Secure Communications over Insecure Channels based on Short Authenticated Strings

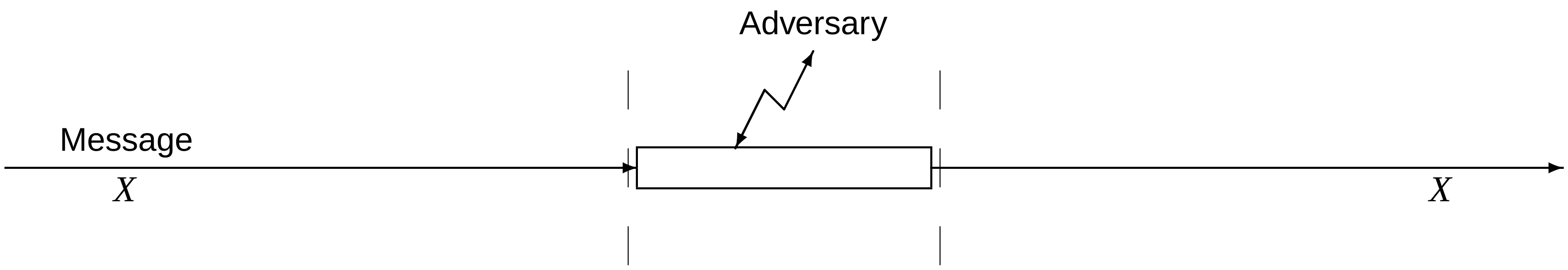
Serge Vaudenay
http://lasecwww.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Crypto'05

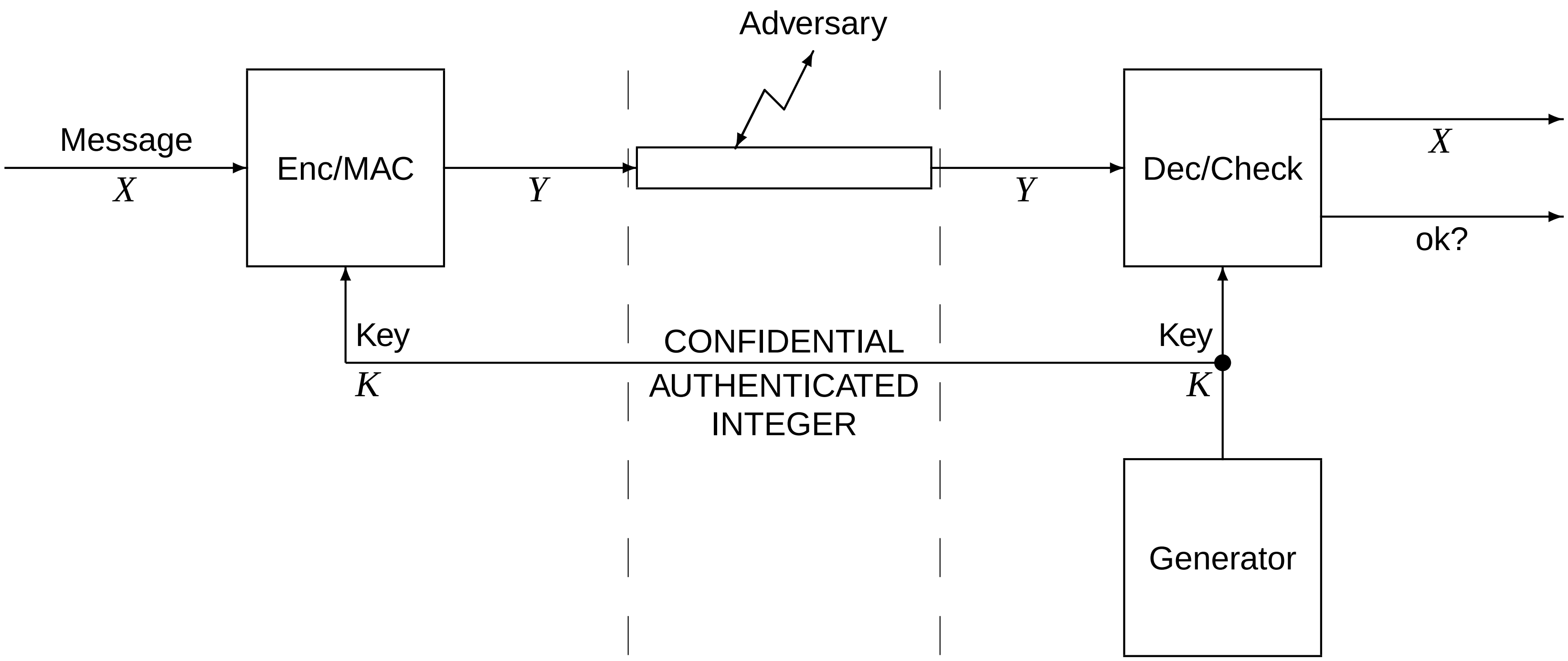
Secure Communications

Basic Security Properties

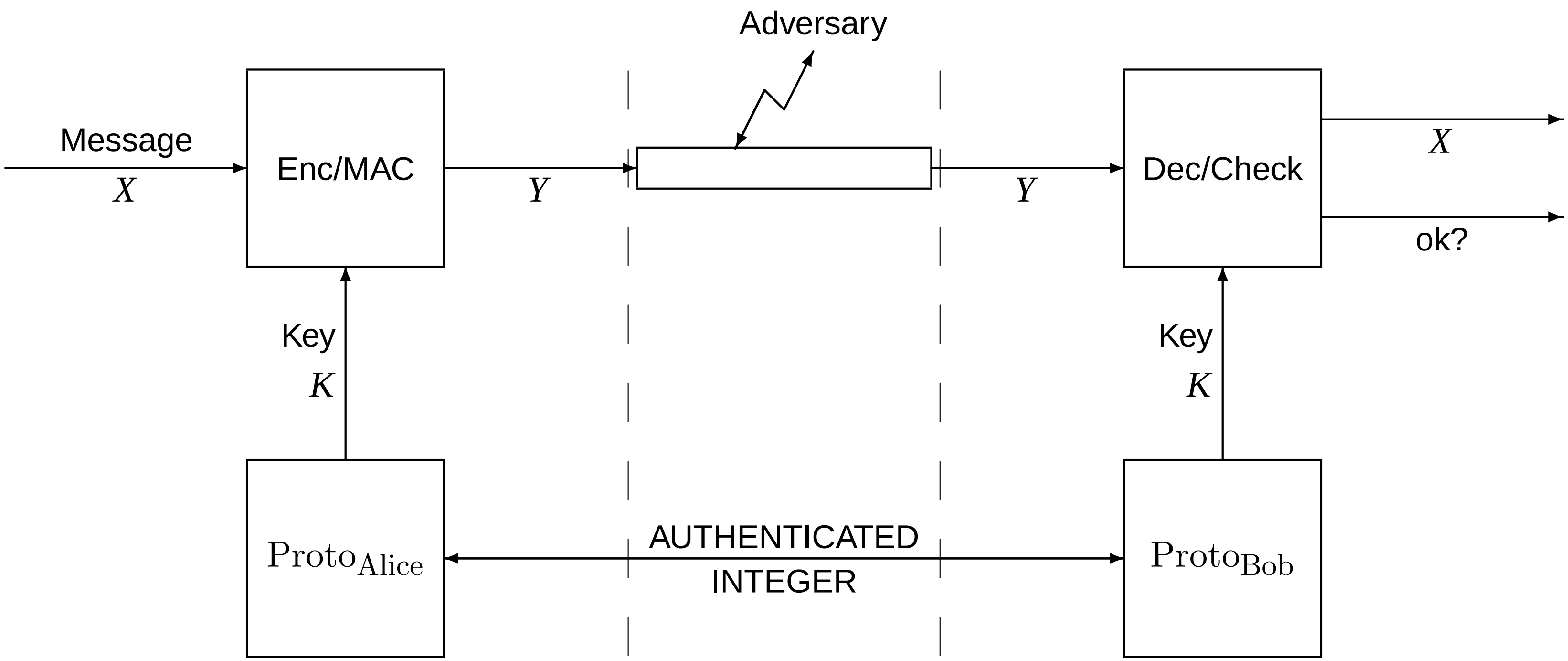


- ⋆ **Confidentiality** (C): only the legitimate receiver can get X
- ⋆ **Authentication** + **Integrity** (A+I): only the legitimate sender can insert X and the received message must be equal to X

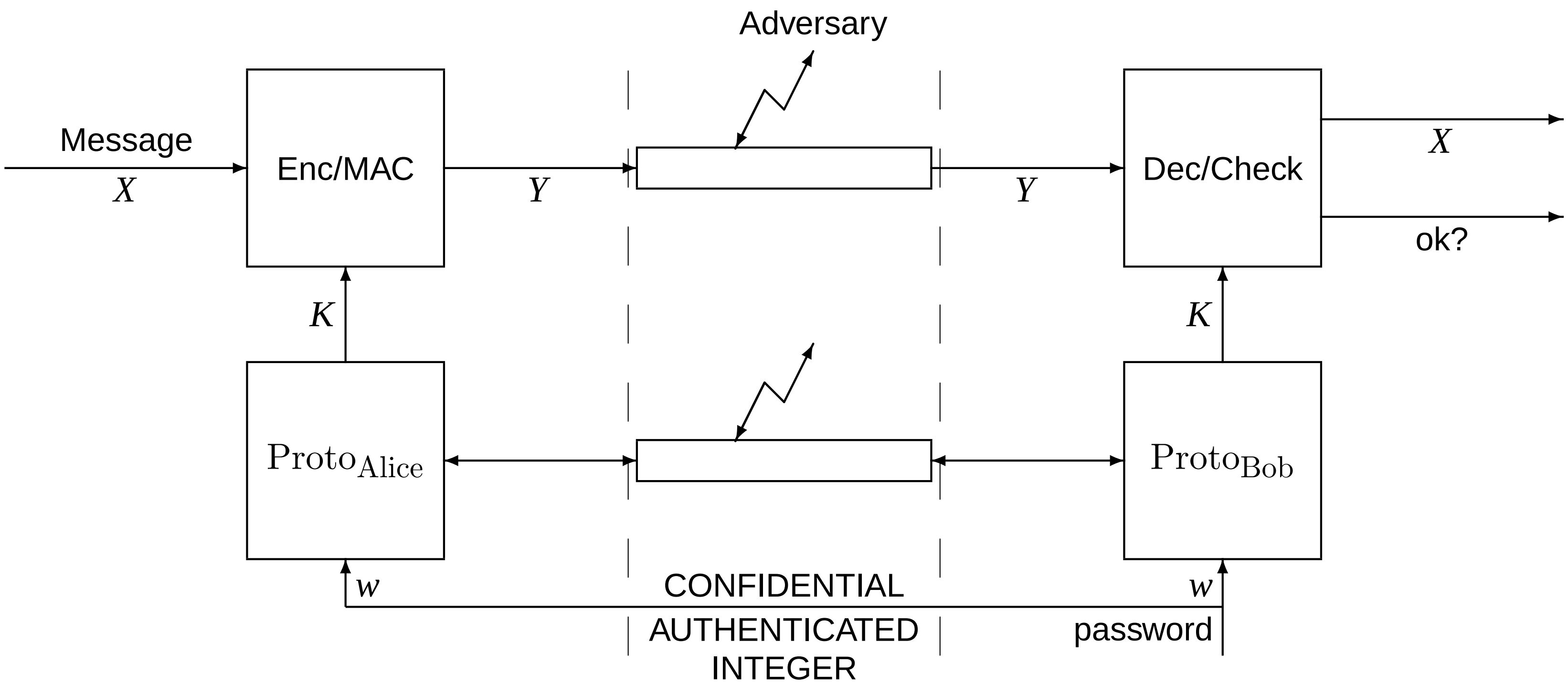
...based on C+A+I Channels: the Conventional Model



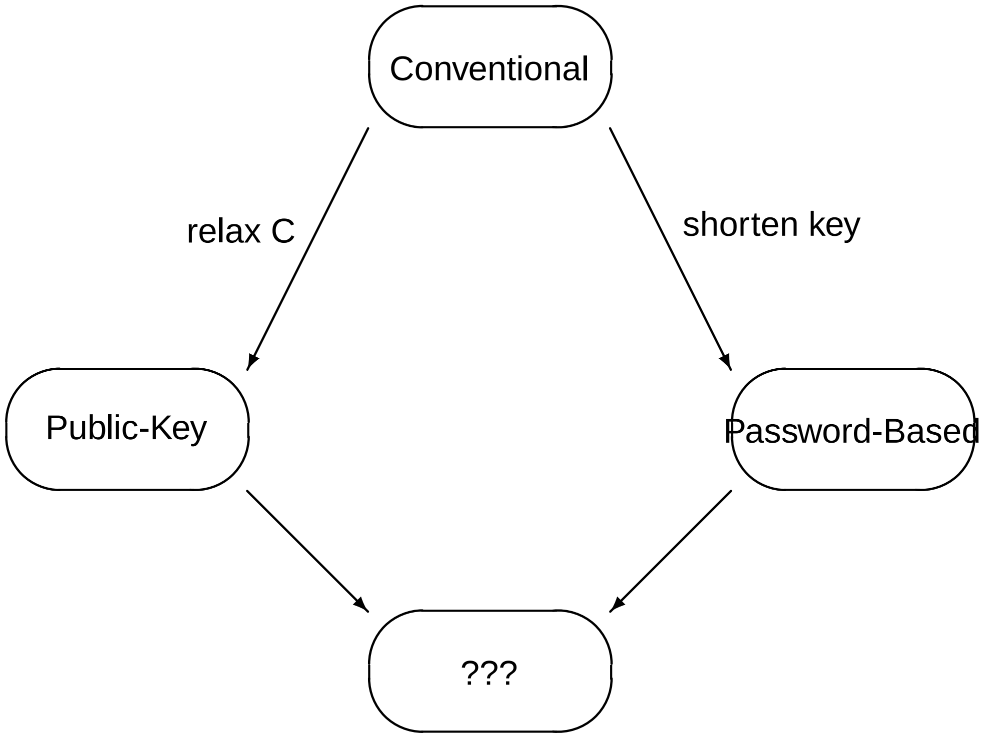
...based on A+I Channels: the Merkle Model 1975



...based on C+A+I Narrowband Channels: the Bellovin-Merritt Model 1992



The Missing Stone

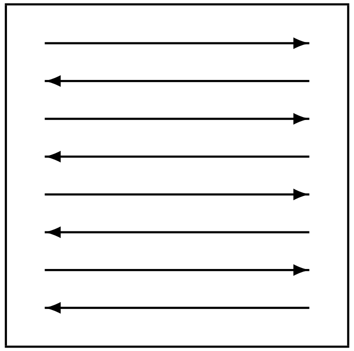


Cryptography Based on Short Authenticated Strings (SAS)

Message Authentication Protocols

Alice (ID) Bob

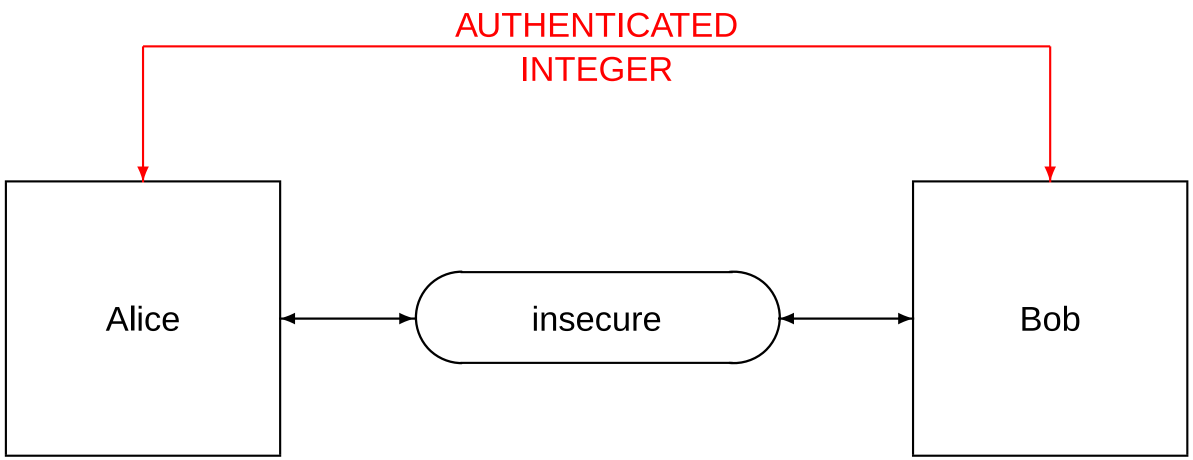
input: m



output: ID, m

- ⋆ can be used to transmit a public key
- ⋆ can be used (in both ways) to run the Diffie-Hellman protocol

Communication Model



⋆ secure channel (A+I) with low bandwidth

Communication Model: Adversary Capabilities

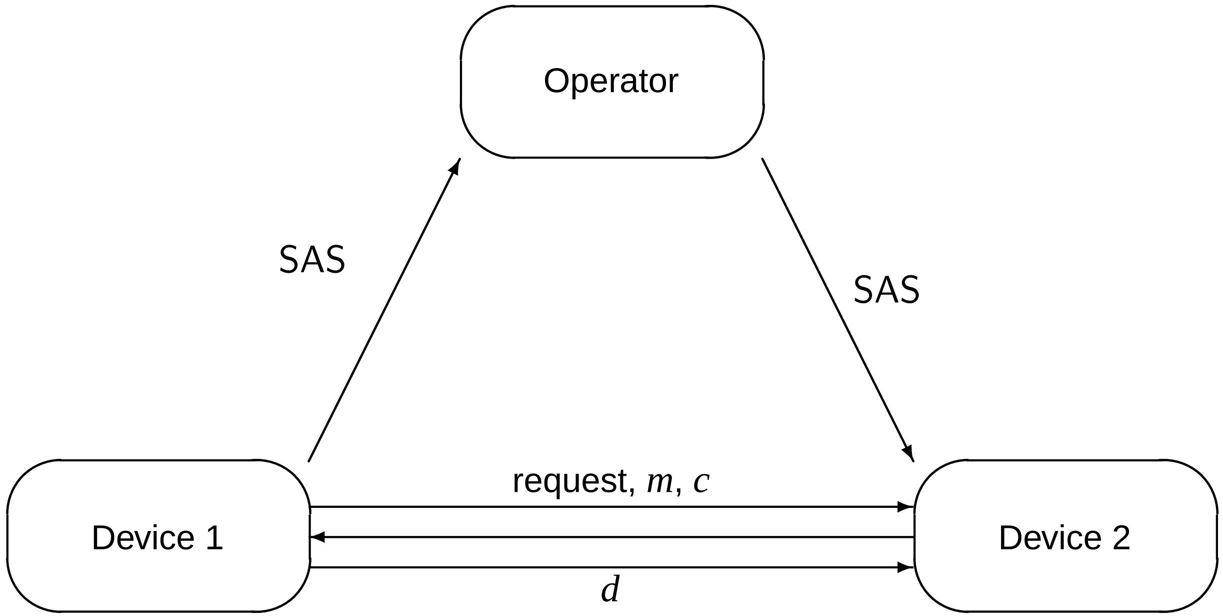
Regular channels: the adverary can do whatever he/she wants with the messages: modify, create, swap, remove, stall, ...

(Weak) authenticated channels: the adversary cannot modify nor create messages. He/she can swap, remove, stall, ...

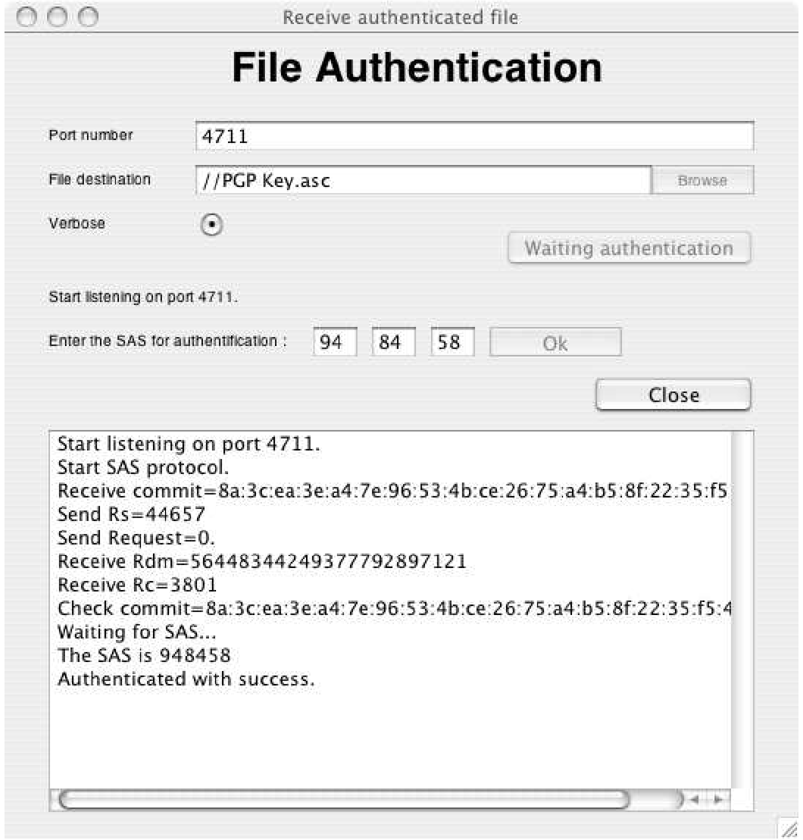
(Strong) authenticated channels: same plus some additional assumptions!

E.g. messages must be either deliver at once or removed (stall-free channels).

Application I: Personal Area Network Setup (Bluetooth, UWB, ...)



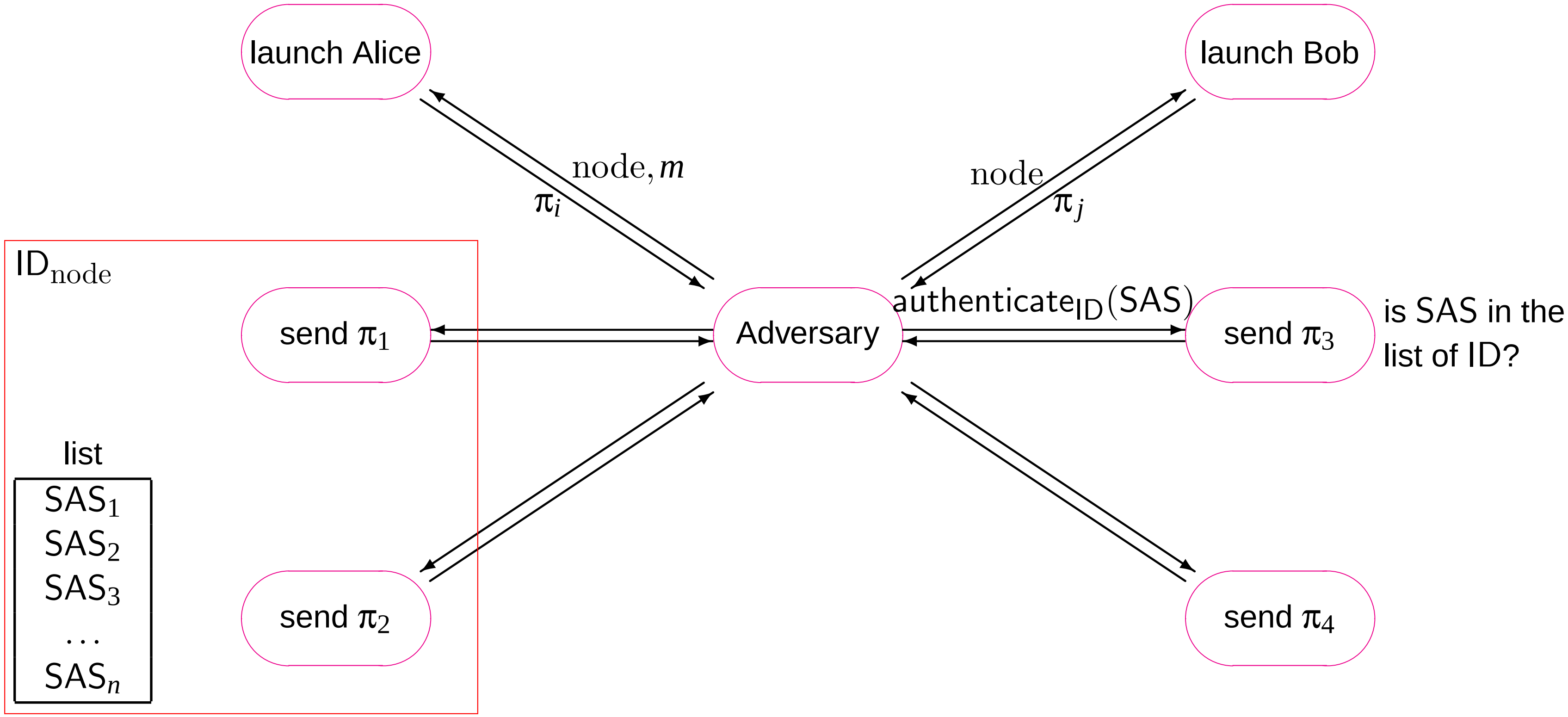
Application II: Peer-to-Peer PGP Channel Setup



Application III: Disaster Recovery

- on the road, after a key loss (computer crash, stolen laptop)
  $\longrightarrow$ set up of a security association
- PKI collapse (company bankrupt, main key sold, act of God)
  $\longrightarrow$ set up of a security association

Adversarial Model



Goal: to make an instance of Bob output $\mathsf{ID}, \hat{m}$ without any instance on Alice on node ID with input $\hat{m}$.

Folklore Protocol (Balfanz-Smetters-Stewart-Chi Wong 2002)

Alice		Bob
input: m		
	$\xrightarrow{m}$	
$h \leftarrow H(m)$	$\xrightarrow{\mathsf{authenticate}_{\mathrm{Alice}}(h)}$	check $h = H(\hat{m})$
		output: $\mathrm{Alice}, \hat{m}$

Security

Theorem 1. *If* H *is a collision resistant hash function onto* $\{0,1\}^k$*, the protocol resists to impersonation attempts.*

- ☺ provable security, efficient (assuming collision resistance)
- ☹ this requires SAS of at least 160 bits

Gehrmann-Mitchel-Nyberg 2004: The MANA I Protocol

Alice		Bob
input: m		
	$\xrightarrow{m}$	
pick $K \in_U \{0,1\}^k$		
$\mu \leftarrow H_K(m)$	$\xrightarrow{\text{authenticate}_{\text{Alice}}(K\|\|\mu)}$	check $\mu = H_K(\hat{m})$
		output: $\text{Alice}, \hat{m}$

Insecurity of MANA I

Alice		Bob
input: m		
pick $K \in_U \{0,1\}^k$	$\xrightarrow{m} \cdots$	
$\mu \leftarrow H_K(m)$	$\xrightarrow{\mathsf{authenticate}_{\mathrm{Alice}}(K\|\|\mu)} \cdots$	
	[find $\hat{m}$ s.t. $H_K(m) = H_K(\hat{m})$]	
	$\cdots \xrightarrow{\hat{m}}$	
	$\cdots \xrightarrow{\mathsf{authenticate}_{\mathrm{Alice}}(K\|\|\mu)}$	check $\mu = H_K(\hat{m})$
		output: $\mathrm{Alice}, \hat{m}$

Security of MANA I

Theorem 2. *Using a universal hash function family* H *which produces* ℓ*-bit codes and in a* **strong communication model***, the maximal probability of success of an impersonation of Alice when limited to* Q_A *runs of Alice's protocol and* Q_B *runs of Bob's protocol is at most* $Q_A Q_B 2^{-k-\ell}$.

- ☺ we can work with SAS of $k+\ell = 20$ bits
- ☹ strong requirement on the communication model

A SAS-Based Authentication Protocol

SAS-Based Authentication

Alice		Bob
input: m		
pick $R_A \in_U \{0,1\}^k$		pick $R_B \in_U \{0,1\}^k$
$(c,d) \leftarrow \mathsf{commit}(m, R_A)$	$\xrightarrow{m\|\|c}$	
	$\xleftarrow{R_B}$	
	$\xrightarrow{d}$	$\hat{R}_A \leftarrow \mathsf{open}(\hat{m}, \hat{c}, \hat{d})$
$\mathsf{SAS} \leftarrow R_A \oplus \hat{R}_B$	$\xrightarrow{\mathsf{authenticate}_{\mathrm{Alice}}(\mathsf{SAS})}$	check $\mathsf{SAS} = \hat{R}_A \oplus R_B$
		output: $\mathrm{Alice}, \hat{m}$

Security

Theorem 3. *Under reasonable assumptions on the commitment scheme (either extractable or equivocable), the maximal probability of success of an impersonation of Alice when limited to* Q_A *runs of Alice's protocol and* Q_B *runs of Bob's protocol is at most* $Q_A Q_B 2^{-k} + \varepsilon$.

- ☺ provable security, efficient
- ☺ we can work with SAS of 20 bits

Tag-Based Commitment Schemes

Set up: $(K_P, K_S) \leftarrow \mathsf{setup}()$

Commit: $(c,d) \leftarrow \mathsf{commit}(m,r)$ commit to r of k bits with tag m

Decommit: $r \leftarrow \mathsf{open}(m,c,d)$ whenever r is such that (c,d) is a possible output of $\mathsf{commit}(m,r)$

Hiding Game

Adversary		Challenger
	$\xleftarrow{K_P}$	setup$()$
select m	$\xrightarrow{m}$	pick r
	$\xleftarrow{c}$	commit(m,r)
compute r'		

win if $r' = r$

$$\Pr[\mathrm{win}] \leq 2^{-k} + \varepsilon$$

Binding Game

Adversary		Challenger
	$\xleftarrow{K_P}$	setup$()$
select m, c	$\xrightarrow{m\|\|c}$	
	$\xleftarrow{r}$	pick r
compute d		

win if $r \leftarrow \mathsf{open}(m,c,d)$

$$\Pr[\mathrm{win}] \leq 2^{-k} + \varepsilon$$

Extractable Commitment Based on a Random Oracle

Extract: $r \leftarrow \text{extract}_{K_S}(m, c)$ whenever there exists d such that $r \leftarrow \text{open}(m, c, d)$

NB: adversaries can call this oracle (except for some challenge tags)

Commit: to commit on r with tag m:

1. pick a random e, set $d = r||e$
2. send $m||d$ to a random oracle H
3. get c

Decommit: check that $H(m||d) = c$, parse $d = r||e$ and output r

Extract: look at the history of oracle calls and from c get d (provided no collision occured)

$\longrightarrow$ Instanciation: take $H = \text{SHA1}$ and hope it makes sense...

Equivocable Commitment in CRS Model
Based on a Signature Scheme (MacKenzie-Yang 2004)

Simulate commit: $(c,\xi) \leftarrow \mathsf{simcommit}_{K_S}(m)$

Equivocate: $d \leftarrow \mathsf{equivocate}_{K_S}(m,c,r,\xi)$ such that $r \leftarrow \mathsf{open}(m,c,d)$

NB: adversaries can call these oracles (except for some challenge tags) but do not see ξ

Example:

- ⋆ Commitment based on DSA (assuming DSA is secure)
  Pedersen commitment of r over a random base $(g',(g')^s)$ such that $(g' \bmod q, s) = \mathsf{sign}(m)$
  - signing m is equivalent to equivocating the Pedersen commitment
  - given m, it is easy to generate a random $(g',(g')^s)$ pair without K_S
- ⋆ Commitment based on Cramer-Shoup (standard model)

Proof Step 1: Reducing to a One-Shot Attacker

- ⋆ NB: the protocol uses a single SAS
- ⋆ a single failing Bob requires a single SAS from a single Alice

  $\longrightarrow$ there must be one crucial instance of Alice and one crucial instance of Bob
- ⋆ given an attack of probability of success p, we pick a random instance of Alice and a random instance of Bob and we simulate all others

  $\longrightarrow$ we obtain a one-shot attack with probability of success p/Q_AQ_B

Proof Step 2: Several Cases to Consider

An attacker must interleave the two following lists of actions (6 combinations)

	get K_P
B1	$\pi_b \leftarrow \text{launch}(\cdot, \mathrm{Bob}, \emptyset)$

A1	select m	B2	select $\hat{m}\|\|\hat{c}$
	$\pi_a \leftarrow \text{launch}(\cdot, \mathrm{Alice}, m)$		$R_B \leftarrow \text{send}(\pi_b, \hat{m}\|\|\hat{c})$
	$c \leftarrow \text{send}(\pi_a, \emptyset)$		
A2	select $\hat{R}_B$	B3	select $\hat{d}$
	$d \leftarrow \text{send}(\pi_a, \hat{R}_B)$		$\text{send}(\pi_b, \hat{d})$

A3	$\text{authenticate}_{\mathrm{Alice}}(\text{SAS}) \leftarrow \text{send}(\pi_a, \emptyset)$
B4	$\text{send}(\pi_b, \text{authenticate}_{\mathrm{Alice}}(\text{SAS}))$

We must consider either extractable or equivocable commitments (2 combinations)

Example: the A1-B2-A2-B3 Equivocable Case

One-Shot Attacker		Simulator		Binding-game Challenger
	$\xleftarrow{K_P}$		$\xleftarrow{K_P}$	$\mathsf{setup}()$
(A1)	$\xrightarrow{m}$			
	$\xleftarrow{c}$	$c \leftarrow \mathsf{simcommit}(m)$		
(B2)	$\xrightarrow{\hat{m}\Vert\hat{c}}$		$\xrightarrow{\hat{m}\Vert\hat{c}}$	pick $\hat{R}_A$
	$\xleftarrow{R_B}$	pick R_B	$\xleftarrow{\hat{R}_A}$	
(A2)	$\xrightarrow{\hat{R}_B}$	$R_A \leftarrow \hat{R}_A \oplus R_B \oplus \hat{R}_B$		
	$\xleftarrow{d}$	$d \leftarrow \mathsf{equivocate}(m, c, R_A)$		
(B3)	$\xrightarrow{\hat{d}}$			

Example: the A1-B2-A2-B3 Extractable Case

One-Shot Attacker		Simulator		Hiding-game Challenger
	$\xleftarrow{K_P}$		$\xleftarrow{K_P}$	$\mathsf{setup}()$
(A1)	$\xrightarrow{m}$		$\xrightarrow{m}$	pick r
	$\xleftarrow{c}$		$\xleftarrow{c}$	$(c,d) \leftarrow \mathsf{commit}(m,r)$
(B2)	$\xrightarrow{\hat{m} \Vert \hat{c}}$	$\hat{R}_A \leftarrow \mathsf{extract}(\hat{m},\hat{c})$		
	$\xleftarrow{R_B}$	pick R_B		
(A2)	$\xrightarrow{\hat{R}_B}$	$R_A \leftarrow \hat{R}_A \oplus R_B \oplus \hat{R}_B$	$\xrightarrow{R_A}$	
	$\xleftarrow{d}$		$\xleftarrow{d}$	
(B3)	$\xrightarrow{\hat{d}}$			

Other Cases

similar (see Proceedings)

Conclusion

- ⋆ secure communications over insecure channels *can* be manually set up by a human operator
- ⋆ applications: personal area network, peer-to-peer, disaster rescue