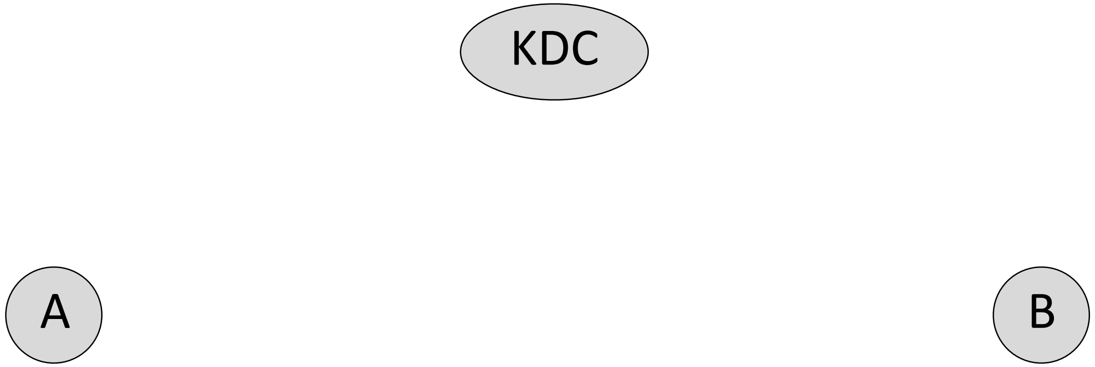
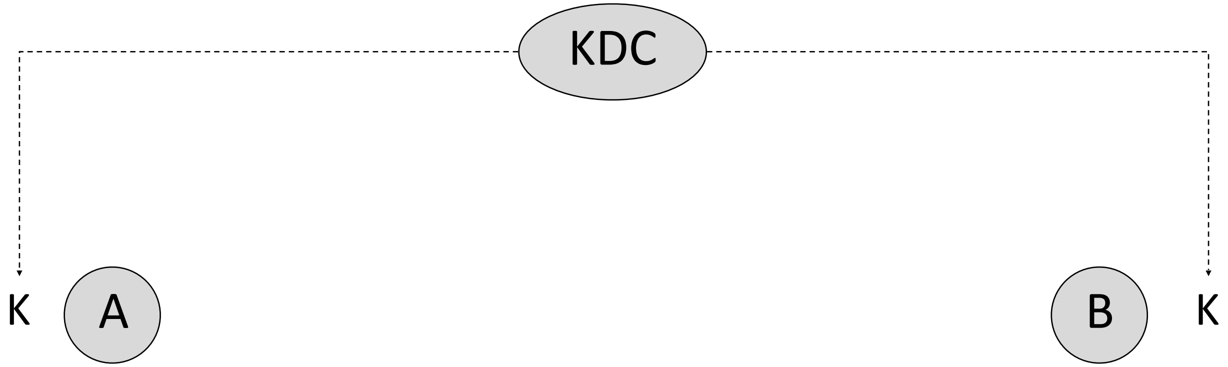
Week 7

Conventional Cryptography

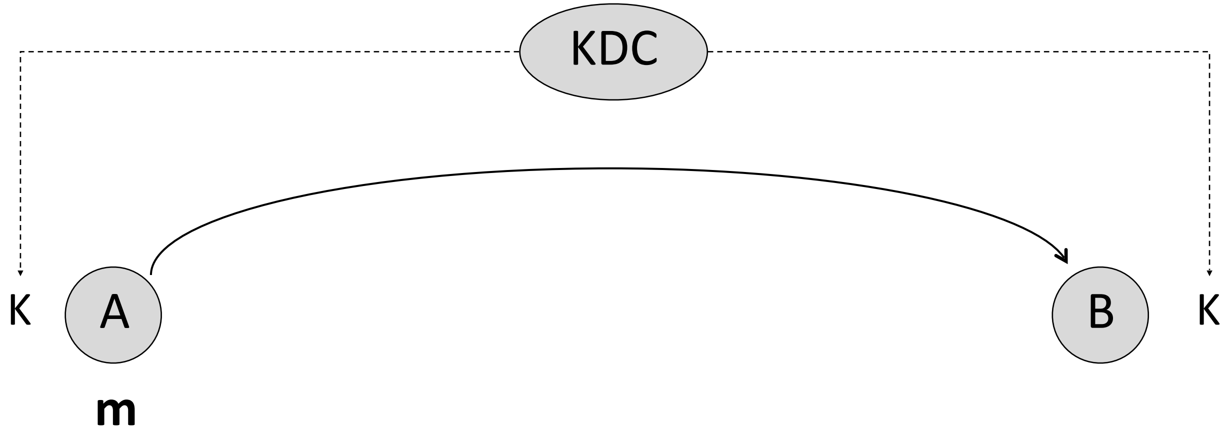


Week 7

Conventional Cryptography

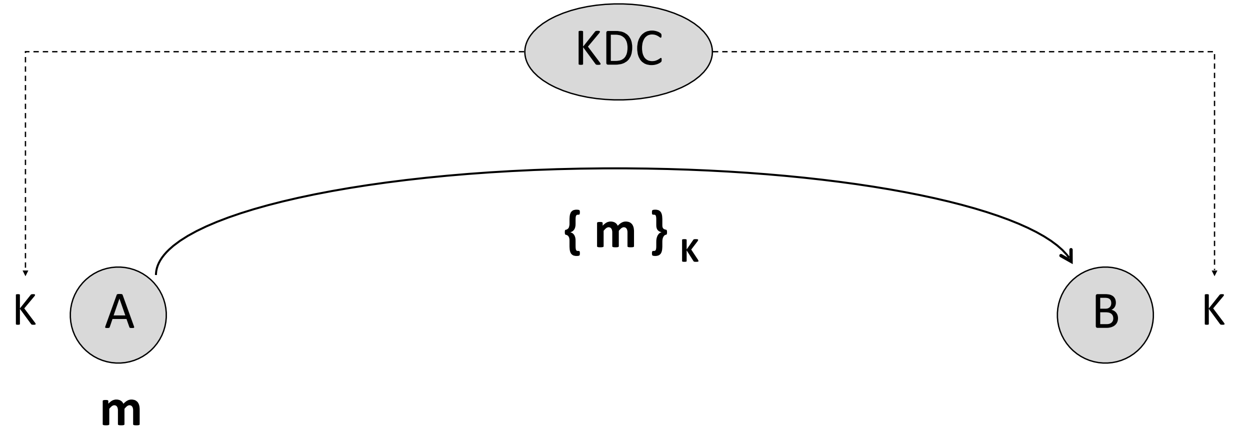


Conventional Cryptography



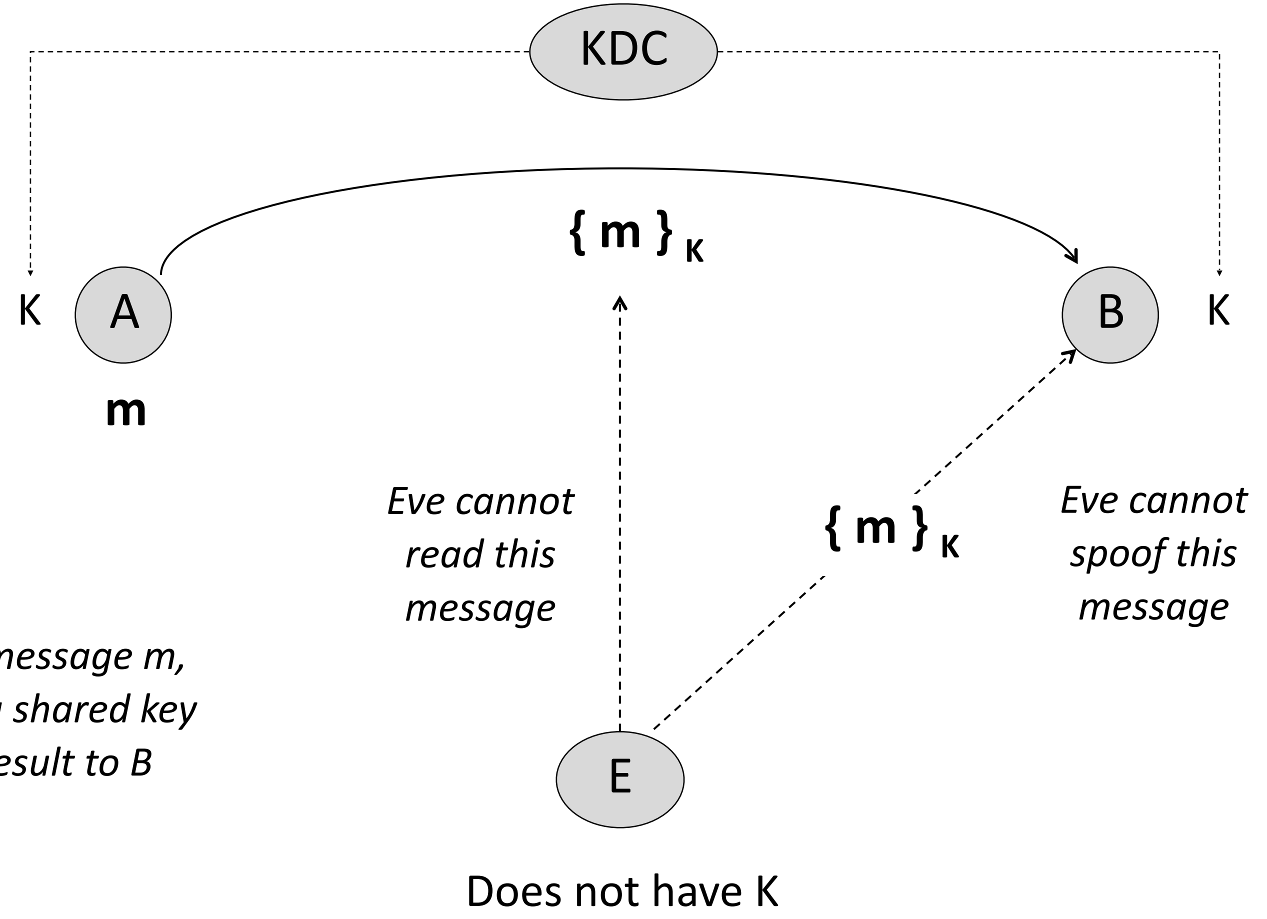
Alice creates message m . . .

Conventional Cryptography



Alice creates message m, encrypts using shared key k, and sends result to B

Conventional Cryptography

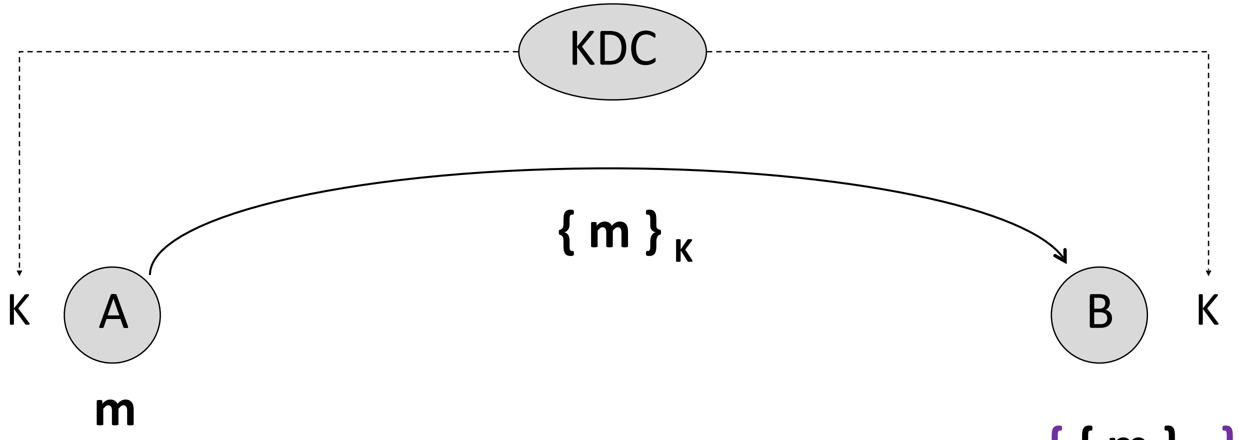


Eve cannot read this message

Eve cannot spoof this message

Alice creates message m, encrypts using shared key k, and sends result to B

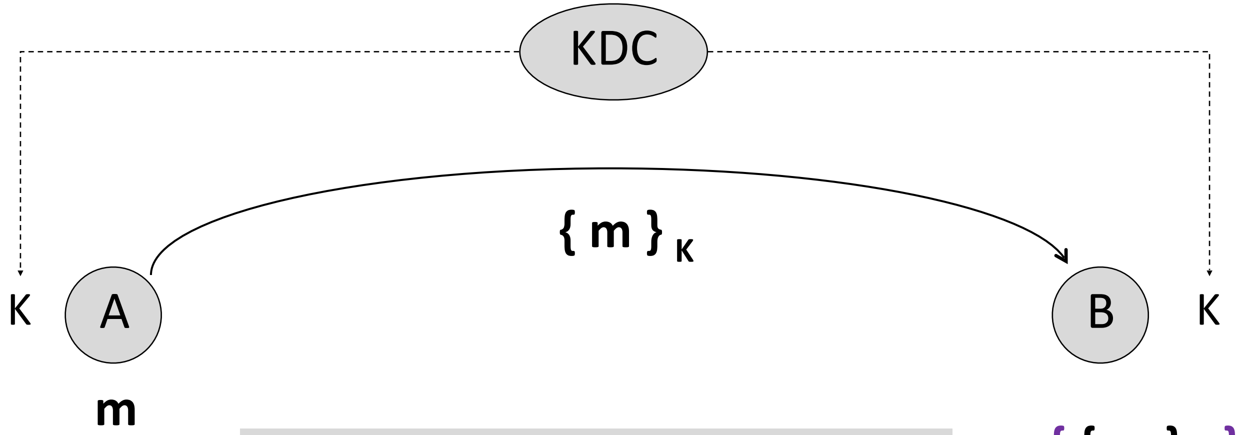
Conventional Cryptography



$$\{ \{ m \}_K \}_K = m$$

Bob receives encrypted message, and decrypts using shared key k, and obtains message m

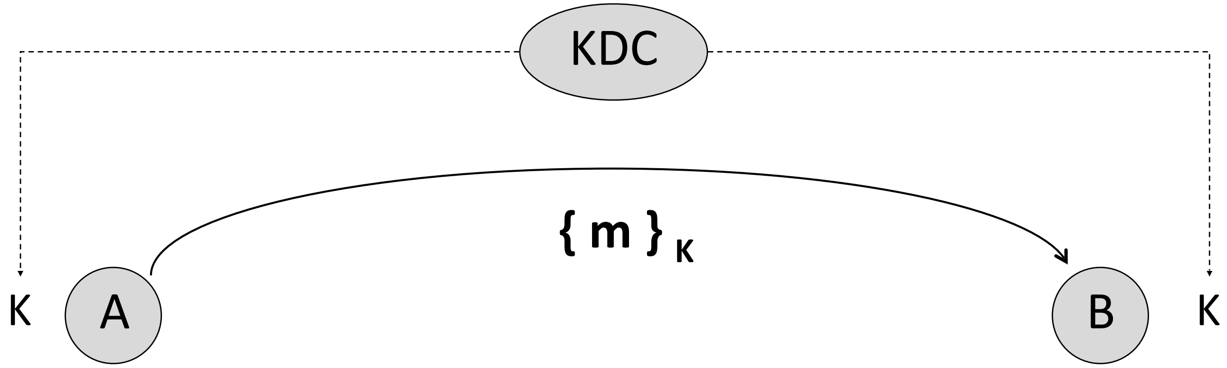
Conventional Cryptography



$\{ \{ m \}_K \}_k = m$

Secrecy Between A and B? **YES**
Authentication of A by B? **YES**

Conventional Cryptography



m

$\{\{ m \}_K\}_k = m$

Secrecy Between A and B? **YES**
Authentication of A by B? **YES**

Does this approach scale? **NO**

Public Key Cryptography Basics

Two Communicants: A and B

1. A generates pair of keys PA and SA
2. B generates pair of keys PB and SB
3. Properties:

$$\{ \{ m \}_{PA} \}_{SA} = m$$

$$\{ \{ m \}_{SA} \}_{PA} = m$$

$$\{ \{ m \}_{PA} \}_{X} = m \quad \Rightarrow \quad (X = SA)$$

$$\{ \{ m \}_{SA} \}_{X} = m \quad \Rightarrow \quad (X = PA)$$

Concept proposed by Whit Diffie and Marty Hellman, Stanford and Ralph Merkle, UC Berkeley – circa 1976

Requirements:

(i) Keep SA, SB secret to A, B

(ii) Make PA, PB public to all

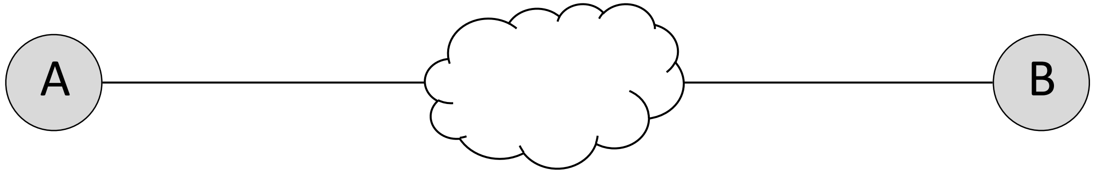
(iii) No KDC required to generate keys

*"**<u>Address Scaling Issue</u>**"*

Understanding Public Key Technology

“Assume A is a client”

“Assume B is a server”

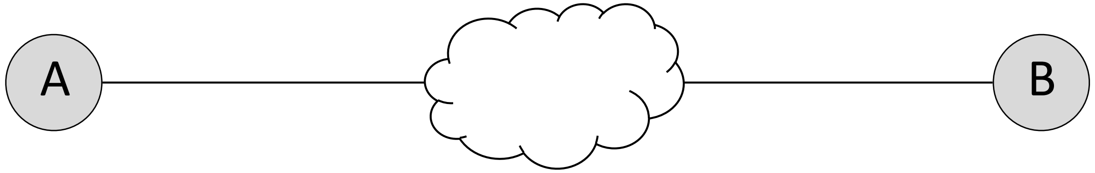


Understanding Public Key Technology

No Key Distribution Center (KDC) Required

“Assume A is a client”

“Assume B is a server”



User A Locally Generates Key Pair:

PA: Public Key of A
SA: Secret Key of A

User B Locally Generates Key Pair:

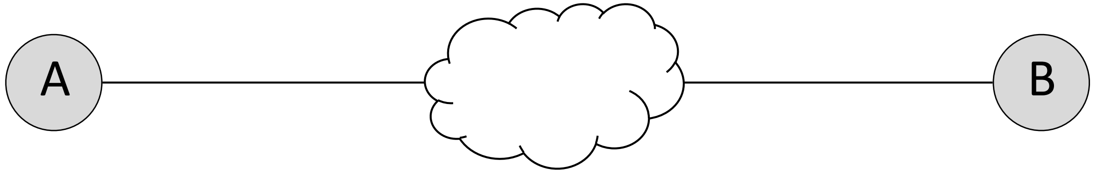
PB: Public Key of B
SB: Secret Key of B

Understanding Public Key Technology

No Key Distribution Center (KDC) Required

“Assume A is a client”

“Assume B is a server”



User A Locally Generates Key Pair:

PA: Public Key of A
SA: Secret Key of A

User B Locally Generates Key Pair:

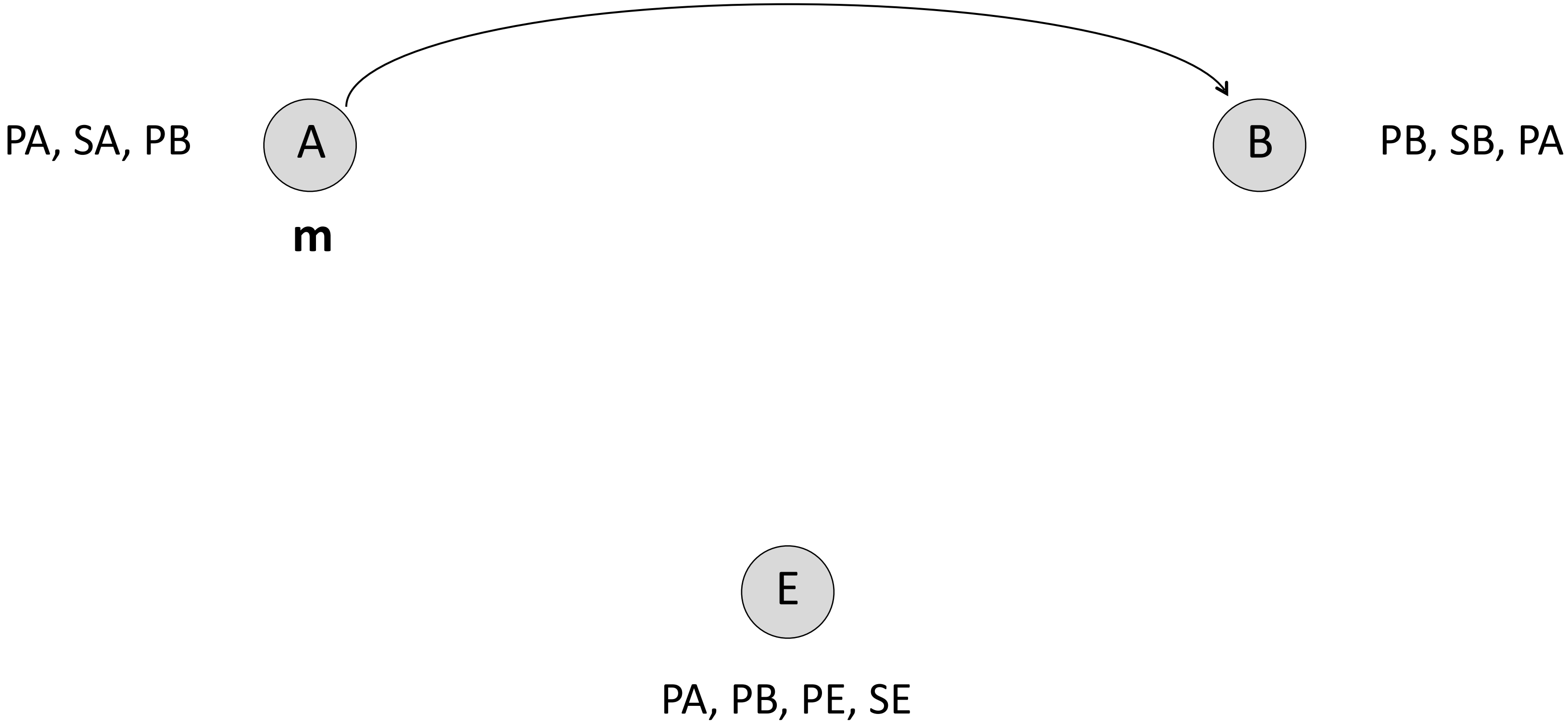
PB: Public Key of B
SB: Secret Key of B

Common Key Generation Algorithm Required (e.g., RSA)

Public Key Infrastructure (PKI)

Sending a Secret Message

Alice creates message m . . .



PA, SA, PB

A

m

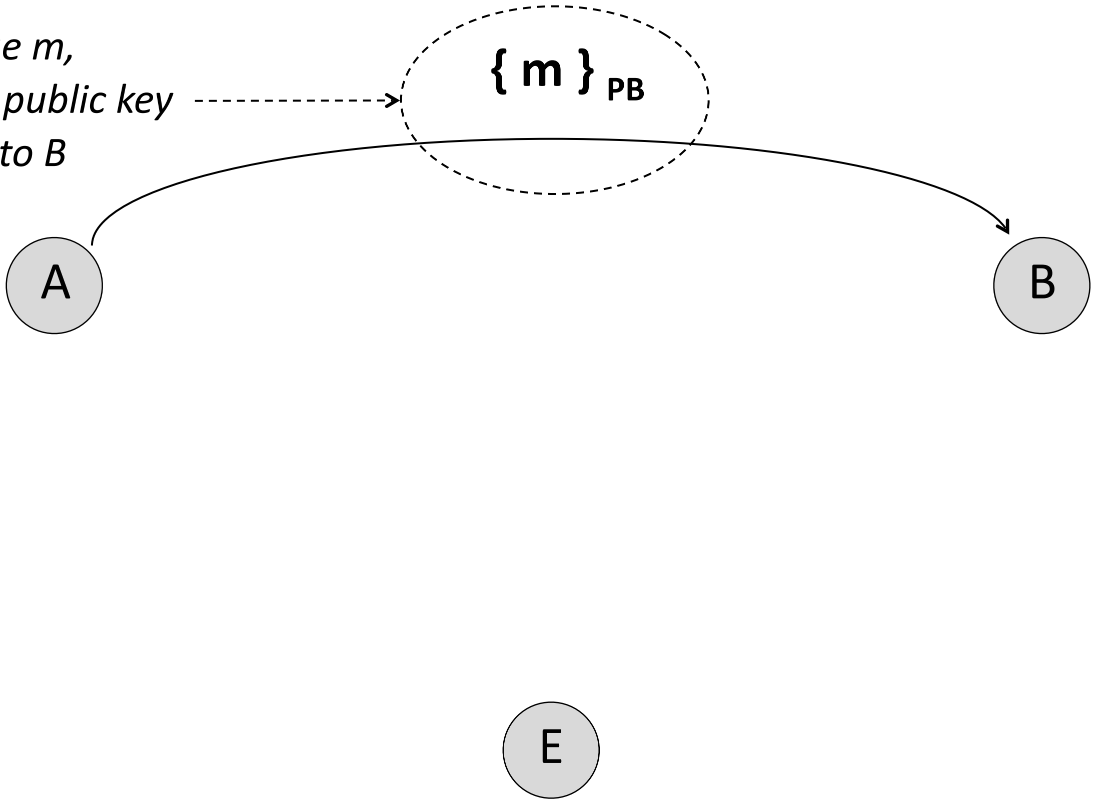
B

PB, SB, PA

E

PA, PB, PE, SE

Sending a Secret Message



Alice creates message m, encrypts using Bob's public key PB, and sends result to B

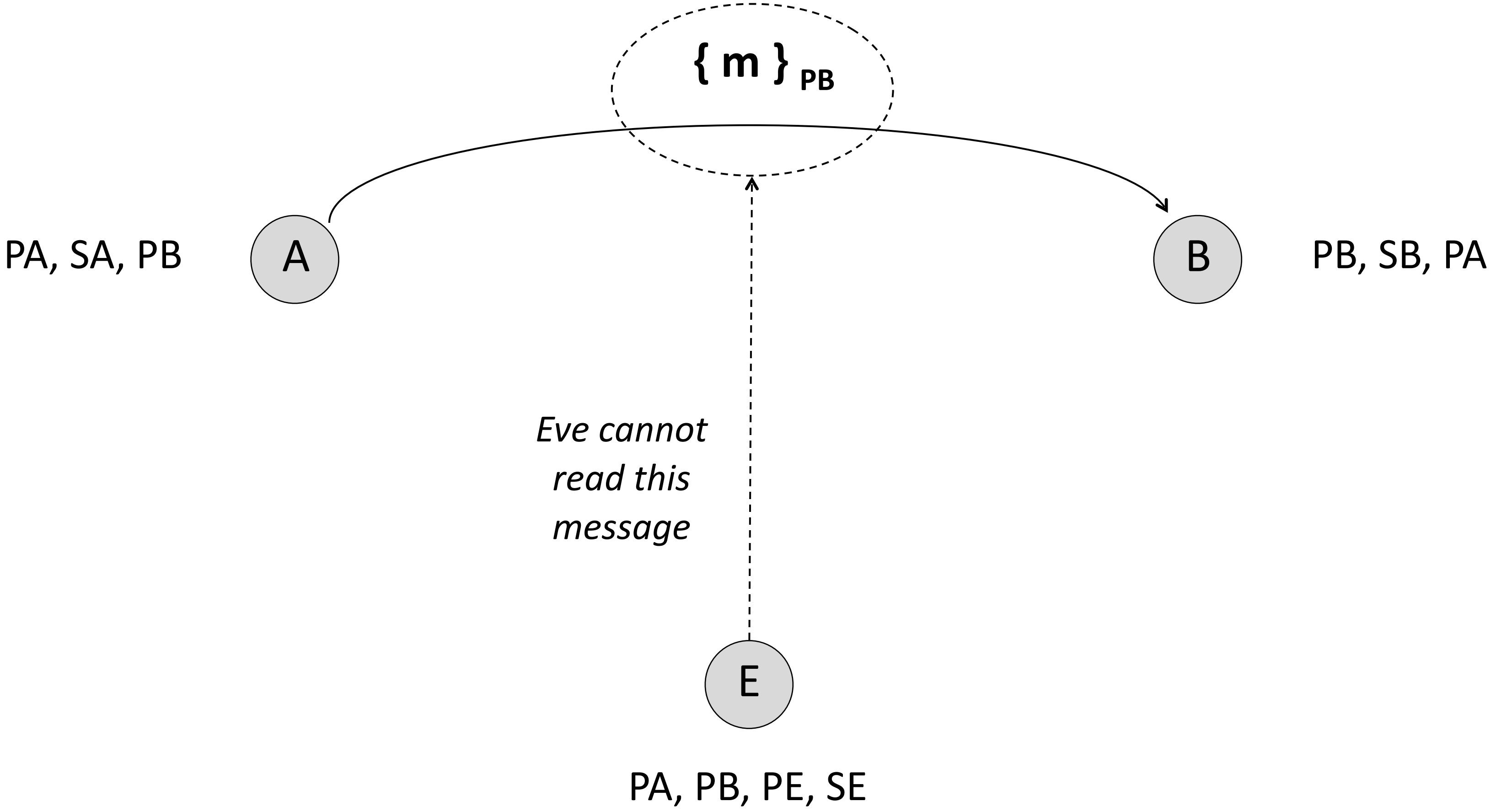
$\{ m \}_{PB}$

PA, SA, PB

PB, SB, PA

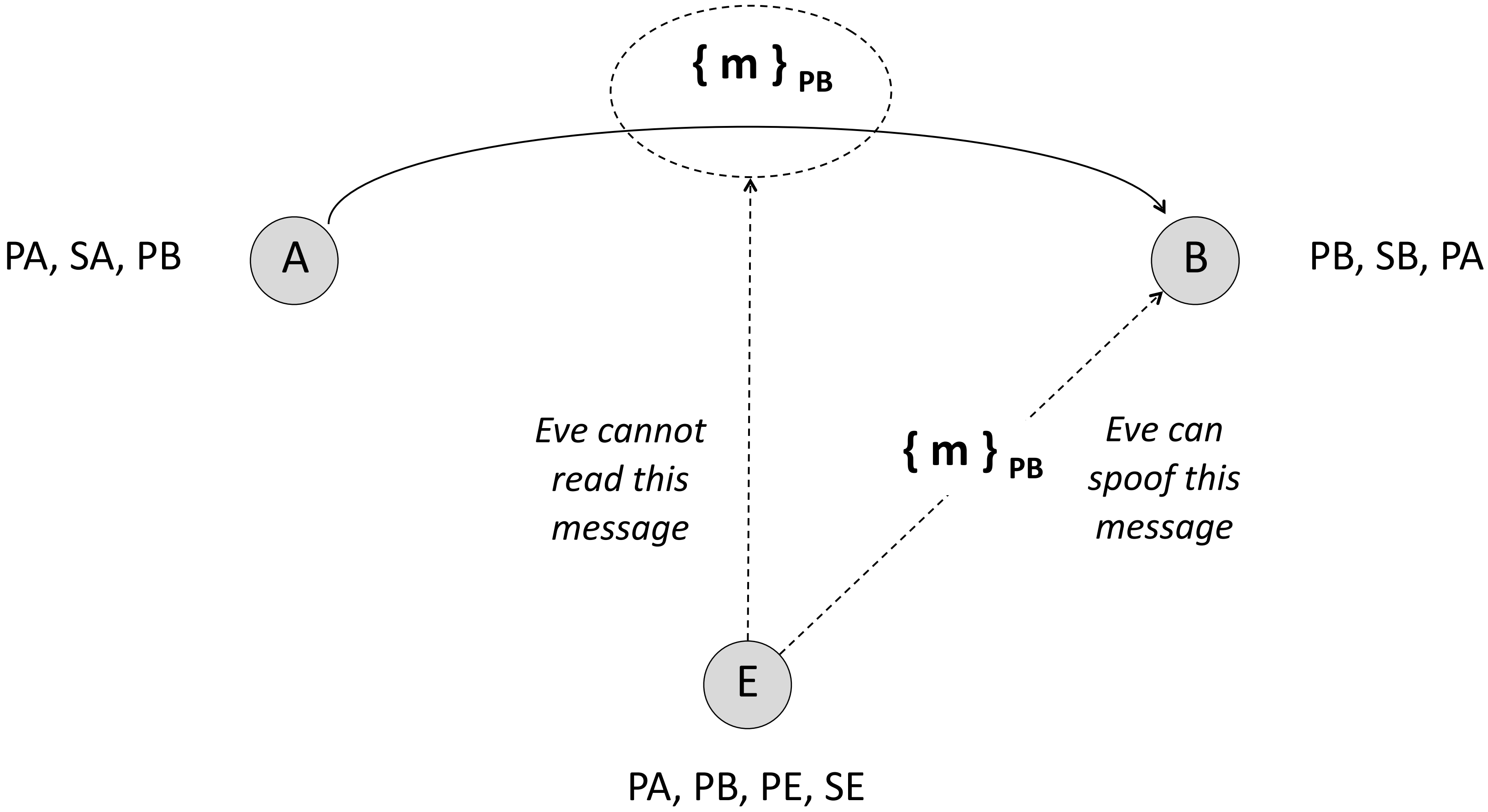
PA, PB, PE, SE

Sending a Secret Message

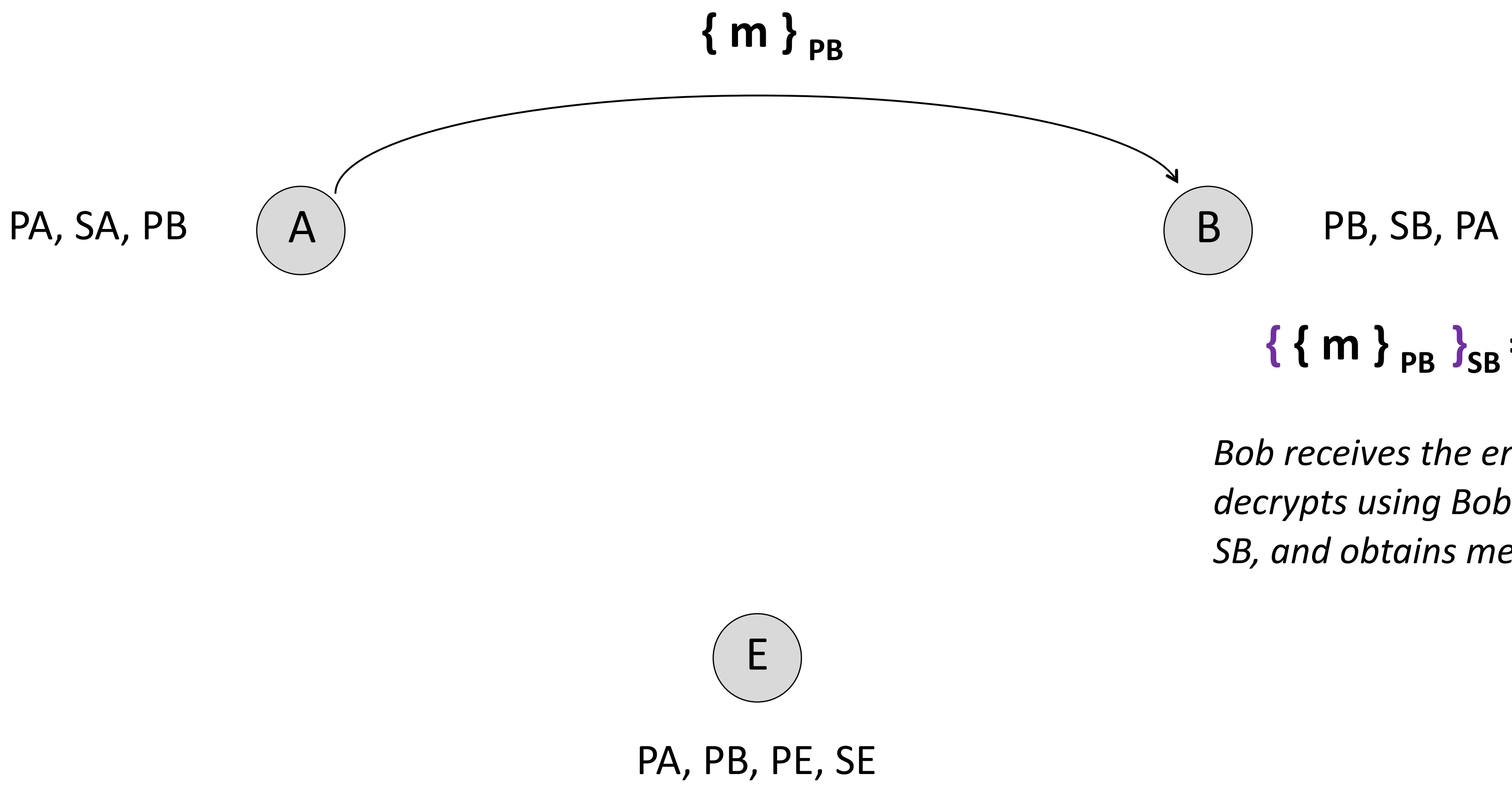


Week 7

Sending a Secret Message



Sending a Secret Message

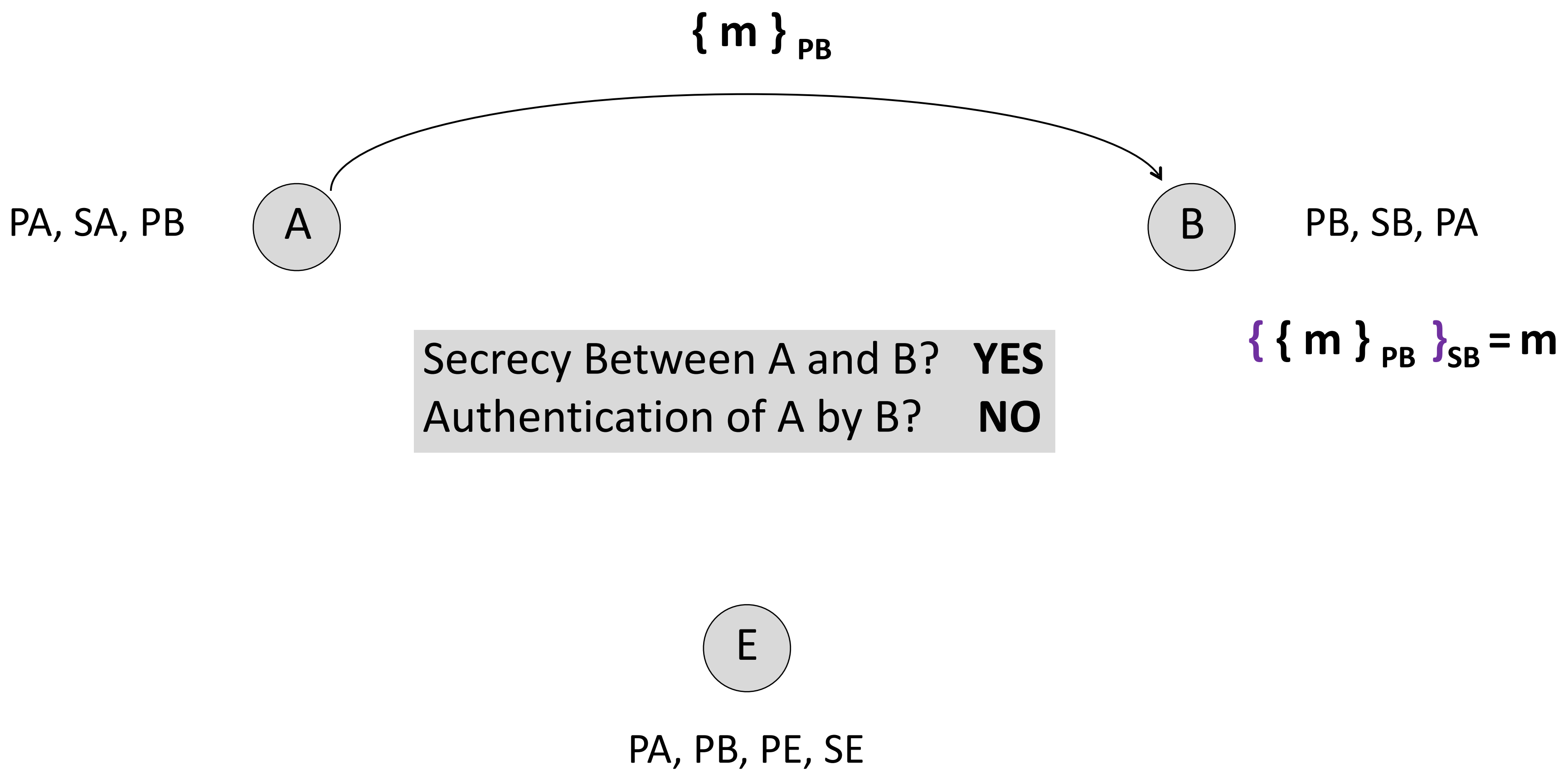


$$\{ \{ m \}_{PB} \}_{SB} = m$$

Bob receives the encrypted message, decrypts using Bob's secret key SB, and obtains message m

Week 7

Sending a Secret Message



$\{ m \}_{PB}$

PA, SA, PB

PB, SB, PA

$\{ \{ m \}_{PB} \}_{SB} = m$

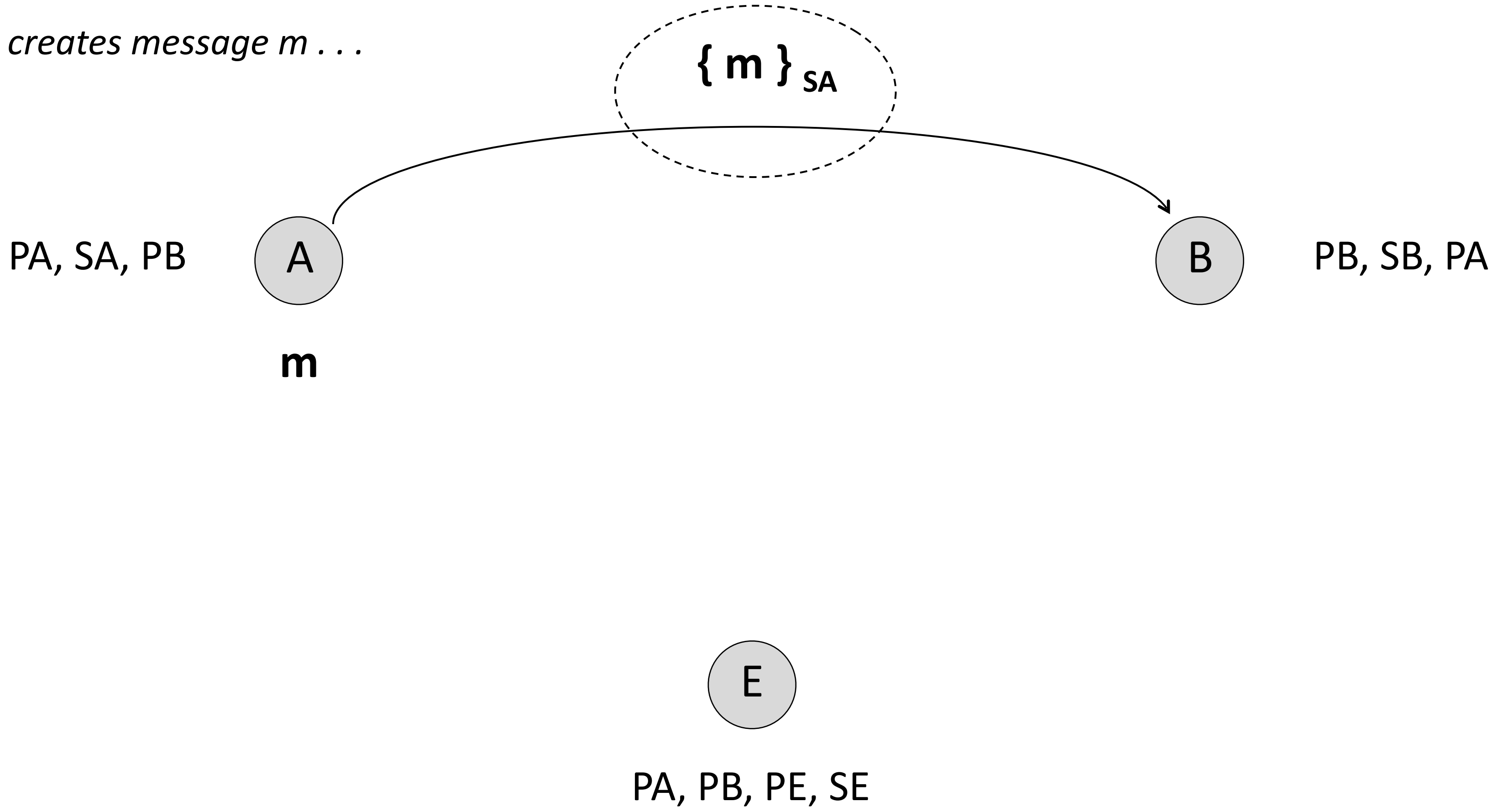
Secrecy Between A and B? **YES**

Authentication of A by B? **NO**

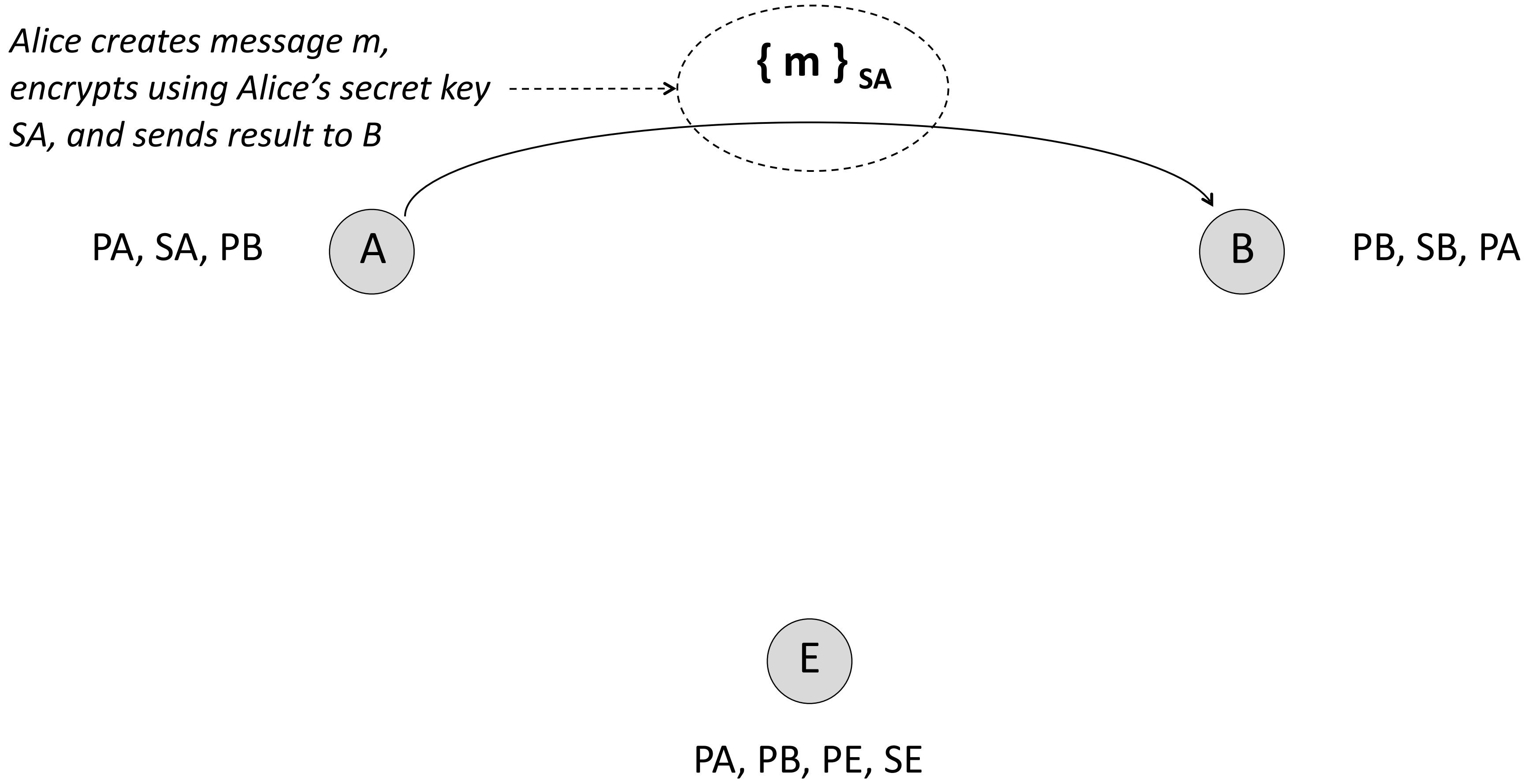
PA, PB, PE, SE

Sending a Signed Message

Alice creates message m . . .



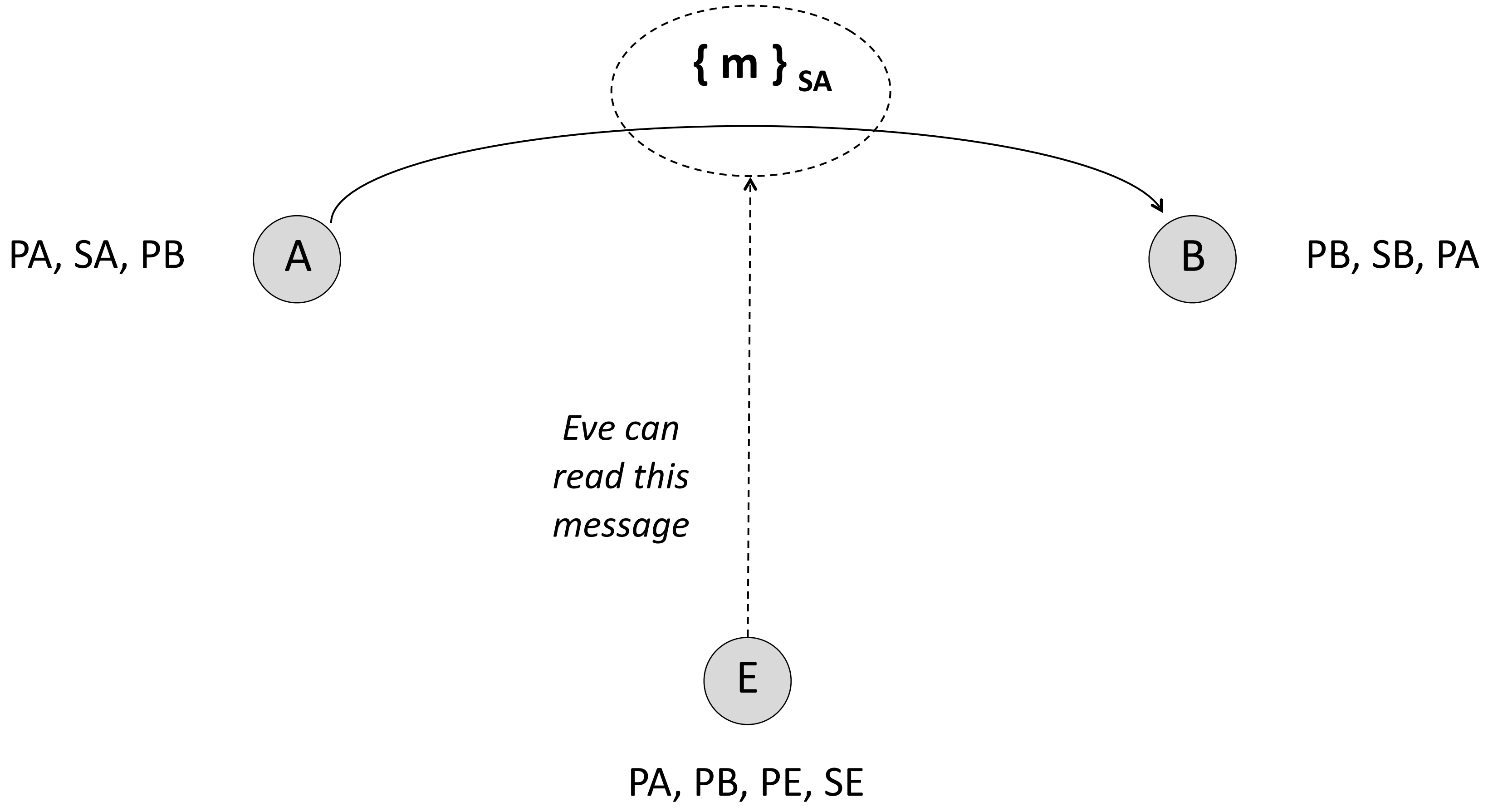
Sending a Signed Message



Alice creates message m, encrypts using Alice's secret key SA, and sends result to B

$\{ m \}_{SA}$

Sending a Signed Message



$\{ \mathbf{m} \}_{\mathbf{SA}}$

PA, SA, PB

A

B

PB, SB, PA

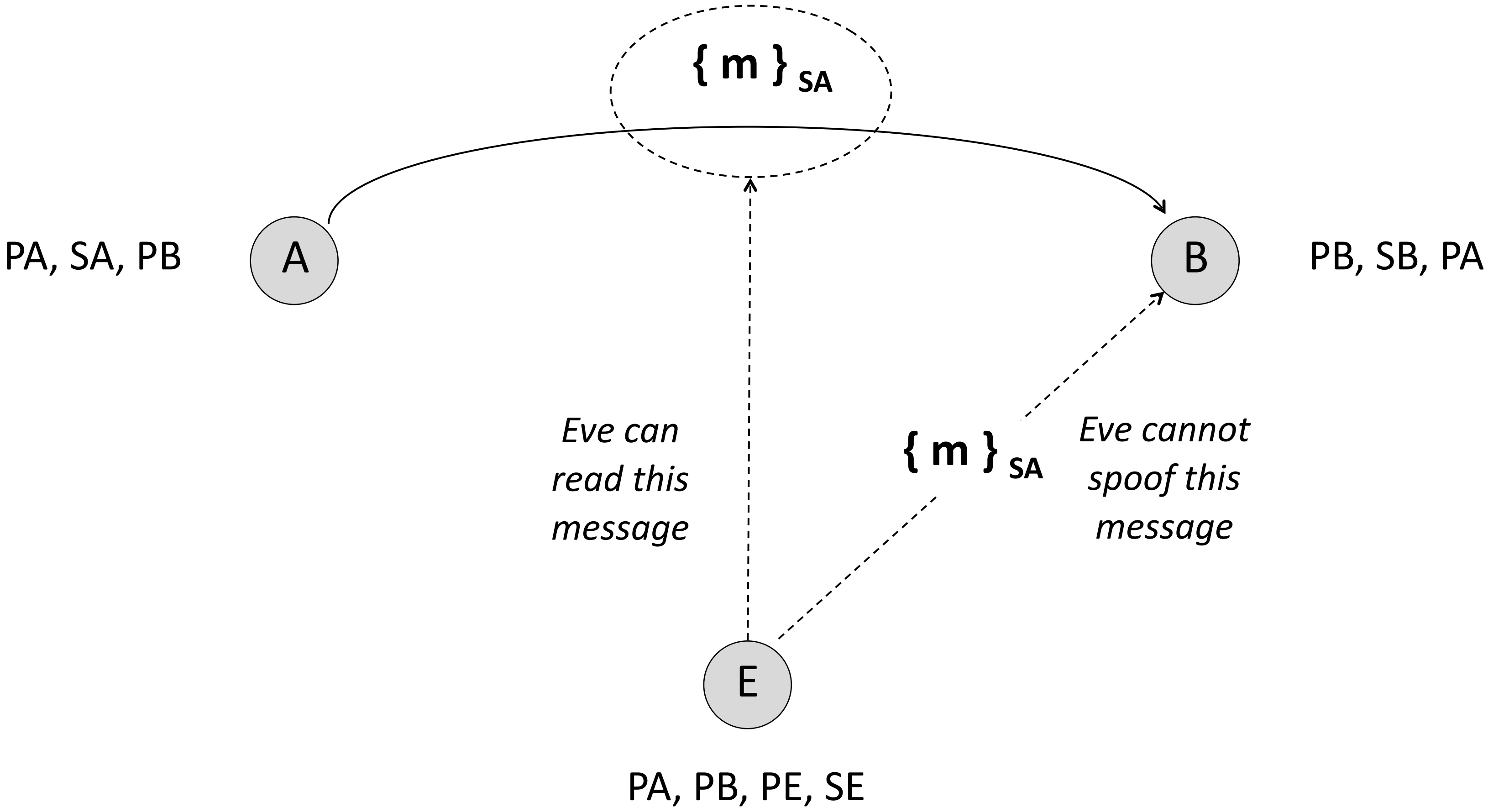
Eve can read this message

E

PA, PB, PE, SE

Week 7

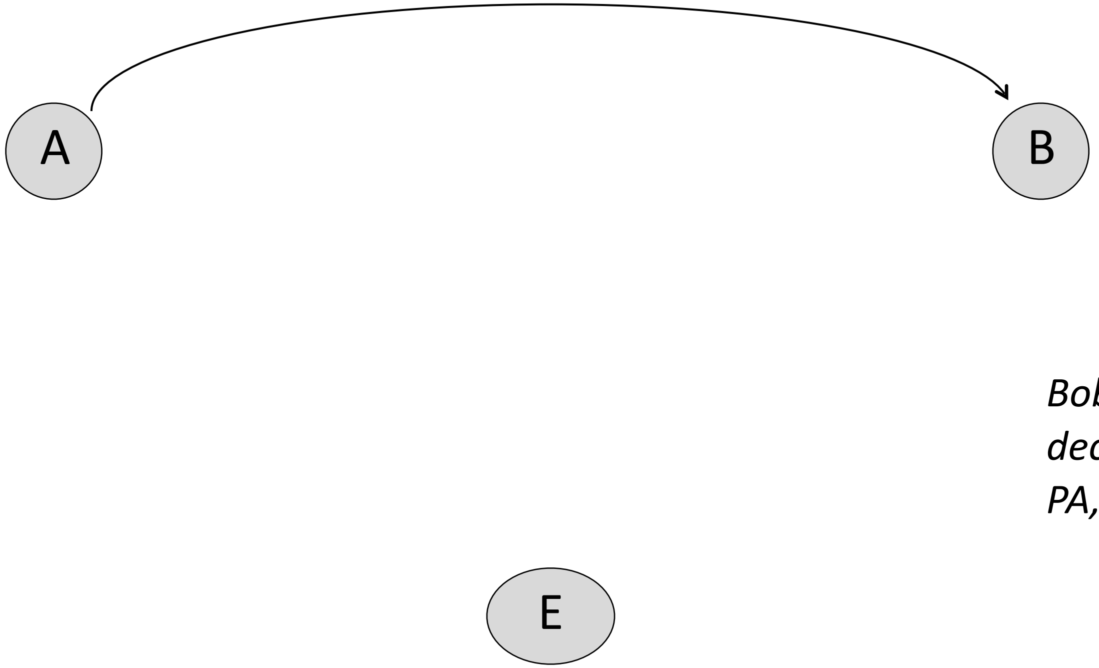
Sending a Signed Message



Sending a Signed Message

$\{ m \}_{SA}$

PA, SA, PB



PB, SB, PA

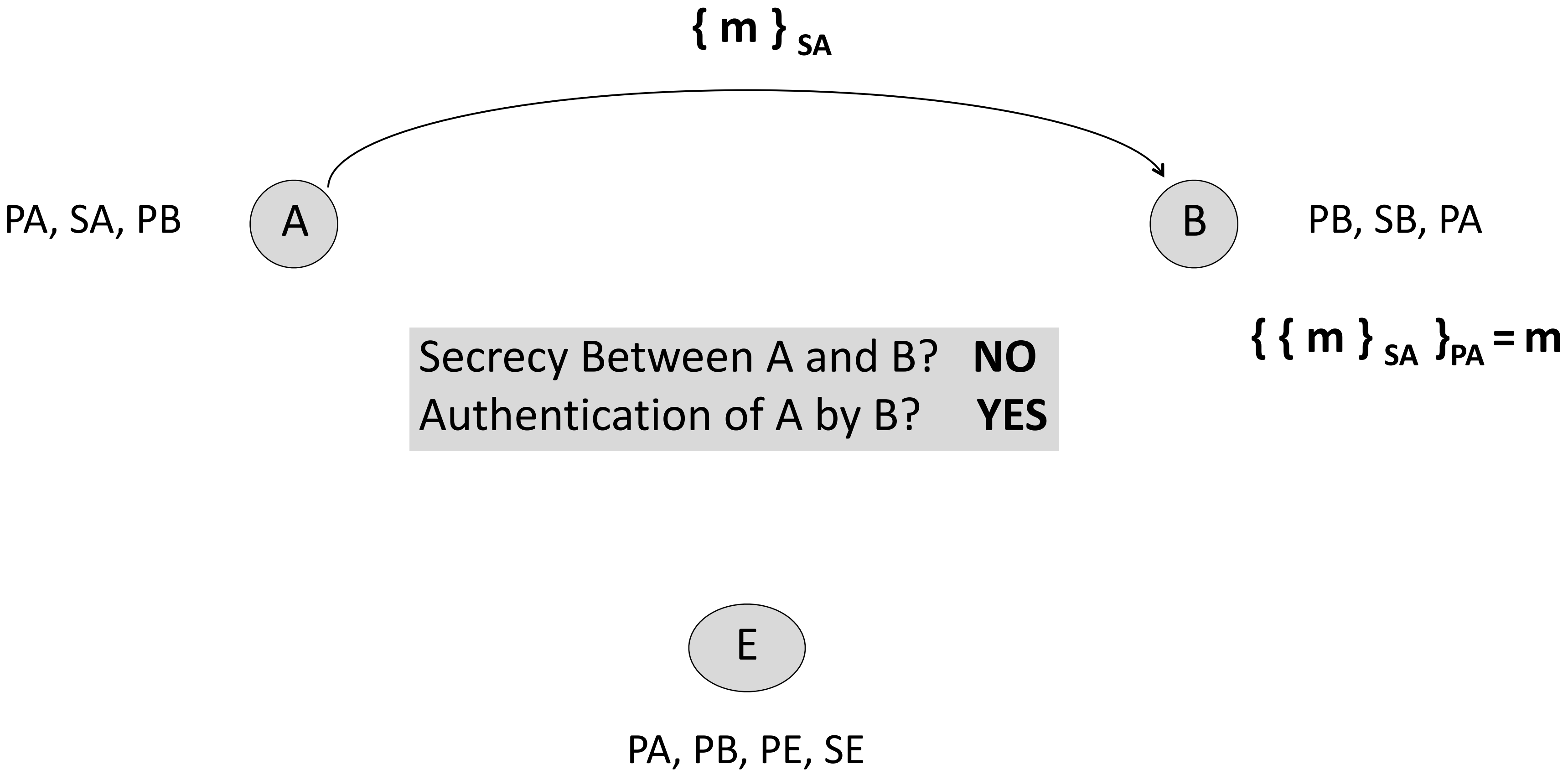
$\{ \{ m \}_{SA} \}_{PA} = m$

Bob receives the encrypted message, decrypts using Alice's public key PA, and obtains message m

PA, PB, PE, SE

Week 7

Sending a Signed Message



$\{ m \}_{SA}$

PA, SA, PB

PB, SB, PA

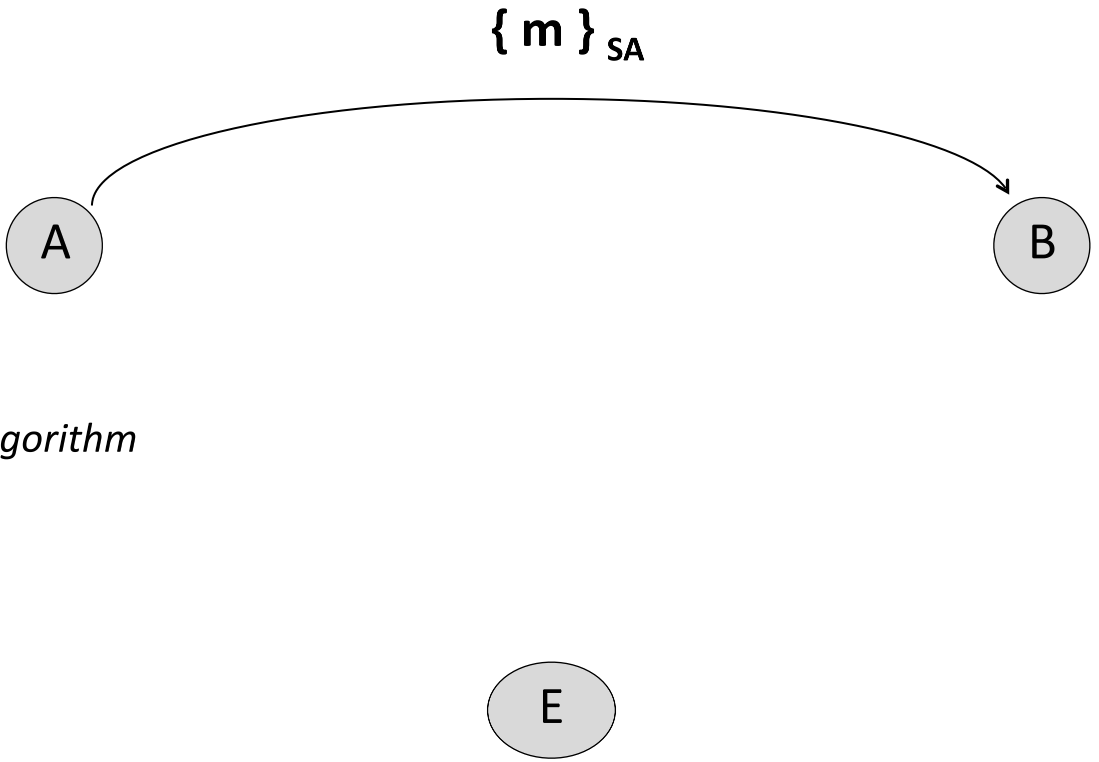
$\{ \{ m \}_{SA} \}_{PA} = m$

Secrecy Between A and B? **NO**

Authentication of A by B? **YES**

PA, PB, PE, SE

Secure Message Exchange



PA, SA, PB

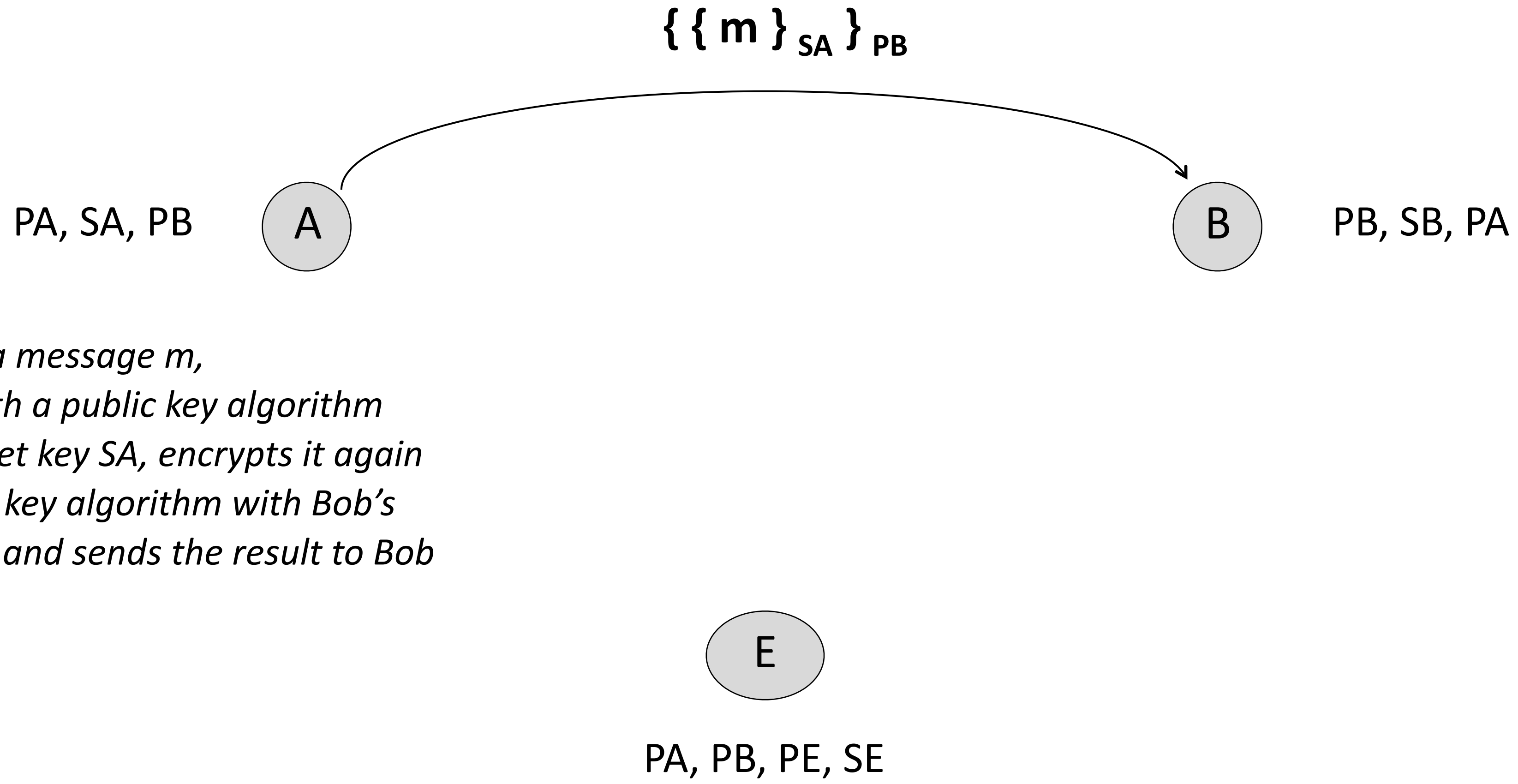
PB, SB, PA

PA, PB, PE, SE

Alice creates a message m,
encrypts it with a public key algorithm
using her secret key SA . . .

Week 7

Secure Message Exchange



$\{ \{ m \}_{SA} \}_{PB}$

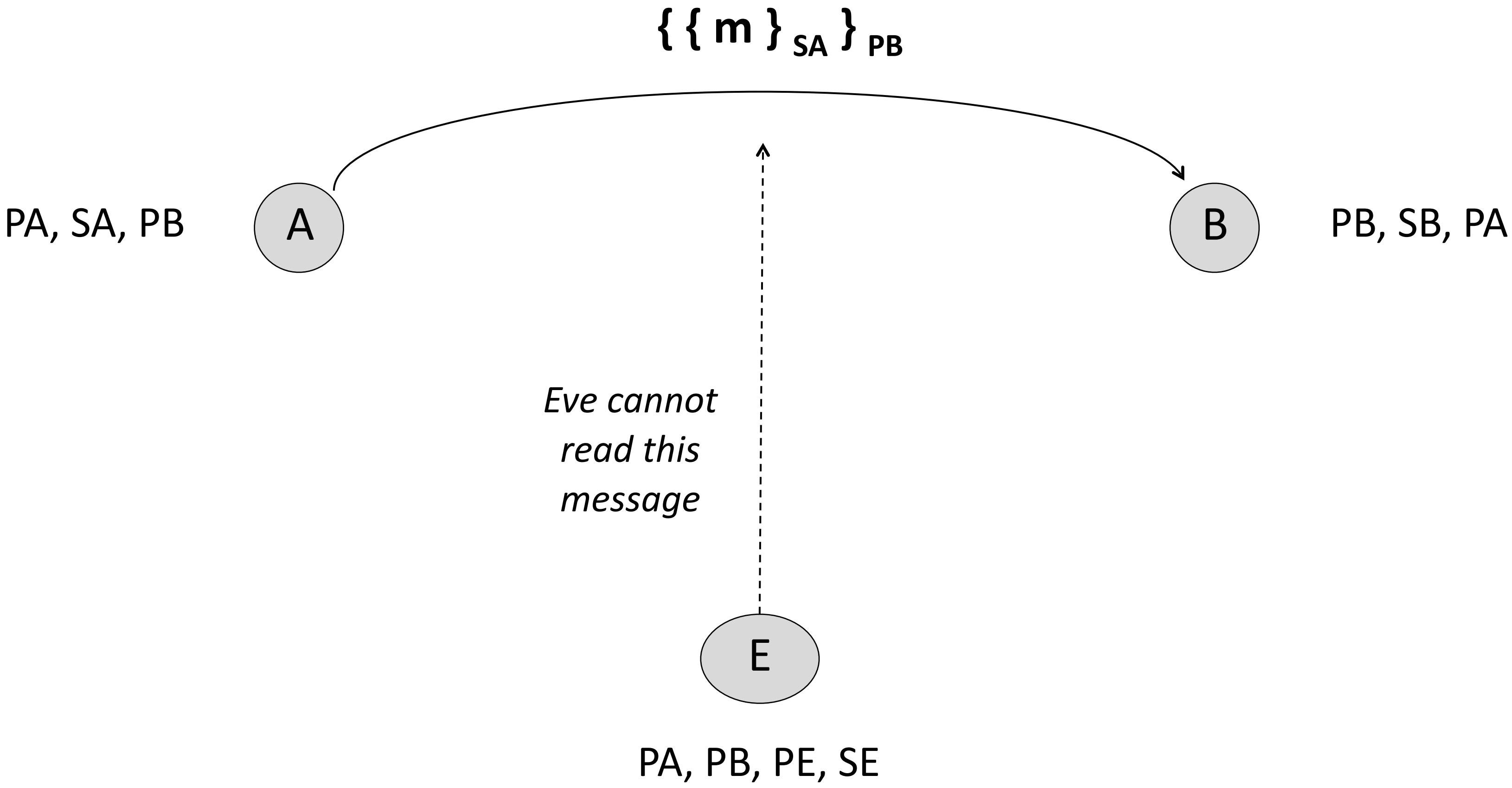
PA, SA, PB

PB, SB, PA

PA, PB, PE, SE

Alice creates a message m,
encrypts it with a public key algorithm
using her secret key SA, encrypts it again
using a public key algorithm with Bob's
public key PB, and sends the result to Bob

Secure Message Exchange



$\{ \{ m \}_{SA} \}_{PB}$

PA, SA, PB

A

B

PB, SB, PA

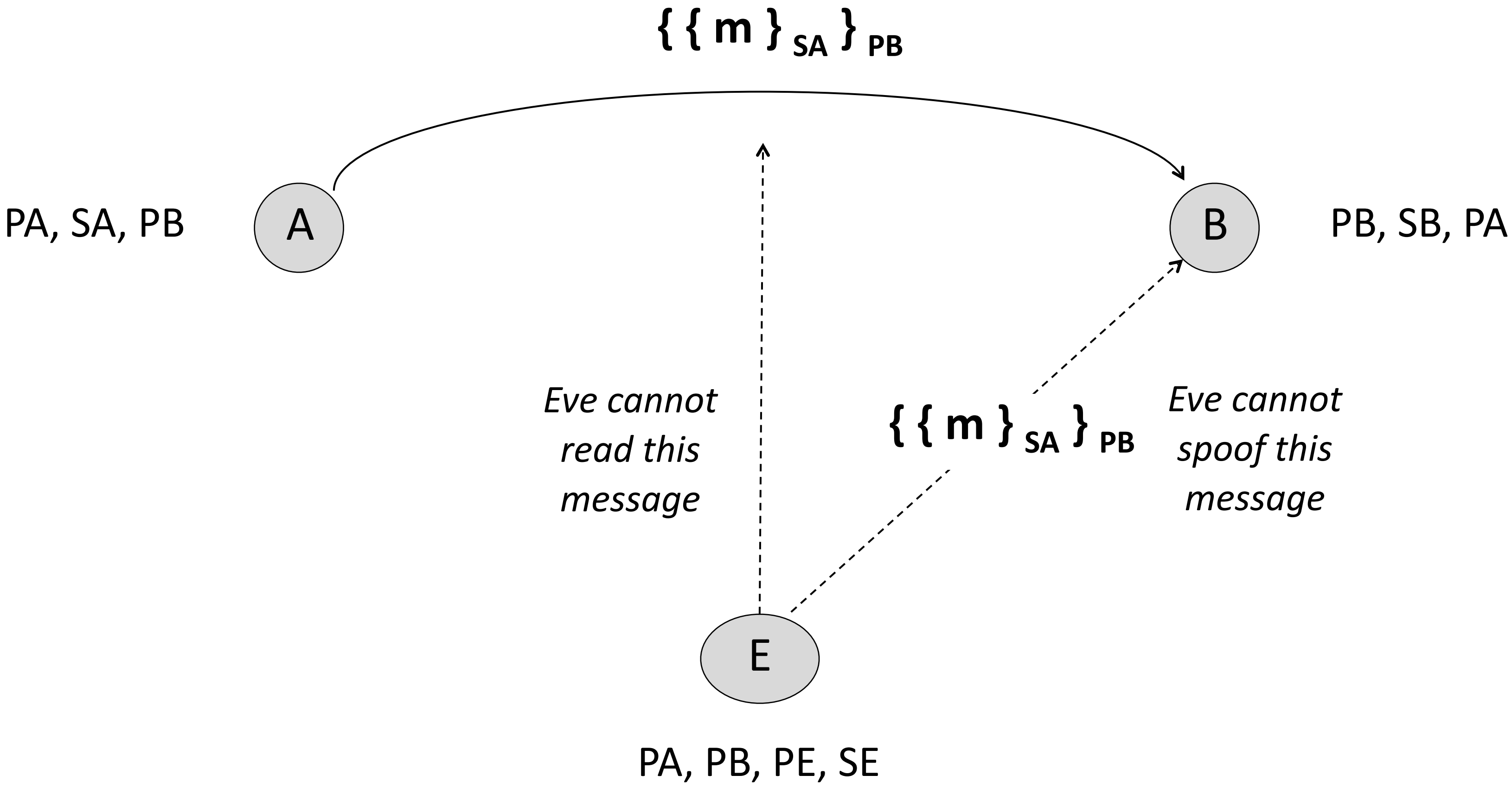
Eve cannot read this message

E

PA, PB, PE, SE

Week 7

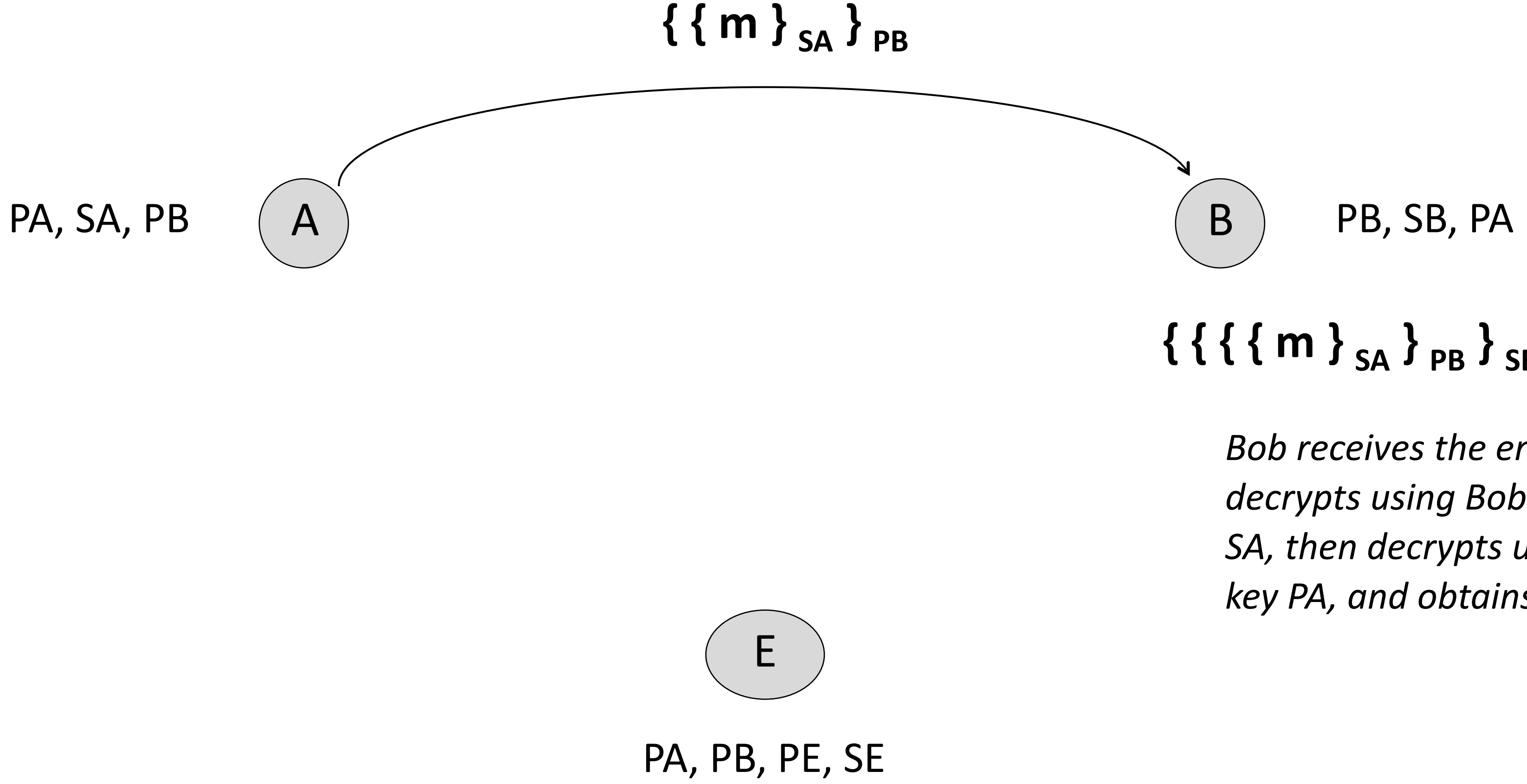
Secure Message Exchange



$\{ \{ m \}_{SA} \}_{PB}$
PA, SA, PB
A
B
PB, SB, PA
Eve cannot read this message
$\{ \{ m \}_{SA} \}_{PB}$
Eve cannot spoof this message
E
PA, PB, PE, SE

Week 7

Secure Message Exchange



$\{ \{ m \}_{SA} \}_{PB}$

PA, SA, PB

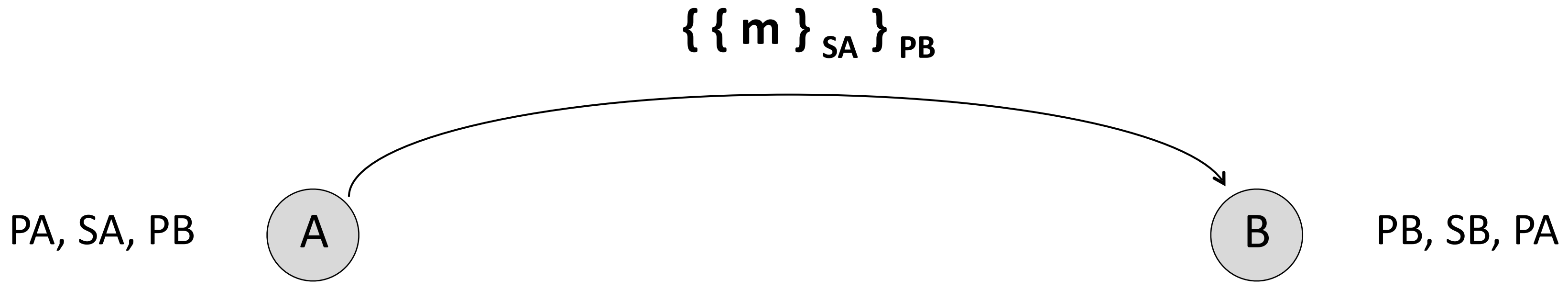
PB, SB, PA

PA, PB, PE, SE

$\{ \{ \{ \{ m \}_{SA} \}_{PB} \}_{SB} \}_{PA} = m$

Bob receives the encrypted message, decrypts using Bob's secret key SA, then decrypts using Alice's public key PA, and obtains message m

Secure Message Exchange



$\{ \{ m \}_{SA} \}_{PB}$

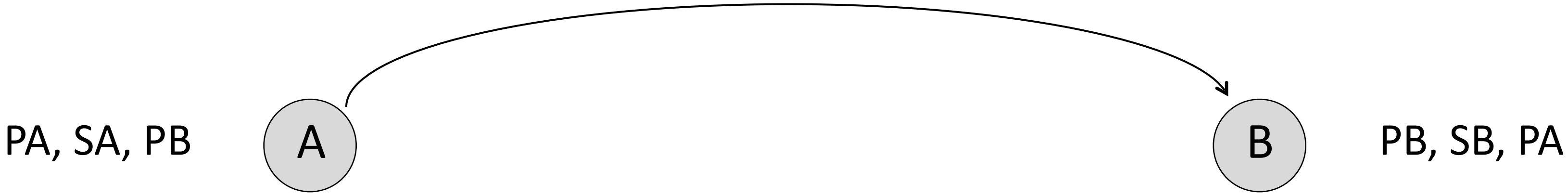
$$\{ \{ \{ \{ m \}_{SA} \}_{PB} \}_{SB} \}_{PA} = m$$

Secrecy Between A and B? **YES**

Authentication of A by B? **YES**

Secure Message Exchange

$\{ \{ m \}_{SA} \}_{PB}$



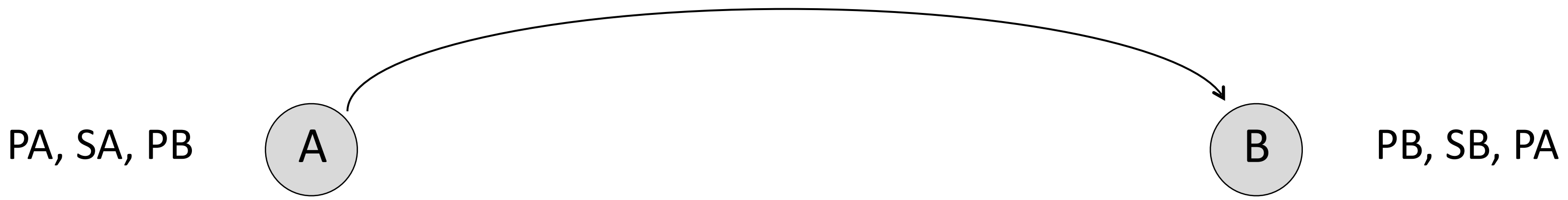
$\{ \{ \{ \{ m \}_{SA} \}_{PB} \}_{SB} \}_{PA} = m$

Secrecy Between A and B? **YES**
Authentication of A by B? **YES**

Does this approach scale? **YES**

Secure Message Exchange

$\{\{m\}_{SA}\}_{PB}$



$\{\{\{\{m\}_{SA}\}_{PB}\}_{SB}\}_{PA} = m$

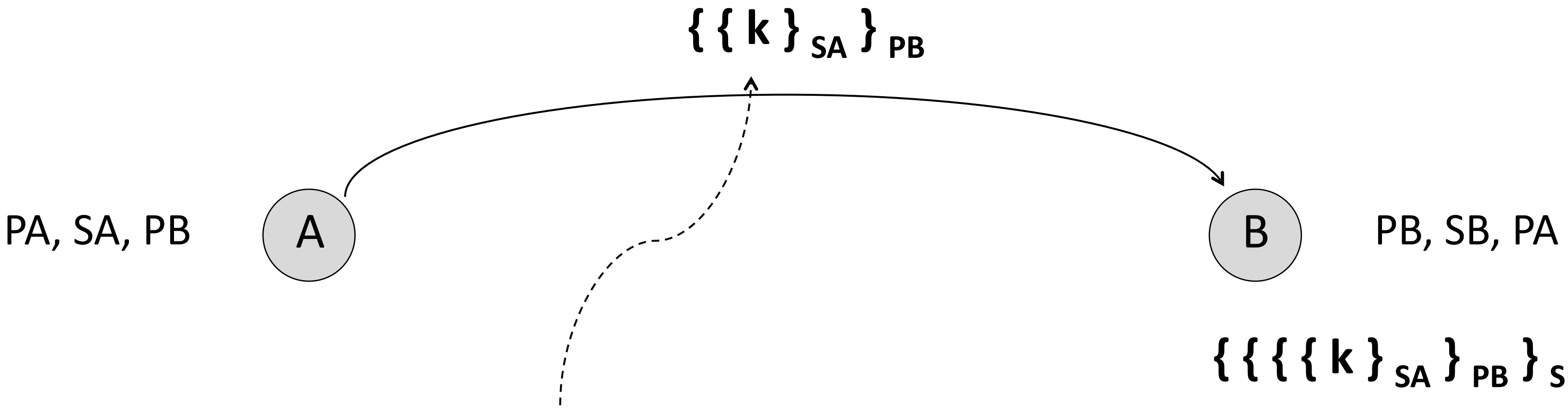
Secrecy Between A and B? **YES**
Authentication of A by B? **YES**

Does this approach scale? **YES**

Is this approach efficient (cryptographically)? **NO**

Week 7

Secure Key Exchange



$\{ \{ k \}_{SA} \}_{PB}$

PA, SA, PB

A

B

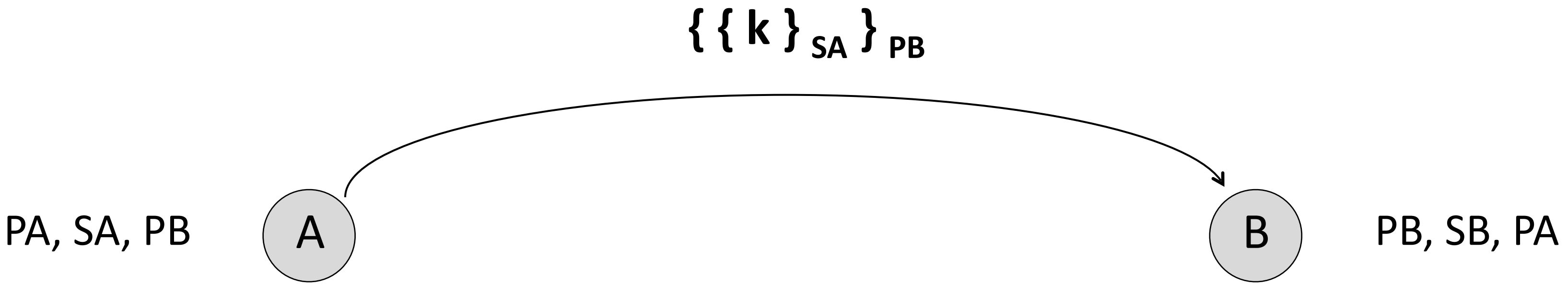
PB, SB, PA

$\{ \{ \{ \{ k \}_{SA} \}_{PB} \}_{SB} \}_{PA} = k$

Alice generates a key k for some bulk encryption algorithm (like 3-DES) and provides this key to B using secure key exchange

- *Scalable*
- *Secret*
- *Authenticated*

Secure Key Exchange



$\{ \{ k \}_{SA} \}_{PB}$

PA, SA, PB A

B PB, SB, PA

$\{ \{ \{ \{ k \}_{SA} \}_{PB} \}_{SB} \}_{PA} = m$

Secrecy Between A and B? **YES**
Authentication of A by B? **YES**

Does this approach scale? **YES**

Is this approach efficient (cryptographically)? **YES**

Week 7

Diffie-Hellman Key Exchange

Goal:

A and B share an encryption key k
with no KDC assistance

Diffie-Hellman Key Exchange

<u>Assume Two Publicly Known Parameters</u>:

p: Large Prime – Typically 1024 Bits
g: Primitive Element

Diffie-Hellman Key Exchange

<u>Step 1</u>:

A and B each locally generate
<u>private</u> random values a and b

Diffie-Hellman Key Exchange

p, g, a
g^a *mod p*

A

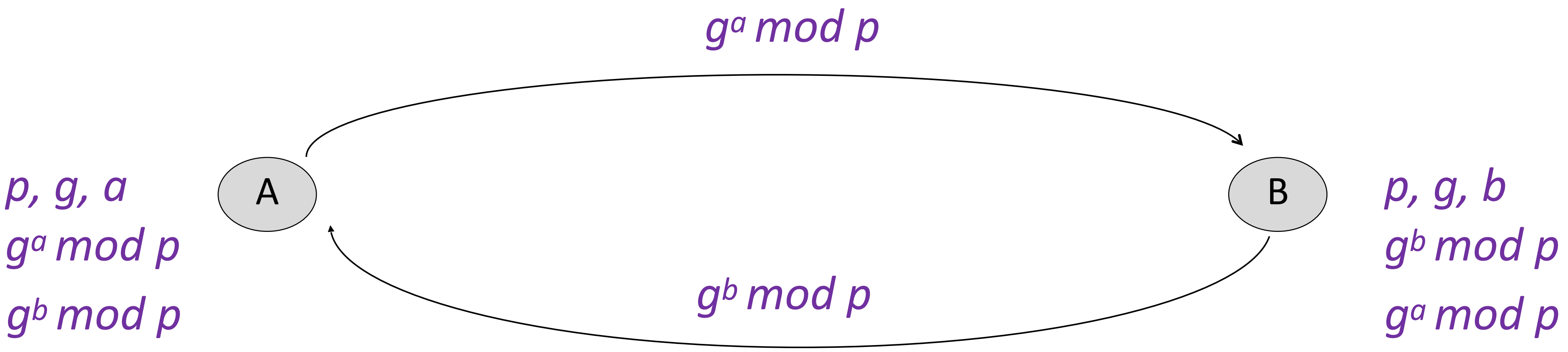
B

p, g, b
g^b *mod p*

<u>Step 2</u>:

A calculates g^a *mod p*
B calculates g^b *mod p*

Diffie-Hellman Key Exchange



g^a *mod p*

p, g, a
g^a *mod p*
g^b *mod p*

A

B

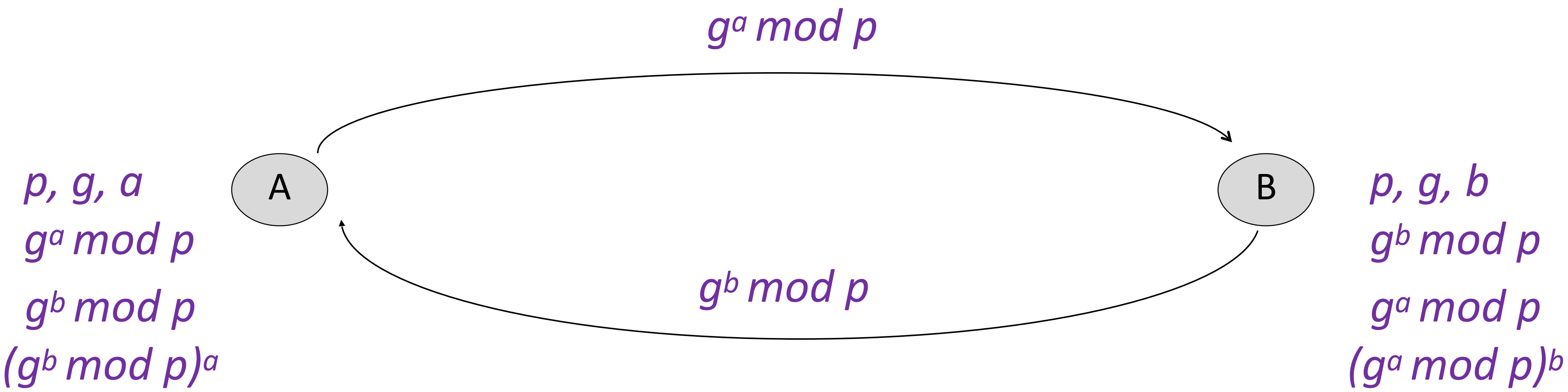
p, g, b
g^b *mod p*
g^a *mod p*

g^b *mod p*

Step 3:

A sends g^a mod p to B
B send g^b mod p to A

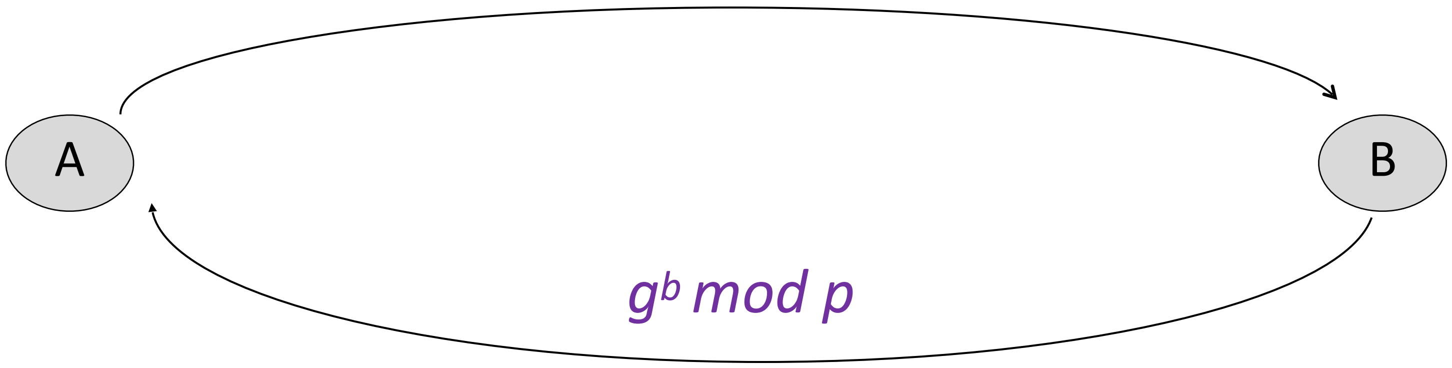
Diffie-Hellman Key Exchange



Step 4:

A computes $(g^a \bmod p)^b$ to B
B computes $(g^b \bmod p)^a$ to A

Diffie-Hellman Key Exchange



$g^a \bmod p$

A

B

$g^b \bmod p$

p, g, a

$g^a \bmod p$

$g^b \bmod p$

$(g^b \bmod p)^a =$

$g^{ba} \bmod p$

p, g, b

$g^b \bmod p$

$g^a \bmod p$

$(g^a \bmod p)^b =$

$g^{ab} \bmod p$

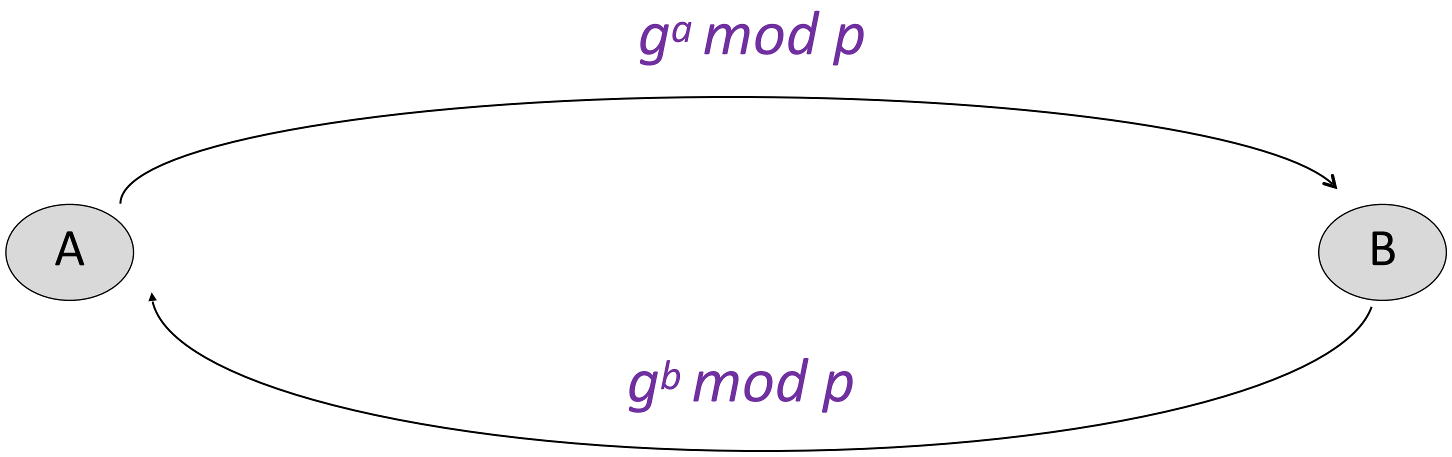
Step 5:

Shared Secret:

$g^{ab} \bmod p$

Week 7

Diffie-Hellman Key Exchange



p, g, a
$g^a \bmod p$
$g^b \bmod p$
$(g^b \bmod p)^a =$
$g^{ba} \bmod p$

p, g, b
$g^b \bmod p$
$g^a \bmod p$
$(g^a \bmod p)^b =$
$g^{ab} \bmod p =$
$g^{ba} \bmod p$

<u>Step 5</u>:

Shared Secret:
$g^{ba} \bmod p$

RSA Algorithm

Step 1: Select two prime numbers p and q, each about 100 decimal digits in length

Step 2: Calculate n = pq and
Ψ = (p – 1)(q – 1)

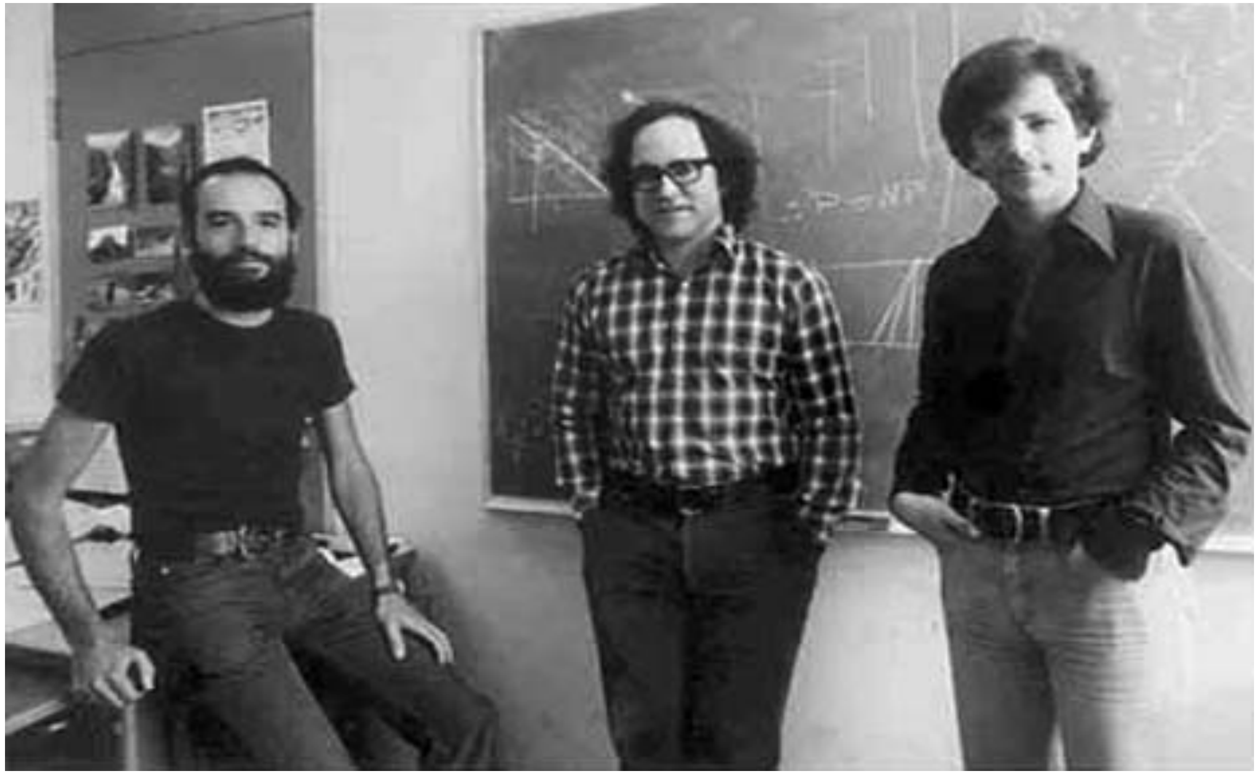
Step 3: Select integer E between 3 and Ψ, which has no common factors with Ψ

Step 4: Select integer D such that DE differs by 1 from a multiple of Ψ

Step 5: Make E, n public, but
keep p, q, D and Ψ secret

Encryption: $C = P^E \bmod n$

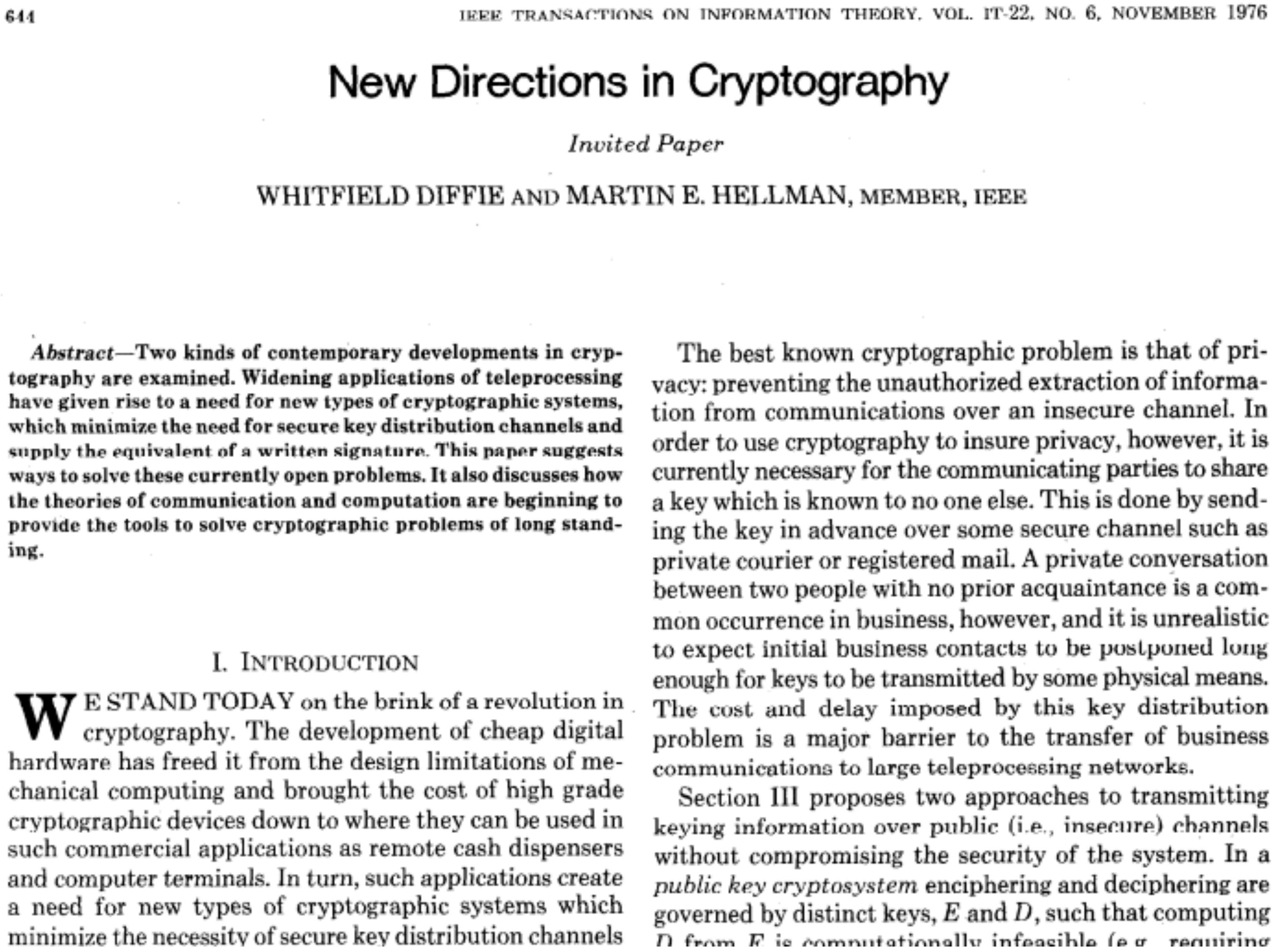
Decryption: $P = C^D \bmod n$



Example: p = 3, q = 5, n = 15, Ψ = 8 Select E = 5, D = 5
Encrypt “2”: $2^5 \bmod 15 = 2$
Decrypt “2”: $2^5 \bmod 15 = 2$

Week 7

Public Key Cryptography – Original Paper By Diffie and Hellman



New Directions in Cryptography

Invited Paper

WHITFIELD DIFFIE AND MARTIN E. HELLMAN, MEMBER, IEEE

***Abstract*—Two kinds of contemporary developments in cryptography are examined. Widening applications of teleprocessing have given rise to a need for new types of cryptographic systems, which minimize the need for secure key distribution channels and supply the equivalent of a written signature. This paper suggests ways to solve these currently open problems. It also discusses how the theories of communication and computation are beginning to provide the tools to solve cryptographic problems of long standing.**

I. Introduction

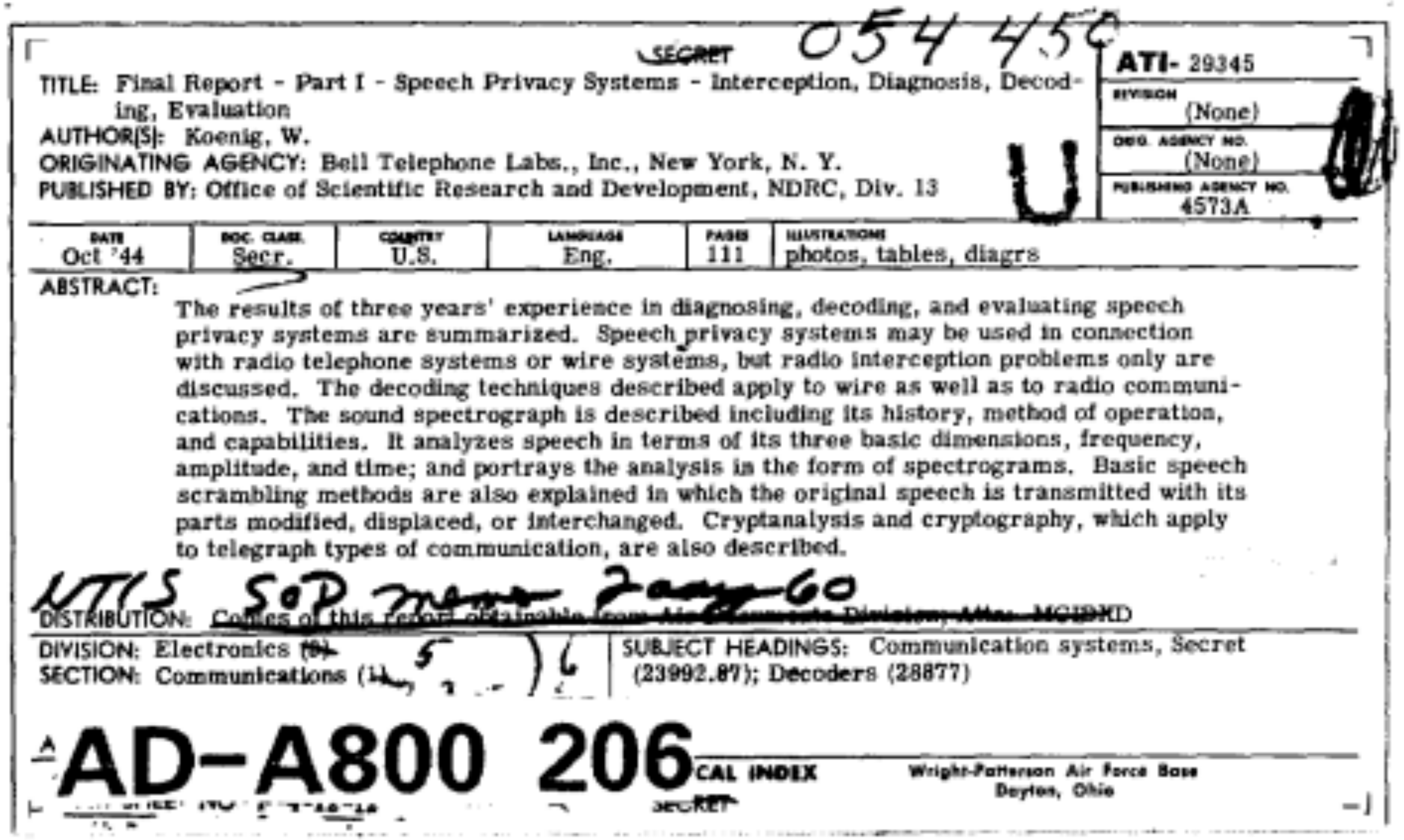
WE STAND TODAY on the brink of a revolution in cryptography. The development of cheap digital hardware has freed it from the design limitations of mechanical computing and brought the cost of high grade cryptographic devices down to where they can be used in such commercial applications as remote cash dispensers and computer terminals. In turn, such applications create a need for new types of cryptographic systems which minimize the necessity of secure key distribution channels

The best known cryptographic problem is that of privacy: preventing the unauthorized extraction of information from communications over an insecure channel. In order to use cryptography to insure privacy, however, it is currently necessary for the communicating parties to share a key which is known to no one else. This is done by sending the key in advance over some secure channel such as private courier or registered mail. A private conversation between two people with no prior acquaintance is a common occurrence in business, however, and it is unrealistic to expect initial business contacts to be postponed long enough for keys to be transmitted by some physical means. The cost and delay imposed by this key distribution problem is a major barrier to the transfer of business communications to large teleprocessing networks.

Section III proposes two approaches to transmitting keying information over public (i.e., insecure) channels without compromising the security of the system. In a *public key cryptosystem* enciphering and deciphering are governed by distinct keys, E and D, such that computing D from E is computationally infeasible (e.g., requiring

Week 7

Bell Labs – Project C43 (1944)



SECRET 054 450

ATI- 29345

DIVISION (None)

ORIG. AGENCY NO. (None)

PUBLISHING AGENCY NO. 4573A

TITLE: Final Report - Part I - Speech Privacy Systems - Interception, Diagnosis, Decoding, Evaluation

AUTHOR(S): Koenig, W.

ORIGINATING AGENCY: Bell Telephone Labs., Inc., New York, N. Y.

PUBLISHED BY: Office of Scientific Research and Development, NDRC, Div. 13

U

DATE	DOC. CLASS.	COUNTRY	LANGUAGE	PAGES	ILLUSTRATIONS
Oct '44	Secr.	U.S.	Eng.	111	photos, tables, diagrs

ABSTRACT:

The results of three years' experience in diagnosing, decoding, and evaluating speech privacy systems are summarized. Speech privacy systems may be used in connection with radio telephone systems or wire systems, but radio interception problems only are discussed. The decoding techniques described apply to wire as well as to radio communications. The sound spectrograph is described including its history, method of operation, and capabilities. It analyzes speech in terms of its three basic dimensions, frequency, amplitude, and time; and portrays the analysis in the form of spectrograms. Basic speech scrambling methods are also explained in which the original speech is transmitted with its parts modified, displaced, or interchanged. Cryptanalysis and cryptography, which apply to telegraph types of communication, are also described.

UTIS SOP memo 7 aug 60

DISTRIBUTION: ~~Copies of this report obtainable from Air Documents Division; Attn: MCIDXD~~

DIVISION: Electronics (~~9~~) 5) 6

SECTION: Communications (~~1~~)

SUBJECT HEADINGS: Communication systems, Secret (23992.87); Decoders (28877)

AD-A800 206

CAL INDEX

Wright-Patterson Air Force Base
Dayton, Ohio

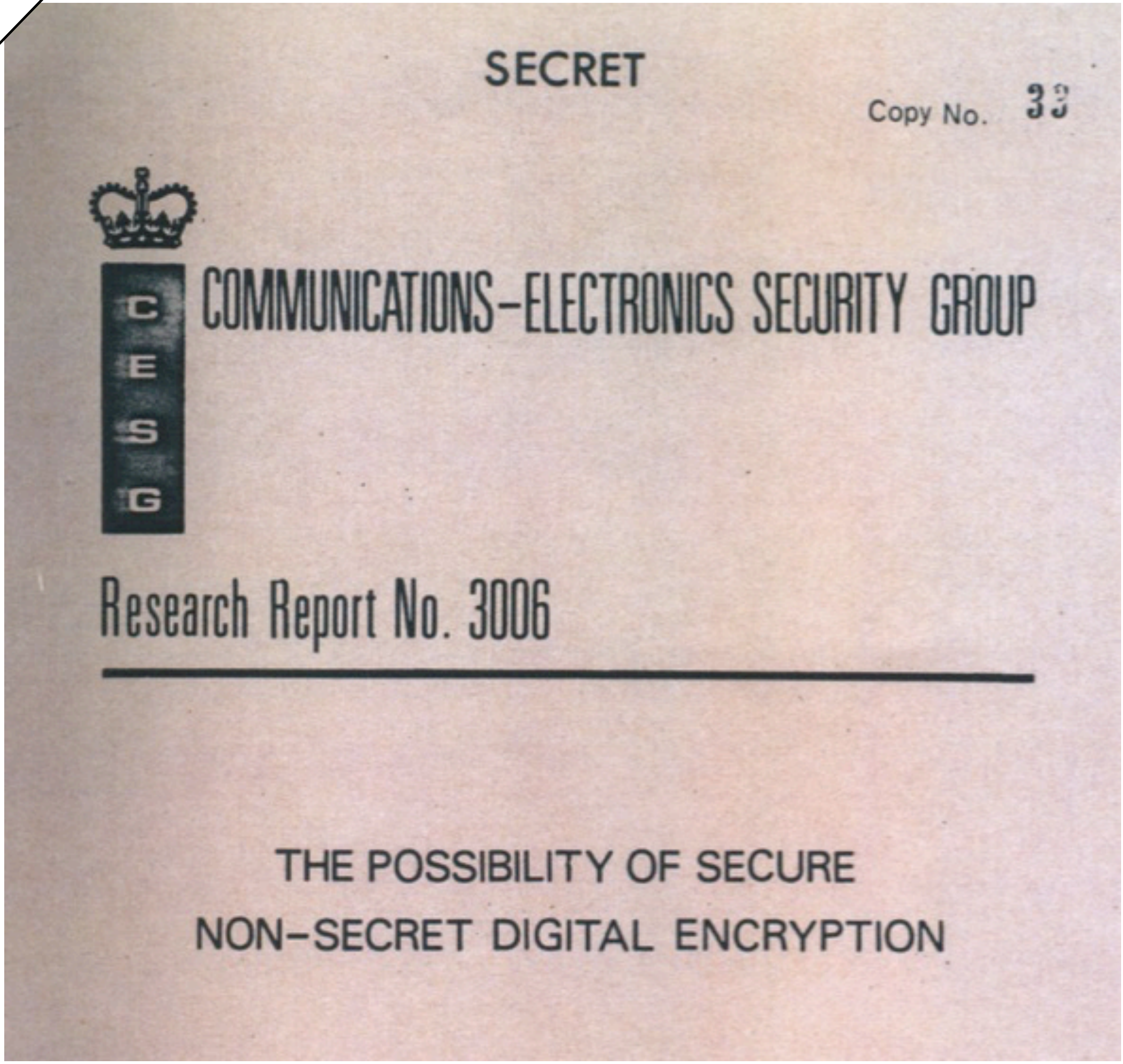
SECRET

GCHQ – Original and New Headquarters in Cheltenham, UK

Week 7

James Ellis, Engineer at GCHQ – Circa 1969

James Ellis' Paper 1970 – Classified for Three Decades



SECRET

Copy No. 33

CESG

COMMUNICATIONS-ELECTRONICS SECURITY GROUP

Research Report No. 3006

THE POSSIBILITY OF SECURE
NON-SECRET DIGITAL ENCRYPTION

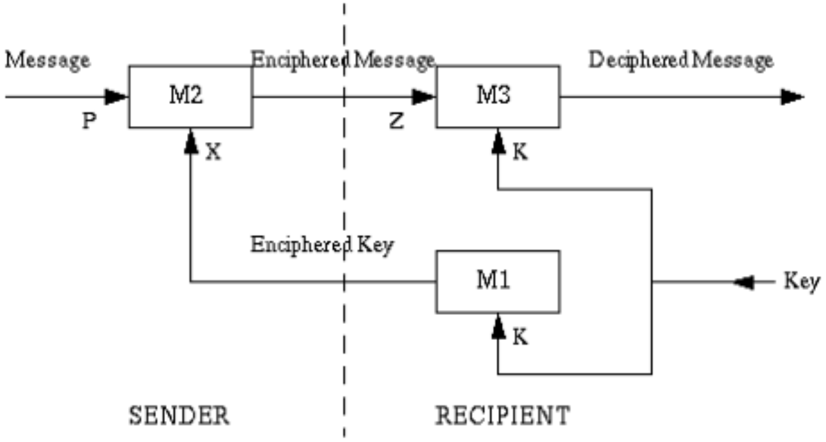


Fig. 1

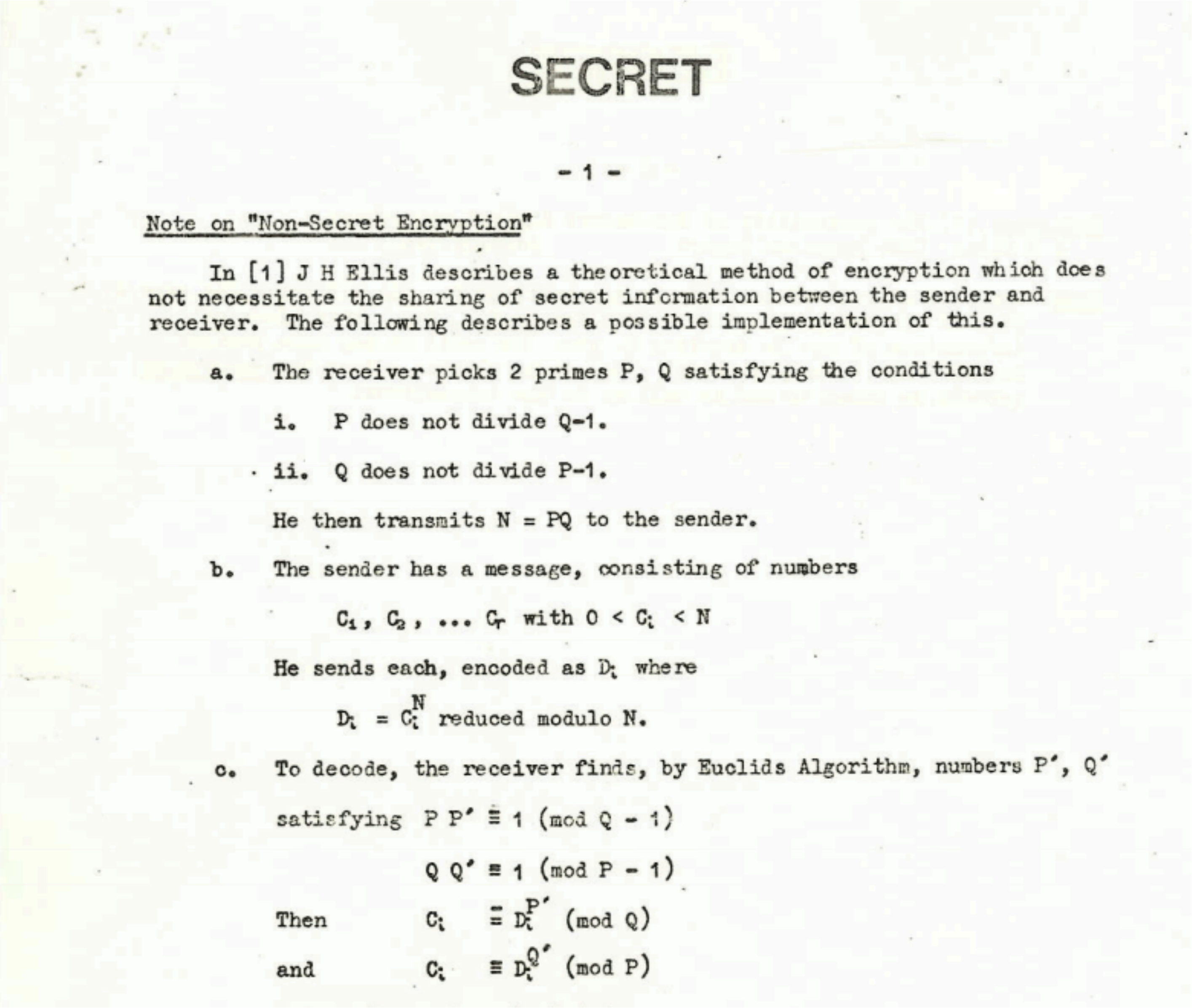
13. The following properties are clearly essential. It must be impossible for the interceptor to obtain p from z without knowing k even though he knows x. Also, since a knowledge of k would enable him to decipher z, he must be unable to obtain k from x. Finally M3 must have the property of being able to decipher z. To obtain these properties we specify the look-up tables corresponding to MI, M2 and M3 in the following way: -

a. Let k have n different possible values and p have m different possible values, for simplicity take them to be the integers 1 to n and 1 to m respectively. Let x have the same range of values as k, and z have the same range as p.

b. MI can be defined as a linear look-up table of n entries whose contents are the numbers 1 to n in a random order, where "random" implies that the output is sufficiently uncorrelated with the input so that the position of a particular entry in the table cannot be found in a simpler way than by searching through the table.

c. M2 corresponds to an n by m rectangular table in which the entries for a fixed value of x consists of the numbers 1 to m in random order, and where the columns for the various values of x are suitable uncorrelated with one another.

Clifford Cocks and Malcolm Williamson



SECRET

- 1 -

Note on "Non-Secret Encryption"

In [1] J H Ellis describes a theoretical method of encryption which does not necessitate the sharing of secret information between the sender and receiver. The following describes a possible implementation of this.

a. The receiver picks 2 primes P, Q satisfying the conditions

i. P does not divide Q-1.

ii. Q does not divide P-1.

He then transmits N = PQ to the sender.

b. The sender has a message, consisting of numbers

$C_1, C_2, \ldots C_r$ with $0 < C_i < N$

He sends each, encoded as D_i where

$D_i = C_i^N$ reduced modulo N.

c. To decode, the receiver finds, by Euclids Algorithm, numbers P′, Q′

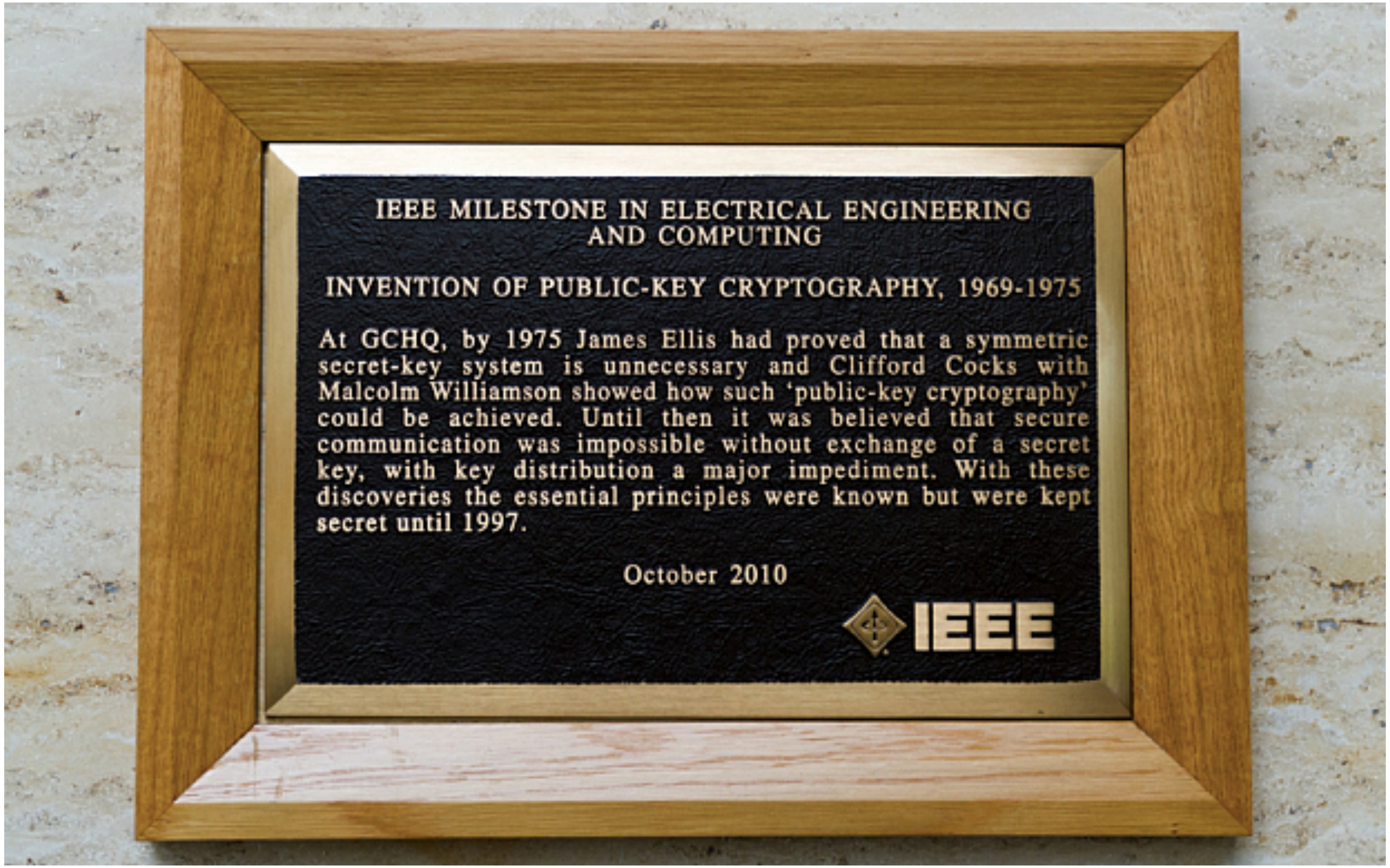
satisfying $P P' \equiv 1 \pmod{Q - 1}$

$Q Q' \equiv 1 \pmod{P - 1}$

Then $C_i \equiv D_i^{P'} \pmod{Q}$

and $C_i \equiv D_i^{Q'} \pmod{P}$

Credit Where Credit is Due



IEEE MILESTONE IN ELECTRICAL ENGINEERING
AND COMPUTING

INVENTION OF PUBLIC-KEY CRYPTOGRAPHY, 1969-1975

At GCHQ, by 1975 James Ellis had proved that a symmetric secret-key system is unnecessary and Clifford Cocks with Malcolm Williamson showed how such 'public-key cryptography' could be achieved. Until then it was believed that secure communication was impossible without exchange of a secret key, with key distribution a major impediment. With these discoveries the essential principles were known but were kept secret until 1997.

October 2010

IEEE