# **Importance of Specifying the Full CA’s Trust Chain in a PEM Certificat**

To create a .pem file for a certificate signed by the intermediary certificate authority (CA) caInter2, which itself is signed by the intermediary CA caInter1, and finally signed by the root CA caRoot0, you need to include the entire certificate chain in the .pem file. The file should include the following PEM blocks in this order:

1. **The end-entity (server or user) certificate**: This is the certificate that was signed by caInter2.
2. **The intermediate CA certificate of caInter2**: This is the certificate of caInter2, which was signed by caInter1.
3. **The intermediate CA certificate of caInter1**: This is the certificate of caInter1, which was signed by the root CA caRoot0.
4. **The root CA certificate (caRoot0)**: This is the self-signed root CA certificate.

Each of these certificates needs to be in PEM format and concatenated together. The structure of the .pem file should look like this:

-----BEGIN CERTIFICATE-----

[End-entity certificate signed by caInter2]

-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----

[Intermediate CA certificate of caInter2 signed by caInter1]

-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----

[Intermediate CA certificate of caInter1 signed by caRoot0]

-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----

[Root CA certificate caRoot0]

-----END CERTIFICATE-----

NOTE

Including only the end-entity certificate and the root CA certificate in the .pem file is not standard practice because it omits the intermediate certificates necessary for constructing the full chain of trust. However, depending on the context and how the consuming application handles certificate chains, it might be possible, but it's not generally recommended.

Here's why including the full chain is important:

1. **Chain of Trust**: Intermediate certificates (caInter2 and caInter1) are needed to create a complete chain of trust from the end-entity certificate back to the root CA. Without the intermediate certificates, some clients might not be able to verify the end-entity certificate even if they trust the root CA, if they process strictly , according to PKI RFC4158
2. **Client Compatibility**: Many clients (e.g., browsers, servers, software libraries and commercial products) are conformant to RFC5280 (CRL verification procedures) and RFC4158 (Certification path validation) expect a complete chain in the presented certificate file. If the intermediates are not included, the client may reject the certificate due to the missing links in the chain.

**Important Considerations**

1. **Application Behavior**: Ensure that the application or client consuming this .pem file can correctly locate and use the intermediate certificates if they are not included. This might involve the intermediates being pre-installed or retrievable from a well-known location.
2. **Security**: Omitting intermediate certificates can lead to issues where the chain of trust is not established, potentially leading to untrusted connections or communication failures.
3. **Best Practices**: Follow the best practice of including all intermediate certificates unless there is a compelling reason not to and you are sure of the behavior of all involved systems and clients.

In summary, while it's technically possible to include just the end-entity and root CA certificates in the .pem file, it's not recommended due to potential verification issues. The preferred method is to include the complete certificate chain to ensure compatibility and proper chain of trust validation.

**RFC References**

1. RFC 5280: Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile

RFC 5280 is a critical document that defines the standards for X.509 certificates and their validation. According to this RFC, a complete chain of trust must be established from the end-entity certificate to a trusted root CA. The relevant sections are:

Section 6.1. Basic Certificate Processing:

This section describes the certificate path validation algorithm, which requires constructing a chain of certificates from the end-entity certificate to a trust anchor (usually a root CA certificate).

The primary goal of path validation is to verify the integrity of a chain of certificates, ensuring each certificate in the chain is properly signed by its issuer and that the issuer's certificate is authorized for the given context.

To meet this goal, the algorithm requires:

(a) obtaining the certificates in the path,

(b) verifying the signatures on those certificates,

(c) ensuring the certificates have not expired,

(d) checking the revocation status of the certificates,

(e) ensuring the certificates conform to the constraints specified in each certificate.

2. RFC 4158: Internet X.509 Public Key Infrastructure: Certification Path Building

RFC 4158 describes the procedures for constructing certification paths and further emphasizes the importance of intermediate certificates in creating a complete chain of trust:

Section 1. Introduction:

A certification path is a sequence of certificates that enables a certificate user to verify the signature on a certificate. The path starts with a certificate whose public key is trusted by the certificate user and ends with the target certificate.

Section 4.2.1. Basic Path Construction:

This section explains the necessity of including intermediate certificates for the validation process to be successful.

Practical Implications

Clients, such as web browsers, email clients, and other applications that use TLS/SSL, follow these standards to ensure secure and trustworthy communication. When these clients receive a certificate, they need to verify its authenticity and validity by checking the entire certificate chain up to a trusted root CA. If any intermediate certificates are missing, the client cannot complete this validation process, resulting in a failure to establish a secure connection.

Summary

The requirement for clients to check the complete chain of trust is rooted in the need to adhere to these standards (RFC 5280 and RFC 4158). These standards ensure that the chain of certificates from the end-entity to the trusted root is intact, verified, and trusted, thereby maintaining the security and integrity of the communication.

By including all intermediate certificates in the .pem file, you ensure that any client, regardless of its implementation, can correctly validate the certificate chain according to these widely accepted standards.