SSH key types and formats

Please note we have update our SSH client starting with Unimus 2.2.3. The table below applies to version 2.2.3 or newer.

Unimus currently only supports SSH keys without passwords, so the key being imported into Unimus must **not** be password-protected.
The SSH keys imported into Unimus are stored encrypted in the Unimus DB (using the encryption key configured during the Deployment Wizard).

Supported key types:

```
ssh-ed25519, ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521,
rsa-sha2-512, rsa-sha2-256, ssh-rsa, ssh-dss, ssh-dsa
```

SSH private keys in the following formats are currently supported:

	rsa	dsa	ecdsa	ed25519
PEM	yes	yes	yes	yes
PKCS8	yes	yes	yes	yes
OpenSSH (RFC4716)	yes	yes	yes	yes

Different versions of **ssh-keygen** (OpenSSH) use different default key types and formats across the different OpenSSH versions.

Key generation and conversion / transformation

If you have an existing key in OpenSSH format, you can use **ssh-keygen** to transform key formats. This command will transform an OpenSSH private key into a PEM encoded key:

```
ssh-keygen -e -f /path/to/openssh.key -m PEM > /path/to/new_pem.key

-e read OpenSSH formatted key
-f read from file
-m export format
```

Alternatively, you can generate a new key pair already formatted in a supported format:

```
ssh-keygen -t ecdsa -m PEM -f /path/to/key

-t use ecdsa type
-m use PEM format
-f output to file
```