New ASN.1 Modules

Jim Schaad
Paul Hoffman
New Classes Defined

• New module AlgorithmClasses
  – DIGEST-ALGORITHM
  – SIGNATURE-ALGORITHM
  – PUBLIC-KEY
  – KEY-TRANSPORT
  – KEY-AGREE
  – KEY-WRAP
  – KEY-DERIVATION
  – MAC-ALGORITHM
  – CONTENT-ENCRYPTION
Class Tagging

• mda- Message Digest Algorithms
• sa- Signature Algorithms
• kta- Key Transport Algorithms (Asymmetric)
• kaa- Key Agreement Algorithms (Asymmetric)
• kwa- Key Wrap Algorithms (Symmetric)
• kda- Key Derivation Algorithms
• maca- Message Authentication Code Algorithms
• pk- Public Key
• sea- Symmetric Encryption Algorithm
Uses

mda-sha1 DIGEST-ALGORITHM ::= {
    IDENTIFIER id-sha1 PARAMS NULL ARE preferredAbsent }

pk-dsa PUBLIC-KEY ::= { IDENTIFIER id-dsa
    KEY DSAPublicKey PARAMS Dss-Parms ARE inheritable }

sa-dsa-with-sha1 SIGNATURE-ALGORITHM ::= { IDENTIFIER id-dsa-with-sha1 VALUE Dss-Sig-Value
    PARAMS NULL ARE absent USES {mda-sha1}
    PUBKEYS {pk-dsa}}
Definition of AlgorithmIdentifier

AlgorithmIdentifier{ALGORITHM-TYPE, ALGORITHM-TYPE:AlgorithmSet} ::= SEQUENCE {
  algorithm  ALGORITHM-TYPE.
    &id({AlgorithmSet}),
  parameters  ALGORITHM-TYPE.
    &Params({AlgorithmSet}[@algorithm])
OPTIONAL
}

Use of Algorithm Identifier

PublicKeyAlgId ::= AlgorithmIdentifier {
    PUBLIC-KEY,
    {PKIX-PublicKeyAlgorithms | ...} }

SignatureAlgId ::= AlgorithmIdentifier {
    SIGNATURE-ALGORITHM,
    {PKIX-SA | ... }}

PKIX-SA SIGNATURE-ALGORITHM ::= {
    sa-dsa-with-sha1 | sa-md2WithRSAEncryption |
    sa-md5WithRSAEncryption |
    sa-sha1WithRSAEncryption | sa-ecdsa-with-SHA1 }

A2C State

• Known Problems
  – Dealing with Parameterized items w/ CLASS parameters
  – Object/Object Set emissions for some fields
  – Object/Object Set support functions
  – Use of “@.” for relation constraints
Questions

• Reviews
• Moving forward
• Location of Extensibility markers