

























SSH public key algorithms and their support by SSH clients, @openssh.com abbreviated to @, sourced from official vendor documentation (<https://www.openssh.com/specs.html>) and the client's configuration. Support for an algorithm is denoted by a filled circle (●), while an outlined circle (○) indicates the lack thereof. Partial support (◐) indicates that support may be enabled explicitly at build time but is disabled by default. Data points in light gray indicate the need for experimental verification due to the lack of vendor documentation and configurable public key algorithms.

Name	Version	DSA								RSA								ECDSA								EdDSA						
		ssh-dss [4]	ssh-dss-cert-v01@ [5]	x509v3-ssh-dss [3]	spki-sign-dss [4]	pgp-sign-dss [4]	ssh-rsa [4]	rsa-sha2-256 [1]	rsa-sha2-512 [1]	ssh-rsa-cert-v01@ [5]	rsa-sha2-256-cert-v01@ [5]	rsa-sha2-512-cert-v01@ [5]	x509v3-ssh-rsa [3]	x509v3-rsa2048-sha256 [3]	spki-sign-rsa [4]	pgp-sign-rsa [4]	ecdsa-sha2-nistp256 [7]	ecdsa-sha2-nistp384 [7]	ecdsa-sha2-nistp521 [7]	ecdsa-sha2-1.3.132.0.10 [7]	ecdsa-sha2-nistp256-cert-v01@ [5]	ecdsa-sha2-nistp384-cert-v01@ [5]	ecdsa-sha2-nistp521-cert-v01@ [5]	sk-ecdsa-sha2-nistp256@ [6]	sk-ecdsa-sha2-nistp256-cert-v01@ [6]	x509v3-ecdsa-sha2-* [3]	ssh-ed25519 [2]	ssh-ed25519-cert-v01@ [5]	sk-ssh-ed25519@ [6]	sk-ssh-ed25519-cert-v01@ [6]	ssh-ed448 [2]	
AbsoluteTelnet	12.16	●	○	○	○	○	●	●	●	○	○	○	○	○	○	○	●	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○
AsyncSSH	2.18.0	●	●	○	○	○	●	●	●	○	○	○	○	○	○	○	●	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○
Bitvise	9.42	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Cyberduck	9.0.1	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Dropbear	2024.86	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Erlang/OTP SSH	5.2.1	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
FileZilla	3.67.0	◐	○	○	○	○	◐	◐	○	○	○	○	○	○	○	○	◐	◐	◐	○	○	○	○	○	○	○	○	○	○	○	○	○
Golang x/crypto/ssh	0.29.0	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
libssh	0.11.1	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
OpenSSH Portable	9.9p1	◐	◐	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Paramiko	3.5.0	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
PKIX-SSH	15.3	◐	◐	◐	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
PuTTY	0.81	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
SecureCRT	9.5.2	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Tectia SSH	6.6.3.490	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Tera Term	5.2	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Termius	9.8.5	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Win32 OpenSSH	9.5.0.0	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
WinSCP	6.3.4	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
XShell 7	Build 0170	◐	○	○	○	○	◐	◐	◐	◐	◐	◐	○	○	○	○	◐	◐	◐	○	◐	◐	◐	○	○	○	○	◐	○	○	○	○

Actively maintained third-party graphical user interfaces for the Git version control system on Windows and Linux operating systems. For each client, the bundled Git and SSH clients are listed, if any. If Git is not bundled, the clients require a manual installation of Git with either OpenSSH, PuTTY, or TortoiseGitPlink as its SSH client. Git can be configured to use a custom SSH client as part of the configuration or via the `GIT_SSH_COMMAND` environment variable.

Name	Version	OS	Bundled Git Client	Bundled SSH Client	Crypto Library
Fork	2.3.1		Git 2.45.2.windows.1	OpenSSH 9.7p1	OpenSSL 3.2.1
git-cola	4.10.1	 	not included	not included	not included
GitForce	1.0.61	 	not included	not included	not included
GitHub Desktop	3.4.12		Git 2.45.1.windows.1	OpenSSH 9.7p1	OpenSSL 3.2.1
GitKraken	10.6.0	 	Git 2.47.0.windows.2	OpenSSH 9.9p1	OpenSSL 3.2.3
			Git 2.47.0	not included	not included
GitLens	16.1.0	 	not included	not included	not included
MeGit	0.8.0	 	JGit 7.1.0	Apache SSHD 2.14.0	BC Java 1.79.0
SmartGit	24.1.1	 	Git 2.46.0.windows.1	OpenSSH 9.8p1	OpenSSL 3.2.2
			not included	not included	not included
Sourcetree	3.4.21		Git 2.46.0.windows.1	PuTTY 0.81	PuTTY 0.81
Sublime Merge	Build 2102	 	Git 2.39.1.windows.1	OpenSSH 9.1p1	OpenSSL 1.1.1s
			not included	not included	not included
TortoiseGit	2.17.0.2		not included	TortoiseGitPlink 0.81	PuTTY 0.81
Tower	8.3.499		Git 2.47.1.windows.1	OpenSSH 9.9p1	OpenSSL 3.2.3
ungit	1.5.28	 	not included	not included	not included

## References

- [1] Denis Bider. 2018. Use of RSA Keys with SHA-256 and SHA-512 in the Secure Shell (SSH) Protocol. RFC 8332. doi:10.17487/RFC8332
- [2] Ben Harris and Loganaden Velvindron. 2020. Ed25519 and Ed448 Public Key Algorithms for the Secure Shell (SSH) Protocol. RFC 8709. doi:10.17487/RFC8709
- [3] Kevin Igoe and Douglas Stebila. 2011. X.509v3 Certificates for Secure Shell Authentication. RFC 6187. doi:10.17487/RFC6187
- [4] Chris M. Lonvick and Tatu Ylonen. 2006. The Secure Shell (SSH) Transport Layer Protocol. RFC 4253. doi:10.17487/RFC4253
- [5] Damien Miller. 2024. This document describes a simple public-key certificate authentication system for use by SSH. <https://cvsweb.openbsd.org/cgi-bin/cvsweb/src/usr.bin/ssh/PROTOCOL.certkeys?rev=1.20> Accessed: 2025-04-14.
- [6] Damien Miller and Markus Friedl. 2020. This document describes OpenSSH's support for U2F/FIDO security keys. <https://cvsweb.openbsd.org/cgi-bin/cvsweb/src/usr.bin/ssh/PROTOCOL.u2f?rev=1.26> Accessed: 2025-04-14.
- [7] Douglas Stebila and Jonathan Green. 2009. Elliptic Curve Algorithm Integration in the Secure Shell Transport Layer. RFC 5656. doi:10.17487/RFC5656