What is RSA-KEM?

RSA-KEM, it works as following

1. First, encrypt the object to be encrypted with the block cipher (AES-256-CBC).
2. Next, the encryption key used in the block cipher is encrypted with RSA-KEM.

Since public key cipher and block cipher are used together, it is sometimes called hybrid cipher.

Also, for encrypting block encryption, always use the newly generated encryption key.

Therefore, it is sometimes called one-time pad encryption.

When using RSA-KEM, there is no limit on the length of encryption target.

KEM is the Key Encapsulation Mechanism.

The RSA-KEM is a Key Encapsulation Mechanism using RSA public key cipher.

Citing RFC 5990, RSA-KEM looks like this:

RSA-KEM encryption

1. Z = RandomInteger (0, n-1) …Generate a random integer z between 0 and n-1.
2. C = Z^e mod n … Encrypt the integer z with the recipient's RSA public key:
3. KEK = KDF (Z, kekLen) … Derive a key-encrypting key KEK from the integer z.
4. WK = Wrap (KEK, K) … Wrap the keying data using KEK to obtain wrapped keying data WK.
5. EK = C || WK … Output c and WK as the encrypted keying data.

RSA-KEM decryption

1. C || WK = EK … Separate the encrypted keying data EK into a cipher text C of length nLen bytes and wrapped keying data WK:
2. Z = C^d mod n … Decrypt the integer c using the recipient's private key (n,d) to recover an integer z (see note):
3. KEK = KDF (Z, kekLen) … Derive a key-encrypting key KEK of length kekLen bytes from the byte string Z using the underlying key derivation function
4. K = Unwrap (KEK, WK) … Unwrap the wrapped keying data WK with the key-encrypting key KEK using the underlying key-wrapping scheme to recover the keying data K:

RSA-KEM does not directly apply RSA public key cipher to encryption targets.

The target for encryption of RSA public key cipher is a newly generated random number.

> The RSA public key encryption is applied to the newly generated random number that is the seed of the key derivation function.

This is RSA-KEM.