CS.201P: Introduction to Computer Security 3/7/2019

Homework 4

Due Thursday 3/14/2019, 11:59pm, by upload to gradescope.com

Solve the following problems from the [GT] textbook:

1. Exercise R-8.1, R-8.2, and R-8.4. In each case just name a class of attacks which the given attack falls into.

R-1 Eve has tricked Alice into decrypting a bunch of ciphertexts that Alice encrypted last month but forgot about. What type of attack is Eve employing?

Solution: This is a known-plaintext attack.

R-2 Eve has an antenna that can pick up Alice's encrypted cell phone conversations. What type of attack is Eve employing?

Solution: This is a ciphertext-only attack.

R-4 Eve has bet Bob that she can figure out the AES secret key he shares with Alice if he will simply encrypt 20 messages for Eve using that key. For some unknown reason, Bob agrees. Eve gives him 20 messages, which he then encrypts and emails back to Eve. What kind of attack is Eve using here?

Solution: This is a chosen-plaintext attack.

2. Exercise R-8.5.

R-5 What is the encryption of the following string using the Caesar cipher: THELAZYFOX.

Solution: In the ancient cryptosystem, the Caesar cipher, each Latin letter of a plain- text was substituted by the letter that was three positions away in a cyclic listing of the alphabet, that is, modulo the alphabet size. So we get WKHODCBIRA.

3. Exercise R-8.6.

R-6 What are the substitutions for the (decimal) numbers 12, 7, and 2 using the S-box from Figure 3?

Solution: According to figure 3:

	00	01	10	11
00	0011	0100	1111	0001
01	1010	0110	0101	1011
10	1110	1101	0100	0010
11	0111	0000	1001	1100

(a)

	0	1	2	3
0	3	8	15	1
1	10	6	5	11
2	14	13	4	2
3	7	0	9	12

(b)

Figure 3: A 4-bit S-box (a) An S-box in binary. (b) The same S-box in decimal. This particular S-box is used in the Serpent cryptosystem, which was a finalist to become AES, but was not chosen.

So, we have the following calculation:

Number	Binary in (a)	First pair	Second pair	Substitution in (a)
12	1100	11	00	0111 = 7
7	0111	01	11	1011 = 11
2	0010	00	10	1111 = 15

So we get 7, 11, 15.

4. Exercise R-8.7.

R-7 What are the next three numbers in the pseudo-random number generator $3x_i + 2 \bmod 11$, starting from 5?

Solution:

x0 = 5
x1 = (3* 5 +2) mod 11 = 6
x2 = (3 * 6+2) mod 11 = 9
x3 = (3* 9 +2) mod 11 = 7

5. Exercise R-8.9.

Assume that if H is a hex alphabet, i.e. H = {0,1, ...,9, a, ..., f} and Sbox encodes permutation $\pi : H^2 \rightarrow H^2$ (btw, note that H^2 encodes 8-bit strings, so π can be thought of as a permutation on bytes), then π(x), for $x = x_1x_2$ where $x_1,x_2 \in H$, is written in the x_1-th row and x_2-th column of the S-box table. So, for example, the S-box in Figure 8.14 encodes $\pi : H^2 \rightarrow H^2$ s.t. π(00) = 63, π(01) = 7c, π(02) = 77, π(05) = 6b, etc.

R-9 In the inverse of the S-box from Figure 14, what is the substitution for e3, in hexadecimal?

Solution: As can be seen from the table below, e3 has a substitution of 4d.

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Figure 14: The S-box used in the SubBytes step of AES. Each byte is shown in hexadecimal notation, which encodes each 4-bit string as a digit 0–9 or a–f. Each byte is indexed according to the first and second 4-bits in the byte to be transformed.

6. Exercise R-8.10.

R-10 What would be the transformation done by three consecutive applications of the ShiftRows step in the AES encryption algorithm?

Solution:

In every round of the AES encryption, there are 4 steps:

1st = Sub byte
2nd = Shift Row
3rd = Mix column
4th = Add round key

And in the ShiftRow, which is actually permutation, the 0th line of the state matrix is shifted to the right by 0 bytes, the 1st line is shifted to the right by 1 byte, the 2nd line is shifted 2 bytes to the right, and the 3rd line is shifted to the right by 3 bytes. For example:

$$\begin{bmatrix} b_{0,0} & b_{0,1} & b_{0,2} & b_{0,3} \\ b_{1,0} & b_{1,1} & b_{1,2} & b_{1,3} \\ b_{2,0} & b_{2,1} & b_{2,2} & b_{2,3} \\ b_{3,0} & b_{3,1} & b_{3,2} & b_{3,3} \end{bmatrix} \rightarrow \begin{bmatrix} b_{0,0} & b_{0,1} & b_{0,2} & b_{0,3} \\ b_{1,1} & b_{1,2} & b_{1,3} & b_{1,0} \\ b_{2,2} & b_{2,3} & b_{2,0} & b_{2,1} \\ b_{3,3} & b_{3,0} & b_{3,1} & b_{3,2} \end{bmatrix}$$

$$= \begin{bmatrix} c_{0,0} & c_{0,1} & c_{0,2} & c_{0,3} \\ c_{1,0} & c_{1,1} & c_{1,2} & c_{1,3} \\ c_{2,0} & c_{2,1} & c_{2,2} & c_{2,3} \\ c_{3,0} & c_{3,1} & c_{3,2} & c_{3,3} \end{bmatrix}$$

[referenced from textbook]

7. Bob argues that if you use symmetric key encryption twice in a row using the same key then the encryption would be more secure, i.e. that given encryption scheme E you can create a stronger encryption E' defined like this:

$E'(k, m) = E(k,(E(k, m)))$

Show that Bob is (very!) wrong, by pointing out a well-known encryption scheme E which is perfectly secure, but which makes E' completely insecure.

Solution:

Assuming you have a binary one-time pads, XOR. It is very safe and never able to decrypted if the attack does not know the pads if right using. But if you encrypted twice, you just obtain the original plaintext as you decrypted it yourself! For detail,
C = M XOR P,
But M = C XOR P.

8. Exercise R-8.16. Hint: See the textbook or lecture slides on how random prime generation uses primality testing as a subprocedure. You don't have to understand how primality testing subprocedure works, just assume that it works as a black-box, i.e. that calling PrimalityTester(x) returns 1 if x is prime and 0 if x is non-prime.

(In practice primality testers are probabilistic, i.e. they can return 1 while x is not a prime, but they can be tuned so that this happens with infinitesimaly small probabil- ity, so in this exercise assume that the primality tester gives perfect answers.)

R-16 Roughly how many times would you have to call a primality tester to find a prime number between 1,000,000 and 2,000,000?

Solution:

- Using prime-counting function to count the number of prime numbers less than or equal to some real number x , x/ln(x).
- So there are roughly 2000000/log(2000000) – 1000000/log(1000000) = 137848.73 – 72382.41 = 65466 prime numbers between 1000000 and 2000000.
- So in the range of 2000000 – 1000000 = 1000000 numbers, we need to call expectively 1000000/65466 = 15.3 times to call a assuming right prim number.

9. Exercise R-8.14.

R-14 Compute the multiplicative inverse of 5 in Z_{21}.

Solution:

Z21	0	1	2	3	4	5	6	7	8	9
5	0	5	10	15	20	4	9	14	19	3

Z21	10	11	12	13	14	15	16	17	18	19
5	8	13	18	2	7	12	17	1	6	11

So the smallest multiplicative inverse of 5 in Z21 is 17.
And the general form is 17+ 21*k, k ∈ Integer.

10. Exercise R-8.15.

R-15 What is 7^{16} mod 11?

Solution:

P = 11	X^1 mod p	X^2	X^3	X^4	X^5	X^6	7	8	9	10
X = 7	7	5	2	3	10	4	6	9	8	1

P = 11	X^11 mod p	X^12
X = 7	7	5

It begins to circulation，so 7^16 mod 11 is as 7^6 mod 11 = 4.

11. Exercise R-8.17.

R-8.17 What is 7^{120} mod 143?

Solution:

According to Euler's Theorem, since 120 is relatively prime with 143, so 7^120 mod 143 = 1.

12. Exercise R-8.18.

R-18 Show the result of encrypting $M = 4$ using the public key $(e, n) = (3, 77)$ in the RSA cryptosystem.

Solution:

C = 4^3 mod 77 = 64

13. Why (e, N) where e = 1 is not a good RSA public key?

Solution:

Given M, C = M^e mod N = M mod N, it is easy to leak the message of M, especially when N is large, c is very likely to = M.

What's more, since d*e = 1 mod (p-1)(q-1), if e = 1, then d = 1 mod (p-1)(q-1), attackers will obtain the secret key of d easily.

14. Why (e, N) where e = 2 is not a good RSA public key?

(This one is harder...)

Solution:

Given M, C = M^e mod N = M^2 mod N, the first thing is similar to the above question, which is that it is also too small for encryption, making the attacker easily to break.

Another reason is that with e=2 we get 4 possible roots mod N=pq. (with the help from Chinese Remainder Theorem) For arbitrary elements, this works pretty much the same: Calculate the roots in both mod$p$ and mod$q$, then combine the result. So the question comes for the choice which root was the correct one has to be done outside the encryption scheme.

15. Exercise R-8.21.

R-21 Show the result of an Elgamal encryption of the message $M = 8$ using $k = 4$ for the public key $(p, g, y) = (59, 2, 25)$.

Solution:

A = 2^4 mod 59 = 16
B = 8 * (25^4) mod 59 = 6
So (a, b) = (16,6)

16. Exercises R-8.22 and R-8.23.

R-22 Demonstrate that the hash function

$$H(x) = 5x + 11 \bmod 19$$

is not weakly collision resistant, for $H(4)$, by showing how easy it is to find such a collision.

R-23 Demonstrate that the hash function

$$H(x) = 5x + 11 \bmod 23$$

is not strongly collision resistant, by showing how easy it is to find such a collision.

Solution:

H(4) = 31mod 19 = 12

And for the module 19, the next 4 + 19 = 23 will cause a collision, so the period is 19 to find a collision, which is too small.

Similarly, the period is 23, so a consecutive 24 trails will cause a collision, so it is not strongly collision resistant.

17. Exercise C-8.5.

C-5 Alice is using a linear congruential generator, $ax + b \bmod 13$, to generate pseudo-random numbers. Eve sees three numbers in a row, 7, 6, 4, that are generated from Alice's function. What are the values of a and b?

Solution:

Since for the LCG, knowing 3 consecutive numbers will know the a, b from the property of linear.

So, (7a + b) = 13l + 6, (6a + b)= 13j + 4.

So, a = 2, b = 5

18.

18. Assume Bob uses textbook RSA encryption, i.e. given public key $pk = (e, N)$ Bob's encryption $E(pk, m)$ of message m outputs $c = m^e \bmod N$, and that Bob uses it on messages which are small positive integers, i.e. Bob's message space M is $M = \{1, 2, ..., t\}$ for some small threshold t.

    For any t, describe an attack which works in time $O(t)$ and given ciphertext c and public key (e, N) recovers plaintext m.

Solution:

Since the possible message space is small, we can do brute force dictionary attack in O(t) time to find out the plaintext m for given c. For more detailed operation, we can use the public key to generate dict of all the possible message Possible = Mi ^e mod N, and compare the ciphertext c in the dict.

19.

19. Assume that Bob uses textbook RSA as above but t is large-enough so the above attack is not feasible. Show instead that if Bob sends $c = E(pk, m)$ to Alice over an insecure communication link then a man-in-the-middle attacker Mallory, can intercept Bob's ciphertext c and send to Alice ciphertext c' so that Alice always decrypt $2m$ as Bob's message instead of m. Describe Mallory's algorithm, which outputs c' on inputs c and $pk = (N, e)$, and always succeeds in this attack.

Solution:

We have c = m^e mod N, and pk = (N,e),
By multiply c with 2^e and then mod N, we have:
(c * 2^e) mod N = (m^e * 2^e) mod N = (2m)^e mod N
And it will be decrypted to 2m.

20.

20. Assume that Alice's algorithm sends a NACK if the integer message she decrypts is larger than threshold $t = 2^\tau$. For example, think that Alice expects to encode the received message m as an *Int* type, in which case $\tau = 32$, or a *Long*, in which case $\tau = 64$, and throws an error if the received message does not fit in this range.

    Generalize the above attack to show that Mallory can find the exact value m encrypted in Bob's ciphertext c by sending $O(\tau)$ message $c'_1, c'_2, ...$ to Alice and watching the way Alice responds. Describe Mallory's algorithm, which takes as input c and $pk = (N, e)$ and in each $i = 1, 2, ..., c * \tau$ for some constant c, decides on c'_i to send to Alice, and observes whether Alice sends a NACK or not.

Solution:

To find out the m:

Since according to prof's commands, if we can have c′ =(c)^a·Epk(b) , then it holds that c′ =Epk((a∗m+b) modN). It means given ciphertext c=E(pk,m) there is an easy procedure, which Mallory can follow, which uses only the ciphertext c and public key pk which contains modulus N, s.t.\ for any integers a,b Mallory can compute encryption c′ =E(pk,m′) where m′ is related to m as m′ =a∗m+bmodN.

Since Mallory also know the tou at which the message will be dropped for the NACK, so he can try different a and b on the algorithm, to see the response of NACK. Through changing the different length of message m', find the integer threshold just above and below the tou, he can know the exact of m.

21. Exercise C-8.13.

C-8.13 Bob is stationed as a spy in Cyberia for a week and wants to prove that he is alive every day of this week and has not been captured. He has chosen a secret random number, x, which he memorized and told to no one. But he did tell his boss the value $y = H(H(H(H(H(H(x))))))$, where H is a one-way cryptographic hash function. Unfortunately, he knows that the Cyberian Intelligence Agency (CIA) was able to listen in on that message; hence, they also know the value of y. Explain how he can send a single message every day that proves he is still alive and has not been captured. Your solution should not allow anyone to replay any previous message from Bob as a (false) proof he is still alive.

Solution:

Bob should reveal the previous number in the hash chain that leads to y every day. The boss can use the hash function (yes, he has) to prove that the result is y. Since H is one-way, only Bob can effectively perform this inversion because he knows x, can not be reported by others.