Esame di Fondamenti di Cybersecurity

03-06-2024

Leggere attentamente ogni punto del regolamento prima di svolgere l'esame, non rispettare queste regole comporterà l'annullamento (anche in corso) dell'esame:

1. Non è ammesso nessun tipo di materiale, cartaceo o elettronico, questo va da materiale "ufficiale" del corso come slide o registrazioni a materiale autoprodotto o semi prodotto (e.g. appunti o soluzioni di esercitazioni). Chat GPT rientra in questa categoria.
2. Non è ammesso parlare con altre persone via qualsiasi canale, l'esame e' individuale.
3. Scrivete **Nome, Cognome e matricola** su **TUTTI** i fogli tranne questo foglio di istruzioni, negli spazi indicati.
4. E' necessario presentare il badge universitario.
5. I punteggi di ogni domanda sono riportati a fianco della domanda stessa, il massimo punteggio ottenibile tramite questo esame scritto e' 24. L'esame si considera superato se la somma del punteggio di questo esame con il punteggio delle esercitazioni risulta essere maggiore o uguale a 18.
6. La durata della prova e' di **un'ora e 40 minuti**.
7. Potete usare il retro del foglio come brutta copia o considerazioni aggiuntive sulla vostra risposta.
8. L'esame va scritto tramite PENNA NERA o PENNA BLU. Non è possibile usare penne rosse o matite o bianchetto.
9. Rispondete alle domande in maniera esaustiva ma concisa.
10. La consegna dell'esame scritto invalida i precedenti voti. Per ritirarsi all'esame bisogna scrivere **"NON VALUTARE**" su TUTTI i fogli ad esclusione di questo foglio di istruzioni.

Fondamenti di Cybersecurity 03-06-2024

Nome: JOCELYNE Cognome: ELIAS Matricola:

1. Alice usa il crittosistema **RSA** per ricevere messaggi da Bob. Alice sceglie:
- p=11, q=19
- il suo esponente pubblico è e=7

Alice pubblica il prodotto n=pq=209 e l'esponente e=7
- a) Verificare che e=7 è un esponente valido per l'algoritmo RSA
- b) Calcolare d, la chiave privata di Alice

Bob vuole inviare ad Alice il testo P=14, cifrandolo
- c) Che valore Bob invia ad Alice?
- d) Verificare che Alice riesca a decifrare tale messaggio. **(6 punti / 24)**

SOLUZIONE:

Calcoliamo:

Phi(n)= (p-1)(q-1) = (11-1)(19-1)=180

180= (2^2)*(3^2)*5

a) poiché **gcd(7,180)=1**, la scelta di **e** è valida

b) L'esponente privato **d*e = 1 mod(Phi(n)) o d =e^-1 mod(Phi(n))**= 7^-1 mod(180)=**103**

Infatti 103*7= 721 che modulo 180 dà 1 (4*180+1 = 721)

c) Ciphertext inviato da Bob a Alice C= 14^7 mod(209) = **174**

Dettagli del calcolo di C = 14^7 mod(209) usando l'algoritmo di *square-and-multiply*:

b = e = 7 = 1 1 1 in rappresentazione binaria.

i	**(b = 111) b_i**	**z**	**z**
2	1	1^2 14 mod 209	14
1	1	14^2 14 mod 209	27
0	1	27^2 14 mod 209	**174 = C**

d) Alice decifra il plaintext Plaintext P =174^103 mod (209) = **14**

Dettagli del calcolo di P = 174^103 mod(209) usando l'algoritmo di *square-and-multiply*:

b = d = 103 = 1100111 in rappresentazione binaria.

i	**(b = 1100111) b_i**	**z**	**z**
6	1	1^2 174 mod 209	174
5	1	174^2 174 mod 209	179
4	0	179^2 mod 209	64

Fondamenti di Cybersecurity 03-06-2024

Nome: JOCELYNE Cognome: ELIAS Matricola:

3	0	64^2 mod 209	125
2	1	125^2 174 mod 209	78
1	1	78^2 174 mod 209	31
0	1	31^2 174 mod 209	**14 = P**

2. Consideriamo lo ***Shift Cipher*** e supponiamo che le chiavi vengono utilizzate con la stessa probabilità. Dimostrare che lo *Shift Cipher* fornisce la *Perfect Secrecy*. **(5 punti / 24)**

Definizione di Perfect Secrecy*: A cryptosystem has perfect secrecy if Pr[x | y] = Pr[x] for all $x \in P$, $y \in C$. That is, the a posteriori probability that the plaintext is x, given that the ciphertext y is observed, is identical to the a priori probability that the plaintext is x (P is the set of plaintexts and C is the set of ciphertexts, K is the set of keys).*

Le 26 chiavi nello Shift Cipher sono utilizzate con uguale probabilità (= 1/26). Quindi, per qualsiasi distribuzione di probabilità del plaintext, il Shift Cipher ha una segretezza perfetta.
Vediamo la dimostrazione alla pagina successiva!

Fondamenti di Cybersecurity 03-06-2024

Nome: JOCELYNE Cognome: ELIAS Matricola:

$P = C = K = \mathbb{Z}_{26}$, $0 \le k \le 25$

$e_k(x) = (x + k) \bmod 26 \quad (x \in \mathbb{Z}_{26})$

$$Pr[y = \bar{y}] = \sum_{k \in \mathbb{Z}_{26}} Pr[K = k] \times Pr[x = d_k(\bar{y})]$$

$$= \sum_{k \in \mathbb{Z}_{26}} \frac{1}{26} \times Pr[x = y - k]$$

$$= \frac{1}{26} \sum_{k \in \mathbb{Z}_{26}} Pr[x = y - k]$$

$$\sum_{k \in \mathbb{Z}_{26}} Pr[x = y - k] = \sum_{x \in \mathbb{Z}_{26}} Pr[x = x] = 1$$

$$\Rightarrow \boxed{Pr[y = y] = 1/26}, \quad \forall y \in \mathbb{Z}_{26}$$

Poi, $Pr[y/x] = Pr[K = (y - x) \bmod 26]$

$= 1/26$ per ogni $x, y \in \mathbb{Z}_{26}$

la chiave è unica tale che

$e_k(x) = y : k = (y - x) \bmod 26$.

Adesso, Bayes' theorem:

$$Pr[x/y] = \frac{Pr[x] \times Pr[y/x]}{Pr[y]} = \frac{Pr[x] \times 1/26}{1/26}$$

$$\boxed{Pr[x/y] = Pr[x]}$$

Dunque, abbiamo perfect secrecy.

Fondamenti di Cybersecurity 03-06-2024

Nome: JOCELYNE Cognome: ELIAS Matricola:

3. Descrivere in cosa consiste un attacco di tipo ***buffer overflow***. Quando è possibile attuarlo? Esistono precauzioni o contromisure? **(4 punti / 24)**

Buffer overflow, noto anche come buffer overrun o buffer overwrite, è definito nel NISTIR 7298 (Glossary of Key Information Security Terms, Luglio 2019) come segue:
una condizione su un'interfaccia in cui è possibile inserire più input in un buffer o un'area di conservazione dei dati rispetto alla capacità assegnata, sovrascrivendo altre informazioni.
Gli avversari sfruttano tale condizione per mandare in crash un sistema o per inserire codice appositamente predisposto che consente loro di ottenere il controllo del sistema.

Precauzioni o contromisure: (Guardare le slide del corso)

- Write safe code!
- Stack Canaries - StackGuard
- Stack non eseguibile
- Address-space Layout Randomization (ASLR)

4. Spiegare il meccanismo **WEP** (Wired Equivalent Privacy).
    a. In quale standard senza fili/wireless viene utilizzato per assicurare l'autenticazione e la confidenzialità?
    b. Su quale algoritmo di cifratura si basa WEP?
    c. Quale sono le debolezze di questo algoritmo di cifratura?
    d. Discutere un esempio di attacco che può avere luogo con WEP **(5 punti / 24)**

Risposte: (Per maggior dettagli, guardate le slide "Wireless network security")
a. WEP: Wired Equivalent privacy è la prima forma di autenticazione e confidenzialità delle parti nell'ambito delle reti locali senza fili che usano WiFi (lo standard 802.11).
b. WEP si basa su RC4, sviluppato da Ron Rivest nel 1987. RC4 è un Stream Cipher (cifrario a flusso) basato sulla chiave k e sul vettore di inizializzazione (IV) v, genera un keystream (flusso di chiavi) RC4(v,k).
c. Il problema di RC4 sta nella chiave che viene utilizzata come input per generare il keystream (articolo interessante: https://blog.cryptographyengineering.com/2011/12/15/whats-deal-with-rc4/).
d. Dopo 30000 pacchetti trasmessi nella rete le probabilità di collisione sono molto elevate, praticamente impossibili da evitare (the birthday paradox problem)!
$\text{Probability}_{\text{collision}} \approx 1$
Catturando pacchetti con lo stesso IV è possibile fare attacchi statistici per ricavare la chiave segreta. (vedere "Two-time pad attack", slides#40-41 della lezione **OTP and Stream Ciphers**)
Inoltre, catturando pacchetti con un IV noto è possibile far circolare pacchetti vecchi (replay attack) aumentando il traffico …

Fondamenti di Cybersecurity 03-06-2024

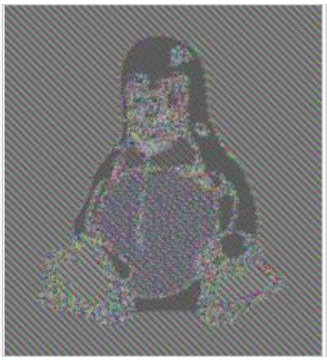
Nome: JOCELYNE Cognome: ELIAS Matricola:

5. Quali sono i problemi della modalità ECB? **(4 punti / 24)**

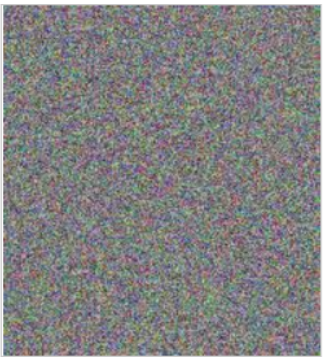
ECB è **deterministico**
Il messaggio originale è suddiviso in blocchi indipendenti (P_1, P_2, … P_N).
La stessa chiave "K" è utilizzata per cifrare/decifrare
Ogni blocco è cifrato separatamente/indipendentemente

Lo stesso blocco di dati viene sempre cifrato nello stesso modo (*m* cifrato con *k* produce sempre lo stesso *c*): questo rivela pattern quando i dati si ripetono!
Questo è lo stesso problema che abbiamo visto con il cifrario di Vigenère
ECB non è semanticamente sicuro per i messaggi che contengono più di un blocco (**known-plaintext attack**).
Esempio illustrativo:

Plain text



Cipher text with **ECB**



Cipher text with **other modes of operation**