Introduction to Cryptanalysis

Good Guys and Bad Guys

- Alice and Bob are the good guys

- Trudy is the bad guy

- Trudy is our generic “intruder”

Good Guys and Bad Guys

- Alice and Bob want to communicate securely
  - Typically, over a network
- Alice or Bob might also want to store their data securely
- Trudy wants to read Alice and Bob’s secrets
- Or Trudy might have other devious plans…
  - Cause confusion, denial of service, etc.

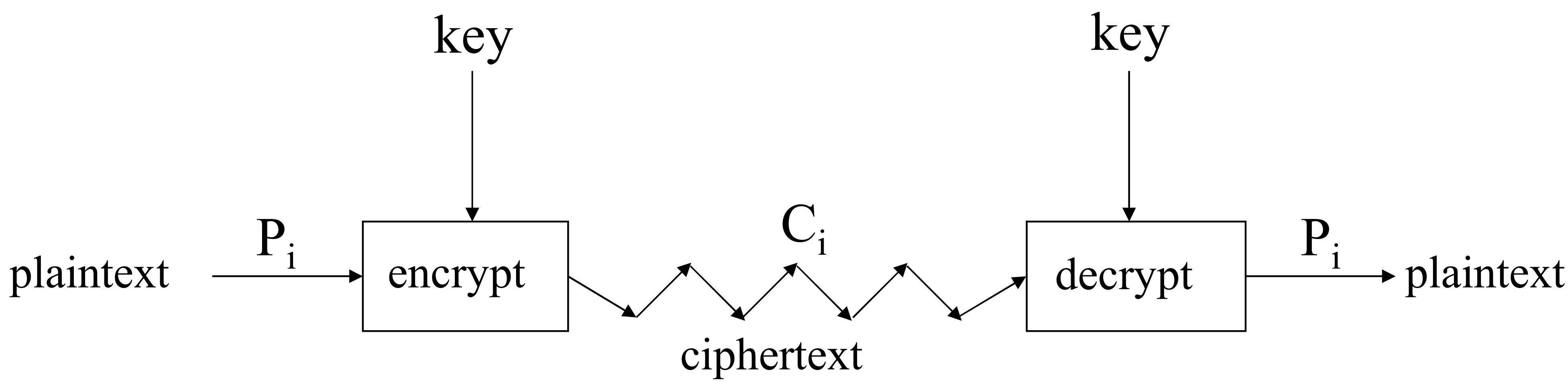
CIA

- **C**onfidentiality, **I**ntegrity and **A**vailability
- **Confidentiality:** prevent unauthorized reading of information
- **Integrity:** prevent unauthorized writing of information
- **Availability:** data is available in a timely manner when needed
  - Availability is a "new" security concern
  - Due to denial of service (DoS) threats

Crypto

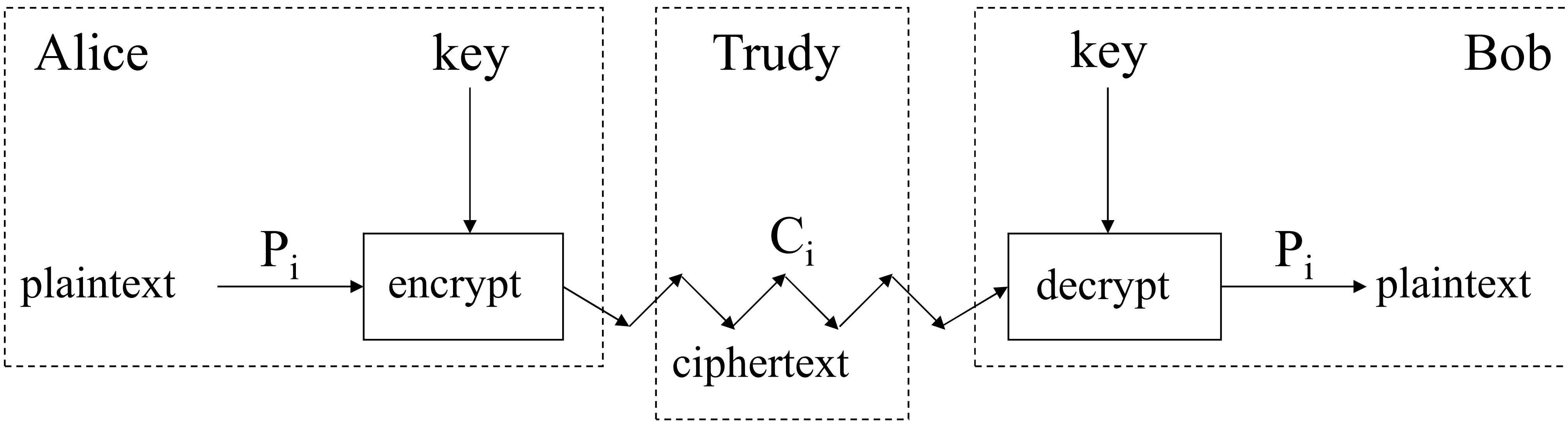
- **Cryptology** — The art and science of making and breaking "secret codes"
- **Cryptography** — making "secret codes"
- **Cryptanalysis** — breaking "secret codes"
- **Crypto** — all of the above (and more)

Crypto as a Black Box



- Note P_i is i^{th} "unit" of plaintext
- And C_i is corresponding ciphertext
- "Unit" may be bit, letter, block of bits, etc.

Who Knows What?



- Trudy knows the ciphertext
- Trudy knows the cipher and how it works
- Trudy might know a little more
- Trudy does **not** know the key

Cryptanalysis

- This course focused on cryptanalysis
- Trudy wants to recover key or plaintext
- Trudy is not bound by any rules
  - For example, Trudy might attack the implementation, not the algorithm itself
  - She might use "side channel" info, etc.

Attacking Block Ciphers

- Standard attacks
  - exhaustive key search
  - dictionary attack
  - differential cryptanalysis
  - linear cryptanalysis
- Side channel attacks against implementations.
  - Timing attacks
  - Power consumption attacks
  - Fault injection attacks

Exhaustive Key Search

- How can Trudy attack a cipher?
- She can simply try all possible keys and test each to see if it is correct
  - **Exhaustive key search**
- To prevent an exhaustive key search, a cryptosystem must have a large **keyspace**
  - Must be too many keys for Trudy to try them all in any reasonable amount of time

Beyond Exhaustive Search

- A large keyspace is necessary for security
- But a large keyspace is not sufficient
- Shortcut attacks might exist
- We'll see many examples of shortcut attacks
- In cryptography we can (almost) never prove that no shortcut attack exists
- This makes cryptography interesting…

Chosen-Plaintext Dictionary Attacks Against Block Ciphers

- Construct a table with the following entries
  - (K, $E_K[0]$) for all possible key K
  - Sort based on the second field (ciphertext)
  - How much time does this take?
- To attack a new key K (under chosen message attacks)
  - Choose 0, obtain the ciphertext C, looks up in the table, and finds the corresponding key
  - How much time does this step take?
- Trade off space for time

Differential Cryptanalysis

- Main idea:
  - This is a chosen plaintext attack,
  - The attacker knows many (plaintext, ciphertext) pairs
  - Difference $\Delta_P = P_1 \oplus P_2$, $\Delta_C = C_1 \oplus C_2$
  - Distribution of Δ_C's given Δ_P may reveal information about the key (certain key bits)
  - After finding several bits, use brute-force for the rest of the bits to find the key.

Basic idea of linear cryptanalysis

- Suppose that
- **(*) Pr** [$M_{i1} \oplus M_{i2} \oplus \ldots \oplus M_{iu}$
  $\oplus\, C_{j1} \oplus C_{j2} \oplus \ldots \oplus C_{jv}$
  $\oplus\, K_{p1} \oplus k_{p2} \oplus \ldots \oplus k_{pw} = 1$] $= 0.5 + \varepsilon$
- Then one can recover some key bits given large number of PT/CT pairs
- For DES, exists (*) with $\varepsilon=2^{-21}$
- Using this method, one can find 14 key bits using $(2^{21})^2$ PT/CT pairs

DES Strength Against Various Attacks

Attack Method	Known	Chosen	Storage complexity	Processing complexity
Exhaustive precomputation	-	1	2^{56}	1
Exhaustive search	1	-	negligible	2^{55}
Linear cryptanalysis	2^{43} 2^{38}	- -	For texts	2^{43} 2^{50}
Differential cryptanalysis	- 2^{55}	2^{47} -	For texts	2^{47} 2^{55}

The weakest point of DES remains the size of the key (56 bits)!

Taxonomy of Cryptanalysis

- Ciphertext only — always an option
- Known plaintext — possible in many cases
- Chosen plaintext
  - "Lunchtime attack"
  - Protocols might encrypt chosen text
- Adaptively chosen plaintext
- Related key
- Forward search (public key crypto only)
- "Rubber hose", bribery, etc., etc., etc.

Definition of Secure

- A cryptosystem is **secure** if the best know attack is to try all possible keys
- Cryptosystem is **insecure** if **any** shortcut attack is known
- By this definition, an insecure system might be harder to break than a secure system!

Definition of Secure

- Why do we define **secure** this way?
- The size of the keyspace is the "advertised" level of security
- If an attack requires less work, then false advertising
- A cipher must be secure (by our definition) and have a "large" keyspace
  - Too big for an exhaustive key search

Theoretical Cryptanalysis

- Suppose that a cipher has a 100 bit key
  - Then keyspace is of size 2^{100}
- On average, for exhaustive search Trudy tests $2^{100}/2 = 2^{99}$ keys
- Suppose Trudy can test 2^{30} keys/second
  - Then she can find the key in about 37.4 trillion years

Theoretical Cryptanalysis

- Suppose that a cipher has a 100 bit key
  - Then keyspace is of size 2^{100}
- Suppose there is a shortcut attack with "work" equal to testing about 2^{80} keys
- If Trudy can test 2^{30} per second
  - Then she finds key in 36 million years
  - Better than 37 trillion, but not practical

Applied Cryptanalysis

- In this class, we focus on attacks that produce plaintext
  - Not interested in attacks that just show a theoretical weakness in a cipher
- We call this **applied cryptanalysis**
- Why applied cryptanalysis?
  - Because it’s a lot more fun…
  - And it’s a good place to start

Applied Cryptanalysis: Overview

- Classic (pen and paper) ciphers
  - Transposition, substitution, etc.
  - Same principles appear in later sections
- World War II ciphers
  - Enigma, Purple, Sigaba
- Stream ciphers
  - Shift registers, correlation attack, ORYX, RC4, PKZIP

Applied Cryptanalysis: Overview

- Block ciphers
  - Hellman's TMTO, CMEA, Akelarre, FEAL
- Hash functions
  - Nostradamus attack, MD4, MD5
- Public key crypto
  - Knapsack, Diffie-Hellman, Arithmetica, RSA, Rabin, NTRU, ElGamal
  - Factoring, discrete log, timing, glitching

Side Channel Analysis

- Time
  - Does the number of CPU cycles depend on exact values used in the operation? ex. RSA exponent
  - Memory access – do exact values impact tables used, time to read from a table and/or number of memory accesses? ex. AES using tables of 32-bit values
- Acoustics
  - Impacted by operations or exact values used?
- Memory
  - Can intermediate values be read from memory by another process?

Timing – Toy Example

k: array of n key bytes

d: 16 byte data

Suppose encryption is a series of n rounds

```
n = 16;
d = plaintext;
for (i=0; i < n; ++i) {
   d = f(d,k[i]);  // do something to the data with k, but
                   // whose time does not depend on k
   d[i] = d[i] int(k[i]) mod 256; // alter one byte, time depends on k
}
```

Timing – Toy Example

What if use a table lookup instead?

table(a,b): function retrieves table a, entry b

```
d = plaintext;
x = 0;
for (i=0; i < n; ++i) {
  // do something to the data with k where time does not depend on k
     d = f(d,k[i]);
  // memory lookup - was table already in cache?
  // (k[i] same as a previous key byte)
     x= table(k[i], d[i]);
}
```

Why Study Cryptography?

- Information security is a big topic
  - Crypto, Access control, Protocols, Software
  - Real world info security problems abound
- Cryptography is the part of information security that works best
- Using crypto correctly is important
- The more we make other parts of security behave like crypto, the better

Why Study Cryptanalysis?

- Study of cryptanalysis gives insight into all aspects of crypto
- Gain insight into attacker's mindset
  - "black hat" vs "white hat" mentality
- Cryptanalysis is more fun than cryptography
  - Cryptographers are boring
  - Cryptanalysts are cool
- But cryptanalysis is hard