

**NAME****moduli** - Diffie-Hellman moduli**DESCRIPTION**

The `/etc/ssh/moduli` file contains prime numbers and generators for use by `sshd(8)` in the Diffie-Hellman Group Exchange key exchange method.

New moduli may be generated with `ssh-keygen(1)` using a two-step process. An initial *candidate generation* pass, using **ssh-keygen -G**, calculates numbers that are likely to be useful. A second *primality testing* pass, using **ssh-keygen -T**, provides a high degree of assurance that the numbers are prime and are safe for use in Diffie-Hellman operations by `sshd(8)`. This **moduli** format is used as the output from each pass.

The file consists of newline-separated records, one per modulus, containing seven space-separated fields. These fields are as follows:

timestamp	The time that the modulus was last processed as YYYYMMDDHHMMSS.
type	Decimal number specifying the internal structure of the prime modulus. Supported types are:
0	Unknown, not tested.
2	"Safe" prime; $(p-1)/2$ is also prime.
4	Sophie Germain; $2p+1$ is also prime.

Moduli candidates initially produced by `ssh-keygen(1)` are Sophie Germain primes (type 4). Further primality testing with `ssh-keygen(1)` produces safe prime moduli (type 2) that are ready for use in `sshd(8)`. Other types are not used by OpenSSH.

tests	Decimal number indicating the type of primality tests that the number has been subjected to represented as a bitmask of the following values:
0x00	Not tested.
0x01	Composite number - not prime.
0x02	Sieve of Eratosthenes.
0x04	Probabilistic Miller-Rabin primality tests.

The `ssh-keygen(1)` moduli candidate generation uses the Sieve of Eratosthenes (flag 0x02). Subsequent `ssh-keygen(1)` primality tests are Miller-Rabin tests (flag 0x04).

trials	Decimal number indicating the number of primality trials that have been performed on the modulus.
size	Decimal number indicating the size of the prime in bits.
generator	The recommended generator for use with this modulus (hexadecimal).
modulus	The modulus itself in hexadecimal.

When performing Diffie-Hellman Group Exchange, `sshd(8)` first estimates the size of the modulus required to produce enough Diffie-Hellman output to sufficiently key the selected symmetric cipher. `sshd(8)` then randomly selects a modulus from `/etc/ssh/moduli` that best meets the size requirement.

**SEE ALSO**

`ssh-keygen(1)`, `sshd(8)`

*Diffie-Hellman Group Exchange for the Secure Shell (SSH) Transport Layer Protocol*, RFC 4419, 2006.