symmetric key encryption

la crittografia è in grad di fornire servizi di:

- data integrity traminte **hash functions**
- secrecy tramite **encryption**
- non repudiation tramite **digital signatures**

componenti di un sistema di sicurezza

- **algoritmo** di crittografia: consiste in una serie di operazioni matematiche da applicare ad un determinato messaggio M;
- **funzione** di crittografia: il blocco di codice che si occupa di implementare l'algoritmo
- **protocolli** di sicurezza: regole per la sicurezza della comunicazione

regole fondamentali

- mei usare algoritmi di criptazione sconosciuti
- sapere sempre il sorgente dell'algoritmo (usare algoritmi open source permette di sapere nel dettaglio tutte le operazione che vengono effettuate sui nostri dati)
- affidarsi alle funzioni più famose, manutenute e popolari
- usare protocolli standard
- limitare il servizio agli utenti che adottano l'ultma versione disponibile dei protocolli

funzioni hash

è una funzione unidirezionale che prende in input un messaggio e restituisce in output una stringa di lunghezza fissa (128, 169, 256 bit, etc..)chiamato **digest**. Diversi input producono diversi output. La funziona **SHA** è molto popolare come funzione hash. Una tipica comunicazione consiste nell'inviare il messaggio assieme al suo hash. Il destinatario computa l'hash del messaggio e se entrambi sono validi allora il messaggio non è stato alterato.

La funzione hash deve essere veloce, in quanto si applica sia a messaggi piccoli che a grandi file. La funzione hash, avendo il dominio molto più grande del codominio, significa che possono presentarsi delle **collisioni**: casi in cui diversi messaggi producono lo stesso hash. La probabilità di una collisione è pari a $\frac{1}{2^n}$ dove n rappresenta la lunghezza del digest. Nonostante sia possibile che si verificano collisioni, viene considerato *compuutazionalmente impossibile*.

proprietà:

- trovare le informazioni dato l'hash deve essere **computazionalmente impossibile**
- dato che da un numero qualsiasi di bit, se ne produce uno fisso, molte informazioni vengono perse. La funzione hash deve essere quindi **non invertibile**, anzi, dal digest non si deve dedurre nessuna informazione sul messaggio originale
- se il messaggio cambia anche solo di un bit, il digest deve essere completamente diverso

Se un attaccante intercetta il messaggio è in grado di modificarlo e ricomputare l'hash. Per questo motivo è fondamentale che il messaggio e l'hash siano inviati in due canali diversi.

HMAC i due partecipanti della connessione hanno una chiave condivisa, usata per computare l'hash del messaggio.

HMAC vs hash

- pro: con HMAC posso inviare hash e messaggio nello stesso canale
- pro: viene garantita l'autenticazione (solo chi conosce la chiave può computare l'hash)
- contro: è necessario un canale dedicato per poter decidere la chiave

HMAC fornisce autenticazione e integrità ma non **secretezza**, essa è ottenuta tramite **encryption**.

symetric key encryption

gli elementi necessari per un sistema di encryption sono:

- **cipher**: l'algoritmo di encryption
- **key**: il messaggio che si vuole criptare
- **principio di kerchoffs**
  - il cipher deve essere pubblico
  - il segreto è quindi la chiave

esistono due tipi di algoritmi:

- algoritmi di **sostituzione**: un simbolo è sostituito da un altro
- algoritmi di **trasposizione**: i simboli sono riordinati

One-Time-Pad (OTP) per fare l'encryption del messaggio viene preso un subset della chiave e viene fatto un XOR con il messaggio.

i moderni algoritmi consistono nel combinare i pricipi di **sostituzione** e **trasposizione** per ottenere un algoritmo con le seguenti caratteristiche:

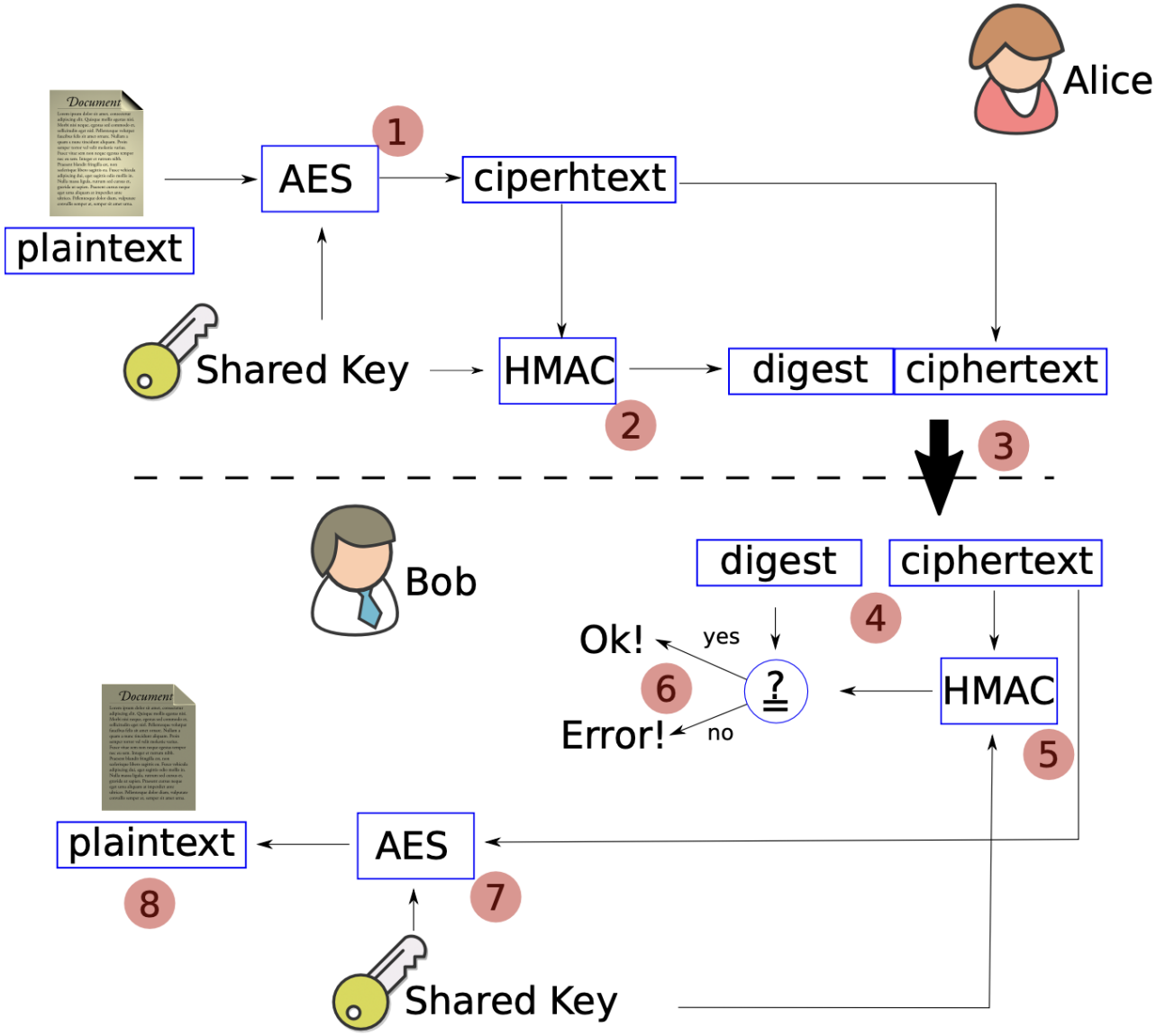
- la lunghezza della chiave è fissa
- la correlazione tra il messaggio originale e quello cifrato è minima

per ottenere questo vengono ripetuti i passaggi di sostituzione e trasposizione per un numero di volte definito.

generazione chiavi la chiave non deve essere facilmente indovinabile, le persone non sono in grado di ricordare stringhe di bit random. Una soluzione consiste nel generare la chiave attraverso una funzione hash

encryption and authentication

una volta che i due host hanno la stessa chiave condivisa possono usarla per creare l'HMAC e per cifrare il messaggio.


Alice
Document
plaintext
1
AES
ciperhtext
Shared Key
HMAC
digest
ciphertext
2
3
Bob
digest
ciphertext
4
Ok!
yes
6
?
HMAC
Error!
no
5
Document
plaintext
AES
8
7
Shared Key