Q1.

1600 bytes in filed

20 bytes in IP head

Offset = (1620 -20)/8 = 200

|  |  |  |  |
| --- | --- | --- | --- |
| Length | ID | Fragflag | offset |
| 1620 | x | 1 | o |
| 1620 | x | 1 | 200 |
| 760 | x | 0 | 400 |

Q2.

Source Ip address; Port number; checksum;

Q3.

We cannot derive the shortest path with Dijkstra algorithm, because it has negative weight, while using Dijkstra, we can go path the negative weight path for unlimited times and get very very very low number and it can keep going forever, therefore we can not find a shortest path through Dijkstra on negative weight graph,

Q4.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Step |  | D(x) | D(v) | D(u) | D(z) | D(y) | D(t) |
| 0 | w | 6,w | 4,w | 3,w | ∞ | ∞ | ∞ |
| 1 | wu |  |  |  | ∞ | ∞ | 5,u |
| 2 | wuv |  |  |  | ∞ | 12,v |  |
| 3 | wuvx |  |  |  | 14,x |  |  |
| 4 | wuvxy |  |  |  |  |  |  |
| 5 | wuvxyz |  |  |  |  |  |  |
| 6 | wuvxyzt |  |  |  |  |  |  |

Q5.

|  |  |  |
| --- | --- | --- |
| Dest | Cost | Next hop |
| B | 4 | D |
| C | 9 | D |
| D | 1 | D |
| E | 19 | D |

When B and D is broken, the system will get into the count to infinity problem:

D, does not update,

B, update based on C, distance 9 + 5 = 14

Then C update distance = 14 + 5 =21

Then B update distance from A to B = 20 stable after 3 circles 6 seconds,

Then C update distance 25 and stable after 4 circles 8 seconds,

After that E stables after 5 circles 10 seconds

Q6.

r = 3

G = 1101

R = 000

D = 111001

R’ = Reminder[D\*2^r/G] = 1 not equal to R(000)

So the result will be droped since the reminder is not 000.

Q7.

Number of collision = 3584 times

Q8.

From our slides we know that under the aloha mode, the max efficiency is 1/e = 0.37

And under the pure aloha mode the max efficiency is 1/(2e)= 0.18

And now we consider this question as two network, N\_1 works on aloha and N\_2 works on pure aloha that both have N node and only works for half of the time, and while N\_1 works, N\_2 idle, vice versa, so the efficiency for N\_1 is 0.5\* 0.37 = 0.18, while N\_2 is 0.09, and the whole system is 0.18 + 0.09 = 0.27

So the max efficiency is 27%.

Q9.

1. a, b, c, f, e, h, g
2. g, f, c, a
3. e, c, a
4. e, h, c, f, g
5. h, e, c

Q10.

The working principle of the RTS/CTS mechanism is that the sending station does not send the data immediately, but instead sends a request to send an RTS (Ready To Send) frame to apply for the medium When the receiving station receives the RTS signal, it immediately responds with a clearance to send CTS (Clear To send) frame after a short frame slot to inform the other party that it is ready to receive data. The two parties successfully exchange the RTS/CTS signal pair (that is, the handshake is completed) before the real data transmission begins, which ensures that when multiple invisible sending stations send signals to the same receiving station at the same time, only the receiving station can actually receive the response The station of the CTS frame can transmit, avoiding collisions. Even if there is a conflict, it is only when the RTS frame is sent. In this case, because the CTS message of the receiving station is not received, everyone will go back and use the competition mechanism to assign a random back-off timer value, and wait for the next time the medium is free.

Q11.

P,q = prime number

N = pq

φ(n)=(p-1)(q-1)  
m^φ(n) ≡ 1 mod n

let  
ed ≡ 1 mod φ(n)

cipher=m^e mod n  
plain = cipher ^ d mod n ≡ m^ed mod n = m

We know that the private key is (n, d) public key is (e, n)

And from the equations above, we know that even we have n,d published, adversary can not get the prime number pair(p,q) that we use and can not get e, and for the same reason, even if we have the e, n published, adversary also can not get p,q therefore could not calculate d. Threfore, the the public key and private key of RSA is interchangeable.

Q12.

1. Encrypted value: 0x11
2. Calculate with code:
3. def mod\_exp(x, y, N):  
    if N == 1:  
    return 0  
    result = 1  
    x = x % N  
    while y > 0:  
    if (y % 2 == 1):  
    result = (result \* x) % N  
    y >>= 1  
    x = (x \* x) % N  
    return result

Encrypted value: 0x52 for (91, 5)

Encrypted value: 0x52 for (91, 29)

Q13.

From φ(n)=(p-1)(q-1)  
m^φ(n) ≡ 1 mod n  
ed ≡ 1 mod φ(n)

1. (31,5), (31,11)

φ(n) = φ(31) = 31-1 =30

ed =55 not ≡ 1 mod 30

so no

1. (77,7), (77,43)

φ(n) = φ(77) = 60

ed = 7 \* 43 = 371 not ≡1 mod 60

so no

1. (55,7), (55,41)

φ(n) = φ(55) = 40

ed = 7\*41 = 287 not ≡1 mod 40

so no

Q15.

Bob decypt the (Ks) with his private key to get the Ks , and then decrypt the message with Ks, then generating the hash value of the message and verify the signature with the hash value and the public key of Alice.

Q 16

1. To prevent replay attack, making sure that the same key won’t be used to attack the system by collision attack.
2. To verify the prove the ownership of the public key of the entity. the certificate issuer is a certificate authority (CA), usually a company that charges customers to issue certificates for them. By contrast, in a web of trust scheme, individuals sign each other's keys directly, in a format that performs a similar function to a public key certificate.
3. Known plaintext attack: The attacker knows at least one sample of both the plaintext and the ciphertext. In most cases, this is recorded real communication. If the XOR cipher is used for example, this will reveal the key as plaintext xor ciphertext. Chosen plaintext attack: The attacker can specify his own plaintext and encrypt or sign it. He can carefully craft it to learn characteristics about the algorithm. For example he can provide an empty text, a text which consists of one "a", two "aa", ... For example: if the Vigenère cipher is used, it is very easy to extract the key length and recover the key by repeating one letter.