

Designing Secure Hash Functions and Block Ciphers

Yevgeniy Dodis, NYU

<http://www.cs.nyu.edu/~dodis/>



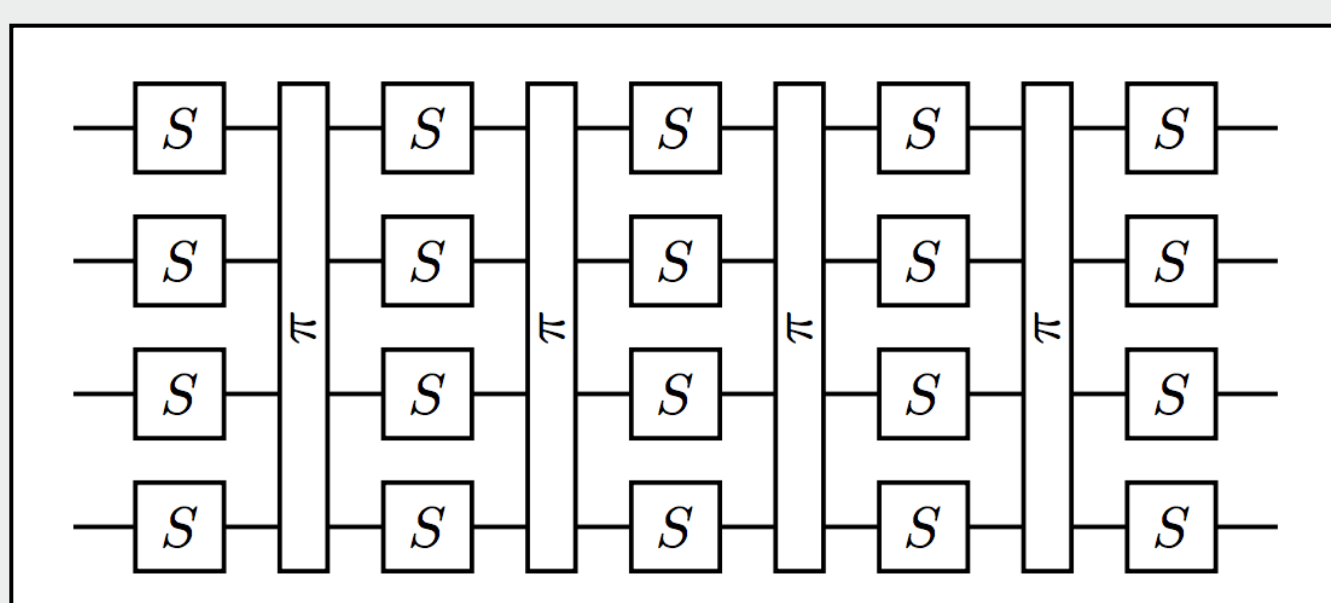
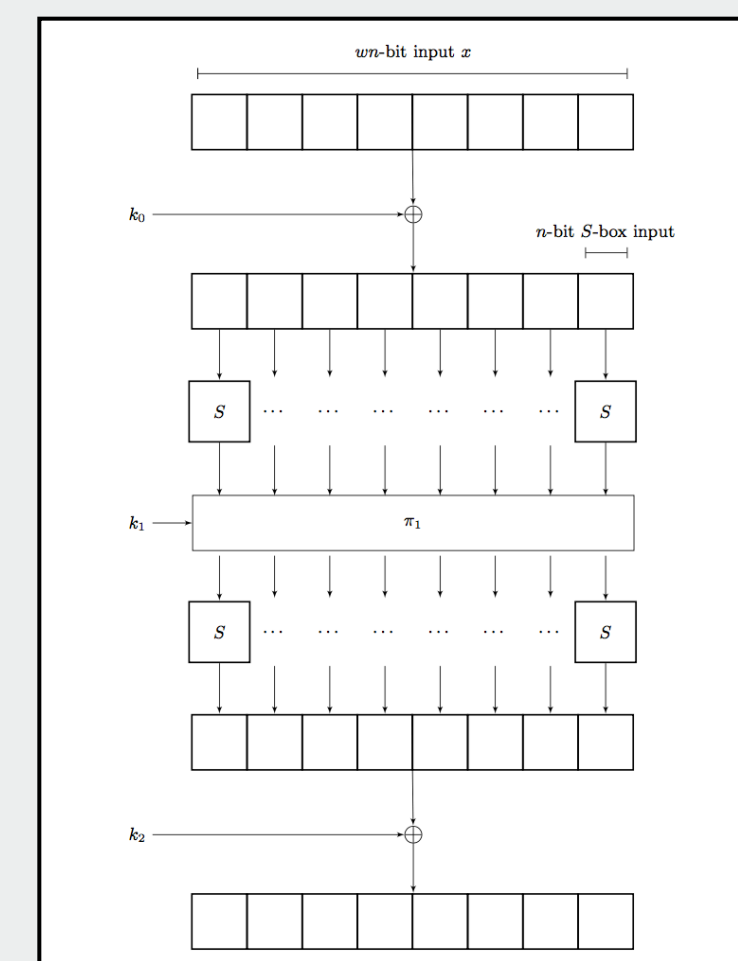
Hash functions and **block ciphers** are used in almost any cryptographic application. Yet, many such practical applications often do not have sufficient theoretical foundations for several related reasons: an application uses a given primitive in an **unforeseen manner**; an application makes **incorrect assumptions** about a primitive; a primitive has **unexpected weaknesses**; a primitive contains a **backdoor**. Hence, there is renewed interest and urgency to study the basic design principles of hash functions, as well as how such hash functions should be appropriately used in applications.

Indifferentiability

- Standard security notion for hash functions and block ciphers
- Allows designing cryptographic primitives from simple ideal components
- Guarantees security even if attacker has access to underlying components

Substitution-Permutation Networks

- Substitution-permutation networks (SPNs) are used in many modern block ciphers
- So far almost no provable security
- **Result:** 3-round linear SPN network is random permutation and optimal

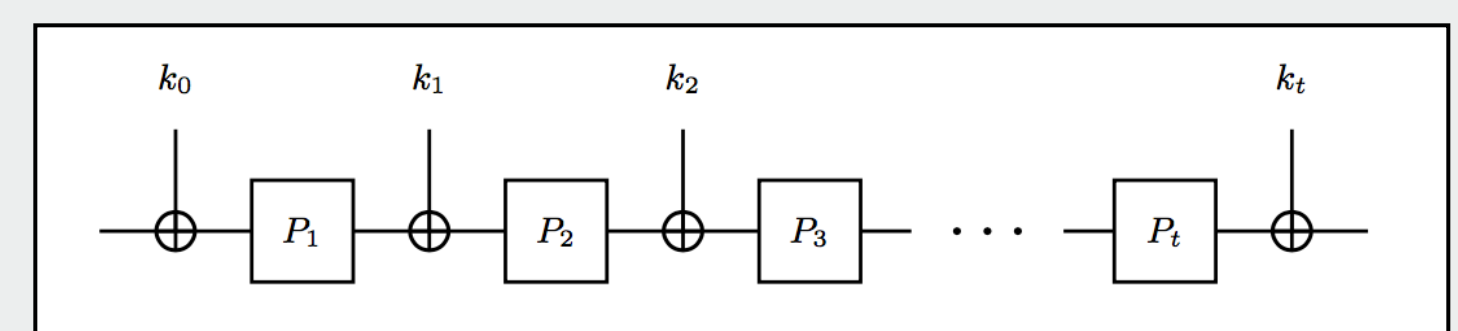


Confusion-Diffusion Networks

- Used extensively in design of hash functions and block ciphers
- **Result:** constant number of rounds sufficient for domain extension of public random function

Key-Alternating Ciphers

- Abstraction of AES and generalization of Even-Mansour cipher
- **Result:** 5-round key-alternating cipher is indistinguishable from ideal cipher



Backdoored PRGs

- NIST's Dual EC PRG backdoored
- **Results:** show both how to build backdoored PRGs and how to immunize PRGs against backdoors

H² and HMAC

- **Results:** hashing twice (H²) insecure; prove security of deployed HMAC applications in indifferentiability framework

Interested in meeting the PIs? Attach post-it note below!

