

Inductive Proofs of Computational Secrecy

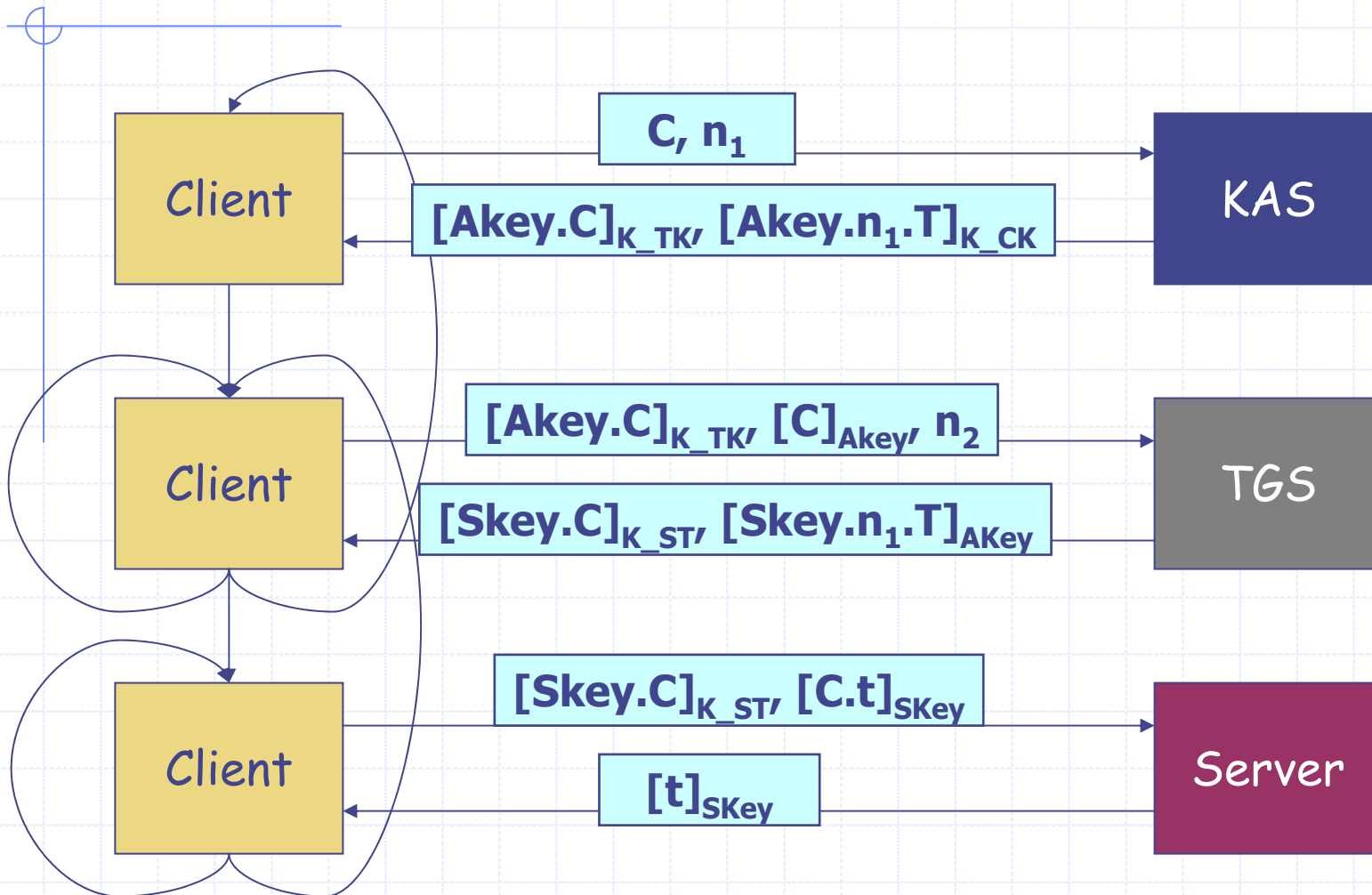
Arnab Roy

joint work with

Anupam Datta, Ante Derek, John C. Mitchell

Stanford University and Carnegie Mellon University

Kerberos Overview



1. Secrecy of Keys

2. Modular

3. Computationally Sound

Proving Computational Security

◆ Symbolic Trace Property

- Proof technique: Induction over protocol actions

◆ Computational Security

- Proof technique: Reduction to games defining security of cryptographic primitives

◆ Approach

- Symbolic Trace Property \Rightarrow Computational Security

Protocols

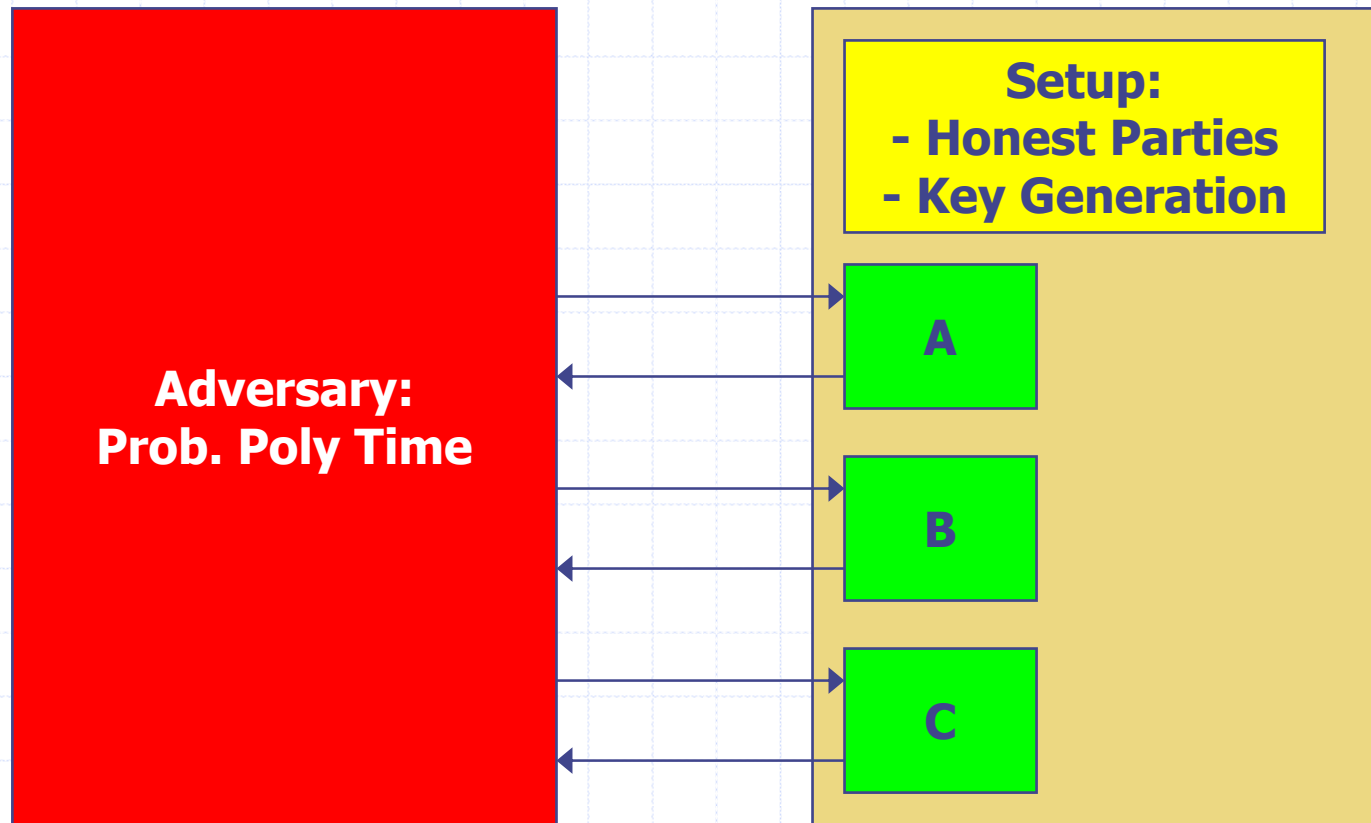
◆ Distributed Programs

- Protocol is a fixed set of 'roles' written as programs
- A 'thread' is an instance of a role being executed by a principal
- A single principal can execute multiple threads

◆ Actions in a role

- `send m; recv m;`
- `m' := pair m0, m1; (m0, m1) := unpair m;`
- `m' := enc m, k; m' := dec m, k;`
- `new m; match m as m'; ...`

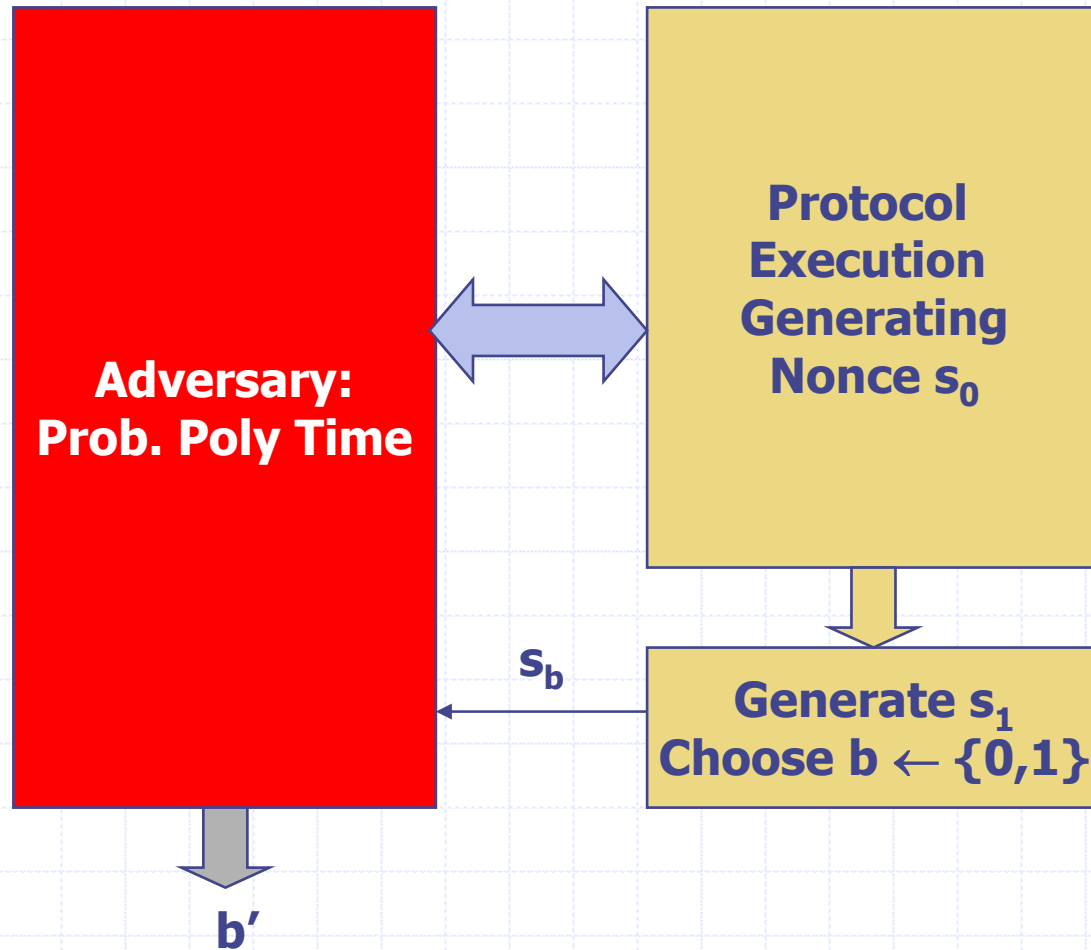
Protocol Execution Model



◆ Result

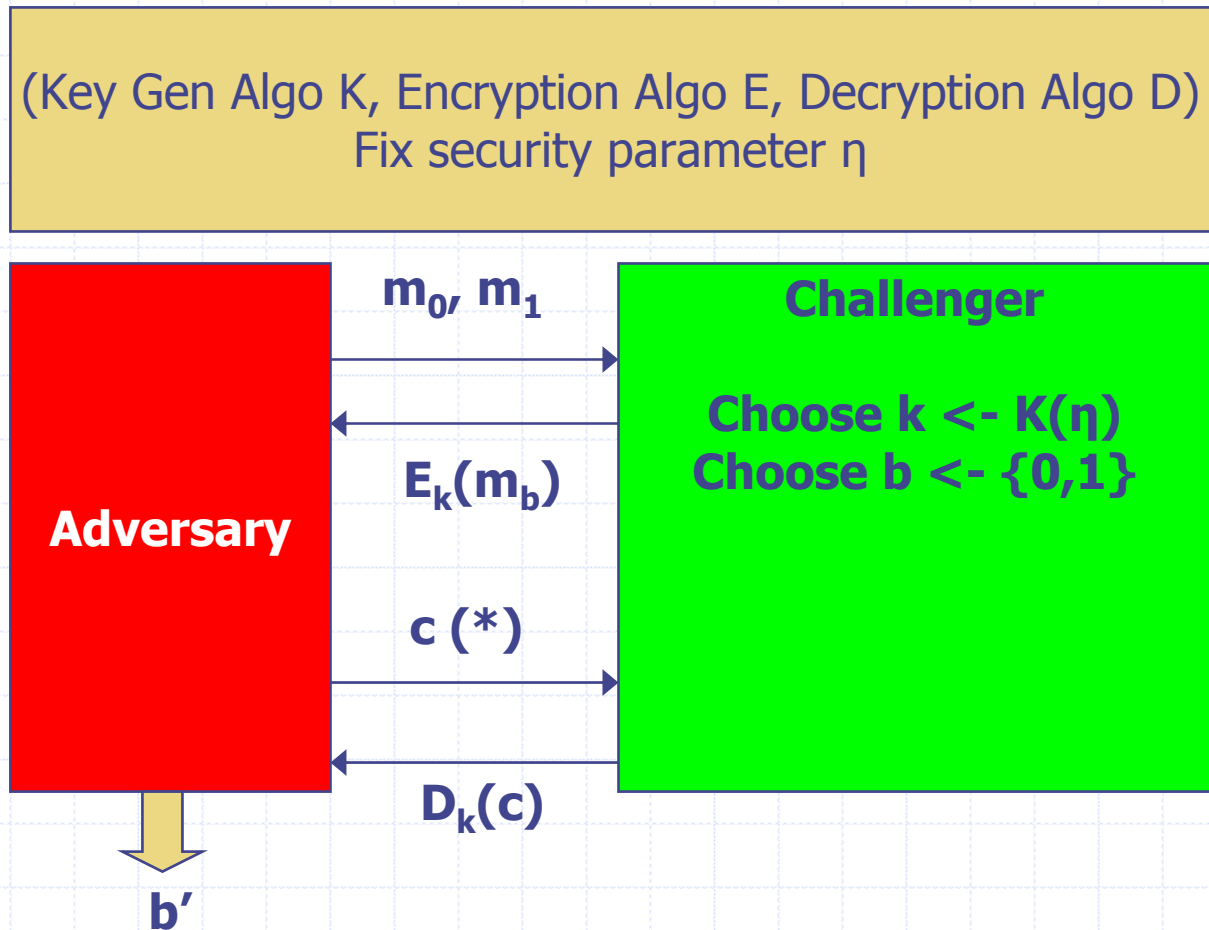
- Set of *computational traces*

Secrecy Notion: Real or Random Game



$$\text{Adv}(A, \eta) = \Pr[b' = b] - 1/2$$

IND-CCA Game



(*): c 's should be different from any encryption response
 $\text{Adv}(A, \eta) = \Pr[b' = b] - 1/2$

IND-CCA Security Definition

- ◆ A negligible function $\nu(x): \mathbb{N} \rightarrow \mathbb{R}$ is a function that asymptotically decreases faster than the reciprocal of any polynomial in x , i.e.,

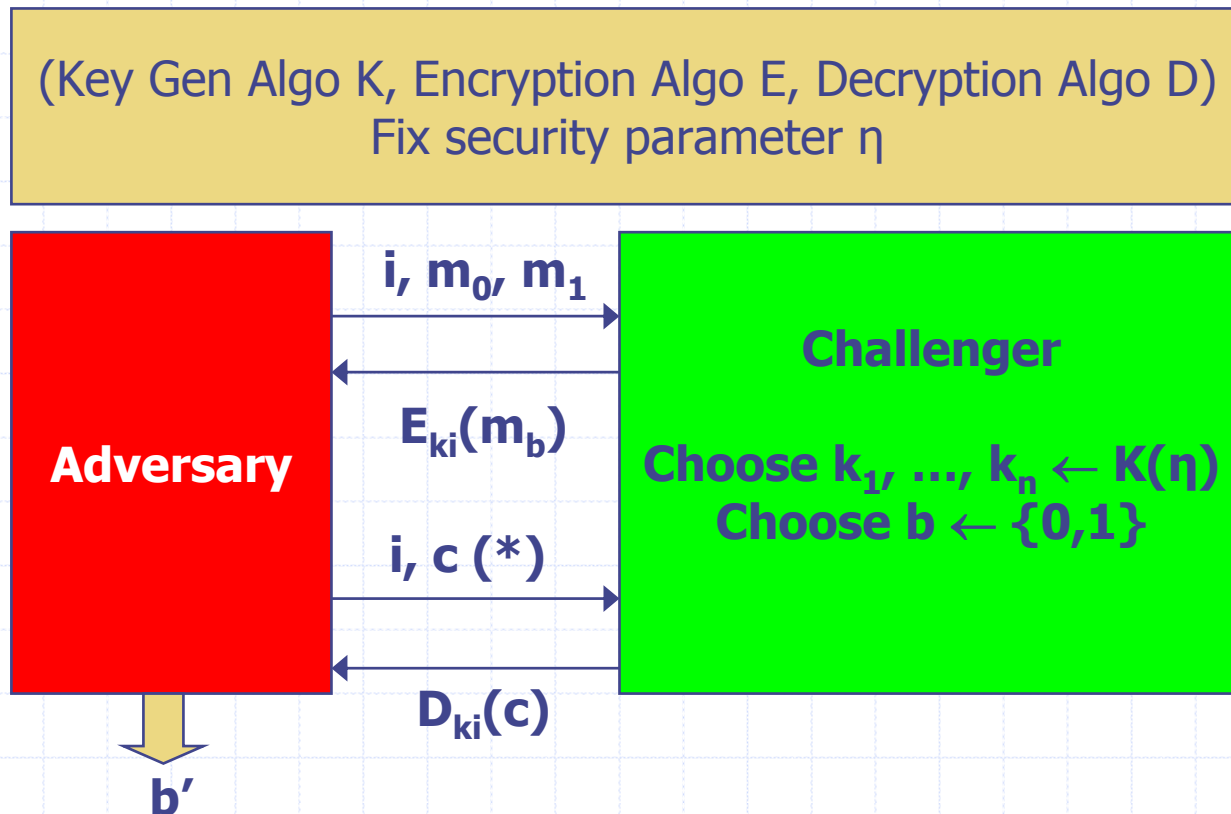
$$\forall \text{polynomial } p. \exists N. \forall n > N. \nu(n) < 1/p(n)$$

- ◆ An encryption scheme $ES = \langle K, E, D \rangle$ is IND-CCA secure if

$$\forall \text{Prob-Polytime } A.$$

$$\text{Adv}(A, \eta) \text{ is a negligible function of } \eta$$

n-IND-CCA Game



(*): c 's should be different from any encryption response corres. to same i

$$\text{Adv}(A, \eta) = \Pr[b' = b] - \frac{1}{2}$$

◆ [BBM00] shows that an encryption scheme is n-IND-CCA secure iff it is IND-CCA secure.

Secrecy Notion: Indistinguishability

◆ Secrecy Property:

- Indistinguishability for the nonce holds if

∀ Prob-Polytime A .

$\text{Adv}(A, \eta)$ is a negligible function of η

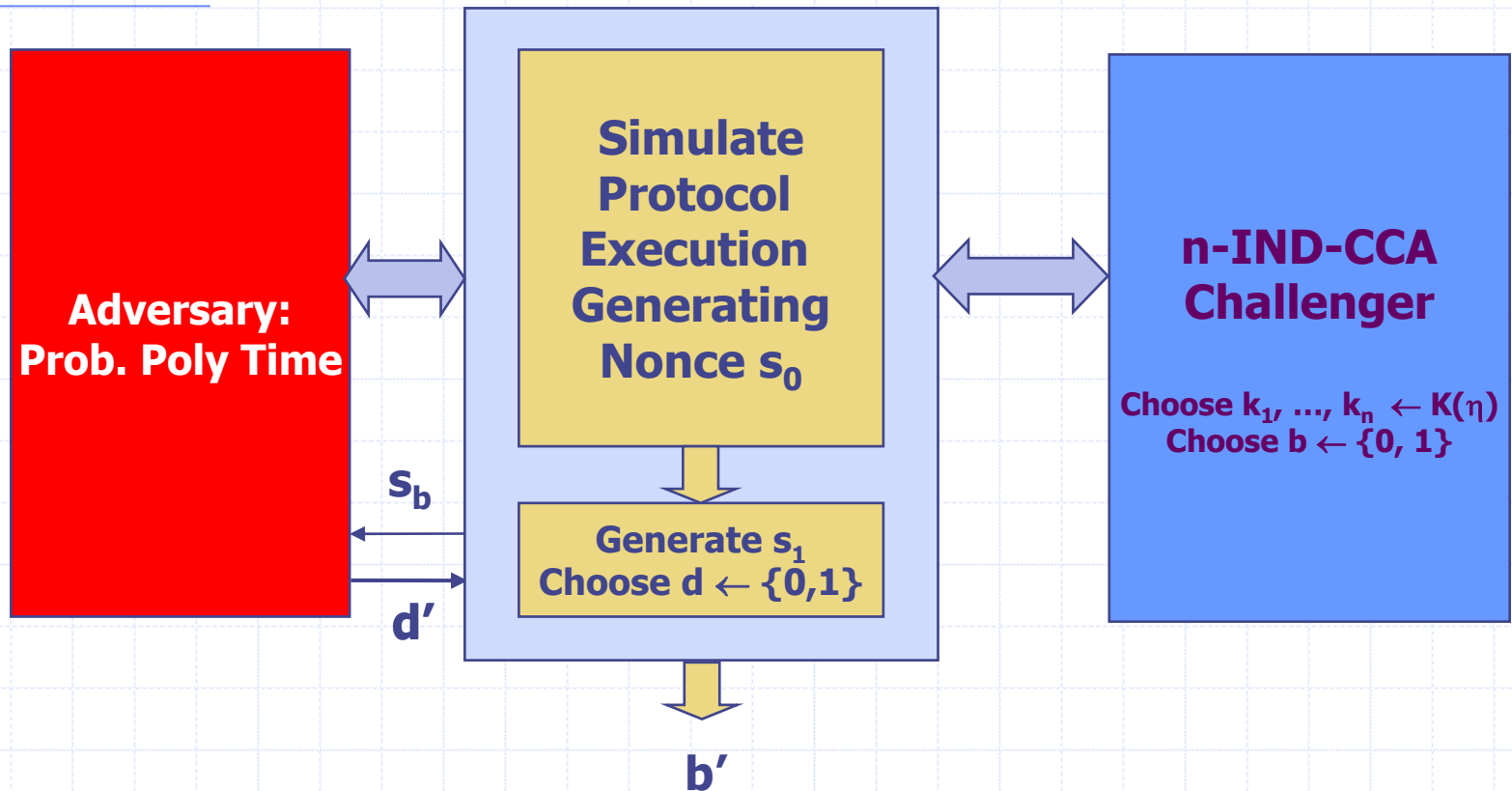
◆ We want to prove:

- If the encryption scheme is IND-CCA secure then indistinguishability for the nonce holds.

◆ Proof Strategy:

- Reduction! – if an adversary can break protocol then there is an adversary which can break CCA (contrapositive)

Reduction

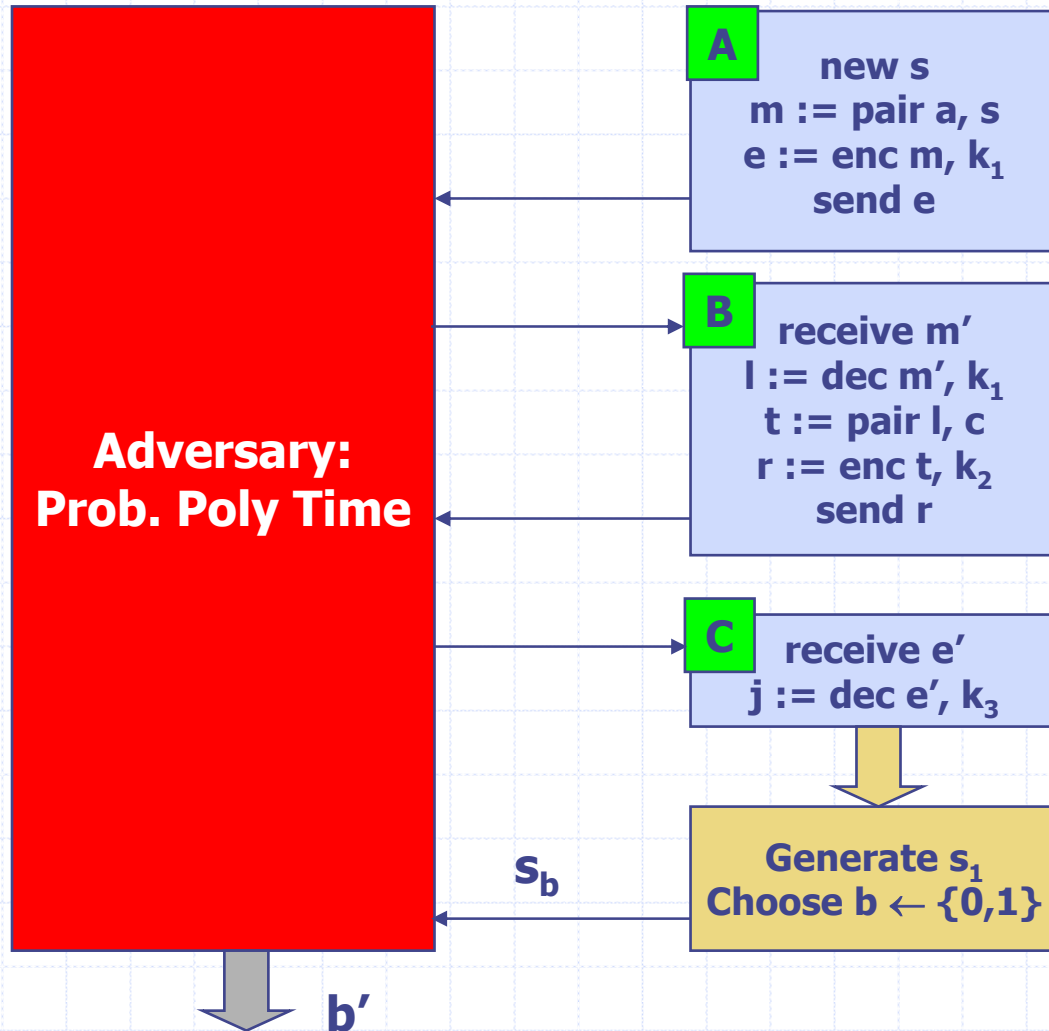


Show that:

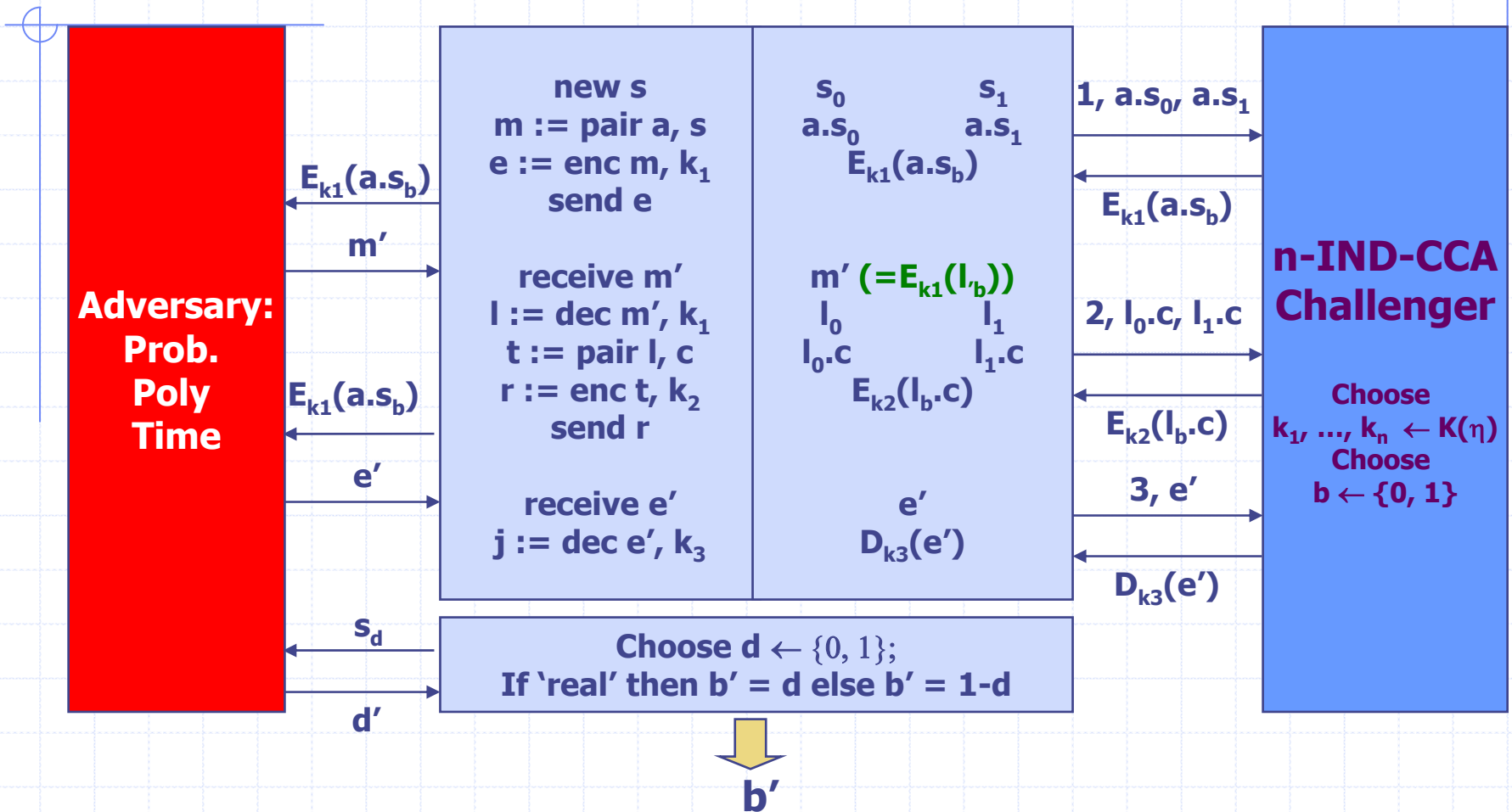
If for nonce indist game $\text{Adv}(A, \eta)$ is non-negligible

Then for Simulator S , $\text{Adv}(S, \eta)$ against n-IND-CCA game is non-negligible

Protocol example



Reduction



$\text{Adv}(A, \eta)$ for nonce indist game = $\text{Adv}(S, \eta)$ against n-IND-CCA game

Secretive Protocols

- ◆ A trace is a *secretive* trace with respect to nonce s and set of keys K if the following properties hold for every thread belonging to honest principals:
 - The thread which generates s , ensures that s is encrypted with a key k in K in any message sent out.
 - Whenever a thread decrypts a message with a key k in K and parses the decryption, it ensures that the results are re-encrypted with some key k' in K in any message sent out.
- ◆ A protocol is *secretive* if it overwhelmingly produces secretive traces.
- ◆ An inductive property over actions of honest parties
 - Formalization in Computational Protocol Composition Logic.

Relating “Secretive” Protocols to Computational Secrecy

◆ Theorem:

If

- the protocol is “secretive”
- the nonce-generator is honest
- the key-holders are honest

**Inductive
property of
protocol**

Then

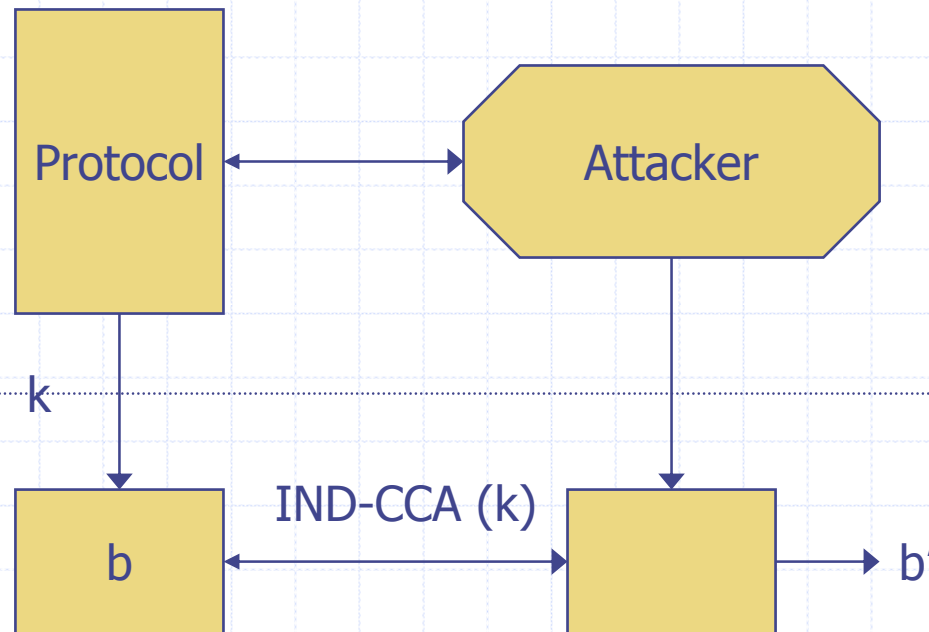
- the key generated from the nonce satisfies key indistinguishability

Proof is by reduction to a multi-party IND-CCA game – one time soundness proof

Good Keys

[DDMW06]

- ◆ Key is “good” for a certain purpose
- ◆ Intuition: Exchanged key is good for encrypting messages if no attacker can win an appropriate game played with that key.



Relating “Secretive” Protocols to “Good” Keys

◆ Theorem:

If

- the protocol is “secretive”
- the nonce-generator is honest
- the nonce may be used as a key
- the key-holders are honest

**Inductive
property of
protocol**

Then

- the key generated from the nonce is a “good” key

**Proof is by reduction to a multi-
party IND-CCA game – one time
soundness proof**

Computational PCL

- ◆ Proof system for direct reasoning
 - $\text{Verify}(X, m, Y) \wedge \text{Honest}(Y) \Rightarrow \text{Sign}(Y, m)$
 - No explicit use of probabilities and computational complexity
 - No explicit arguments about actions of attackers
- ◆ Semantics capture idea that properties hold with high probability against PPT attackers
- ◆ Soundness implies result equivalent to security proof by cryptographic reductions

Proof System to Establish “Secretive” Protocol – “Good” terms

- ◆ Proof of construction of good terms is carried out inductively over actions of honest principals

G0 $\text{Good}(X, a, s, \mathcal{K})$, if a is of an atomic type different from nonce or key

G1 $\text{New}(Y, n) \wedge n \neq s \supset \text{Good}(X, n, s, \mathcal{K})$

G2 $[\text{receive } m;]_X \text{Good}(X, m, s, \mathcal{K})$

G3 $\text{Good}(X, m, s, \mathcal{K}) [a]_X \text{Good}(X, m, s, \mathcal{K})$, for all actions a

G4 $\text{Good}(X, m, s, \mathcal{K}) [\text{match } m \text{ as } m';]_X \text{Good}(X, m', s, \mathcal{K})$

G5 $\text{Good}(X, m_0, s, \mathcal{K}) \wedge \text{Good}(X, m_1, s, \mathcal{K}) [m := m_0.m_1;]_X \text{Good}(X, m, s, \mathcal{K})$

G6 $\text{Good}(X, m, s, \mathcal{K}) [\text{match } m \text{ as } m_0.m_1;]_X \text{Good}(X, m_0, s, \mathcal{K}) \wedge \text{Good}(X, m_1, s, \mathcal{K})$

G7 $\text{Good}(X, m, s, \mathcal{K}) \vee k \in \mathcal{K} [m' := \text{symenc } m, k;]_X \text{Good}(X, m', s, \mathcal{K})$

G8 $\text{Good}(X, m, s, \mathcal{K}) \wedge k \notin \mathcal{K} [m' := \text{symdec } m, k;]_X \text{Good}(X, m', s, \mathcal{K})$

Proof System to Establish “Secretive” Protocol – Induction

- ◆ A protocol is “secretive” if all honest participants send out only “good” terms.

\forall roles ρ in protocol Q .

\forall segments P in role ρ .

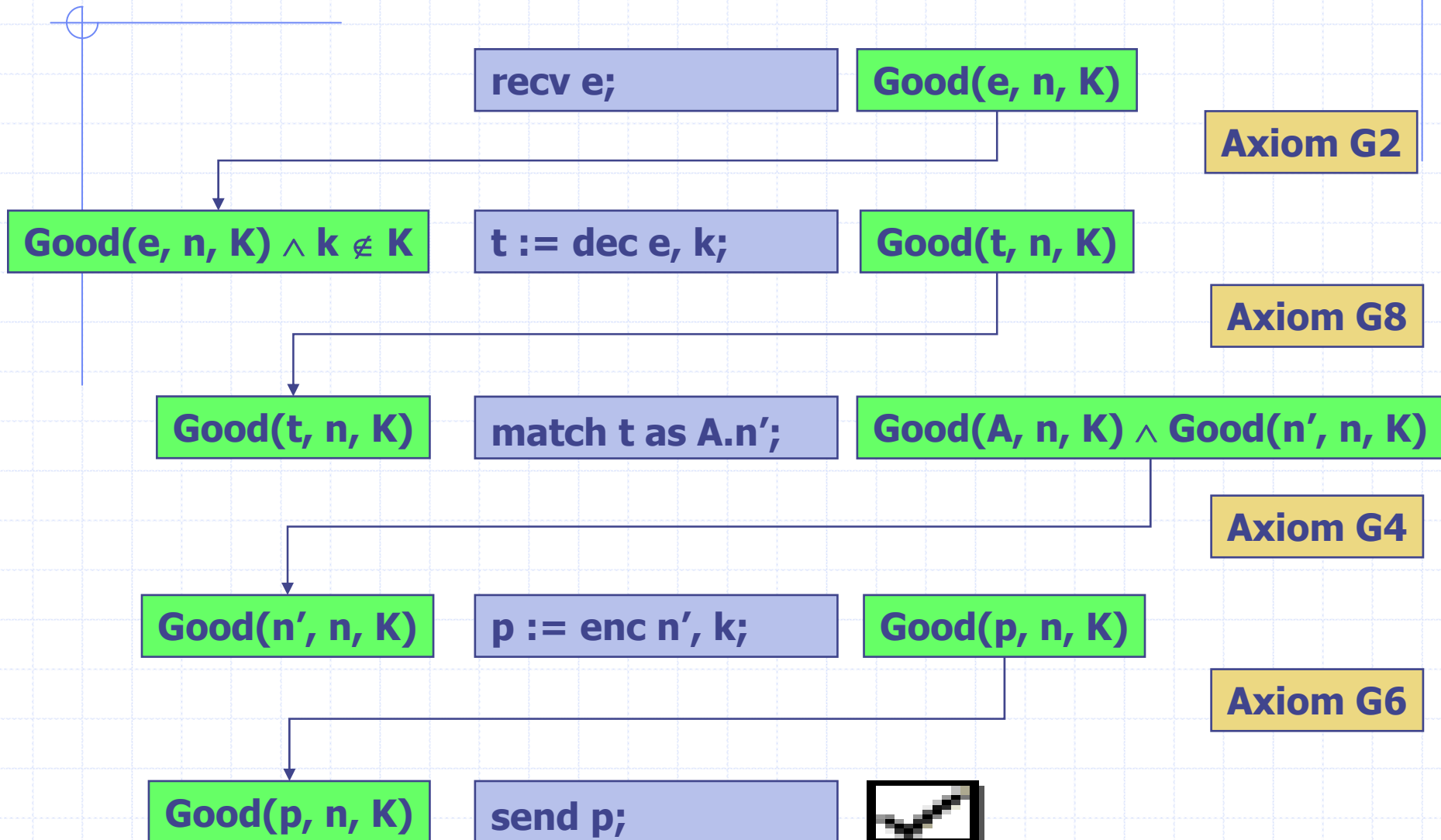
$\text{SendGood}(X, s, K) [P]_x \Phi \supset \text{SendGood}(X, s, K)$
 $Q \vdash \Phi \supset \text{Secretive}(s, K)$

Example

- ◆ Let n be the putative secret and $K = \{k_1, k_2, \dots\}$
- ◆ We want to prove that the protocol satisfies $\text{Secretive}(n, K)$
- ◆ Consider the following fragment of the protocol:

```
recv e;  
t := dec e, k;  
match enc as A.n;  
p := enc n, k;  
send p;
```

Case: $k \notin K$



Case: $k \in K$

recv e;

t := dec e, k;

match t as A.n';

$k \in K$

p := enc n', k;

Good(p, n, K)

Axiom G7

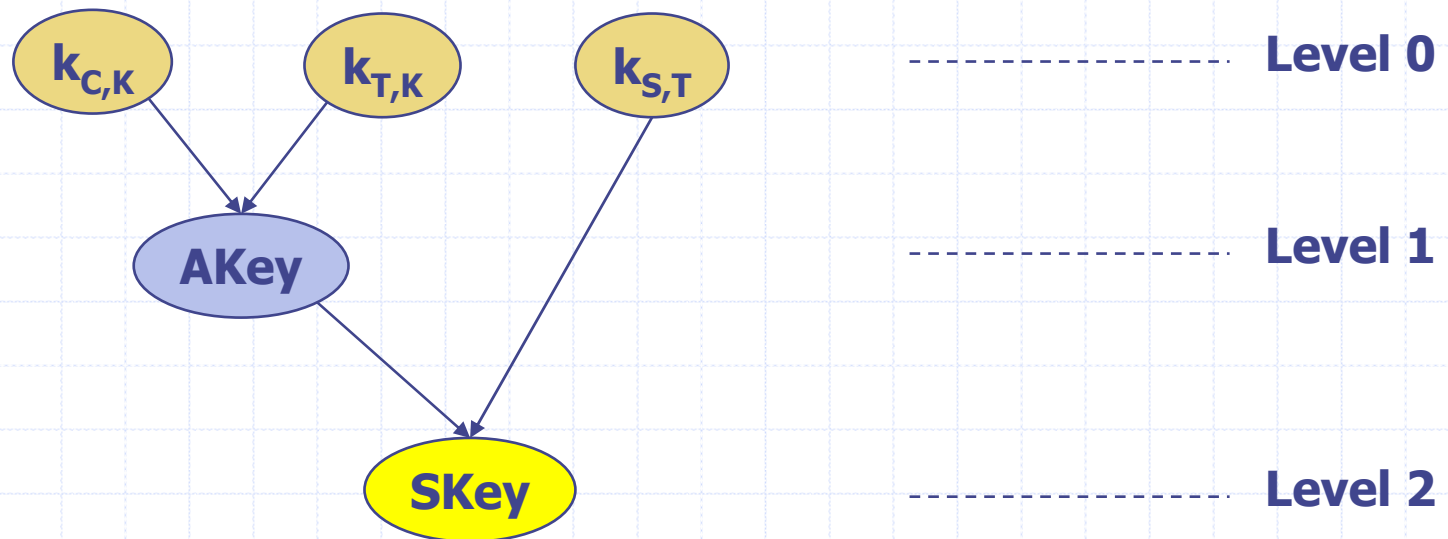
Good(p, n, K)

send p;



Key Graphs

- ◆ Many interesting protocols establish a hierarchy of keys. For example – Kerberos, IEEE 802.11i



Guarantees for the Client

If Client C completes the protocol with Kerberos Authentication Server K, Ticket Granting Server T and Application Server S then C can infer the following guarantees:

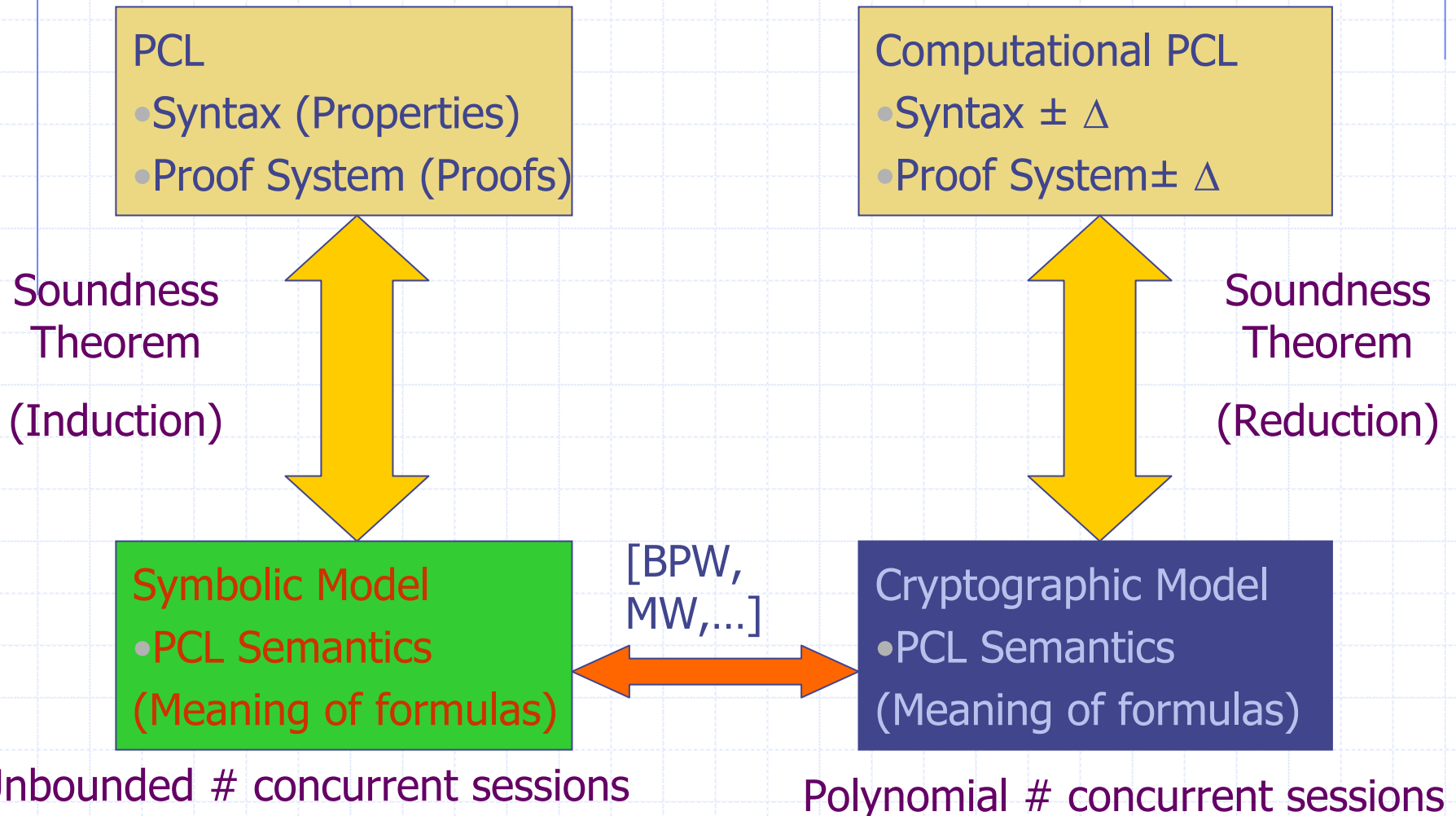
Type	Honesty Assumption	Guarantee
Authenticity	C, K	A message containing a valid ticket granting ticket was indeed sent by K intended for (C, T), with overwhelming probability.
Authenticity	C, K, T	A message containing a valid server ticket was indeed sent by T intended for (C, S), with overwhelming probability.
Secrecy	C, K, T	AKey is a good key for C, K and T.
Secrecy	C, K, T, S	SKey is a good key for C, K, T and S.

Related Work

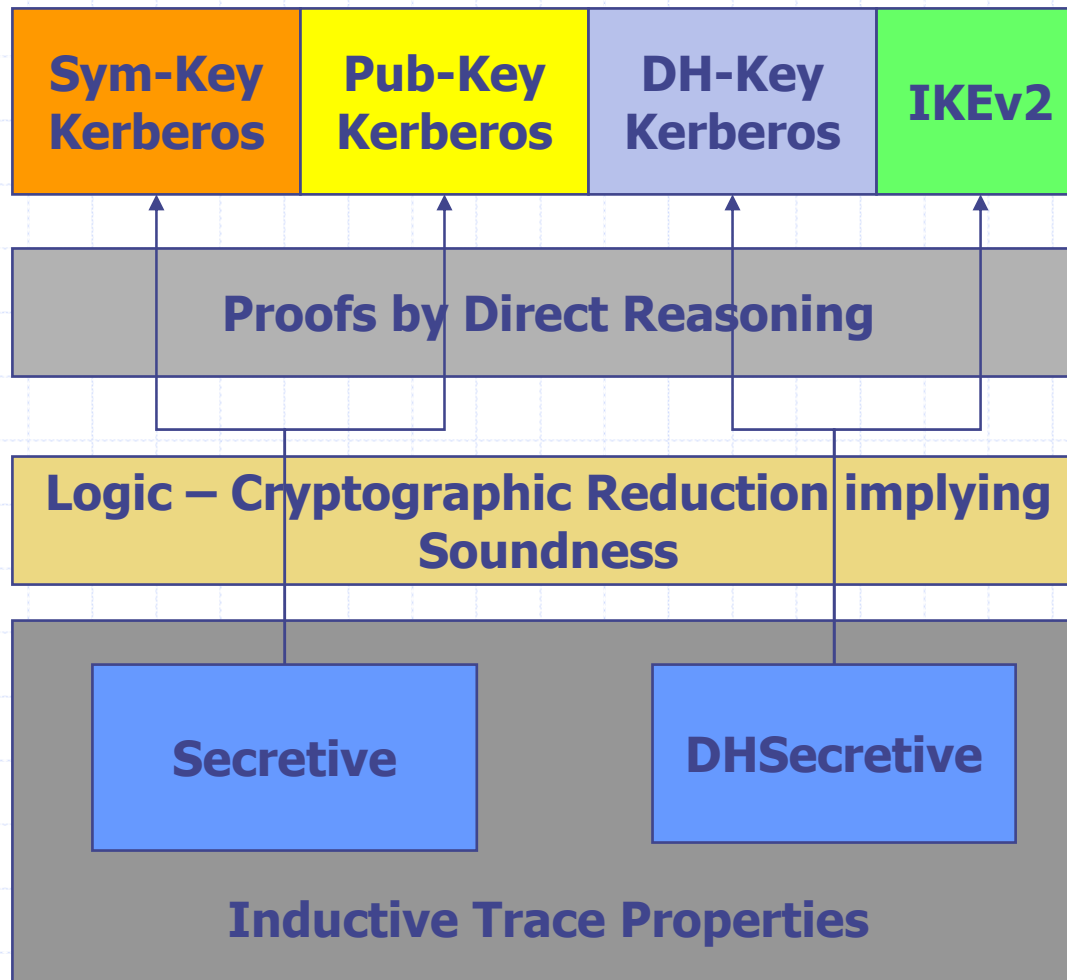
- ◆ Backes et al - "Cryptographically Sound Security Proofs for Basic and Public-key Kerberos", ESORICS 2006
- ◆ Our proofs and proof system has these advantages:
 - Modularity – composition theorems
 - Secrecy of keys
 - Extension to Diffie-Hellman – upcoming talk in TGC 2007

PCL: Big Picture

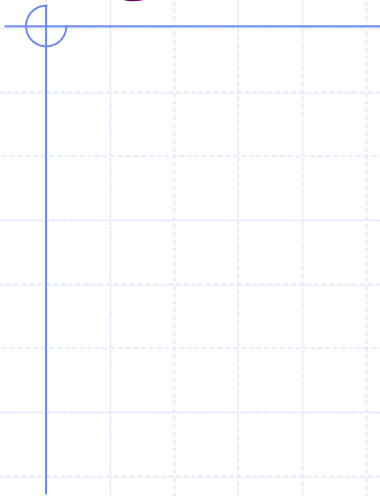
High-level proof principles



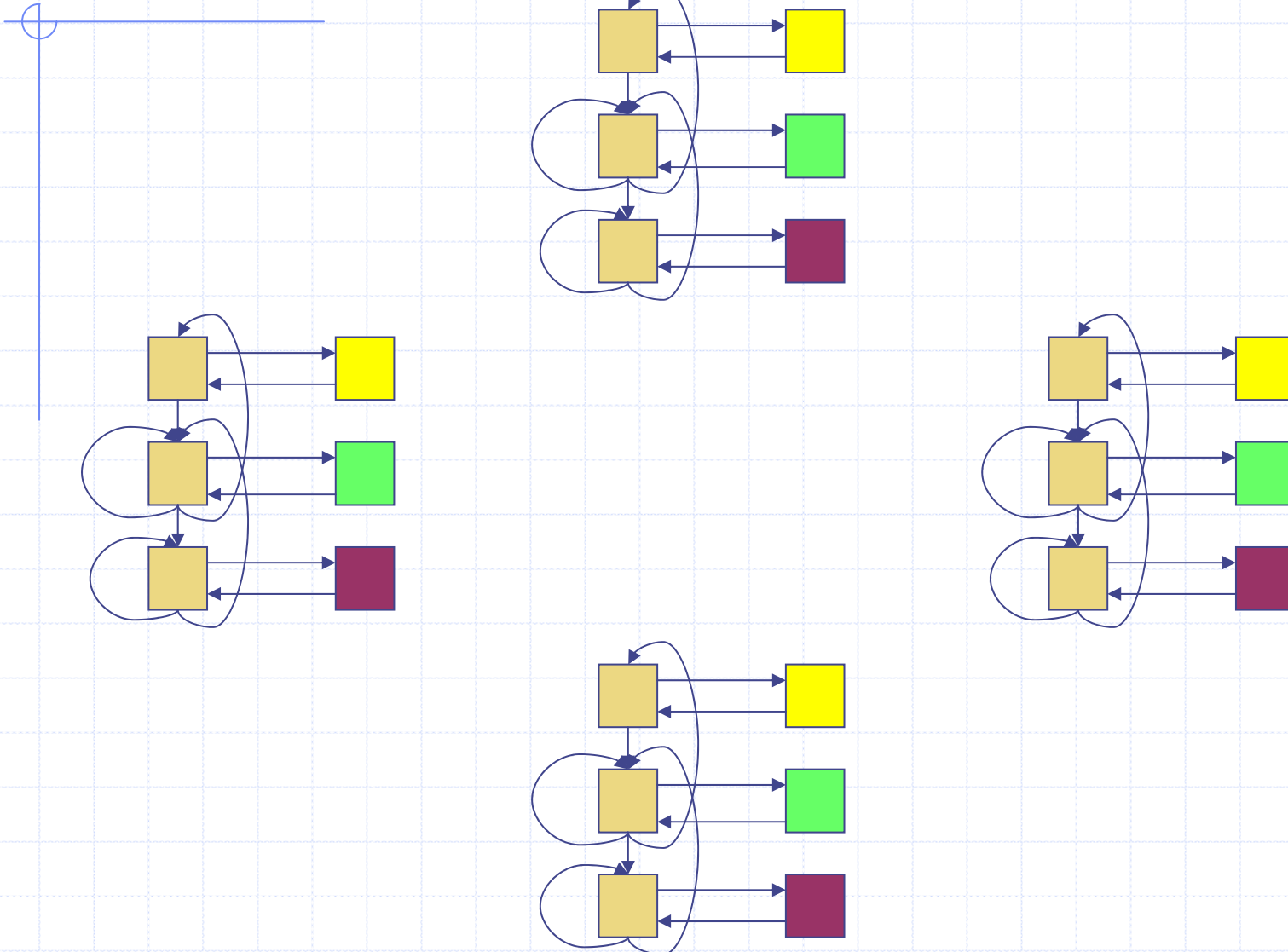
Summary of Results



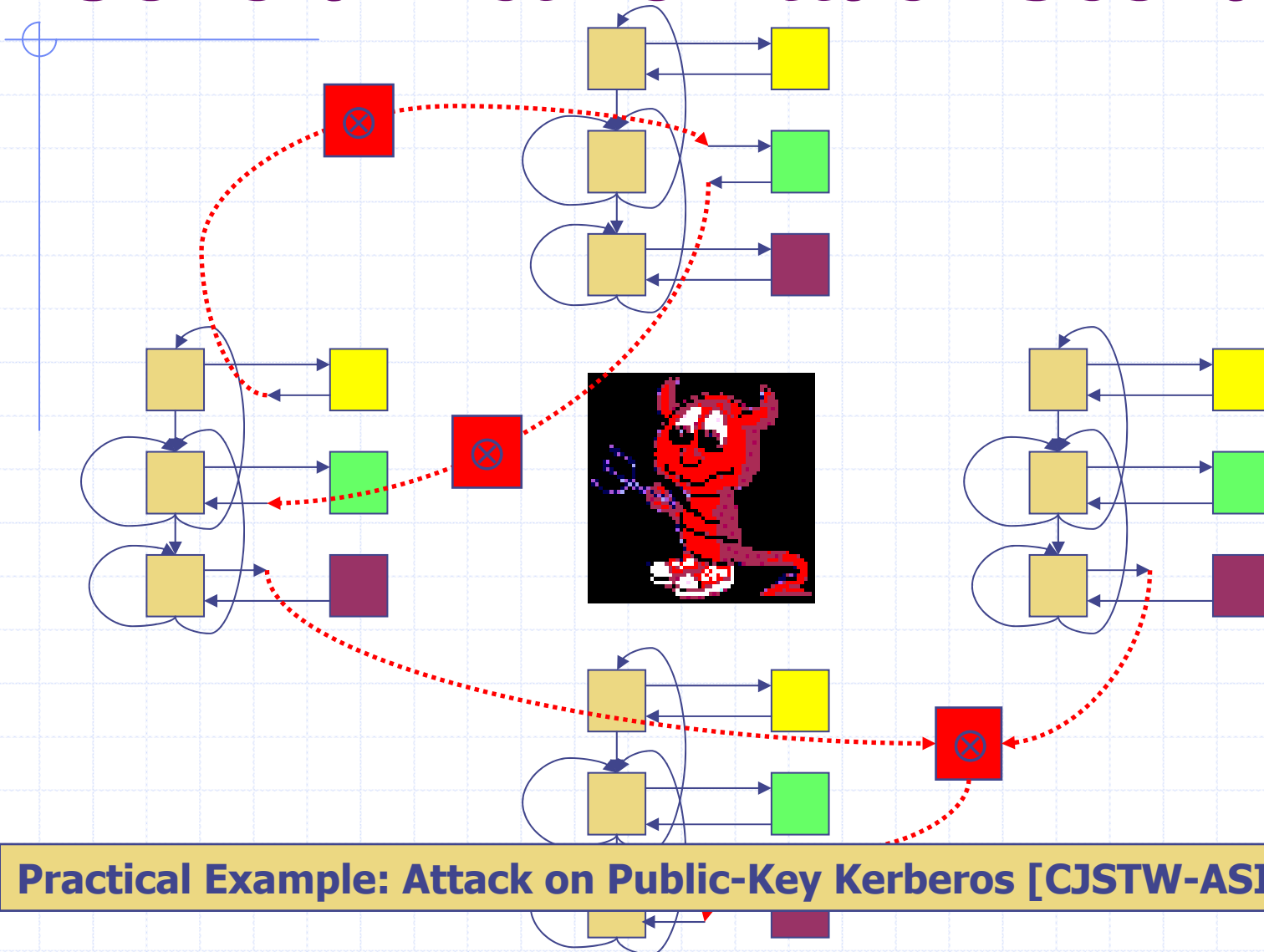
Questions?



General Active Attack Scenario



General Active Attack Scenario



Some Results

Property	Security Req.	Language
Secrecy: Indist for level-1	IND-CCA	Secret not used as a key
Secrecy: GoodKey for level-1	IND-CCA	Secret used as a symmetric key
Secrecy: Indist for key DAGs	IND-CCA	Secret not used as a key
Secrecy: GoodKey for key DAGs	IND-CCA	Secret used as a symmetric key.
Authentication for key DAGs	IND-CPA+INT-CTXT	Auth of msg encrypted with the secret.
Secrecy: GoodKey for DH Key	IND-CPA+DDH	Decrypted msgs not sent out.

Related Works

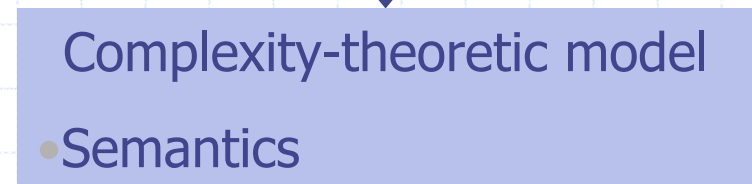
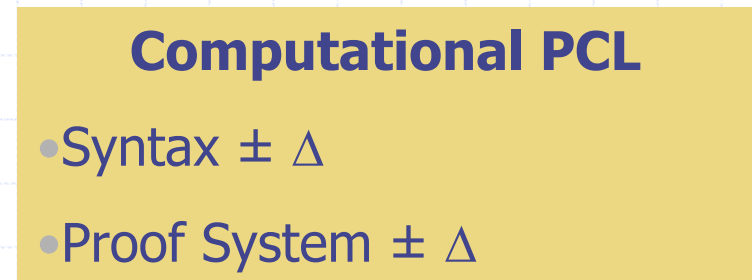
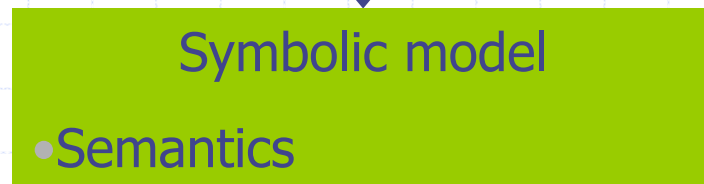
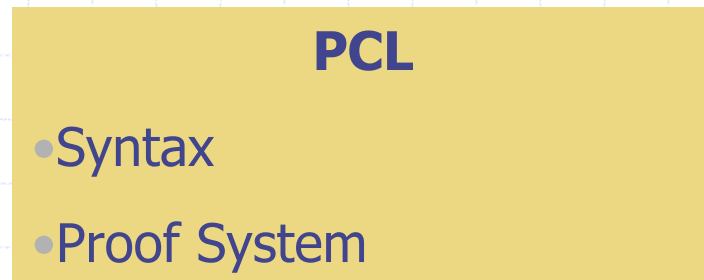
◆ Correspondence Theorems

- Micciancio-Warinschi-TCC04
- Cortier-Warinschi-ESOP05

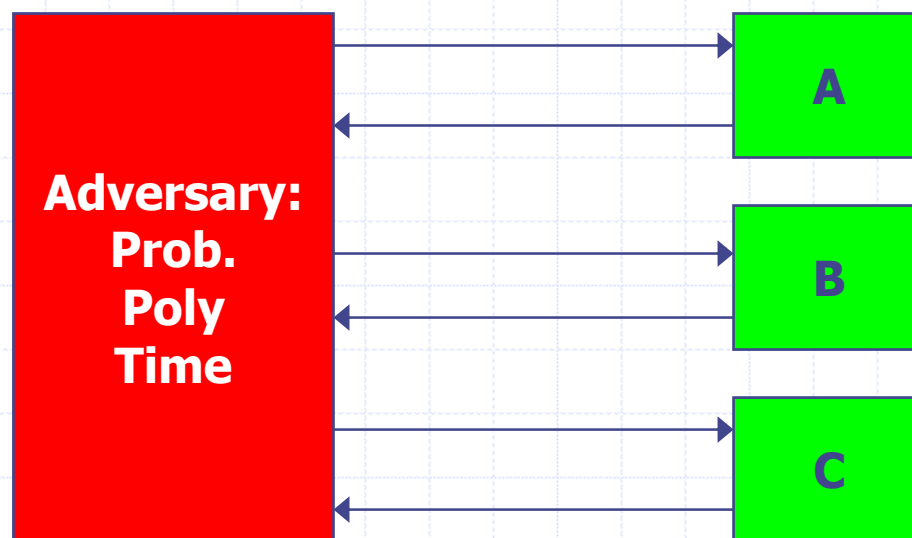
◆ Key usability and Diffie-Hellman

- Datta-Derek-Mitchell-Warinschi-CSFW06

Evolution



Execution Model



◆ Result

- Set of *computational traces*

Logic: Syntax

◆ Modal operator

- $\theta [\textit{actions}]_p \phi$

◆ Predicates in ϕ

■ Actions

- ◆ Send(X,m)
- ◆ Receive(X,m)
- ◆ Verify(X,m)

■ Knowledge

- ◆ Indist(X,m), Possess(X,m)

■ Honesty

- ◆ Honest(X)

Logic: Complexity-theoretic semantics

- ◆ Given a protocol Q , adversary A
 - T set of all possible traces
 - $[[\varphi]](T)$ a subset of T that *respects* φ in a certain way

- ◆ Intuition: φ is valid when $[[\varphi]](T)$ is an overwhelming subset of T

Logic: Proof system

- ◆ Information-theoretic reasoning

$$[\text{new } n]_X (Y \neq X) \Rightarrow \text{Indist}(Y, n)$$

- ◆ Cryptographic reductions

$$\text{Verify}(X, m, Y) \wedge \text{Honest}(Y) \Rightarrow \text{Sign}(Y, m)$$

- ◆ Asymptotic calculations

$$\frac{\varphi \qquad \varphi \Rightarrow \psi}{\psi}$$

Key Indistinguishability

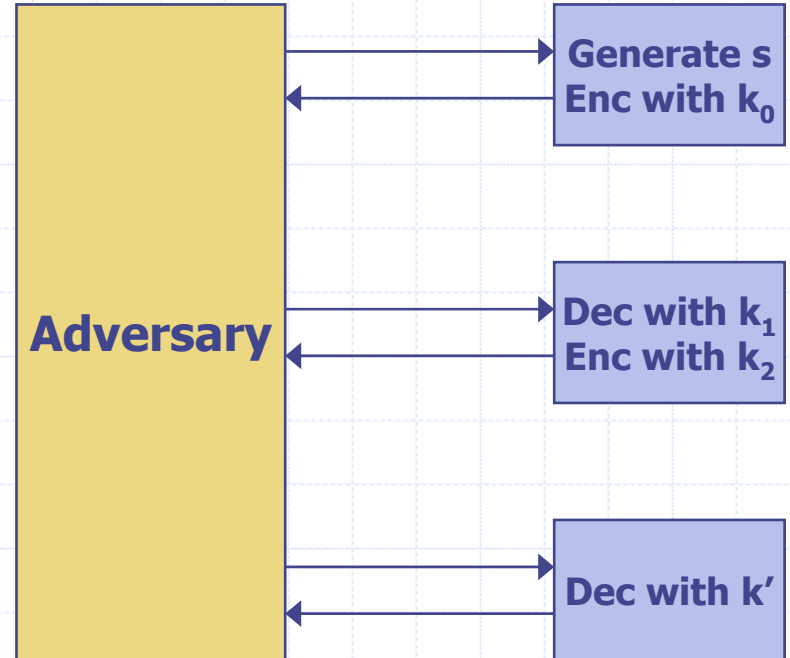
- ◆ Doesn't behave well under composition
 - Not an invariant of protocol parts that use the key
- ◆ Many "good" key exchange protocols don't achieve key indistinguishability
 - Key confirmation (STS, SSL)
 - Kerberos

Inductive Structure of "Secretive" Protocols

◆ Pick a nonce s and set of Keys $K = \{k_0, k_1, k_2\}$

◆ "Secretive" Protocol Constraints

- Terms explicitly containing s are encrypted by a key in K before sending out.
- New terms obtained through decryption by a key in K are re-encrypted by a key in K before sending out by an honest principal.



Relating “Secretive” Protocols to “Good” Keys

◆ Theorem:

If

- the protocol is “secretive”
- the nonce-generator is honest
- the key-holders are honest

Do an axiomatic proof – use the proof system

Then

- the key generated from the nonce is a “good” key

Proof is by reduction to a multi-party IND-CCA game [BBM00] – one time soundness proof

Application to Kerberos V5

- ◆ Kerberos has a staged architecture
 - First stage generates a nonce and sends it encrypted.
 - Second stage uses this nonce as a key to encrypt another nonce.
 - Third stage uses the nonce exchanged in the second stage to encrypt other terms.
- ◆ Our proof system is sufficient to prove the “GoodKey”-ness of both the nonces.
- ◆ Authentication properties are proved assuming that the encryption scheme provides ciphertext integrity.
- ◆ Modular proofs are made possible by composition theorems.
- ◆ Modular approach also facilitates analysis of Kerberos with PKINIT after basic Kerberos is done.

The Secrecy Spectrum

