Introduction to Cryptology

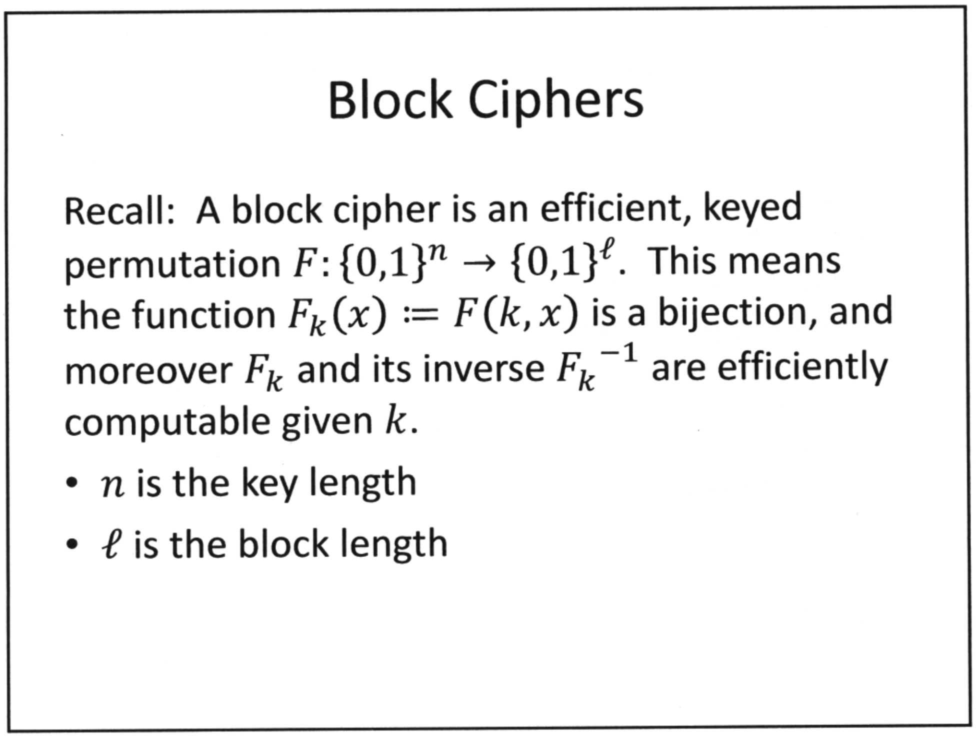
Lecture 15

Announcements

- HW6 postponed to Thursday, 3/31
- EC opportunity
  - Due on Thursday, May 5
  - Maximum of 2 people signed up per paper

Agenda

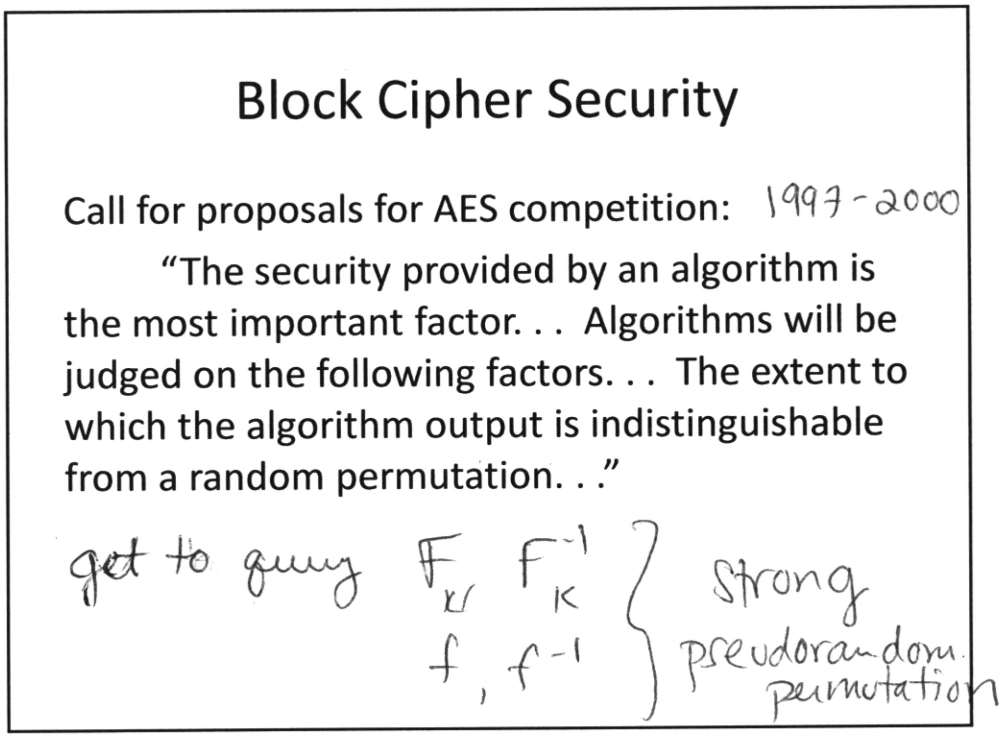
- Last time
  - Collision-Resistant Hash Functions (5.1)
  - Domain Extension (Merkle-Damgard) (5.2)
- This time
  - MACs from CRHF (5.3)
    - Hash-and-Mac
    - HMAC
  - New topic: Practical constructions of Block Ciphers



Block Ciphers

Recall: A block cipher is an efficient, keyed permutation $F: \{0,1\}^n \rightarrow \{0,1\}^\ell$. This means the function $F_k(x) := F(k, x)$ is a bijection, and moreover F_k and its inverse ${F_k}^{-1}$ are efficiently computable given k.

- n is the key length
- ℓ is the block length



Block Cipher Security

Call for proposals for AES competition: 1997 - 2000

"The security provided by an algorithm is the most important factor. . . Algorithms will be judged on the following factors. . . The extent to which the algorithm output is indistinguishable from a random permutation. . ."

$$\text{get to query } \left.\begin{matrix} F_k, F_k^{-1} \\ f, f^{-1} \end{matrix}\right\} \text{ strong pseudorandom permutation}$$

First Idea

- Random permutations over small domains are "efficient."
  - What does this mean?
- First attempt to define F_k:
  - The key k for F will specify 16 permutations $f_1, \dots, f_{16}$ that each have an 8-bit block length.
  - Given an input $x \in \{0,1\}^{128}$, parse it as 16 bytes $x_1, \dots, x_{16}$ and then set
$$F_k(x) = f_1(x_1) || \cdots || f_{16}(x_{16})$$
- Is this a permutation?
- Is this indistinguishable from a random permutation?

Shannon's Confusion-Diffusion Paradigm

Above step is called the "confusion" step. Is combined with a "diffusion" step: the bits of the output are permuted or "mixed," using a mixing permutation.

- Confusion/Diffusion steps taken together are called a round.
- Multiple rounds required for a secure block cipher.

Example: First compute intermediate value $\mathrm{y} = f_1(x_1) || \cdots || f_{16}(x_{16})$. Then permute the bits of y.

Substitution-Permutation Network (SPN)

In practice, round-functions are not random permutations, since it would be difficult to implement this in practice.

- Why?

Instead, round functions have a specific form:

- Rather than having a portion of the key k specify an arbitrary permutation f, we instead fix a public "substitution function" (i.e. permutation) S, called an S-box.
- Let k define the function f given by $f(x) = S(k \oplus x)$.

Informal Description of SPN

1. Key mixing: Set $x := x \oplus k$, where k is the current-round sub-key.
2. Substitution: Set $x := S_1(x_1)||\cdots||S_8(x_8)$, where x_i is the i-th byte of x.
3. Permutation: Permute the bits of x to obtain the output of the round.
4. Final mixing step: After the last round there is a final key-mixing step. The result is the output of the cipher.
   - Why is this needed?

- Different sub-keys (round keys) are used in each round.
  - Master key is used to derive round sub-keys according to a key schedule.

Formal description of SPN

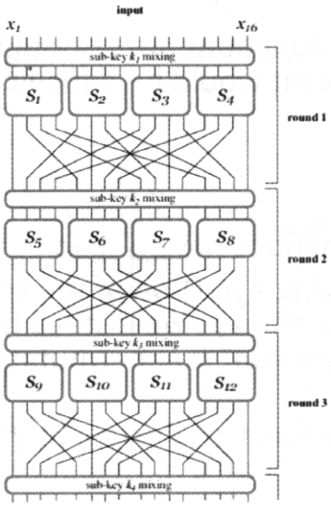


FIGURE 6.2: A substitution-permutation network.

SPN is a permutation

Proposition: Let F be a keyed function defined by an SPN in which the S-boxes are all permutations. Then regardless of the key schedule and the number of rounds, F_k is a permutation for any k.