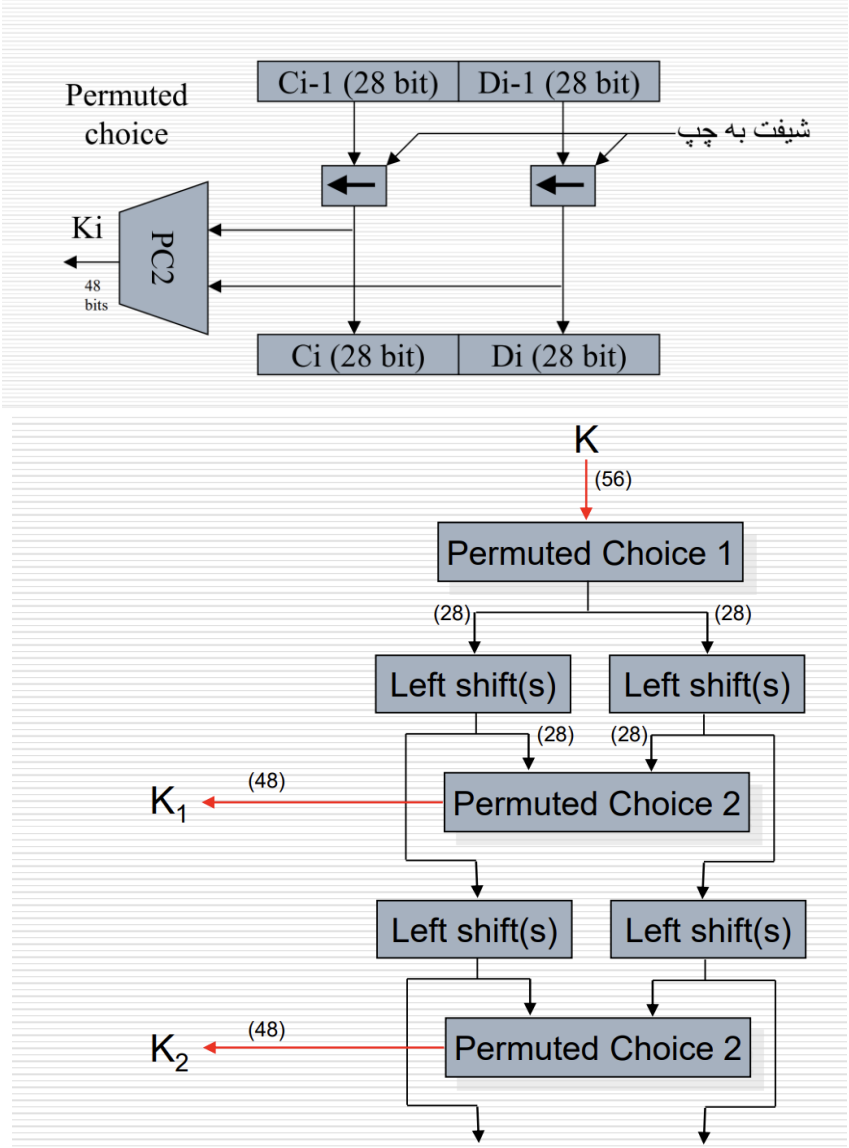
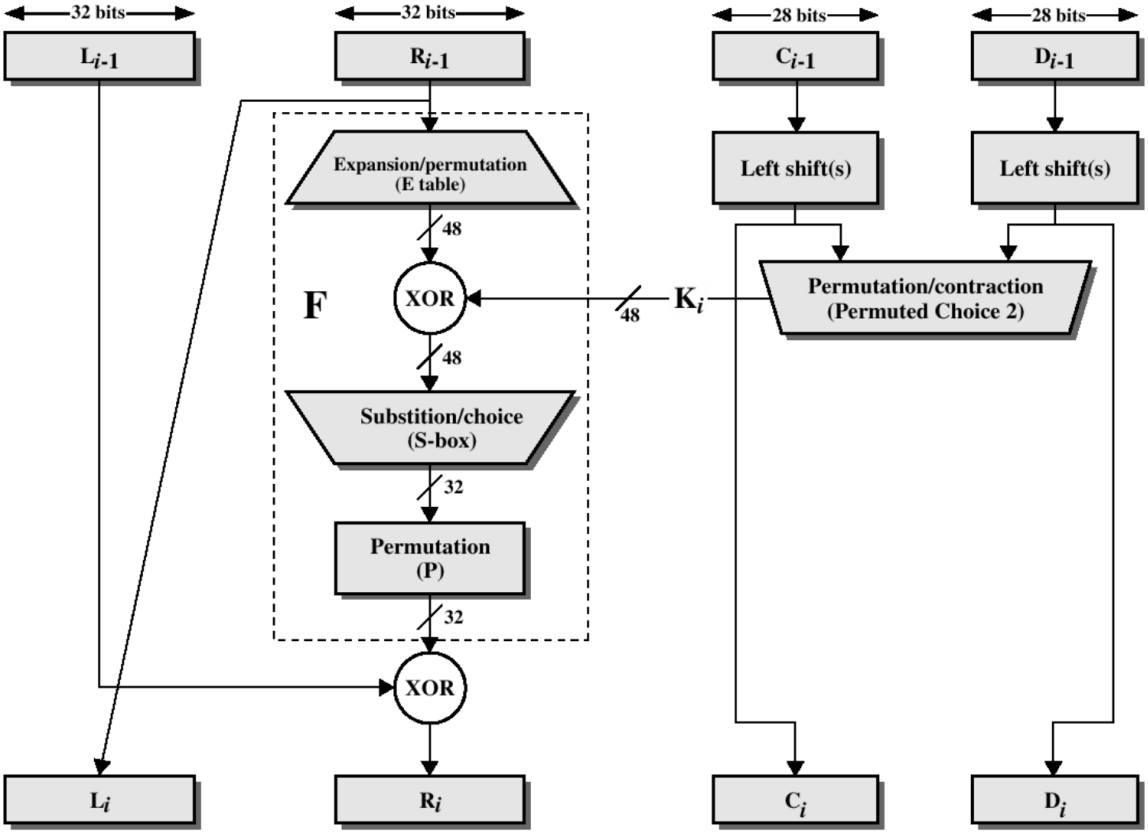
(1

در هر دور از الگوریتم DES ابتدا ورودی 64 بیتی ما به دو قسمت تقسیم شده و 32 بیت سمت راست به صورت مستقیم در 32 بیت سمت چپ دور بعد به کار گرفته می شود و 32 بیت سمت راست ابتدا با کلید ایجاد شده در هر دور رمزنگاری میشود و سپس با 32 بیت سمت چپ xor میشود و به عنوان ورودی برای 32 بیت سمت راست در دور بعد استفاده میشود.

برای ایجاد کلید هم به صورت زیر در هر دور کلید به دو قسمت 28 بیتی تقسیم شده شیفت به چپ انجام می شود و با استفاده از تابع permuted choice کلید برای هر مرحله انتخاب می شود و همان کلید های شیفت داده شده برای دور بعد هم انتخاب می شود.



نمودار رمزنگاری در DES:



(2

عملیات XOR که در DES استفاده میشود خاصیت زیر را دارد:

$$A \oplus B = \bar{A} \oplus \bar{B}$$

که نشان دهنده این است که خاصیت مکمل را دارد و تمام عملیات های دیگر مانند XOR,permutation و جایگزینی است که همه آنها نسبت به مکمل پذیری تقارن دارند

(3

-1

در مد کاری CBC ابتدا در اولین مرحله یک عدد رندم به نام IV با متن XOR میشود و سپس کلید خصوصی با خروجی XOR رمز میشود و سپس متن رمز شده به عنوان ورودی XOR در مرحله بعدی استفاده میشود.

در **ECB**، هر بلوک داده به‌طور مستقل رمزنگاری می‌شود. اگر **دو بلوک یکسانی** در متن اصلی وجود داشته باشد، خروجی رمز شده‌ی آنها نیز یکسان خواهد بود. این باعث می‌شود که مهاجم بتواند **الگوهای مشابه** را در متن رمز شناسایی کند. در مقابل، در **CBC**، هر بلوک به خروجی قبلی وابسته است؛ حتی اگر دو بلوک مشابه وجود داشته باشد، خروجی‌های رمز شده متفاوت خواهند بود، زیرا هر بلوک با **خروجی قبلی XOR** شده است.

درCBC ، **هر بلوک رمز شده به بلوک قبلی وابسته است** .این یعنی اگر یک بیت از یک بلوک تغییر کند، **خروجی رمز شده از آن نقطه به بعد تغییر** می‌کند.

ستفاده از **بردار IV تصادفی** برای بلوک اول تضمین می‌کند که هر بار رمزنگاری با داده‌های یکسان، **نتایج متفاوتی** خواهد داشت.

-2

در **ECB**، **بلوک‌های یکسان از متن ساده** به **بلوک‌های یکسانی در متن رمز** تبدیل می‌شوند. این به مهاجم اجازه می‌دهد که **الگوهای تکراری** را در متن رمز شناسایی کند، حتی بدون داشتن کلید.

به دلیل **پردازش مستقل بلوک‌ها**، مهاجم می‌تواند با **حمله تکرار (Replay Attack)** یا **تحلیل آماری**، اطلاعاتی از پیام اصلی به‌دست آورد.

چون هر بلوک به‌طور جداگانه رمزنگاری می‌شود، یک مهاجم می‌تواند با **جایگزینی یا تکرار** بلوک‌ها در متن رمز، بدون نیاز به کلید، تغییرات معناداری در پیام رمز شده ایجاد کند.

شرایطی که استفاده از آن خطرناک است:

پیام‌های با الگوهای تکراری

در سیستم‌هایی که **صحت پیام اهمیت زیادی** دارد (مانند تراکنش‌های بانکی یا احراز هویت)، مهاجم می‌تواند با **تغییر یا جایگزینی بلوک‌ها**، حملاتی را اجرا کند.

اگر پیام حاوی **اطلاعات حساس یا محرمانه** باشد و از **ECB**استفاده شود، مهاجم می‌تواند با تحلیل تکرار بلوک‌ها به اطلاعاتی درباره محتوای پیام دست یابد.

-3

در این مد کاری ابتدا یک IV (عدد رندم) در شیفت رجیستر قرار میگیرد و با کلید شخصی داخل تابع رمزنگاری میشود و S بیت از آن برای عملیات بعدی انتخاب میشود و پس از آن خروجی با پیام اصلی XOR میشود و خروجی همان پیام رمز شده است که برای استفاده در مرحله بعدی در شیفت رجیستر قرار میگیرد.

نحوه انتقال خطا:
به دلیل اینکه خروجی داخل شیفت رجیستر قرار میگیرد در صورتی که دچار خطا شده باشد تا زمانی که خروجی در شیفت رجیستر باشد خطا پیام های بعدی را نیز تحت تاثیر قرار میدهد.

-4

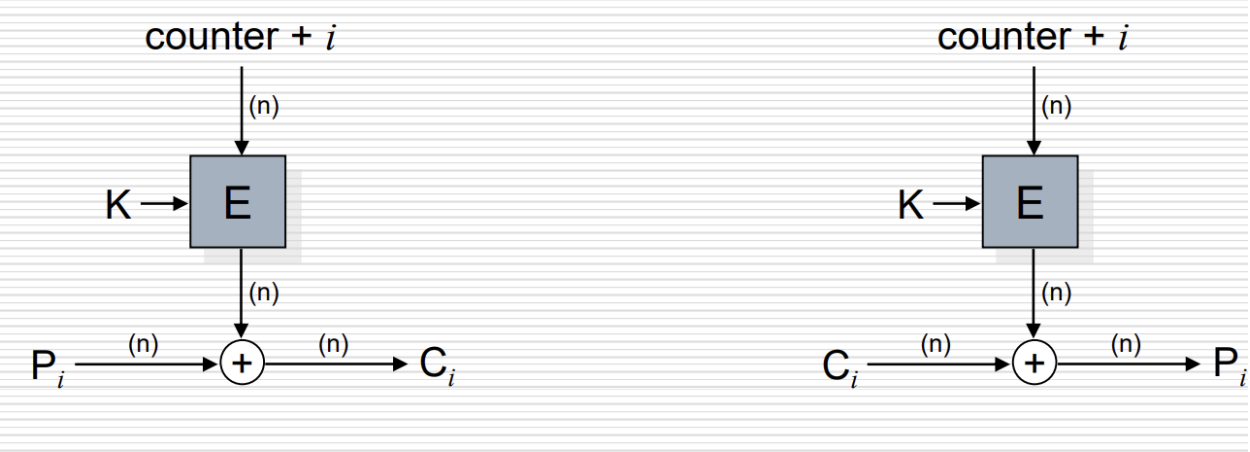
در الگوریتم CFB بخش رمز شده پس از XOR با پیام اصلی داخل شیفت رجیستر قرار میگیرد و در صورتی که دارای خطا باشد این خطا تا زمانی که پیامی که حاوی خطا است داخل شیفت رجیستر باشد این خطا انتقال پیدا میکند ولی در مد کاری OFB در صورتی که پیام رمز شده حاوی خطا باشد به داخل شیفت رجیستر نمی رود در واقع فقط همان بخش از پیامی که حاوی خطا است در مرحله رمزگشایی هم حاوی خطا است و به خاطر اینکه وارد شیفت رجیستر نمی شود خطا انتقال پیدا نمیکند و این مشکل در OFB رفع شده است.

-5

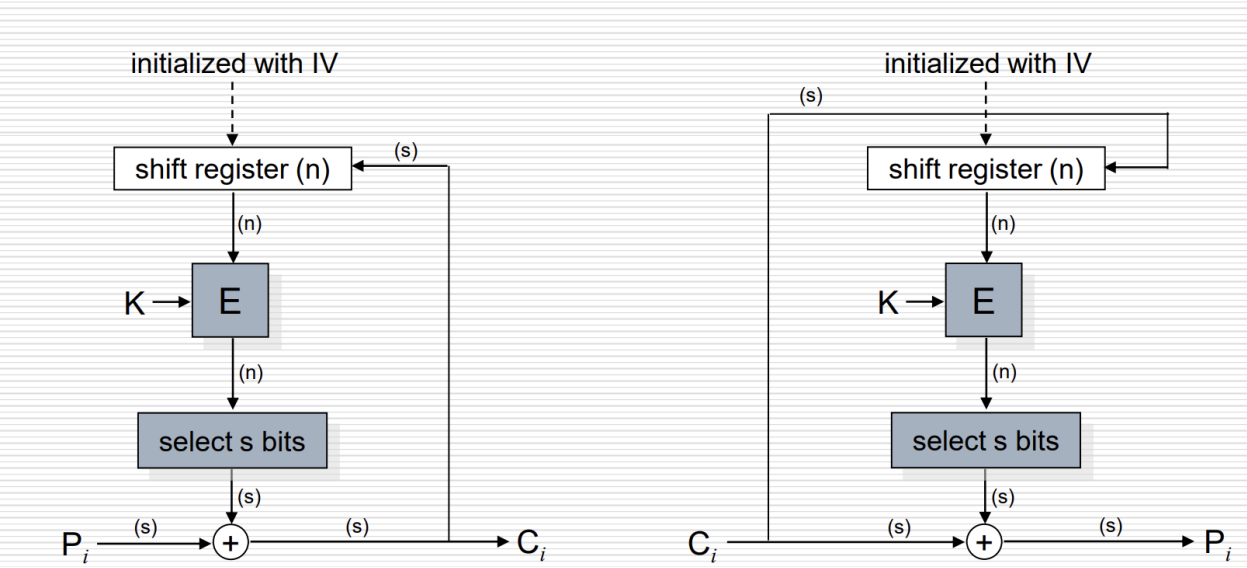
به خاطر اینکه در این الگوریتم ما از یک کانتر برای ایجاد کلید رمز شده استفاده میشود تمام عملیات های قبل از XOR با پیام اصلی میتواند به صورت موازی انجام شود و همینطور قطعه های پیام اصلی هم به صورت موازی XOR می شوند و به همین خاطر تمامی مراحل موازی انجام شده و همین اتفاق در رمزگشایی هم اتفاق می افتد و تمام مراحل به صورت موازی انجام میشود و در نتیجه سرعت بالایی دارد.

-6

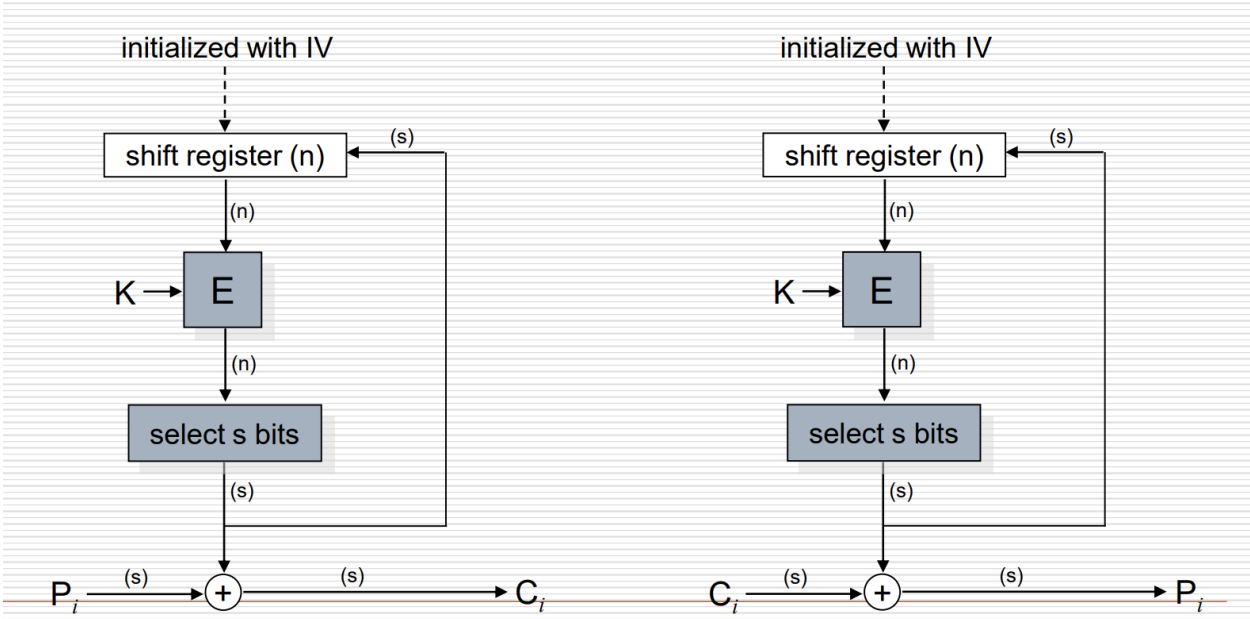
CTR



CFB


initialized with IV
shift register (n)
(s)
(n)
K
E
(n)
select s bits
(s)
P_i
(s)
(s)
C_i
initialized with IV
(s)
shift register (n)
(n)
K
E
(n)
select s bits
(s)
C_i
(s)
(s)
P_i

OFB


initialized with IV
shift register (n)
(s)
(n)
K
E
(n)
select s bits
(s)
P_i
(s)
(s)
C_i
initialized with IV
shift register (n)
(s)
(n)
K
E
(n)
select s bits
(s)
C_i
(s)
(s)
P_i

امتیازی:

1)

برگشت‌پذیری(Invertibility)

یکی از ویژگی‌های مهم XOR این است که برگشت‌پذیر است این ویژگی در فرایند رمزگشایی بسیار مفید است، زیرا همان عملیات XOR برای رمزگشایی نیز استفاده می‌شود.

عملیات XOR به ایجاد پیچیدگی در رابطه بین کلید و متن اصلی کمک می‌کند، که دو اصل مهم در رمزنگاری هستند:

Diffusion : هر بیت از ورودی می‌تواند بر تمام بیت‌های خروجی تأثیر بگذارد.

Confusion : رابطه بین کلید و متن رمز شده پیچیده و غیرقابل پیش‌بینی می‌شود.

2)

DESشامل چند مرحله کلیدی است:

جایگشت:

یک جایگشت یک تابع یک به یک است. هر جایگشت فقط ترتیب بیت‌ها را جابه‌جا می‌کند و هیچ دو ورودی متفاوتی نمی‌توانند به یک خروجی برسند.

عملیاتXOR :

XOR بین دو بیت، در صورتی که یکی از ورودی‌ها (کلید یا متن اصلی) معلوم باشد، برگشت پذیر است.

S-Box :

هر S-Box در DES یک تابع جایگذاری است. با اینکه هر S-Box ورودی 6 بیتی را به خروجی 4 بیتی تبدیل می‌کند، این فرآیند در ترکیب با کل ساختار DES برگشت‌پذیر است، زیرا تمامیS-Box ها و کل فرایند DES در کنار هم برگشت‌پذیر عمل می‌کنند.

Swap :

در هر مرحله، نیم‌بلوک‌ها (32 بیت) جابه‌جا می‌شوند. این عملیات نیز یک تابع یک به یک است.

(3

نحوه کاردر مد CTR (Counter Mode)، هر بلوک با یک شمارنده (Counter) منحصربه فرد ترکیب می‌شود و سپس رمزگذاری انجام می‌شود. شمارنده برای هر بلوک متفاوت است و از مقدار قبلی مستقل است.

قابلیت موازی‌سازی : از آنجایی که بلوک‌ها مستقل از هم رمزگذاری می‌شوند (به دلیل استفاده از شمارنده)، می‌توان تمام بلوک‌ها را هم‌زمان و موازی رمزگذاری کرد.

نتیجه CTR : بهترین گزینه برای پیاده‌سازی موازی است.