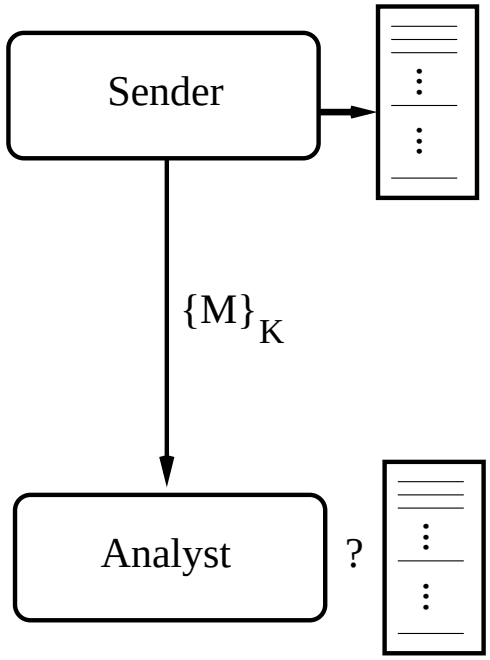
Foundations of Computer Security

Lecture 42: A Perfect Cipher

Dr. Bill Young
Department of Computer Sciences
University of Texas at Austin

A Perfect Cipher



A *perfect cipher* would be one for which no reduction of the search space is gained from knowing:

1. the encryption algorithm, and
2. the ciphertext.

Shannon proved that a perfect cipher requires as many possible keys as plaintexts, with the key chosen randomly.

One Time Pad

A *one-time pad*, invented by Miller (1882) and independently by Vernam and Mauborgne (1917), is a theoretically perfect cipher.

The idea is to use a key that is the same length as the plaintext, and to use it only once. The key is XOR'd with the plaintext.

Example: Given a 15-bit binary message:

plaintext:	101100101111001
key:	110100010101000
ciphertext:	011000111011101

Notice the space of plaintexts, ciphertexts, and keys are all the same: 15-bit binary strings.

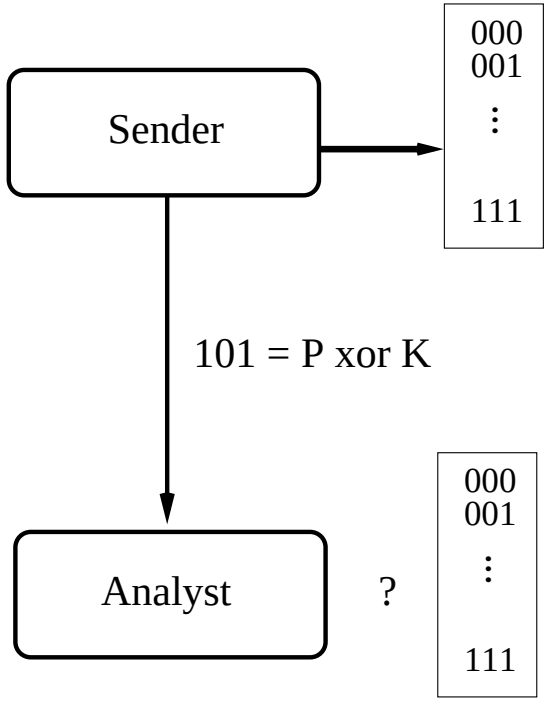
One Time Pad

Why is the one-time pad perfect? Consider the space of three-bit messages.

Suppose the attacker
intercepts the ciphertext ("101")
and knows that a one-time pad is in use.

Every possible plaintext
could be the pre-image of that ciphertext
under a plausible key. Therefore, no
reduction of the search space is possible.

Why does it matter that the key be random?



Key Distribution

The main problem with the one-time pad is practical, rather than theoretical.

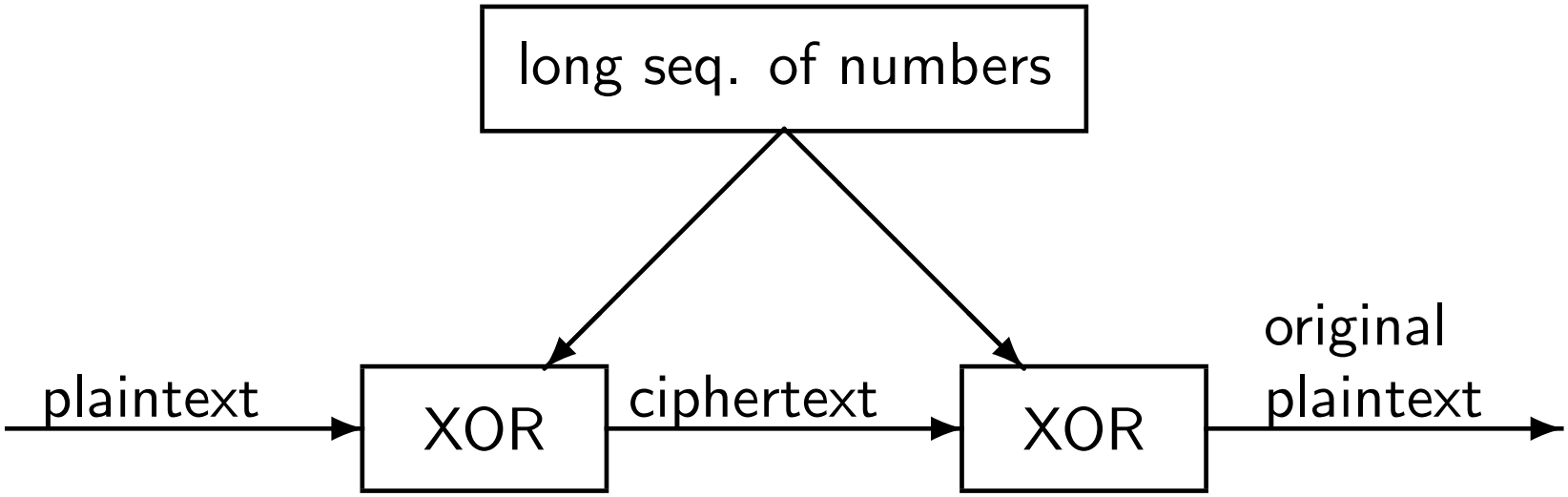
Given the need to communicate securely, how do the sender and receiver agree on a *secret* (key) that they can use in the algorithm.

- If sender and receiver already have a secure channel, why do they need the key?
- If they don't, how do they distribute the key securely?

This is the *key distribution* problem.

Vernam Cipher

The *Vernam cipher* is a type of one-time pad suitable for use on computers.



One Time Pad Approximation

Approximate the one-time pad using a PRNG to generate a key.

Another computer running the same random number generator function can produce the key from the *seed*. This works well because a pseudorandom sequence may have a very long *period*.

It is susceptible to compromise by someone who knows the algorithm and the seed.

Lessons

- The one-time pad is a theoretically perfect encryption algorithm.
- However, it requires as much key material as there is plaintext, and suffers from the key distribution problem.
- An approximation suitable for computers uses a PRNG to generate a seed.

Next lecture: Transposition Ciphers