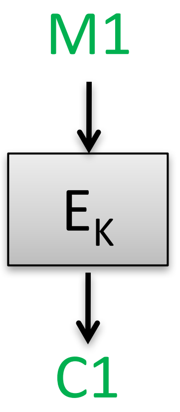
Today in Cryptography (5830)

Recap of block ciphers, Feistel
Length-preserving encryption
Blockcipher modes of operation

Recap: Block ciphers & Feistel



Block cipher is a map $E : \{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$

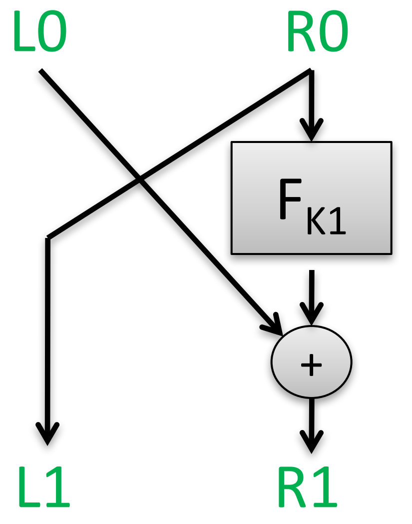
Each key K defines permutation $E_K : \{0,1\}^n \rightarrow \{0,1\}^n$

Permutation: 1-1, onto

Inverse D_K : $D_K(E_K(M)) = M$

E and D must be efficiently computable

Should behave like random permutation when K secret



Feistel networks turn function into permutation.

- Used in DES with specialized round function F

Best attacks against DES

Attack	Attack type	Complexity	Year
Biham, Shamir	Chosen plaintexts, recovers key	2^{47} plaintext, ciphertext pairs	1992
DESCHALL	Brute-force attack	$2^{56/4}$ DES computations 41 days	1997
EFF Deepcrack	Brute-force attack	~4.5 days	1998
Deepcrack + DESCHALL	Brute-force attack	22 hours	1999

- DES is still used in some places
- 3DES (use DES 3 times in a row with more keys) expands keyspace and still used widely in practice

Advanced Encryption Standard (AES)

Rijndael (Rijmen and Daemen)

$n = 128$
$k = 128, 192, 256$

Number of keys for k=128:
340,282,366,920,938,463,463,374,607,431,768,211,456

Substitution-permutation design.
For k=128 uses 10 rounds of:

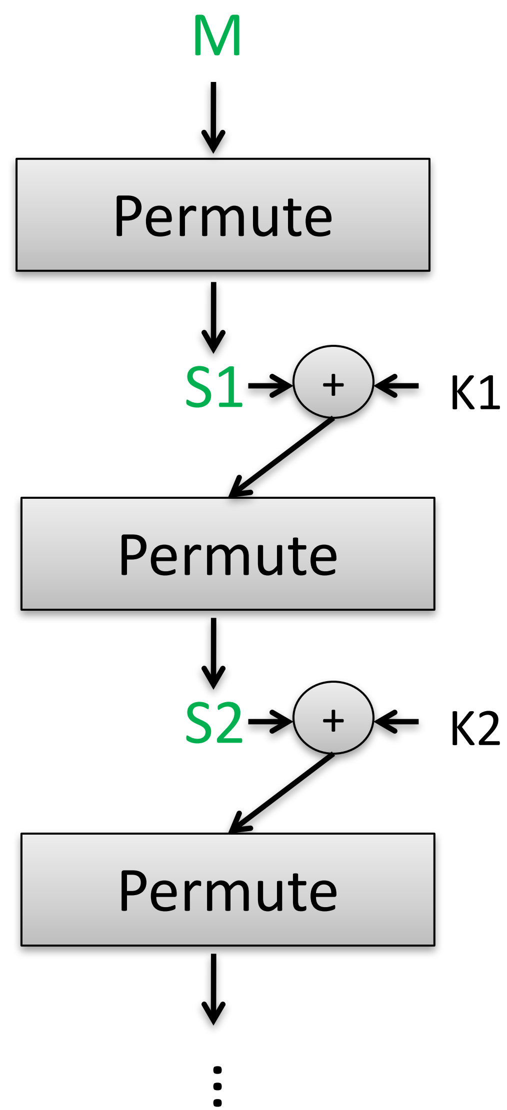
1) Permute:

SubBytes (non-linear S-boxes)

ShiftRows + MixCols (invertible linear transform)

2) XOR in a round key derived from K

(Actually last round skips MixCols)



Best attacks against AES

Brute-force attack (try all keys): worst case time about 2^{128}

Attack	Attack type	Complexity	Year
Bogdanov, Khovratovich, Rechberger	chosen ciphertext, recovers key	$2^{126.1}$ time + some data overheads	2011

No direct attacks of practical interest known

Effective side-channel attacks do exist,
need to implement very carefully
OpenSSL (underlying cryptography.io) does pretty good job

Applications of block ciphers (sometimes called modes of operation)

Let's assume we have a secure block cipher.

- **Length-preserving encryption**
  - Useful for cases where ciphertexts must be same length as plaintexts.
  - Should only be used when absolutely needed

- **Length-extending encryption**
  - Insecure variants: CTR mode, ECB mode, CBC mode
  - We'll build secure ones in a few lectures

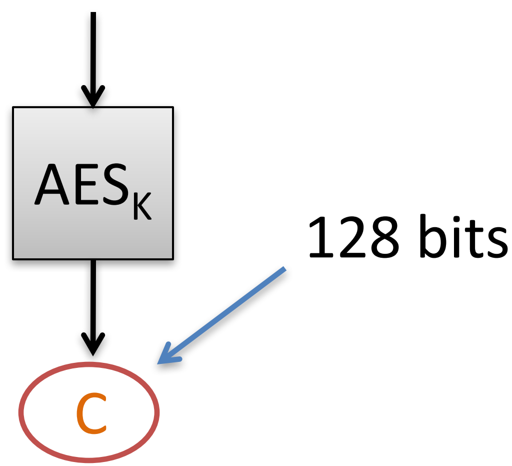
Example: Credit card number encryption

Jane Doe	1343-1321-1231-2310
Thomas Ristenpart	9541-3156-1320-2139
John Jones	5616-2341-2341-1210
Eve Judas	2321-4232-1340-1410

Database schemas and software require <= 16 decimal digits

$AES_K : \{0,1\}^{128} \longrightarrow \{0,1\}^{128}$

M = 2321-4232-1345-1415



Ciphertexts are too big for replacing plaintext within database!

16-digit restriction limits to $10^{16} \approx 2^{50}$ values

$2^{50} \ll 2^{128}$

Example: Credit card number encryption

Jane Doe	1343-1321-1231-2310
Thomas Ristenpart	9541-3156-1320-2139
John Jones	5616-2341-2341-1210
Eve Judas	2321-4232-1340-1410

Database schemas and software require <= 16 decimal digits

46 million credit card accounts stolen

>100 million credit card accounts stolen

>355,000 million credit card accounts stolen

Example: Credit card number encryption

Jane Doe	
Thomas Ristenpart	
John Jones	
Eve Judas	

Database schemas and software require <= 16 decimal digits

Encryption tool whose ciphertexts are also credit-card numbers

$$E_K : [0..9]^{16} \longrightarrow [0..9]^{16}$$

Valid credit-card number

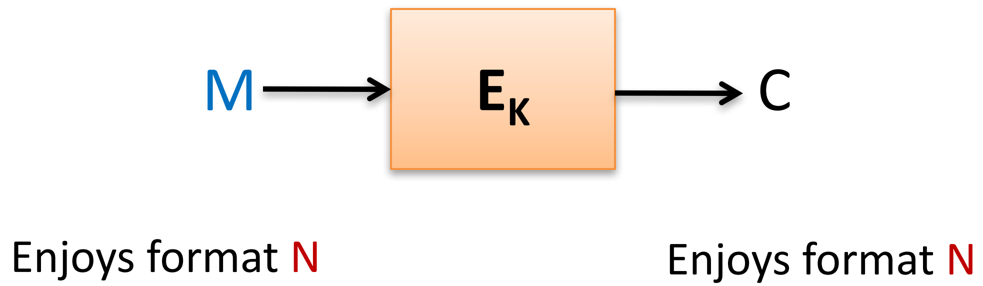
↓

$\mathbf{E}_K$

↓

Valid credit-card number

Format-preserving encryption (FPE)



Disk sectors / payment card numbers just two examples
Some others:

1) Valid addresses for a certain country

2) 4096-byte disk sectors

3) Assigned Social Security Numbers (9 digits, without leading 8 or 9)

4) Composition of (1) and (3)

How to build FPE on 48 bits?

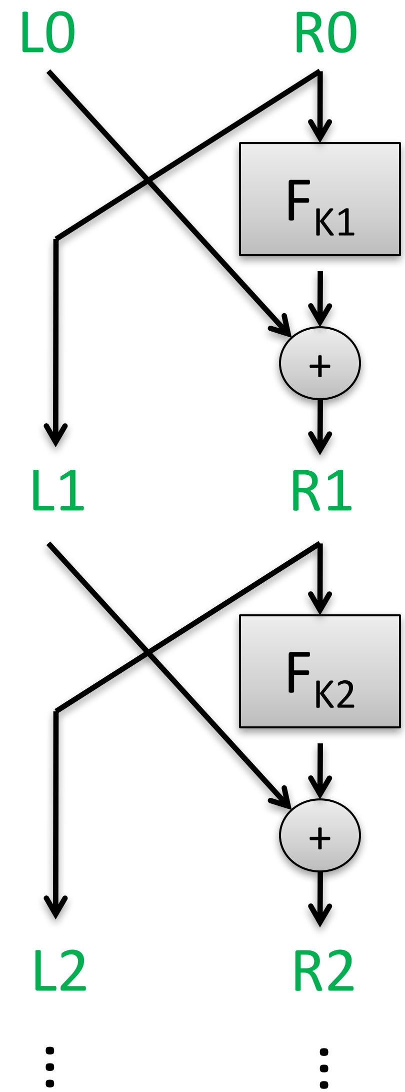
Simplification of FFX encryption

Input M = 48 bits
L0 = 24 bits
R0 = 24 bits

$F_{K1}(R) = AES(K, 1 \ || \ R)$
$F_{K2}(R) = AES(K, 2 \ || \ R)$
...

XOR uses low 24 bits of F output

Use 10 rounds



Balanced Feistel security

- Luby & Rackoff showed that if round functions are random and n is relatively large, then
  - 3 rounds suffice for chosen-plaintext attack security in sense of pseudorandom permutation
  - 4 rounds suffice for chosen-ciphertext attack security pseudorandom permutation
  - Proofs hold up to $q \approx 2^{n/4}$
- Sometimes n is not very large:
  - FFX designers suggested 10 rounds as heuristic

FPE now widely used in practice

Voltage security

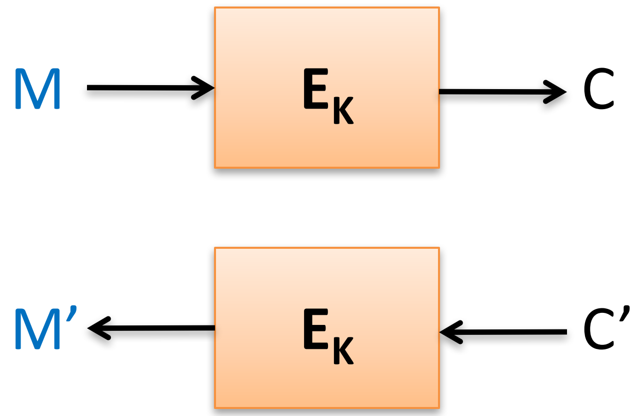
Heartland Payment Systems

Fifth Third Processing Solutions

skyhigh

CipherCloud
Trust in the Cloud™

Security problems with length-preserving encryption?



Looks like a random string (subject to permutivity)

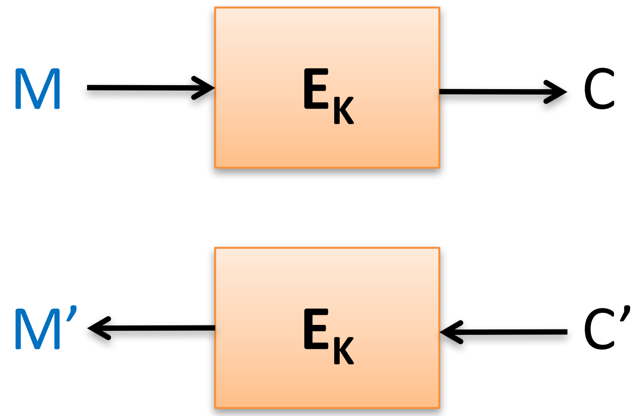
Looks like a random string (subject to permutivity)

Any C’ ≠ C

But determinism has problems:

	Plaintext	Ciphertext
Jane Doe	1343-1321-1231-2310	1049-9310-3210-4732
Thomas Ristenpart	9541-3156-1320-2139	7180-4315-4839-0142
John Jones	2321-4232-1340-1410	5731-8943-1483-9015
Eve Judas	1343-1321-1231-2310	1049-9310-3210-4732

Security problems with length-preserving encryption?



Looks like a random string (subject to permutivity)

Looks like a random string (subject to permutivity)

Any C' ≠ C

But determinism has problems:

	Plaintext	Ciphertext
Jane Doe	1343-1321-1231-2310	1049-9310-3210-4732
Thomas Ristenpart	9541-3156-1320-2139	7180-4315-4839-0142
John Jones	2321-4232-1340-1410	5731-8943-1483-9015
Eve Judas	1343-1321-1231-2310	1049-9310-3210-4732

Length-extending encryption security

- Not a bit of information about plaintext leaked
  - Equality of plaintexts hidden
  - Even in case of active attacks
    - Padding oracles we will see later
- Eventually: authenticity of messages as well
  - Decryption should reject modified ciphertexts

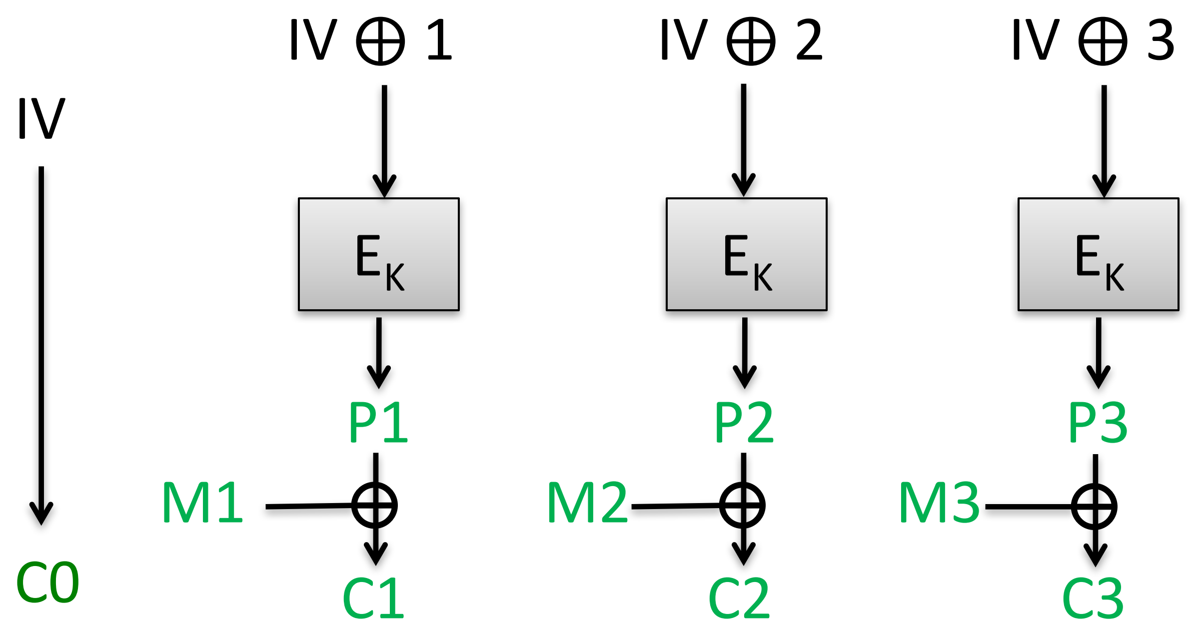
CTR[E] mode encryption using block cipher E

Counter mode (CTR)

Pad message M to M1,M2,M3,... where each is n bits except last

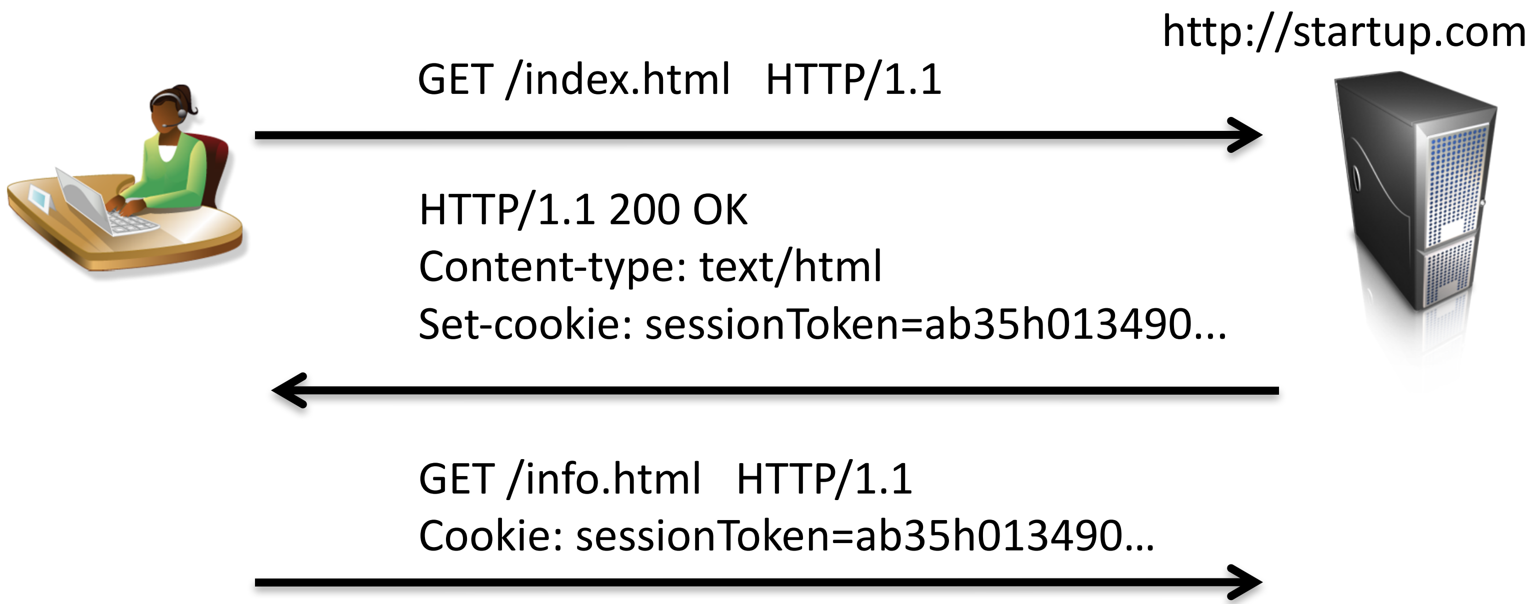
Choose random n-bit string IV

Then:



Maybe use less than full n bits of P3

Malleability example: Encrypted cookies



abc35h013490… = CTR[E](K, “admin=0”)

Malicious client can simply flip a few bits to change admin=1

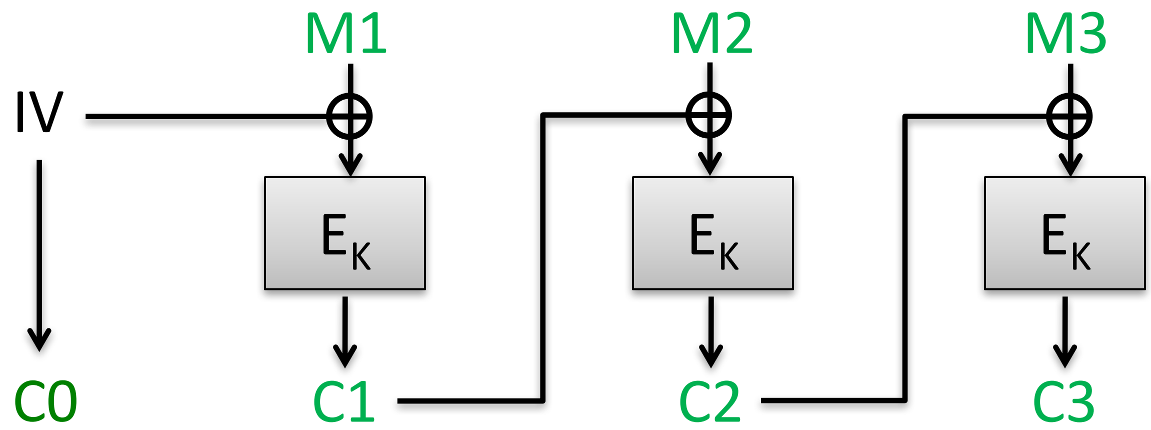
CBC mode

Ciphertext block chaining (CBC)

Pad message M to M1,M2,M3,... where each block Mi is n bits

Choose random n-bit string IV

Then:



How do we decrypt?

CBC-mode SE scheme

Kg():
$K \leftarrow\$ \{0,1\}^k$

Pick a random key

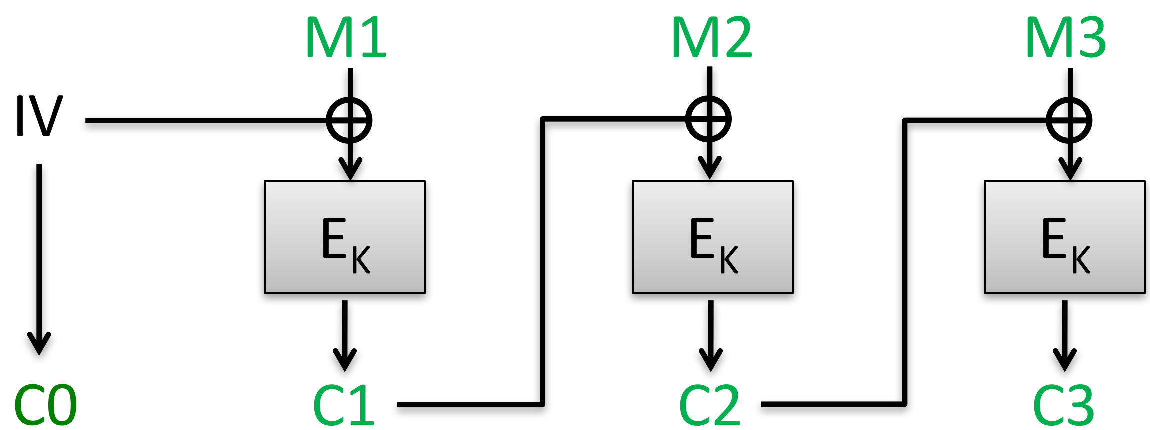
Enc(K,M):
$L \leftarrow |M|$; $m <= \text{ceil}(L/n)$
$C_0 \leftarrow IV \leftarrow\$ \{0,1\}^n$
$M_1,\ldots,M_m \leftarrow \text{PadCBC}(M,n)$
For $i = 1$ to m do
    $C_i \leftarrow E_K(C_{i-1} \oplus M_i)$
Return $(C_0, C_1, \ldots, C_m)$

PadCBC unambiguously pads M to a string of mn bits

Dec($K,(C_0, C_1, \ldots, C_m)$):
For $i = 1$ to m do
    $M_i \leftarrow C_{i-1} \oplus D_K(C_i)$
$M \leftarrow \text{UnpadCBC}(M_1,\ldots,M_m,n)$
Return M

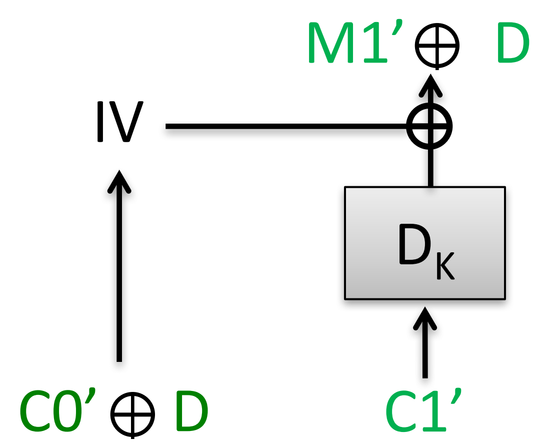
UnpadCBC removes padding, returns appropriately lengthed string

CBC mode also has "malleability" issues



How do we change bits of M1 received by server??

For any D:



Padding for CBC mode

- CBC mode handles messages with length a multiple of n bits
- We use padding to make it work for arbitrary encryption schemes

- Padding checks often give rise to padding oracle attacks (next lecture)

Summary

- We have good blockciphers
- You can use Feistel to help build length-preserving encryption out of AES, DES
- Length-preserving encryption leaks message equality
- CTR mode, CBC mode (being randomized) do not leak message equality, but are *malleable*