba had to choose which way to go, and he decided to go to the left. The left-hand passage ended in a dead end. Ali Baba searched all the way from the fork to the dead end, but he did not find the thief. Ali Baba said to himself that the thief was perhaps in the other passage. So he searched the right-hand passage, which also came to a dead end. But again he did not find the thief. "This cave is pretty strange," said Ali Baba to himself, "Where has my thief gone?"

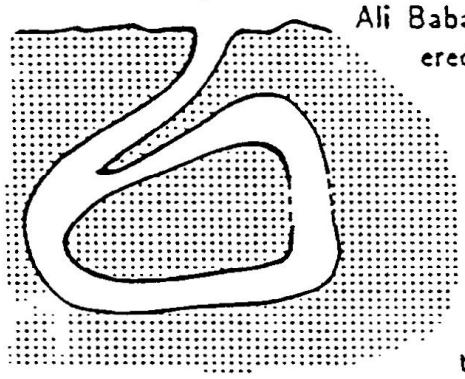
The following day another thief grabbed Ali Baba's basket and fled, as the first thief had fled, into the strange cave. Ali Baba pursued him, and again did not see which way the thief went. This time Ali Baba decided to search to the right. He went all the way to the end of the right-hand passage, but he did not find the thief. He said to himself that, like the first thief, the second thief had also been lucky in taking the passage Ali Baba did not choose to search. This had undoubtedly let the thief leave again and to blend quietly into the crowded bazaar.

The days went by, and every day brought its thief. Ali Baba always ran after the thief, but he never caught any of them. On the fortieth day a fortieth thief grabbed Ali Baba's turban and fled, as thirty-nine thieves had done before him, into the strange cave. Ali Baba yet again did not see which way the thief went. This time Ali Baba decided to search the left-hand passage, but again he did not find the thief at the end of the passage. Ali Baba was very puzzled.

He could have said to himself, as he had done before, that the fortieth thief had been as lucky as each of the other thirty-nine thieves. But this explanation was so

far-fetched that even Ali Baba did not believe it. The luck of the forty thieves was just too good to be a matter of chance. There was only one chance in a million million that all of the forty would escape! So Ali Baba said to himself that there must be another more likely explanation. He began to suspect that the strange cave guarded a secret!

And Ali Baba set out to discover the secret of the strange cave. He decided to hide under some sacks at the end of the right-hand passage. After a very uncomfortable wait he saw a thief arrive who, sensing he was pursued by his victim, whispered the magic words, "Open sesame." Ali Baba was amazed to see the wall of the cave slide open. The thief ran through the opening. Then the wall slid closed again. The pursuer arrived, and was all upset to find only Ali Baba under the sacks at the dead end of the passage. The thief had escaped. But Ali Baba was all happy, for he was finding out the Secret of the Strange Cave.



Ali Baba experimented with the magic words. He discovered to his amazement that when the wall slid open the right-hand passage was connected with the left-hand passage. Now Ali Baba knew how all of the forty thieves had escaped from him.

Ali Baba worked and worked with the magic words, and he finally managed to replace them with new magic words, a little like you change the combination for some padlocks. The very next day a thief was caught. Ali Baba recorded this story and his discovery in a lovely illuminated manuscript. He did not write down the new magic words, but he included some subtle clues about them.

The Fate of the Manuscript

Ali Baba's lovely illuminated manuscript arrived in Italy in the Middle Ages. Today it is in the United States, near Boston. There it has recently held the full attention of several curious researchers. Through decryption of the subtle clues, these researchers have even recovered the new magic words!

After several archaeological excavations in the ruins of the old Baghdad bazaar,the strange cave was located. It was not a myth! And, despite the centuries, the magic words still worked. All agog, the curious researchers went through the end wall between the two passages.

The television networks were quickly made aware of the unusual events taking place in Baghdad. A big American network even got an exclusive on the story. One of the researchers, a certain Mick Ali, a descendent perhaps of Ali Baba, wanted to demonstrate that he knew the secret. But he did not want to reveal the secret. Here is what he did.

First, a television crew filmed a detailed tour of the cave with the two dead-end passages. Then everybody went out of the cave. Mick Ali went back in alone and went down one of the passages. Then the reporter, accompanied by the camera, went inside only as far as the fork. There he flipped a coin to choose between right and left. If the coin come up heads he would tell Mick to come out on the right. If the coin came up tails he would tell Mick to come out on the left. It was heads, so the reporter called

out loud, "Mick, come out on the right." And Mick did just that.

In memory of the forty thieves this demonstration scene was played forty times. Each of the times everybody went back out of the cave and Mick entered alone all the way in to one of the passages. Then the reporter and the camera went as far as the fork where he chose by flipping a coin which order to give to Mick. Mick succeeded in all forty scenes.

Anybody who did not know the secret of the cave would have been exposed on the first failure. Each new test divided by two the chances of success for someone without the secret. On the other hand, the secret allowed Mick to come out each time by the required exit.

The Jealous Reporter

Employed by another television network, a jealous reporter wanted to also film a story on the strange cave. Mick refused to participate because he had given exclusive rights to the story to the first network.

But Mick mischievously suggested to the jealous reporter that the story could be filmed without possessing the secret. The jealous reporter thought and thought, and finally he understood. He said to himself, "I even know a stage actor who looks like Mick Ali and who could be mistaken for him."

And the second story was filmed. In the course of the filming half of the scenes were spoiled because Mick's double did not know how to get from one passage to the other! The jealous reporter edited the tape and only kept the successful scenes until he had forty of them.

The two stories were broadcast at the same hour on the same evening by the two competing American networks. The matter was taken to court. Both videotapes were placed into evidence. But the judges and the experts could not tell the tapes apart. Which tape was simulated? Which tape was genuine? The tapes alone were not enough to judge by.

The simulation surely conveyed no knowledge of the secret. But the simulation and the genuine tape were indistinguishable. So the genuine tape did not convey knowledge of the secret either. The reporter who had gotten the exclusive story had been convinced at the time that Mick Ali knew the secret, but the reporter could not pass his conviction on to the judges in court or to the television audience either.

Mick Ali had achieved his real objective. He wanted, in fact, to show that it is possible to convince without revealing, and so without unveiling his secret.

The Tests in Parallel

Meanwhile, other researchers in Israel observed that by using several secrets and making tests in parallel, one could reduce the number of scenes in the films. In other words, the length of the authentication.

They imagined an apartment building with one cave per floor, each having its own magic words. They needed was one extra actor per cave. All the floors could be filmed at once to see where the actor came out on each floor.

They even proposed an arithmetic solution where a reply with a single number as proof could replaced many actors.

Still, a compromise between the number of secrets and the number of scenes to

film may not always be optimal. It would be much better to have a single secret and a single scene.

Besides that, simulation by successive attempts becomes less and less practical as the number of secrets increases. Do we have no conveyance of knowledge when you cannot simulate with successive attempts?

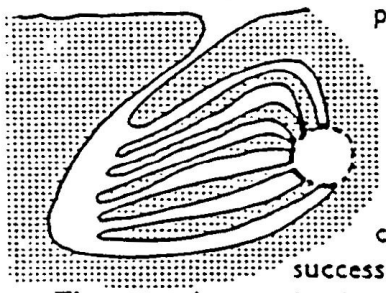
The Prior Agreement

All of this really intrigued some European researchers. They made an observation that applies equally to the serial version and to the parallel version. To save time filming, the jealous reporter and Mick Ali's double would have been pretty clever to think of agreeing in advance on a list of forty random selections between right and left. During the filming, the jealous reporter would have then pretended to choose the questions at random in his head, and the double, who knew in advance the questions he would be asked, would not need to know the secret and could still pass all of the tests one after the other.

Therefore to the simulation technique of successive attempts where only the successful scenes are kept was added a simulation technique of prior agreement between prover and verifier.

A Single Test, A Single Secret

In response to this observation a new cave was set up with more passages ending at a fork (The Revised Cave). Certainly the physical construction of the cave becomes problematic when the number of passages increases. It is impossible to build a cave with a million million passages. But whatever the number of passages, you could simulate by prior agreement. A more arithmetic scheme would allow a verifier to choose a question from a set of a million million questions. With a single test you could directly reach the level of security obtained with forty successive tests in the cave with two passages.



The court is completely unable to tell the videotapes apart: one depicting a demonstration, the other a simulation by prior agreement. Therefore, even when the size of the question is large the demonstration does not show knowledge of the secret's value.

Epilogue

And so, my children, you have heard how Ali Baba learned the secret of the strange cave, and how his descendent, the clever researcher Mick Ali, was able to convince a television reporter that he knew the secret without having to tell him what the secret was. Countless people saw Mick Ali on the television, and he became famous and had adventures around the world. He still has not revealed the secret of the strange cave, but has convinced many others, including me, that he does know it. The keeping of secrets reminds me of the story of the Merkle Hellman and his super-increasing knapsack. But the hour grows late. That is another story for another time. □ □ □

Zero-knowledge protocols

- Fiat-Shamir (based on the factorization)
- Guillou-Quisquater ((based on the factorization)
- Schnorr (based on the discrete logarithm)

Fiat-Shamir

Amos Fiat and Adi Shamir

Fiat-Shamir: premices

An authority:

- chooses two secret primes: p et q
- compute the public value n, where $n = pq$

Each prover:

- chooses its private information s such that $s \in [1, n-1]$ is prime with n
- compute its public value v, where $v = s^2 \mod n$

Fiat-Shamir: authentication

1. the prover chooses a random value $r \in [1, n-1]$, computes the commitment $x = r^2 \mod n$, and sends x to the verifier

2. the verifier chooses a random bit e (the challenge) and sends it to the prover

3. the prover computes the response $y = r \cdot s^e \mod n$ and sends y to the verifier

4. if $y^2 \equiv x \cdot v^e \pmod n$ then the verifier accepts this round of authentication

These steps are realized t times in a row

Fiat-Shamir: complete

The prover sends :

$$y \equiv rs^e \pmod{n}$$

Verification:

$$y^2 \equiv r^2 s^{2e} \pmod{n}$$

$$y^2 \equiv xv^e \pmod{n}$$

Fiat-Shamir: sound

If an opponent succeeds in authenticating itself, repeatedly and with a non negligible probability, this cannot be by guessing e, therefore it is able to *build* good responses y

Suppose that this opponent executes two rounds of the protocol during which it receives two different questions $e_1 = 1$ and $e_2 = 0$, and it provides the corresponding good responses y_1 and y_2 by using the same value r when computing the responses

We have: $y_1 = rs$ and $y_2 = r$, therefore $\frac{y_1}{y_2} = s$ the secret

Fiat-Shamir: simulatable

The simulator chooses randomly a value y and computes:

- $x = y^2 \mod n$ to answer to a question $e = 0$
- $x = y^2 v^{-1} \mod n$ to answer to a question $e = 1$.

Therefore we have a simulation based on a prior knowledge of the challenges

Guillou-Quisquater

Louis Guillou,
France Telecom R&D

Jean-Jacques Quisquater et Louis Guillou

Guillou-Quisquater: premices

An authority:

- chooses two secret primes p et q
- computes the public value n, where $n = pq$
- chooses a public security parameter b (a prime of 40 bits)
- computes the secret a such that $a \cdot b \equiv 1 \pmod{\phi(n)}$
- computes the prover's private information u based on the identity of the prover: $u = (h(ID_{prover}))^{-a} \bmod n$; and sends u to the prouver

Guillou-Quisquater: authentication

1. the prover randomly chooses $k \in [0, n-1]$, computes the commitment $\gamma = k^b \mod n$ and sends γ and ID_{prover} to the verifier

2. the verifier computes $v = h(ID_{prover})$, chooses a random value $r \in [0, b-1]$ (the challenge) and sends r to the prover

3. the prover computes the response $y = k \cdot u^r \mod n$ and sends y to the verifier

4. if $\gamma \equiv v^r \cdot y^b \pmod n$, the verifier accepts the authentication

Guillou-Quisquater: complete

The verifier computes:

$$v^r y^b \equiv (h_{ID_{prover}})^r \; k^b \; u^{rb} \pmod{n}$$

$$v^r y^b \equiv (h_{ID_{prover}})^r \; \gamma \; (h_{ID_{prover}})^{-rab} \pmod{n}$$

$$v^r y^b \equiv (h_{ID_{prover}})^r \; \gamma \; (h_{ID_{prover}})^{-r} \pmod{n}$$

$$v^r y^b \equiv \gamma \pmod{n}$$

Quillou-Quisquater: sound

Suppose an opponent that succeeds in authenticating itself twice with a non negligible probability; suppose that it receives two different questions r_1 and r_2, and it provides the corresponding good responses y_1 and y_2 by using the same value k when computing the responses:

$\gamma \equiv v^{r_1} {y_1}^b \equiv v^{r_2} {y_2}^b \pmod{n}$

$v^{r_1 - r_2} \equiv (\frac{y_2}{y_1})^b \pmod{n}$ (with $r_1 > r_2$)

Let $t = (r_1 - r_2)^{-1} \mod b$ (because $0 < r_1 - r_2 < b$ and b is prime)

$v^{(r_1 - r_2)t} \equiv (\frac{y_2}{y_1})^{bt} \pmod{n}$

Let $(r_1 - r2)t = lb + 1$

$v \equiv (\frac{y_2}{y_1})^{bt} v^{-lb} \pmod{n}$

We have that $a = b^{-1} \mod \phi(n)$, then $v^a \equiv (\frac{y_2}{y_1})^{abt} v^{-abl} \pmod{n}$

Therefore: $u^{-1} \equiv (\frac{y_2}{y_1})^t v^{-l} \pmod{n}$

And: $u \equiv (\frac{y_1}{y_2})^t v^l \pmod{n}$

Guillou-Quisquater: simulatable

The simulator knows r, v and b. It randomly chooses y and computes $\gamma \equiv v^r y^b \pmod{n}$

Therefore we have a simulation based on a prior knowledge of the challenges r

Schnorr

Claus-Peter Schnorr

Schnorr: premices

An authority chooses:

- a large prime p (at least 512 bits)
- a large public prime factor q of $p - 1$ (at least 140 bits)
- a public value $\alpha \in \mathbb{Z}^*_p$ of order q
- a public security parameter t such that $q > 2^t$

Each prover randomly chooses its private information $a \in [0, q - 1]$ and compute the corresponding public value $v = \alpha^{-a} \mod p$

Schnorr: authentication

1. the prover randomly chooses $k \in [0, q-1]$, computes the commitment $\gamma = \alpha^k \mod p$ and sends γ to the verifier

2. the verifier randomly chooses the challenge $r \in \left[1, 2^t\right]$ and sends it the the prover

3. the prover computes the response $y = k + a \cdot r \mod q$ and sends y to the verifier

4. if $\gamma \equiv \alpha^y \cdot v^r \pmod p$, the verifier accepts the authentication

Schnorr: complete

The verifier verifies:

$$\alpha^y v^r \equiv \alpha^k \alpha^{ar} \alpha^{-ar} \pmod{p}$$

$$\alpha^y v^r \equiv \alpha^k \pmod{p}$$

$$\alpha^y v^r \equiv \gamma \pmod{p}$$

Schnorr: sound

Suppose an opponent that succeeds in authenticating itself twice with a non negligible probability; suppose that it receives two different questions r_1 and r_2, and it provides the corresponding good responses y_1 and y_2 by using the same value k when computing the responses:

We have:

$$\gamma \equiv \alpha^{y_1} v^{r_1} \equiv \alpha^{y_2} v^{r_2} \pmod{p}$$

$$\alpha^{y_1 - y_2} \equiv v^{r_2 - r_1} \pmod{p}$$

$$y_1 - y_2 \equiv a(r_1 - r_2) \pmod{q}$$

Since $|r_1 - r_2| < 2^t$ and q is a prime $> 2^t$, we have $gcd(r_1 - r_2, q) = 1$ and

$$a \equiv (y_1 - y_2)(r_1 - r_2)^{-1} \pmod{q}$$

Schnorr: simulatable

The simulator knows r, α and v. It randomly chooses y and computes $\gamma \equiv \alpha^y v^r \pmod{p}$

Therefore we have a simulation based on a prior knowledge of the challenges r