范既可作为网上银行系统建设和改造升级的安全性依据，也可作为各单位开展安全检查和内部审计的依据。

第 5 章包括 4 节内容：系统标识、系统定义、系统描述、安全性描述。系统描述又分为 3 小节：客户端（含专用安全设备）、通信网络、服务器端。

第 6 章包括 3 节内容：安全技术规范、安全管理规范、业务运作安全管理。安全技术规范又分为 4 小节：客户端安全（客户端程序、客户端环境安全），专用安全设备安全（USB Key、文件证书、OTP 令牌、动态密码卡、其他专用安全设备），网络通信安全（通讯协议、安全认证），服务器端安全（物理安全、网络安全、主机安全、应用安全、数据安全及备份恢复）。安全管理规范又分为 6 小节：安全管理机构、安全策略、管理制度、人员安全管理、系统建设管理、系统运维管理。业务运作安全规范又分为 3 小节：业务申请及开通、业务安全交易机制（身份认证、交易流程、交易监控）、客户教育及权益保护。

附录 A 将网上银行分为 7 个安全区域：外联区、DMZ 区、生产应用区、生产数据区、管理区域、测试区域、系统互联区域等。

附录 C 从以下 10 个方面规范了物理安全的基本要求：物理位置的选择、物理访问控制、防盗窃和防破坏、防雷击、防火、防水和防潮、防静电、温湿度控制、电力供应、电磁保护。

第29章 国际标准

29.1 PKCS 系列

PKCS 是公钥密码标准（Public Key Cryptography Standards）的缩写，它是由美国 RSA 实验室与遍布全球的安全系统开发者一起合作制定的一组规范，以推动公钥密码的发展。最早发布的 PKCS 文档是早期一群公钥技术使用者在 1991 年召开的一次会议的成果；目前 PKCS 规范已被广泛引用和实施，部分 PKCS 规范已经成为多个国际组织正式或事实上的标准，如 ANSI X9 文档系列、PKIX、SET、S/MIME、SSL 等。PKCS 系列主要包括以下标准。

PKCS #1：RSA Cryptography Standard（RSA 密码标准）；
PKCS #2：已撤销，用以规范 RSA 加密摘要的转换方式，已并入 PKCS#1；
PKCS #3：Diffie-Hellman Key Agreement Standard（DH 密钥协商标准）；
PKCS #4：已撤销，用以定义 RSA 密钥的格式，已并入 PKCS #1；
PKCS #5：Password-Based Cryptography Standard（基于口令的密码标准）；
PKCS #6：Extended-Certificate Syntax Standard（扩展的证书语法标准）；
PKCS #7：Cryptographic Message Syntax Standard（密码消息语法标准）；
PKCS #8：Private-Key Information Syntax Standard（私钥信息语法标准）；
PKCS #9：Selected Attribute Types（可供选择的属性类型）；
PKCS #10：Certification Request Syntax Standard（证书请求语法标准）；
PKCS #11：Cryptographic Token Interface Standard（密码 Token 接口标准）；
PKCS #12：Personal Information Exchange Syntax Standard（个人信息交换语法标准）；
PKCS #13：Elliptic Curve Cryptography Standard（椭圆曲线密码标准），正在制定中；
PKCS #14：Pseudo-random Number Generation（伪随机数生成算法 PRNG），正在制定中；
PKCS #15：Cryptographic Token Information Format Standard（密码 Token 信息格式标准）。

1. PKCS #1：RSA Cryptography Standard（RSA 密码标准）

PKCS #1 v2.1 定义了基于 RSA 公钥算法的加密解密和签名验签机制，其最新跟踪标准为 RFC 3447，主要包括以下内容。

① 密钥类型：包括公钥和私钥。

RSA 公钥格式用 ASN.1 描述如下：

```
RSAPublicKey ::= SEQUENCE {
    modulus           INTEGER,  -- n
    publicExponent    INTEGER   -- e
}
```

RSA 私钥格式用 ASA.1 描述如下：

```
RSAPrivateKey ::= SEQUENCE {
    version             Version,
    modulus              INTEGER,    -- n
    publicExponent       INTEGER,    -- e
    privateExponent      INTEGER,    -- d
    prime1               INTEGER,    -- p
    prime2               INTEGER,    -- q
    exponent1            INTEGER,    -- d mod (p-1)
    exponent2            INTEGER,    -- d mod (q-1)
    coefficient          INTEGER,    -- (inverse of q) mod p
    otherPrimeInfos      OtherPrimeInfos OPTIONAL
}
```

② 数据转换原子操作：包括 I2OSP（Integer-to-Octet-String primitive）和 OS2IP（Octet-String-to-Integer primitive）两种。

③ 密码原子操作：包括加密 RSAEP、解密 RSADP、签名 RSASP1 和验签 RSAVP1 等 4 种。

④ 加密解密方案：包括 RSAES-OAEP 和 RSAES-PKCS-v1_5 等两种。

⑤ 签名验签方案：包括 RSASSA-PSS 和 RSASSA-PKCS-v1_5 等两种。

⑥ 签名编码方法：包括 EMSA-PSS 和 EMSA-PKCS-v1_5 等两种。

2. PKCS #3：Diffie-Hellman Key Agreement Standard（DH 密钥协商标准）

PKCS #3 v1.4 描述了一种基于 DH 算法进行密钥协商的方法。无需预先沟通，交易双方就可以协商出一个只有双方知道的秘密密钥，该密钥可以对后续双方的数据通信进行加密保护。

3. PKCS #5：Password-Based Cryptography Standard（基于口令的密码标准）

PKCS #5 v2.0 描述了一种基于口令产生对称密钥的方法。使用 MD2 或 MD5 从口令中派生密钥，并采用 DES 的 CBC 模式加密。这个功能主要用于加密从一个计算机传送到另一个计算机的私人密钥，而不是用于加密消息。该方法在 RFC 2898 中重新发布，主要包括以下内容：

① 密钥获取函数：包括 PBKDF1 和 PBKDF2。

② 加密解密方案：包括 PBES1 和 PBES2。

③ 消息认证方案：包括 MAC 产生和 MAC 验证等。

4. PKCS #6：Extended-Certificate Syntax Standard（扩展的证书语法标准）

PKCS #6 v1.5 描述扩展证书的语法格式。该扩展证书只是对 X.509 证书格式进行了扩展，并兼容 X.509 证书格式。扩展证书格式用 ASN.1 描述如下：

```
ExtendedCertificateOrCertificate ::= CHOICE {
    certificate Certificate, -- X.509
    extendedCertificate [0] IMPLICIT ExtendedCertificate
}
ExtendedCertificate ::= SEQUENCE {
```

```
    extendedCertificateInfo ExtendedCertificateInfo,
    signatureAlgorithm SignatureAlgorithmIdentifier,
    signature Signature }
SignatureAlgorithmIdentifier ::= AlgorithmIdentifier
Signature ::= BIT STRING
ExtendedCertificateInfo ::= SEQUENCE {
    version Version,
    certificate Certificate,
    attributes Attributes }
Version ::= INTEGER
Attributes ::= SET OF Attribute
```

5. PKCS #7：Cryptographic Message Syntax Standard（密码消息语法标准）

PKCS #7 v1.5 描述了密码消息的通用语法。该语法允许嵌套，如一个数字信封可以包含另一个数字信封，或可以对已做数字信封的数据进行签名；该语法也允许扩展各种属性，还可以用于分发证书和 CRL。PKCS #7 与 PEM 兼容，可以直接将加密的消息转换成 PEM 消息，反之亦然。PKCS #7 支持多种基于证书的管理系统，PEM 就是其中之一。在 RFC 5652 中有增强定义。该标准主要包括消息通用语法和 6 种内容类型（明文、签名、信封、签名信封、摘要、密文）。

① 消息通用语法用 ASN.1 描述如下：

```
ContentInfo ::= SEQUENCE {
contentType ContentType,
content
[0] EXPLICIT ANY DEFINED BY contentType OPTIONAL }
ContentType ::= OBJECT IDENTIFIER
```

② 明文消息内容用 ASN.1 描述如下：

```
Data ::= OCTET STRING
```

③ 签名消息内容用 ASN.1 描述如下：

```
SignedData ::= SEQUENCE {
    version Version,
    digestAlgorithms DigestAlgorithmIdentifiers,
    contentInfo ContentInfo,
    certificates [0] IMPLICIT ExtendedCertificatesAndCertificates
OPTIONAL,
    Crls [1] IMPLICIT CertificateRevocationLists OPTIONAL,
    signerInfos SignerInfos }
DigestAlgorithmIdentifiers ::= SET OF DigestAlgorithmIdentifier
SignerInfos ::= SET OF SignerInfo
SignerInfo ::= SEQUENCE {
    version Version,
    issuerAndSerialNumber IssuerAndSerialNumber,
```

```
        digestAlgorithm DigestAlgorithmIdentifier,
        authenticatedAttributes [0] IMPLICIT Attributes OPTIONAL,
        digestEncryptionAlgorithm DigestEncryptionAlgorithmIdentifier,
        encryptedDigest EncryptedDigest,
        unauthenticatedAttributes [1] IMPLICIT Attributes OPTIONAL }
    EncryptedDigest ::= OCTET STRING
    DigestInfo ::= SEQUENCE {
        digestAlgorithm DigestAlgorithmIdentifier,
        digest Digest }
    Digest ::= OCTET STRING
```

④ 信封消息内容用 ASN.1 描述如下：

```
    EnvelopedData ::= SEQUENCE {
        version Version,
        recipientInfos RecipientInfos,
        encryptedContentInfo EncryptedContentInfo }
    RecipientInfos ::= SET OF RecipientInfo
    EncryptedContentInfo ::= SEQUENCE {
        contentType ContentType,
        contentEncryptionAlgorithm
        ContentEncryptionAlgorithmIdentifier,
        encryptedContent [0] IMPLICIT EncryptedContent OPTIONAL }
    EncryptedContent ::= OCTET STRING
    RecipientInfo ::= SEQUENCE {
        version Version,
        issuerAndSerialNumber IssuerAndSerialNumber,
        keyEncryptionAlgorithm
        KeyEncryptionAlgorithmIdentifier,
        encryptedKey EncryptedKey }
    EncryptedKey ::= OCTET STRING
```

⑤ 签名信封消息内容用 ASN.1 描述如下：

```
    SignedAndEnvelopedData ::= SEQUENCE {
        version Version,
        recipientInfos RecipientInfos,
        digestAlgorithms DigestAlgorithmIdentifiers,
        encryptedContentInfo EncryptedContentInfo,
        certificates [0] IMPLICIT ExtendedCertificatesAndCertificates
    OPTIONAL,
        Crls [1] IMPLICIT CertificateRevocationLists OPTIONAL,
        signerInfos SignerInfos }
```

⑥ 摘要消息内容用 ASN.1 描述如下：

```
    DigestedData ::= SEQUENCE {
```