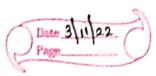
1) Diffie-Hellman key exchange is also known as emponential key enchange. It is a method of digital encryption that uses numbers raised to specific powers to produce decryption keys on the basis of components which aren't transmitted directly. This is used to exchange the secret key between the sender and the receiver. For example: - Credit card transaction email. 2) q = 17 => n Alice's Secret key = 4

Ra a = 5 Bob's secret key = 6 .. Public key = a secret key mod h For alice: For Bob: $5^4 \mod 17$ $Pk_B = 13$ $Pk_B = 2$ Sevet key = PK sevet ky mod n Promode Pra mode = 2 4 mod 17 = \$ \$ 13 mod 17 . The secret key they exchanged is 16



| 0 | Encryption & Decryption code for Vignère cipher |
|---|--|
| | Envergetion |
| | def enought_cipher Text (string, key): key = list (key) if len (string) == len (key): return (key) else: for i in range (len (string) - len (key)): key append (key [i % len (key)]) return (" ". join (key)) |
| | def encrypt - cipher Text (string, key): cipher - text = [] for is in range (lun(string)): x = (lord (string [i]) + ord (key [i])) / 26) + ord (4) cipher - text : append (chr (x)) return ("". join (cipher - text)) |
| | Deveyption: |
| | dej decrypt-original Tent (cipeur - tret, key): orig - tent = [] for i in range (len (cipher - tent)): x = ((ord (cipher - tent [i]) - ord(key[i]))/26)+ord('A') orig - tent . append (Chr (n)) return (" ". join (orig - tent)). |
| | |

Scanned with CamSca