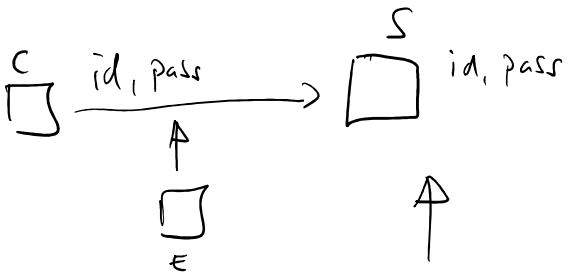
Tutorial XIII

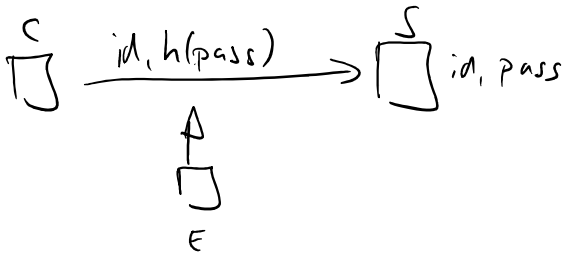
12 December 2019 10:02

identification - Kerberos protocol

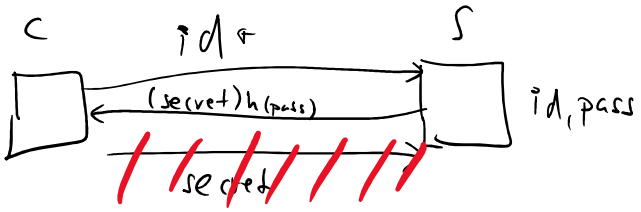
LFSR - linear feedback shift registers

Identification



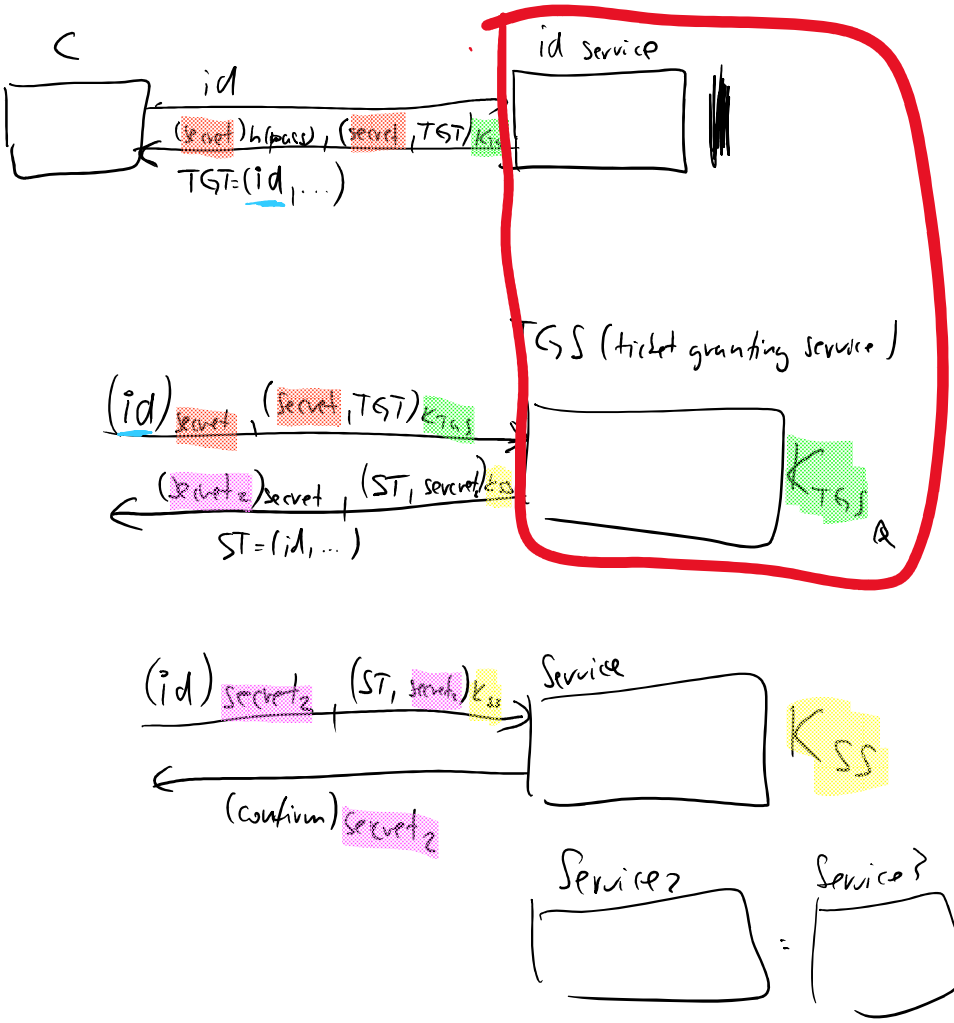


Basic idea



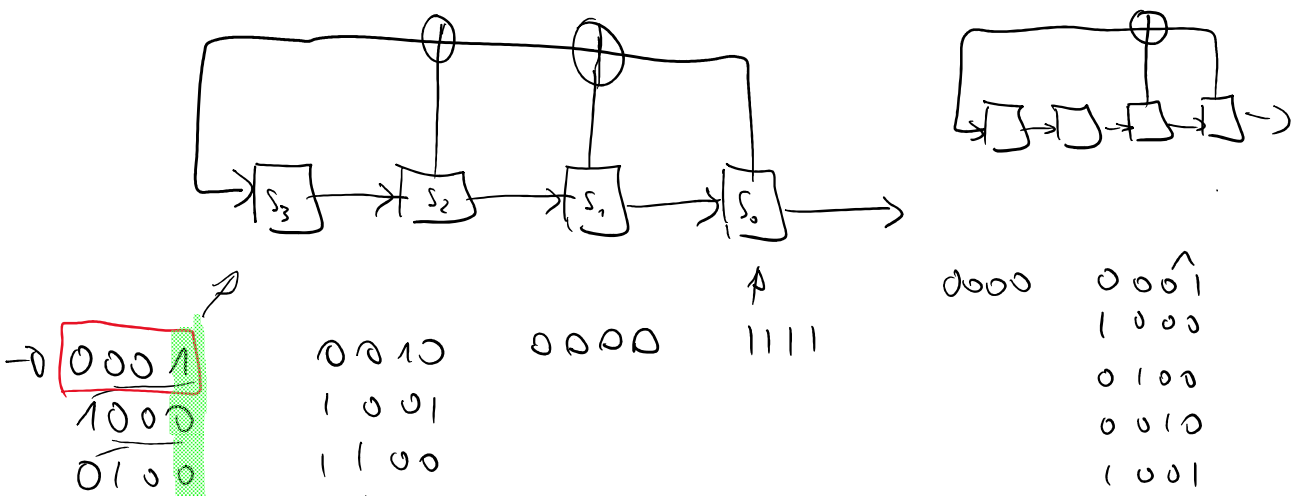
$h(pass) \sim h(pass, salt)$

Kerberos



LFSR - linear feedback shift register

(Pseudorandom number generators)

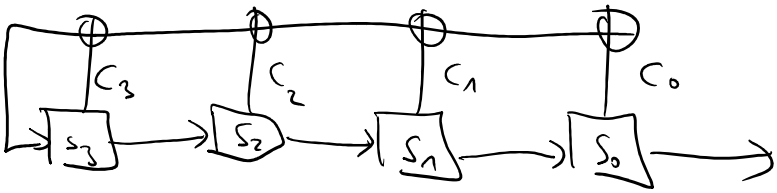


1000
0100
1010
1101
0110
0011
0001

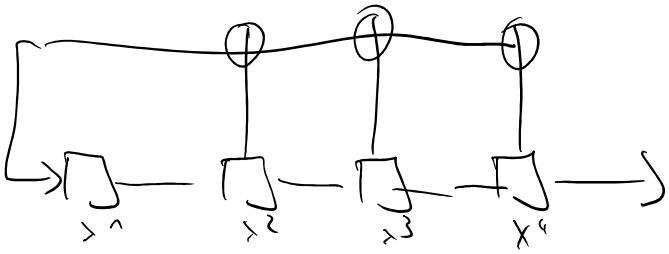
1001
1100
1110
0111
1011
0101
0010

0010
1001
1100
0110
1011
0101
1010
1101
1110
0111
0011
0001

Characteristic polynomials



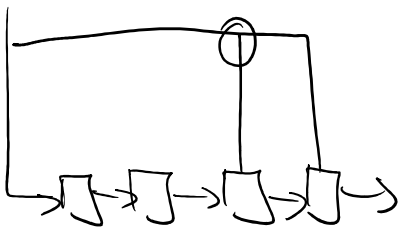
$c_i = 1$ iff s_i is used for feedback



$c_3 = 0 \quad c_2 = c_1 = c_0 = 1$

$f(x) = x^4 + x^3 + x^2 + 1$

$f(x) = \dots + \sum_{i}^{n} c_i x^i + 1$



$c_3 = c_2 = 0 \quad c_1 = c_0 = 1$

$f(x) = x^4 + x^3 + 1$

$$f(x) = \sum_{i=1}^{n} c_{n-i} x^i + 1$$

Thm

LFSR has a period $2^n - 1$ iff $f(x)$ is primitive in $\mathbb{Z}_2$.

irreducible polynomial – cannot be written as a product of 2 polynomials

Thm

each irreducible polynomial divides $x^k - 1$ for some k.
$\left(\deg(f) \le k \le 2^{\deg(f)} \right)$

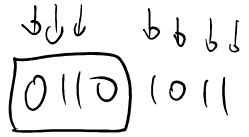
primitive polynomial – irreducible of $\deg(f(x)) = n$,

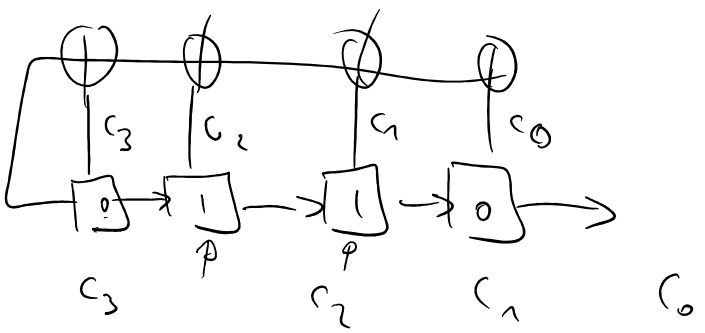
divides $x^k - 1$ for $k = 2^{\deg(f(x))}$

---

seed → LFSR → $2^{\text{seed}} - 1$ key to OTP

Susceptible to known plaintext attack ⇒ Attacker knows part of the key





$c_2 \oplus c_1 = 1$ 0 1 (1)

$c_3 \cdot (c_2 \oplus c_1) \oplus c_1 \oplus c_0$ $c_2 \oplus c_1$ 0 1

$c_3 \left(c_3 \cdot (c_2 \oplus c_1) \oplus c_1 \oplus c_0 \right) + \left((c_2 \oplus c_1) \cdot c_2 + c_0 \right)$ 1 $c_3 \cdot (c_2 \oplus c_1) \oplus c_1 \oplus c_0$ $c_2 \oplus c_1$ 0

$c_3 \cdot \left(c_3 \cdot \left[(c_2 + c_1) + c_1 + c_0 \right] + (c_2 + c_1) \cdot c_2 + c_0 \right) = 1$

Now we can solve 4 equations for $c_0 c_1 c_2 c_3$. We then have LFSR and it's initial vector $\Rightarrow$ whole key.