CRYPTOMATHIC

How to Generate a Key Pair for Key Export

Supported public key formats

Two public key formats are supported:

- PEM public key (header '-----BEGIN PUBLIC KEY-----')
- PEM x509 certificate (header '-----BEGIN CERTIFICATE-----')

The exported key will be wrapped (i.e., encrypted) with a public RSA key using PKCS #1 RSA encryption with OAEP encoding (SHA-256 and MGF1). Any tool supporting this scheme can be used to export and unwrap the exported payload when provided with the corresponding RSA private key.

In the following, we will use openssl:

openssl genrsa -out rsakeypair.pem 2048

This will generate a **2048-bit RSA** keypair and store it in the file **rsakeypair.pem**. Now extract the public key of the keypair:

openssl rsa -in rsakeypair.pem -pubout -outform PEM -out public-key.pem

This will extract the public key and store it in the file **public-key.pem**. The content of public-key.pem can be pasted directly into the public key input field of the BYOK web service page for key export preparation.

The BYOK web service page also supports a public key as a PEM-formatted x509 certificate. To issue a self-signed certificate for the public key do the following:

openssl req -new -x509 -key rsakeypair.pem -out self-signed-x509.pem

This will issue a self-signed **x509** certificate and store it in the file **self-signed-x509.pem**. The content can then be pasted directly into the public key input field of the BYOK web service page.

Export

The user preparing the export and the user performing the actual export must not be the same.

The exported payload is the raw binary key encryption. To unwrap the payload do the following:

openssl pkeyutl -decrypt -in <exported payload in binary> -inkey rsakeypair.pem -out clear-key -pkeyopt rsa_padding_mode:oaep -pkeyopt rsa_oaep_md:sha256

This will store the unwrapped key in the file **clear-key**.