

The AES encryption algorithm

En undersökning av The Advanced Encryption Standard (AES)

Klass:

NA20

Handledare:

Jimmy Nylén

Författare:

Gabriel Lindeblad

Program:

Naturvetenskapsprogrammet

Abstract

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut lacinia ex eget sagittis congue. Nullam cursus egestas dolor, suscipit gravida magna ultrices sit amet. Nullam placerat dui eu arcu pharetra, sit amet tempor dolor convallis. Aenean sodales condimentum turpis, commodo maximus augue. Aenean vel nibh dui. Pellentesque ex libero, lacinia nec mauris vel, convallis consectetur felis. Maecenas ut nibh sed magna maximus imperdiet at id purus. In vel consequat metus. Donec non tincidunt nunc. Sed pulvinar odio ut sapien vestibulum, quis mollis arcu tempor. Maecenas ut sem leo. Sed leo risus, mollis eu ex vitae, feugiat consequat metus. Aenean interdum volutpat urna, nec tempor mi accumsan quis. Morbi blandit maximus urna non aliquet aes.

Innehåll

Ordlista	4
Akronymer	5
1 Inledning	6
1.1 Syfte	6
1.2 Frågeställningar	6
1.3 Avgränsning	6
2 Bakgrund	7
2.1 Kryptografi	7
2.1.1 Uppkomst	7
2.1.2 Utveckling	7
2.2 AES Uppkomst	7
3 Teori	8
3.1 Kryptering	8
3.2 Blockchiffer	8
3.2.1 Körlägen	8
3.2.1.1 ECB	8
3.2.1.2 CBC	9
3.2.1.3 OFB	11
3.3 Symetrisk & Asymmetrisk Kryptering	12
3.4 AES	13
3.4.1 Finite Fields	13
3.4.2 AES S-Box	13
3.4.3 Struktur	13
3.4.3.1 SubBytes operation	13
3.4.3.2 ShiftRows operation	13
3.4.3.3 MixColumns operation	13
3.4.3.4 AddRoundKey operation	13
3.4.4 Nyckel utökning	13
3.4.4.1 RotWord	13
3.4.4.2 SubWord	13
3.4.4.3 Rcon	13
3.4.5 AES-128bit	13
3.4.6 AES-192bit	13
3.4.7 AES-256bit	13
4 Metod & Genomförande	14
4.1 Implementering	14
4.2 Test Uppsättning	14
4.3 Genomförande	14
5 Resultat	15
5.1 Nyckellängds Test	15
5.2 Körläges Test	15

INNEHÅLL

6 Diskussion & Slutord	16
6.1 Felkällor	16
6.2 Förbättringar	16
6.3 Slutsats	16
6.4 Slutord	16
Källförteckning	17
Figurer	19

Ordlista

Caesarchiffer	Caesarchiffer är ett substitutions chiffer, vilket helt enkelt bygger på att man byter ut varje bokstav i medelandet med en annan. Ersättningens bokstaven bestäms genom att man hoppar ett visst antal hopp i alfabetet som exempelvis 3 hopp, vilket då innebär att ifall man har bokstaven a då skulle den bli ett d istället.[Wik21a]
Nyckelström	En nyckelström är i kryptografin en ström av Pseudoslump karaktärer som kan kombineras med exempelvis ett medelande för att producera en skiffrertext.[Wik21b]
Pseudoslump	Pseudoslump är en rad av nummer som kan se ut att vara helt slumptäckta men har blivit framställda genom en upprepbar process.[Wik22i]
Python	Python är ett högnivå programmerings språk byggt på programmerings språket C. De är skapat av Guido van Rossum och släpptes i Februari 1991.[Pyt22]
RSA	Rivest-Shamir-Adleman (RSA) är en av de mest välkända krypteringsalgoritmerna och var en av de första algorithmerna som byggde på en asymetrisk kryptering. RSA bygger på multiplikation av stora primtal där primtalen är nycklarna.[Wik22j]
VSCode	Visual Studio Code är en programutvecklingsmiljö som är skapad av Microsoft. Det är ett öppet källkods projekt som är tillgängligt för det flesta operativsystem och kan användas för att skriva kod i flera olika språk.[Wik21c]
XOR	Ett logisk operation inom datorvetenskap som fungerar ungefär som + uttrycket, med den enda skillnaden att $1 \oplus 1 = 0$. Detta samt att xor är en binär operation, vilket innebär att termerna bara kan vara 0 eller 1 och resultatet det samma.[LEW12]

Akronymer

AES Advanced Encryption Standard

CBC Cipher Block Chaining läge

DES Data Encryption Standard

ECB Electronic Code Book läge

IV Initialization Vector

OFB Output Feedback läge

1 Inledning

Kryptering, en bärande grundsten i dagens digitaliserade samhälle. De är väggen mellan oss och resten av världen, ett läs runt våra liv. Kryptering bygger på ett simpelt koncept, att dölja informationen från all förutom den menade mottagaren. Ett koncept som exempelvis fanns redan för 2000 år sedan när Julius Caesar använde de vi idag kallar Caesarchiffer för att skicka hemliga meddelanden.¹

Sedan dess har kryptografi självklart utvecklats enormt och vi har gått från de på ett sätt enkla men även eleganta Caesarchiffer som användes då till moderna algoritmer såsom Advanced Encryption Standard och Data Encryption Standard. Dessa algoritmer har samma syfte som Caesarchiffer men har utvecklats under en tid där datorer står som de dominerande informationshanteringsverktyget, vilket även är vad som används i denna rapport för att undersöka just en av dessa algoritmer.

1.1 Syfte

Syftet med denna undersökning är att undersöka krypterings algoritmen AES, för att utveckla en förståelse för mer avancerade krypterings algoritmer. Samt att bygga en uppfattning om hur man på olika sätt kan implementera krypterings algoritmer och vad de får för betydelse för deras säkerhet och hastighet.

1.2 Frågeställningar

- Hur påverkas tiden de tar att kryptera något mellan de olika nyckel längderna 128-bit, 192-bit och 256-bit nyckel?
- Hur påverkas skifertexten av de olika körlägen och vilken betydelse får de för den resultatet?
- Hur förändras tiden det tar att kryptera något beroende på ifall algoritmen körs i ECB, CBC eller OFB samt vilken betydelse det får ur ett tillämpningsperspektiv?

1.3 Avgränsning

Denna rapport är en avgränsad utvärdering av AES och dess användning som fokuserar på hur nyckellängd och körläge påverkar krypteringstiden. Detta samt hur den resulterande skiffer texten påverkas av vissa körlägen och hur detta i sin tur kan påverka säkerheten.

Denna analys av algoritmens säkerhet utelämnar faktorer såsom möjliga attacker där ibland exempelvis Brute-Force² & Side-Channel³ attacker. Undersökningen är även begränsad till en mjukvaruimplementering och tar inte hänsyn till möjliga skillnader som kan uppstå när algoritmen implementeras på en hårdvarunivå.

¹Dennis Luciano och Gordon Prichett. “Cryptology: From Caesar ciphers to public-key cryptosystems”. I: *The College Mathematics Journal* 18.1 (1987), s. 2–17.

²Neeraj Kumar. “Investigations in brute force attack on cellular security based on des and aes”. I: *IJCCEM International Journal of Computational Engineering & Management* 14 (2011), s. 50–52.

³“Algebraic Side-Channel Attacks on the AES: Why Time also Matters in DPA”. I: *Cryptographic Hardware and Embedded Systems - CHES 2009*. Utg. av Christophe Clavier och Kris Gaj. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, s. 97–111. ISBN: 978-3-642-04138-9.

2 Bakgrund

2.1 Kryptografi

Ordet kryptografi härstammar från de två grekiska orden kryptos som betyder gömd och grafein som betyder skrift.⁴ I sin simplaste form handlar kryptografi alltså om att gömma information. Detta är något som har visat sig på många olika sätt genom historien från något så simpelt som att skriva ett medelande i text då många i början inte kunde läsa till att idag istället använda komplexa algoritmer.⁵ Begreppet kryptografi har dock också fått en utökade betydelse med tiden då det idag även inkluderar olika metoder för att säkerställa autenticiteten av informationen och avsändaren.⁶

2.1.1 Uppkomst

Kryptografins historia kan man nästan säga börjar vid den tidigaste formen av skrift, vilket grundar sig i de faktum att de flesta inte kunde läsa. Detta är ju såklart något som förändrats på senare tid och i takt med de så har även kryptografin utvecklats. Exempel på utvecklingen går att se så tidigt som 1900 f.Kr då vissa egyptiska skribenter använde sig utav hieroglyfer på ett avvikande sätt, vilket troligen då gjordes i syfte att dölja informationen från dom som inte visste vad det skulle betyda.⁷

Den tidiga kryptografin är även något som kan observeras hos romarna där man använde Caesarchiffer och hos grekerna. Där grekernas metod byggde på att man virade en tejpbil runt någon form av ett cylinderformat objekt och sedan skrev medelandet på tejpen. När tejpen sedan togs av så är texten oläslig och mottagaren behövde vira upp tejpen på ett cylinderformat objekt med samma diameter för att läsa det.⁸

2.1.2 Utveckling

2.2 AES Uppkomst

⁴Wikipedia. *Kryptografi*. 2020. URL: <https://sv.wikipedia.org/w/index.php?title=Kryptografi&oldid=48532107> (hämtad 2022-09-07).

⁵Tony M Damico. “A brief history of cryptography”. I: *Inquiries Journal* 1.11 (2009).

⁶Nationalencyklopedin. *kryptografi*. 2022. URL: <http://www.ne.se/uppslagsverk/encyklopedi/1%C3%A5ng/kryptografi> (hämtad 2022-09-07).

⁷Dam09.

⁸Dam09.

3 Teori

3.1 Kryptering

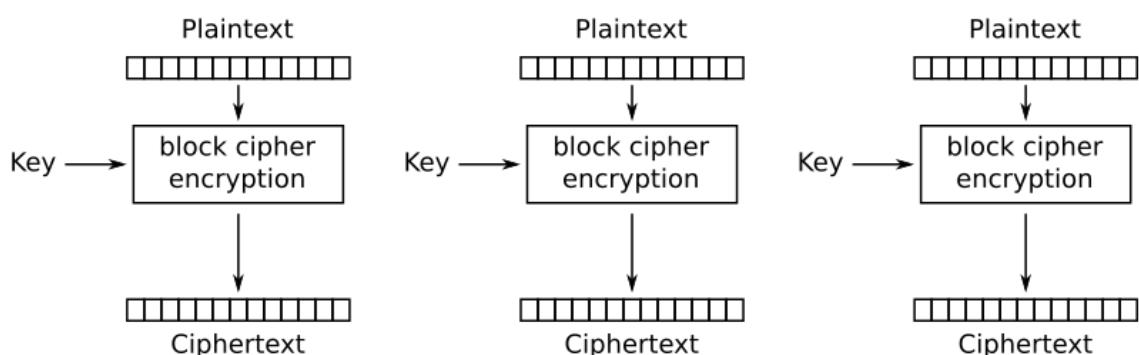
3.2 Blockchiffer

3.2.1 Körlägen

3.2.1.1 ECB

Electronic Code Book läge (ECB) är en av det enklaste blockchiffer körlägena som finns. ECB i sig är ganska lätt att förstå och bygger i huvudsak bara på att man delar upp den data man vill kryptera i delar kallade block och tar sedan varje block för sig och kör genom algoritmen, vilket tydligt visas i figur 3.1 & 3.2.⁹

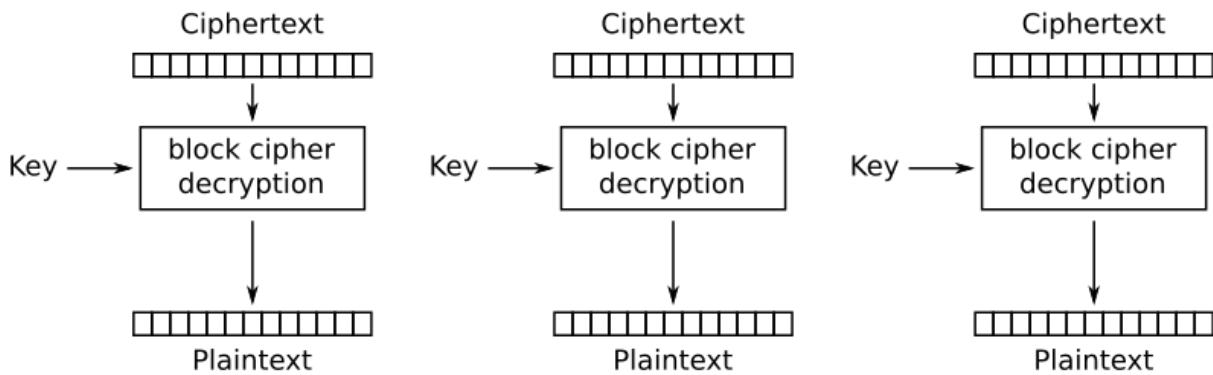
Figur 3.1 visar hur ECB fungerar vid kryptering. Här visas hur varje block för sig krypteras med hjälp av en blockchiffer algoritm tillsammans med den givna nyckeln.



Figur 3.1: Electronic Code Book läge kryptering [Wik22c]

Figur 3.2 visar istället hur ECB fungerar vid dekryptering, vilken är en till stort sett identisk operation med det enda undantaget att blockchiffret körs i dekrypterings läge istället för krypterings läge.

⁹Wikipedia. *Block cipher mode of operation*. 2022. URL: https://en.wikipedia.org/w/index.php?title=Block_cipher_mode_of_operation&oldid=1106163325 (hämtad 2022-09-25).



Figur 3.2: Electronic Code Book läge dekryptering [Wik22d]

På grund av ECB körlägets simplicitet så finns det dock även ett ganska stort problem med detta körläge. Det handlar om att ECB inte på något sätt förhindrar att två block med samma innehåll som krypteras inte resulterar i ett identiskt krypterat block.¹⁰

Vad detta innebär är att för större mängder data är att det börjar bildas mönster i skiffertexten. Detta är något som väldigt tydligt visar sig ifall man krypterar en bild, vilket går att se när man jämför figur 8.1 & 8.2. Det här faktumet är även varför ECB inte är ett säkert körläge och därför inte används näst intill aldrig i praktiken.¹¹

ECB har däremot även sina fördelar då de bland annat kan parallelliseras både när de gäller krypteringen och dekrypteringen. Detta samt att ECB även gör de möjligt att slumpmässigt dekryptera enskilda block av en skiffertext utan att man behöver dekryptera hela texten.¹²

3.2.1.2 CBC

Cipher Block Chaining läge är ett av de mest vanligen använda körlägena för många blockchiffer. Till skillnad från ECB så förhindrar CBC att två block med samma innehåll kan ge samma krypterade block. Detta gör CBC genom att lägga till ett extra steg utöver vad som finns i ECB. Steget är en XOR-operation mellan det krypterade blocket näckommande block innan de körs genom blockchiffer algoritmen.¹³ Matematisk sett kan detta formuleras såhär:

$$\begin{aligned} S_i &= K_n(B_i \oplus S_{i-1}) \\ S_0 &= IV \end{aligned}$$

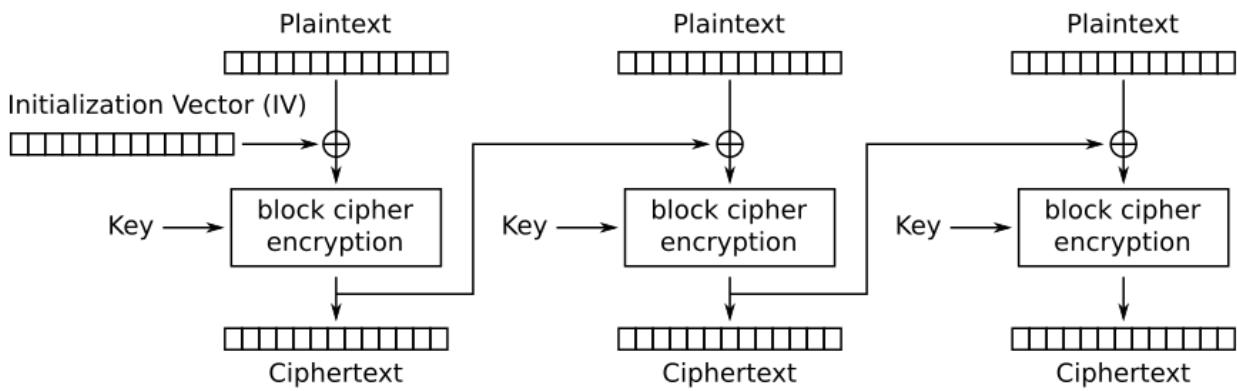
Där S_i är det krypterade blocket(skiffertexten), B_i är det blocket som ska krypteras, K_n är blockchiffer algoritmen där n står för nyckeln och S_{i-1} är det krypterade blocket före det blocket som ska krypteras. IV är en Initialization Vector (IV) som används vid krypteringen av de första blocket då de inte finns något föregående block att använda. i står för index där de första blocket har index värdet 1. Hela den här processen kan även ses i figur 3.3.

¹⁰Wik22a.

¹¹Wik22a.

¹²Wik22a.

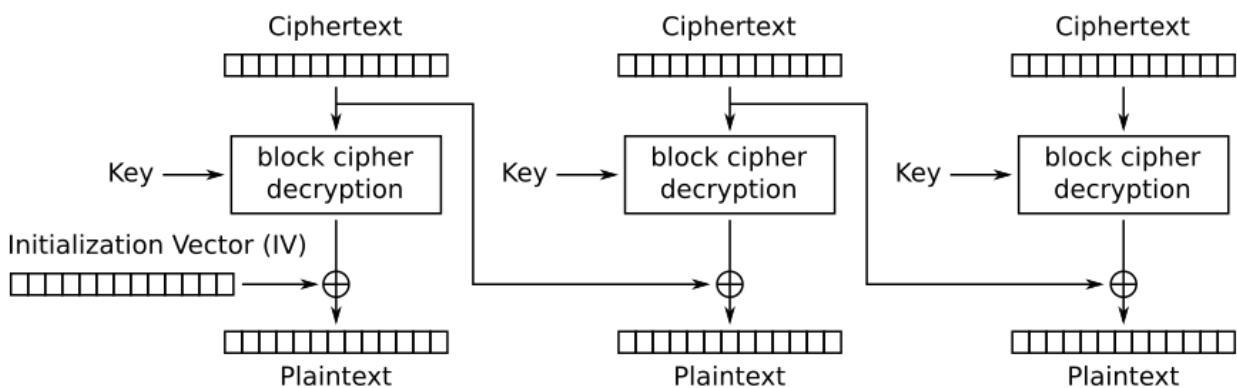
¹³Wik22a.



Figur 3.3: Cipher Block Chaining läge kryptering [Wik22e]

När de gäller dekrypteringsprocessen för CBC så bär den precis som för ECB stora likheter med krypteringsprocessen. Det två skillnaderna som finns är att blockchiffert körs i dekrypteringsläge istället för krypteringsläge. Samt att för varje block så genomförs en XOR-operation mellan det dekrypterade blocket och föregående block innan dekrypteringen av blocket.¹⁴ Även detta går att både matematiskt formulera och visuellt visa så här:

$$\begin{aligned} B_i &= K_n(S_i) \oplus S_{i-1} \\ S_0 &= IV \end{aligned}$$



Figur 3.4: Cipher Block Chaining läge dekryptering [Wik22f]

Fördelarna som kommer från den extra operationen i CBC till skillnad från ECB är då att varje block blir beroende av föregående block. Detta innebär att dom mönster som kunde dyka upp i ECB inte längre kan uppstå, vilket då gör CBC till ett mer säkert körläge än ECB. Dock kräver cbc en ytterligare faktor för att se till så att inte olika medelanden kan ge samma krypterade block. Därför så krävs en Initialization Vector (IV) som används vid första blocket.¹⁵

CBC är dock inte prefekt och har i sig också några nackdelar. Där ibland exempelvis de faktum att en incorrect IV leder till att de första blocket inte kan dekrypteras korrekt, detta påverkar dock inte de resterande blocken. På grund av det så kan man exempelvis lösa problemet genom

¹⁴Wik22a.

¹⁵Wik22a.

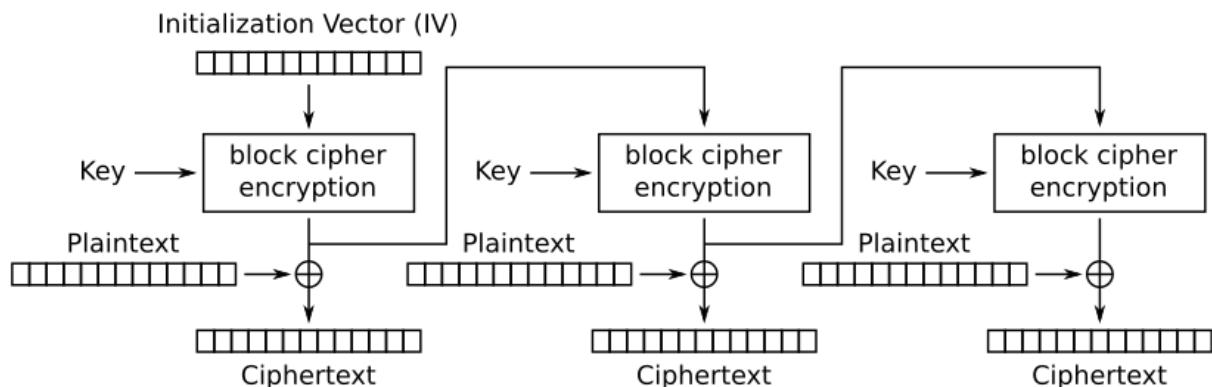
att första blocket bara innehåller någon typ av fyllnad, vilket då gör dekrypteringen möjlig utan tillgång till IV.¹⁶

Utöver detta så begränsas även CBC till att bara vara parallellisierbar under dekrypteringen och inte krypteringen, vilket är en konsekvens av att varje block i CBC är beroende av föregående block. CBC behåller dock fortfarande möjligheten som ECB har att slumpmässigt dekryptera enskilda block utan att behöva dekryptera hela skiftexten.¹⁷

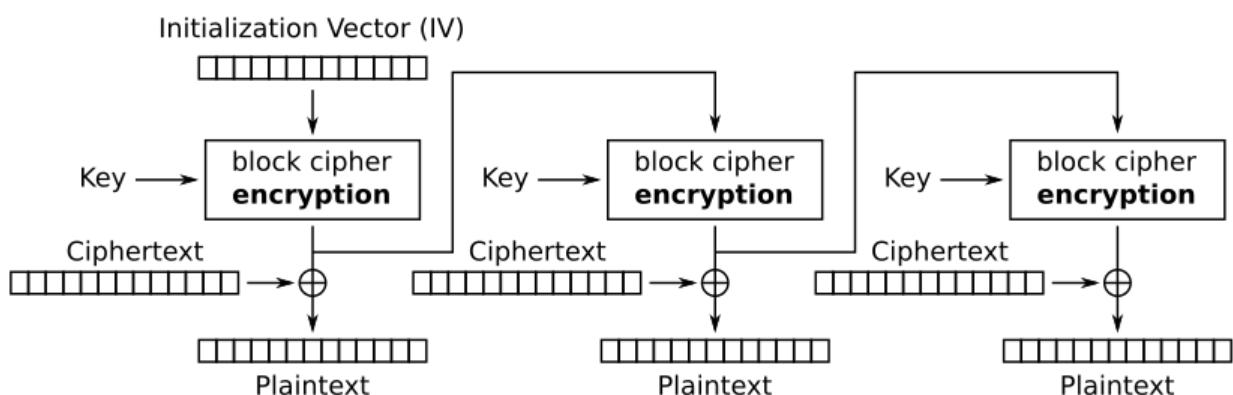
3.2.1.3 OFB

Output Feedback läge är ett ytterligare körläge som skiljer sig en del från ECB och CBC som redan presenterats. Den största skillnaden från det andra körlägena är att OFB inte använder blockchiffer algoritmen för att kryptera eller dekryptera blocken. Istället så körs IV genom blockchiffer algoritmen och den resulterande Nyckelström tillförs sedan genom en XOR-operation till blocket som ska krypteras eller dekrypteras.¹⁸

Tack vare XOR-operationens symmetriska natur så är så väl krypteringen som dekrypteringen av OFB identisk. Detta visas tydligt i figur 3.5 & 3.6:



Figur 3.5: Output Feedback läge kryptering [Wik22g]



Figur 3.6: Output Feedback läge dekryptering [Wik22h]

Utöver detta kan man även matematiskt beskriva OFB, vilket visas i ekvationen nedan:

¹⁶Wik22a.

¹⁷Wik22a.

¹⁸Wik22a.

$$\begin{aligned}
 S_i &= B_i \oplus O_i \\
 B_i &= S_i \oplus O_i \\
 O_i &= K_n(I_i) \\
 I_i &= O_{i-1} \\
 I_0 &= IV
 \end{aligned}$$

Här visas OFB körläget matematiskt där S_i är det krypterade blocket, B_i är blocket som ska krypteras och I_0 är IV. Men här finns även O_i som man kan säga är själva Nyckelströmmen som används för att kryptera eller dekryptera blocket. O_i i sin tur bygger då på att O_{i-1} körs genom blockchiffer algoritmen igen och sedan används för nästa blocks kryptering.

På grund av att OFB är utformat på det här sättet och att själva blocken som ska krypteras inte används fram till sista steget så är det möjligt att genomföra blockskiffer operationerna i förväg, vilket gör det möjligt att även parallellisera OFB. Dock kan OFB inte parallelliseras om man inte gör blockskiffer operationerna i förväg. Utöver detta saknar även OFB möjligheten att slumpmässigt dekryptera enskilda block utan att behöva dekryptera hela skiffertexten.¹⁹

3.3 Symmetrisk & Asymmetrisk Kryptering

Symmetrisk och asymmetrisk kryptering handlar om hur nycklar används i olika krypteringsalgoritmer. För symmetriska krypterings algoritmer så betyder detta att samma nyckel är vad som används för både kryptering och dekryptering. Medan asymmetrisk kryptering bygger på att man använder olika nycklar för kryptering och dekrypterings processerna.²⁰

De symmetriska krypterings algoritmernas huvudsakliga nackdel ligger i de faktum att de krävs en delad känd nyckel mellan båda parter. Detta är något som asymmetriska krypterings algoritmer inte behöver, vilket har lett till att man ofta använder asymmetriska krypterings algoritmer för att sköta nyckelutbytet för de symmetriska krypterings algoritmerna. Anledningen till detta är att det symmetriska krypterings algoritmerna ofta är bättre för större data mängder då dom bland annat behöver mycket kortare nyckellängder.²¹

Exempel på symmetriska krypterings algoritmer är bland annat AES och DES varav AES kommer förklaras djupare senare i denna rapport.²² Medan exempel på asymmetriska krypterings algoritmer är bland annat RSA.²³

¹⁹Wik22a.

²⁰Wikipedia. *Symmetric-key algorithm*. 2022. URL: https://en.wikipedia.org/w/index.php?title=Symmetric-key_algorithm&oldid=1106743629 (hämtad 2022-09-25).

²¹Wik22b.

²²Wik22b.

²³Wikipedia, the free encyclopedia. *RSA*. 2022. URL: <https://sv.wikipedia.org/w/index.php?title=RSA&oldid=50280992>.

3.4 AES

3.4.1 Finite Fields

3.4.2 AES S-Box

3.4.3 Struktur

3.4.3.1 SubBytes operation

SubBytes operationen bygger på ...

3.4.3.2 ShiftRows operation

3.4.3.3 MixColumns operation

3.4.3.4 AddRoundKey operation

3.4.4 Nyckel utökning

3.4.4.1 RotWord

3.4.4.2 SubWord

3.4.4.3 Rcon

3.4.5 AES-128bit

3.4.6 AES-192bit

3.4.7 AES-256bit

4 Metod & Genomförande

Metoden för denna undersökning bygger på en implementering av AES i programmeringsspråket Python. Detta tillsammans med ett antal konstruerade tester även dom implementerade i Python är vad som används för själva undersökningen av AES. Själva koden är skriven med hjälp av programmet VSCode och är byggd huvudsakligen för Python 3.10.

4.1 Implementering

Implementeringen av AES är uppdelad i ett antal funktioner till stor del är baserat på hur strukturen och uppdelningen av AES beskrivs i “AES proposal: Rijndael”²⁴...

4.2 Test Uppsättning

Test uppsättningen går att se i filen Analyze.py ...

4.3 Genomförande

²⁴DR99.

5 Resultat

5.1 Nyckellängds Test

5.2 Körläges Test

6 Diskussion & Slutord

6.1 Felkällor

6.2 Förbättringar

6.3 Slutsats

6.4 Slutord

XOR

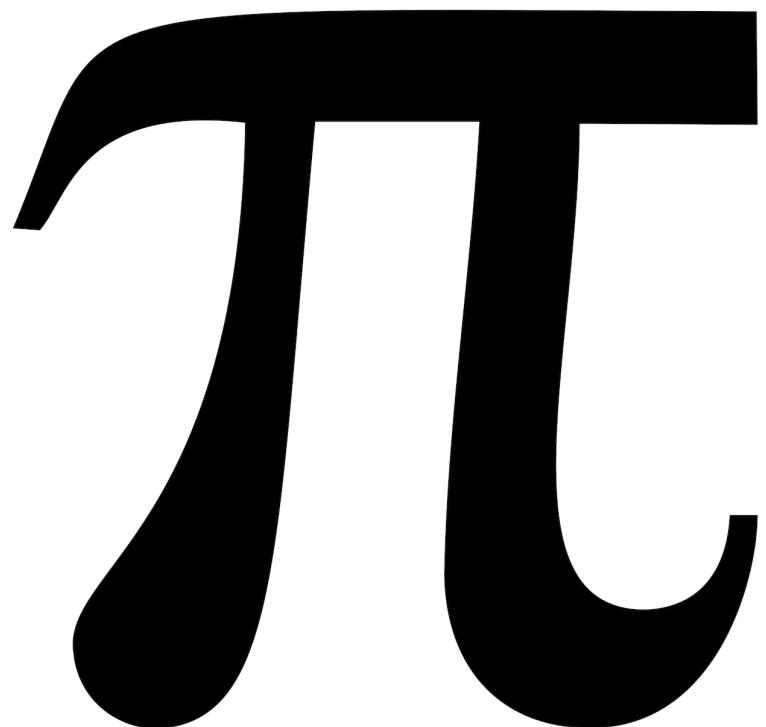
Källförteckning

- [CG09] “Algebraic Side-Channel Attacks on the AES: Why Time also Matters in DPA”. I: *Cryptographic Hardware and Embedded Systems - CHES 2009*. Utg. av Christophe Clavier och Kris Gaj. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, s. 97–111. ISBN: 978-3-642-04138-9.
- [Dam09] Tony M Damico. “A brief history of cryptography”. I: *Inquiries Journal* 1.11 (2009).
- [DR99] Joan Daemen och Vincent Rijmen. “AES proposal: Rijndael”. I: (1999).
- [Kum11] Neeraj Kumar. “Investigations in brute force attack on cellular security based on des and aes”. I: *IJCEM International Journal of Computational Engineering & Management* 14 (2011), s. 50–52.
- [LEW12] FEATURE MICHAEL LEWIN. “All about XOR”. I: *For details of ACCU, our publications and activities, visit the ACCU website: www. accu. org* (2012), s. 14.
- [LP87] Dennis Luciano och Gordon Prichett. “Cryptology: From Caesar ciphers to public-key cryptosystems”. I: *The College Mathematics Journal* 18.1 (1987), s. 2–17.
- [Nat22] Nationalencyklopedin. *kryptografi*. 2022. URL: <http://www.ne.se/uppslagsverk/encyklopedi/1%C3%A5ng/kryptografi> (hämtad 2022-09-07).
- [Pyt22] Python Software Foundation. *What is Python?* 2022. URL: <https://docs.python.org/3/faq/general.html#what-is-python> (hämtad 2022-09-01).
- [Wik20] Wikipedia. *Kryptografi*. 2020. URL: <https://sv.wikipedia.org/w/index.php?title=Kryptografi&oldid=48532107> (hämtad 2022-09-07).
- [Wik21a] Wikipedia, the free encyclopedia. *Caesarchiffer*. 2021. URL: <https://sv.wikipedia.org/w/index.php?title=Caesarchiffer&oldid=48885737> (hämtad 2022-09-23).
- [Wik21b] Wikipedia, the free encyclopedia. *Keystream*. 2021. URL: <https://en.wikipedia.org/w/index.php?title=Keystream&oldid=1039345792>.
- [Wik21c] Wikipedia, the free encyclopedia. *Visual Studio Code*. 2021. URL: https://sv.wikipedia.org/w/index.php?title=Visual_Studio_Code&oldid=48905230.
- [Wik22a] Wikipedia. *Block cipher mode of operation*. 2022. URL: https://en.wikipedia.org/w/index.php?title=Block_cipher_mode_of_operation&oldid=1106163325 (hämtad 2022-09-25).
- [Wik22b] Wikipedia. *Symmetric-key algorithm*. 2022. URL: https://en.wikipedia.org/w/index.php?title=Symmetric-key_algorithm&oldid=1106743629 (hämtad 2022-09-25).
- [Wik22c] Wikipedia, the free encyclopedia. *Block cipher mode of operation*. 2022. URL: https://en.wikipedia.org/w/index.php?title=Block_cipher_mode_of_operation&oldid=1106163325#/media/File:ECB_encryption.svg (hämtad 2022-09-25).
- [Wik22d] Wikipedia, the free encyclopedia. *Block cipher mode of operation*. 2022. URL: https://en.wikipedia.org/w/index.php?title=Block_cipher_mode_of_operation&oldid=1106163325#/media/File:ECB_decryption.svg (hämtad 2022-09-25).
- [Wik22e] Wikipedia, the free encyclopedia. *Block cipher mode of operation*. 2022. URL: https://en.wikipedia.org/w/index.php?title=Block_cipher_mode_of_operation&oldid=1106163325#/media/File:CBC_encryption.svg (hämtad 2022-09-26).
- [Wik22f] Wikipedia, the free encyclopedia. *Block cipher mode of operation*. 2022. URL: https://en.wikipedia.org/w/index.php?title=Block_cipher_mode_of_operation&oldid=1106163325#/media/File:CBC_decryption.svg (hämtad 2022-09-26).
- [Wik22g] Wikipedia, the free encyclopedia. *Block cipher mode of operation*. 2022. URL: https://en.wikipedia.org/w/index.php?title=Block_cipher_mode_of_operation&oldid=1106163325#/media/File:OFB_encryption.svg.

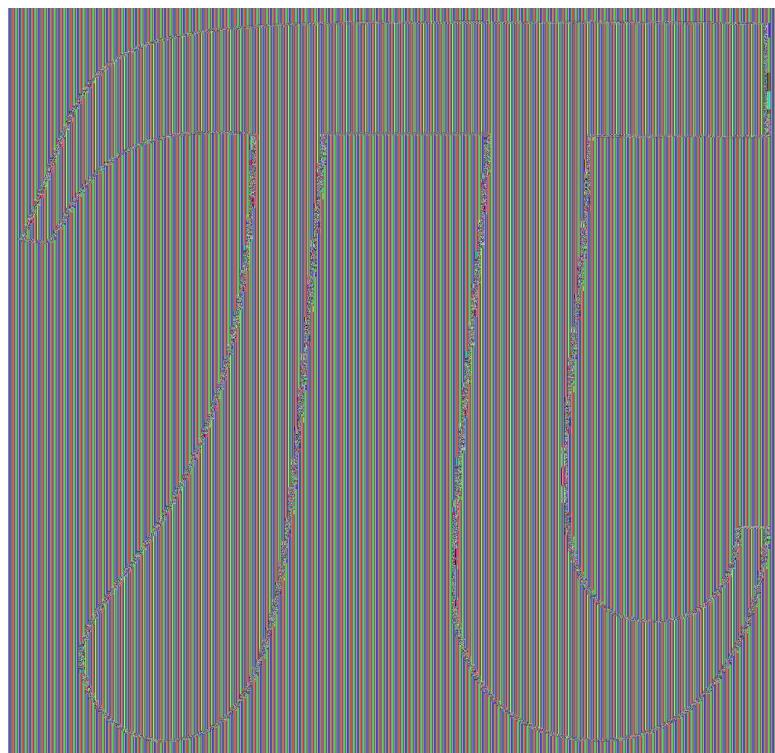
- [Wik22h] Wikipedia, the free encyclopedia. *Block cipher mode of operation*. 2022. URL: https://en.wikipedia.org/w/index.php?title=Block_cipher_mode_of_operation&oldid=1106163325#/media/File:OFB_decryption.svg.
- [Wik22i] Wikipedia, the free encyclopedia. *Pseudorandomness*. 2022. URL: <https://en.wikipedia.org/w/index.php?title=Pseudorandomness&oldid=1112429322>.
- [Wik22j] Wikipedia, the free encyclopedia. *RSA*. 2022. URL: <https://sv.wikipedia.org/w/index.php?title=RSA&oldid=50280992>.
- [Wik99] Wikipedia, the free encyclopedia. *Advanced Encryption Standard*. 1999. URL: [https://en.wikipedia.org/wiki/Advanced_Encryption_Standard#/media/File:AES_\(Rijndael\)_Round_Function.png](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard#/media/File:AES_(Rijndael)_Round_Function.png) (hämtad 2022-09-02).

Figurer

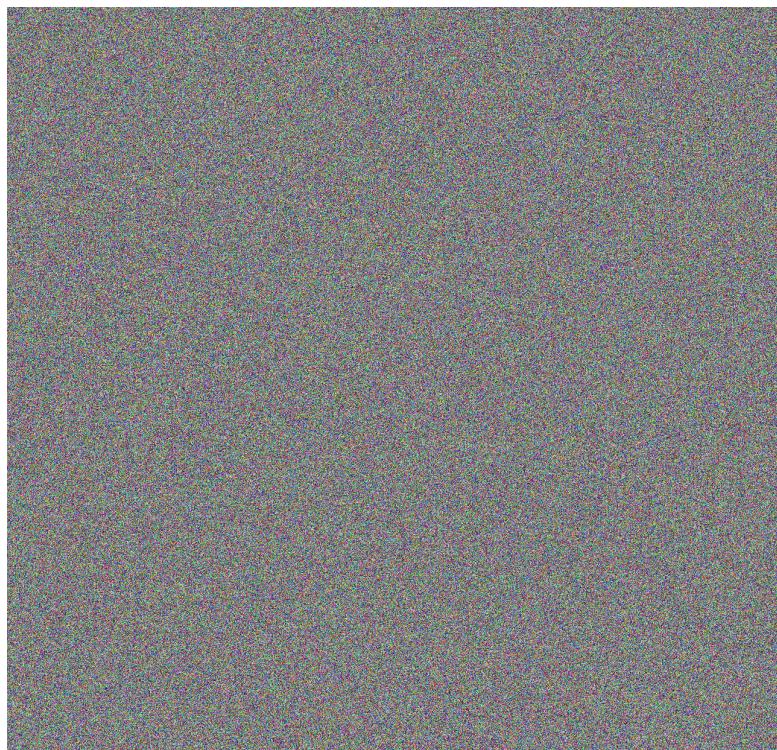
3.1	Electronic Code Book läge kryptering [Wik22c]	8
3.2	Electronic Code Book läge dekryptering [Wik22d]	9
3.3	Cipher Block Chaining läge kryptering [Wik22e]	10
3.4	Cipher Block Chaining läge dekryptering [Wik22f]	10
3.5	Output Feedback läge kryptering [Wik22g]	11
3.6	Output Feedback läge dekryptering [Wik22h]	11
8.1	Orginal bild	20
8.2	Efter ECB Kryptering	20
8.3	Efter CBC Kryptering	21
8.4	Efter OFB Kryptering	21
8.5	Uppställning av vanliga rundor	22



Figur 8.1: Orginal bild



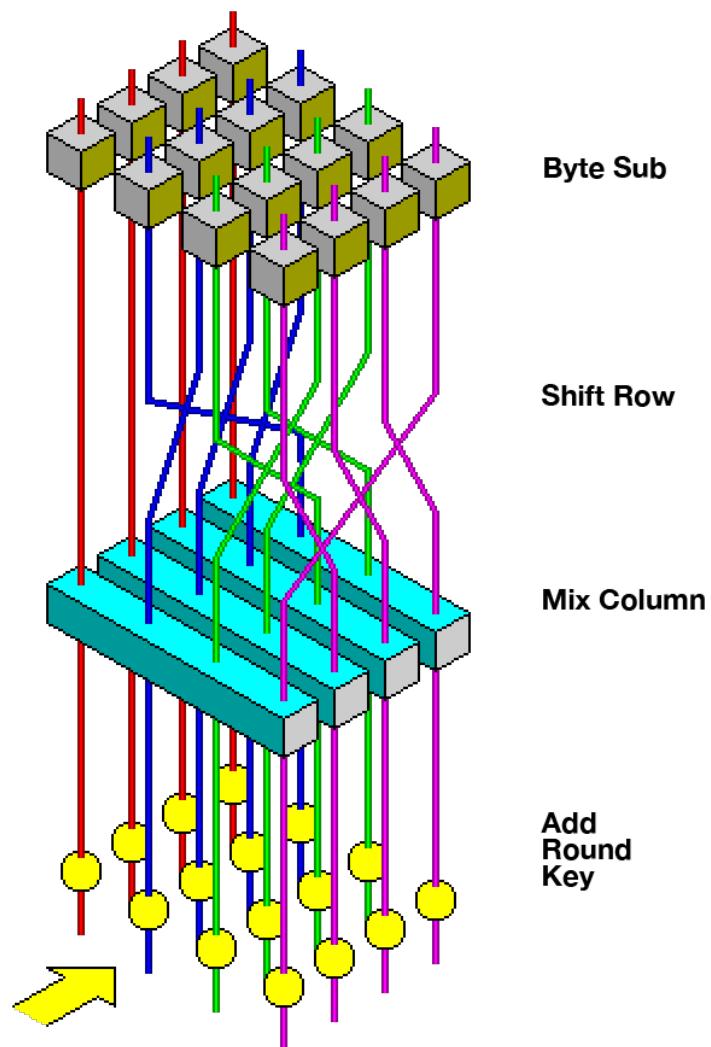
Figur 8.2: Efter ECB Kryptering



Figur 8.3: Efter CBC Kryptering



Figur 8.4: Efter OFB Kryptering



Figur 8.5: Uppställning av vanliga rundor

Källa: Wikipedia, the free encyclopedia. *Advanced Encryption Standard*. 1999. URL: [https://en.wikipedia.org/wiki/Advanced_Encryption_Standard#/media/File:AES_\(Rijndael\)_Round_Function.png](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard#/media/File:AES_(Rijndael)_Round_Function.png) (hämtad 2022-09-02)