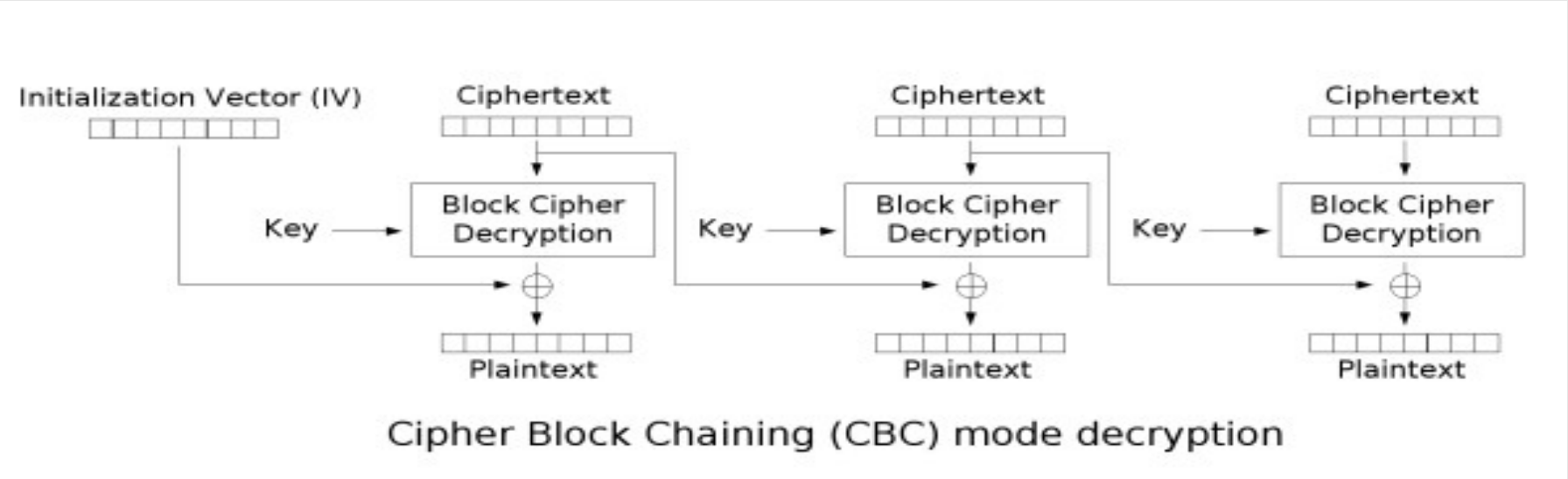
Crittografia avanzata
Lezione dell'11 Aprile 2011

Malleabilità

- Proprietà di un cryptosystem di permettere di intervenire su parti del messaggio cifrato ottenendo effetti desiderati sul testo cifrato
  - es. bit flipping
- Presente anche nei “modes”: es. CBC: dato che il blocco n è ottenuto dalla cifratura del blocco n di testo, in XOR con il blocco cifrato n-1, modificare quest'ultimo comporta una modifica nel testo decifrato (a spese di un errore nel blocco n-1)



Cipher Block Chaining (CBC) mode decryption

Effetto valanga

- Piccole modifiche del testo comportano modifiche diffuse del testo cifrato
  - Riconducibile al concetto di diffusione
  - strict avalanche criterion (SAC): cambiando un bit del testo, ogni bit del testo cifrato ha il 50% di probabilità di cambiare
  - bit independence criterion (BIC): cambiando un bit del testo, il cambiamento di due qualsiasi bit del testo cifrato è indipendente

Crittoanalisi differenziale

- Studia come cambiando bit nel plaintext cambiano i bit nel ciphertext
- Serve per riconoscere dove l'algoritmo esibisce comportamenti non-random
  - Relazioni fra i plaintext portano a relazioni fra i ciphertext
- Chosen-ciphertext attack per recuperare (informazioni sulla) chiave
- Usata per studiare componenti non lineari (es. S-box)
- Può essere contrastata utilizzando molte trasformazioni (es. passando da molte S-box) in modo che la

Subliminal channel

- Detto anche “covert channel”
- Nasconde informazione in un canale/messaggio legittimo
- Esempi:
  - Telnet over ping
  - Spazi in un testo
  - query/response http
- Nota: non necessariamente prevede cifratura

Deniable encryption

- La possibilità di negare l'esistenza di dati cifrati, o di poterli decifrare
  - Più praticamente, la possibilità di decifrare i dati in un messaggio innocuo con una seconda chiave
- Analogo a meccanismi di autenticazione che prevedono due password, una normale e una che autentica ma avverte che si è sotto coercizione

Meccanismi

- Esempio banale
  - M1 = messaggio segreto
  - M2 = messaggio innocuo
  - K1 = chiave per il messaggio segreto
  - K2 = M2 ⊕ K1 ⊕ M1
  - Dato il messaggio cifrato K1 ⊕ M1, se si fornisce come chiave K2 si ottiene M2
  - Senza K1 non si ottiene M1
    - Non è “plausibile”, lo XOR è evidentemente falsificabile
- Ci sono meccanismi più sofisticati e scalabili

Filesystem

- I dati cifrati non sono distinguibili da quelli casuali
- Si inizializza il filesystem con dati casuali
- Si scrivono i file cifrati (nessuna differenza evidente)
- Nelle aree vuote si mettono una FAT e dei file innocui in posizioni “coerenti”

Steganografia

- La crittografia nasconde il contenuto di un messaggio
- La steganografia nasconde la presenza stessa del messaggio (cifrato o meno)
- Es. Modifica di un bit nel colore/luminosità di un’immagine
- Senza la chiave non è possibile affermare che esiste il messaggio

Steganalisi

- Permette di evidenziare la presenza di un messaggio
- Es. modifica anomala della palette di un’immagine

Watermarking

- È difficile impedire la copiatura di software/immagini/CD/DVD…
- È possibile inserire un'informazione nei file mediante steganografia
- Scopi:
  - scoprire chi ha duplicato il file
  - inserire un marchio indelebile
- E dopo che lo si è scoperto?
- Collusion-resistant watermarking

Bit(s) commitment con funzioni hash

- Problema: produrre una stringa S, e fornire all'istante t la prova di averla prodotta, senza fornirla; all'istante t+x si scopre S
- Soluzione: si fornisce H(S) all'istante t
  - H non è invertibile ed è collision-resistant, quindi:
  - Alice non può produrre S' tale che H(S')=H(S)
  - Bob non può risalire a S da H(S)

Zero-knowledge proofs interattive

- Problema: dimostrare di conoscere la soluzione di un problema senza rivelarla
- Soluzione: Alice usa un numero casuale per trasformare il problema in un altro, isomorfo
- Alice risolve il nuovo problema (sfruttando la soluzione che conosce e l'isomorfismo) e fa il commit della soluzione
- Bob sceglie se:
  - Chiedere di dimostrare che i problemi sono isomorfi
  - Chiedere si vedere la soluzione al secondo problema
    - Se vedesse entrambi conoscerebbe anche la soluzione al primo
- Bob chiede di ripetere l'operazione fino a quando non è rassicurato dal calcolo delle probabilità
- Un isomorfismo è una funzione biunivoca fra due strutture algebriche tale che la funzione e la sua inversa siano omomorfismi, ovvero preservino le proprietà delle operazioni su tali strutture

Prove interattive (cut and choose)

- Alice conosce il segreto S e lo può usare con un numero (casuale) C per fare A(C) e B(c):
  - Se può fare A(c) e B(c), allora conosce S
  - Se Bob conosce il risultato di A e B, allora conosce C
  - Bob è in grado di verificare la correttezza di A(c) e B(c)
- Bob chiede ad Alice di fare A(c) e B(c), si fa rivelare a scelta A(c) o B(c) (es. A(c)) e lo verifica
- Alice potrebbe aver “tentato” di calcolare solo A(c), con il 50% di probabilità di indovinare la richiesta di Bob
- Bob chiede di ripetere il test n volte, con n numeri diversi; Alice ha $1/2^n$ probabilità di indovinare tutte le richieste di Bob
- Attenzione al man-in-the-middle...

Blind signatures

- Problema: Alice vuole far firmare a Bob qualcosa, senza che Bob sappia cos`é (es. notariato)
- Si usa un blinding factor che si possa eliminare dopo la firma
- Es. con RSA:
- e=chiave pubblica, d=chiave privata, m=messaggio, n è un modulo pubblico, k è il blinding factor 1<=k<=n noto ad Alice
- $t = mk^e \bmod n$;
- Alice fa firmare t a Bob $t^d = (mk^e)^d \bmod n \equiv m^d k \bmod n$
- Alice elimina il blinding factor dividendo per k

HDCP e dot product

- $[a_1 ... a_n] \bullet [b_1 ... b_n] = \sum a_i b_i$
- La funzione è lineare
- Per ogni coppia di coppie di numeri u_a, v_a, e u_b, v_b, si ha che $u_a \bullet v_b = u_b \bullet v_a$
- Chiavi private di 40-56 bit
- Bastano 40 coppie di chiavi per poter costruire il sistema di equazioni necessario per rompere il sistema