Prezime, Ime	JMBAG

Bodovi
/30

Kriptografija i Kriptoanaliza — Međuispit

3. prosinca 2021.

1. (2) Klasična kriptografija.
   Dekriptirati tekst `FČMIJZ` koji je kriptiran *Playfairovom* šifrom s ključem `OVOJEMOJKLJUČ` ako se koristi sljedeći alfabet s 25 znakova:

   `A B C Č Ć DĐ-(D ili Đ) E F G H I J K L M N O P R SŠ-(S ili Š) T U V Z Ž.`

   Rješenje: Iz ključa se generira matrica 5×5:

O	V	J	E	M
K	L	U	Č	A
B	C	Ć	DĐ	F
G	H	I	N	P
R	SŠ	T	Z	Ž

   Dekriptiraju se parovi slova: FČ, MI, JZ, te imamo DĐA JP ET = DAJ PET!

2. (4) Napadi na kriptosustave.

   (a) (1) Koliko ukupno ulaznih razlika te koliko ukupmo izlaznih razlika treba razmotriti u postupku diferencijalne kriptoanalize nekog simetričnog kriptosustava s jednom supstitucijskom tablicom koja mijenja jedan bajt na ulazu s novim bajtom na izlazu? Odgovor: Ukupan broj mogućih ulaznih razlika iznosi _____ i ukupan broj izlaznih razlika iznosi _____.
   256/256

   (b) (1) Koje svojstvo nekih simetričnih kriptoalgoritama omogućuje napad pretraživanje pola prostora rješenja?
   $C = DES(M, K)$ i $C' = DES(M', K')$, gdje je X' oznaka za bitovni komplement vrijednosti X

   (c) (0.5) Prilikom napada poznatim čistim tekstom (*known-plaintext attack*) napadač (**zaokružiti** točan odgovor)

   a) ima na raspolaganju neke jasne tekstove M
   b) ima na raspolaganju neke parove jasni i kriptirani tekst (M,C)
   c) može dekriptirati poruku za po svojoj volji odabrani kriptirani tekst C
   d) može kriptirati poruku za po svojoj volji odabrani jasni tekst M

   (b)

   (d) (0.5) Za koliko se bitova efektivno smanjuje veličina ključa ako za neki kriptosustav vrijedi sa vjerojatnošću 100% izraz: $\mathsf{M}[1,5,19,3]$ XOR $\mathsf{C}[2,3,8,9,11,19] = \mathsf{K}[2,5,19]$, gdje je primjerice $K[2,5,17]$ = drugi bit ključa XOR peti bit ključa XOR sedamnaesti bit ključa?
   Za 1 bit,

   (e) (1) Ukratko opisati napad jednostavnom analizom potrošnje električne energije na neki uređaj na kojem se izvodi kriptoalgoritam AES gdje je cilj doznati duljinu ključa.
   AES koristi 3 veličine ključa: 128, 192 i 256 bitova te prema veličini ključa obavlja se kriptiranje i dekriptiranje u 10, 12 ili 14 krugova pa treba samo izbrojati te krugove.

3. (9) Kriptosustavi DES i AES.

   (a) (4) Pretpostavimo da kriptiramo dva toka podataka koristeći „Output Feedback" (OFB) tako da u oba toka iskoristimo isti inicijalizacijski vektor (IV). Neka je prvi kriptirani blok prvog toka $C_1 = (\texttt{10 39 23 3C 26})_{Hex}$ i neka je prvi kriptirani blok drugog toka $C_2 = (\texttt{19 3C 23 30 26})_{Hex}$. Ako napadač zna da je prvi blok jasnog teksta prvog toka jednak $M_1 = (\texttt{4C 69 76 65 73})_{Hex}$, što time može zaključiti o prvom bloku jasnog teksta drugog toka M_2? Napadaču je na raspolaganju sljedeća tablica.

ASCII Char	Binary	Hex	ASCII Char	Binary	Hex
L	01001100	4C	<	00111100	3C
l	01101100	6C	FF	00001100	0C
i	01101001	69	9	00111001	39
v	01110110	76	DLE	00010000	10
E	01000101	45	NUL	00000000	00
e	01100101	65	DLE	00010000	10
s	01110011	73	#	00100011	23
HT	00001001	09	&	00100110	26
ENQ	00000101	05	EM	00011001	19
0	00110000	30			

Rješenje: Neka je OFB blok toka za kriptiranje. S obzirom da koristimo isti inicijalizacijski vektor IV, vrijedi da je OFB jednak za prvi i drugi tok pa imamo:

$$C_1 = M_1 \oplus OFB$$
$$C_2 = M_2 \oplus OFB$$

Iz toga vrijedi da je $C_1 \oplus C_2 = M_1 \oplus M_2$, stoga $M_2 = C_1 \oplus C_2 \oplus M_1$.

$$\begin{aligned}
C_1 &= (\texttt{10 39 23 3C 26})_{hex} \\
&= (\texttt{00010000 00111001 00100011 00111100 00100110})_2 \\
C_2 &= (\texttt{19 3C 23 30 26})_{hex} \\
&= (\texttt{00011001 00111100 00100011 00110000 00100110})_2 \\
C_1 \oplus C_2 &= (\texttt{09 05 00 0C 00})_{hex} \\
&= (\texttt{00001001 00000101 00000000 00001100 00000000})_2 \\
M_1 &= (\texttt{4C 69 76 65 73})_{hex} \\
&= (\texttt{01001100 01101001 01110110 01100101 01110011})_2 \\
&= (\texttt{L i v e s})_{ascii} \\
M_2 &= C_1 \oplus C_2 \oplus M_1 \\
&= (\texttt{45 6C 76 69 73})_{hex} \\
&= (\texttt{01000101 01101100 01110110 01101001 01110011})_2 \\
&= (\texttt{E l v i s})_{ascii}
\end{aligned}$$

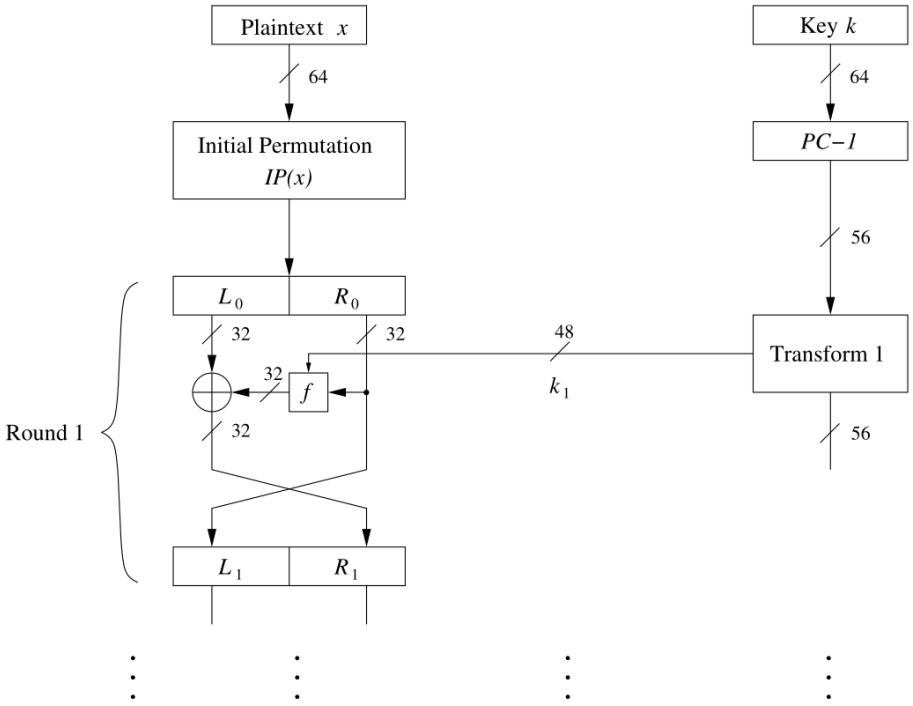
Stoga, napadač može zaključiti da je prvi blok jasnog teksta drugog toka jednak

$$\begin{aligned}
M_2 &= (\texttt{01000101 01101100 01110110 01101001 01110011})_2 \\
&= (\texttt{E l v i s})_{ascii}
\end{aligned}$$

Primjetite da smo odmah, bez računanja, mogli zaključiti da se slova "v" i "s" nalaze u drugoj poruci! Da bi zadatak bio priznat dovoljno je naći neku (ASCII, binarnu ili heksadecimalnu) reprezentaciju od M_2.

(b) (2) Razmatramo svojsto difuzije u AES-u. Pretpostavimo da u matrici $M_{4\times4}$ jasnog teksta promijenimo *prvi* bajt $M(0,0)$. Koji stupci matrice bloka će se promijeniti nakon *prve runde* AES-a kao posljedica promjene $M(0,0)$? Obrazložite.
Navedena promjena prvog bajta $M(0,0)$ uzrokovat će promjenu samo u prvom stupcu. Operacije unutar AES-a koje osiguravaju difuziju su *shiftRows* i *mixColumns*. Operacija *shiftRows* prvi redak rotira za 0, stoga će $M(0,0)$ ostati na svom mjestu. S druge strane, operacija *mixColumns* množi polinom svakog stupca s fiksnom matricom koja uzrokuje promjenu u tom stupcu. Prema tome, promjena $M(0,0)$ će se nakon *mixColumns* proširiti samo na prvi stupac.

(c) (2) Skicirajte i matematički opišite jednu iteraciju Feistelove mreže.
Rješenje:
Pogledati sliku 1.



Slika 1: Jedna iteracija Feistelove mreže

$$L_0 = IP[1:32]$$
$$R_0 = IP[33:64]$$
$$L_1 = R_0$$
$$R_1 = L_0 \oplus f(R_0, k_1)$$

(d) (1) Objasnite kako se dekriptira poruka kod simetrične enkripcije zasnovane na Feistelovoj mreži.
Rješenje:

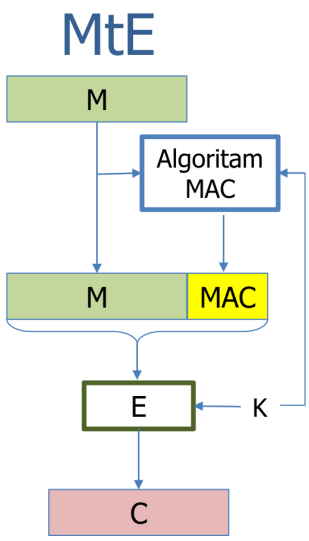
$$R_i = L_{i+1}$$
$$L_i = R_{i+1} \oplus f(L_{i+1}, k_i)$$

4. (3) Hash i Autentifikacijsko kriptiranje

(a) (2) Skicirati algoritam kriptiranja MAC-then-Encrypt (MtE).
Rješenje:
Pogledati sliku 2.

(b) (1) Pretpostavimo da želimo naći koliziju u funkciji sažetka s 128-bitnim izlazom. Koliki je ugrubo očekivani broj sažetaka koje trebamo napraviti da bismo s vjerojatnošću od 50% našli koliziju? Kako nazivamo ovaj napad?
Trebamo ugrubo napraviti $2^{128/2} = 2^{64}$ sažetka da bismo s vjerojatnošću od 50% našli koliziju. Riječ je o rođendanskom napadu.

5. (4) Kriptiranje toka podataka i generatori slučajnih brojeva.
Generator pseudoslučajnih brojeva G radi tako da na početku postavi stanje na zadano sjeme, te računa novo stanje tako da primjenjuje kriptografsku hash funkciju `SHA256`, ali samo na *zadnja dva bajta* trenutnog stanja. Točnije, pseudokod generatora G je sljedeći.



Slika 2: MAC-the-Encrypt

```
S = seed
ponavljaj:
  S = SHA256(zadnja_dva_bajta_od_S)
  ispisi(prvi_bajt_od_S)
```

(a) (2) Je li G siguran generator pseudoslučajnih brojeva? Ako je, obrazložite zašto. Ako nije, detaljno opišite napad koji pokazuje da G nije siguran generator pseudoslučajnih brojeva.
Rješenje:

i. Napad 1
Generator pseudoslučajnih brojava G nije siguran, sljedeći niz koraka predstavlja napad.
   A. Uzmemo neki prefiks ispisanog niza od $G(k)$ koji koristi nama nepoznato početno stanje (sjeme) k, npr prvih 100 bajtova od $G(k)$. Na temelju ovog prefiksa nastojimo predvidjeti sljedeće izlaze od $G(k)$.
   B. Za svako početno stanje (sjeme) $i \in 2^{16}$ generatora G ispišemo prvih 100 bajtova od $G(i)$. S obzirom da `SHA256` izgleda slučajno, astronomski je mala vjerojatnost da ćemo za dva različita sjemena dobiti isti niz od 100 bajtova.
   C. Ako imamo podudaranje, onda znamo da je $k = i$, tj. znamo sjeme k i možemo predvidjeti sljedeće bajtove od $G(k)$.

ii. Napad 2
   A. Uzmemo neki prefiks duži od 2^{16} bajtova ispisanog niza od $G(k)$ koji koristi nama nepoznato početno stanje (sjeme) k.
   B. S obzirom da postoji samo 2^{16} različitih stanja, ovo znači da postoji jedno stanje koje će se sigurno ponoviti (pigeonhole principle).
   C. Kad se to stanje ponovi onda ce se i ispis ponoviti. Napadac tada moze uočiti uzorak (ciklus) i predvidjeti ispis bez znanja početnog sjemena k.

(b) (2) Pretpostavimo da koristimo G u protočnoj enkripciji E, detaljno opišite jedan napad koji pokazuje da E nije sigurna enkripcija.
Rješenje:
E nije sigurna enkripcija. Napadač može napraviti sljedeće.

i. Ako napadač sazna prefiks poruke m is skriveni tekst E, napadač može dobiti $\mathsf{pref}(G(k)) = \mathsf{pref}(m) \oplus \mathsf{pref}(E(m,k))$. Ovdje također možemo pretpostaviti da napadač iskoristi chosen-plaintext model napada da sazna do sada ispisani $G(k)$.

ii. Sada kada napadač zna (prefiks od) $G(k)$, gornjim postupkom može saznati početno stanje (sjeme) k i predvidjeti $G(k)$.

iii. Predviđanjem $G(k)$ napadač može dekriptirati sljedeće bajtove poruke m.

6. (8) Asimetrični kriptosustavi.

(a) Pretpostavimo da je riječ o kriptosustavu RSA (bez nadopunjavanja i sažetka) s javnim ključem $pk = (3, 55)$ i privatnim ključem $sk = (27, 55)$.

i. (1) $\varphi(N) = \varphi(55) = \varphi(5 * 11) = (5 - 1) * (11 - 1) = 40$.

ii. (1) Pokažite da je par (sk, pk) javnog i privatnog ključa korektan.
$3 * 27 = 81 = 1 + 2 * 40 = 1 \mod \varphi(N)$.

iii. (1) Odredite enkripciju poruke (broja) 2. $2^3 = 8 \mod 55 = 8$.

iv. (2) Odredite dekripciju poruke (broja) 5 koristeći algoritam uzastopnog kvadriranja. Obavezno navesti postupak.
Rješenje: $27 = 16 + 8 + 2 + 1 = (11011)_2$

i	4	3	2	1	0
$a[i]$	1	1	0	1	1
d	5	15	5	15	**25**

(b) (1) **Zaokružite** sljedeće probleme za koje smatramo da ih nije moguće riješiti u razumnom vremenu, čak i ako na raspolaganju imamo jako velike računalne resurse. U svakom od problema pretpostavljamo da su zadani brojevi proizvoljni, da su veličine 2048 bitova i da nisu poznate nikakve druge informacije o njima.

a) Rastaviti broj a na proste faktore.

b) Odrediti je li broj a složen.

c) Pronaći sve faktore prostog broja p.

d) Izračunati Eulerovu funkciju $\varphi(a)$ za broj a.

e) Izračunati modularni inverz $a^{-1} \pmod b$ za brojeve a i b.

f) Izračunati modularno potenciranje $b^a \pmod c$ za brojeve a, b i c.

(a) i (d)

(c) (1) Precizno definirajte jedan siguran način kombiniranja sustava enkripcije javnim ključem RSA i simetrične enkripcije AES128CBC. Traži se nešto od sljedeća tri odnosno nešto vrlo slično:

- $\mathsf{RSA}(K, PK)$, $\mathsf{AES128CBC}(M, K)$
- $\mathsf{RSA}(Pad(K), PK)$, $\mathsf{AES128CBC}(M, K)$
- $\mathsf{RSA}(\texttt{materijal\_za\_kljuc}, PK)$, $\mathsf{AES128CBC}(M, Hash(\texttt{materijal\_za\_kljuc}))$

Prihvatljivo je da student ne napiše kompaktno rješenje nego objasni korake. *Potrebno je napisati da je K odnosno materijal za ključ slučajno generiran..*

(d) (1) Objasnite barem dva razloga zbog kojih kombiniramo sustave enkripcije javnim ključem i simetričnu enkripciju.
Traže se neka dva od sljedeća tri odnosno nešto ekvivalentno ili vrlo slično njima:

- Sustav enkripcije javnim ključem zna kriptirati samo brojeve, a mi želimo kriptirati proizvoljne poruke.
- Sustav enkripcije javnim ključem je nekoliko redova veličine sporiji pa želimo njime kriptirati samo ključ, a dugačku poruku kriptirati bržim simetričnim sustavom.
- Sustav enkricije javnim ključem nije siguran ako kriptiramo proizvoljne poruke pa želimo njime kriptirati samo slučajne ključeve ili materijal za ključ.