```
des-EDE3-CBC-NoParms ALGORITHM-ID ::= { OID dES-EDE3-CBC PARMS NULL }

sha256Identifier ALGORITHM-ID ::= { OID id-SHA256 PARMS NULL }


pkcs-1 OBJECT IDENTIFIER ::= {
     iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) 1 }

rsaEncryption OBJECT IDENTIFIER ::=  { pkcs-1 1 }
-- Public key syntax
RSAPublicKey  ::=  SEQUENCE  {
     modulus INTEGER, -- n
     publicExponent INTEGER -- e
}

sha256WithRSAEncryption OBJECT IDENTIFIER  ::=  { pkcs-1 11 }

dES-EDE3-CBC OBJECT IDENTIFIER  ::=  {
     iso(1) member-body(2) us(840) rsadsi(113549) encryptionAlgorithm(3) 7 }
-- dES-EDE3-CBC parameters
CBCParameter ::= IV
IV ::= OCTET STRING (SIZE (8..8))

id-SHA256 OBJECT IDENTIFIER ::= {
     joint-iso-itu-t(2) country(16) us(840) organization(1)
     gov(101) csor(3) nistalgorithm(4) hashalgs(2) 1 }


InfoTypeAndValue ::= SEQUENCE {
      infoType       INFORMATION-ID.&id({InfoSet}),
      infoValue      INFORMATION-ID.&Type({InfoSet}{@infoType})
}

INFORMATION-ID ::= CLASS {
     &id    OBJECT IDENTIFIER UNIQUE,
     &Type
}
WITH SYNTAX { SYNTAX &Type IDENTIFIED BY &id }

InfoSet INFORMATION-ID ::= { genmInfoReqContent |
                             genpInfoResContent
}

genmInfoReqContent INFORMATION-ID ::= {
  SYNTAX         GenmInfoReqContent
  IDENTIFIED BY  id-registeredcert-mg-genminforeq
}

genpInfoResContent INFORMATION-ID ::= {
  SYNTAX         GenmInfoReqContent
  IDENTIFIED BY  id-registeredcert-mg-genpinfores
}

PKIMessage ::= SEQUENCE {
      header           PKIHeader,
      body             PKIBody,
      protection   [0] PKIProtection OPTIONAL,
      extraCerts   [1] SEQUENCE SIZE (1..MAX) OF Certificate OPTIONAL
}

PKIHeader ::= SEQUENCE {
      pvno                INTEGER     { ietf-version2 (1) },
      sender              GeneralName,
      recipient           GeneralName,
      protectionAlg   [1] AlgorithmIdentifier     OPTIONAL,
      senderKID       [2] KeyIdentifier           OPTIONAL,
      transactionID   [4] OCTET STRING,
      senderNonce     [5] OCTET STRING,
      recipNonce      [6] OCTET STRING            OPTIONAL
}

PKIBody ::= CHOICE {       -- message-specific body elements
      ir      [0]  CertReqMessages,        --Initialization Request
      ip      [1]  CertRepMessage,         --Initialization Response
      genm    [21] GenMsgContent,          --General Message
      genp    [22] GenRepContent,          --General Response
      error   [23] ErrorMsgContent         --Error Message
}

PKIProtection ::= BIT STRING

ProtectedPart ::= SEQUENCE {
      header    PKIHeader,
      body      PKIBody
}

KeyIdentifier ::= OCTET STRING

PKIStatus ::= INTEGER {
      granted                (0),
      -- you got exactly what you asked for
```