December 19, 2023, Midterm 1+2, Computer & Network Security

SURNAME:________________ **NAME:** ________________ **MATRICOLA:**________________

Q1: Alice sends two 1-byte messages stream-encrypted with the same keystream (argh!). The two ciphertexts are:

c1 = 11000110

c2 = 11010111

You don't know of course the keystream, but you get to know from side information that m2 is the double of m1, and that m1 is between 1 and 127. Please determine the values m1 and m2.

(the solution is considered valid only if you discuss how you reach the final result – if you cannot find the final result at least explain how you would address the problem)

Solution: c1 XOR c2 = 0001.0001

→M1=0000.1111 (15), m2=0001.1110 (30)

Q2: respond to the following questions with **a single number or sentence if the question has no solution** (no need to provide derivation/computation steps, just optionally add a very short motivation if believed helpful)

An authentication system uses **one-time codes of 6 digits**. Example: 654321, 001123, 999176, etc. Approximatively, how many codes an attacker must retrieve in order to have a collision probability around 50%?	Approx 10^3 = 1000
If you wish to increase the robustness of a system, so that a brute force attack to the used key would increase of a factor of about **500 millions**, how many extra bits you should add to the key?	29 bit
What is the modular inverse **3^{-1} mod 19**	13, as 13x3=39 = 1 mod 19
What is the modular inverse **11^{-1} mod 132**	Not coprime, hence 11 not invertible mod 132
How many operations (squares or multiplications) are necessary for computing ***35^262 mod 863*** *(note, you do NOT need to compute the actual result!!)*	262 = 1.0000.0110 → 8 squares + 3 mul → 11 (10 also ok, depends on whether you count or do not count the first mul)

Q3: (TLS Beast Attack) – You have seen the following ciphertext obtained from a CBC-based block encryption scheme which uses block sizes of 8 bits (1^{st} block = IV):

(0110.1100) | 1110.0000 | 0001.1111 | **1010.0011** | 0000.0001

You can now predict that **ALL the next encryptions will use IV=0000.0000**. If you know from external sources of information that the third ciphertext block (the one underlined) contains a plaintext value with the format 00000XXX (in practice, a number between 0 and 7):

i) how many CPAs you need to perform in order to find such number, and

ii) which are your chosen plaintexts?

(in your example, assume each guess you make is successful, so as to avoid exploring all possible scenarios)

Guess 0000.0XXX → CPA = 0000.0000 XOR 0001.1111 XOR 0000.0XXX

(example, if guess = 0000.0011, CPA plaintext = 00011100)

Testing 1 bit at a time is NOT applicable for this exercise (the original BEAST attack combined the above with a chosen boundary attack), so the number of guesses remains exponential – for the exercise this is o(2^3), with worst case = 7

December 19, 2023, Midterm 1+2, Computer & Network Security

SURNAME:________________ **NAME:** ________________ **MATRICOLA:**________________

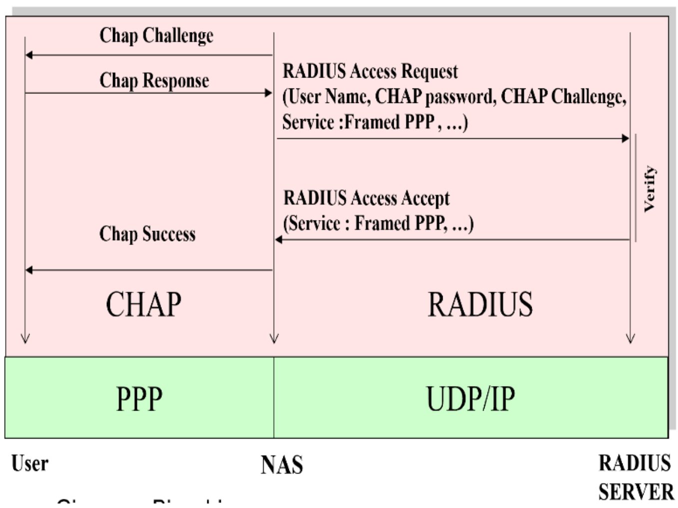
Q4: Network protocol related multiple answer questions *(comments can be added on the right if/when necessary)*

In TLS1.3 0-RTT, replay attacks can occur on	1. The first client message 2. The first server response 3. All handshake messages except the finished 4. No replay attack is possible
If we count as "1" the cost of a signature verification, and assume that a modular exponentiation also costs 1, when referring to the three DH variants (anonymous, fixed, ephemeral) we can approximately say that	1. All three DH variants cost the same 2. Fixed and Ephemeral cost 2, while anonymous cost 1 3. Anon costs 1, fixed costs 2, and ephemeral costs 3 4. Other (explain why): Question with different possible interpretations: if the cost was for the operations performed while "receiving" the DH coefficient (which I had implicitly in mind while preparing the exam), then the correct answer would be (3): Anon requires 1 EXP, Fixed requires 1EXP and 1 SIGGVERIFY, while Ephemeral requires 1EXP and 2 SIGNVERIFY. But if we also add the cost of SENDING our DH coefficient, then result would become ANON=2, FIXED=2, EPH=5
In cellular systems, the anonymity key protects	1. The user identifier 2. The sequence number 3. The AUTN 4. The random challenge
What is the best way to combine encryption (ENC) and integrity (MAC)?	1. MAC then ENC 2. ENC then MAC 3. ENC and MAC 4. All combinations have problem, must use AEAD
The possibility of using "dummy" packets in IPsec ESP can improve which security objective?	1. Message Confidentiality 2. Traffic Flow Confidentiality 3. Message Integrity 4. Traffic Flow Integrity (Session Integrity)

Q5: Please explain, eventually with a small example, why RSA key transport in TLS 1.2 **does NOT provide** forward secrecy

See related lecture (example on slides)

Q6: Please discuss which attack can be made against the way (traditional) RADIUS supports CHAP – see figure on the right



See related lecture

December 19, 2023, Midterm 1+2, Computer & Network Security

SURNAME:__________________ NAME: __________________ MATRICOLA:__________________

Q7: Consider an RSA system using modulus N =517 = 11x47.

1) For choosing the public key, would you have **some preference between e_1=3, e_2=5 or e_3=11**? If some values are preferable than others, please explain why.

5 cannot be used, as it is not coprime with PHI(n)=460,

2) from now on, also for simplicity of computation, let's use e=3, and compute the corresponding private key d.

d=3^{-1} mod 460 = 307

3) Compute **the signature of message M=7.** *to reduce computation efforts, here a few precomputed modular exponentiations, just in case you find them useful:*

7^x = {7, 49, 343, 333, 263, 290, 479, 251, 206} per x=1,2,3,4,5,6,7,8,9
7^x = {408, 507, 56, 100, 474, 34, 430, 177, 353} per x=10,20,30,40,50,60,70,80,90
7^x = {298, 397, 430, 441, 100} per x=100,200,300,400,500

Sign = 7^{307} mod 517 = 204

4) Assume that the above M=7 is the amount of euros you wish to pay for a given good, and that the transfer is valid only if properly signed. Is it **possible for an attacker who sees the above signed message but does NOT KNOW your private key to make you pay i) 343 euros instead of 7, ii) 507 euros instead of 7**?

- If yes, for either (i) and (ii), show the forged signature for the case 343;
- If yes only for (i) show the forged signature for the case 343 and discuss why this is NOT possible for 507;
- If it is not possible, specify why.

Yes for BOTH cases – the problem is that without hashing, the signature is malleable!

The case M2 = 343 is evident, as it is immediate to see that 343=7 x 7 x 7 = 7^3. Hence (all next operations mod n):

> SIGN(343) = 343^d = $(7^3)^d$ = (commutativity of exponentiations) $(7^d)^3$ =SIGN$(7)^3$
>
> And since in this special case 3 coincides with the pubkey e, not even need to do any computation, as $(7^d)^3$ = $(7^d)^e$ = 7 😊

Actually, a signature forging is also possible for <u>any value generated by 7^x mod n</u>!! Hence including 507, as 507 = 7^20 mod n (this was included in the above precomputations, so easy to see). To compute the signature you don't need d, but you just perform the following computation:

SIGN(507) = SIGN(7)^20 mod n = 309