PGP ALGORITHMS

By : Jaishree M.S, Sneha S

INTRODUCTION

PGP or Pretty Good Privacy was a popular program used to encrypt and decrypt email over the internet, as well as authenticate messages with digital signatures and encrypted stored files.. It uses three different PGP algorithms:

1. PGP Public Key Algorithm
2. PGP Hash Algorithm
3. PGP Encryption Algorithm

1. PGP PUBLIC KEY ALGORITHM

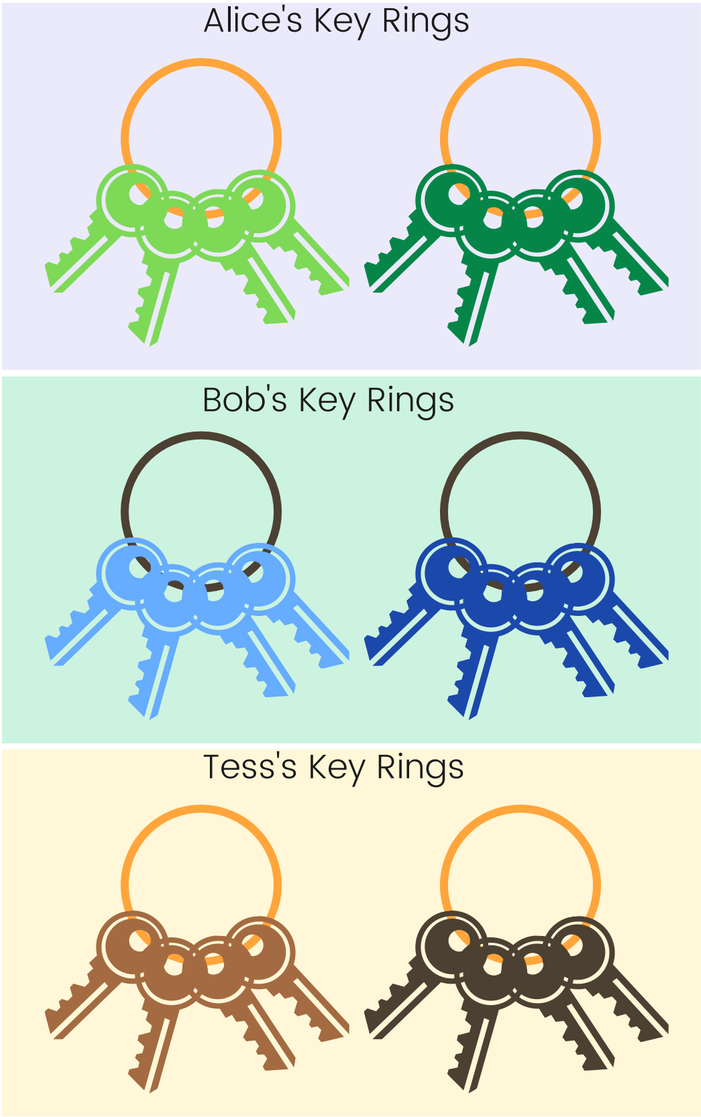
- A PGP key is a public encryption key.
- It used to sign and encrypt e-mails and file's.
- When you create a PGP key , public key and private is generated.
- The public key algorithms that we use are,
- 1. RSA 2. DSS 3. Diffie- hellman key .
- PGP often uses RSA to encrypt it's public key.

2. PGP HASH ALGORITHM

- It creates an mathematical summary know as hash to send digital signatures.
- This hash code is encrypted with sender's private key and decrypted with recievers public key.
- Some commonly used hash algorithms are,
- MD5 , SHA1, LANMAN

3. PGP ENCRYTION ALGORITHM

- It is an data encryption that gives privacy and authentication
- Often used to encrypt and decrypt text, e-mails and file's to increase the security of e-mails.
- It mostly uses RSA algorithm. It is the first public key which encrypts a key using IDEA.



KEY RINGS

Each User needs to have 2 sets of Rings:

- Ring of Private/Public keys
- Ring of Public keys

Alice, for example, has several pairs of private/ public keys belonging to her and public keys belonging to other people. Everyone can have more than one public key. Two cases may arise.

First Case

Alice needs to send a message to one of the people in the community.

- She uses her private key to sign the digest.
- She uses the receiver's public key to encrypt a newly created session key.
- She encrypts the message and signs the digest with the session key created.

Second Case

Alice receives a message from one of the persons in the community.

- She uses her private key to decrypt the session key.
- She uses the session key to decrypt the message and digest.
- She uses her public key to verify the digest.

Thank you!