Network security

- Confidenzialità: solo il sender e receiver devono essere in grado di capire i contenuti dei messaggi. Il sender li critta, il receiver li decritta.
- Autenticazione: sender e receiver vogliono conferma dell'identità reciproca.
- Integrità dei messaggi: sender e receiver vogliono essere sicuri che il messaggio non venga alterato senza essere rilevato.
- accesso e disponibilità: i servizi devono essere accessibili e disponibili agli utenti.

Minacce alla sicurezza

- intercettazione dei messaggi
- inserire messaggi nella connessione
- creare sorgenti finte nei pacchetti
- prendere controllo della connessione togliendo uno tra sender e receiver e mettendosi al posto suo
- non rendendo il servizio disponibile

crittazione di un messaggio

- m: messaggio
- $K_A(m)$: messaggio crittato con chiave K_A
- $m = K_B(K_A(m))$: messaggio decrittato con chiave K_B

Irrompere in uno schema di crittazione

- analizzando il ciphertext
- cerca tutte le possibili chiavi (brute force)
- analisi statistica
- plaintext con parti corrispondenti al ciphertext

crittografia a chiavi simmetriche

Si sostituisce ciascuna lettera del plaintext con un altra sfasata di k posti (cifrario monoalfabetico).
Tuttavia basandosi su informazioni statistiche è possibile decodificare qualche lettera del messaggio, determinando alcuni degli accoppiamenti e quindi riducendo il numero di possibili combinazioni di un fattore di mille o un milione.
La cifratura polialfabetica elimina questo problema usando molteplici sostituzioni monoalfabetiche.

Data Encryption Standard (DES)

Variante di crittografia simmetrica più sofisticata: si divide il messaggio in blocchi della stessa taglia e si usa una permutazione arbitraria per ogni blocco. Questa è la tabella necessaria per fare la codifica/decodifica. Bisogna usare schemi di crittografia dove l'avversario non è in grado di decifrare nulla usando della casualità.

000	**100**
001	111
010	101
100	011
101	010
110	000
111	001

Quindi associata a questa tabella di permutazione si usa uno xor + randomness.

- n_i: blocco i del plaintext
- c_i: blocco i del ciphertext
- r_i: blocco random i
  $c_i = K_S(n_i \text{ xor } r_i)$

$n_i = K_S^{-1}(c_i) \text{ xor } r_i$
La randomness però implica l'invio di messaggi di dimensione doppia, dato che per la decrittazione sono richiesti i blocchi casuali.

Crittografia a chiave pubblica

Il sender e receiver non condividono la chiave di decrittazione. Quella pubblica è conosciuta a tutti, mentre quella di decrittazione è conosciuta solo dal receiver. Quindi tutti possono crittare il messaggio, ma solo il ricevente lo può decrittare.
Data una chiave pubblica K, deve essere impossibile risalire a K^{-1}.

RSA

Dato che ogni messaggio (o sotto-messaggio) può essere rappresentato da una sequenza di bit, si converte il messaggio in binario e poi inviato sottoforma di numero in base 10. Quindi crittare il messaggio equivale a crittare il numero corrispondente.
Il RSA si basa sulla scelta di chiave pubblica e privata, e algoritmi di crittazione e decrittazione.
Per generare le chiavi bisogna

1. Scegliere due numeri primi p e q dell'ordine di 1024 bit.
2. Calcolare $n = p \times q$ e $z = (p-1) \times (q-1)$
3. Scegliere un numero $e, < n, \neq 1$ e che non abbia divisori in comune con z.
4. Trovare un numero $d | \ (e \times d) \mod z = 1$
   Il messaggio c criptato sarà dunque
   $c = m^c \mod n$
   e il messaggio m decrittato sarà
   $m = c^d \mod n$
   Il valore da tenere segreto è d, usato per la decodifica.