2011 年度春期

情報数学特論

(理工学専攻情報学領域)

(担当: 角皆)

本講義の概要

暗号 — 情報化社会の安全を支える数理技術

- 暗号理論とその利用
  - ⋆ 公開鍵暗号とその仕組み
  - ⋆ 鍵共有などの暗号プロトコル
  - ⋆ その実現 (RSA 暗号・楕円曲線暗号)
- 暗号を支える数学
  - ⋆ 有限体とその上の線型代数・Galois 理論
  - ⋆ 楕円曲線の理論の入門
  - ⋆ 計算量の理論

情報通信を行なう際の要請

- **効率的に** $\longrightarrow$ 情報理論
- **確実に** $\longrightarrow$ 符号理論
- **安全に** $\longrightarrow$ 暗号理論

情報通信を行なう際の要請

- **効率的に** $\longrightarrow$ **情報理論**
- **確実に** $\longrightarrow$ **符号理論**
- **安全に** $\longrightarrow$ **暗号理論**

情報通信を行なう際の要請

- **効率的に** $\longrightarrow$ **情報理論**
- **確実に** $\longrightarrow$ **符号理論**
- **安全に** $\longrightarrow$ **暗号理論**

暗号理論 (共通鍵・公開鍵暗号)

- **安全な情報生活の為に**
  - ⋆ **秘密通信**
  - ⋆ **ディジタル認証・署名**
  - ⋆ **秘密分散**
  - ⋆ **鍵共有**

- **安全な暗号の実現**
  (RSA 暗号・楕円曲線暗号など)

- **安全性を計る (計算量の理論)**

暗号理論 (共通鍵・公開鍵暗号)

- 安全な情報生活の為に
  - ⋆ 秘密通信
  - ⋆ ディジタル認証・署名
  - ⋆ 秘密分散
  - ⋆ 鍵共有

- 安全な暗号の実現
  (RSA 暗号・楕円曲線暗号など)

- 安全性を計る (計算量の理論)

暗号の利用

- **古典的： 戦争・謀略など**
- **現代： 情報通信一般**

$\longrightarrow$ **個人の独立を守るための重要な数理技術**

暗号の利用

- **古典的： 戦争・謀略など**
- **現代： 情報通信一般**

$\longrightarrow$ **個人の独立を守るための重要な数理技術**

暗号の利用

- **古典的： 戦争・謀略など**
- **現代： 情報通信一般**

$\longrightarrow$ **個人の独立を守るための重要な数理技術**

既に身近な暗号の利用

コンピュータを使う際の
パスワードによる本人認証にも
暗号 (暗号化) が使われている

入力したパスワードを
保管してあるデータと照合しているのだが、

実は、
パスワードそのものを保管しているのでは**ない**

定まった方式 (暗号化関数) で
パスワードを変換して保管している

パスワードによる本人認証

- 暗号化関数でパスワードを変換して保管
- 入力したパスワードを
  暗号化関数で変換して照合

暗号化関数に要請される性質は ？

保管してある文字列が露見しても
元のパスワードが判明しない

… 一方向性関数 (one-way function)

パスワードによる本人認証

- 暗号化関数でパスワードを変換して保管
- 入力したパスワードを
  暗号化関数で変換して照合

暗号化関数に要請される性質は？

保管してある文字列が露見しても
元のパスワードが判明しない

… **一方向性関数** (one-way function)

パスワードによる本人認証

良い暗号化関数で変換して保管していても

通信経路の途中で盗聴されてしまっては

パスワードが露見してしまう

$\longrightarrow$ 暗号による秘密通信

パスワードによる本人認証

良い暗号化関数で変換して保管していても

通信経路の途中で盗聴されてしまっては

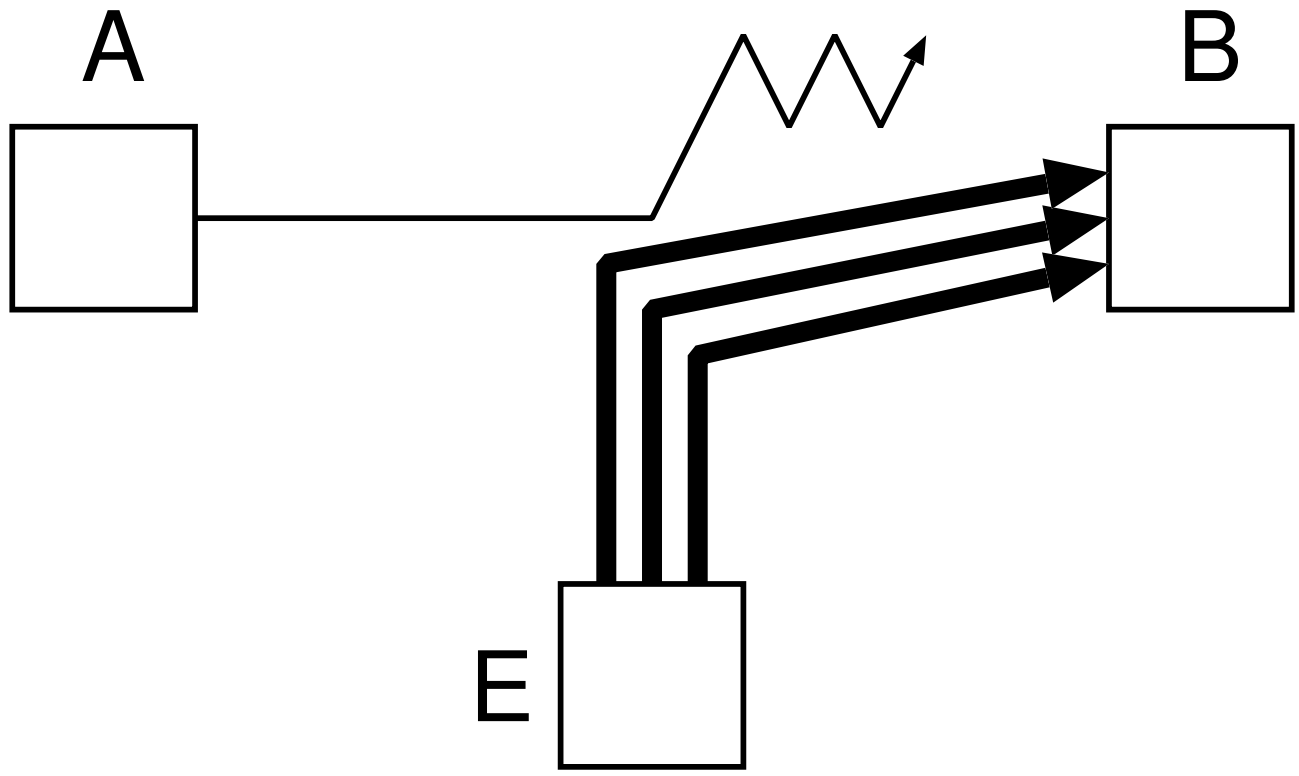
パスワードが露見してしまう

$\longrightarrow$ **暗号**による秘密通信

安全な情報伝達を阻害するもの

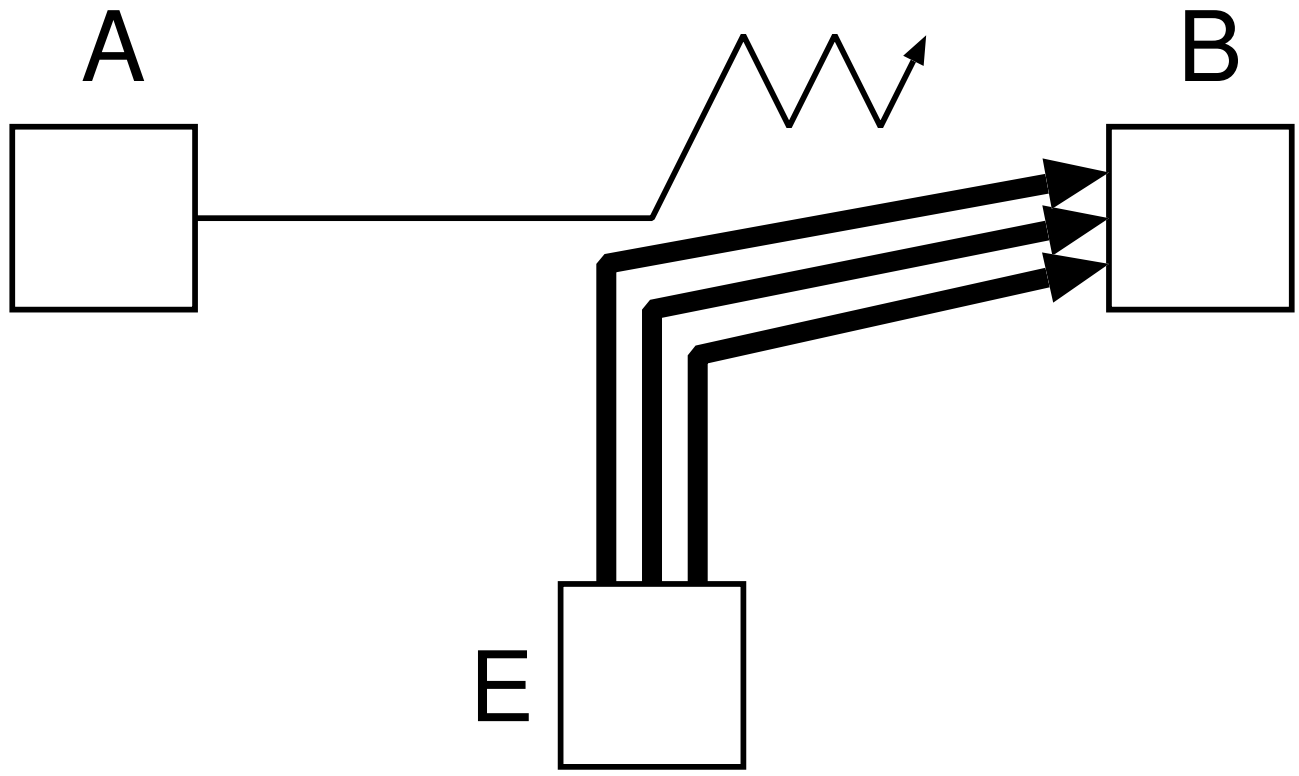
- 妨害 (DoS 攻撃など)
- 盗聴
- 改竄
- なり済まし　　など

DoS (Denial of Service) 攻撃



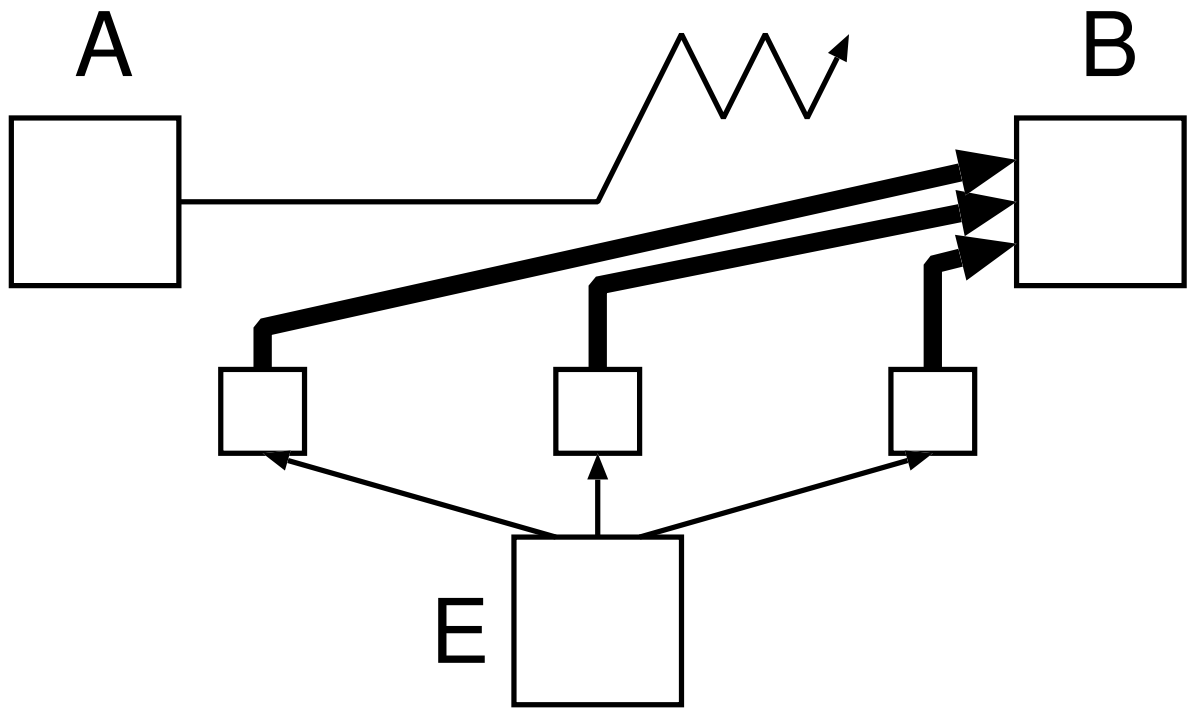
B を機能停止に追い込むには
E に相当のマシンパワーが必要
そこで実際には ...

DoS (Denial of Service) 攻撃



B を機能停止に追い込むには
E に相当のマシンパワーが必要
そこで実際には ...

DoS (Denial of Service) 攻撃



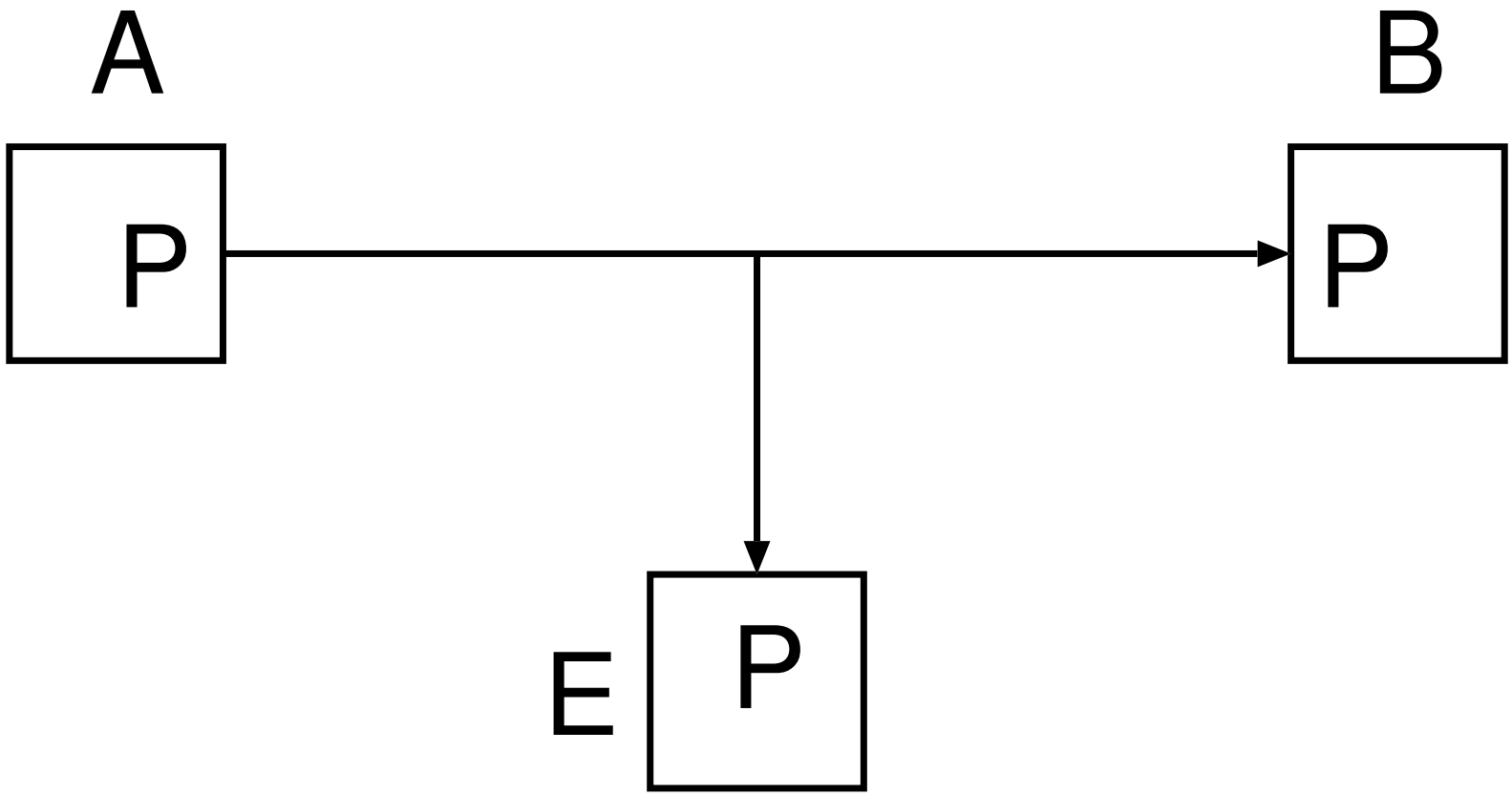
実際には、

コンピュータウイルス・乗っ取りなどで

制御下に置いた多数の機械から一斉に攻撃

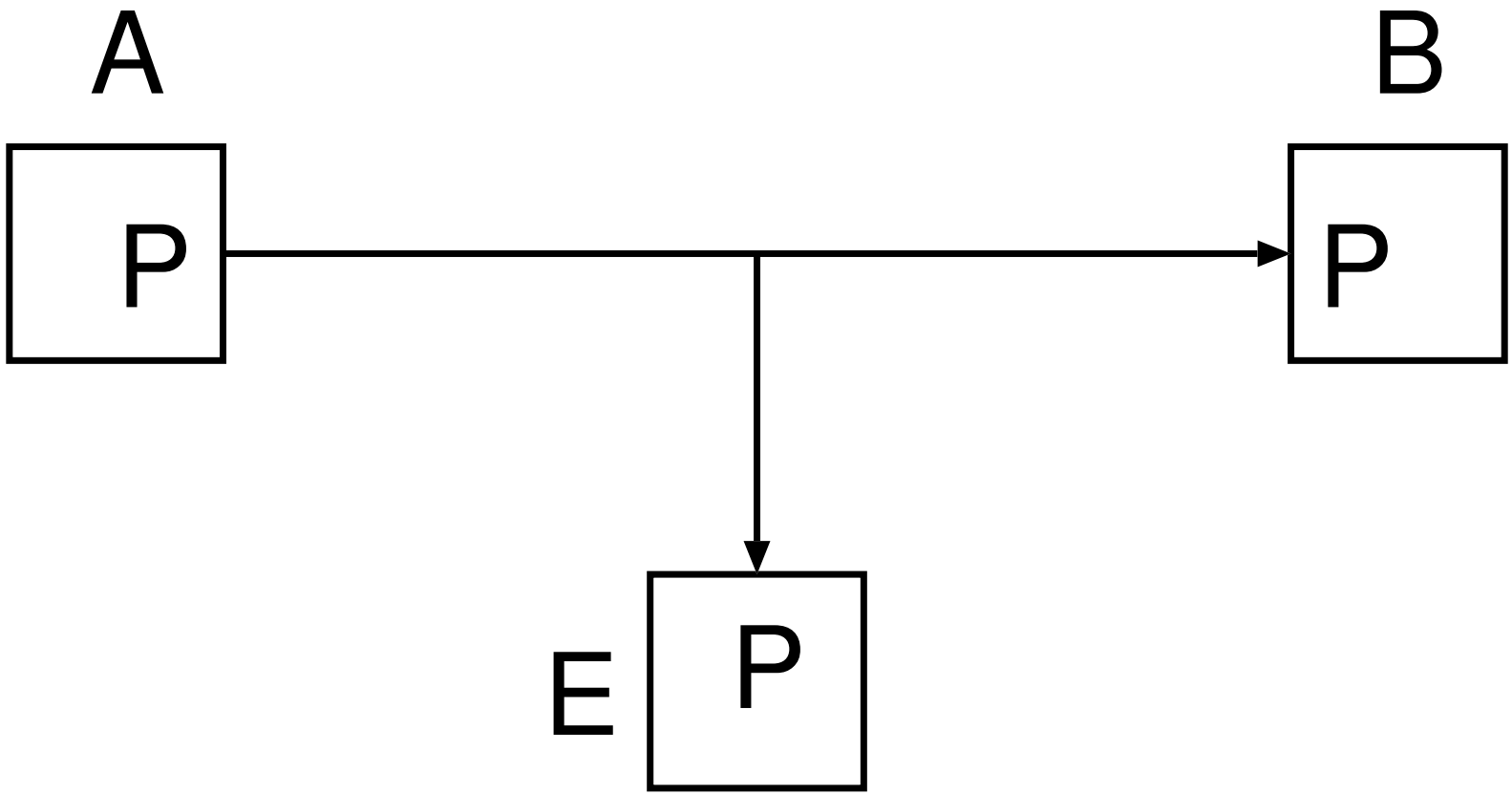
(Distributed DoS, DDoS)

盗聴 (tapping)



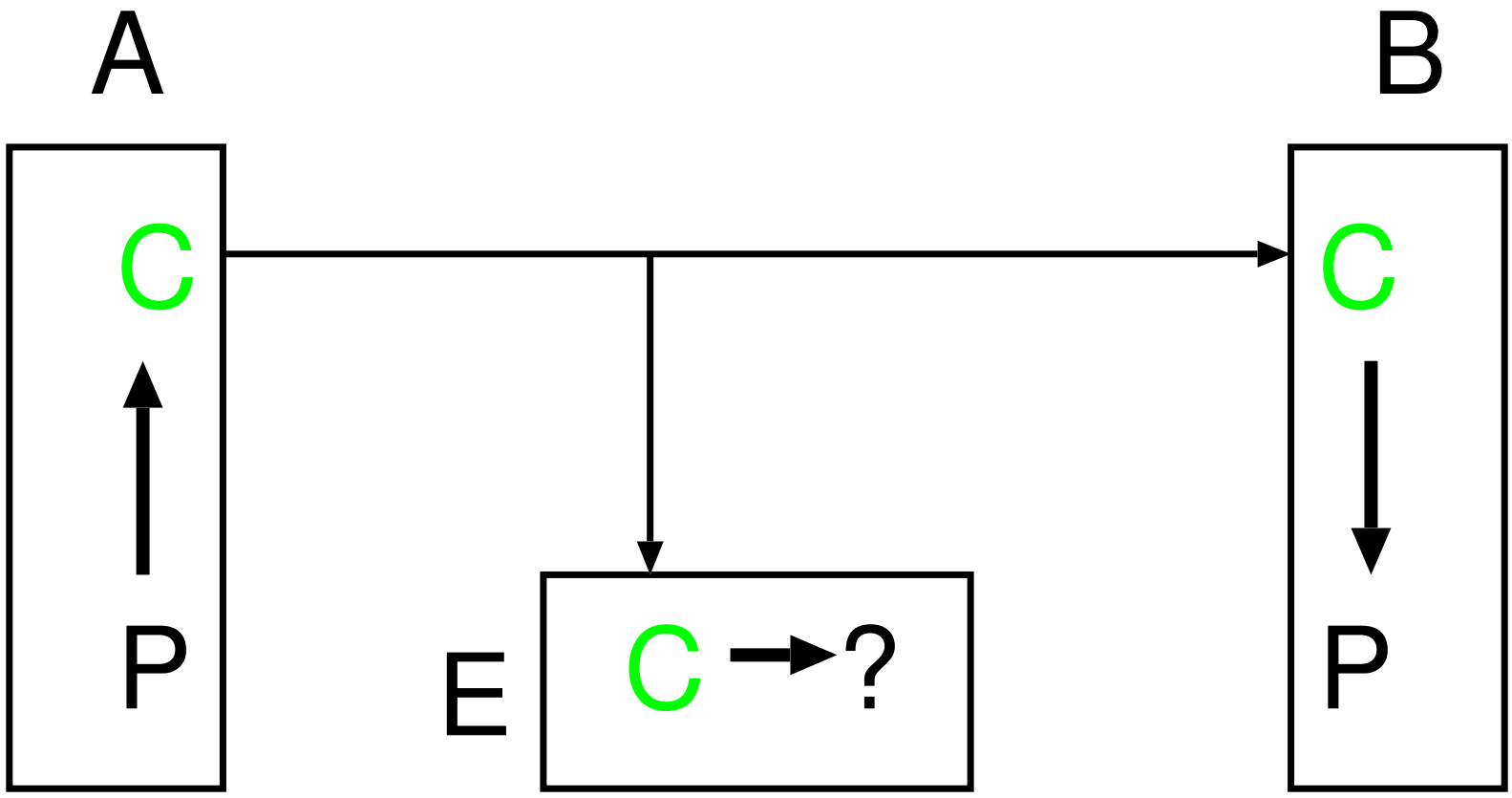
現在の計算機ネットワークの仕組みでは、
事実上、通信経路は誰にでも見られる

盗聴 (tapping)



現在の計算機ネットワークの仕組みでは、
事実上、通信経路は誰にでも見られる

暗号通信で盗聴対策

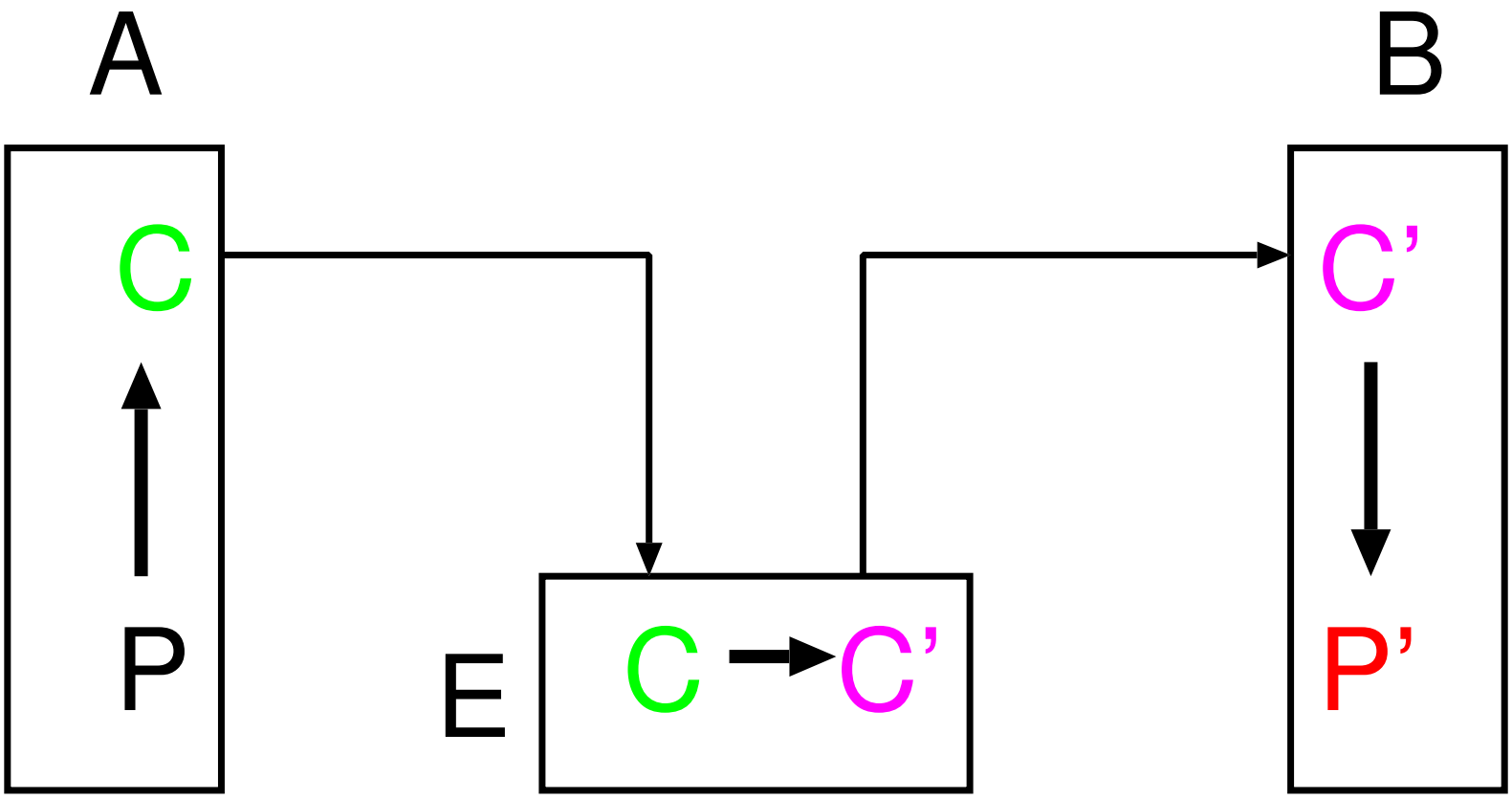


P : 平文 (plain text),　C : 暗号文 (ciphertext)

P → C : 暗号化 (encryption)

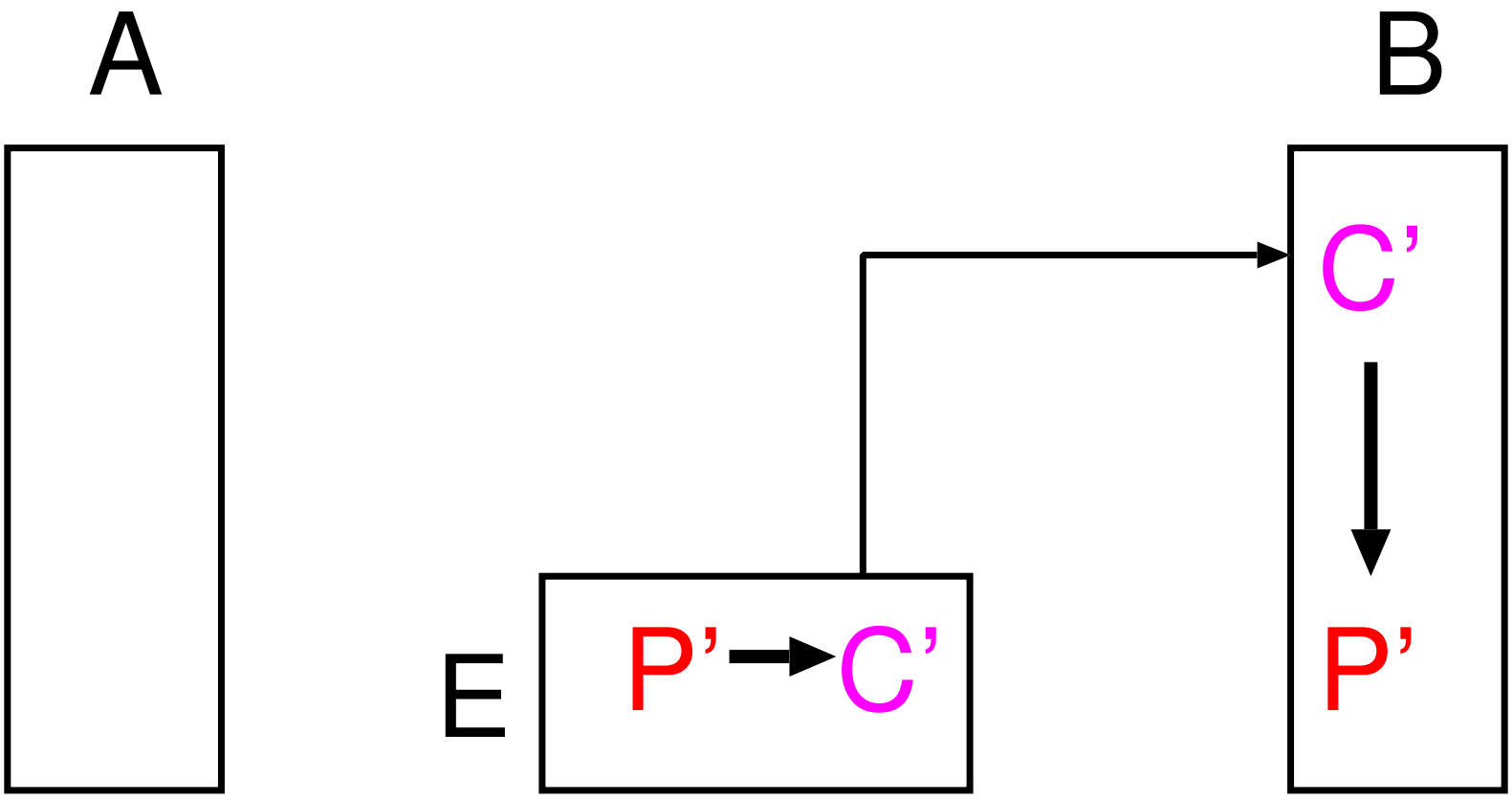
C → P : 復号 (decryption) • 解読 (cryptanalysis)

改竄 (tampering)



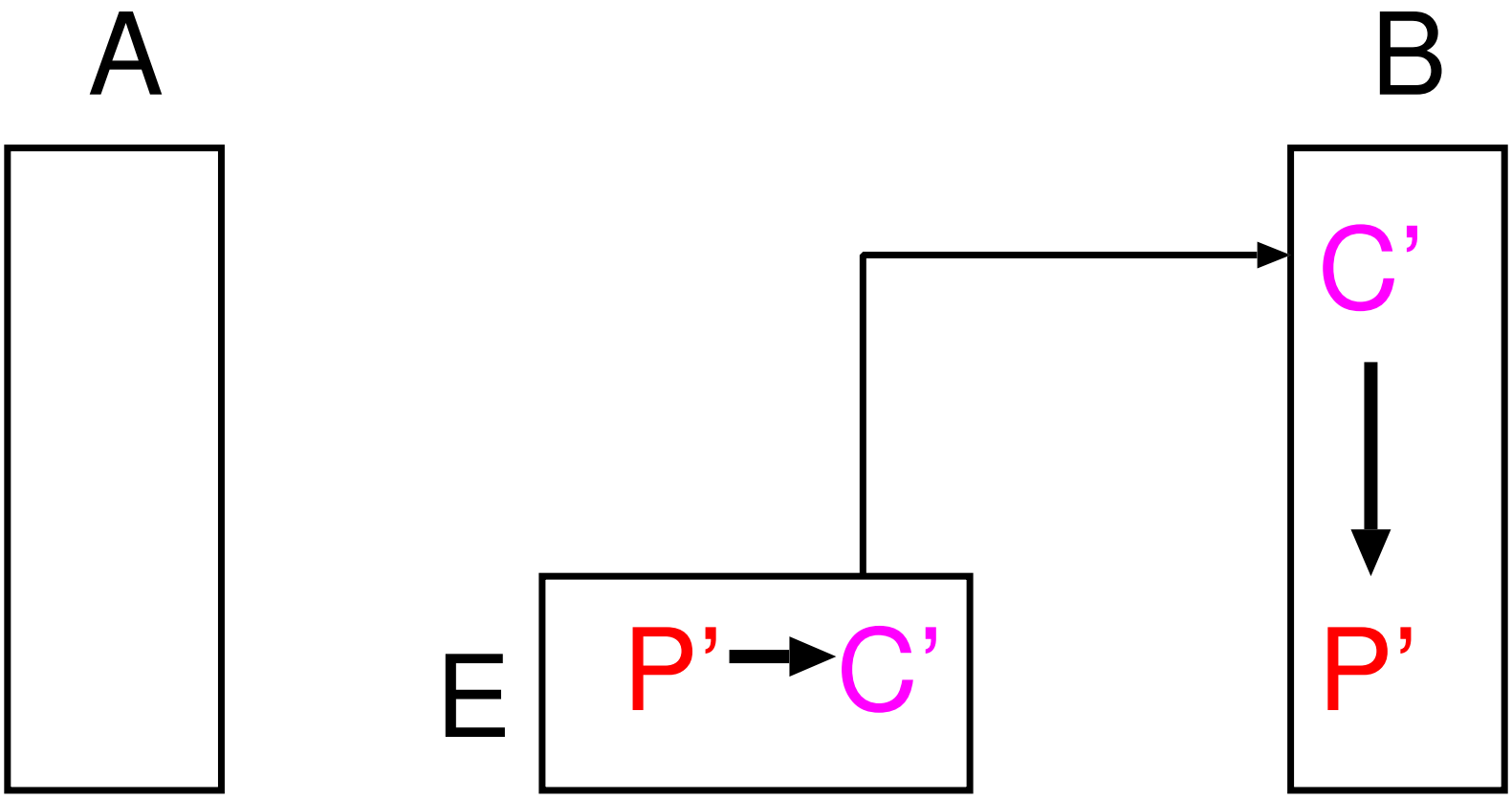
A が送信した情報であることを
確かめられるような仕組みが必要
(電子認証・電子署名)

なり済まし (disguise)



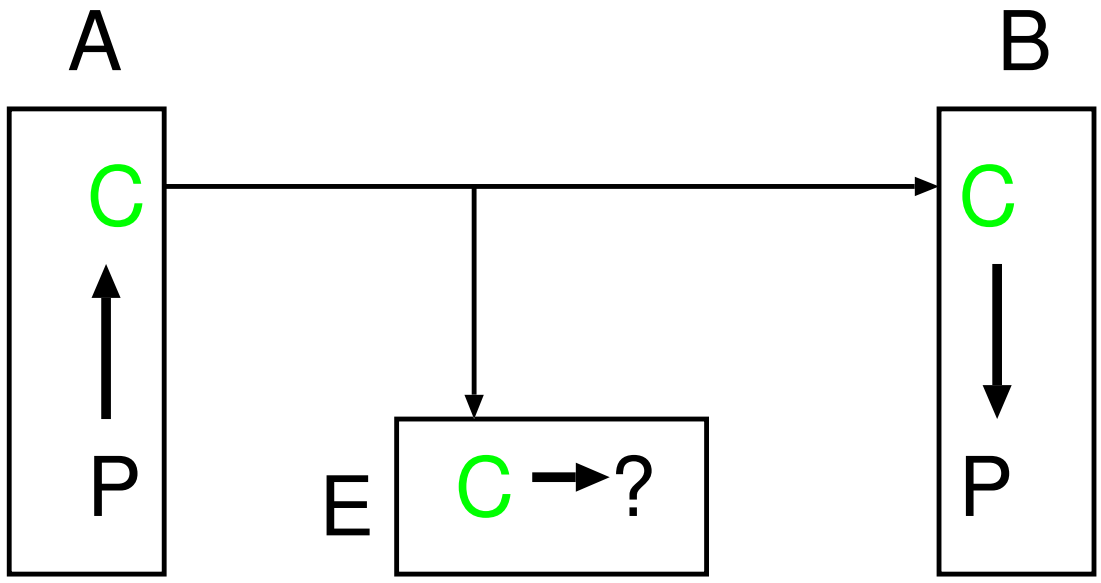
A が送信した情報であることを
確かめられるような仕組みが必要
(電子認証・電子署名)

なり済まし (disguise)



A が送信した情報であることを
確かめられるような仕組みが必要
(電子認証・電子署名)

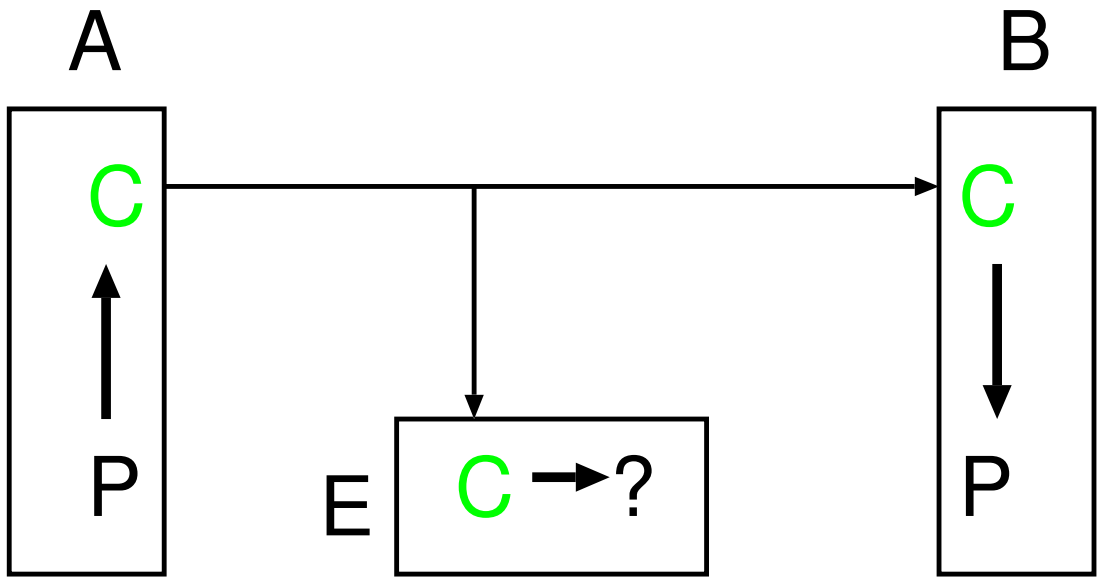
暗号 (cryptography)



- 送信者 A が平文 P を暗号化、暗号文 C を送信
- 受信者 B が暗号文 C を受信、平文 P に復号
- 盗聴者 E は暗号文 C を知っても
  平文 P を復元できない

⟶ B だけが復号鍵を持っていることが必要

暗号 (cryptography)



- 送信者 A が平文 P を暗号化、暗号文 C を送信
- 受信者 B が暗号文 C を受信、平文 P に復号
- 盗聴者 E は暗号文 C を知っても
  平文 P を復元できない

$\longrightarrow$ B だけが**復号鍵**を持っていることが必要

暗号通信

公開された情報伝達路 (盗聴可能 と仮定) で、

暗号方式を公開 して通信

- 秘密鍵暗号 (共通鍵暗号)
- 公開鍵暗号

暗号通信

公開された情報伝達路 (盗聴可能 と仮定) で、

暗号方式を公開 して通信

- **秘密鍵暗号 (共通鍵暗号)**
- **公開鍵暗号**

秘密鍵 (共通鍵) 暗号

暗号化鍵・復号鍵が同じ

- 換字暗号・Caesar 暗号
- 線型ブロック暗号
- Vernam 暗号
- DES (Data Encryption Standard)
- AES (Advances Encryption Standard)

秘密鍵 (共通鍵) 暗号の特徴

暗号化鍵・復号鍵が同じ

- 一般に原理は簡単で高速
- 事前の鍵共有の必要
- 通信相手毎に別の鍵が必要

現代における暗号への要請

現在の情報化社会では
様々な場面で暗号が使われている

例：インターネット取引 (ネットショッピングなど)

- 不特定多数の人と暗号通信をしたい
- 事前に鍵を共有できない

$\longrightarrow$ 共通鍵暗号では実現が困難

$\longrightarrow$ 公開鍵暗号・鍵共有方式のアイデア
(1976〜77)

現代における暗号への要請

現在の情報化社会では
様々な場面で暗号が使われている

例：インターネット取引 (ネットショッピングなど)

- **不特定多数の人と暗号通信をしたい**
- **事前に鍵を共有できない**

$\longrightarrow$ 共通鍵暗号では実現が困難

$\longrightarrow$ 公開鍵暗号・鍵共有方式のアイデア
(1976〜77)

現代における暗号への要請

現在の情報化社会では
様々な場面で暗号が使われている

例：インターネット取引 (ネットショッピングなど)

- **不特定多数の人と暗号通信をしたい**
- **事前に鍵を共有できない**

$\longrightarrow$ 共通鍵暗号では実現が困難

$\longrightarrow$ 公開鍵暗号・鍵共有方式のアイデア
(1976〜77)

公開鍵暗号

暗号化鍵 (公開鍵)・復号鍵 (秘密鍵) が別

- 事前の鍵共有の必要無し
  $\longrightarrow$ 見ず知らずの人からも送ってもらえる

- 認証・署名機能がある
  $\longrightarrow$ 改竄・なり済ましの対策
  $\longrightarrow$ 否認防止の機能も持つ

公開鍵暗号

暗号化鍵 (公開鍵)・復号鍵 (秘密鍵) が別

- 事前の鍵共有の必要無し
  $\longrightarrow$ 見ず知らずの人からも送ってもらえる

- 認証・署名機能がある
  $\longrightarrow$ 改竄・なり済ましの対策
  $\longrightarrow$ 否認防止の機能も持つ

公開鍵暗号

暗号化鍵 (公開鍵)・復号鍵 (秘密鍵) が別

- 事前の鍵共有の必要無し
  $\longrightarrow$ 見ず知らずの人からも送ってもらえる

- 認証・署名機能がある
  $\longrightarrow$ 改竄・なり済ましの対策
  $\longrightarrow$ 否認防止の機能も持つ

公開鍵暗号

但し、一般には、
暗号化・復号が共通鍵暗号に比べて低速

そこで、

- **始めに公開鍵暗号方式で鍵を送付・共有**
- **その鍵を用いて秘密鍵暗号方式で通信**

というように、組合わせて用いることが多い

公開鍵暗号

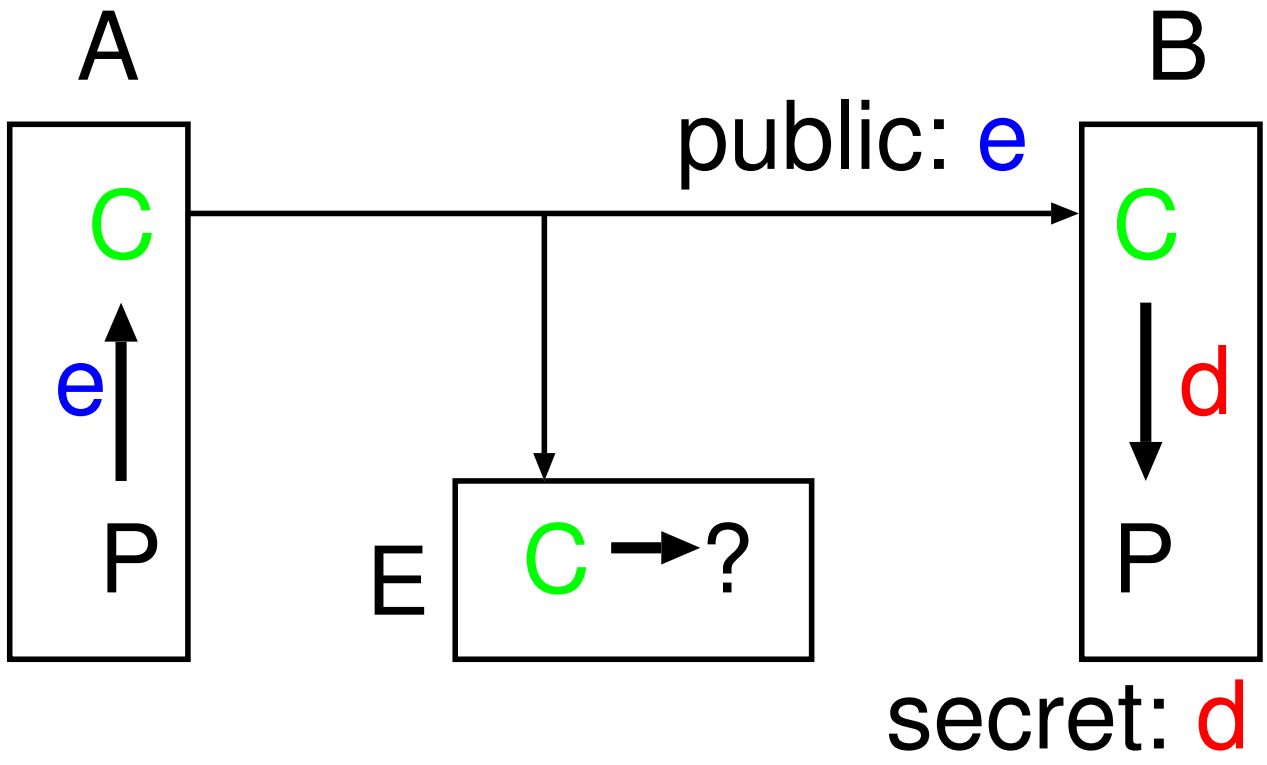
但し、一般には、
暗号化・復号が共通鍵暗号に比べて低速

そこで、

- 始めに公開鍵暗号方式で鍵を送付・共有
- その鍵を用いて秘密鍵暗号方式で通信

というように、組合わせて用いることが多い

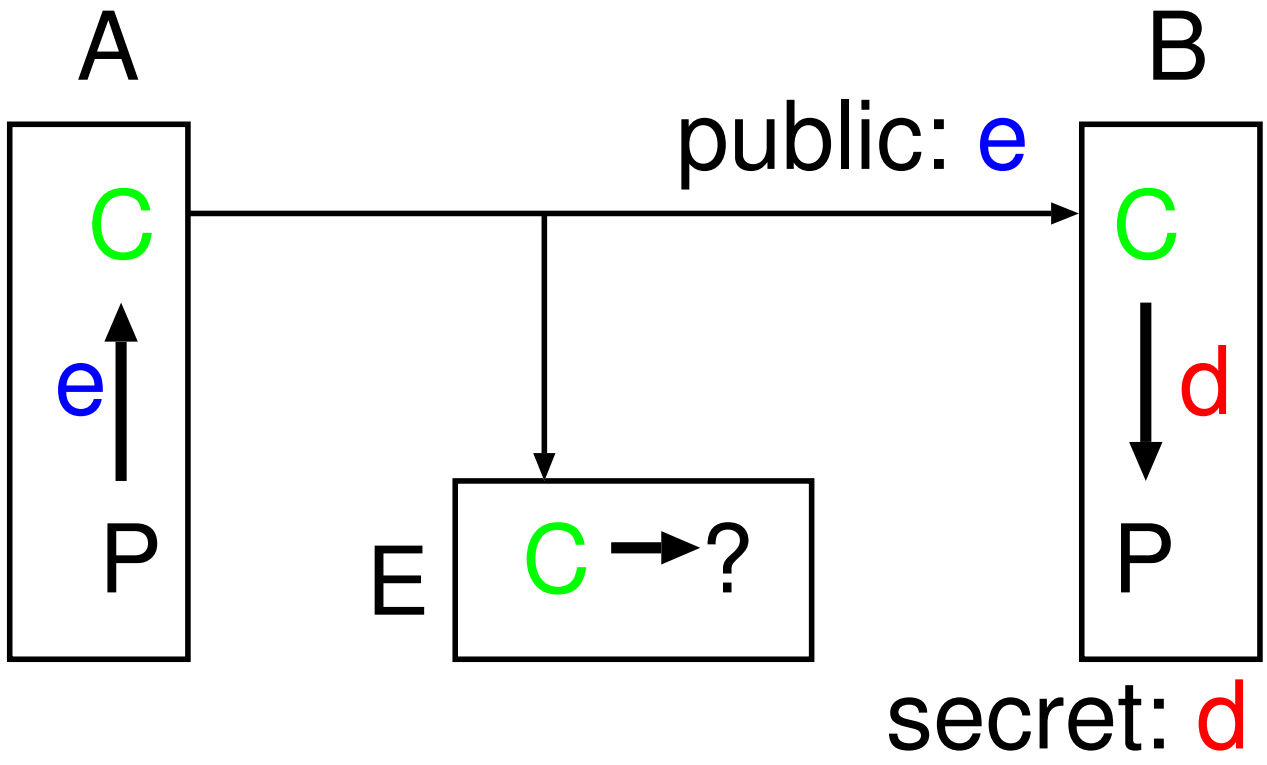
公開鍵暗号による暗号通信



しかし、これだと誰でも暗号化できるので、
A 氏が送った保証がない

⟶ 署名の必要性

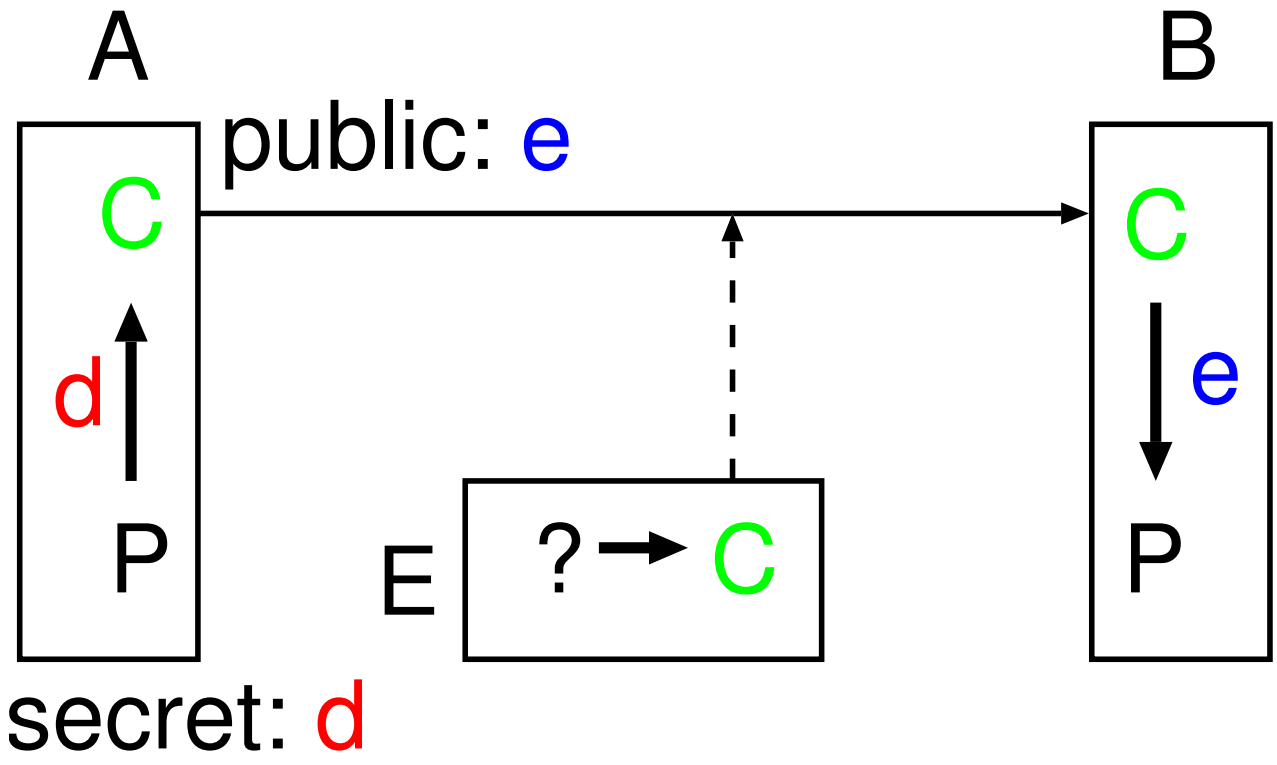
公開鍵暗号による暗号通信



しかし、これだと誰でも暗号化できるので、
A 氏が送った保証がない

⟶ 署名の必要性

公開鍵暗号を用いた認証・署名

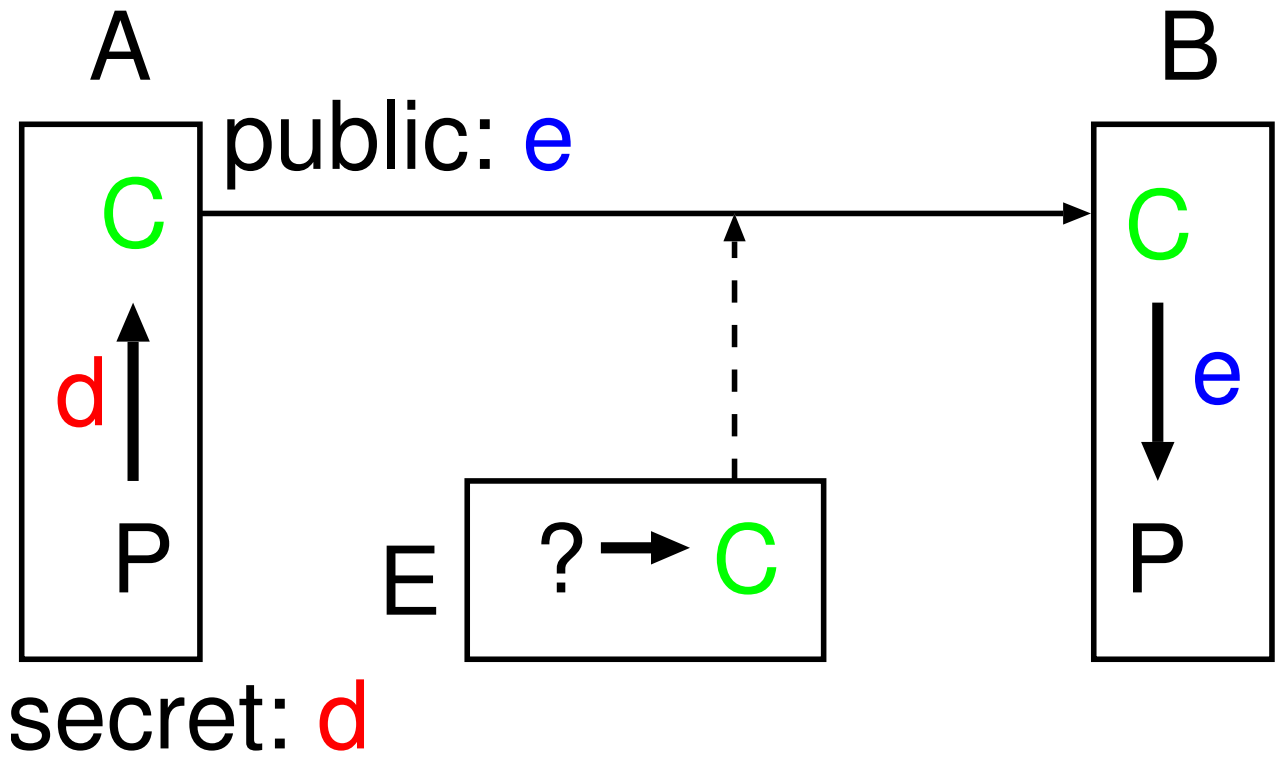


盗聴者 E 氏は

平文 P は判らないが、暗号文 C は盗聴可能

⟶ いつも同じ署名は使えない

公開鍵暗号を用いた認証・署名



盗聴者 E 氏は

平文 P は判らないが、暗号文 C は盗聴可能

⟶ いつも同じ署名は使えない

公開鍵暗号を用いた認証・署名

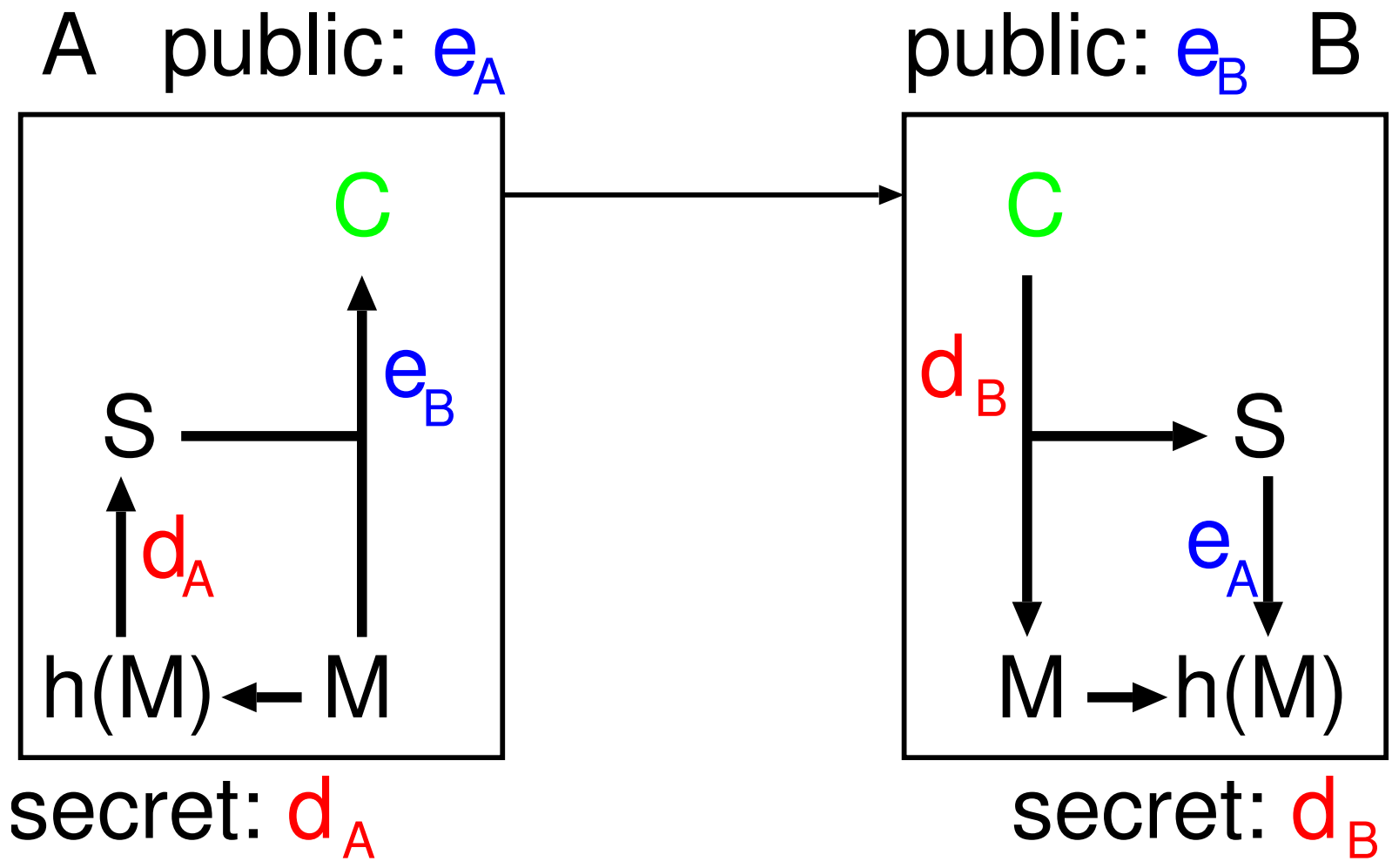
実際には、メッセージ本文 M に対して、

M から決まる短い値 (ハッシュ値) $\mathrm{h}(\mathrm{M})$ を
送信者 A 氏の秘密鍵で暗号化した文字列 S

を本文 M に添付して、

受信者 B 氏の公開鍵で一緒に暗号化して送る

公開鍵暗号を用いた認証・署名 2



公開鍵暗号の特徴

- 暗号化は誰でも出来る
- 復号は秘密鍵を知らないと出来ない
  (もの凄く時間が掛かる)

そんな都合の良い仕組みが本当にあるのか ？

$\longrightarrow$ ある!! (多分大丈夫)

計算困難な問題 を利用 (素因数分解・離散対数問題)

公開鍵暗号の特徴

- 暗号化は誰でも出来る
- 復号は秘密鍵を知らないと出来ない
  (もの凄く時間が掛かる)

そんな都合の良い仕組みが本当にあるのか ？

$\longrightarrow$ ある!! (多分大丈夫)

計算困難な問題 を利用 (素因数分解・離散対数問題)

公開鍵暗号の特徴

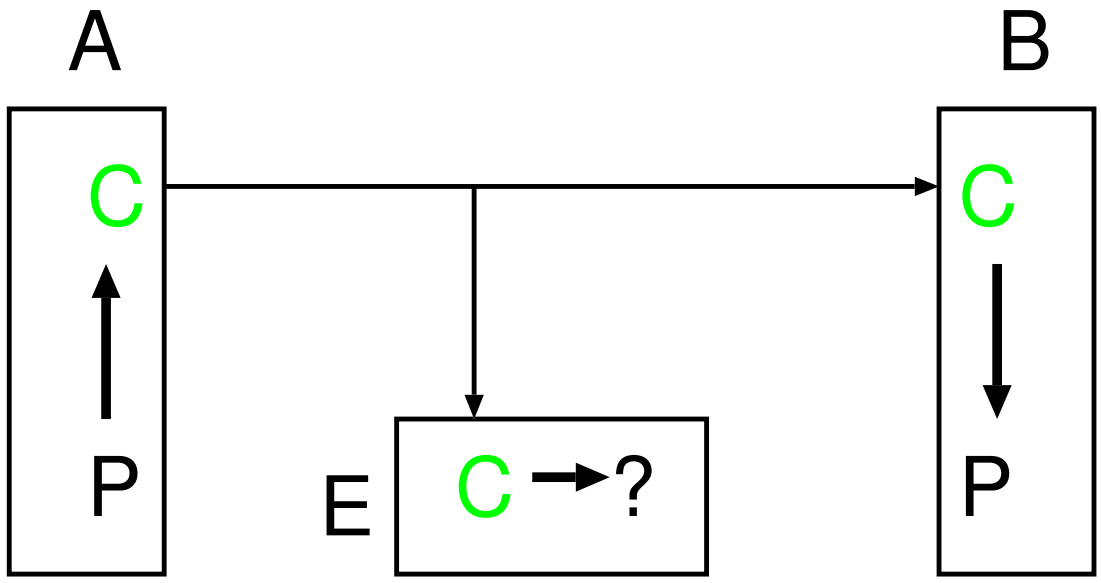
- 暗号化は誰でも出来る
- 復号は秘密鍵を知らないと出来ない
  (もの凄く時間が掛かる)

そんな都合の良い仕組みが本当にあるのか ？

$\longrightarrow$ ある!! (多分大丈夫)

計算困難な問題 を利用 (素因数分解・離散対数問題)

暗号の原理 (再掲)



- 送信者 A が平文 P を暗号化、暗号文 C を送信
- 受信者 B が暗号文 C を受信、平文 P に復号
- 盗聴者 E は暗号文 C を知っても
  平文 P を復元できない

⟶ B だけが**復号鍵**を持っていることが必要

計算困難な問題を利用した暗号

暗号に必要な性質：

- **一般には、盗聴しても解読できない**
- **復号鍵を持っている人だけは、受信すると復号できる**

暗号化関数への要請：

- **そのものの計算はさほど困難でない**
- **逆関数の計算は一般に困難**
- **特定の情報を知っている場合だけは、逆関数の計算が困難でない**

… 落とし戸関数 (trapdoor function)

計算困難な問題を利用した暗号

暗号に必要な性質：

- **一般には、盗聴しても解読できない**
- **復号鍵を持っている人だけは、受信すると復号できる**

暗号化関数への要請：

- **そのものの計算はさほど困難でない**
- **逆関数の計算は一般に困難**
- **特定の情報を知っている場合だけは、逆関数の計算が困難でない**

… 落とし戸関数 (trapdoor function)

計算困難な問題を利用した暗号

暗号に必要な性質：

- **一般には、盗聴しても解読できない**
- **復号鍵を持っている人だけは、受信すると復号できる**

暗号化関数への要請：

- **そのものの計算はさほど困難でない**
- **逆関数の計算は一般に困難**
- **特定の情報を知っている場合だけは、逆関数の計算が困難でない**

· · · 落とし戸関数 (trapdoor function)

計算困難な問題を利用した暗号

落とし戸関数 (と思われる) 例： 乗法関数

$$f : \mathbf{N} \times \mathbf{N} \longrightarrow \mathbf{N}$$

$$(n_1, n_2) \longmapsto n_1 n_2$$

- そのものの計算はさほど困難でない
- 逆 (素因数分解) の計算は困難 (と思われる)
- 約数を知っていれば、簡単に分解できる

$\longrightarrow$ 約数 (と同等な情報) を秘密鍵として
暗号が作れる

計算困難な問題を利用した暗号

解読 (逆関数の計算) が困難
(手間・時間が掛かる) なことが
暗号の安全性の裏付けとなる

· · · 計算量的安全性

“計算の手間” はどのように計るのか

⟶ 計算量の理論

計算困難な問題を利用した暗号

解読 (逆関数の計算) が困難
(手間・時間が掛かる) なことが
暗号の安全性の裏付けとなる

$\cdots$ **計算量的安全性**

“計算の手間” はどのように計るのか

$\longrightarrow$ **計算量の理論**

代表的な公開鍵暗号方式

- **RSA 暗号 (Rivest-Shamir-Adleman)**
- **Diffie-Hellman 鍵共有**
- **ElGamal 暗号**

暗号を支える数学

主に、離散・有限の世界の数学

- 初等整数論 (素因数分解・合同式など)
- 群・環・体・加群などの代数の言葉
- 特に有限体とその上の線型代数・Galois 理論
- 利用する現象として、楕円曲線の理論など