```
Appendix A - Complete ASN.1 Definition - RFC1777

Lightweight-Directory-Access-Protocol DEFINITIONS IMPLICIT TAGS ::=

BEGIN

LDAPMessage ::=
    SEQUENCE {
         messageID      MessageID,
                        -- unique id in request,
                        -- to be echoed in response(s)
         protocolOp     CHOICE {
                             searchRequest       SearchRequest,
                             searchResponse      SearchResponse,
                             modifyRequest       ModifyRequest,
                             modifyResponse      ModifyResponse,
                             addRequest          AddRequest,
                             addResponse         AddResponse,
                             delRequest          DelRequest,
                             delResponse         DelResponse,
                             modifyDNRequest     ModifyDNRequest,
                             modifyDNResponse    ModifyDNResponse,
                             compareDNRequest    CompareRequest,
                             compareDNResponse   CompareResponse,
                             bindRequest         BindRequest,
                             bindResponse        BindResponse,
                             abandonRequest      AbandonRequest,
                             unbindRequest       UnbindRequest
                        }
    }

BindRequest ::=
    [APPLICATION 0] SEQUENCE {
         version        INTEGER (1 .. 127),
                        -- current version is 2
         name           LDAPDN,
                        -- null name implies an anonymous bind
         authentication CHOICE {
                             simple        [0] OCTET STRING,
                                       -- a zero length octet string
                                       -- implies an unauthenticated
                                       -- bind.
                             krbv42LDAP    [1] OCTET STRING,
                             krbv42DSA     [2] OCTET STRING
                                       -- values as returned by
                                       -- krb_mk_req()
                                       -- Other values in later versions
                                       -- of this protocol.

                        }
    }

BindResponse ::= [APPLICATION 1] LDAPResult

UnbindRequest ::= [APPLICATION 2] NULL

SearchRequest ::=
    [APPLICATION 3] SEQUENCE {
         baseObject     LDAPDN,
         scope          ENUMERATED {
                             baseObject            (0),
                             singleLevel           (1),
```

```
                             wholeSubtree          (2)
                        },
         derefAliases   ENUMERATED {
                             neverDerefAliases     (0),
                             derefInSearching      (1),
                             derefFindingBaseObj   (2),
                             alwaysDerefAliases    (3)
                        },
         sizeLimit      INTEGER (0 .. maxInt),
                        -- value of 0 implies no sizelimit
         timeLimit      INTEGER (0 .. maxInt),
                        -- value of 0 implies no timelimit
         attrsOnly     BOOLEAN,
                        -- TRUE, if only attributes (without values)
                        -- to be returned.
         filter         Filter,
         attributes     SEQUENCE OF AttributeType
    }

SearchResponse ::=
    CHOICE {
         entry          [APPLICATION 4] SEQUENCE {
                             objectName     LDAPDN,
                             attributes     SEQUENCE OF SEQUENCE {
                                              AttributeType,
                                              SET OF
                                                AttributeValue
                                            }
                        },
         resultCode     [APPLICATION 5] LDAPResult
    }

ModifyRequest ::=
    [APPLICATION 6] SEQUENCE {
         object         LDAPDN,
         modifications  SEQUENCE OF SEQUENCE {
                             operation     ENUMERATED {
                                             add      (0),
                                             delete   (1),
                                             replace  (2)
                                           },
                             modification  SEQUENCE {
                                             type     AttributeType,
                                             values   SET OF
                                                        AttributeValue
                                           }
                        }
    }


ModifyResponse ::= [APPLICATION 7] LDAPResult

AddRequest ::=
    [APPLICATION 8] SEQUENCE {
         entry          LDAPDN,
         attrs          SEQUENCE OF SEQUENCE {
                             type          AttributeType,
                             values        SET OF AttributeValue
                        }
    }

AddResponse ::= [APPLICATION 9] LDAPResult
```

```
DelRequest ::= [APPLICATION 10] LDAPDN

DelResponse ::= [APPLICATION 11] LDAPResult

ModifyRDNRequest ::=
    [APPLICATION 12] SEQUENCE {
         entry          LDAPDN,
         newrdn         RelativeLDAPDN -- old RDN always deleted
    }

ModifyRDNResponse ::= [APPLICATION 13] LDAPResult

CompareRequest ::=
    [APPLICATION 14] SEQUENCE {
         entry          LDAPDN,
         ava            AttributeValueAssertion
    }

CompareResponse ::= [APPLICATION 15] LDAPResult

AbandonRequest ::= [APPLICATION 16] MessageID

MessageID ::= INTEGER (0 .. maxInt)

LDAPDN ::= LDAPString

RelativeLDAPDN ::= LDAPString

Filter ::=
    CHOICE {
        and            [0] SET OF Filter,
        or             [1] SET OF Filter,
        not            [2] Filter,
        equalityMatch  [3] AttributeValueAssertion,
        substrings     [4] SubstringFilter,
        greaterOrEqual [5] AttributeValueAssertion,
        lessOrEqual    [6] AttributeValueAssertion,
        present        [7] AttributeType,
        approxMatch    [8] AttributeValueAssertion
    }

LDAPResult ::=
    SEQUENCE {
        resultCode    ENUMERATED {
                        success                      (0),
                        operationsError              (1),
                        protocolError                (2),
                        timeLimitExceeded            (3),
                        sizeLimitExceeded            (4),
                        compareFalse                 (5),
                        compareTrue                  (6),
                        authMethodNotSupported       (7),
                        strongAuthRequired           (8),
                        noSuchAttribute              (16),
                        undefinedAttributeType       (17),
                        inappropriateMatching        (18),
                        constraintViolation          (19),
                        attributeOrValueExists       (20),
                        invalidAttributeSyntax       (21),
                        noSuchObject                 (32),
                        aliasProblem                 (33),
```

```
                        invalidDNSyntax              (34),
                        isLeaf                       (35),
                        aliasDereferencingProblem    (36),
                        inappropriateAuthentication  (48),
                        invalidCredentials           (49),
                        insufficientAccessRights     (50),
                        busy                         (51),
                        unavailable                  (52),
                        unwillingToPerform           (53),
                        loopDetect                   (54),
                        namingViolation              (64),
                        objectClassViolation         (65),
                        notAllowedOnNonLeaf          (66),
                        notAllowedOnRDN              (67),
                        entryAlreadyExists           (68),
                        objectClassModsProhibited    (69),
                        other                        (80)
                      },
        matchedDN     LDAPDN,
        errorMessage  LDAPString
    }

AttributeType ::= LDAPString
                -- text name of the attribute, or dotted
                -- OID representation

AttributeValue ::= OCTET STRING

AttributeValueAssertion ::=
    SEQUENCE {
        attributeType        AttributeType,
        attributeValue       AttributeValue
    }

SubstringFilter ::=
    SEQUENCE {
        type               AttributeType,
        SEQUENCE OF CHOICE {
          initial          [0] LDAPString,
          any              [1] LDAPString,
          final            [2] LDAPString
      }
    }

LDAPString ::= OCTET STRING

maxInt INTEGER ::= 65535
END
```