Notes for January 31st

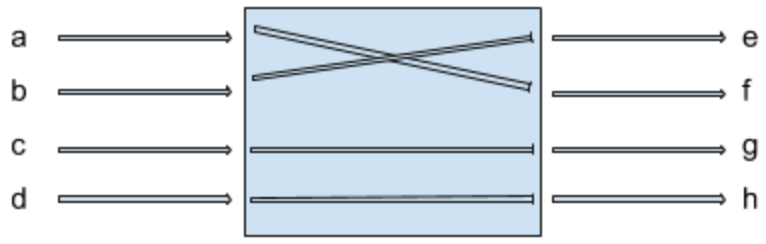
Block Ciphers

- Obtaining a Pseudo Random Permutation
  - Permutation: to change the order of
    - Permutation of n-bit inputs: 2^n
      - 0 or 1 at every bit
    - How many possible permutations of 2^n?
      - 2^n! Big, big number.
  - Random Permutation:
    - Take space of all permutations and pick one. Randomly pick one.

What it means to have a secure encryption scheme.

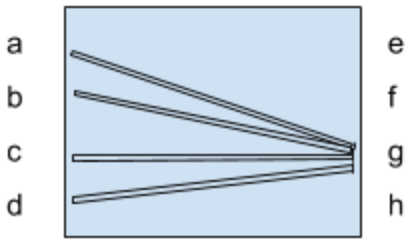
Encryption schemes must have correctness.

- Correctness: $Dec_k(Enc_k(m)) = m$
- To get correctness property:
  - $RandomPerm^{-1}(RandomPerm(m))$
    "Decryption" "Encryption"



Every input corresponds to a unique output.

Permutation



It's possible for two inputs to get the same output.

Function

Use permutations, not functions!

How many random permutations of n-bit inputs? $2^n!$

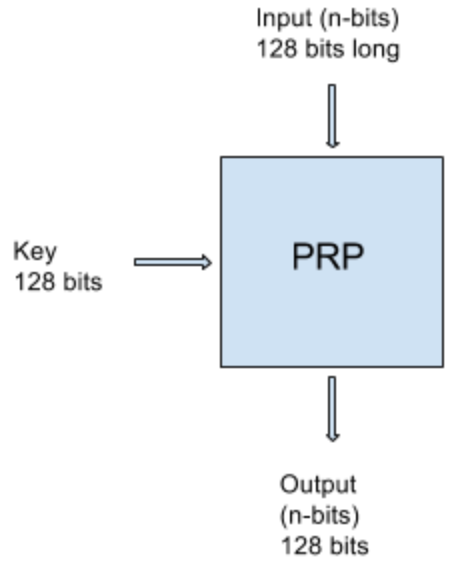
Writing down that permutation would use that many bits.

You need $\log_2(2^n!)$ bits → to represent how many bits chosen.

- $O(2^n)$

Pseudo Random Permutations

… not actually random.



Advanced Encryption Scheme/System.
Algorithm specified by NIST.
AES is a PRP.

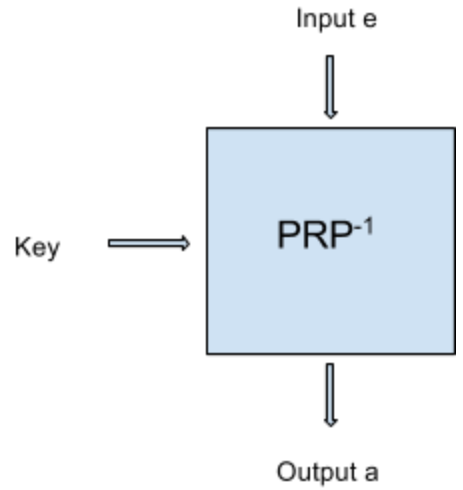
Input bit length must match output bit length.
Key is secret.

If key is unknown, but inputs/outputs are known, adversary wouldn't know if output came from AES or PRP.

We believe AES is a PRP. We do AES everywhere. It would be terrifying if there was a flaw in it.

2^{128} is a big number. Called 128-bit security level.

Forward/Backwards direction



Ciphertext = PRP_k(plaintext) ← Encryption scheme for n-bit messages

$c = Enc_k(m) = PRP_k(m)$

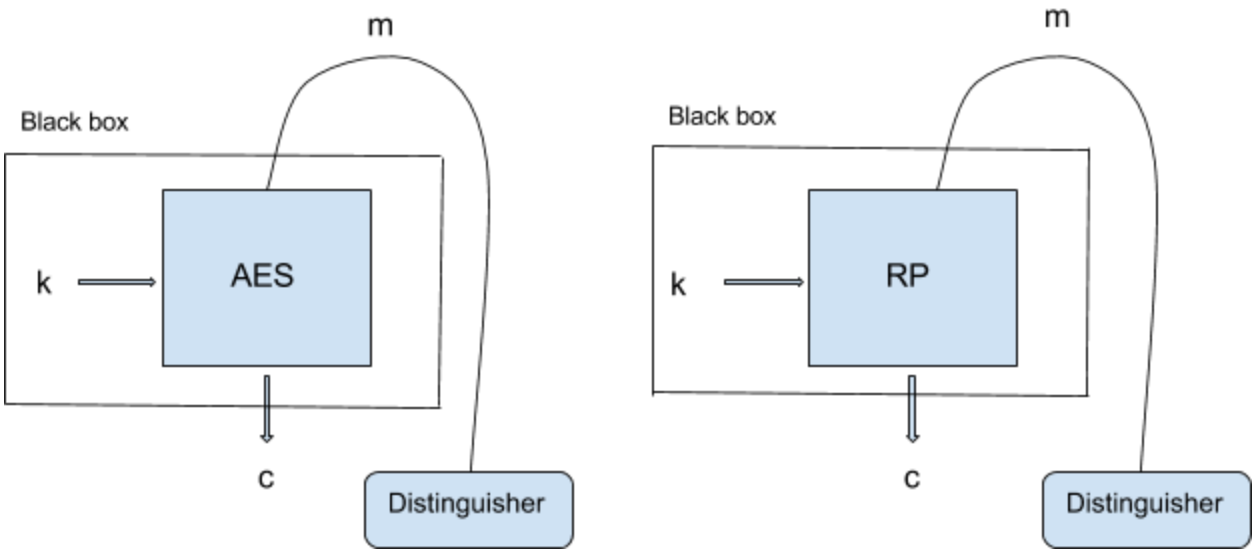
If key is 4 bits long, then it's no good. You can try all possibilities.
Say a key was 4 bits long, you can trivially break encryption scheme.
Attack: for all possible keys ← $2^4 = 16$

- Compute $PRP^{-1}_k(c)$, stop when you find "valid looking" message.
- Stupid attack! Don't want this scheme.

If keys are short, you will never be secure.
Want a number so large that no computers could compute it.
Most keys on internet are <u>at least</u> 128 bits long.

$2^n!$ Possible permutations ~ $(2^n)^n$
Need $\log_2(2^n)^n \sim n2^n$ bits to represent choice of random permutation.

In this example, a distinguisher knows the input and output of some magical black box but does not know the key.



Distinguisher cannot tell if blackbox is an AES or RP!
We haven't broken AES, but that doesn't mean it's PRP.

What happens with a 2n-bit message? 100n-bit?

Encrypted Codebook Mode! (So insecure!)

- It's just a substitution cipher.

For a 100n-bit message

- Chop up into n bits, so there are 100 blocks, and run PRP on each block.
- Then do frequency analysis.

Here is a wiki link that breaks it down:
https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation#Electronic_Codebook_.28ECB.29

Cipher Block Chaining (CBC)

- Take plaintext and break into blocks.
- Have initialization vector (IV) (n bits of randomness), which does not need to be secret since it's part of the ciphertext, and XOR the two, bit by bit.
- Put that input (n bits) through block cipher encryption with key and get ciphertext. Use that ciphertext and XOR it with next plaintext, and so on.
- This is still used today.

AES CBC mode, AES is the block cipher encryption and CBC is how it's used.
CBC is not vulnerable to length-extension.

Here is a wiki link that breaks down how to encrypt and decrypt CBC:
https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation#Cipher_Block_Chaining_.28CBC.29

Counter Mode (CTR mode) → similar to OTP

Nonce = fresh (random) number
n-bit number should not repeat.
Nonce is first block. It is also part of the cipher.

- Take counter (starts with nonce). Counter starts with 0.
- Put it through block cipher encryption with key,
- XOR it with plaintext, and get ciphertext.
- Do this with the next counter (same nonce – it's fresh for each block).

After BCE (block cipher encryption), get keystream. Very similar to OTP. Key is secret.
How to decrypt?

- Swap plaintext and ciphertext.
- Decrypting doesn't use PRP^{-1} (the backwards direction).
- This is a parallel algorithm.
- It's sequential.

Here is a wiki link that breaks it down:
https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation#Counter_.28CTR.29