Internal Assessment Test II Solution– Dec 2021

| Sub: | Cryptography | Sub Code: | 18CS744 |
|---|---|---|---|

Q.1 (a). Explain the Elgamal Cryptographic system

Ans:

- In 1984, T. Elgamal announced a public-key scheme based on discrete logarithms, closely related to the Diffie-Hellman technique.
- The Elgamal2 cryptosystem is used in some form in a number of standards including the digital signature standard (DSS).
- As with Diffie-Hellman, the global elements of Elgamal are a prime number q and a, which is a primitive root of q. User A generates a private/public key pair as follows:

1. Generate a random integer $X_A$, such that $1 < X_A < q - 1$.
2. Compute $Y^A = \alpha^{X_A} \bmod q$.
3. A's private key is $X_A$ and A's public key is $\{q, \alpha, Y_A\}$.

Any user B that has access to A's public key can encrypt a message as follows:

1. Represent the message as an integer $M$ in the range $0 \leq M \leq q - 1$. Longer messages are sent as a sequence of blocks, with each block being an integer less than $q$.
2. Choose a random integer $k$ such that $1 \leq k \leq q - 1$.
3. Compute a one-time key $K = (Y_A)^k \bmod q$.
4. Encrypt $M$ as the pair of integers $(C_1, C_2)$ where

$$C_1 = \alpha^k \bmod q; C_2 = KM \bmod q$$

User A recovers the plaintext as follows:

1. Recover the key by computing $K = (C_1)^{X_A} \bmod q$.
2. Compute $M = (C_2 K^{-1}) \bmod q$.

We can restate the Elgamal process as follows, using Figure 10.3.

1. Bob generates a random integer $k$.
2. Bob generates a one-time key $K$ using Alice's public-key components $Y_A$, $q$, and $k$.
3. Bob encrypts $k$ using the public-key component $\alpha$, yielding $C_1$. $C_1$ provides sufficient information for Alice to recover $K$.
4. Bob encrypts the plaintext message $M$ using $K$.
5. Alice recovers $K$ from $C_1$ using her private key.
6. Alice uses $K^{-1}$ to recover the plaintext message from $C_2$.

Thus, K functions as a one-time key, used to encrypt and decrypt the message.

| Global Public Elements | |
|---|---|
| $q$ | prime number |
| $\alpha$ | $\alpha < q$ and $\alpha$ a primitive root of $q$ |

| Key Generation by Alice | |
|---|---|
| Select private $X_A$ | $X_A < q - 1$ |
| Calculate $Y_A$ | $Y_A = \alpha^{X_A} \bmod q$ |
| Public key | $\{q, \alpha, Y_A\}$ |
| Private key | $X_A$ |

| Encryption by Bob with Alice's Public Key | |
|---|---|
| Plaintext: | $M < q$ |
| Select random integer $k$ | $k < q$ |
| Calculate $K$ | $K = (Y_A)^k \bmod q$ |
| Calculate $C_1$ | $C_1 = \alpha^k \bmod q$ |
| Calculate $C_2$ | $C_2 = KM \bmod q$ |
| Ciphertext: | $(C_1, C_2)$ |

| Decryption by Alice with Alice's Private Key | |
|---|---|
| Ciphertext: | $(C_1, C_2)$ |
| Calculate $K$ | $K = (C_1)^{X_A} \bmod q$ |
| Plaintext: | $M = (C_2 K^{-1}) \bmod q$ |

Figure 10.3 The Elgamal Cryptosystem

Q1. b) Explain automatic key distribution scheme for connection oriented protocol.

- The approach assumes that communication makes use of a connection-oriented end-to-end protocol, such as TCP.
- The noteworthy element of this approach is a session security module (SSM).

1. When one host wishes to set up a connection to another host, it transmits a connection request packet (step 1).
2. The SSM saves that packet and applies to the KDC for permission to establish the connection (step 2).
3. The communication between the SSM and the KDC is encrypted using a master key shared only by this SSM and the KDC. If the KDC approves the connection request, it generates the session key and delivers it to the two appropriate SSMs, using a unique permanent key for each SSM (step 3).
4. The requesting SSM can now release the connection request packet, and a connection is set up between the two end systems (step 4).
5. All user data exchanged between the two end systems are encrypted by their respective SSMs using the one-time session key.

The automated key distribution approach provides the flexibility and dynamic characteristics needed to allow a number of terminal users to access a number of hosts and for the hosts to exchange data with each other.

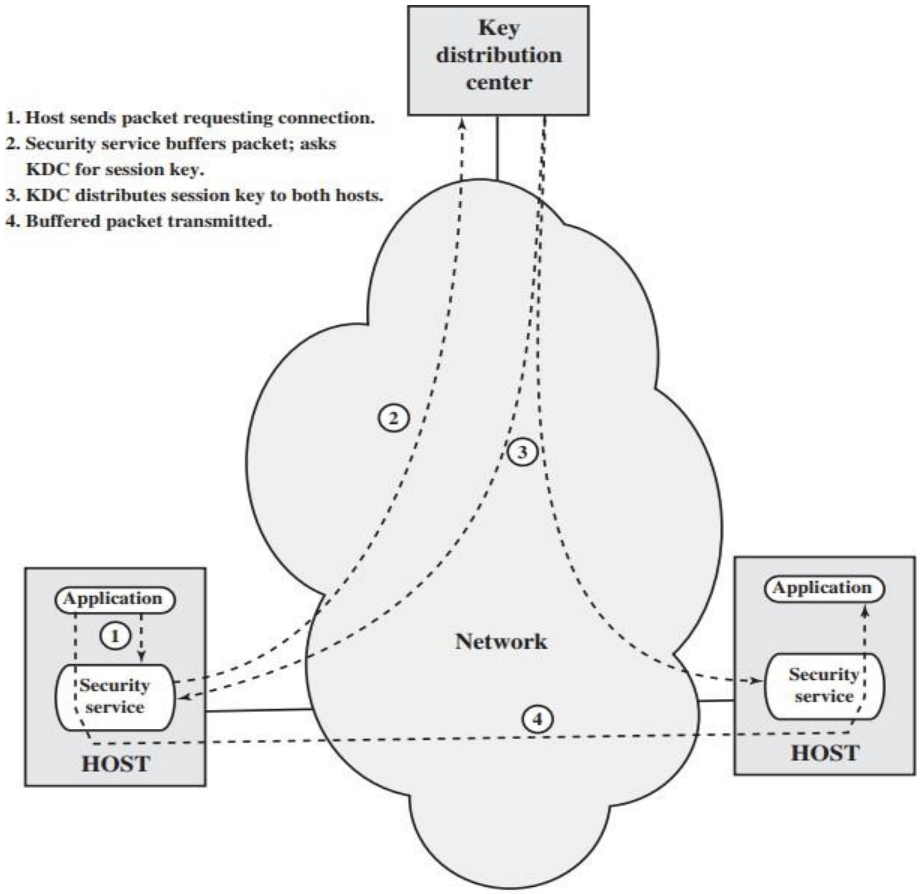



Figure 14.4 Automatic Key Distribution for Connection-Oriented Protocol

Q.2. Explain Elliptic curves over $Z_p$

Ans:

1. Elliptic curve cryptography makes use of elliptic curves in which the variables and coefficients are all restricted to elements of a finite field.
2. Two families of elliptic curves are used in cryptographic applications: prime curves over Zp and binary curves over GF($2^m$).
3. For a prime curve over Zp, we use a cubic equation in which the variables and coefficients all take on values in the set of integers from 0 through p - 1 and in which calculations are performed modulo p.
4. For a binary curve defined over GF($2^m$), the variables and coefficients all take on values in GF($2^m$) and in calculations are performed over GF($2^m$).
5. The prime curves are best for software applications, because the extended bit-fiddling operations needed by binary curves are not required; and those binary curves are best for hardware applications, where it takes remarkably few logic gates to create a powerful, fast cryptosystem.

6. We examine these two families in this section and the next. There is no obvious geometric interpretation of elliptic curve arithmetic over finite fields.
7. The algebraic interpretation used for elliptic curve arithmetic over real numbers does readily carry over, and this is the approach we take.
8. For elliptic curves over Zp, as with real numbers, we limit ourselves to equations of the form of Equation $y^2 = x^3 + ax + b$, but in this case with coefficients and variables limited to Zp:

$$y^2 \bmod p = (x^3 + ax + b) \bmod p \qquad \textbf{(10.5)}$$

For example, Equation (10.5) is satisfied for $a = 1, b = 1, x = 9, y = 7, p = 23$:

$$7^2 \bmod 23 = (9^3 + 9 + 1) \bmod 23$$
$$49 \bmod 23 = 739 \bmod 23$$
$$3 = 3$$

Q.2. b. Explain distribution of key in decentralized system

Ans:

- A decentralized approach requires that each end system be able to communicate in a secure manner with all potential partner end systems for purposes of session key distribution.
- Thus, there may need to be as many as [n(n - 1)]/2 master keys for a configuration with n end systems.
- A session key may be established with the following sequence of steps.

1. A issues a request to B for a session key and includes a nonce, N1.
2. B responds with a message that is encrypted using the shared master key. The response includes the session key selected by B, an identifier of B, the value f(N1), and another nonce, N2.
3. Using the new session key, A returns f(N2) to B.

Thus, although each node must maintain at most (n - 1) master keys, as many session keys as required may be generated and used. Because the messages transferred using the master key are short, cryptanalysis is difficult. As before, session keys are used for only a limited time to protect them.

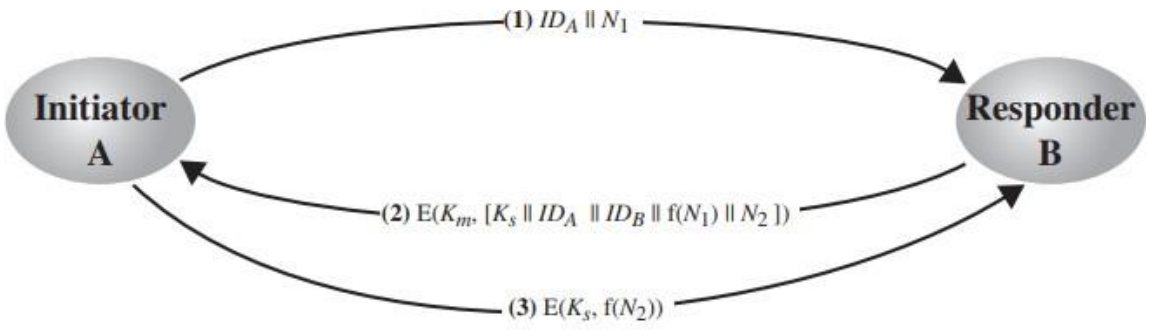



Figure 14.5 Decentralized Key Distribution

Q.3. List the ways in which secret keys can be distributed to two communicating parties. Explain symmetric key distribution using symmetric encryption.

Ans:

For two parties A and B, key distribution can be achieved in a number of ways, as follows:

1. A can select a key and physically deliver it to B.
2. A third party can select the key and physically deliver it to A and B.
3. If A and B have previously and recently used a key, one party can transmit the new key to the other, encrypted using the old key.
4. If A and B each has an encrypted connection to a third-party C, C can deliver a key on the encrypted links to A and B.

**Symmetric key distribution using symmetric encryption:**

Let us assume that user A wishes to establish a logical connection with B and requires a one-time session key to protect the data transmitted over the connection. A has a master key, Ka, known only to itself and the KDC; similarly, B shares the master key $K_b$ with the KDC. The following steps occur.

**1.** A issues a request to the KDC for a session key to protect a logical connection to B. The message includes the identity of A and B and a unique identifier, N1, for this transaction, which we refer to as a nonce.

**2.** The KDC responds with a message encrypted using Ka. Thus, A is the only one who can successfully read the message, and A knows that it originated at the KDC. The message includes two items intended for A:

- The one-time session key, $K_s$, to be used for the session
- The original request message, including the nonce, to enable A to match this response with the appropriate request.

Thus, A can verify that its original request was not altered before reception by the KDC and, because of the nonce, that this is not a replay of some previous request.

In addition, the message includes two items intended for B:

- The one-time session key, Ks, to be used for the session
- An identifier of A (e.g., its network address), $ID_A$

These last two items are encrypted with $K_b$ (the master key that the KDC shares with B). They are to be sent to B to establish the connection and prove A's identity.

**3.** A stores the session key for use in the upcoming session and forwards to B the information that originated at the KDC for B, namely, E ($K_b$, [$K_s$||$ID_A$]). Because this information is encrypted with $K_b$, it is protected from eavesdropping. B now knows the session key ($K_s$), knows that the other party is A (from $ID_A$), and knows that the information originated at the KDC (because it is encrypted using $K_b$).

At this point, a session key has been securely delivered to A and B, and they may begin their protected exchange. However, two additional steps are desirable:

**4.** Using the newly minted session key for encryption, B sends a nonce, N2, to A.

**5.** Also, using Ks, A responds with f(N2), where f is a function that performs some transformation on N2.

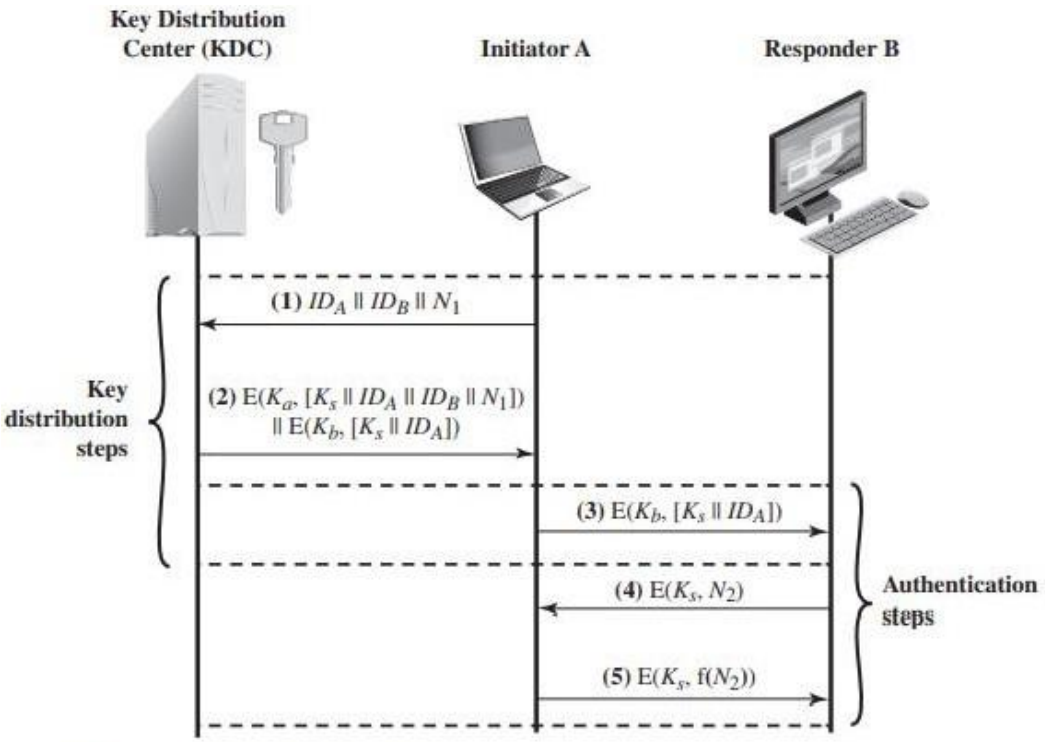



Figure 14.3 Key Distribution Scenario

Q.4. Determine all the points on the Elliptic curve $E_{11}(1,6)$. Consider the point G=(2,7) on the curve. Calculate 2G and prove that the resultant point also lies on the curve.

Ans:

The points are:

(2,4), (2,7), (3,5), (3,6),(5,2),(5,9),(7,2),(7,9),(8,3),(8,8),(10,2), (10,9)

To find 2G = G+G

Let 2G is $(X_3,Y_3)$

$\alpha = 3\,x_2^2$ +a / 2 $y_1$ = 13/14 mod 11 ≡ 2/3 mod 11 ≡ 8

$X_3 = \alpha^2 - X_1 - X_2$ = 60 mod 11 ≡ 5

$Y_3 = \alpha ( X_1 - X_3) - Y_1$ = 8*(2-5)-7 ≡ -31 mod 11 ≡ -9 mod 11 ≡ 2

Prove that (5,2) lies on the curve.

$Y^2$ mod 11=$x^3$+x+6 mod 11

LHS: 4 mod 11 = 4

RHS: 136 mod 11 = 4

So it is proved.

Q.5. Explain X.509 certificate format with a neat diagram. Show the steps of exchanging the certificates between two users A and B if user A belongs to CA Y and B belongs to CA Z.

Ans:

The heart of the X.509 scheme is the public-key certificate associated with each user. Theseuser certificates are assumed to be created by some trusted certification authority (CA) andplaced in the directory by the CA or by the user.

- **Version:** Differentiates among successive versions of the certificate format; the default is version 1. If the *issuer unique identifier* or *subject unique identifier* are present, the value must be version 2. If one or more extensions are present, the version must be version 3.
- **Serial number:** An integer value unique within the issuing CA that is unambiguously associated with this certificate.
- **Signature algorithm identifier:** The algorithm used to sign the certificate together with any associated parameters. Because this information is repeated in the signature field at the end of the certificate, this field has little, if any, utility.
- **Issuer name:** X.500 name of the CA that created and signed this certificate.
- **Period of validity:** Consists of two dates: the first and last on which the certificate is valid.
- **Subject name:** The name of the user to whom this certificate refers. That is, this certificate certifies the public key of the subject who holds the corresponding private key.
- **Subject's public-key information:** The public key of the subject, plus an identifier of the algorithm for which this key is to be used, together with any associated parameters.
- **Issuer unique identifier:** An optional-bit string field used to identify uniquely the issuing CA in the event the X.500 name has been reused for different entities.

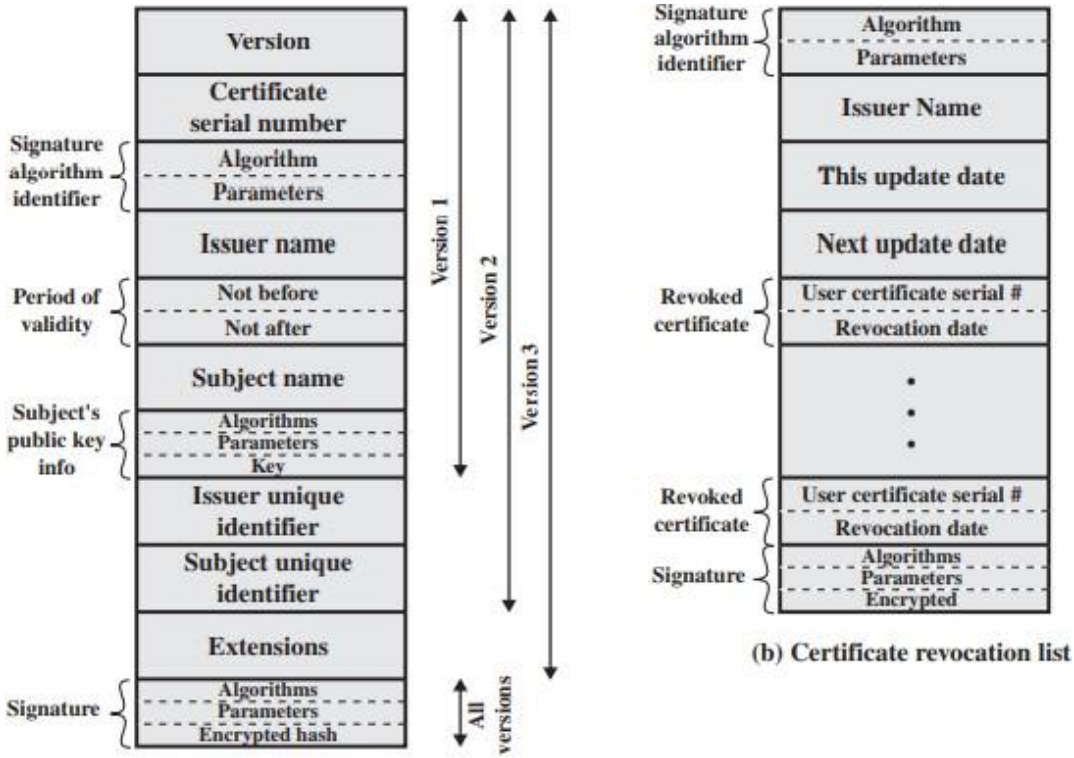



Figure 14.15 X.509 Formats

Q. 6. List four general categories of schemes for public key distribution. Explain public key certificate and publicly available directory method.

Ans:

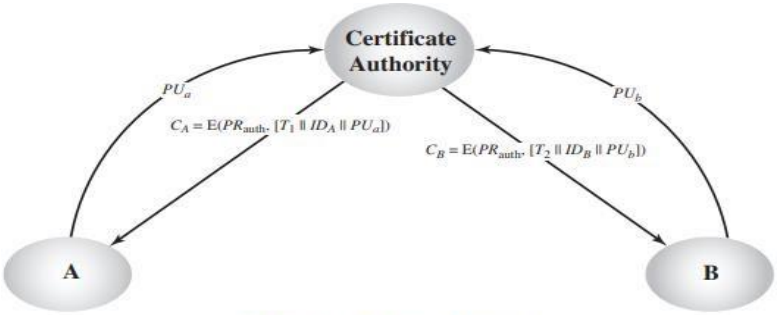
**Four general categories of schemes for public key distribution:**

- Public announcement
- Publicly available directory
- Public-key authority
- Public-key certificates

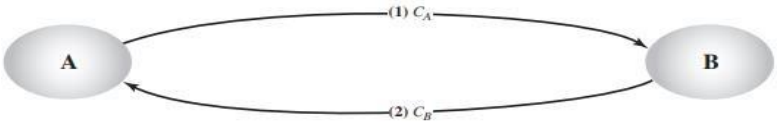
**Public key certificate**

1. To overcome the drawbacks of public key authority method, an alternative approach, first suggested by Kohnfelder, is to use certificates that can be used by participants to exchange keys without contacting a public-key authority.
2. In essence, a certificate consists of a public key, an identifier of the key owner, and the whole block signed by a trusted third party.
3. Typically, the third party is a certificate authority, such as a government agency or a financial institution, that is trusted by the user community.

4. A user can present his or her public key to the authority in a secure manner and obtain a certificate. The user can then publish the certificate.
5. Anyone needing this user's public key can obtain the certificate and verify that it is valid by way of the attached trusted signature. We can place the following requirements on this scheme:
    a. Any participant can read a certificate to determine the name and public key of the certificate's owner.
    b. Any participant can verify that the certificate originated from the certificate authority and is not counterfeit.
    c. Only the certificate authority can create and update certificates.
    d. Any participant can verify the currency of the certificate.




(a) Obtaining certificates from CA




(b) Exchanging certificates

Figure 14.13 Exchange of Public-Key Certificates

**Publicly available directory**

1. A greater degree of security can be achieved by maintaining a publicly available dynamic directory of public keys.
2. Maintenance and distribution of the public directory would have to be the responsibility of some trusted entity or organization.
3. Such a scheme would include the following elements:
    a. The authority maintains a directory with a {name, public key} entry for each participant.
    b. Each participant registers a public key with the directory authority. Registration would have to be in person or by some form of secure authenticated communication.
    c. A participant may replace the existing key with a new one at any time, either because of the desire to replace a public key that has already been used for a large amount of data, or because the corresponding private key has been compromised in some way.
    d. Participants could also access the directory electronically. For this purpose, secure, authenticated communication from the authority to the participant is mandatory.

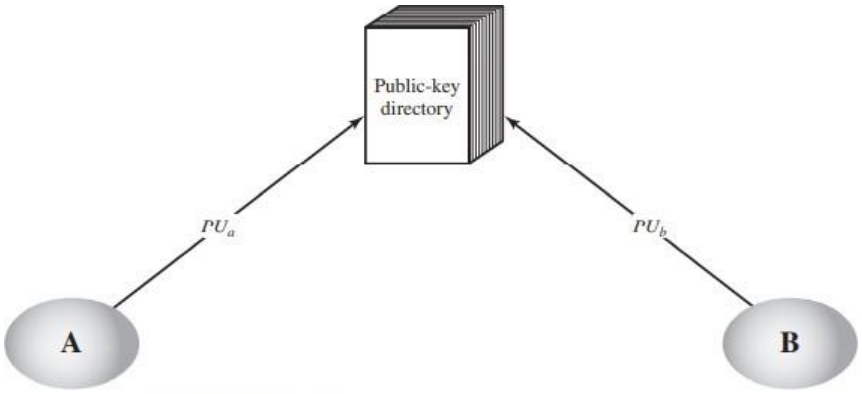



Figure 14.11 Public-Key Publication

- **Subject unique identifier:** An optional-bit string field used to identify uniquely the subject in the event the X.500 name has been reused for different entities.
- **Extensions:** A set of one or more extension fields. Extensions were added in version 3 and are discussed later in this section.
- **Signature:** Covers all of the other fields of the certificate; it contains the hash code of the other fields encrypted with the CA's private key. This field includes the signature algorithm identifier.