Security and Trust II: Information Assurance Sec. 5: Pervasive security

Peter-Michael Seidel

Outline

Outline

Introduction

- Idea of pervasive computation
- New security landscape
- Tools of authentication
- Process model
- Network model

Authentication with timed channels

Authentication with social channels

Conclusions

Mouse

Mouse

Symbols with which the human represents the concepts can be arranged before his eyes; moved, stored, recalled, operated upon according to extremely complex rules. . .

Mouse

In the limit of what we might now imagine, this could be a computer which could construct sophisticated images in automatic response to human direction. . .

Mouse

. . . and could involve concepts that we have never yet imagined.

Douglas C. Engelbart
Augmenting Human Intellect (1962)

Mouse

Computation as evolution of concepts depends on the human-computer interaction:

- screen
- windows
- icons (objects)
- printouts

The mouse manages real estate of computation.

Computational spaces

Computer in a black box

- 80 character line interface
- input strings and output strings

Computational spaces

Computer in a black box

- 80 character line interface
- input strings and output strings

Computer in a space of interaction

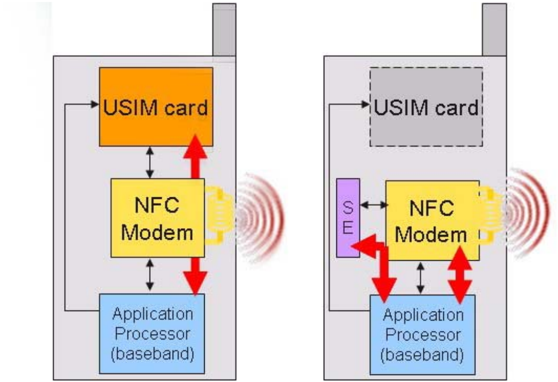
- concepts are symbols, icons, objects
- computation pervades physical space

Pervasive computation

- ubiquitous devices
- programmable environment — disappearing computer
- **computation is coevolution of computational agents**

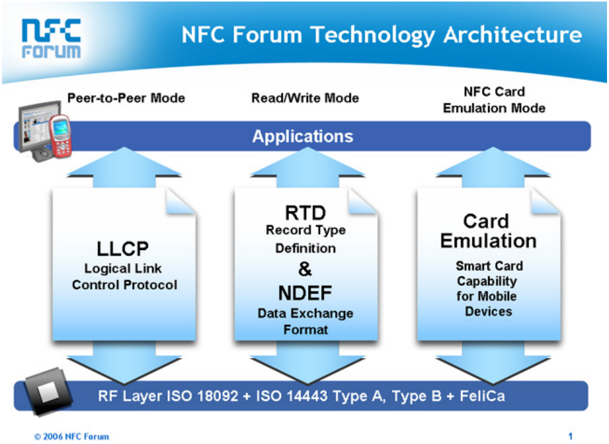
Example: Near Field Communication (NFC)

Phone with a contactless smart card:



Secure Element (SE) is a miniSD flash memory, or a USIM card, or a separate microcontroller.

NFC modes of operation: standards



NFC applications: Payment and exchange

- card mode (← Chip & Pin, EMV)

  2008 transaction value: $ 2.4 billion (Juniper)

  2011 transaction value: $ 24-36 billion (Juniper, Strategy Analytics)
- RW mode:
  - electronic tickets, transportation systems
  - off-line micropayments (← Chip-Knip)
- P2P mode:
  - digital cash transactions
  - electronic barter
  - street markets and transient merchants
  - vending

NFC applications

Proximity commercial networking

- RFID-based shopping
    - discount coupons, mobile rewards distribution
    - warehouse navigation
    - dynamic pricing
        - shop auction
        - shopping derivatives: futures, calls, boolean betting. . .
        - discount for social hubs, celebrities
        - discount for viral marketing, C2C assistance, shop help
    - general shopping assistance

NFC applications

Proximity commercial networking

- RFID-based shopping
  - discount coupons, mobile rewards distribution
  - warehouse navigation
  - dynamic pricing
    - shop auction
    - shopping derivatives: futures, calls, boolean betting. . .
    - discount for social hubs, celebrities
    - discount for viral marketing, C2C assistance, shop help
  - general shopping assistance
- RW mode: bootstrap other networks
  - distribute URLs
  - **drag and drop local links**

NFC applications

Proximity social networking: Beyond the address book

[1] e.g., a fragment of a personal page, reputation certificate, "electronic pheromone"

NFC applications

Proximity social networking: Beyond the address book

- P2P mode: support local networks
  - exchange public keys, personal (business) cards

[1] e.g., a fragment of a personal page, reputation certificate, "electronic pheromone"

NFC applications

Proximity social networking: Beyond the address book

- P2P mode: support local networks
  - exchange public keys, personal (business) cards
- RW mode: generate local networks
  - check in selected personal data[1] at a smart place
    - club, school, shopping mall. . .
  - local recommender system forms clusters
    - sport partners, homework help, one-night stands. . .
    - queryless social search
    - social navigation assistance: friends, foes, fashion. . .

[1] e.g., a fragment of a personal page, reputation certificate, "electronic pheromone"

NFC applications

Proximity social networking: Beyond the address book

- P2P mode: support local networks
  - exchange public keys, personal (business) cards
- RW mode: generate local networks
  - check in selected personal data[1] at a smart place
    - club, school, shopping mall...
  - local recommender system forms clusters
    - sport partners, homework help, one-night stands...
    - queryless social search
    - social navigation assistance: friends, foes, fashion...
  - receive other *relevant* information
    - *recommendation driven* advertising in physical space

[1] e.g., a fragment of a personal page, reputation certificate, "electronic pheromone"

NFC applications

Proximity social networking: Beyond the address book

- P2P mode: support local networks
  - exchange public keys, personal (business) cards
- RW mode: generate local networks
  - check in selected personal data[1] at a smart place
    - club, school, shopping mall...
  - local recommender system forms clusters
    - sport partners, homework help, one-night stands...
    - queryless social search
    - social navigation assistance: friends, foes, fashion...
  - receive other *relevant* information
    - *recommendation driven* advertising in physical space
  - **point-and-click**
    - drag one proximity link to another: introduce friends
    - bootstrap Bluetooth, WLAN networks: "silent concert"

[1] e.g., a fragment of a personal page, reputation certificate, "electronic pheromone"

New security capabilities

Theorem (Even-Yacobi, 1980)

Every deterministic fair exchange protocol must involve a trusted third party: it is always an escrow protocol.

New security capabilities

Theorem (Even-Yacobi, 1980)

Every deterministic fair exchange protocol must involve a trusted third party: it is always an escrow protocol.

Why?

New security capabilities

Theorem (Even-Yacobi, 1980)

Every deterministic fair exchange protocol must involve a trusted third party: it is always an escrow protocol.

Why?

Exchange is like a race where the winning horse is the **last** to finish.

New security capabilities

Pervasive solution

New security capabilities

Pervasive solution

Swap the horses!

New security capabilities

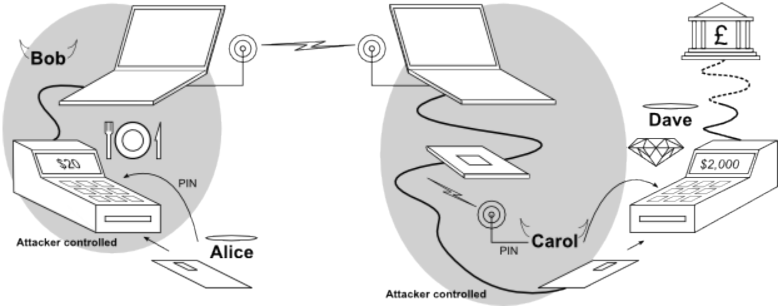
Pervasive solution

Swap the horses!

Figure 1 Design session in a mediated space

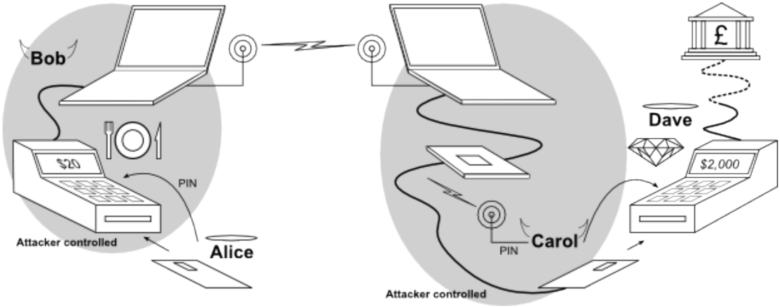
. . . i.e. swap the devices, or the send buttons.

New security problems



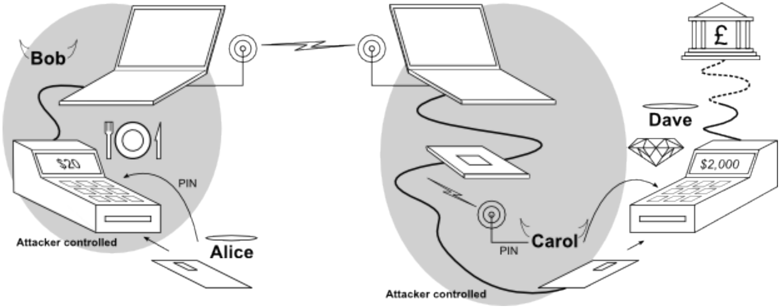
New security problems

The attack requires a long range link.



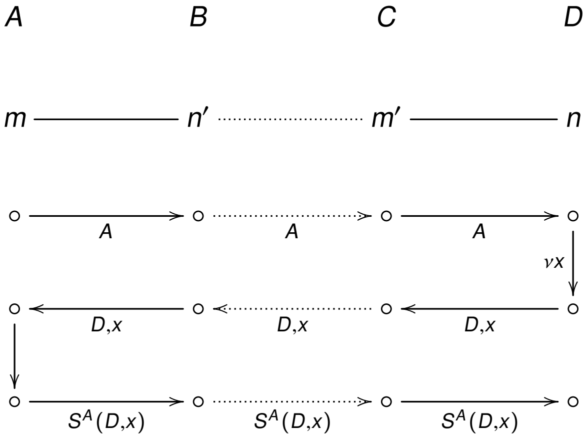
New security problems

The attack requires a long range link.



The NFC phones provide just that!

Agreement is not enough



Summary

Pervasive computation is

- not in cyberspace
  - not distance-free

Summary

Pervasive computation is

- not in cyberspace
  - not distance-free
- but in physical space
  - **principal's position needs to be authenticated.**

Proximity authentication

Degrees of authentication:

- ping authentication: matching records of the messages
- agreement: matching records of intent
- proximity authentication: matching views of the positions

Tools of authentication

You authenticate yourself by leveraging over:

- what you know: secrets, digital keys
- what you have: tokens, smart cards, physical keys
- what you are: biometric properties, handwriting

Tools of authentication

You authenticate yourself by leveraging over:

- what you know: secrets, digital keys
  - can be copied and given away
- what you have: tokens, smart cards, physical keys
  - can be given away, but not copied
- what you are: biometric properties, handwriting
  - cannot be given away, or copied

Tools of authentication

You authenticate yourself by leveraging over:

- what you know: secrets, digital keys
  - can be copied and given away
- what you have: tokens, smart cards, physical keys
  - can be given away, but not copied
- what you are: biometric properties, handwriting
  - cannot be given away, or copied

Idea of proximity authentication

- Most security tokens do not authenticate position directly
- Their physical properties must be used to authenticate position.

Process model

terms $(\mathcal{T}, \sqsubseteq)$,

principals $(\mathcal{W}, \leqslant)$,

actions $\mathcal{A}$ generated by:

action	constructor	form
send	$\mathcal{W}^2 \times \mathcal{T} \overset{\langle\rangle}{\hookrightarrow} \mathcal{A}$	$\langle A \xrightarrow{B}: t\rangle$
receive	$\mathrm{Var}^2_{\mathcal{W}} \times \mathrm{Var}_{\mathcal{T}} \overset{()}{\hookrightarrow} \mathcal{A}$	$(Y \xrightarrow{Z}: x)$
match	$\mathcal{T} \times \mathrm{Op}_{\mathcal{T}} \times \mathrm{Var}_{\mathcal{W}} \overset{(/)}{\hookrightarrow} \mathcal{A}$	$(t/p(x))$
new	$\mathrm{Var}_{\mathcal{T}} \overset{(\nu)}{\hookrightarrow} \mathcal{A}$	(νx)
...	...	...

Process model

processes $\mathbb{P} \xrightarrow{P} \mathcal{A} \times \mathcal{W}$ where

- $(\mathbb{P}, \triangleright)$ is a well-founded partial order
- $P_{\mathcal{W}}(p) \# P_{\mathcal{W}}(q) \quad \Rightarrow \quad p \# q$

runs $\left(P, \ \surd : \ \text{recvs}(P) \longrightarrow \text{sends}(P)\right), \ (x) \ \not\triangleright \surd(x)$

- $\mathbb{P}^{\surd} = \mathbb{P} / \left(\surd(x) \triangleright (x)\right)$

Network model

A *communication network* consists of

network graph $\mathcal{N} = (L \underset{\varrho}{\overset{\delta}{\rightrightarrows}} N)$, where

- N is the set of nodes,
- $L = \sum_{N \times N} \mathcal{N}_{mn}$ is the set of links,
- $\mathcal{N}_{mn} = \langle \delta, \varrho \rangle^{-1}(m, n)$

Network model

A *communication network* consists of

network graph $\mathcal{N} = (L \underset{\varrho}{\overset{\delta}{\rightrightarrows}} N)$, where

- N is the set of nodes,
- $L = \sum_{N \times N} \mathcal{N}_{mn}$ is the set of links,
- $\mathcal{N}_{mn} = \langle \delta, \varrho \rangle^{-1}(m, n)$

control assignment $© : \mathcal{W} \longrightarrow \wp N$, satisfying

$$A \leqslant B \implies ©A \subseteq ©B$$
$$A \# B \implies ©A \cap ©B = \emptyset$$

Network model

A *communication network* consists of

network graph $\mathcal{N} = (L \underset{\varrho}{\overset{\delta}{\rightrightarrows}} N)$, where

- N is the set of nodes,
- $L = \sum_{N\times N} \mathcal{N}_{mn}$ is the set of links,
- $\mathcal{N}_{mn} = \langle \delta, \varrho \rangle^{-1}(m, n)$

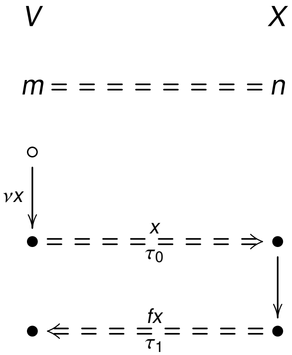
control assignment $© : \mathcal{W} \longrightarrow \wp N$, satisfying

$$A \leqslant B \implies ©A \subseteq ©B$$
$$A \# B \implies ©A \cap ©B = \emptyset$$

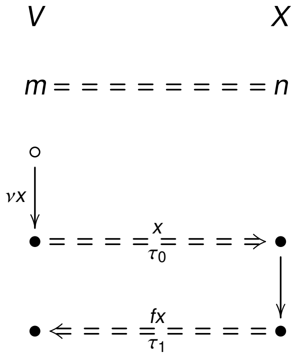
channel typing $\theta : L \longrightarrow \mathcal{C}$,

Outline

Timed challenge-response



Timed challenge-response



$$V : (\nu x)_V \Big(\tau_0 \langle x \rangle_V \triangleright \tau_1 (fx)_V \Longrightarrow \exists X.\, d(V, X) \leq \frac{c}{2} (\tau_1 - \tau_0) \Big) \quad \text{(crt)}$$

Distance bounding protocols

Idea: Combine (cr) and (crt)

- with **one challenge and two responses**:
  - $r^{VP}x$, satisfying (cr)
  - $f^{VP}x$, satisfying (crt)

Distance bounding protocols

Idea: Combine (cr) and (crt)

- with **one challenge and two responses**:
  - $r^{VP}x$, satisfying (cr)
  - $f^{VP}x$, satisfying (crt)
- with **two challenges and one response**:
  - $c^{VP}y$ and $fr^{VP}(x, y)$, satisfying (cr)
  - x and $fr^{VP}(x, y)$, satisfying (crt)

Distance bounding protocols

Idea: Combine (cr) and (crt)

- with **one challenge and two responses**:
  - $r^{VP}x$, satisfying (cr)
  - $f^{VP}x$, satisfying (crt)
- with **two challenges and one response**:
  - $c^{VP}y$ and $fr^{VP}(x,y)$, satisfying (cr)
  - x and $fr^{VP}(x,y)$, satisfying (crt)
- with **one challenge and one response**:
  - x and $fr^{VP}x$, satisfying

$$V: \ (\nu x)_V \Big(\tau_0 \langle x \rangle_V \qquad \triangleright \qquad \tau_1 (fr^{VP}x)_V$$

$$\Longrightarrow \ \tau_0 \langle x \rangle_V \triangleright (x)_P \triangleright \langle fr^{VP}x \rangle_{P_\triangleright} \triangleright \tau_1 (fr^{VP}x)_V \qquad \text{(crp)}$$

$$\wedge \qquad d(V,P) \leq \tau_1 - \tau_0 \Big)$$

Distance bounding protocols

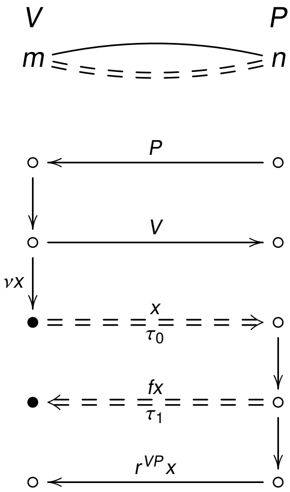
Idea: Combine (cr) and (crt)

- with **one challenge and two responses**:
  - $r^{VP}x$, satisfying (cr)
  - $f^{VP}x$, satisfying (crt)
- with **two challenges and one response**:
  - $c^{VP}y$ and $fr^{VP}(x, y)$, satisfying (cr)
  - x and $fr^{VP}(x, y)$, satisfying (crt)
- with **one challenge and one response**:
  - x and $fr^{VP}x$, satisfying

$$
\begin{aligned}
V : \ (\nu x)_V \Big(& \tau_0 \langle x \rangle_V \qquad \triangleright \qquad \tau_1 (fr^{VP}x)_V \\
\Longrightarrow \ & \tau_0 \langle x \rangle_V \triangleright (x)_P \triangleright \langle fr^{VP}x \rangle_{P_\triangleright} \triangleright \tau_1 (fr^{VP}x)_V \qquad \text{(crp)} \\
\wedge \ & d(V, P) \leq \tau_1 - \tau_0 \Big)
\end{aligned}
$$

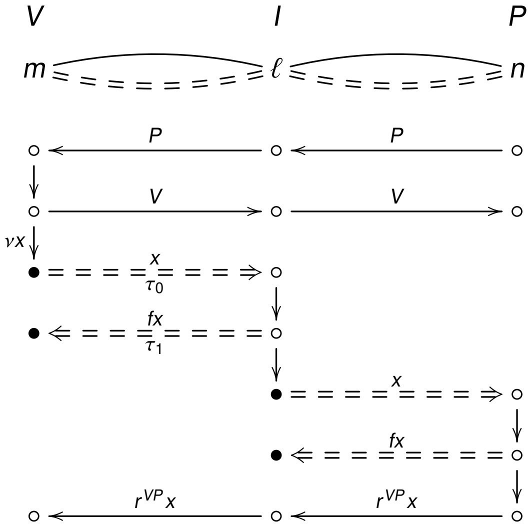
Distance bounding with two responses

Idea



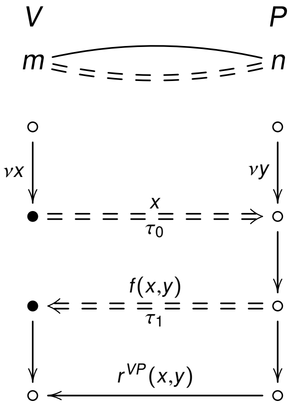
Distance bounding with two responses

Problem

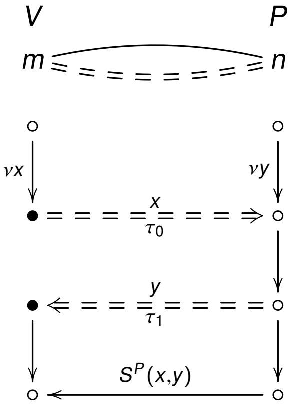


Distance bounding with two responses

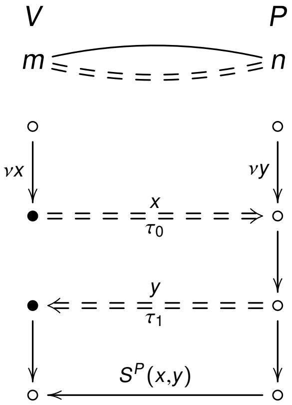
Basic template



Brands-Chaum 1

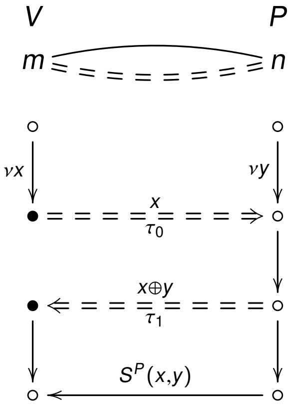


Brands-Chaum 1

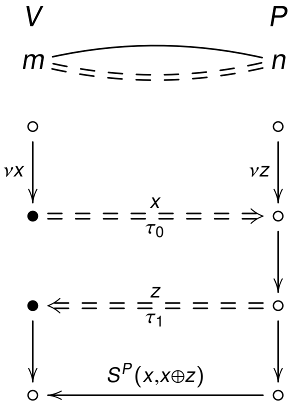


- V : P honest $\Longrightarrow$ $d(V,P) < \tau_1 - \tau_0$
- V : $\forall X.\ X$ responds $\Longrightarrow d(V,X) + d(X,P) < \tau_1 - \tau_0$

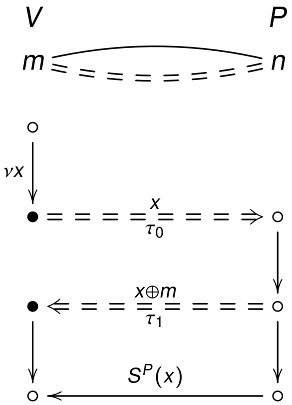
Discharge the honesty assumption?



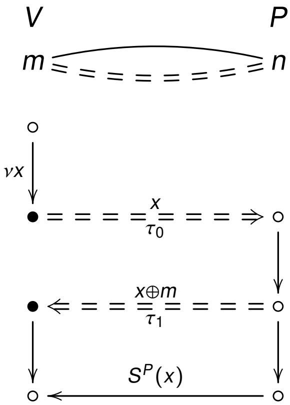
P can still cheat



Brands-Chaum 2

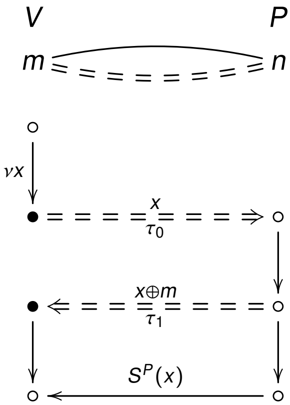


Brands-Chaum 2



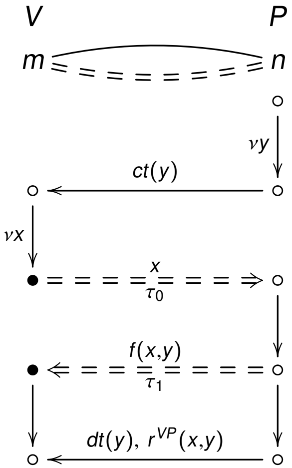
- Peggy cannot cheat

Brands-Chaum 2



- Peggy cannot cheat
- Ivan can impersonate her, and relay $S^P(x)$

Solution 1: Commitment



Digression: Symbolic commitment

Definition

A *commitment schema* over a set of messages $\mathcal{T}$ consists of three publicly known functions

- *commitment ct* : $\mathcal{T} \longrightarrow \mathcal{T}$,
- *decommitment dt* : $\mathcal{T} \longrightarrow \mathcal{T}$, and
- *open ot* : $\mathcal{T} \times \mathcal{T} \longrightarrow \mathcal{T}$,

Digression: Symbolic commitment

Definition

A *commitment schema* over a set of messages $\mathcal{T}$ consists of three publicly known functions such that

- ct is a one-way collision-free function,
- $ot\left(ct(w), dt(w)\right) = w$.
- $dt\left(ot(u, v)\right) = v$.

Digression: Symbolic commitment

Definition

A *commitment schema* over a set of messages $\mathcal{T}$ consists of three publicly known functions such that

- ct is a one-way collision-free function,
- $ot\left(ct(w), dt(w)\right) = w$.
- $dt\left(ot(u, v)\right) = v$.

Use of commitment

- Alice commits to w by sending $u = ct(w)$.
- Later, Alice decommits by sending $v = dt(w)$.
- Bob verifies that $ct\left(ot(u, v)\right) = u$.

Digression: Symbolic commitment

Examples

$$\begin{aligned} ct(w) &= H(w) \\ dt(w) &= w \\ ot(u,v) &= v \end{aligned} \qquad \begin{aligned} ct(w) &= H(w)_0 \\ dt(w) &= w{::}H(w)_1 \\ ot(u,v) &= v_0 \end{aligned} \qquad \begin{aligned} ct(w) &= E(w_0, w_1) \\ dt(w) &= w_0 \\ ot(u,v) &= v{::}D(v,u) \end{aligned}$$

Digression: Symbolic commitment

Examples

$$ct(w) = H(w) \qquad ct(w) = H(w)_0 \qquad ct(w) = E(w_0, w_1)$$
$$dt(w) = w \qquad dt(w) = w{::}H(w)_1 \qquad dt(w) = w_0$$
$$ot(u, v) = v \qquad ot(u, v) = v_0 \qquad ot(u, v) = v{::}D(v, u)$$

where

- $H : \mathcal{T} \longrightarrow \mathcal{T}$ is a one-way collision free function,
- $(-)_0, (-)_1 : \mathcal{T} \longrightarrow \mathcal{T}$ and $(-{::}-) : \mathcal{T} \times \mathcal{T} \longrightarrow \mathcal{T}$ satisfy
  - $(u{::}v)_0 = u$ and $(u{::}v)_1 = v$
  - $(w_0{::}w_1) = w$
- $E, D : \mathcal{T} \times \mathcal{T} \longrightarrow \mathcal{T}$ satisfy
  - $E(x, D(x, y)) = y$, and
  - $E(x, -) : \mathcal{T} \longrightarrow \mathcal{T}$ is one-way for all $x \in \mathcal{T}$.

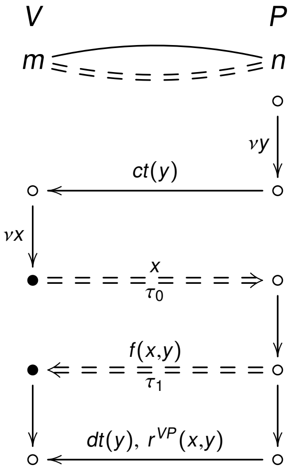
Digression: Symbolic commitment

Homework

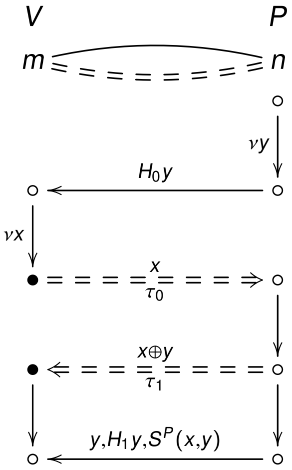
1. Verify that each of the above triples of functions satisfies the requirements for a commitment schema.
2. Given a projection-pairing system $(-)_0, (-)_1, (-::-)$ as in the preceding slide, set
   - $ct(w) = w_0$
   - $dt(w) = w_1$
   - $ot(u, v) = (u::v)$

   Is this a commitment schema? The other way around, does every commitment schema provide a projection-pairing system?

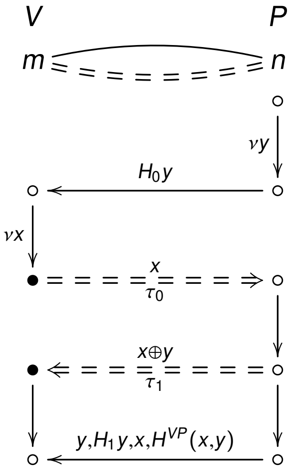
Solution 1: Commitment



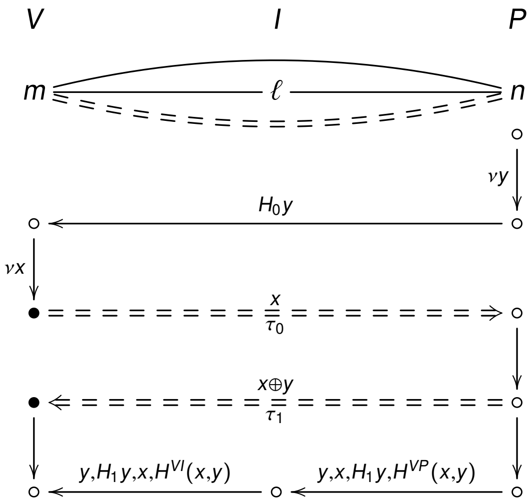
Brands-Chaum 3



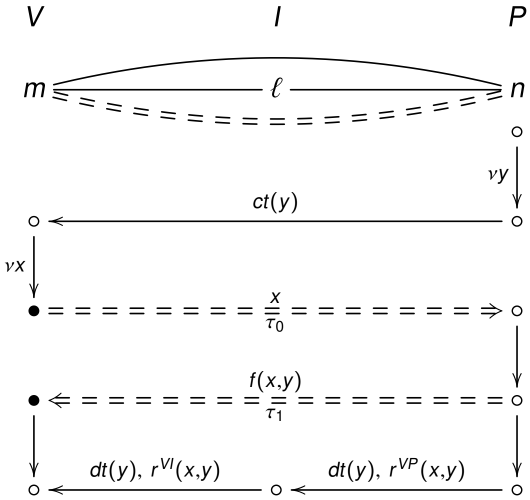
Čapkun-Hubaux



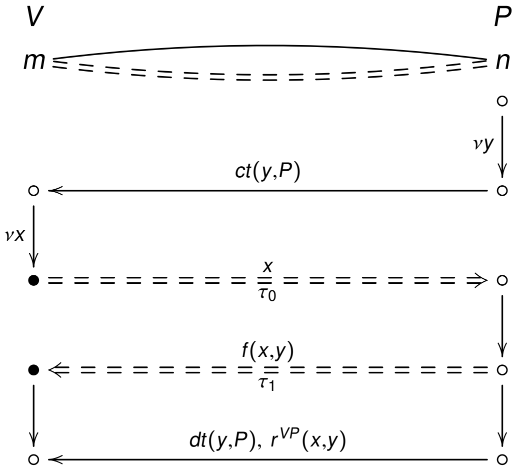
. . . but Peggy's identity can be spoofed



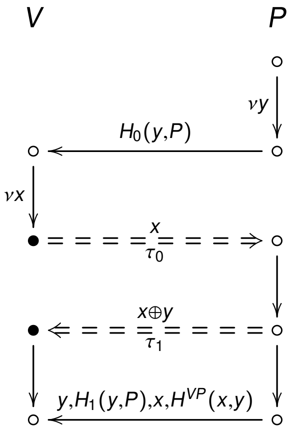
. . . and in general



. . . so we need

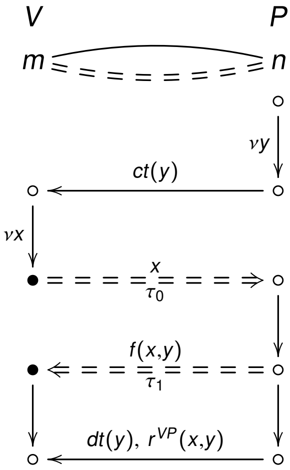


Meadows et al



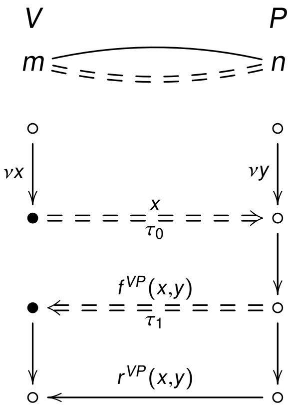
Solution 1: Commitment

This was an implementation of



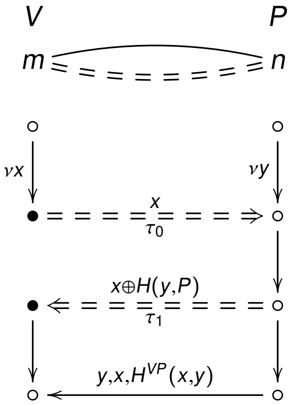
Solution 2: One-way response

Another idea is to commit in the timed response:

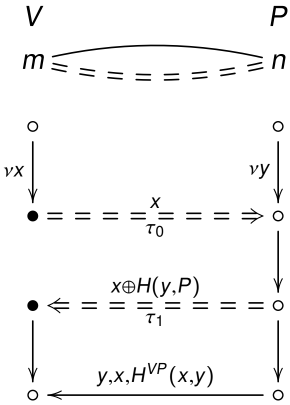


where $f^{VP}(x,-)$ is a one-way function for every x.

Meadows et bo

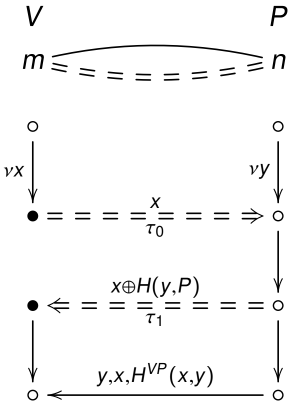


Meadows et bo



- $V: \exists X.\, d(V,X) < \tau_1 - \tau_0 \wedge X \sim P$

Meadows et bo



- $V: \ \exists X.\ d(V,X) < \tau_1 - \tau_0 \wedge X \sim P$
- $V: \ \forall X.\ X \text{ responds} \Longrightarrow d(V,X) + d(X,P) < \tau_1 - \tau_0$

Distance bounding protocols

Idea: Combine (cr) and (crt)

- with **one challenge and two responses**:
  - $r^{VP}x$, satisfying (cr)
  - $f^{VP}x$, satisfying (crt)
- with **two challenges and one response**:
  - $c^{VP}y$ and $fr^{VP}(x, y)$, satisfying (cr)
  - x and $fr^{VP}(x, y)$, satisfying (crt)
- with **one challenge and one response**:
  - x and $fr^{VP}x$, satisfying

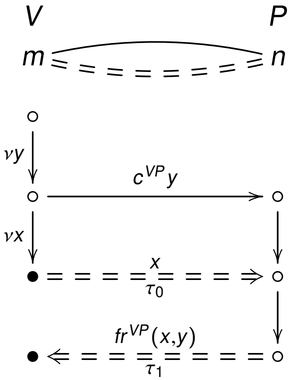
$$V : (\nu x)_V\Big(\tau_0\langle x\rangle_V \quad \triangleright \quad \tau_1(fr^{VP}x)_V$$

$$\Longrightarrow \tau_0\langle x\rangle_V \triangleright (x)_P \triangleright \langle fr^{VP}x\rangle_{P\triangleright} \triangleright \tau_1(fr^{VP}x)_V \qquad \text{(crp)}$$

$$\wedge \quad d(V, P) \leq \tau_1 - \tau_0\Big)$$

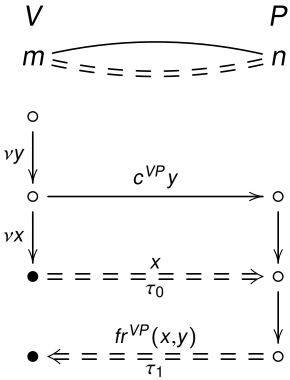
Distance bounding with two challenges

Idea



Distance bounding with two challenges

Idea

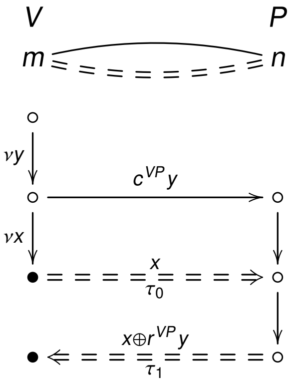


where

- $fr^{VP}(x,-)$ satisfies (cr) for all x
- $fr^{VP}(-,y)$ satisfies (crt) for all y

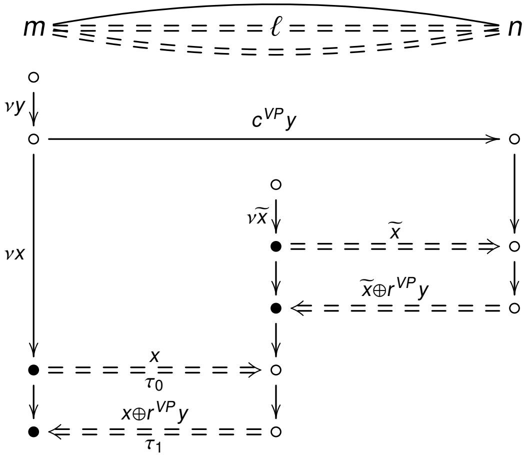
Distance bounding with two challenges

Try



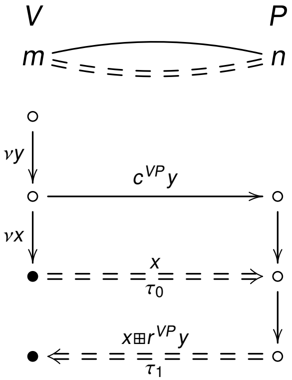
Distance bounding with two challenges

Problem



Distance bounding with two challenges

Idea 2: Find ⊞

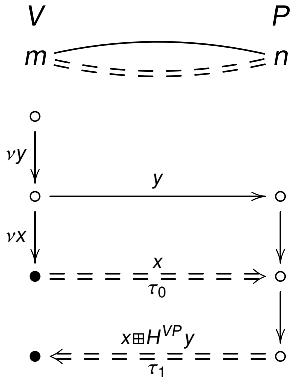


where

- r^{VP} satisfies (cr)
- $x \boxplus (-)$ is one-way function for every x
- $(-) \boxplus y$ satisfies (crt) for every y

Hancke-Kuhn

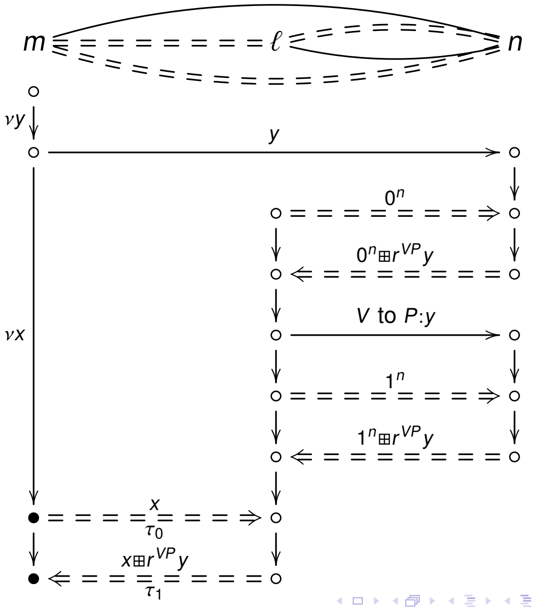
Candidate



$$x \boxplus z \;=\; [z_i^{(x_i)}] \quad \text{where } z = z^{(0)}{::}z^{(1)}$$

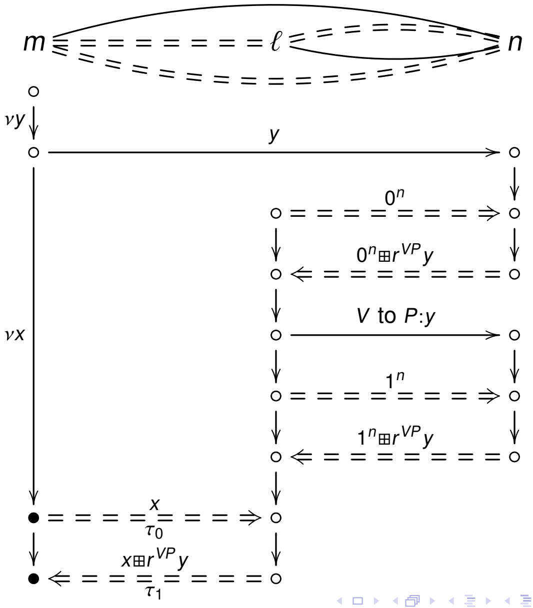
Hancke-Kuhn

Problem



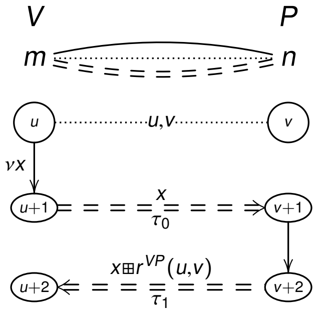
Hancke-Kuhn

Problem: $a \boxplus z, \overline{a} \boxplus z \vdash (-) \boxplus z$, for any a



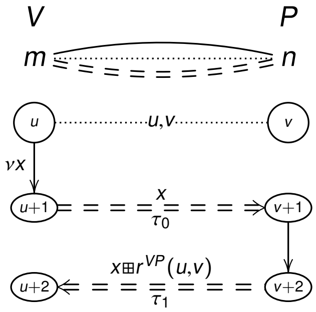
Simple distance bounding template

Idea 3: Use **counters** to disable querying of $(-) \boxplus r^{VP} y$



Simple distance bounding template

Idea 3: Use **counters** to disable querying of $(-) \boxplus r^{VP} y$

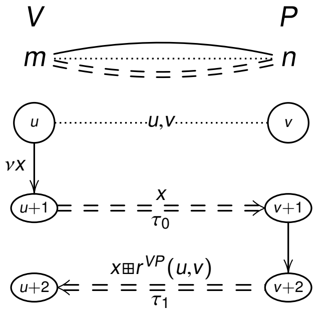


where

- r^{VP} satisfies (cr)
- $x \boxplus (-)$ is one-way function for every x
- $(-) \boxplus z$ satisfies (crt) for every z

Simple distance bounding template

Idea 3: Use **counters** to disable querying of $(-) \boxplus r^{VP} y$



where

- r^{VP} satisfies (cr)
- $x \boxplus (-)$ is one-way function for every x
- $(-) \boxplus z$ satisfies (crt) for every z
- the counters u, v are public, but never reused

Outline

Introduction

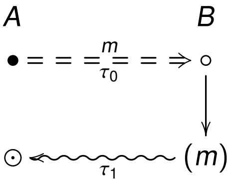
Authentication with timed channels

Authentication with social channels

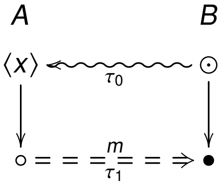
- Social channel and its use
- Social commitment
- Authentication before decommitment
- Authentication after decommitment
- Socially authenticated key exchange
- Security homology

Conclusions

Preliminary example: a timed social protocol



Preliminary example: a timed social protocol



Social channel bandwidth

- $\sigma : \mathcal{T} \longrightarrow \mathcal{T}$: a short digest (hash) function

Social channel bandwidth

- $\sigma : \mathcal{T} \longrightarrow \mathcal{T}$: a short digest (hash) function

such that

- $\sigma \sigma t = \sigma t$
  - "The digest does not change short terms."

Social channel bandwidth

- $\sigma : \mathcal{T} \longrightarrow \mathcal{T}$: a short digest (hash) function

such that

- $\sigma\sigma t = \sigma t$
  - "The digest does not change short terms."
- $\forall s\ \exists t.\ s \neq t\ \wedge \sigma s = \sigma t\ \wedge\ s \vdash t$
  - "For every term s, it is feasible to find a different term t with the same digest."

Social actions

- $\lessdot B \xrightarrow{A}: \beta \gtrdot$ — B shows an action β to A

Social actions

- $<B \xrightarrow{A}: \beta>$ — B shows an action β to A

axiomatized as follows:

- $<B \xrightarrow{A}: \beta> \implies A : \beta_B$
  - "If A sees B perform β, then A knows that B has performed β."

Social actions

- $\lessdot B \xrightarrow{A} : \beta \gtrdot$ — B shows an action β to A

axiomatized as follows:

- $\lessdot B \xrightarrow{A} : \beta \gtrdot \implies A : \beta_B$
  - "If A sees B perform β, then A knows that B has performed β."
- $\lessdot B \xrightarrow{A} : \beta \gtrdot \triangleright \lessdot C \xrightarrow{A} : \gamma \gtrdot \implies A : \beta_B \triangleright \gamma_C$
  - "If A sees β_B before γ_C, then she knows that β_B occurred before γ_C."

Social actions

- $\lessdot B \xrightarrow{A} : t \gtrdot$ — B shows a term t to A

Social actions

- $\lessdot B \xrightarrow{A}: t \gtrdot$ — B shows a term t to A

axiomatized as follows:

- $\lessdot B \xrightarrow{A}: t \gtrdot \implies \sigma t \in \Gamma_A$
  - "If B shows A a term t, then A sees the digest σt."

Social actions

- $\lessdot B \xrightarrow{A}: t \gtrdot$ — B shows a term t to A

axiomatized as follows:

- $\lessdot B \xrightarrow{A}: t \gtrdot \implies \sigma t \in \Gamma_A$
  - "If B shows A a term t, then A sees the digest σt."
- $\lessdot B \xrightarrow{A}: t \gtrdot \implies A : \exists u.\, \sigma u = \sigma t \wedge \lessdot B \xrightarrow{A}: u \gtrdot_B$
  - "If B shows A a term t, then A knows that B has shown her some term with the digest σt."

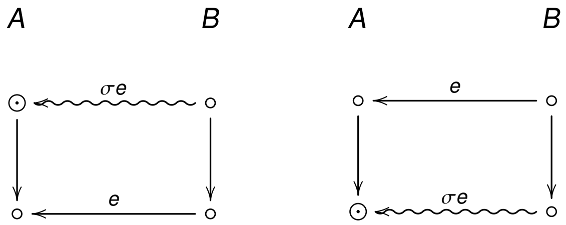
Social actions

Graphic notation

- $\beta_B \rightsquigarrow \odot_A$ represents $\lessdot B \xrightarrow{A} : \beta \gtrdot$
- $\circ_B \overset{\sigma t}{\rightsquigarrow} \odot_A$ represents $\lessdot B \xrightarrow{A} : t \gtrdot$

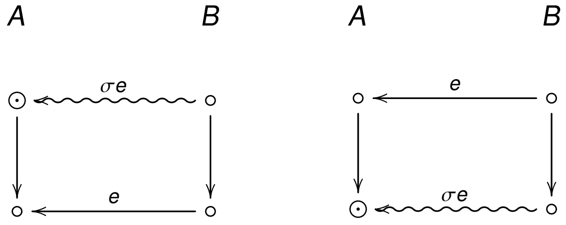
Socially authenticated key distribution

Bob announces his public key



Socially authenticated key distribution

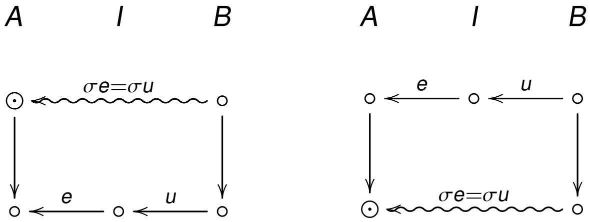
Bob announces his public key



- $e, \sigma e \in \Gamma_A$
- $A : \; B \text{ honest} \implies \exists u.\, \sigma u = \sigma e \wedge \langle B \xrightarrow{A} : u \rangle_B$

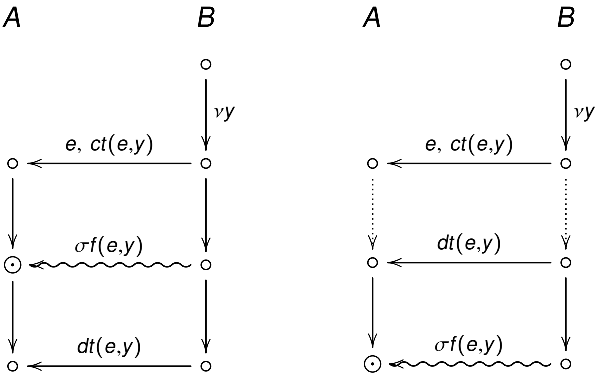
Socially authenticated key distribution

...but Ivan may have replaced it

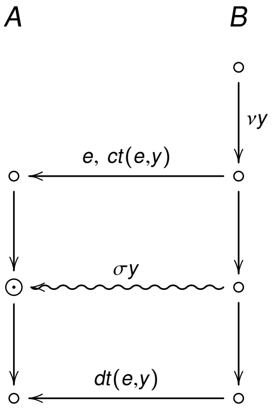


- $e, \sigma e \in \Gamma_A$
- $A : B \text{ honest} \implies \exists u.\, \sigma u = \sigma e \wedge \langle B \xrightarrow{A} : u \rangle_B$

Social commitment

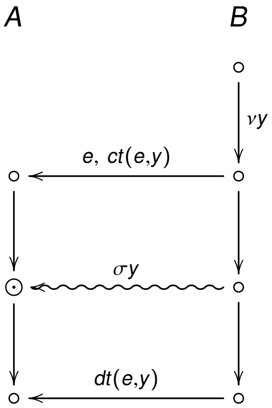


Authentication before decommitment



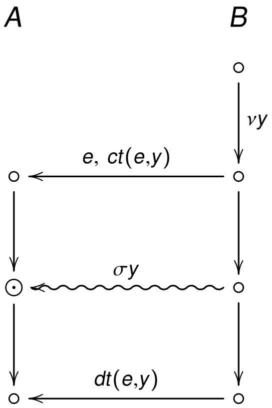
- $A : \exists y.\, \sigma y = s \wedge \lessdot B \xrightarrow{A} : s \gtrdot_B$

Authentication before decommitment



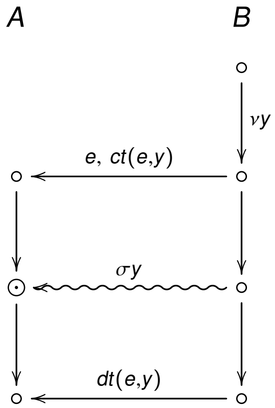
- $A:\ B$ honest $\Longrightarrow \exists y.\ \lessdot B \xrightarrow{A}: \sigma y \gtrdot_B$

Authentication before decommitment



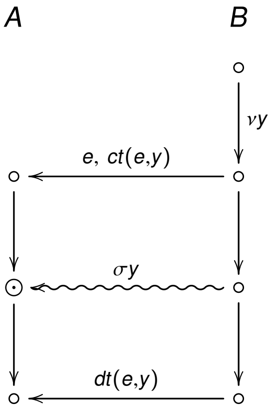
- A : B honest $\Longrightarrow \exists u \exists y.\ \langle u, ct(u,y)\rangle_B \trianglerighteq \lessdot \sigma y \gtrdot_B$

Authentication before decommitment



- A : B honest $\Longrightarrow \exists u.\ (\nu y)_B \trianglerighteq \langle u, ct(u,y)\rangle_B \trianglerighteq \lessdot \sigma y \gtrdot_B$

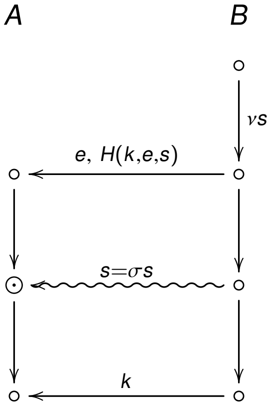
Authentication before decommitment



- A : B honest $\Longrightarrow (\nu y)_B \trianglerighteq \langle e, ct(e,y)\rangle_B \trianglerighteq \lessdot\sigma y\gtrdot_B \trianglerighteq \langle dt(e,y)\rangle_B$

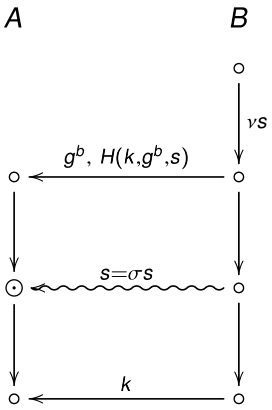
Authentication before decommitment

Wong-Stajano template



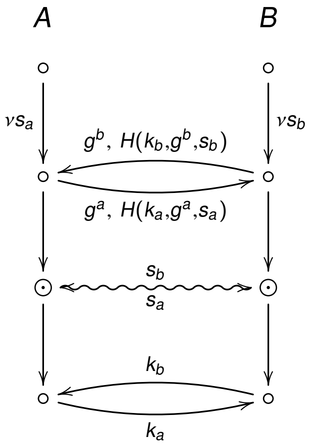
Authentication before decommitment

Wong-Stajano-$\frac{1}{2}$



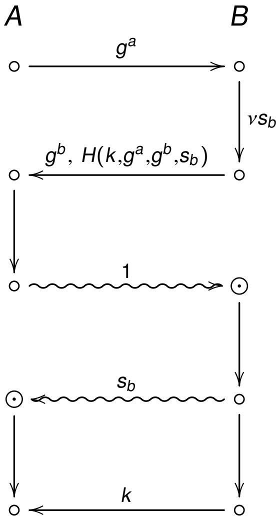
Authentication before decommitment

Wong-Stajano

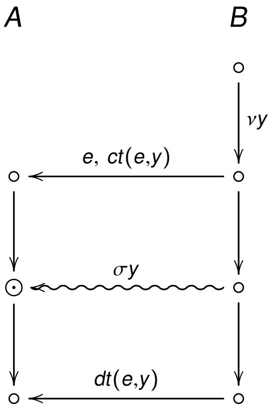


Authentication before decommitment

Wong-Stajano 3



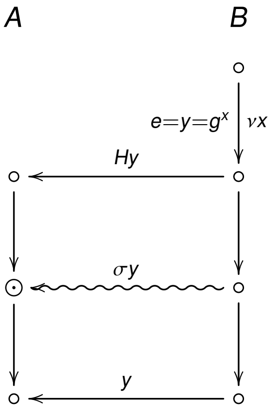
Authentication before decommitment



- A : B honest $\Longrightarrow (\nu y)_B \trianglerighteq \langle e, ct(e,y)\rangle_B \trianglerighteq \lessdot\sigma y\gtrdot_B \trianglerighteq \langle dt(e,y)\rangle_B$

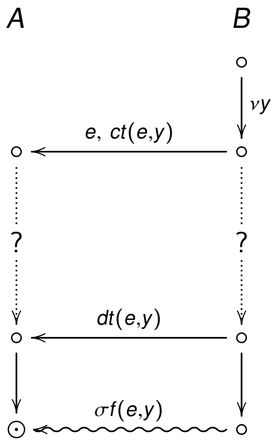
Authentication before decommitment

Hoepman-$\frac{1}{2}$

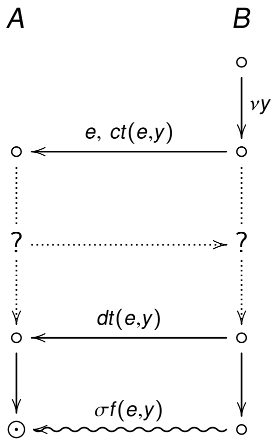


- $A : B \text{ honest} \implies (\nu x)_B \trianglerighteq \langle H(g^x)\rangle_B \trianglerighteq \lessdot\sigma(g^x)\gtrdot_B \trianglerighteq \langle g^x\rangle_B$

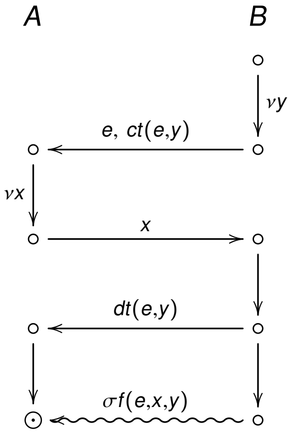
Authentication after decommitment



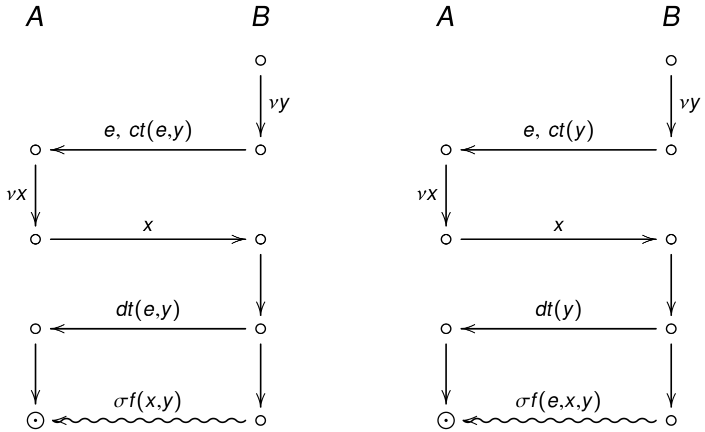
Authentication after decommitment



Authentication after decommitment

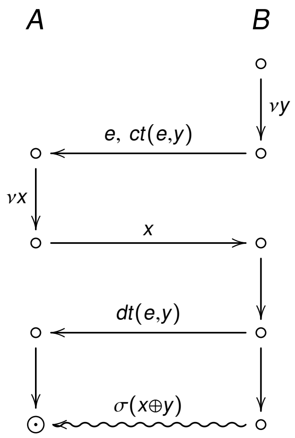


Authentication after decommitment



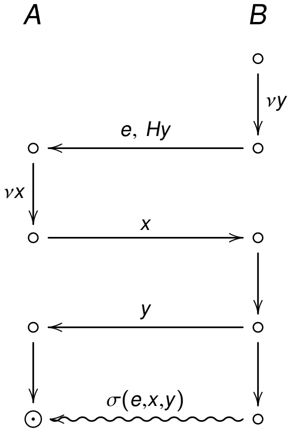
Authentication after decommitment

Vaudenay: SAS-$\frac{1}{2}$



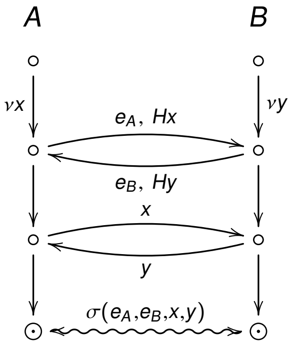
Authentication after decommitment

Nguyen-Roscoe: HCBK-$\frac{1}{2}$



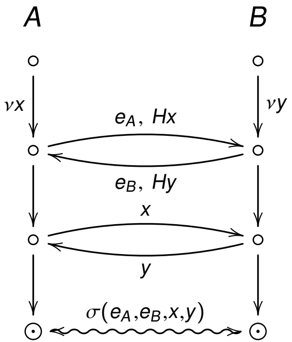
Mutual authentication after decommitment

Nguyen-Roscoe: HCBK (2-party)



Mutual authentication after decommitment

Nguyen-Roscoe: HCBK (2-party)



Assumption: Initiator establishes the order

Mutual authentication after decommitment

Nguyen-Roscoe: HCBK (2-party)

$$\Big((\nu x)_A \ \langle e_A, Hx\rangle_A \ (u_1, u_2)_A \ \otimes$$
$$(\nu y)_B \ \langle e_B, Hy\rangle_B \ (v_1, v_2)_B\Big) \ ;$$

$$\Big(\langle x\rangle_A \ (u_3)_A \ (u_1, u_2/e_B, Hu_3)_A \ \lessdot \sigma(e_A, e_B, x, u_3) \gtrdot_A \ \otimes$$
$$\langle y\rangle_B \ (v_3)_B \ (v_1, v_2)/e_A, Hv_3)_B \ \lessdot \sigma(e_A, e_B, v_3, y) \gtrdot_B \Big)$$

Multi-party authentication after decommitment

Nguyen-Roscoe: HCBK

Assumptions (to be discharged)

- agreed ordering of the principals

Multi-party authentication after decommitment

Nguyen-Roscoe: HCBK

Assumptions (to be discharged)

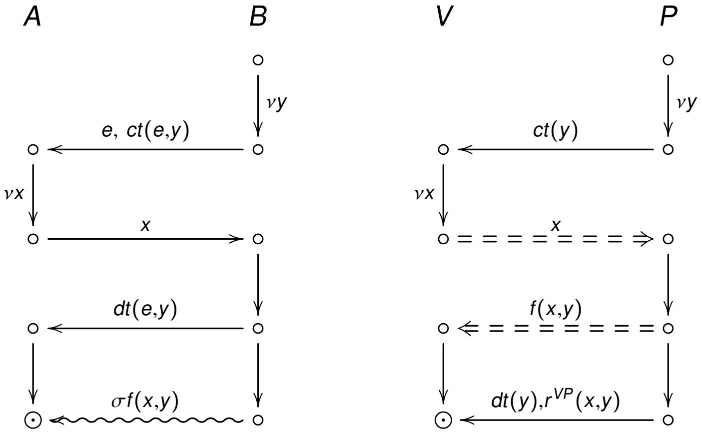
- agreed ordering of the principals
    - all principals must digest at the same payload

Multi-party authentication after decommitment
Nguyen-Roscoe: HCBK

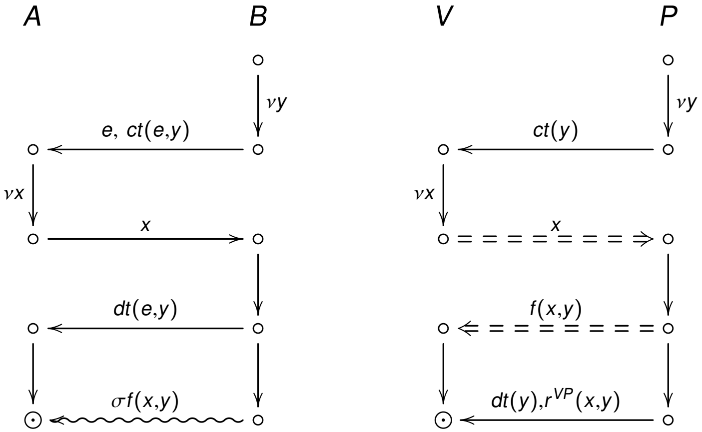
Assumptions (to be discharged)

- agreed ordering of the principals
  - all principals must digest at the same payload
- social protocol to compare the digests

Structural similarity — conceptual difference



Structural similarity — conceptual difference



Social authentication is not challenge-response:

x on the left is not a challenge, but a binder, analogous to y.

Outline

Summary

- computation is becoming pervasive: in physical space
- new security landscape
  - need stronger authentication: proximity...
  - weaker cryptography: low power devices
  - bootstrap distance, proximity, routing...