GnuPG

2023-12-01

Key IDs

In the context of GnuPG, a Key ID is a short identifier that is used to refer to a specific public key in the keyring. The Key ID is derived from the full public key by taking the last eight characters of the fingerprint, which is a hash value that uniquely identifies the key.

For example, if the fingerprint of a public key is `D8195AD695B68F92C0103AEDF3F98125261A1979`, then the corresponding Key ID would be `261A1979`.

The Key ID is useful for verifying the authenticity of a public key, especially when communicating with others securely. It can also be used to search for keys in public key servers, and to indicate which keys are trusted or revoked in a keyring.

Key Management

Generate a new key pair

```
gpg --full-gen-key
```

Follow the prompts to choose key type, size, expiration, and user ID, and to set a passphrase.

List all keys

```
gpg --list-keys --keyid-format short
```

Example output:

```
pub   rsa4096/D23A8374 2023-04-28 [SC]
      D0D1F8386B66D643ADB94A22D63DDAAAD23A8374
uid         [ultimate] Kyle Kestell <kyle@kestell.org>
sub   rsa4096/74621B36 2023-04-28 [E]
```

Explanation of the output:

`pub` Indicates that this is a public key.
`rsa4096` Indicates that the key uses RSA encryption with a key length of 4096 bits.
`D23A8374` This is the short key ID, which is the last 8 characters of the full key ID.
`2023-04-28` This is the date when the key was created.
`[SC]` Indicates that the key is capable of both signing and encrypting data.
`D0D1F838...` This is the full key ID, which is a 40-character string that uniquely identifies the key.
`uid` Indicates that this is a user ID associated with the key.
`[ultimate]` Indicates that this is the ultimate trust level of the key.
`Kyle Kestell <kyle@kestell.org>` This is the name and email address associated with the key.

`sub` Indicates that this is a subkey, which is a secondary key associated with the primary key.
`rsa4096` Indicates that the subkey uses RSA encryption with a key length of 4096 bits.
`74621B36` This is the short key ID of the subkey.
`2023-04-28` This is the date when the subkey was created.
`[E]` Indicates that the subkey is capable of encrypting data.

List private keys

```
gpg --list-secret-keys
```

Export your public key

```
gpg --output pubkey.gpg --armor --export 'Key ID'
```

Export your private key

```
gpg --output privkey.gpg --armor --export-secret-keys 'Key ID'
```

Replace ‘Key ID’ with the name you used to create the keys.

Import a public or private key

```
gpg --import key.gpg
```

Replace `key.gpg` with the filename of the key you want to import.

Delete a public key

```
gpg --delete-key 'Key ID'
```

Delete a private key

```
gpg --delete-secret-key 'Key ID'
```

Replace ‘Key ID’ with the name of the key owner.

Signing Files

Sign a file

```
gpg --sign file.txt
```

This will create a signed file `file.txt.gpg`. You’ll need to enter your passphrase to sign the file.

Verify a signed file

```
gpg --verify file.txt.gpg
```

This will verify the signature on the file.

Encrypting Files

Encrypt a file

```
gpg --encrypt --recipient 'Key ID' file.txt
```

This will encrypt `file.txt` for the recipient (yourself, in this case), creating `file.txt.gpg`.

Decrypt a file

```
gpg --output file.txt --decrypt file.txt.gpg
```

This will decrypt `file.txt.gpg` into `file.txt`. You'll need to enter your passphrase to decrypt the file.

Sign and Encrypt Files

Sign and encrypt a file

```
gpg --sign --encrypt --recipient 'Key ID' file.txt
```

This will sign and then encrypt `file.txt` for the recipient (yourself, in this case), creating `file.txt.gpg`.

Decrypt and verify a signed and encrypted file

```
gpg --decrypt file.txt.gpg
```

This will decrypt `file.txt.gpg` into `file.txt` and verify the signature. You'll need to enter your passphrase to decrypt the file.

Trusting Keys

Trust a key

```
gpg --edit-key 'Key ID'
```

This command opens the GnuPG command prompt. You can then use the `trust` command to trust the key.

Send a key to a keyserver

```
gpg --send-keys --keyserver keyserver.ubuntu.com 'Your KeyID'
```

This command uploads your public key to a keyserver, making it available to others.

Receive a key from a keyserver

```
gpg --recv-keys --keyserver keyserver.ubuntu.com 'Their KeyID'
```

This command downloads someone else's public key from a keyserver.

Search for a key on a keyserver

```
gpg --keyserver keyserver.ubuntu.com --search-keys 'Key ID'
```

Refresh your keyring

```
gpg --refresh-keys
```

This command updates your keyring with the latest keys and signatures from the keyserver.