Cryptography in CCNA

Definitions

Crypotography

Cryptography (or cryptology; from Greek kryptós, "hidden, secret"; and graphein, "writing", or -logia, "study") is the practice and study of techniques for secure communication in the presence of third parties.

Cryptography is heavily based on mathematical theory and computer science practice; cryptographic algorithms are designed around computational hardness assumptions, making such algorithms hard to break in practice by any adversary. It is theoretically possible to break such a system, but it is infeasible to do so by any known practical means.

Key in Cryptography

In cryptography, a key is a piece of information (a parameter) that determines the functional output of a cryptographic algorithm or cipher. Without a key, the algorithm would produce no useful result. In encryption, a key specifies the particular transformation of plaintext into ciphertext, or vice versa during decryption.

Key Exchange

Key exchange (also known as "key establishment") is any method in cryptography by which cryptographic keys are exchanged between users, allowing use of a cryptographic algorithm.

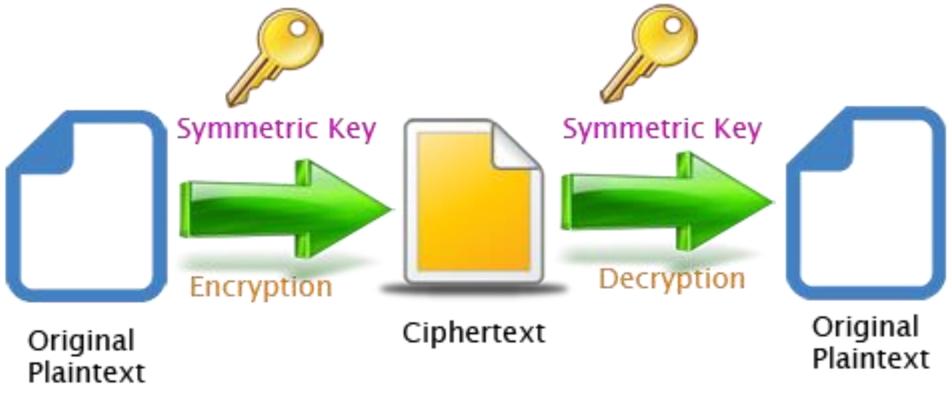
Symmetric-key cryptography

Symmetric-key cryptography refers to encryption methods in which both the sender and receiver share the same key (or, less commonly, in which their keys are different, but related in an easily computable way).

Symmetric key ciphers are implemented as either block ciphers or stream ciphers. A block cipher enciphers input in blocks of plaintext as opposed to individual characters, the input form used by a stream cipher.

The Data Encryption Standard (DES), the Advanced Encryption Standard (AES) and tripleDES are block cipher designs. RC4 is a widely used stream cipher

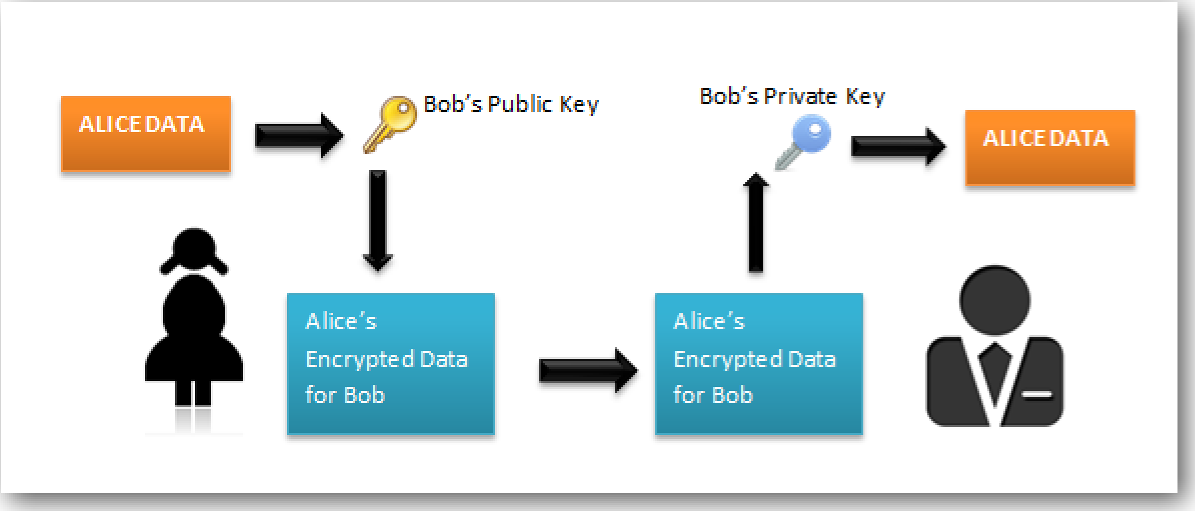
This type of encryption scheme is often called "shared secret" encryption, or "secret key" encryption.



Public-key(asymmetric-key) cryptography

In a groundbreaking 1976 paper, Whitfield Diffie and Martin Hellman proposed the notion of public-key (also, more generally, called asymmetric key) cryptography in which two different but mathematically related keys are used—a public key and a private key. A public key system is so constructed that calculation of one key (the 'private key') is computationally infeasible from the other (the 'public key'), even though they are necessarily related. Instead, both keys are generated secretly, as an interrelated pair.

In public-key cryptosystems, the public key may be freely distributed, while its paired private key must remain secret. In a public-key encryption system, the public key is used for encryption, while the private or secret key is used for decryption.

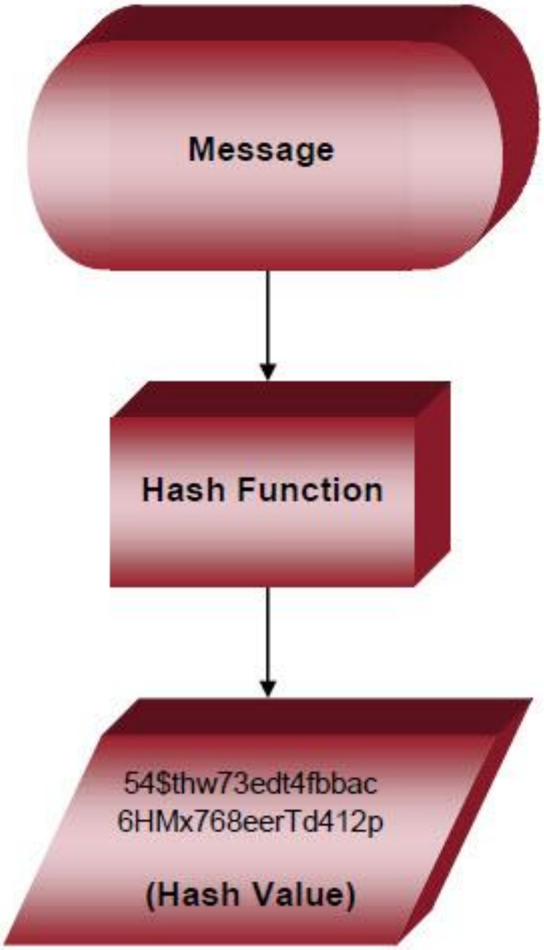

ALICE DATA
Bob's Public Key
Bob's Private Key
ALICE DATA
Alice's Encrypted Data for Bob
Alice's Encrypted Data for Bob

The Diffie–Hellman and RSA algorithms, in addition to being the first publicly known examples of high quality public-key algorithms, have been among the most widely used.

Cryptographic hash functions

Cryptographic hash functions take a message of any length as input, and output a short, fixed length hash which can be used in (for example) a digital signature. For good hash functions, an attacker cannot find two messages that produce the same hash. MD4, MD5, SHA-1 and SHA-2 are some examples of cryptographic hash functions.

Using the same hashing function and message should produce the same hash; modifying any portion of the data should produce an entirely different hash. A user should **not** be able to produce the original message from a given hash, but they **should** be able to tell if a given message produced a given hash.



Let us see where cyptography is used in ccna.

Enable secret command

Enable secrets are hashed using the MD5 algorithm.

To determine which scheme has been used to encrypt a specific password, check the digit preceding the encrypted string in the configuration file. If that digit is a 7, the password has been encrypted using the weak algorithm. If the digit is a 5, the password has been hashed using the stronger MD5 algorithm.

For example, in the configuration command:

```
enable secret 5 $1$iUjJ$cDZ03KKGh7mHfX2RSbDqP.
```

The enable secret has been hashed with MD5, whereas in the command:

```
username jdoe password 7
07362E590E1B1C041B1E124C0A2F2E206832752E1A01134D
```

The password has been encrypted using the weak reversible algorithm.

service password-encryption command

this global command encrypts all clear text passwords with weaker encryption. You will be able to see digit 7 before
the hashed password in running config.

SSH

SSH provides strong encryption, server authentication, and integrity protection. It may also provide compression.

We use the command crypto key generate rsa for configuring a router or switch for ssh.
RSA is asymmetric encryption.SSH 1.99 is not an actual version but a method to identify backward compatibility.

I used to wonder when I enable ssh v2 IOS used to reply ssh 1.99 is enabled.

SSH uses encryption in 3 or 4 areas

1. Data encryption for hiding data(symmetric encryption like DES, 3DES, AES)

2.Key exchange(asymmetric key exchange like DH, RSA)

3.Data intregrity(hashing like md5 or sha-1)

4. Authentication (optional; the encryption RSA)

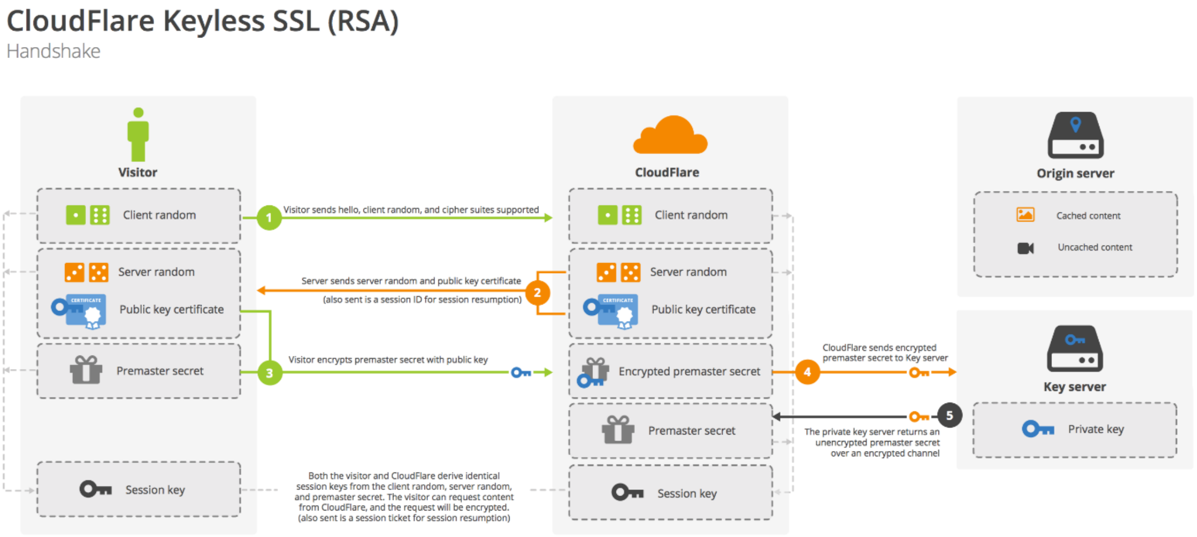
There are 4 alternative methods used in ssh for authentication and they are
1.password authentication
2.public-key based authentication(DSSor RSA)
3.keyboard interactive
4.GSSAPI authentication

How key exchange works in ssh?

Ssh server and client negotiates which symmetric algorithm to implement for their session. These algorithm requires a symmetric key that both server and client should have it. They should not communicate shared key(symmetric key) in plain text between them. For this purpose they use key exchange algorithm. SSH may use DH key exchange or RSA key exchange. RSA in low end computers and DH in high end computers because DH requires more CPU processing.

RSA Key Exchange:

Ssh server will create a pair of public key and private key. The private key will not be send to the client . the public key (P) will be send to the clinet. The client generates a string of random byte K, client upon receiving the public key it will encrypt the K with P(public key) producing H.then H is sent to the server.Server then decrypts the H with private key to obtain K. Both server and client do same math on K,P and other parameters to produce shared key that is going to be used for encrypting the data. One point to note that RSA private key is never received by client.


CloudFlare Keyless SSL (RSA)
Handshake
Visitor
Client random
Server random
Public key certificate
Premaster secret
Session key
CloudFlare
Client random
Server random
Public key certificate
Encrypted premaster secret
Premaster secret
Session key
Origin server
Cached content
Uncached content
Key server
Private key
1
Visitor sends hello, client random, and cipher suites supported
2
Server sends server random and public key certificate
(also sent is a session ID for session resumption)
3
Visitor encrypts premaster secret with public key
4
CloudFlare sends encrypted premaster secret to Key server
5
The private key server returns an unencrypted premaster secret over an encrypted channel
Both the visitor and CloudFlare derive identical session keys from the client random, server random, and premaster secret. The visitor can request content from CloudFlare, and the request will be encrypted. (also sent is a session ticket for session resumption)
CERTIFICATE

you may right click on the image and chose open in new tab to view the image in full screen.

What is server and client in ssh?

suppose you want to configure R2 from R1.you start a ssh session from R1 to R2. in this scenario, R2 is ssh server and R1 is ssh client. you are required to configure ssh in ssh server and not required to configure ssh in ssh client in cisco devices.(this paragraph is purely based on my understanding)

RSA algorithm can be used for authentication and key exchange in ssh. When we use the command crypto key generate rsa , which one they are referring to is not clear to me. My strong guess is that it mentions about RSA key exchange and not RSA public key authentication.

The main use of hash(md5 or sha-1) in SSH is for data integrity purposes and to verify the authenticity of communication. These are used to ensure that the received message text is intact and unmodified.

Little about hashing

Hashing is not used for encrypting the data for hiding the data in communication and then sent to client. if they do so client cannot decrypt the hashed value. Hashing is always one way. You can encrpyt with hashing but cannot decrypt. some people have the idea that hashing is not encryption but it is. Hashing is used for data integrity. In SSH, data is hashed and output of hash is then appended to data before sending to client/server. The Client/Server on receiving this packet does same hash algorithm on data and verifies that the hash output locally generated is same as hash output appended with data. if matching data is not modified and if not matching data is modified during communication.

CHAP

We have studied CHAP authentication in PPP. In chap, a one-way hash function (MD5) is implemented in its process.

The Challenge-Handshake Authentication Protocol (CHAP) is used to periodically verify the identity of the peer using a 3-way handshake. This is done upon initial link establishment, and MAY be repeated anytime after the link has been established.

1. After the Link Establishment phase is complete, the authenticator sends a "challenge" message to the peer.
2. The peer responds with a value calculated using a "one-way hash" function.
3. The authenticator checks the response against its own calculation of the expected hash value. If the values match, the authentication is acknowledged; otherwise the connection SHOULD be terminated.
4. At random intervals, the authenticator sends a new challenge to the peer, and repeats steps 1 to 3.

note: CHAP sends md5 hash value. The password is know to both peer before hand. Instead of sending password in clear text they sent Md5 hash value and then compared(locally genereted hash value of password and received hash value are compared).

VPN

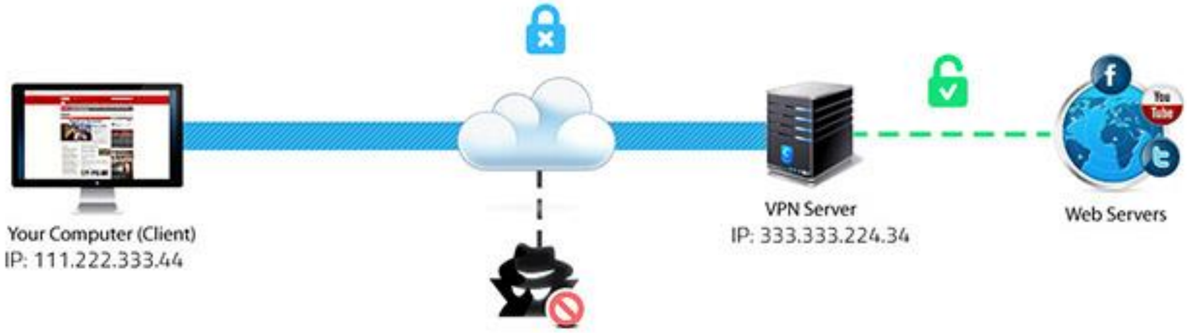
VPN is a tunnel through internet. It is used for wan purpose and also used for safe web browsing by hiding your ip address. VPN uses IPsec or SSL protocol which provides encryption.GRE tunnel is also a tunnel but it is not encrypted. To use GRE tunnel as VPN we have to use IPsec in gre tunnel which we will study in CCNP.

There are two types of VPN and they are

1.site to site VPN which are further classified as intranet VPN and extranet VPN.

2.Remote access VPN which are further classified as corporate VPN and personal VPN.

Personal VPNs provide safe web browsing, anonymous web browsing(hides Ip address), unblocks blocked websites.You can try hotspot shield, ultrasurf freeware personal vpn softwares for safe web browsing. It may encrypt only the traffic in web browsers(IE, google chrome). For encrypting torrent application, you may have to purchase shareware vpn softwares. These kind of software make a tunnel with our PC to their server and then browse internet and so it looks like we are browsing from their LAN. these softwares mostly helpful in middle east and china where many wesites are censored.



IPsec in VPN

Ipsec uses cryptography in 4 areas.

1. data encryption- hiding data(symmetric encryption like AES, 3-DES)

2. Authentication- verifying identities

3.hashing-protection aganist changes(md5, sha1)

4.key exchange(DH)

Ipsec after encrypting the data it encapsulates the packet and then sent through the tunnel. When we use vpn software for hiding ip address, this encapsulation step is the one that actually hides your ip address.(this sentence is purely my own observation)

In ssh, we saw client and server. In ipsec, we have peers(no client and server).

There are three alternative method for authentication in Ipsec and they are

1.kerberos version 5 authentication

2. preshared key authentication

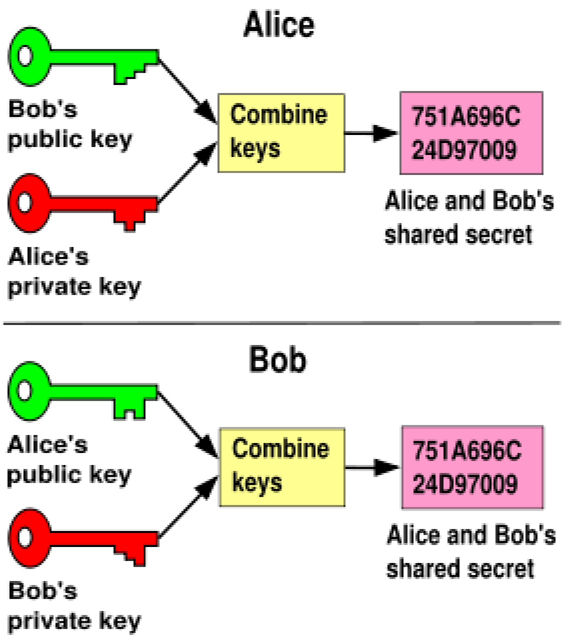
3. public key authentication

In preshared key authentication they use symmetric encryption and in public key authentication they use asymmetric encryption. Kerberos builds on symmetric encryption and optionally may use asymmetric encryption during certain phases of authentication.

In ipsec, key exchange is maintained by IKEv1 or IKEv2(within the framework of ISAKMP) which uses DH key exchange algorithm. DH is not used for data encryption.

How key exchange is done in Ipsec?

It is different from SSH(RSA key exchange). Both the peers create the DH key pair. Then both the peers transfer their public key. DH creates shared key by doing DH math with Private key locally created and public key of other peer. Both peer will create this shared key and it will be same. Suppose R1 and R2 are two peers. R1 creates R1-private and R1-public and R2 creates R2-private and R2-public. R1-public is send to R2 and R2-public is send to R1. DH in R1will do math with R1-private and R2-public to create an output K.same way, DH in R2 will do math with R2-private and R1-public to create an output K. This output K obtained by R1 and R2 will be same. This output K is the shared key(symmetric key).



Like ssh, Ipsec uses symmetric encryption for data confidentality and then exchange the key. Authentication is different in ssh and ipsec. Both ssh and ipsec uses hash for data integrity verification.

SNMPv3

SNMPv3 uses cryptography in 3 areas.
1.data integrity(md5 or sha1)
2.authentication(md5 or sha1)
3.privacy(DES,3DES,AES)
In SNMP, level name noAuthnoPriv is used to denote that there is no encryption for authentication and privacy(data hiding by encryption). if the level name is AuthnoPriv, only authentication is encrypted and privacy is not encrypted.if the level name is AuthPriv then both authentication and privacy is encrypted(this paragraph is based on my own observation)

GLBP

GLBP can also be configured for encrpted authentication(md5 or sha1).

Symmetric encryption is mostly used for data confidentiality and asymmetric encryption mostly used for authentication and key exchange. Asymmetric encryption is slower than symmetric is the reason it is not preferred for bulk data encryption. If we go for data confidentiality it is also considered wise to go for authentication and key exchange. What i learned so far is that, symmetric encryption for data encryption, asymmetric encryption for authentication and key exchange, hashing for data integrity and authentication.

I know key exchange in ssh or ipsec is not ccna syllabus, i discusses it here for more clarity. i didnt discuss anything about authentication, it is a vast topic.whenever i wrote my own sentence which is not verified with standard document i have quoted it by saying it is my own observation.ssh and ipsec is also very vast topic to study.i worked so hard to make this document.

I really appreciate Whitfield Diffie and Martin Hellman who came up with asymmetric cryptography for the first time in 1976. modern encryption is simply a complex mathematic algorithm which require a computer to compute it. Cryptography is an interesting topic but I am not that genious to pursue cryptography as my career.