Luku 2

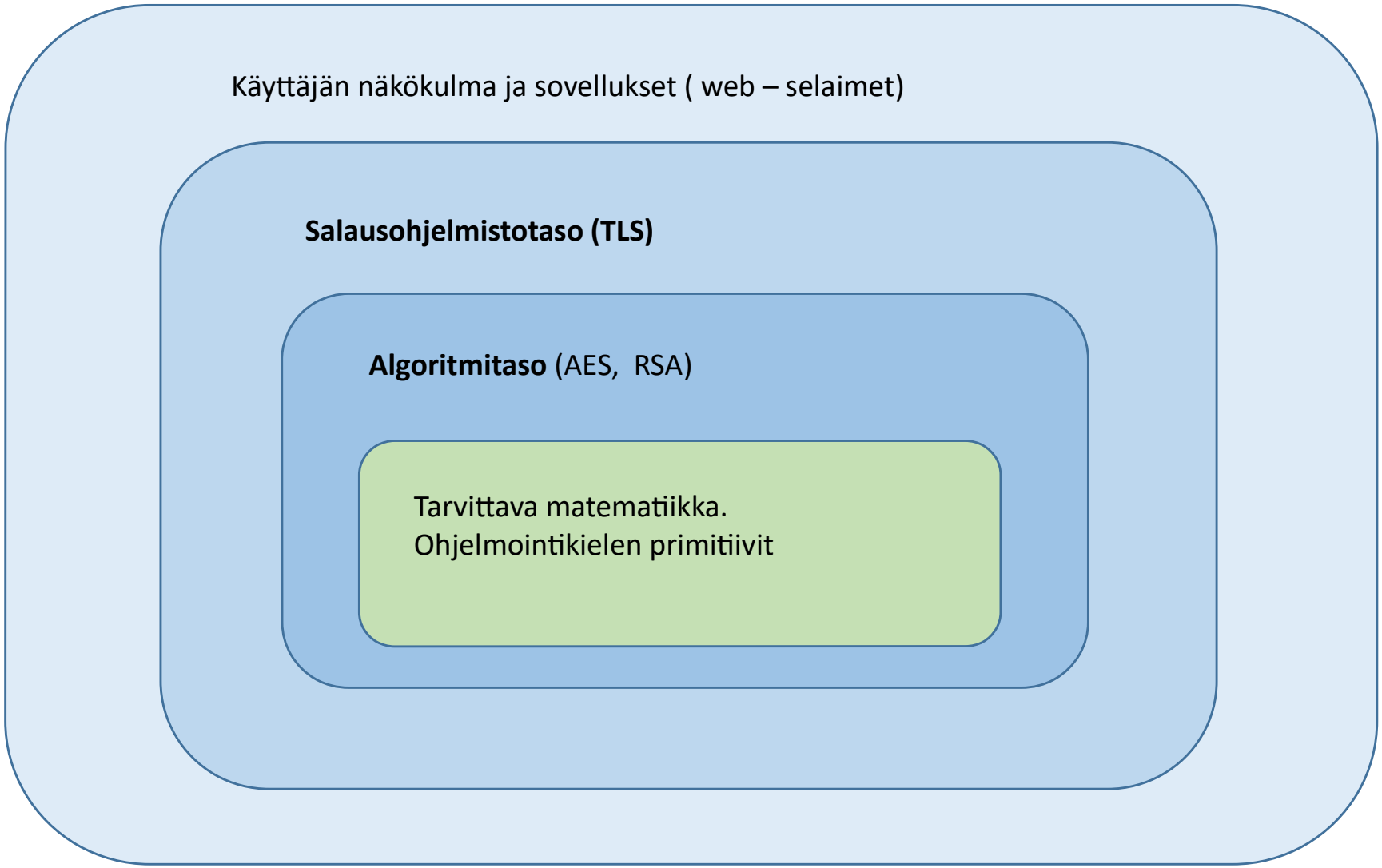
Modernien salausten päätyypit

Salausohjelmistojen toiminta eri tarkastelutasoilla

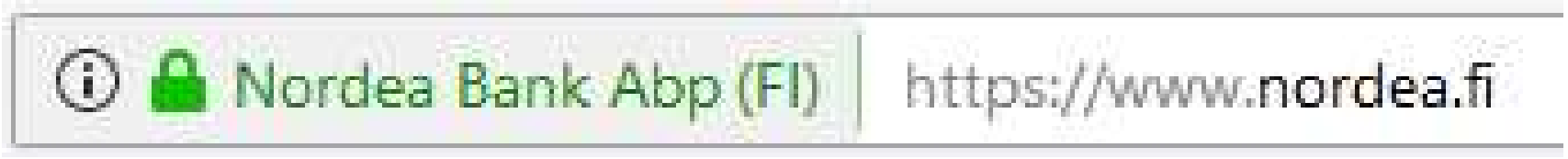
- Sovellusohjelmataso, käyttäjän näkökulma
- Salausohjelmistotaso
- Algoritmitaso
- Matemaattisten rakenteiden taso

Lohkosalaus

Jonosalaus


Käyttäjän näkökulma ja sovellukset (web – selaimet)
Salausohjelmistotaso (TLS)
Algoritmitaso (AES, RSA)
Tarvittava matematiikka.
Ohjelmointikielen primitiivit

Sovellustaso (selaimet)



Suojattuja yhteyksiä käytetään useimmiten selainten kautta. Lukon vihreä väri selaimen URL rivillä kertoo, että yhteys suojattu ja salattu. Punainen väri tarkoittaa, että suojauksessa on ongelma.

Internet-yhteyksien salausohjelmisto on lähes aina TLS. Se on mukana kaikissa internet selaimissa.

Salausohjelmistotaso

Mozilla Firefox selaimessa klikkaus lukon kuvaan antaa seuraavat tiedot:

Tekniset tiedot
Yhteys salattu (TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, 256-bittinen avain, TLS 1.2)

TLS ver 1.2 on ns. hybridisalausohjelmisto, jossa on useita toimintoja ja algoritmeja toimintojen toteuttamiseksi. Yli 95% Internetin yli tapahtuvista suojatuista yhteyksistä käyttää TLS:ää

Hybridisalausohjelmistojen perustoiminnot:

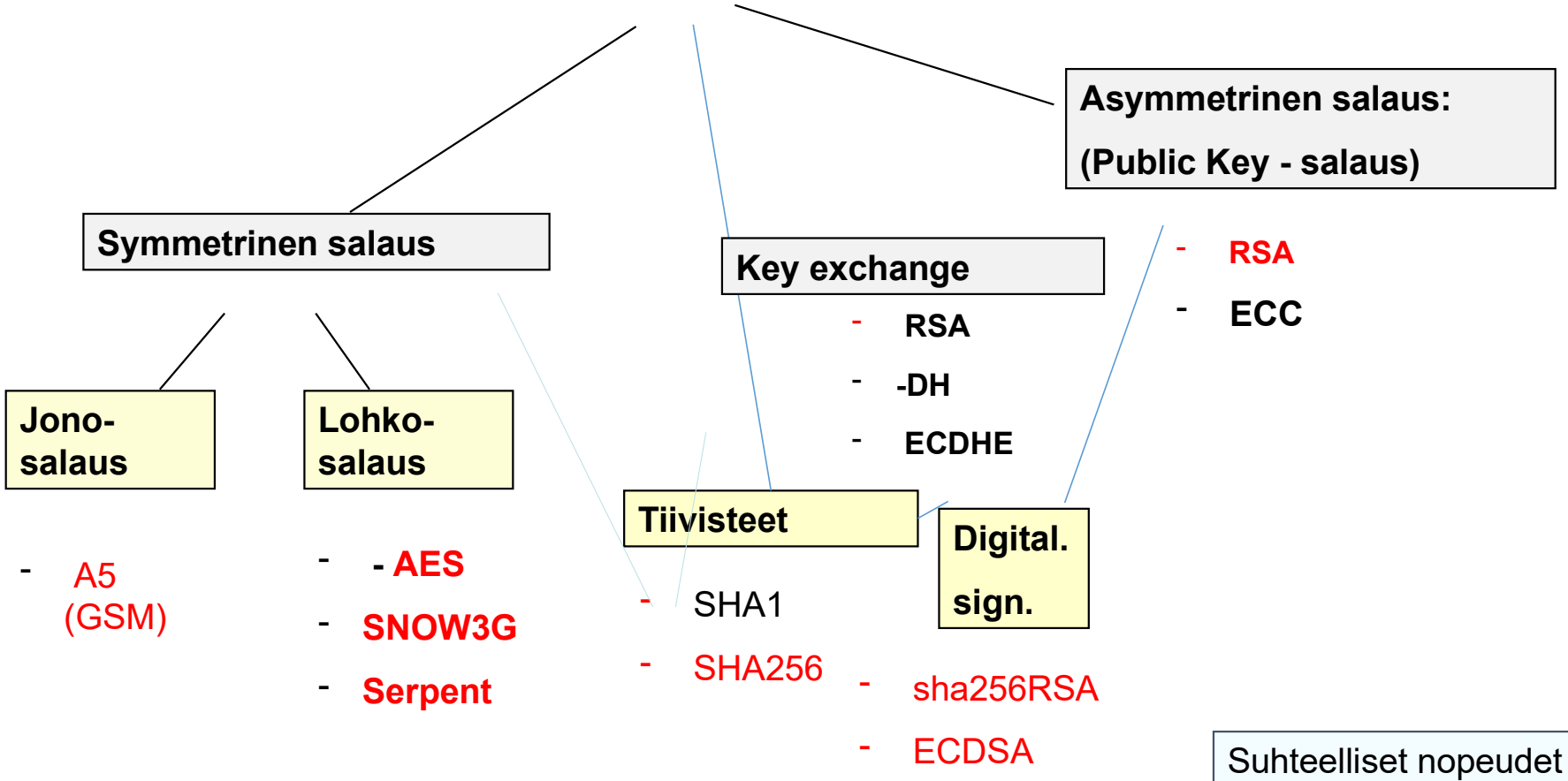
1. Autentikointi (käyttäjän todentaminen)
2. Key exchange (symmetrisestä avaimesta sopiminen)
3. Tietoliikenteen salaus
4. Digitaalinen allekirjoitus

Algoritmitaso

Alla Nordea pankin TLS – yhteyden käyttämiä algoritmeja

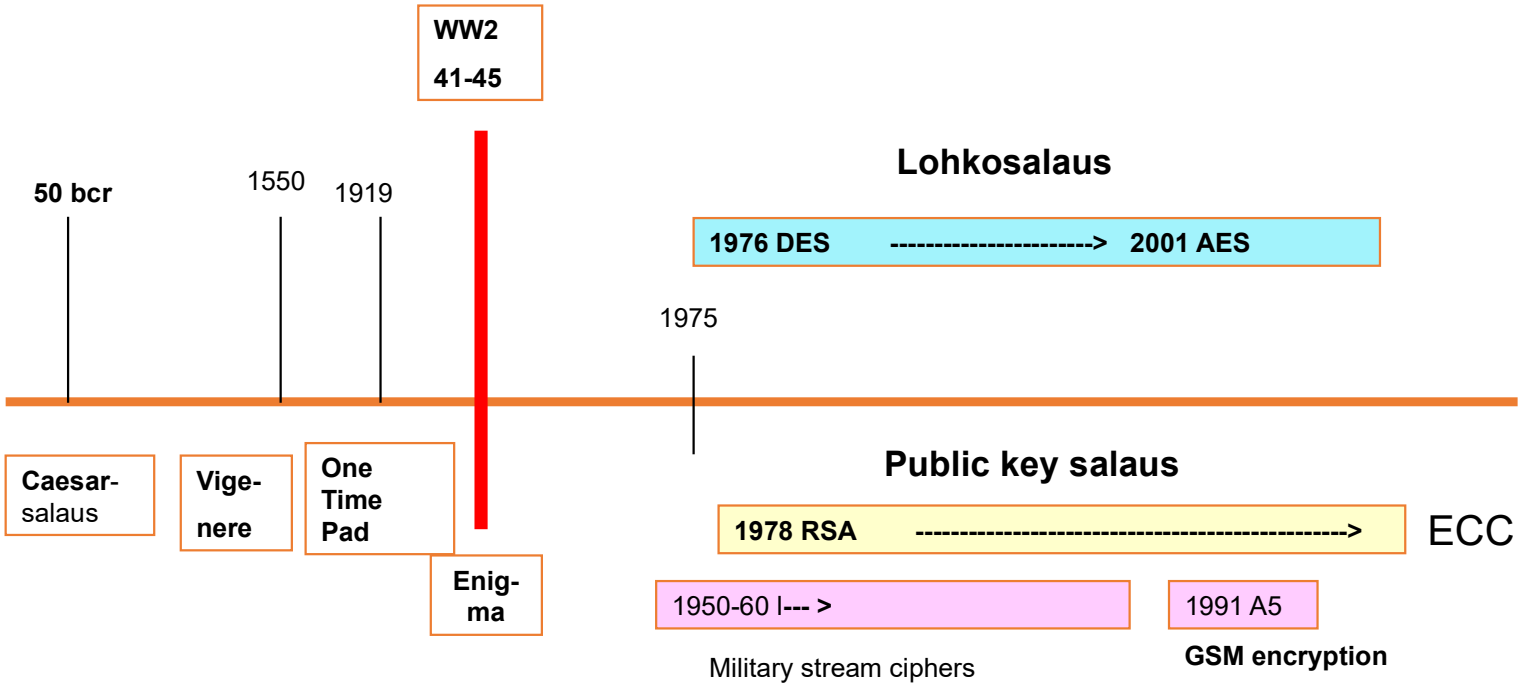
Function	Algorithm name
Palvelimen todennus	**RSA**
Avaimesta sopiminen	**ECDHE**
Tiedonsiirron salaus	**AES256 GCM-moodi**
Digitaalinen allekirjoitus	**sha384RSA**

Algoritmityyppejä



Suhteelliset nopeudet	
Hash	3
Stream cipher	2
Block cipher	1
Asymmetric	0.02

Salausmenetelmien aikajana



Algoritmit vaihtuvat hitaasti. RSA on ollut PK salauksen standardi jo 45 vuotta. Lohkosalausstandardia on vaihdettu kerran 45 vuoden aikana.
Syitä konseratismiin: 1. Algoritmien muuttamisen kustannukset. 2. Vanhojen algoritmien turvallisuus on matemaattisesti todistettu ja niitä on testattu vuosikymmeniä.

4. Matemaattinen taso

Salausalgoritmit perustuvat matematiikkaan

1. Fermat'n ja Eulerin teoreemat	
2. Satunnaislukugeneraattorit	**pseurodandomness**
3. Alkulukutestit	**Fermat'n ja Rabin- Millerin testit**
4. Jakojäännösaritmetiikka	**lukujoukot Z_n**
5. Nopea potenssiinkorotus mod n	**Powermod - algoritmi**
6. Käänteisluku mod n	**Extended Euclid's algorithm**
7. Sykliset ryhmät Z_p^*	**ryhmäteoria**
8. Elliptiset käyrät	**ryhmäteoria**

Symmetrisen salauksen tyyppejä

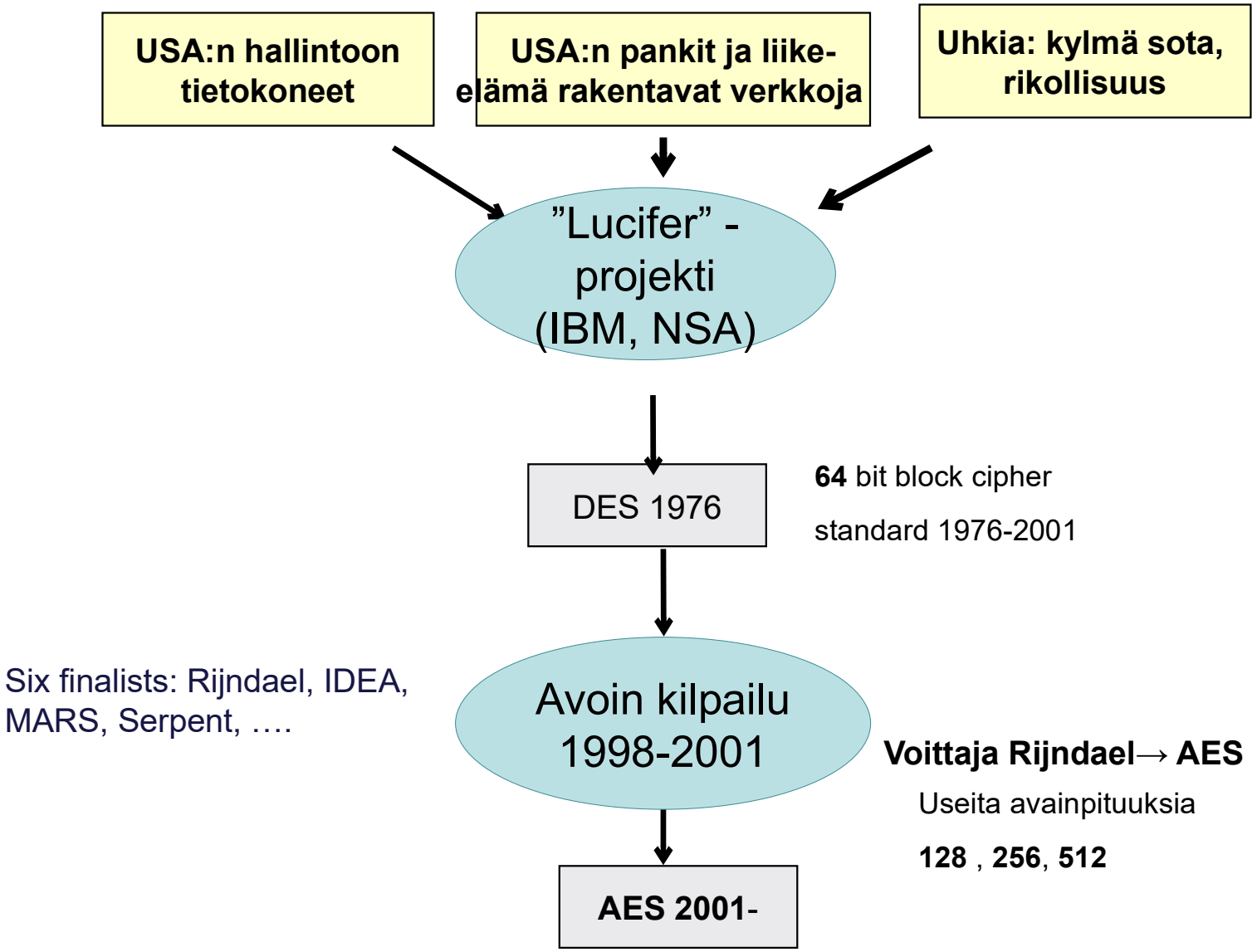
1. Lohkosalaus (block ciphers)
-sekä langallisissa, että langattomissa yhteyksissä
Aiheeseen liittyvää terminologiaa:
-Diffuusio ja konfuusio periaatteet
-Permutaatio-substituutio -verkot
- Käyttömoodit (modes of operation)
-AES : block cipher standard

2. Syncroninen jonosalaus (synchronous stream ciphers)
- ei juuri enää käytössä
- GSM (G2)– encryption A5

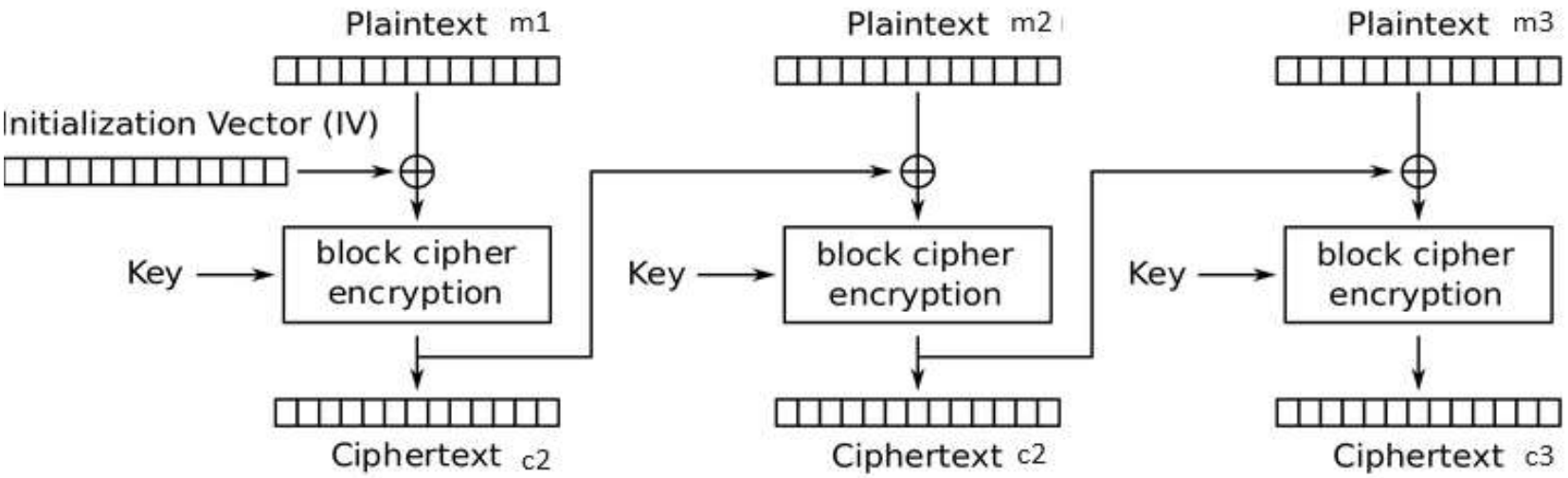
Lohkosalaus

- Keskeinen rooli salausohjelmistoissa. Käytetään datan tehokkaaseen ja luotettavaan salaukseen
- Symmetrinen salaus, jossa viesti salataan lohkoissa (lohkon koko tavallisesti128 bittiä)
- Nykyinen standardi on AES (2001...). Sitä käytetään sekä langallisten ja langattomien yhteyksien tiedonsiirron salaukseen

Lohkosalauksen historiaa



Kaavio lohkosalaimen toiminnasta CBC moodissa
(CBC = cipher block chaining)



Viesti m jaetaan 128 bitin lohkoihin m1, m2, m3, …

Avain k on 128 bittinen AES128:ssa. Salakirjoituslohkot ovat c1, c2, c3, ...

CBC moodissa jokainen salakirjoituslohko lisätään XOR- yhteenlaskulla seuraavaan viestilohkoon ennen tämän salausta.

$$c_n = AES(m_n, k, c_{n-1})$$

Lohkosalauksen vaatimukset

- Lohkosalauksen tulisi toimia sekä software, että hardware (chip) toteutuksina (AES salaus täyttää ehdon)
- Algoritmin tulee olla luotettava ja nopea
- Sitä voidaan käyttää sekä tiedostojen, että tietoliikenteen salaukseen
- Algoritmin oltava julkinen ja avoin (Kerckhoffin periaate)
- Avaimen vähimmäispituus 128 bittiä

Shannonin diffuusio ja konfuusio

Lohkosalauksen toiminta perustuu kahteen Claude Shannonin periaatteeseen: diffuusio ja konfuusio

Diffuusio: (salakirjoituksen ja viestin riippuvuus)
Viestilohkon yhden bitin muuttamisen tulisi muuttaa noin 50% salakirjoituslohkon biteistä. (=> tuloksen tulee siis riippua yhtä paljon kaikista viestilohkon biteistä)

Konfuusio: (salakirjoituksen ja avaimen riippuvuus)
Jokaisen salakirjoituslohkon bitin tulee riippua kaikista avaimen biteistä. Yhdenkin avaimen bitin muuttamisen pitäisi muuttaa salakirjoitusta täydellisesti, ei vain sen osaa.

Salakirjoituksen tulee binäärimuodossa muistuttaa tilastollisesti kolikonheittojen jakaumaa ja toteuttaa pseudorandom bittijonon kolme ominaisuutta (selitetään myöhemmin)

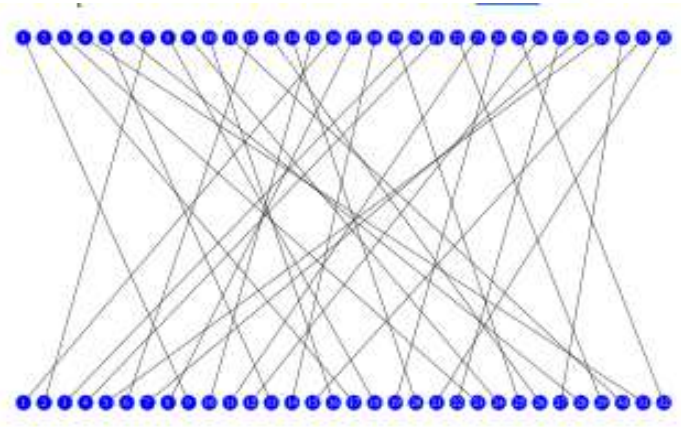
Permutaatio – Substituutio verkot

Käytännössä diffuusio ja konfuusioperiaatteet toteutetaan käyttämällä permutaatio- ja substituutio verkkoa.

Substituutio = tietyn bittijonon korvaaminen kuvaaminen toiseksi bittijonoksi, toteutetaan perinteisesti SBOX -taulukkoa käyttäen. Alla on DES salauksen Sbox nro 6 Esimerkissä bittijono 011011 kuvautuu jonoksi 1001

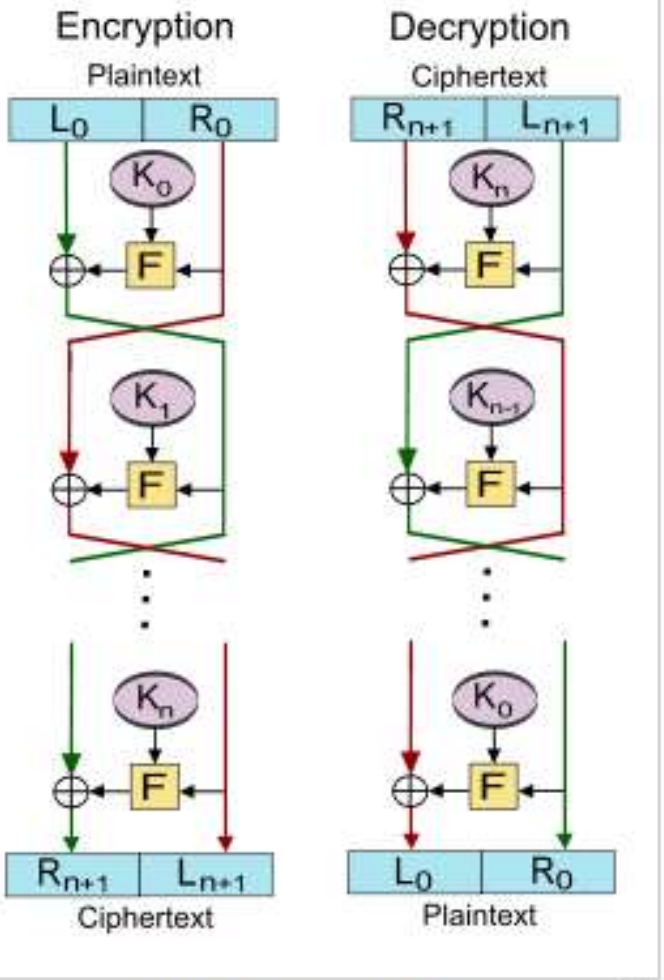
S_5		Middle 4 bits of input															
		0000	**0001**	**0010**	**0011**	**0100**	**0101**	**0110**	**0111**	**1000**	**1001**	**1010**	**1011**	**1100**	**1101**	**1110**	**1111**
Outer bits	**00**	0010	1100	0100	0001	0111	1010	1011	0110	1000	0101	0011	1111	1101	0000	1110	1001
	01	1110	1011	0010	1100	0100	0111	1101	0001	0101	0000	1111	1010	0011	1001	1000	0110
	10	0100	0010	0001	1011	1010	1101	0111	1000	1111	1001	1100	0101	0110	0011	0000	1110
	11	1011	1000	1100	0111	0001	1110	0010	1101	0110	1111	0000	1001	1010	0100	0101	0011

Permutaatio = lohkon bittien uudelleen järjestäminen käyttäen P-box taulukkoa. table. Kuvassa 64 bittiä järjestetään uudelleen DES:n P-box:lla



Useita kierroksia: Permutation- Substitution luuppia ajetaan useita kertoja. (DES salaimessa 16 kertaa) , jotta salakirjoituksesta saadaan poistettua kaikki säännönmukaisuudet ja tulos muistuttaa kolikonheiton tulosta.

Feistelin verkko



Many block ciphers, such as DES and Blowfish utilize structures known as *Feistel ciphers*

IBM engineer Feistel created PS network in 1960's using permutations and SBOX:s (substitution tables).
Feistel network appeared originally in DES block cipher, it was later reused in Blowfish cipher and later in **Kasumi , which was recently a standard block cipher in 3G and 4G** (Today Kasumi is replaced by AES and SNOW3G)

Feistel algorithm has 16 rounds. Each round uses different subkey Ki derived from the master key K

Each round starts with initial permutation and ends with final permutation (called P-BOX:s)

Between permutations there are eight S-BOX:s, which obscure the connection between the key and the ciphertext (Shannon's confusion)

AES : lohkosalausstandardi

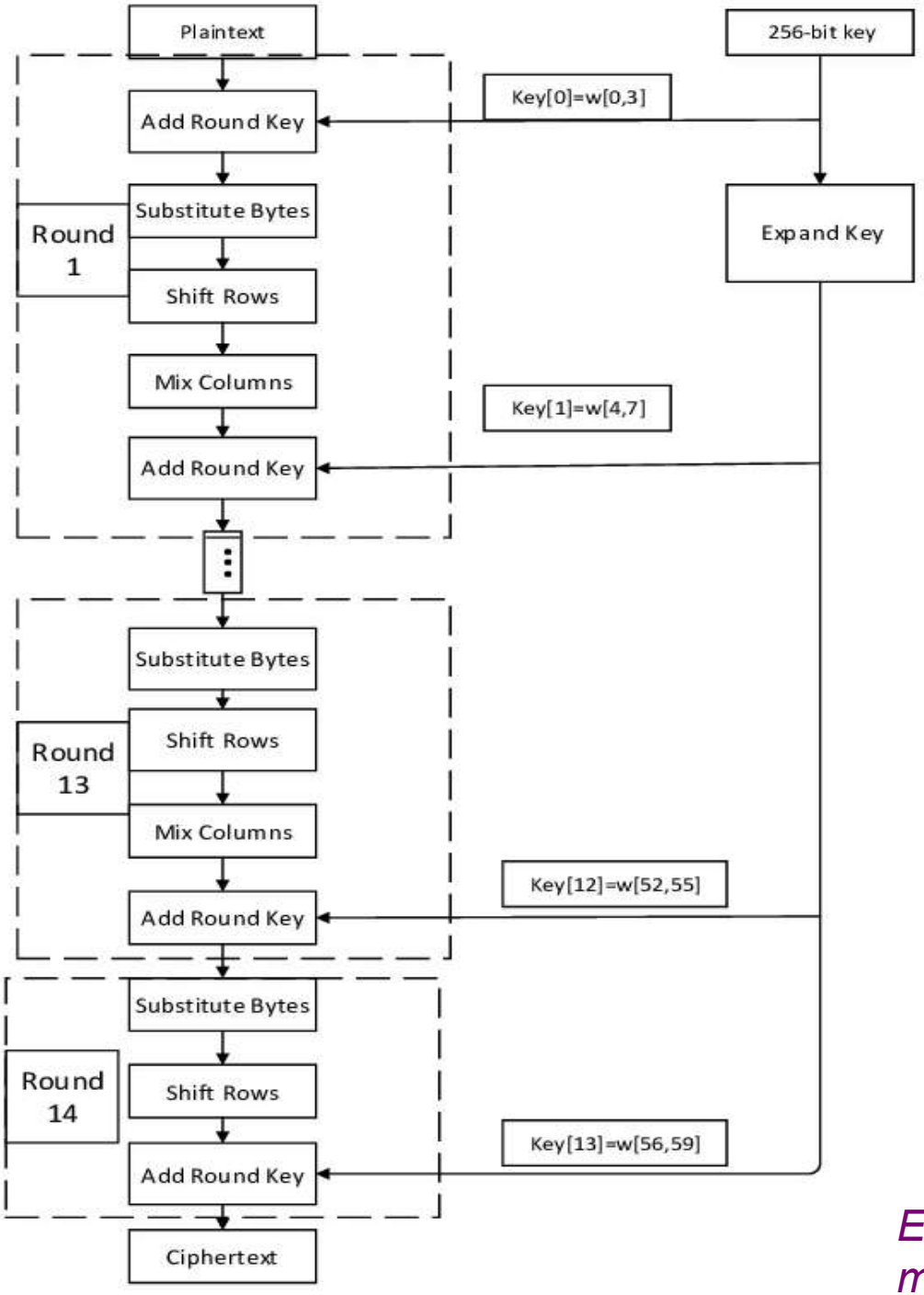
1990 luvun lopulla NIST järjesti avoimen kisan uudesta lohkosalausstandardista. Finaalissa oli 6 ehdotusta, joista Rijndael valittiin uudeksi standardiksi v. 2001. Voittaja sai nimen AES (*Advanced Encryption Standard* AES.

AES:n vahvuudet

* Sillä on useita versioita AES128, AES196, AES256, AES512, joilla se voi vastata myös tulevaisuudessa yhä tehokkaampien tietokoneiden haasteisiin.

* Turvallisuus perustuu todistettavaan, läpinäkyvään matematiikkaan.

Yleisesti ajatellaan, että AES voi toimia lohkosalausstandardina pitkään, vuosikymmeniä.



AES256:n avain on 256 bittinen

Algoritmissa on 14 kierrosta joissa jokaisessa käytetään omaa pääavaimesta johdettua aliavainta.

Kierroksissa on useita vaiheita, joilla toteutetaan diffuusio- ja konfuusio-periaatteita.

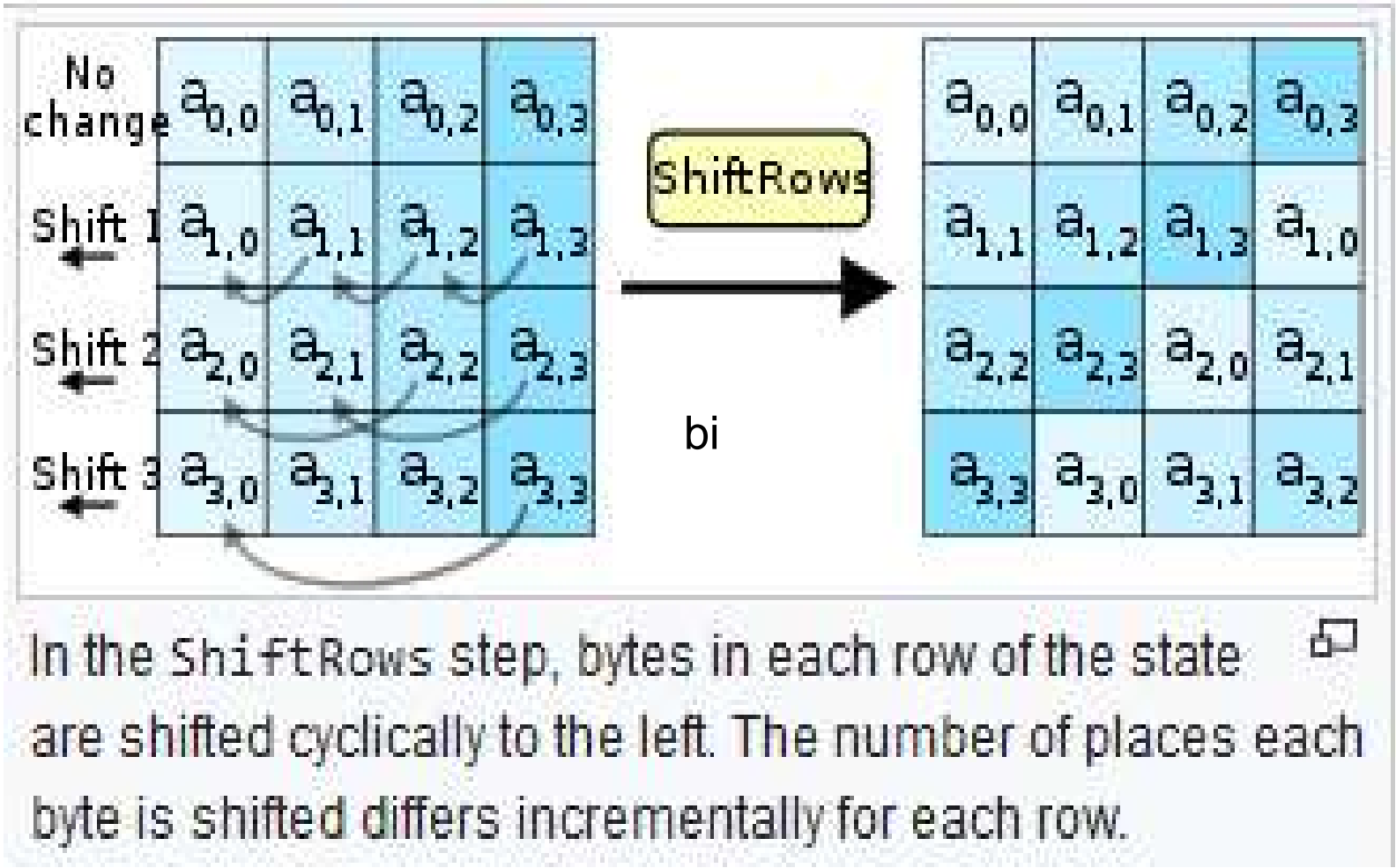
Vaiheet kussakin kierroksessa ovat:
-Substitute bytes (Sbox)
-Shift rows
-Mix columns
-Add round key

Hienoin osa, Sbox perustuu Galoisin teoriaan. Sovelluksen, jolla teoriaa käytetään AES:ssa kehitti 1991 suomalainen kryptologi Kaisa Nyberg

Evariste Galois (1811-1832) was a great French mathematician who died at the age of 21

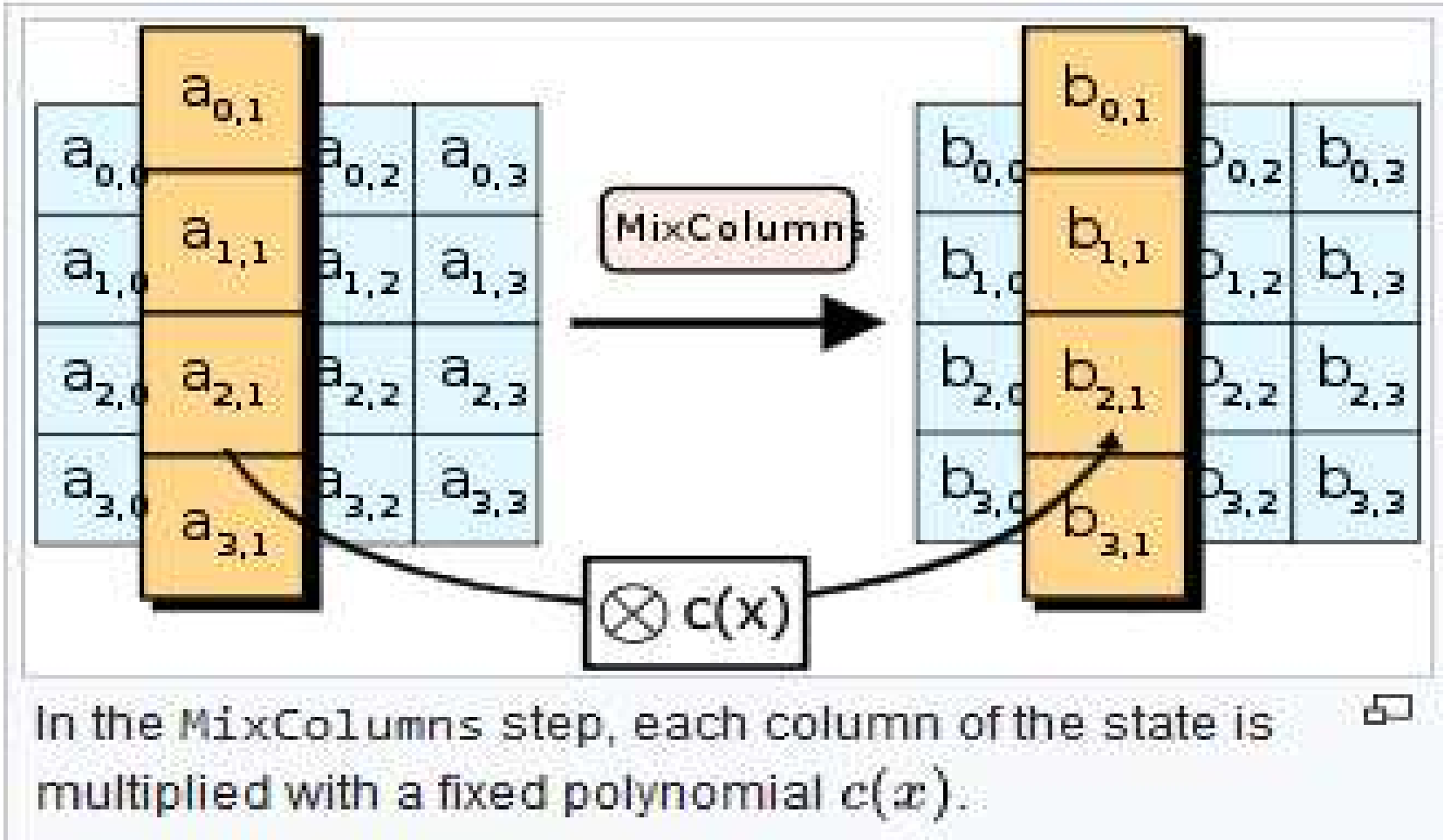
AES kierroksen vaiheet kuvina

ShiftRows step



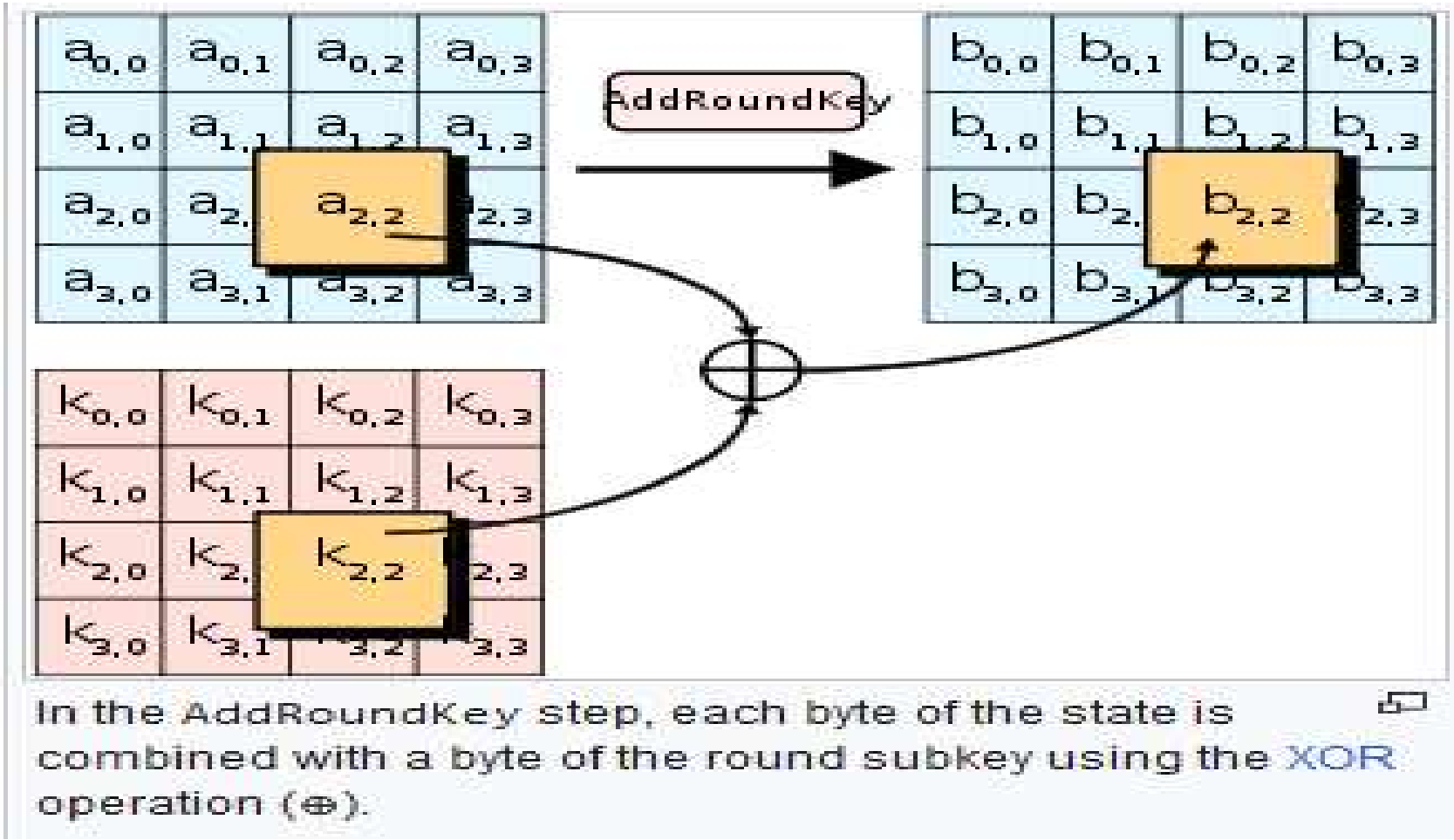
In the ShiftRows step, bytes in each row of the state are shifted cyclically to the left. The number of places each byte is shifted differs incrementally for each row.

MixColumns step



In the `MixColumns` step, each column of the state is multiplied with a fixed polynomial $c(x)$.

AddRoundKey Step



In the AddRoundKey step, each byte of the state is combined with a byte of the round subkey using the XOR operation (⊕).

AES:n substituutiot on toteutettu suomalaisen Kaisa Nybergin algoritmilla, joka käyttää ranskalaisen matemaatikon Evariste Galois'n teoriaa.

Kertolasku a*b Galois'n Kunnassa 2^8, jonka alkiot ovat 8 bitin binäärilukuja.
Esim: a=11010100, b = 01101001

Vaihe 1) a ja b tulkitaan 7. asteen binäärikertoimisiksi polynomeiksi.
$a = x^7 + x^6 + x^4 + x^2$, $b = x^6 + x^5 + x^3 + 1$'

Vaihe 2) Suoritetaan tavanomainen polynomien kertolasku
$(x^7 + x^6 + x^4 + x^2)(x^6 + x^5 + x^3 + 1) = x^{13} + 2x^{12} + x^{11} + 2x^{10} + 2x^9 + x^8 + 3x^7 + x^6 + x^5 + x^4 + x^2$
Kertoimet muutetaan binääriluvuiksi: parillinen → 0 , pariton → 1. Saadaan
$a*b = x^{13} + x^{11} + x^8 + x^7 + x^6 + x^5 + x^4 + x^2$
Vaihe2 voidaan tehdä yhdellä WolframAlpha käskyllä : expand (x7 +x6 +x4 +x2)(x6 +x5 +x3 +1) mod 2 , joka antaa $x^{13} + x^{11} + x^8 + x^7 + x^6 + x^5 + x^4 + x^2$

Vaihe3) Vaiheen 2 tulos jaetaan 8. asteen polynomilla "Field Irreducible polynomial", joka AES:ssa on $x^8 + x^4 + x^3 + x + 1$. Jakojäännös on polynomi $x^6 + x^5 + x^4 + x^3 + x$ vastaten binäärilukua 01111010, joka on tulo a*b.

W.A:lla käytetään komentoa remainder $(x^{13} + x^{11} + x^8 + x^7 + x^6 + x^5 + x^4 + x^2)/(x^6 + x^5 + x^4 + x^3 + x)$, joka antaa jakojäännökseksi $-x^6 + x^5 + x^4 - x^3 + 2x^3 + x$.
Muunnoksella parillinen → 0 , pariton → 1 saadaan binäärikertoiminen polynomi $x^6 + x^5 + x^4 + x^3 + x$, joka vastaa binäärilukua 01111010.

Käyttömoodit: Modes of operation

1) ECB Electronic Codebook

Viestilohkot salataan toisistaan riippumatta => Samoilla viestilohkoilla on aina samat salakirjoituslohkot. Siksi ECB moodi ei ole turvallinen käytettäväksi lohkosalauksessa.

2) CBC Cipher Block Chaining (TLS:ssä vuoteen 2017 asti)

Jokainen salakirjoituslohko toimii lisäsyötteenä seuraavan viestilohkon salauksessa. Siten kunkin lohkon salakirjoitus vaikuttaa seuraavien lohkojen salakirjoituksiin.

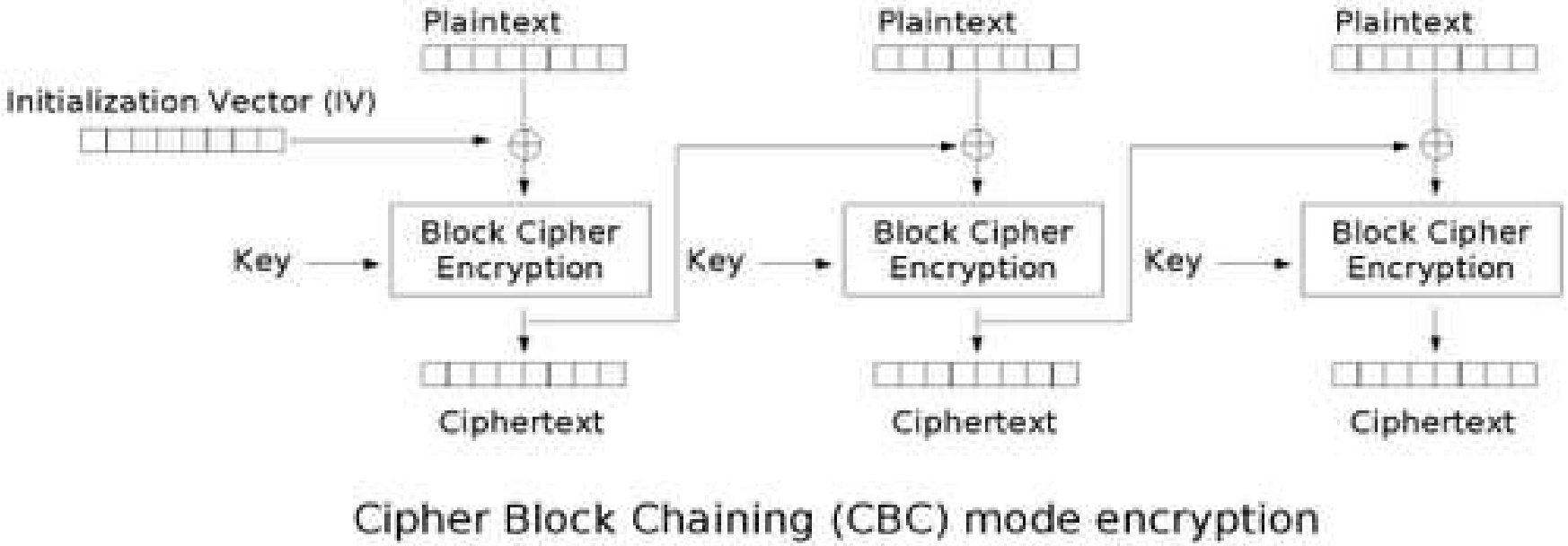
Seuraavassa vuoden 2017 Nordea pankin TLS -yhteyden kuvaus

3) **GCM Galois Counter Mode** (nykyinen moodi TLS:ssä)

Alla v, 2023 Nordea pankin TLS kuvaus Firefox selaimesta

Käyttömoodien AES_CBC ja AES_GCM vertailua

Eräät web-palvelut käyttävätä vielä AES_CBC moodia.



Cipher Block Chaining (CBC) mode encryption

Data salataan 128 bitin lohkoissa. Viestin lisätään XOR yhteenlaskulla edellisen lohkoh salakirjoitust (1. lohkon tapauksessa IV = initialization vector). Se, että kukin salakirjoituslohko vaikuttaa kaikkiin seuraaviin, vahvistaa salausta ja tekee jonkin viestin osan manipuloinnin siirron aikana mahdottomaksi.

GCM – mode (”Galois Counter Mode”)
on korvannut CBC moodin verkkopankeissa. Se tarjoaa vahvemman tiedon eheyden ja lähettäjän identiteetin tarkistamisen kuin edeltäjänsä.

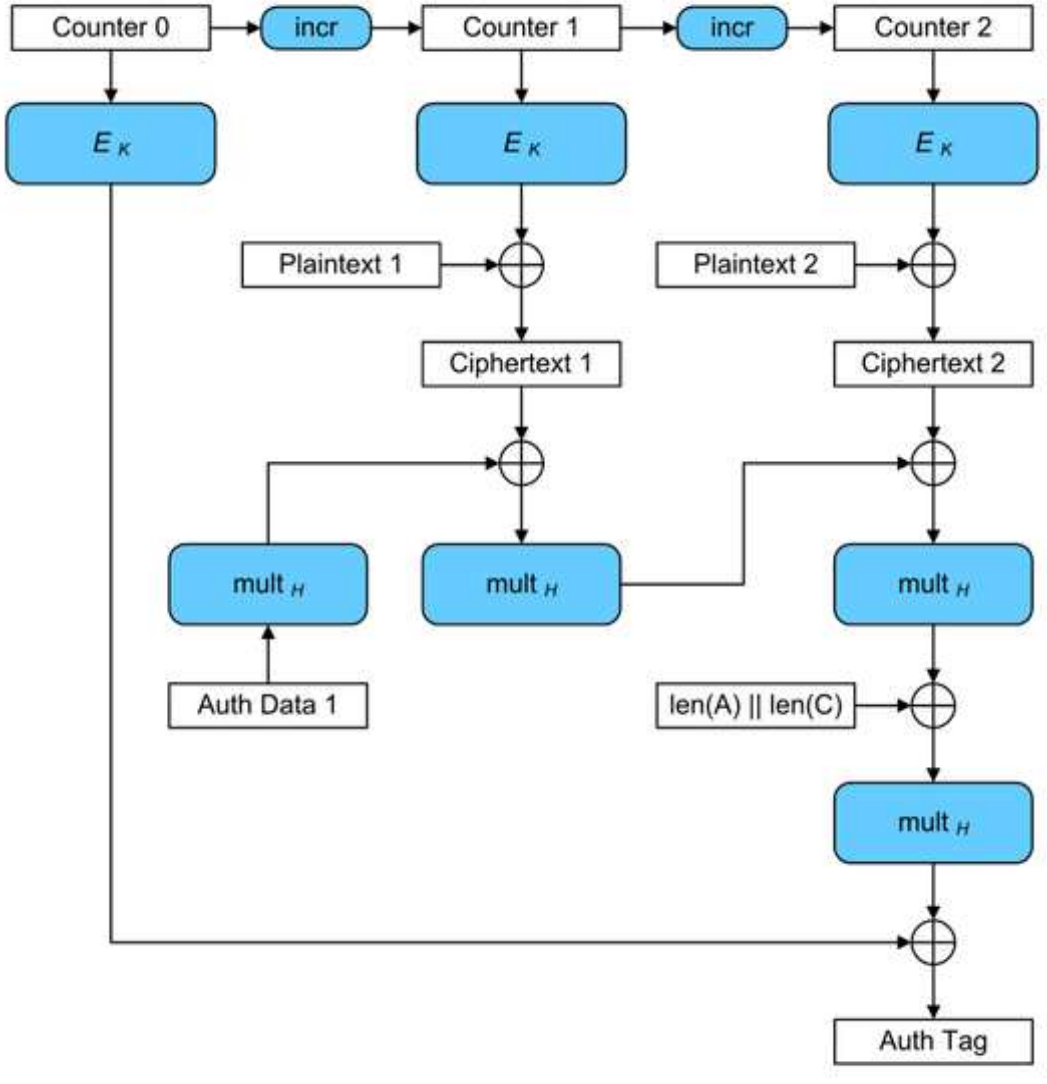
Tekniset tiedot

Yhteys salattu (TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384, 256-bittinen avain, TLS 1.2)

GSM moodissa laskurin arvo (1,2,…) salataan AES:lla käyttäen avainta K. Tulos (128 bits) summataan XOR:lla viestilohkoon m1 tuloksena salakirjoituslohko c1.

Autentikointidata (lähettäjän henkilötiedot + aiemmat salakirjoituslohkot) kerätään 384 bittiseen tiivisteeseen käyttäen Galois’n kunnan kertolaskusääntöä.

Kaiken tavoitteena on estää hyökkäyksiä ja varmistaa tiedonsiirron eheys ja lähettäjän identiteetti.



Suomen lohkosalainsuositukset

Turvaluokka	IV	III	II	I
Sallitut lohkosalaimet	AES 128 Serpent(128)	AES 196 Serpent(196)	AES 256 Serpent(256)	none *)

*) Turvaluokan 1 asiakirjoja ei saa olla elektronisessa muodossa.

In practice AES encryption is used.

Source: Cyper security center of Finland

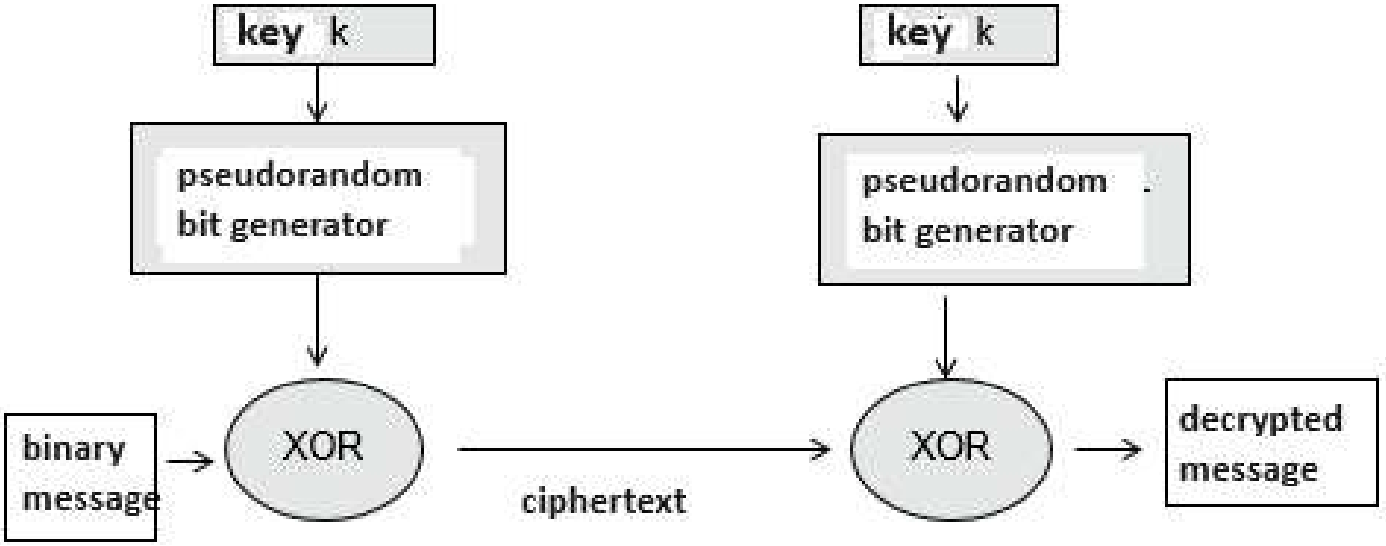
Onko AES ”post-quantum secure” ?

Kvanttitietokoneet tuovat muutoksia salausalgoritmeihin.
Nykytiedon mukaan AES128 ei ole kvanttitietokonemaailmassa enää turvallinen, mutta AES256 ja AES512 antavat vielä riittävän turvamarginaalin *)

*) On olemassa kvanttialgoritmi nimeltä Groverin algoritmi, jolla voidaan murtaa AES128 avain muutamissa sekunneissa käyttäen 128 cubitin kvanttitietokonetta (jota ei ole vielä olemassa).

Kuitenkaan Groverin algoritmilla ei voida murtaa AES256: 256 bittistä avainta. AES256:n turvallisuus jää samalle tasolle kuin nykyään on AES128:lla nykyisessä ympäristössä, jossa ei ole kvanttilaskentaa käytössä.

Synkroninen jonosalaus



GSM puhelujen salaus A5 (1991) **on sykroninen jonosalaus.** Se jäljittelee One Time Pad:n bittiversiota .

GSM puhelimen mikropiiri tuottaa pseudosatunnaista bittijonoa, joka lisättynä XOR yhteenlaskua käyttäen viestin bitteihin tuottaa salatun bittivirran.

GSM puhelu olisi täysin turvallinen, jos avaimena käytetty bittijono olisi täysin satunnainen. Se on kuitenkin deterministinen jono, joka näyttää satunnaiselta.

GSM salaus voidaan GSM murtaa. GSM verkko on korvattu 3G, 4G ja 5G verkoilla, jotka ovat turvallisia.

Pseudosatunnaisuuden käsite

PN- jono (PN = ”pseudo noise”) eli pseudorsatunnainen bittijono on **avaimeen perustuva <u>deterministinen bittijono</u>**, joka kuitenkin täyttää **Samuel W Golombin** 1950 – luvulla esittämät kolme ominaisuutta..

PN jonon ominaisuudet:

G1. Jonossa on n. 50% ykkösiä ja 50% nollia

G2. Samaa bittiä (0 tai 1) toistavien sekvenssien suhteelliset frekvenssit

0**1**0	**or**	1**0**1	single bit block
0**11**0		1**00**1	two identical bits block
0**111**0		1**000**1	three identical bits block
0**1111**0		1**0000**1	four identical bits block

noudattavat suhteita **1/2 , 1/4 , 1/8 , 1/16**

G3. **Kun jonoon tehdään rotaatio** ja rotaatiolla saatua jonoa verrataan alkuperäiseen, tulee yhtä- ja erisuurien bittien lukumäärien olla 50 %. Tätä ominaisuutta mittaa ns. autokorrelaatiokerroin, jonka pitää olla lähellä nolla kaikille rotaatioille. G3 tarkoittaa, että bittijonossa ei ole sisäistä periodisuutta.

Salauksessa tarvittavia lisäominaisuuksia

4. Kaikilla satunnaislukugeneraattoreilla on jokin periodi. (Ne siis tuottavat samat luvut periodin täytyttyä). **Periodin tulisi olla hyvin suuri**

5. Satunnaislukujen generoinnin pitää olla **hyvin nopeaa**

6. Jostakin generoidusta **satunnaisluvusta** ei saa laskea edeltäviä tai seuraavia lukuja.

Satunnaisten salausavainten generointi

Mm. AES:n symmetriset avaimet generoidaan käyttämällä satunnaislukugeneraattoreita.

Mikäli jotkut generaattorien tuottamat salausavaimet olisivat todennäköisempiä kuin toiset, olisi se riski yhteyden turvallisuudelle.

Tyyppejä:

PRNG = pseudo random number generator
- tuottaa deterministisen jonon pseudosatunnaisia lukuja syötteenä annetusta siemenluvusta (seed)
- luvut täyttävät tilastollisen satunnaisuuden vaatimukset, mutta ne eivät ole aidosti satunnaisia. PRNG ei ole kryptografisesti turvallinen.

CSPRNG = cryptographically safe pseudo random number generator.
- soveltuu salausavainten generointiin
- käyttää syötteenä satunnaista dataa, jota löytyy eri lähteistä: mm. tietokoneen aikaa, käyttöjärjestelmän satunnaista dataa.

CSPRNG ”Cryptographically secure pseudo random number generation”

-tarvitaan mm. AES:n symmetristen avainten generointiin

Joitakin CSPRNG menetelmiä

1. **Enthropy source method**. Ennustamatonta datastriimiä kerätään pääasiassa tietokoneen käyttöjärjestelmästä (näppäilyjä, aikaleimoja, keskeytyksiä, anturidataa, ….). Tästä datasta saadaan tiivistefunktiolla satunnaislukuja generaattorin käyttöön.

2. ”**Streching entropy”.** Tavalla 1 saatavan satunnaisdatan määrä on rajallinen, jos tarvitaan nopeasti paljon satunnaislukuja. Määrää voidaan lisätä esim. käyttämällä AES salausta counter moodissa.. Aikaleima voi olla counterin alkuarvo, jonka jälkeen AES tuottaa siitä ja counterin seuraavista arvoista satunnaislukua nopeassa tahdissa.

Voiko satunnaislukugeneraattori olla tietoturvauhka?

Mitä suurimmassa määrin: Jos jotkut ”satunnaisluvut” ovat todennäköisempiä kuin toiset, se lyhentää avaimen murtamisaikaa

Ovatko satunnaislukugeneraattoriin liittyvät uhkat toteutuneet?

Dual ECDRPG oli NIST:n standardoima yleinen satunnaislukugeneraattori OpenSSL:ssä vuoteen 2014. Siinä oli takaportti, jonka Eduard Snowden paljasti.

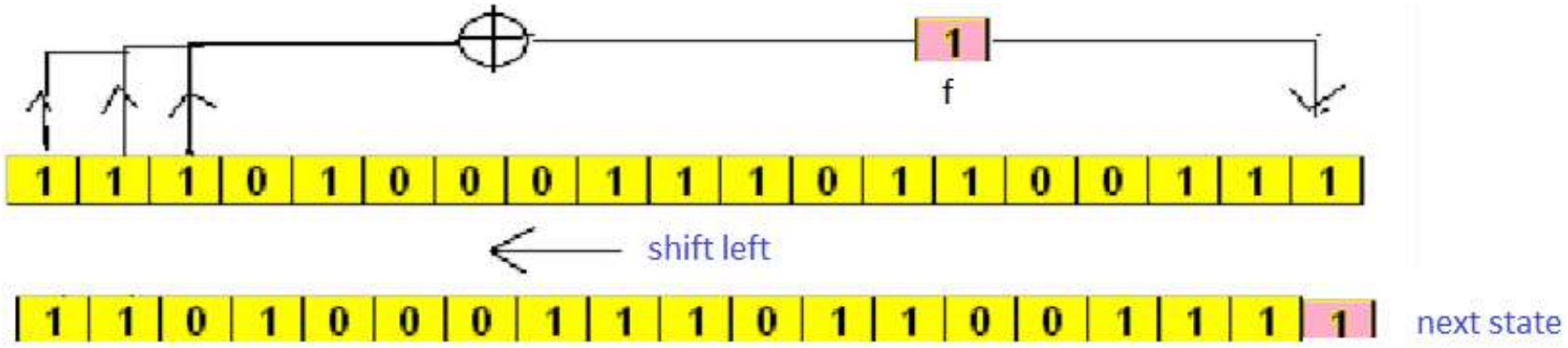
=> Kyseinen generaattori vedettiin 2014 pois standardeista

LFSR - linear feedback shift register

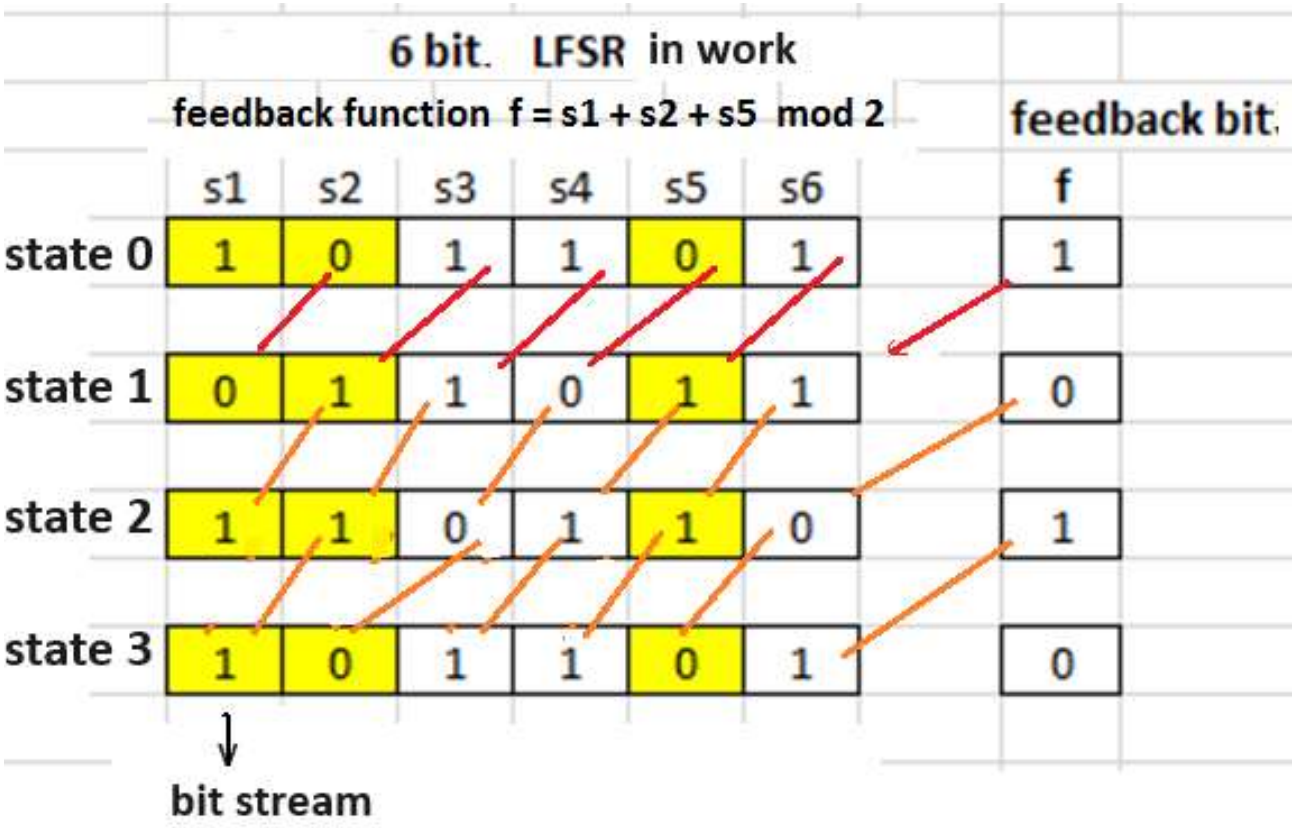
= mikropiiri, jolla satunnaislukuja on tuotettu taskulaskimien RAND funktioissa ja myöhemmin GSM puhelimissa

* LFSR = n bitin rekisteri PN jonojen tuottamiseen.
Jokaisella kellopulssilla bitit siirtyvät askelen vasemmalle.
Uusi oikeanpuolimmaisin bitti (ns. feedback bitti), lasketaan XOR yhteenlaskulla tieyistä rekisterin biteistä ennen niiden siirtymistä vasemmalle

Feedback bit f = 1+ 1+1 mod 2 = 1



Esim. 6 bitin LFSR:n toiminnasta



Feedback bittin f laskukaava on

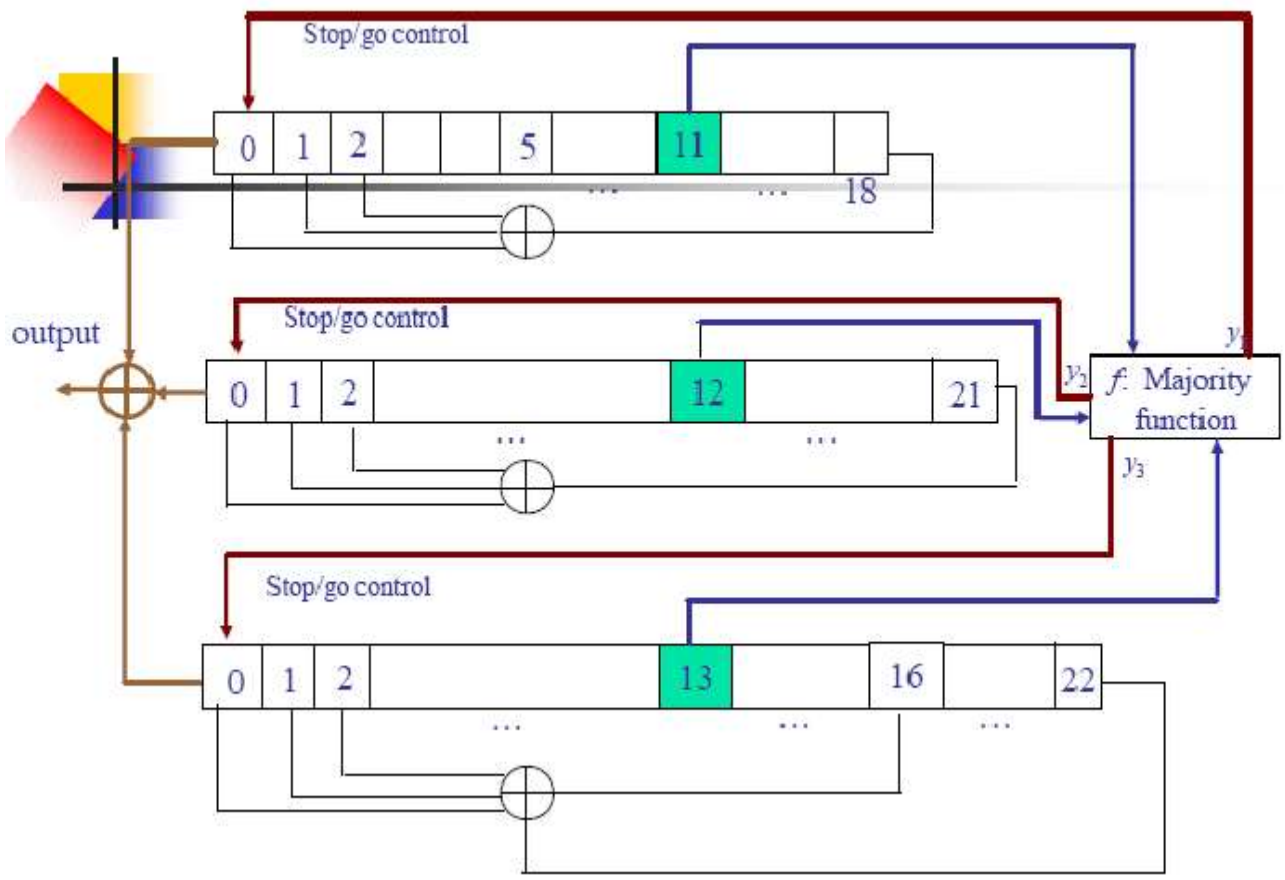
$f = s_1+s_2+s_5 \pmod 2$

(arvot s_i ennen siirtymää)

PN jono saadaan esimerkiksi LFSR rekisteri s1 – biteistä.
Jono toteuttaa Golombin vaatimukset G1- G3. PN jonolle.

LFSR rekisterejä käytettiin GSM puhelimissa. Myös eräiden 3G,4G ja 5G verkkojen SNOW3G salaus käyttää niitä

GSM puhelimen salauksen PRNG



GSM – puhelimessa on 3 LFSR- rekisteriä, pituudet 19, 22 ja 23 bittiä

Rekisterien alkutila (64 bits = 19+22+23) on salausavain

Jokaisella kellopulssilla rekisteri tuottaa yhden pseudo satunnaisbitin (XOR-sum of zero bits).

Puhelu salataan yhdistämällä XOR yhteenlaskulla generaattorin bitit digitoidun puheen bitteihin.

Majority funktio f on piiriin lisätty turvallisuutta lisäävä tekijä.

Majority function f: Jos kuvan vihreillä biteillä on sama arvo, kaikki kolme rekisteriä siirtyvät. Muussa tapauksessa siirtyvät vain ne, joiden vihreillä biteillä on sama arvo..

GSM:n tuottama bittijono toteuttaa tilastollisen satunnaisuuden ominaisuudet G1, G2, G3

0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1,
1, 1, 1, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0, 1, 1, 1, 0, 1,
1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0,
1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1, 1,
1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0,
1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 1, 1, 0, 0, 1,
1, 1, 0, 1, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 0, 0, 0,
0, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 1, 1, 0, 0, 1,
1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1, 0,
1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1, 1,
0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 1, 1, 1,
1, 1, 1, 0, 0, 0, 1, 0, 1, 1, 0, 1, 0, 1, 0, 1, 0, 0,
1, 1, 0, 1, 0, 0, 0, 1, 0, 0, 1, 1, 1, 0, 1, 0, 0, 0,
1, 0, 1, 1, 1, 0, 0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 1,
1, 1, 0, 0, 1, 1, 1, 0, 1, 0, 0, 0, 1, 1, 1, 0, 0, 0,

GSM salaus: Avainjono ja digitaalinen ääni summataan XOR:lla. Tuloksena on salattu signaali

Purku
Operaattorin palvelimella on identtinen PRNG ja sama rekisterin alkutila. Purku tapahtuu XOR:lla samalla tavalla kuin salaus.
GSM puhelu on salattu vain ilmassa, ei kaapelissa.

GSM:n turvallisuus

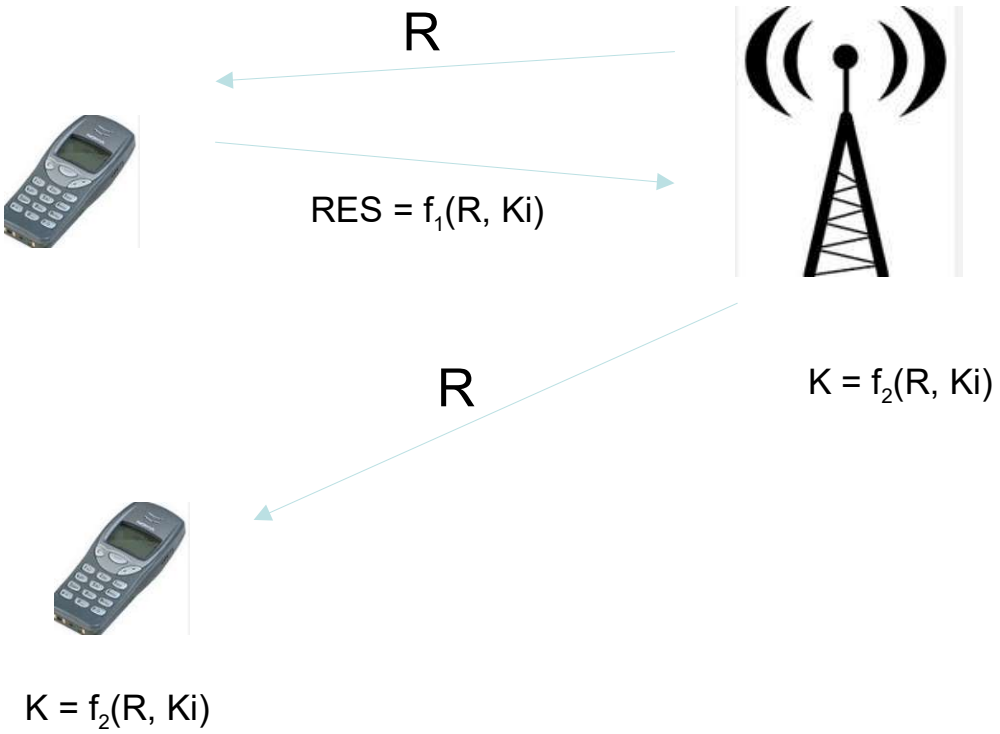
1. Salausavain on vain 64 bittiä, mikä on alle 80 bitin turvaminimin
2. Avainbittien virta on periodinen, todellinen periodi on $4/3*2^{23}$
3. GSM signaali on salattu vain ilmassa, ei enää kaapelissa (jota voidaan salakuunnella)

Käyttäjän autentikointi ja avaimesta sopiminen GSM verkossa

Tämä kalvo on jo osin historiaa, sillä G2- verkkoja ei enää ole. Nykyverkoissa on kaksi-suuntainen autentikointi, jossa on paljon samoja piirteitä kuin GSM;ssä

Autentikointi A3. Operaattori lähettää puhelimelle satunnaisen haasteluvun R. Puhelin laskee vastauksen RES luvusta R ja SIM-avaimesta.

Operaattori laskee RES:n samalla tavoin. Jos luvut täsmäävät, puhelin on todennettu



Key agreement A8. Operaattori lähettää satunnaisluvun R. Puhelin laskee 64 bitin avaimen K luvusta R ja Sim-avaimesta.

Avain K antaa puhelimen LFSR rekisterien alkutilan bitit.

3G, 4G, 5G verkkojen salaus

3G ja 4G käyttävät lohkosalausta: **AES128 tai SNOW3G** *)

- avaimen pituus = **128 bittiä**
- käytössä on kaksisuuntainen autentikointi: sekä puhelin, että verkko todistavat autenttisuutensa vastapuolelle

5G salaus käyttää 256 -bittisiä versioita AES ja SNOW3G salauksista..

*) Snow3G on lohkosalaus, joka on kehitetty Lundin yliopistossa Ruotsissa. Snow3G hyödyntää LFSR rekisteriteknologiaa, jota käytettiin jos GSM puhelimissa.