**This is a typical PEM-encoded X.509 certificate. Let's break down the components:**

1. **Certificate Header:**

```
-----BEGIN CERTIFICATE-----
```

This marks the beginning of the certificate**.**

2. **Certificate Body (Base64-encoded data):**

```
MIIIHjCCBgagAwIBAgIDIvcqMA0GCSqGSIb3DQEBCwUAMI
 ...
L4q069Rdrh0ZnuTbvk34gDPt
```

This section contains the actual certificate data in Base64 encoding.

3. **Certificate Footer:**

```
-----END CERTIFICATE-----
```

This marks the end of the certificate.

4. **Certificate Data:**

    - The certificate data is a binary structure encoded in Base64.
    - It includes information such as the version, serial number, signature algorithm, issuer, validity period, subject, public key details, extensions, and the digital signature.

**Extract Signature Information and analysis :**

- **Use the pkcs7 command to extract information from the P7M file. The pkcs7 command can be used to view and verify PKCS#7 signatures.**

```
openssl pkcs7 -inform DER -print_certs -text -in ch.txt.p7m
```

**Certificate: Data: Version: 3 (0x2) Serial Number: 2291498 (0x22f72a) Signature Algorithm: sha256WithRSAEncryption Issuer: C=IT, O=Telecom**

**Italia Trust Technologies S.r.l., OU=Qualified Trust Service Provider, CN=TI Trust Technologies QTSP CA 1/organizationIdentifier=VATIT-04599340967 Validity Not Before: Apr 3 20:38:58 2023 GMT Not After : Apr 3 20:38:58 2026 GMT Subject: C=IT, CN=FABRIZIO D'AMORE, SN=D'AMORE, GN=FABRIZIO/serialNumber=TINIT-DMRFRZ60P04H501I/dnQualifier=TITT030423203857337, O=Università degli Studi di Roma "La Sapienza"/organizationIdentifier=VATIT-80209930587**

1. **Version: 3 (0x2):**
   - Indicates the version of the X.509 certificate. In this case, it's version 3.
2. **Serial Number: 2291498 (0x22f72a):**
   - A unique identifier assigned by the certificate issuer to distinguish this certificate from others.
3. **Signature Algorithm: sha256WithRSAEncryption:**
   - Describes the algorithm used to sign the certificate. In this case, it's SHA-256 with RSA encryption.
4. **Issuer:**
   - Identifies the entity (Telecom Italia Trust Technologies S.r.l.) that issued the certificate.
5. **Validity:**
   - Specifies the time period during which the certificate is considered valid.
     - Not Before: Apr 3 20:38:58 2023 GMT
     - Not After: Apr 3 20:38:58 2026 GMT

**Subject:**

1. **C=IT:**

   - Country code. In this case, "IT" represents Italy.

2. **CN=FABRIZIO D'AMORE:**
   - Common Name. It is the user or system to which the certificate is issued. Here, it is "FABRIZIO D'AMORE."
3. **SN=D'AMORE:**
   - Surname or Last Name. In this case, "D'AMORE."
4. **GN=FABRIZIO:**
   - Given Name or First Name. Here, "FABRIZIO."
5. **serialNumber=TINIT-DMRFRZ60P04H501I:**
   - Serial Number. A unique identifier associated with the subject. In this example, it includes a specific serial number.
6. **dnQualifier=TITT030423203857337:**
   - Distinguished Name (DN) qualifier. It provides additional information to distinguish between entities with similar Distinguished Names.
7. **O=Università degli Studi di Roma "La Sapienza":**
   - Organization. Indicates the organization to which the subject belongs. In this case, it is "Università degli Studi di Roma 'La Sapienza'."
8. **organizationIdentifier=VATIT-80209930587:**
   - Organization Identifier. Specifies a unique identifier for the organization. Here, it includes an identifier related to the VAT number in Italy.

The **Subject** field contains a structured set of attributes describing the entity associated with the certificate. It includes details about the individual's name, organization, and unique identifiers.

These details are part of the certificate's X.509 structure and contain essential information for authentication and validation purposes.

This part of the certificate information is related to the Subject Public Key. Let's break it down:

Subject Public Key Info: Public Key Algorithm: rsaEncryption Public-Key: (2048 bit) Modulus: 00:a1:c2:6f:ad:97:91:9c:fb:88:c2:f0:de:22:f8: ... (truncated for brevity) ... 40:79:25:7a:d1:d9:6c:7a:f7 Exponent: 65537 (0x10001)

1. **Public Key Algorithm (rsaEncryption):**
    - Indicates the algorithm used for the public key. In this case, it's RSA encryption.
2. **Public-Key (2048 bit):**
    - Specifies the size of the public key. Here, it's a 2048-bit key.
3. **Modulus:**
    - The modulus is a part of the RSA algorithm and represents the product of large prime numbers. It is a key component of the public key.
    - The modulus is a long hexadecimal number representing the RSA public key modulus.
4. **Exponent (65537):**
    - The public exponent, a constant value used in the RSA algorithm. In this case, it's 65537 (0x10001).

This section provides information about the RSA public key used in the certificate. The modulus and exponent together constitute the public key, and the algorithm used for encryption is RSA. The modulus is a large number that forms the basis for the security of the RSA algorithm

This part of the X.509 certificate contains various extensions and additional information:

X509v3 extensions: X509v3 Certificate Policies: Policy: 1.3.76.33.1.1.20 CPS: https://www.trusttechnologies.it/download/documentazione/ User Notice: Explicit Text: Il titolare fa uso del certificato solo per le finalit di lavoro per le quali esso ك rilasciato. Explicit Text: The certificate holder must use the certificate only for the purposes for which it is issued. Policy: 0.4.0.194112.1.2 Policy: 1.3.76.16.6 qcStatements: 0..0......F..0......F.....0......F..0S.....F..0I0G.Ahttps://www.trusttechnologies.it/download/disclosure-statement-qc..EN0......F..0......F... X509v3 Subject Alternative Name: othername: 1.3.6.1.4.1.1466.115.121.1.50::+393286762246 X509v3 Key Usage: critical Non Repudiation X509v3 Authority Key Identifier: 7D:87:2D:F8:1F:B3:B0:D4:B6:89:80:2C:B4:AA:CE:DE:3C:08:22:06 X509v3 CRL Distribution Points: Full Name: URI:http://ca.tipki.it/TTQTSPCA1/CDP10 X509v3 Subject Key Identifier: 5B:A0:E6:4A:00:8B:ED:D9:0D:4B:28:D5:15:5D:45:FA:93:24:80:17 Authority Information Access: CA Issuers - URI:http://ca.tipki.it/TTQTSPCA1/CERT OCSP - URI:http://ocsp.tipki.it

1. **X509v3 Certificate Policies:**
    - Describes the policies associated with the certificate.
    - Includes Policy Identifiers, Certificate Practice Statement (CPS) links, and User Notices.
2. **qcStatements:**
    - Indicates Qualified Certificate Statements, providing additional details about the certificate.
3. **X509v3 Subject Alternative Name:**
    - Specifies alternative names associated with the subject of the certificate.
4. **X509v3 Key Usage:**
    - Critical extension indicating the intended purpose of the public key.
    - In this case, "Non Repudiation" is specified.
5. **X509v3 Authority Key Identifier:**
    - Provides a unique identifier for the issuing certificate authority.
6. **X509v3 CRL Distribution Points:**

   - Specifies the locations where Certificate Revocation Lists (CRLs) are available.

7. **X509v3 Subject Key Identifier:**
   - Provides a unique identifier for the subject's public key.
8. **Authority Information Access:**
   - Includes URIs for accessing information about the issuing certificate authority.

These extensions and information provide additional context and usage details for the X.509 certificate.

This part refers to the signature of the certificate, which is created using the private key of the certificate issuer.

Signature Algorithm: sha256WithRSAEncryption Signature Value:
92:9c:12:4f:55:cb:25:15:39:c4:a6:03:b1:80:b9:d8:e4:d9: ... f8:80:33:ed

1. **Signature Algorithm: sha256WithRSAEncryption:**
   - Indicates the algorithm used for the digital signature, in this case, SHA-256 with RSA encryption.
2. **Signature Value:**
   - Represents the actual signature value.
   - The signature is a series of hexadecimal values separated by colons.
   - Each pair of hexadecimal values represents a byte in the signature.
   - The full signature is too long to display entirely, so it's truncated with an ellipsis (...).

In RSA, the process involves generating a hash of the certificate contents and then encrypting this hash with the private key of the certificate issuer. The resulting encrypted hash is the digital signature.

In this case, the signature algorithm is SHA-256 with RSA encryption, and the signature value contains the actual result of this process. Each pair of hexadecimal values represents a byte in the binary signature. The signature is crucial for verifying the authenticity and integrity of the certificate.

This command extracts and prints the entire PKCS#7 structure (including certificates and the signature) and saves it in a file named **signature.pem**.

```
C:\Users\Asus\hm06>openssl pkcs7 -inform DER -in ch.txt.p7m -print_certs -out signature.pem -text -noout
```

Result in base_64 :

```
C:\Users\Asus\hm06>openssl base64 -in signature.pem -out signature_base64.txt
```

This command extracts the public key from a certificate file (**certificate.pem**) and writes it to a new file (**public_key.pem**). The **-pubkey** option specifies that the public key should be output, and **-noout** prevents any additional information from being included in the output.

```
C:\Users\Asus\hm06>openssl x509 -pubkey -noout -in certificate.pem > public_key.pem
```

This command base64-encodes the content of the **public_key.pem** file and writes the result to a new file (**public_key_base64.txt**).

```
C:\Users\Asus\hm06>openssl base64 -in public_key.pem -out public_key_base64.txt
```

### Verification :

This commands use OpenSSL to perform CMS (Cryptographic Message Syntax) verification on a PKCS#7 file (ch.txt.p7m). The CMS verification checks the digital signature and optionally verifies the signer's certificate.

In summary, the verification process involves initiating CMS verification, specifying the input PKCS#7 file, configuring verification options, extracting the signature, and confirming the successful verification of the digital signature. The final result is the display of the original message content that was signed.

```
C:\Users\Asus\hm06>openssl cms -verify -in ch.txt.p7m -inform DER -noverify -binary -out signature.der
CMS Verification successful
```

**Result - Message Content:**

- The displayed content, **"Challenge for HW06"** is the original message content that was signed and included in the PKCS#7 structure.

```
C:\Users\Asus\hm06>type signature.der
Challenge for HW06
```