



Web



Digital signatures



WEB &gt; GUIDES &gt; PSPDFKIT SERVER &gt; DIGITAL SIGNATURES

# Explore digital signatures and cryptographic formats



PSPDFKit Server has been deprecated and replaced by [Document Engine](#). To migrate to Document Engine and unlock advanced document processing capabilities, refer to our [migration guide](#). Learn more about these enhancements on our [blog](#).

Nutrient supports the following signing methods:

- ✦ **PAdES** (PDF Advanced Electronic Signatures, ETSI.CAdES.detached)
- ✦ **CMS** (adbe.pkcs7.detached)

Supported signing algorithms are:

- ✦ **RSA**
- ✦ **ECDSA**

Supported hashing algorithms include:

- ✦ **MD4**
- ✦ **MD5**
- ✦ **SHA-2** (SHA-224, SHA-256, SHA-384, SHA-512, SHA-512-256)

A hashing algorithm is an algorithm that converts a large amount of data (your PDF document) into a fixed-length string. If someone changes one bit of data in your PDF document, the hashing algorithm



ASK AI

produces a different string, so if you store that hash inside the PDF document, you could potentially know when it has been changed by a third party. However, as easy as it sounds, this approach is insecure. If someone knows the hashing algorithm that was used to sign the PDF, they could replace the hash inside the PDF so that it validates as authentic. To avoid this, Nutrient works with encryption algorithms.

More specifically, Nutrient supports `RSA`, one of the most popular public key encryption algorithms. `ECDSA`, an algorithm based on elliptic curve cryptography, is also supported. Cryptographic systems based on `ECDSA` are becoming the de facto standard for messaging and systems security.

## Supported certificate encodings

Certificate stores can be encoded in either of these two formats:

- ⌘ DER- or PEM-encoded PKCS#7
- ⌘ DER- or PEM-encoded X.509

If you're unsure about the format of your certificate, try to load it with the format specified ([learn more](#)):

```
1 openssl x509 -inform pem -in cerfile.cer -noout -text
2 openssl x509 -inform der -in cerfile.cer -noout -text
```



Learn how to [generate an RSA certificate](#) for testing purposes.

Was this helpful?

✓ YES

✗ NO

Questions? [Contact us](#)

