NAME

sshpk−conv − convert between key formats

SYNOPSYS

sshpk−conv −t FORMAT [FILENAME] [OPTIONS...]

sshpk−conv −i [FILENAME] [OPTIONS...]

DESCRIPTION

Reads in a public or private key and converts it between different formats, particularly formats used in the SSH protocol and the well−known PEM PKCS#1/7 formats.

In the second form, with the **−i** option given, identifies a key and prints to stderr information about its nature, size and fingerprint.

EXAMPLES

Assume the following SSH−format public key in **id_ecdsa.pub**:

```
ecdsa−sha2−nistp256 AAAAE2VjZHNhLXNoYTI...9M/4c4= user@host
```

Identify it with **−i**:

```
$ sshpk−conv −i id_ecdsa.pub
id_ecdsa: a 256 bit ECDSA public key
ECDSA curve: nistp256
Comment: user@host
Fingerprint:
  SHA256:vCNX7eUkdvqqW0m4PoxQAZRv+CM4P4fS8+CbliAvS4k
  81:ad:d5:57:e5:6f:7d:a2:93:79:56:af:d7:c0:38:51
```

Convert it to **pkcs8** format, for use with e.g. OpenSSL:

```
$ sshpk−conv −t pkcs8 id_ecdsa
−−−−−BEGIN PUBLIC KEY−−−−−
MFkwEwYHKoZIzj0CAQYIKoZIzj0DAQcDQgAEAsA4R6N6AS3gzaPBeLjG2ObSgUsR
zOt+kWJoijLnw3ZMYUKmAx+lD0I5XUxdrPcs1vH5f3cn9TvRvO9L0z/hzg==
−−−−−END PUBLIC KEY−−−−−
```

Retrieve the public half of a private key:

```
$ openssl genrsa 2048 | sshpk−conv −t ssh −c foo@bar
ssh−rsa AAAAB3NzaC1yc2EAAA...koK7 foo@bar
```

Convert a private key to PKCS#1 (OpenSSL) format from a new−style OpenSSH key format (the **ssh−keygen −o** format):

```
$ ssh−keygen −o −f foobar
 ...
$ sshpk−conv −p −t pkcs1 foobar
−−−−−BEGIN RSA PRIVATE KEY−−−−−
MIIDpAIBAAKCAQEA6T/GYJndb1TRH3+NL....
−−−−−END RSA PRIVATE KEY−−−−−
```

OPTIONS

−i, −−identify
Instead of converting the key, output identifying information about it to stderr, including its type, size and fingerprints.

−p, −−private
Treat the key as a private key instead of a public key (the default). If you supply **sshpk−conv** with a private key and do not give this option, it will extract only the public half of the key from it and work with that.

−f PATH, −−file=PATH
Input file to take the key from instead of stdin. If a filename is supplied

as a positional argument, it is equivalent to using this option.

−o PATH, −−out=PATH
Output file name to use instead of stdout.

−T FORMAT, −−informat=FORMAT

−t FORMAT, −−outformat=FORMAT
Selects the input and output formats to be used (see FORMATS, below).

−c TEXT, −−comment=TEXT
Sets the key comment for the output file, if supported.

FORMATS

Currently supported formats:

pem, pkcs1
The standard PEM format used by older OpenSSH and most TLS libraries such as OpenSSL. The classic **id_rsa** file is usually in this format. It is an ASN.1 encoded structure, base64−encoded and placed between PEM headers.

ssh
The SSH public key text format (the format of an **id_rsa.pub** file). A single line, containing 3 space separated parts: the key type, key body and optional key comment.

pkcs8
A newer PEM format, usually used only for public keys by TLS libraries such as OpenSSL. The ASN.1 structure is more generic than that of **pkcs1** .

openssh
The new **ssh−keygen −o** format from OpenSSH. This can be mistaken for a PEM encoding but is actually an OpenSSH internal format.

rfc4253
The internal binary format of keys when sent over the wire in the SSH protocol. This is also the format that the **ssh−agent** uses in its protocol.

SEE ALSO

ssh−keygen(1), openssl(1)

BUGS

Encrypted (password−protected) keys are not supported.

Report bugs at Github *https://github.com/arekinath/node−sshpk/issues*