Common Cryptographic MIB (CCMIB)
draft-turner-ccmib-03

Abstract

This document defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. In particular, it describes managed objects used to manage key management implementations including asymmetric keys, symmetric keys, trust anchors, and cryptographic-related firmware.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 9, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents
carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction ................................................. 2
2. Terminology .................................................. 3
3. The Internet-Standard Management Framework ................. 3
4. Structure of the MIB module .................................. 3
5. Definition of the CC MIB module .............................. 3
   5.1. Assignments .............................................. 3
   5.2. Feature Hierarchy ........................................ 4
   5.3. Device Info ............................................... 6
   5.4. Key Management Information .............................. 24
   5.5. Key Transfer Pull ........................................ 81
   5.6. Key Transfer Push ......................................... 96
   5.7. Security Policy Information ............................. 109
   5.8. Secure Connection Information ........................... 115
6. IANA Considerations .......................................... 123
7. Security Considerations ...................................... 123
8. References ..................................................... 124
   8.1. Normative References .................................... 124
   8.2. Informative References .................................. 125
Appendix A. Contributors ......................................... 126
Authors’ Addresses .............................................. 126

1. Introduction

RFC EDITOR: PLEASE REMOVE THE FOLLOWING PARAGRAPH PRIOR TO PUBLICATION

The source for this draft is maintained in GitHub. Suggested changes should be submitted as pull requests at https://github.com/seanturner/draft-turner-ccmib. Instructions are on that page as well. Editorial changes can be managed in GitHub.

This document defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. In particular, it describes managed objects used to manage key management implementations including asymmetric keys, symmetric keys, trust anchors, and cryptographic-related firmware.
2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. The Internet-Standard Management Framework

For a detailed overview of the documents that describe the current Internet-Standard Management Framework, please refer to section 7 of [RFC3410].

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. MIB objects are generally accessed through the Simple Network Management Protocol (SNMP). Objects in the MIB are defined using the mechanisms defined in the Structure of Management Information (SMI). This memo specifies a MIB module that is compliant to the SMIv2, which is described in RFC 2578 [RFC2578], STD 58, RFC 2579 [RFC2579], and STD 58, RFC 2580 [RFC2580].

4. Structure of the MIB module

5. Definition of the CC MIB module

5.1. Assignments

This MIB module makes reference to the following document: [RFC2578].
CC-ASSIGNMENTS-MIB DEFINITIONS ::= BEGIN

IMPORTS
   MODULE-IDENTITY, enterprises
   FROM SNMPv2-SMI; -- RFC 2578

ccAssignmentsMIB MODULE-IDENTITY
   LAST-UPDATED "201609302154Z"
   ORGANIZATION "CCMIB CCB"
   CONTACT-INFO
      "CC MIB Configuration Control Board
       Email: CCMIB.CCB@us.af.mil"
   DESCRIPTION
      "This MIB defines the CC MIB tree hierarchical assignments below it and acts as a reservation mechanism.

Copyright (c) 2019 IETF Trust and the persons identified as authors of the code. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, is permitted pursuant to, and subject to the license terms contained in, the Simplified BSD License set forth in Section 4.c of the IETF Trust’s Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info).

This version of this MIB module is part of RFC xxxx; see the RFC itself for full legal notices."
   REVISION "201609302154Z"
   -- RFC EDITOR: Please update XXXX with the assigned RFC number.
   DESCRIPTION "CC MIB 1.0.5 FINAL. Published as RFC xxxx."
   ::= { ccmib 3 }

ccmib OBJECT IDENTIFIER ::= { enterprise 34493 }

--
-- Note: Current top-level OID assignments within the CC MIB tree:
-- ccmib.3 : CC-ASSIGNMENTS-MIB (this MIB)
-- ccmib.3.1 : CC-FEATURE-HIERARCHY-MIB

END

5.2. Feature Hierarchy

This MIB module makes reference to the following document: [RFC2578].
IMPORTS
ccAssignmentsMIB FROM CC-ASSIGNMENTS-MIB -- FROM Section 5.1
MODULE-IDENTITY
FROM SNMPv2-SMI; -- FROM RFC 2578

ccFeatureHierarchyMIB MODULE-IDENTITY
LAST-UPDATED "201609302154Z"
ORGANIZATION "CCMIB CCB"
CONTACT-INFO
"CC MIB Configuration Control Board
Email: CCMIB.CCB@us.af.mil"
DESCRIPTION
"This MIB defines the CC MIB features in hierarchical MIB tree assignments. It acts as a reservation mechanism for other MIB sets to be anchored below it.

Copyright (c) 2019 IETF Trust and the persons identified as authors of the code. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, is permitted pursuant to, and subject to the license terms contained in, the Simplified BSD License set forth in Section 4.c of the IETF Trust’s Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info).

This version of this MIB module is part of RFC xxxx; see the RFC itself for full legal notices."
-- RFC Ed.: RFC-editor please fill in xxxx.
REVISION "201609302154Z"
DESCRIPTION "CC MIB 1.0.5 FINAL. Published as RFC xxxx."
-- RFC Ed.: RFC-editor please fill in xxxx.
::= { ccAssignmentsMIB 1 }

ccDeviceInfo OBJECT IDENTIFIER
 ::= { ccFeatureHierarchyMIB 2 }
ccKeyManagement OBJECT IDENTIFIER
 ::= { ccFeatureHierarchyMIB 3 }
ccKeyTransferPull OBJECT IDENTIFIER
 ::= { ccFeatureHierarchyMIB 4 }
ccKeyTransferPush OBJECT IDENTIFIER
 ::= { ccFeatureHierarchyMIB 5 }
ccSecurePolicyInfo OBJECT IDENTIFIER
 ::= { ccFeatureHierarchyMIB 6 }
ccSecureConnectionInfo OBJECT IDENTIFIER
 ::= { ccFeatureHierarchyMIB 7 }
5.3. Device Info

This MIB module makes reference to the following documents: [RFC1213], [RFC2578], [RFC2579], [RFC2580], [RFC3411], and [RFC3418].

CC-DEVICE-INFO-MIB DEFINITIONS ::= BEGIN

IMPORTS
  ccDeviceInfo
  FROM CC-FEATURE-HIERARCHY-MIB -- FROM Sec 5.2
  MODULE-COMPLIANCE, OBJECT-GROUP,
  NOTIFICATION-GROUP
  FROM SNMPv2-CONF -- FROM RFC 2580
  OBJECT-TYPE, Unsigned32, NOTIFICATION-TYPE,
  MODULE-IDENTITY, TimeTicks
  FROM SNMPv2-SMI -- FROM RFC 2578
  SnmpAdminString
  FROM SNMP-FRAMEWORK-MIB -- FROM RFC 3411
  DateAndTime, TruthValue, TimeStamp
  FROM SNMPv2-TC; -- FROM RFC 2579

ccDeviceInfoMIB MODULE-IDENTITY
LAST-UPDATED "201609302154Z"
ORGANIZATION "CCMIB CCB"
CONTACT-INFO
  "CC MIB Configuration Control Board
  Email: CCMIB.CCB@us.af.mil"
DESCRIPTION
  "This MIB defines the CC MIB Device Information objects.

Copyright (c) 2019 IETF Trust and the persons identified as authors of the code. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, is permitted pursuant to, and subject to the license terms contained in, the Simplified BSD License set forth in Section 4.c of the IETF Trust’s Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info).

This version of this MIB module is part of RFC xxxx; see the RFC itself for full legal notices."
-- RFC Ed.: RFC-editor please fill in xxxx.
REVISION "201609302154Z"
DESCRIPTION ""CC MIB 1.0.5 FINAL. Published as RFC xxxx."
-- RFC Ed.: RFC-editor please fill in xxxx.

::= { ccDeviceInfo 1 }

-- *****************************************************************
-- Device Information Segments
-- *****************************************************************

cDeviceInfoConformance OBJECT IDENTIFIER
   ::= { ccDeviceInfoMIB 1 }
cDeviceComponentVersInfo OBJECT IDENTIFIER
   ::= { ccDeviceInfoMIB 2 }

-- General Device Information Scalars
-- *****************************************************************

cSystemDate OBJECT-TYPE
   SYNTAX      DateAndTime
   MAX-ACCESS  read-write
   STATUS      current
   DESCRIPTION
   "The host’s notion of the local date and time of day. Note, some
   implementations will not allow changing of this object
   and will send an inconsistentValue error."
   ::= { cDeviceInfoScalars 1 }

cSystemUpTime OBJECT-TYPE
   SYNTAX      TimeTicks
   MAX-ACCESS  read-only
   STATUS      current
   DESCRIPTION
   "The amount of time since this host was last initialized.
   Note that this is different from sysUpTime in the SNMPv2-MIB
   RFC 3418 because sysUpTime is the uptime of the network
   management portion of the system."
   ::= { cDeviceInfoScalars 2 }

cSystemInitialLoadParameters OBJECT-TYPE
   SYNTAX      SnmpAdminString (SIZE(0..128))
   MAX-ACCESS  read-write
   STATUS      current

DESCRIPTION
"This object contains the parameters (e.g., a pathname and parameter) supplied to the load device when requesting the initial operating system configuration from that device.

Note that writing to this object just changes the configuration that will be used the next time the operating system is loaded and does not actually cause the reload to occur."

::= { cDeviceInfoScalars 3 }

cSecurityLevel OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE(0..255))
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"The security level that this object is working at. Different communities of interest may have different conventions. The following values are defined and when used by agents have specific meaning: UNCLASSIFIED, RESTRICTED, CONFIDENTIAL, SECRET, TOP_SECRET."

::= { cDeviceInfoScalars 4 }

cElectronicSerialNumber OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The Electronic Serial Number of the device. This may be the chassis serial number or an internal serial number."

::= { cDeviceInfoScalars 5 }

cLastChanged OBJECT-TYPE
SYNTAX TimeTicks
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The value of cSystemUpTime the last time any configurable object within the MIBs supported by the device has been modified, created, or deleted by either SNMP, agent, or other management method (e.g., via an HMI). Managers can use this object to ensure that no changes to any configuration within the device have happened since the last time it examined the device. A value of 0 indicates that no objects have been changed since the agent initialized."

::= { cDeviceInfoScalars 6 }

cResetDevice OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"The indication of whether a device should be reset. Setting this object to ‘true’ will perform a reset operation of the device. This must not affect the state of any persistent configuration data, zeroize any of the key material or erase the audit log. When read this object should return false. When set to false this object must not perform any operation but should accept this as a valid SET operation."
::= { cDeviceInfoScalars 7 }

cSanitizeDevice OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"The indication of whether persistent data should be erased. Setting this object to ‘true’ will erase all persistent data and return the box to an uninitialized state. It will zeroize all keying data, erase all persistent storage and auditing information. Setting this object will certainly render the device unreachable from distant managers since it will be unconfigured. When read this object should return false. When set to false this object must not perform any operation but should accept this as a valid SET operation."
::= { cDeviceInfo Scalars 8 }

cRenderInoperable OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"The indication of whether persistent data should be erased. Setting this object to ‘true’ will erase all persistent data and return the box to an uninitialized state. It will zeroize all keying data, erase all persistent storage and auditing information. In addition, when supported, the device is expected to perform some internal function that will make the box unusable without returning to the factory or some equivalent. Setting this object will certainly render the device unreachable from distant managers since it will be unconfigured. When read this object should return false. When set to false this object must not perform any operation but should accept this as a valid SET operation."
::= { cDeviceInfo Scalars 9 }
cVendorName OBJECT-TYPE
SYNTAX      OCTET STRING
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"This object stores the device’s vendor name and is intended
to be displayed and meaningful to the human operator (e.g.
Flinstones Inc). In other words, this object is not intended
to store the vendor’s authoritative identification value
(i.e., sysObjectID RFC 1213)."
::= { cDeviceInfoScalars 10 }

cModelIdentifier OBJECT-TYPE
SYNTAX      OCTET STRING
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"This object stores the device’s model identifier. In
general, this would include the model name and model
number."
::= { cDeviceInfo Scalars 11 }

cHardwareVersionNumber OBJECT-TYPE
SYNTAX      OCTET STRING
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"This object stores the device’s hardware version."
::= { cDeviceInfoScalars 12 }

-- *****************************************************************
-- Device Information Notifications
-- *****************************************************************

cFirmwareInstallFailed  NOTIFICATION-TYPE
STATUS      current
DESCRIPTION
"A notification from the device to the management station
indicating a firmware install failed."
::= { cDeviceInfoNotify 1 }

cFirmwareInstallSuccess  NOTIFICATION-TYPE
OBJECTS     {
            cFirmwareName,
cFirmwareVersion,
cFirmwareSource
        }
STATUS      current
DESCRIPTION
"A notification from the device to the management station indicating a firmware install succeeded."
::= {cDeviceInfoNotify 2}

cResetDeviceInitialized NOTIFICATION-TYPE
STATUS current
DESCRIPTION
"A notification from the device to the management station indicating that the device is being reset due to a change in the value of cResetDevice. This notification should be sent before the device performs any other reset operations (such as shutting down interfaces, etc.)."
::= {cDeviceInfoNotify 3}

cSanitizeDeviceInitialized NOTIFICATION-TYPE
STATUS current
DESCRIPTION
"A notification from the device to the management station indicating that the device is being sanitized due to a change in the value of cSanitizeDevice. This notification should be sent before the device performs any other sanitize operations (such as shutting down interfaces, etc.)."
::= {cDeviceInfoNotify 4}

cTamperEventIndicated NOTIFICATION-TYPE
STATUS current
DESCRIPTION
"A notification from the device to the management station indicating that the device has detected a tamper event. This notification should be sent before the device performs any operations (such as shutting down interfaces, etc.)."
::= {cDeviceInfoNotify 5}

CBatteryLow NOTIFICATION-TYPE
OBJECTS
{ cBatteryType,
  cBatteryOpStatus,
  cBatteryLowThreshold
}
STATUS current
DESCRIPTION
"A notification from the device to the management station indicating a battery has reached the threshold at which a battery warning is indicated."
::= {cDeviceInfoNotify 6}

cBatteryRequiresReplacement NOTIFICATION-TYPE
OBJECTS  { cBatteryType, cBatteryOpStatus }
STATUS   current
DESCRIPTION
"A notification from the device to the management station indicating a battery should be charged or changed immediately."
::= { cDeviceInfoNotify 7 }

cDeviceOnBattery NOTIFICATION-TYPE
OBJECTS   { cBatteryType, cBatteryOpStatus }
STATUS   current
DESCRIPTION
"A notification from the device to the management station indicating the device is on battery power. This notification is sent when the device is no longer connected to an external power source and is operating using a battery for main power."
::= { cDeviceInfoNotify 8 }

cDeviceComponentDisabled NOTIFICATION-TYPE
OBJECTS   {
  cDeviceComponentName,
  cDeviceComponentVersion,
  cDeviceComponentOpStatus
}
STATUS   current
DESCRIPTION
"A notification from the device to the management station indicating a component described in the cDeviceComponentVersTable has been disabled."
::= { cDeviceInfoNotify 9 }

cDeviceComponentEnabled NOTIFICATION-TYPE
OBJECTS   {
  cDeviceComponentName,
  cDeviceComponentVersion
}
STATUS   current
DESCRIPTION
"A notification from the device to the management station indicating a component described in the cDeviceComponentVersTable has been enabled."
::= { cDeviceInfoNotify 10 }
cDeviceComponentVersTableCount  OBJECT-TYPE
SYNTAX    Unsigned32
MAX-ACCESS read-only
STATUS    current
DESCRIPTION
"The number of rows in the cDeviceComponentVersTable."
::= { cDeviceComponentVersInfo 1 }

cDeviceComponentVersTableLastChanged  OBJECT-TYPE
SYNTAX    TimeStamp
MAX-ACCESS read-only
STATUS    current
DESCRIPTION
"The last time any entry in the table was modified, created,
or deleted by either SNMP, agent, or other management method
(e.g., via an HMI). Managers can use this object to ensure
that no changes to configuration of this table have happened
since the last time it examined the table. A value of 0
indicates that no entry has been changed since the agent
initialized. The value in CC-DEVICE-INFO-MIB cSystemUpTime
should be used to populate this column."
::= { cDeviceComponentVersInfo 2 }

cDeviceComponentVersTable  OBJECT-TYPE
SYNTAX    SEQUENCE OF CDeviceComponentVersEntry
MAX-ACCESS not-accessible
STATUS    current
DESCRIPTION
"The table containing a description of the specification
versions of components or specifications supported by the
ECU. Note that it is possible for multiple versions of a
given specification to be registered within the table."
::= { cDeviceComponentVersInfo 3 }

CDeviceComponentVersEntry  OBJECT-TYPE
SYNTAX    CDeviceComponentVersEntry
MAX-ACCESS not-accessible
STATUS    current
DESCRIPTION
"A row containing a module descriptive name and its version
that is supported by this device."
INDEX    { cDeviceComponentName, cDeviceComponentVersion }
::= { cDeviceComponentVersTable 1 }

cDeviceComponentVersEntry  ::= SEQUENCE
{  
cDeviceComponentName        SnmpAdminString,
  cDeviceComponentVersion     SnmpAdminString,
}
cDeviceComponentName  OBJECT-TYPE
SYNTAX      SnmpAdminString (SIZE(1..32))
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"The module name or specification name. The string value to
be used in this field should be documented in the text of
the specification a given row is reporting information on.

Specification names beginning with a prefix of 'vendor-' are
reserved for private use by the vendor of the device.

The string 'device' (exact) is reserved for vendors to
register a software revision version of the device.

The string 'hardware' (exact) is reserved for vendors to
register a model number of the hardware of the device."
::= { cDeviceComponentVersEntry 1 }

cDeviceComponentVersion  OBJECT-TYPE
SYNTAX      SnmpAdminString (SIZE(1..32))
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"The version of the specification or module name listed in
the cDeviceComponentName object field in this row. The
string value to be used in this field should be documented
in the text of a specification, of the device, or elsewhere.
If the cDeviceComponentName begins with a 'vendor-' prefix,
the format of this field is vendor specific."
::= { cDeviceComponentVersEntry 2 }

cDeviceComponentOpStatus  OBJECT-TYPE
SYNTAX      INTEGER { up(1), notReady(2),
                   administrativelyDown(3) }
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
"The current operational state of the interface feature.

This row may be used to enable/disable components or modules
in the device, and some implementations may allow for
various versions of a component to be activated. Devices may
use this construct to roll back versions of a device"
software, or to allow various software feature versions to be installed.

Agents may reject the changing this object for certain rows. An example of this is changing the operational status of a row that describes the software the device and not a particular feature. In this event, the agent should return an inconsistentValue error.

::= { cDeviceComponentVersEntry 3 }

cDeviceComponentDescription OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-write
STATUS current
DESCRIPTION "A description of the component. Agents may reject the changing this object certain rows. In this event, the agent should return an inconsistentValue error."

::= { cDeviceComponentVersEntry 4 }

-- *****************************************************************
-- CC MIB cBatteryInfoTable
-- *****************************************************************

cBatteryInfoTableCount OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "The number of rows in the cBatteryInfoTable."

::= { cBatteryInfo 1 }

cBatteryInfoTableLastChanged OBJECT-TYPE
SYNTAX TimeStamp
MAX-ACCESS read-only
STATUS current
DESCRIPTION "The last time any entry in the table was modified, created, or deleted by either SNMP, agent, or other management method (e.g., via an HMI). Managers can use this object to ensure that no changes to configuration of this table have happened since the last time it examined the table. A value of 0 indicates that no entry has been changed since the agent initialized. The value in CC-DEVICE-INFO-MIB cSystemUpTime should be used to populate this column."

::= { cBatteryInfo 2 }

cBatteryInfoTable OBJECT-TYPE
The table containing information on each of the batteries installed in the device.

 ::= ( cBatteryInfo 3 )

cBatteryInfoEntry OBJECT-TYPE
SYNTAX CBatteryInfoEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"A row containing information on a specific battery. If a device cannot return status of a battery it should not create a row in this table for that battery."
INDEX { cBatteryIndex }
 ::= ( cBatteryInfoTable 1 )

CBatteryInfoEntry ::= SEQUENCE
{
  cBatteryIndex             Unsigned32,
  cBatteryType             INTEGER,
  cBatteryOpStatus         INTEGER,
  cBatteryLowThreshold     Integer32
}

cBatteryIndex OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"A numerical index used to identify the battery. This value uniquely identifies a battery on this device. The value should be persistent for a given battery, but management stations should not depend on it as it may not be possible for some devices to retain identical indexes (especially across reboots)."
 ::= ( cBatteryInfoEntry 1 )

cBatteryType OBJECT-TYPE
SYNTAX INTEGER { other(1), main(2), clock(3), security(4) }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The type of battery. Main(2) batteries are used for operation of the device when not connected to a power source. Clock(3) is used to describe batteries which cannot
provide main power to the device but maintain clock or other persistent data. Security(4) is used for batteries which perform specific security functions or which may render the device inoperable when the battery is depleted. If a battery is used for both clock and security, Security should be returned. Other(1) describes a battery which is not otherwise defined here.

::= { cBatteryInfoEntry 2 }

cBatteryOpStatus OBJECT-TYPE
SYNTAX INTEGER { unknown(1), batteryNormal(2), batteryLow(3), batteryDepleted(4), batteryMissing(5) }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Indication of the status of the battery."
::= { cBatteryInfoEntry 3 }

cBatteryLowThreshold OBJECT-TYPE
SYNTAX Integer32 (0..100)
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"The percentage of capacity at which the cBatteryLow notification will be generated. A value of zero indicates that the notification should never be sent for this battery. This object should not be implemented if the device will detect a low battery, but the actual percentage is not measurable. This object only needs be writable for implementations that support modification of the warning level percentage."
::= { cBatteryInfoEntry 4 }

-- *****************************************************************
-- CC MIB cFirmwareInformationTable
-- *****************************************************************

cFirmwareInformationTableCount OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The number of rows in the cFirmwareInformationTable."
::= { cFirmwareInfo 1 }

cFirmwareInformationTableLastChanged OBJECT-TYPE
SYNTAX TimeStamp
MAX-ACCESS  read-only  
STATUS      current  
DESCRIPTION  "The last time any entry in the table was modified, created, or deleted by either SNMP, agent, or other management method (e.g., via an HMI). Managers can use this object to ensure that no changes to configuration of this table have happened since the last time it examined the table. A value of 0 indicates that no entry has been changed since the agent initialized. The value in CC-DEVICE-INFO-MIB cSystemUpTime should be used to populate this column."  
::= { cFirmwareInfo 2 }

cFirmwareInformationTable  OBJECT-TYPE  
SYNTAX      SEQUENCE OF CFirmwareInformationEntry  
MAX-ACCESS  not-accessible  
STATUS      current  
DESCRIPTION  "A table that lists firmware versions available in the device, along with their versions and type. This is used to list currently loaded firmware versions of running firmware and other available firmware versions in support of returning to a previous version of the firmware."  
::= { cFirmwareInfo 3 }

cFirmwareInformationEntry  OBJECT-TYPE  
SYNTAX      CFirmwareInformationEntry  
MAX-ACCESS  not-accessible  
STATUS      current  
DESCRIPTION  "A row containing a firmware package name, version, and source."  
INDEX      { cFirmwareName }  
::= { cFirmwareInformationTable 1 }

CFirmwareInformationEntry ::= SEQUENCE  
{  
cFirmwareName                    OCTET STRING,  
cFirmwareVersion                 SnmpAdminString,  
cFirmwareSource                  SnmpAdminString,  
cFirmwareRunning                 TruthValue,  
cFirmwareRowStatus               RowStatus  
}

cFirmwareName  OBJECT-TYPE  
SYNTAX      OCTET STRING (SIZE(1..255))  
MAX-ACCESS  read-only  
STATUS      current  
DESCRIPTION
"Unique identifier provided in the firmware package."
::= { cFirmwareInformationEntry 1 }

cFirmwareVersion OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE(1..255))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Version of firmware (provided in the package); for legacy firm-ware packages, this column would be the empty string, ''."
::= { cFirmwareInformationEntry 2 }

cFirmwareSource OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE(1..255))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"This column is used by the implementation to describe how the firmware was received. Agents may use any string which adequately describes the interface such as 'USB.' Agents may also reference entries in the ifTable when appropriate. If received using a Cryptographic Device Material server, the exact URI that was used to retrieve the firmware package would be configured in this column."
::= { cFirmwareInformationEntry 3 }

cFirmwareRunning OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"Indicates if the firmware is currently running. Only one row in the table should have this object set to True at any given time. If this object is set from False to True, the agent must install the firmware, uninstall the previous running firmware and change the cFirmwareRunning object for the previous running firmware from True to False."
::= { cFirmwareInformationEntry 4 }

cFirmwareRowStatus OBJECT-TYPE
SYNTAX RowStatus
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"The status of the row, by which old entries may be deleted from this table."
At a minimum, implementations must support destroy management functions. Support for active, notInService, and notReady management functions is optional.

::= {cFirmwareInformationEntry 5}

-- *****************************************************************
-- Module Conformance Information
-- *****************************************************************

cDeviceInfoCompliances OBJECT IDENTIFIER ::= { cDeviceInfoConformance 1}

cDeviceInfoGroups OBJECT IDENTIFIER ::= { cDeviceInfoConformance 2}

cDeviceInfoSystemCompliance MODULE-COMPLIANCE
  STATUS current
  DESCRIPTION "Compliance levels for system information."
  MODULE
  MANDATORY-GROUPS { cDeviceInfoSystemGroup }
  GROUP cDeviceInfoSystemNotifyGroup
  DESCRIPTION "This notification group is optional for implementation."

OBJECT cSystemInitialLoadParameters
  MIN-ACCESS not-accessible
  DESCRIPTION "Implementation of this object is optional."

OBJECT cSecurityLevel
  MIN-ACCESS not-accessible
  DESCRIPTION "Implementation of this object is optional."

cSanitizeDevice
  MIN-ACCESS not-accessible
  DESCRIPTION "Implementation of this object is optional."

OBJECT cRenderInoperable
  MIN-ACCESS not-accessible
  DESCRIPTION "Implementation of this object is optional."

::= { cDeviceInfoCompliances 1 }

cDeviceInfoComponentCompliance MODULE-COMPLIANCE
STATUS    current
DESCRIPTION
  "Compliance levels for component information."
MODULE
MANDATORY-GROUPS { cDeviceInfoComponentGroup }

GROUP cDeviceInfoComponentNotifyGroup
DESCRIPTION
  "This notification group is optional for implementation."
::= { cDeviceInfoCompliances 2 }

cDeviceInfoBatteryCompliance MODULE-COMPLIANCE
STATUS    current
DESCRIPTION
  "Compliance levels for battery information."
MODULE
MANDATORY-GROUPS { cDeviceInfoBatteryGroup }

GROUP cDeviceInfoBatteryNotifyGroup
DESCRIPTION
  "This notification group is optional for implementation."

OBJECT cBatteryLowThreshold
MIN-ACCESS not-accessible
DESCRIPTION
  "Implementation of this object is optional."
::= { cDeviceInfoCompliances 3 }

cDeviceInfoFirmwareCompliance MODULE-COMPLIANCE
STATUS    current
DESCRIPTION
  "Compliance levels for firmware information."
MODULE
MANDATORY-GROUPS { cDeviceInfoFirmwareGroup }

GROUP cDeviceInfoFirmwareNotifyGroup
DESCRIPTION
  "This notification group is optional for implementation."
::= { cDeviceInfoCompliances 4 }

cDeviceInfoSystemGroup OBJECT-GROUP
OBJECTS {
  cSystemDate,
cSystemUpTime,
cSystemInitialLoadParameters,
cSecurityLevel,
cElectronicSerialNumber,
cLastChanged,
cResetDevice,
cSanitizeDevice,
cRenderInoperable,
vVendorName,
ModelIdentifier,
HardwareVersionNumber
}

STATUS current
DESCRIPTION
"This group is composed of objects related to system
information."
::= { cDeviceInfoGroups 1 }

cDeviceInfoComponentGroup OBJECT-GROUP
OBJECTS {
  cDeviceComponentVersTableCount,
cDeviceComponentVersTableLastChanged,
cDeviceComponentName,
cDeviceComponentVersion,
cDeviceComponentOpStatus,
cDeviceComponentDescription
}

STATUS current
DESCRIPTION
"This group is composed of objects related to component
information."
::= { cDeviceInfoGroups 2 }

cDeviceInfoBatteryGroup OBJECT-GROUP
OBJECTS {
  cBatteryInfoTableCount,
cBatteryInfoTableLastChanged,
cBatteryType,
cBatteryOpStatus,
cBatteryLowThreshold
}

STATUS current
DESCRIPTION
"This group is composed of objects related to battery
information."
::= { cDeviceInfoGroups 3 }

cDeviceInfoFirmwareGroup OBJECT-GROUP
OBJECTS {
  cFirmwareInformationTableCount,
cFirmwareInformationTableLastChanged,
cFirmwareName,
cFirmwareVersion,
cFirmwareSource,  
cFirmwareRunning,  
cFirmwareRowStatus
}  
STATUS current  
DESCRIPTION  
"This group is composed of objects related to firmware information."  
 ::= { cDeviceInfoGroups 4 }

cDeviceInfoSystemNotifyGroup NOTIFICATION-GROUP  
NOTIFICATIONS  
  cResetDeviceInitialized,  
cSanitizeDeviceInitialized,  
cTamperEventIndicated,  
cSanitizeDeviceInitialized
}  
STATUS current  
DESCRIPTION  
"This group is composed of notifications related to system information."  
 ::= { cDeviceInfoGroups 5 }

cDeviceInfoComponentNotifyGroup NOTIFICATION-GROUP  
NOTIFICATIONS  
  cDeviceComponentDisabled,  
cDeviceComponentEnabled
}  
STATUS current  
DESCRIPTION  
"This group is composed of notifications related to component information."  
 ::= { cDeviceInfoGroups 6 }

cDeviceInfoBatteryNotifyGroup NOTIFICATION-GROUP  
NOTIFICATIONS  
  cBatteryLow,  
cBatteryRequiresReplacement,  
cDeviceOnBattery
}  
STATUS current  
DESCRIPTION  
"This group is composed of notifications related to battery information."  
 ::= { cDeviceInfoGroups 7 }

cDeviceInfoFirmwareNotifyGroup NOTIFICATION-GROUP  
NOTIFICATIONS
cFirmwareInstallFailed,  
cFirmwareInstallSuccess  
}

STATUS current
DESCRIPTION   
"This group is composed of notifications related to firmware  
information."
::= ( cDeviceInfoGroups 8 )

END

5.4. Key Management Information

This MIB module makes references to the following documents:  
[RFC2578], [RFC2579], [RFC2580], [RFC3411], [RFC5280], [RFC5914],  
[RFC6030], and [RFC6353].

CC-KEY-MANAGEMENT-MIB DEFINITIONS ::= BEGIN

IMPORTS  
ccKeyManagement   
FROM CC-FEATURE-HIERARCHY-MIB              -- FROM Sec 5.2
OBJECT-TYPE, Unsigned32, NOTIFICATION-TYPE,  
MODULE-IDENTITY   
FROM SNMPv2-SMI                            -- FROM  
SnmpAdminString  
FROM SNMP-FRAMEWORK-MIB                    -- FROM  
RowPointer, RowStatus, DateAndTime,  
TruthValue, TimeStamp  
FROM SNMPv2-TC                             -- FROM  
MODULE-COMPLIANCE, OBJECT-GROUP,  
NOTIFICATION-GROUP  
FROM SNMPv2-CONF                           -- FROM  
SnmpTLSFingerprint  
FROM SNMP-TLS-TM-MIB;                      -- FROM  

ccKeyManagementMIB  MODULE-IDENTITY
LAST-UPDATED  "201609302154Z"
ORGANIZATION  "CCMIB CCB"
CONTACT-INFO   
"CC MIB Configuration Control Board  
Email: CCMIB.CCB@us.af.mil"
DESCRIPTION   
"This MIB defines the CC MIB Key Management objects."

Copyright (c) 2019 IETF Trust and the persons  
identified as authors of the code. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, is permitted pursuant to, and subject to the license terms contained in, the Simplified BSD License set forth in Section 4.c of the IETF Trust’s Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info).

This version of this MIB module is part of RFC xxxx; see the RFC itself for full legal notices."

-- RFC Ed.: RFC-editor please fill in xxxx.
REVISION  "201609302154Z"
DESCRIPTION  "CC MIB 1.0.5 FINAL. Published as RFC xxxx."

-- RFC Ed.: RFC-editor please fill in xxxx.
::= { ccKeyManagement 1 }

-- *****************************************************************
-- Key Management Information Segments
-- *****************************************************************

cSymmetricKeyInfo  OBJECT IDENTIFIER  ::= { ccKeyManagementMIB 1 }
cAsymKeyInfo  OBJECT IDENTIFIER  ::= { ccKeyManagementMIB 2 }
cTrustAnchorInfo  OBJECT IDENTIFIER  ::= { ccKeyManagementMIB 3 }
cCKLInfo  OBJECT IDENTIFIER  ::= { ccKeyManagementMIB 4 }
cCDMStoreInfo  OBJECT IDENTIFIER  ::= { ccKeyManagementMIB 5 }
cCertSubAltNameInfo  OBJECT IDENTIFIER  ::= { ccKeyManagementMIB 6 }
cCertPathCtrlsInfo  OBJECT IDENTIFIER  ::= { ccKeyManagementMIB 7 }
cCertPolicyInfo  OBJECT IDENTIFIER  ::= { ccKeyManagementMIB 8 }
cPolicyMappingInfo  OBJECT IDENTIFIER  ::= { ccKeyManagementMIB 9 }
cNameConstraintInfo  OBJECT IDENTIFIER  ::= { ccKeyManagementMIB 10 }
cKeyManagementScalars  OBJECT IDENTIFIER  ::= { ccKeyManagementMIB 11 }
cKeyManagementNotify  OBJECT IDENTIFIER  ::= { ccKeyManagementMIB 12 }
cKeyManagementConformance  OBJECT IDENTIFIER  ::= { ccKeyManagementMIB 13 }
cRemoteKeyMaterialInfo  OBJECT IDENTIFIER  ::= { ccKeyManagementMIB 14 }
cZeroizeAllKeys OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-write
STATUS current
DESCRIPTION  
"Setting this object to ‘true’ removes all entries in key material tables and zeroizes key materials. It is applicable to symmetric keys, asymmetric keys, and Trust Anchors (TA). It must not modify any other information in the device such as the persistent storage or the audit log. When read this object should return false. If this object is set to the same value as the current value, the device must not perform any operation but should accept this as a valid SET operation. Note after being set to true, an agent should reset this object to false once it has zeroized all the keys stored in the device."
 ::= { cKeyManagementScalars 1 }

cZeroizeSymmetricKeyTable OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-write
STATUS current
DESCRIPTION  
"Setting this object to ‘true’ removes all entries in the cSymmetricKeyTable and zeroizes the associated key materials. This operation must not modify any other information in the device such as the persistent storage or the audit log. When read this object should return false. If this object is set to the same value as the current value, the device must not perform any operation but should accept this as a valid SET operation. Note after being set to true, an agent should reset this object to false once it has zeroized the specific key materials stored in the device."
 ::= { cKeyManagementScalars 2 }

cZeroizeAsymKeyTable OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-write
STATUS current
DESCRIPTION  
"Setting this object to ‘true’ removes all entries in the cAsymKeyTable, cCertSubAltNameTable, and zeroizes the associated key materials. This operation must not modify any other information in the device such as the persistent..."
storage or the audit log. When read this object should return false. If this object is set to the same value as the current value, the device must not perform any operation but should accept this as a valid SET operation. Note after being set to true, an agent should reset this object to false once it has zeroized the specific key materials stored in the device."
::= { cKeyManagementScalars 3 }

cZeroizeTrustAnchorTable  OBJECT-TYPE
SYNTAX      TruthValue
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION  
"Setting this object to 'true' removes all entries in the cTrustAnchorTable. This operation must not modify any other information in the device such as the persistent storage or the audit log. When read this object should return false. If this object is set to the same value as the current value, the device must not perform any operation but should accept this as a valid SET operation. Note after being set to true, an agent should reset this object to false once it has zeroized the specific key materials stored in the device.

Some implementations may restrict the deletion of Trust Anchors to specific protocols (e.g., TAMP)."
::= { cKeyManagementScalars 4 }

cZeroizeCDMStoreTable  OBJECT-TYPE
SYNTAX      TruthValue
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION  
"Setting this object to 'true' removes all entries in the cCDMStoreTable that are of type symkey, asymkey, and trustAnchor. This operation must not modify any other information in the device such as the persistent storage or the audit log. When read this object should return false. If this object is set to the same value as the current value, the device must not perform any operation but should accept this as a valid SET operation. Note after being set to true, an agent should reset this object to false once it has zeroized the specific key materials stored in the device."
::= { cKeyManagementScalars 5 }

cKeyMaterialTableOID  OBJECT-TYPE
SYNTAX      OBJECT IDENTIFIER
MAX-ACCESS  read-write
The OID of the table for which (1) a successful or failed configuration occurred upon a key material load or (2) a key material has expired, will expire, or had its expiration date changed (3) a key material has been zeroized.

::= { cKeyManagementScalars 6 }

cKeyMaterialFingerprint OBJECT-TYPE
SYNTAX SnmpTLSFingerprint
MAX-ACCESS accessible-for-notify
STATUS current
DESCRIPTION "The fingerprint of the key material to be transmitted in a notification."
::= { cKeyManagementScalars 7 }

cSymKeyGlobalExpiryWarning OBJECT-TYPE
SYNTAX Unsigned32
UNITS "days"
MAX-ACCESS read-write
STATUS current
DESCRIPTION "A global setting, indicating the number of days prior to the expiration date of a symmetric key (value of cSymKeyExpirationDate in the associated cSymmetricKeyTable entry) for which the cKeyMaterialExpiring notification will be transmitted.

The value in this object is only used if no value exists for the associated cSymmetricKeyTable entry’s cSymKeyExpiryWarning object."
::= { cKeyManagementScalars 8 }

cAsymKeyGlobalExpiryWarning OBJECT-TYPE
SYNTAX Unsigned32
UNITS "days"
MAX-ACCESS read-write
STATUS current
DESCRIPTION "A global setting, indicating the number of days prior to the expiration date of an asymmetric key (value of cAsymKeyExpirationDate in the associated cAsymKeyTable entry) for which the cKeyMaterialExpiring notification will be transmitted.

The value in this object is only used if no value exists for the associated cAsymKeyTable entry’s cAsymKeyExpiryWarning object."
object.
 ::= { cKeyManagementScalars 9 }

cGenerateKeyType  OBJECT-TYPE
 SYNTAX      INTEGER { x509v3(1), psk(2) }
 MAX-ACCESS  read-write
 STATUS      current
 DESCRIPTION
 "The type of key material to be generated

[2] Symmetric Pre-Shared Key."
 ::= { cKeyManagementScalars 10 }

cGenerateKey OBJECT-TYPE
 SYNTAX      TruthValue
 MAX-ACCESS  read-write
 STATUS      current
 DESCRIPTION
 "Setting this object to ‘true’ will force the generation of
 key material, based on the type of key material described in
 cGenerateKeyType. Post-generation, the agent must create an
 entry in the appropriate key material table that captures
 information on this key.

Note after being set to true, an agent should reset this
object to false once the key material has been generated and
an entry created in the appropriate table."
 ::= { cKeyManagementScalars 11 }

-- *****************************************************************
-- Key Management Notifications
-- *****************************************************************

   cKeyMaterialLoadSuccess  NOTIFICATION-TYPE
 OBJECTS     { cKeyMaterialTableOID }
 STATUS      current
 DESCRIPTION
 "An attempt to load the device with key material, identified
 by the table identifier (e.g., cSymmetricKeyTable), has
 succeeded. This notification may be sent upon a single
 successful key material load or may be sent upon a series of
 successful single key material loads."
 ::= { cKeyManagementNotify 1 }

cKeyMaterialLoadFail  NOTIFICATION-TYPE
 OBJECTS     { cKeyMaterialTableOID }
 STATUS      current
DESCRIPTION

"An attempt to load the device with key material, identified by the table identifier (e.g., cSymmetricKeyTable), has failed."

::= { cKeyManagementNotify 2 }

cKeyMaterialExpiring  NOTIFICATION-TYPE
OBJECTS     {
    cKeyMaterialFingerprint,
    cKeyMaterialTableOID
}
STATUS      current
DESCRIPTION

"Key Material, identified by Key Fingerprint and OID of the associated key material table, is about to expire. This notification is transmitted prior to the key material’s configured expiration date (cSymKeyExpirationDate/cAsymKeyExpirationDate) as indicated by a global setting (cSymKeyGlobalExpiryWarning/cAsymKeyGlobalExpiryWarning) or the granular setting per key material table entry (cSymKeyExpiryWarning/cAsymKeyExpiryWarning) if configured."

::= { cKeyManagementNotify 3 }

cKeyMaterialExpired  NOTIFICATION-TYPE
OBJECTS     {
    cKeyMaterialFingerprint,
    cKeyMaterialTableOID
}
STATUS      current
DESCRIPTION

"Key Material, identified by Key Fingerprint and OID of the associated key material table, has expired."

::= { cKeyManagementNotify 4 }

cKeyMaterialExpirationChanged  NOTIFICATION-TYPE
OBJECTS     {
    cKeyMaterialFingerprint,
    cKeyMaterialTableOID
}
STATUS      current
DESCRIPTION

"The expiration date of Key Material, identified by Key Fingerprint and the OID of the associated key material table, has changed. This can happen by either the ‘Expiration’ object in the table changing or by the device making a change due to some other automated security policy change such as automatically extending a key when no new key
is available."
::= { cKeyManagementNotify 5 }

cKeyMaterialZeroized NOTIFICATION-TYPE
OBJECTS   {
    cKeyMaterialFingerprint,
    cKeyMaterialTableOID
}
STATUS    current
DESCRIPTION
"A key material, identified by fingerprint and OID of the
associated key material table, has been securely deleted and
zeroized. This notification is transmitted upon setting the
Row Status object of the associated key material table entry
to 'destroy', setting the cZeroizeAllKeys object to 'true',
setting the cZeroizeSymmetricKeyTable object to 'true',
setting the cZeroizeAsymKeyTable object to 'true', setting
the cZeroizeTrustAnchorTable object to 'true', or setting
the cZeroizeCDMStoreTable object to 'true'."
::= { cKeyManagementNotify 6 }

cCKLLoadSuccess NOTIFICATION-TYPE
OBJECTS   {
    cCKLIndex,
    cCKLIssuer
}
STATUS    current
DESCRIPTION
"An attempt to load the device with CKL, identified by
cCKLIndex and cCKLIssuer (indexes to the cCKLTable), has
succeeded."
::= { cKeyManagementNotify 7 }

cCKLLoadFail NOTIFICATION-TYPE
STATUS    current
DESCRIPTION
"An attempt to load the device with CKL has failed."
::= { cKeyManagementNotify 8 }

ccdMAdded NOTIFICATION-TYPE
OBJECTS   {
    cCDMStoreIndex,
    cCDMStoreType
}
STATUS    current
DESCRIPTION
"A new cryptographic device material (CDM) entry has been
added to the cCDMStoreTable, as identified cCDMStoreIndex
and cCDMStoreType.
::= { cKeyManagementNotify 9 }

cCDMDeleted NOTIFICATION-TYPE
OBJECTS
{ cCDMStoreIndex,
  cCDMStoreType,
  cCDMStoreFriendlyName
}

STATUS current

DESCRIPTION
"A cryptographic device material (CDM) entry has been
deleted from the cCDMStoreTable, as identified
cCDMStoreIndex, cCDMStoreType and cCDMStoreFriendlyName."
::= { cKeyManagementNotify 10 }

cTrustAnchorAdded NOTIFICATION-TYPE
OBJECTS
{ cTrustAnchorFingerprint,
  cTrustAnchorFormatType,
  cTrustAnchorUsageType
}

STATUS current

DESCRIPTION
"A trust anchor has been added to the cTrustAnchorTable, as
identified by cTrustAnchorFingerprint, cTrustAnchorFormatType, and cTrustAnchorUsageType."
::= { cKeyManagementNotify 11 }

cTrustAnchorUpdated NOTIFICATION-TYPE
OBJECTS
{ cTrustAnchorFingerprint,
  cTrustAnchorFormatType,
  cTrustAnchorUsageType
}

STATUS current

DESCRIPTION
"A trust anchor has been updated in the cTrustAnchorTable,
as identified by cTrustAnchorFingerprint, cTrustAnchorFormatType, and cTrustAnchorUsageType."
::= { cKeyManagementNotify 12 }

cTrustAnchorRemoved NOTIFICATION-TYPE
OBJECTS
{ cTrustAnchorFingerprint,
  cTrustAnchorFormatType,
  cTrustAnchorUsageType
}
A trust anchor has been removed from the cTrustAnchorTable, as identified by cTrustAnchorFingerprint, cTrustAnchorFormatType, and cTrustAnchorUsageType.

::= { cKeyManagementNotify 13 }

-- *****************************************************************
-- CC MIB cSymmetricKeyTable
-- *****************************************************************

cSymmetricKeyTableCount OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The number of rows in the cSymmetricKeyTable."
::= { cSymmetricKeyInfo 1 }

cSymmetricKeyTableLastChanged OBJECT-TYPE
SYNTAX TimeStamp
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The last time any entry in the table was modified, created, or deleted by either SNMP, agent, or other management method (e.g., via an HMI). Managers can use this object to ensure that no changes to configuration of this table have happened since the last time it examined the table. A value of 0 indicates that no entry has been changed since the agent initialized. The value in CC-DEVICE-INFO-MIB cSystemUpTime should be used to populate this column."
::= { cSymmetricKeyInfo 2 }

cSymmetricKeyTable OBJECT-TYPE
SYNTAX SEQUENCE OF CSymmetricKeyEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"The table containing the various types of symmetric keys used by the device."
::= { cSymmetricKeyInfo 3 }

cSymmetricKeyEntry OBJECT-TYPE
SYNTAX CSymmetricKeyEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"A row containing information about a Symmetric Key."

INDEX
{ cSymKeyFingerprint }
::= { cSymmetricKeyTable 1 }

CSymmetricKeyEntry ::= SEQUENCE {
  cSymKeyFingerprint          SnmpTLSFingerprint,
  cSymKeyUsage                BITS,
  cSymKeyID                   OCTET STRING,
  cSymKeyIssuer               OCTET STRING,
  cSymKeyEffectiveDate        DateAndTime,
  cSymKeyExpirationDate       DateAndTime,
  cSymKeyExpiryWarning        Unsigned32,
  cSymKeyNumberOfTransactions Unsigned32,
  cSymKeyFriendlyName         SnmpAdminString,
  cSymKeyClassification       BITS,
  cSymKeySource               OCTET STRING,
  cSymKeyRowStatus            RowStatus
}

cSymKeyFingerprint  OBJECT-TYPE
SYNTAX      SnmpTLSFingerprint
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
"An inherent identification of the symmetric key and the primary index to the cSymmetricKeyTable.

This MIB does not provide any additional requirements on developing the fingerprint. Implementations are cautioned to develop the hash in a manner that does not compromise the security of the key material."
::= { cSymmetricKeyEntry 1 }

cSymKeyUsage  OBJECT-TYPE
SYNTAX      BITS { oneTimePassword(0), challengeResponse(1),
                      unlock(2), encrypt(3), decrypt(4),
                      integrity(5), verify(6), keyWrap(7),
                      unwrap(8), derive(9), generate(10),
                      sharedSecret(11) }
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
"The intended usage for the key: One Time Password (OTP), Challenge/Response (CR), Unlock, Encrypt, Decrypt, Integrity, Verify, KeyWrap, Unwrap, Derive, Generate, Shared Secret. From RFC 6030 section 5.

OTP: The key is used for One Time Password (OTP) generation."
CR: The key is used for Challenge/Response purposes.

Unlock: The key is used for an inverse challenge response in the case where a user has locked the device by entering a wrong password too many times (for devices with password input capability).

Encrypt: The key is used for data encryption purposes.

Integrity: The key is used to generate a keyed message digest for data integrity or authentication purposes.

Verify: The key is used to verify a keyed message digest for data integrity or authentication purposes (this is the opposite key usage of 'Integrity').

Decrypt: The key is used for data decryption purposes.

KeyWrap: The key is used for key wrap purposes.

Unwrap: The key is used for key unwrap purposes.

Derive: The key is used with a key derivation function to derive a new key.

Generate: The key is used to generate a new key based on a random number and the previous value of the key.

Shared Secret: The key is used as a shared secret between entities.

Bit value translation:
1000 0000 0000 0000 = OneTimePassword
0100 0000 0000 0000 = ChallengeResponse
0010 0000 0000 0000 = Unlock
0001 0000 0000 0000 = Encrypt
0000 1000 0000 0000 = Decrypt
0000 0100 0000 0000 = Integrity
0000 0010 0000 0000 = Verify
0000 0001 0000 0000 = KeyWrap
0000 0000 1000 0000 = Unwrap
0000 0000 0100 0000 = Derive
0000 0000 0010 0000 = Generate
0000 0000 0001 0000 = SharedSecret

::= { cSymmetricKeyEntry 2 }

cSymKeyID OBJECT-TYPE
SYNTAX OCTET STRING (SIZE(1..255))
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"Represents a unique identifier assigned to this symmetric key. This would typically be an identifier inherent to the key material, such as a serial number or other form of identifier derived from a tag or other key wrapper. This object differs from cSymKeyFriendlyName which is a user-defined ID."
::= { cSymmetricKeyEntry 3 }

cSymKeyIssuer OBJECT-TYPE
SYNTAX OCTET STRING (SIZE(1..255))
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"Represents the name of the entity which issued the key. Use a distinguished name (DN) when one is available."
::= { cSymmetricKeyEntry 4 }

cSymKeyEffectiveDate OBJECT-TYPE
SYNTAX DateAndTime
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The effective date of the key."
::= { cSymmetricKeyEntry 5 }

cSymKeyExpirationDate OBJECT-TYPE
SYNTAX DateAndTime
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The expiration date of the key."
::= { cSymmetricKeyEntry 6 }

cSymKeyExpiryWarning OBJECT-TYPE
SYNTAX Unsigned32
UNITS "days"
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The number of days prior to the expiration date of this key (cSymKeyExpirationDate) for which the cKeyMaterialExpiring notification will be transmitted.

If configured, the scalar value of cSymKeyGlobalExpiryWarning will be ignored. The value of
cSymKeyGlobalExpiryWarning will only be used if this column is not populated, populated with 0, or not implemented.

::= { cSymmetricKeyEntry 7 }

cSymKeyNumberOfTransactions OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
"Indicates the maximum number of times a key can be used after having received it. If this column is not implemented, then there is no restriction regarding the number of times a key can be used.

When this number is reached, implementations supporting this object should stop using this key and send a cKeyMaterialExpired notification."

::= { cSymmetricKeyEntry 8 }

cSymKeyFriendlyName OBJECT-TYPE
SYNTAX      SnmpAdminString
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
"A human readable label of the key for easier reference. It is used only for helpful or informational purposes."

::= { cSymmetricKeyEntry 9 }

cSymKeyClassification OBJECT-TYPE
SYNTAX      BITS { unclassified(0), restricted(1),
              confidential(2), secret(3), topSecret(4) }
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
"The classification of the key.
Bit value translation:
1000 0000 = unclassified
0100 0000 = restricted
0010 0000 = confidential
0001 0000 = secret
0000 1000 = topSecret
This column does not exist for devices that do not have the concept of classification."

::= { cSymmetricKeyEntry 10 }

cSymKeySource OBJECT-TYPE
SYNTAX      OCTET STRING (SIZE(1..255))
MAX-ACCESS  read-create
STATUS     current
DESCRIPTION
"The source of the key material. This can be the URI of a key source entity. If the key was derived from a user-input password, the string should say PASSWORD.

Keys developed by the device should contain the string DEVICE-GENERATED. If the key was filled locally then this column should begin with the word FILL followed by the fill protocol. If the source is unknown, this column should not be populated or be set to an empty string, ’’.’’.
::= { cSymmetricKeyEntry 11 }

cSymKeyRowStatus OBJECT-TYPE
SYNTAX     RowStatus
MAX-ACCESS read-create
STATUS     current
DESCRIPTION
"The status of this row by which existing entries may be deleted from this table. Setting this column to destroy is synonymous with zeroizing the key. Any reference(s) to this object, upon setting this RowStatus to destroy, should be destroyed as well.

Upon populating this row, this column should automatically be set to notReady. Only after valid information has been entered by the manager, can the manager set this column to active.

At a minimum, implementations must support active and destroy management functions. Implementations must support createAndWait and createAndGo management functions for this object if the symmetric key material can be manually entered by the manager."
::= { cSymmetricKeyEntry 12 }

-- *****************************************************************
-- CC MIB cAsymKeyTable
-- *****************************************************************
cAsymKeyTableCount OBJECT-TYPE
SYNTAX     Unsigned32
MAX-ACCESS read-only
STATUS     current
DESCRIPTION
"The number of rows in the cAsymKeyTable."
::= { cAsymKeyInfo 1 }
cAsymKeyTableLastChanged OBJECT-TYPE
SYNTAX      TimeStamp
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"The last time any entry in the table was modified, created,
or deleted by either SNMP, agent, or other management method
(e.g., via an HMI). Managers can use this object to ensure
that no changes to configuration of this table have happened
since the last time it examined the table. A value of 0
indicates that no entry has been changed since the agent
initialized. The value in CC-DEVICE-INFO-MIB cSystemUpTime
should be used to populate this column."
::= { cAsymKeyInfo 2 }

cAsymKeyTable OBJECT-TYPE
SYNTAX      SEQUENCE OF CAsymKeyEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
"The table containing the Asymmetric Key Material and
Certificates used by the device. Enumeration values, when
applicable follow the conventions in RFC 5280."
::= { cAsymKeyInfo 3 }

CAsymKeyEntry  ::= SEQUENCE {
cAsymKeyFingerprint         SnmpTLSFingerprint,
cAsymKeyFriendlyName        SnmpAdminString,
cAsymKeySerialNumber        OCTET STRING,
cAsymKeyIssuer              OCTET STRING,
cAsymKeySignatureAlgorithm  OCTET STRING,
cAsymKeyPublicKeyAlgorithm  OCTET STRING,
cAsymKeyEffectiveDate       DateAndTime,
cAsymKeyExpirationDate      DateAndTime,
cAsymKeyExpiryWarning       Unsigned32,
cAsymKeySubject             OCTET STRING,
cAsymKeySubjectType         BITS,
cAsymKeySubjectAltName      SnmpAdminString,
cAsymKeyUsage       BITS,
cAsymKeyClassification     BITS,
cAsymKeySource          OCTET STRING,
cAsymKeyRowStatus        RowStatus,
cAsymKeyVersion          INTEGER,
cAsymKeyRekey            TruthValue,
cAsymKeyType             OCTET STRING,
cAsymKeyAutoRekeyEnable  TruthValue
}

cAsymKeyFingerprint OBJECT-TYPE
SYNTAX    SnmpTLSFingerprint
MAX-ACCESS read-only
STATUS    current
DESCRIPTION "An inherent identification of the asymmetric key and the primary index to the cAsymKeyTable."
::= { cAsymKeyEntry 1 }

cAsymKeyFriendlyName OBJECT-TYPE
SYNTAX    SnmpAdminString
MAX-ACCESS read-write
STATUS    current
DESCRIPTION "A human readable label of the key for easier reference. It is used only for helpful or informational purposes."
::= { cAsymKeyEntry 2 }

cAsymKeySerialNumber OBJECT-TYPE
SYNTAX    OCTET STRING (SIZE(1..255))
MAX-ACCESS read-only
STATUS    current
DESCRIPTION "The unique positive integer assigned to the Asymmetric Key. For Public Key Certificate (PKC) this serial number is assigned by the Certification Authority (CA). The value is this column can be up to 20 bytes long per Section '4.1.2.2. Serial Number' of RFC 5280. Other types of Key Material may have different serial number format as defined by the issuer (e.g., a Key Material ID)."
::= { cAsymKeyEntry 3 }

cAsymKeyIssuer OBJECT-TYPE
SYNTAX    OCTET STRING (SIZE(1..255))
MAX-ACCESS read-only
STATUS    current
DESCRIPTION "The issuer of this key material. For Public Key
Certificates, this is the distinguished name (DN) of the entity that has signed and issued the Public Key Certificate (PKC). Other issuers shall be defined by the class of device and will reference the Key Management System that delivers the key material for that device.

::= { cAsymKeyEntry 4 }

cAsymKeySignatureAlgorithm OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Signature algorithm used by a Certification Authority to sign this asymmetric key material (e.g., X.509 Certificate). If no signature/signature algorithm is provided/used, this column would not exist.

Note, this is a free form OCTET STRING column, meaning implementations may utilize a standardized definition of string values or use a proprietary definition of string values for supported signature algorithms."

::= { cAsymKeyEntry 5 }

cAsymKeyPublicKeyAlgorithm OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Public key algorithm with which the public key is used (as associated with the asymmetric key material (e.g., X.509 Certificate)).

Note, this is a free form OCTET STRING column, meaning implementations may utilize a standardized definition of string values or use a proprietary definition of string values for supported public key algorithms."

::= { cAsymKeyEntry 6 }

cAsymKeyEffectiveDate OBJECT-TYPE
SYNTAX DateAndTime
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"The date on which the validity period of the Asymmetric Key begins. This column must not exist when the key material does not have an inherent and associated effective date."

::= { cAsymKeyEntry 7 }
cAsymKeyExpirationDate  OBJECT-TYPE
SYNTAX    DateAndTime
MAX-ACCESS read-write
STATUS    current
DESCRIPTION
"The date on which the validity period of the Asymmetric
Key ends. This column must not exist when the key material
does not have an inherent and associated expiration date."
 ::= { cAsymKeyEntry 8 }

cAsymKeyExpiryWarning  OBJECT-TYPE
SYNTAX    Unsigned32
UNITS     "days"
MAX-ACCESS read-write
STATUS    current
DESCRIPTION
"The number of days prior to the expiration date of this
key (cAsymKeyExpirationDate) for which the
cKeyMaterialExpiring notification will be transmitted.

If configured, the scalar value of
cAsymKeyGlobalExpiryWarning will be ignored. The value of
cAsymKeyGlobalExpiryWarning will only be used if this
column is not populated, populated with 0, or not
implemented."
 ::= { cAsymKeyEntry 9 }

cAsymKeySubject  OBJECT-TYPE
SYNTAX    OCTET STRING (SIZE(1..255))
MAX-ACCESS read-only
STATUS    current
DESCRIPTION
"The entity associated with this Asymmetric Key.

For non-X.509 based key material, or when this object does
not apply for the key material, this column will not
exist."
 ::= { cAsymKeyEntry 10 }

cAsymKeySubjectType  OBJECT-TYPE
SYNTAX    BITS { other(0), certificationAuthority(1),
crlIssuer(2) }
MAX-ACCESS read-only
STATUS    current
DESCRIPTION
"Defines the type of subject based on the following
choices. certificationAuthority(1) - When set to 1
indicates that the subject (cAsymKeySubject) of the Public
Key Certificate (PKC) is a Certification Authority (CA).
crlIssuer(2) - When set to 1 indicates that the subject
certificateSubject of the Public Key Certificate (PKC)
is a Certificate Revocation List (CRL) issuer.
Bit value translation:
1000 0000 = other
0100 0000 = certificationAuthority
0010 0000 = crlIssuer
For non-X.509 based key material, or when this object does
not apply for the key material, this column will not
exist."
::= { cAsymKeyEntry 11 }

cAsymKeySubjectAltName OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE(1..32))
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"A reference string that points to a set of Certificate
Subject Alternative Subject Names in the
cCertSubAltNameTable.
This column should contain an empty string if the
Certificate has no associating Subject Alternative Names.
For non-X.509 based key material, or when this object does
not apply for the key material, this column will not
exist."
::= { cAsymKeyEntry 12 }

cAsymKeyUsage  OBJECT-TYPE
SYNTAX     BITS { other(0), digitalSignature(1),
nonRepudiation(2), keyEncipherment(3),
dataEncipherment(4), keyAgreement(5),
keyCertSign(6), cRLSign(7), encipherOnly(8),
decipherOnly(9) }
MAX-ACCESS read-write
STATUS  current
DESCRIPTION
"Provides the intended type of usage for the Asymmetric
Key. The following types are supported (defined in Section
4.2.1.3 Key Usage of RFC 5280 for PKC):
other(0), digitalSignature(1), nonRepudiation(2),
keyEncipherment(3), dataEncipherment(4), keyAgreement(5),
keyCertSign(6), cRLSign(7), encipherOnly(8), and
decipherOnly(9)
Bit value translation:
1000 0000 0000 0000 = other
0100 0000 0000 0000 = digitalSignature  
0010 0000 0000 0000 = nonRepudiation  
0001 0000 0000 0000 = keyEncipherment  
0000 1000 0000 0000 = dataEncipherment  
0000 0100 0000 0000 = keyAgreement  
0000 0010 0000 0000 = keyCertSign  
0000 0001 0000 0000 = cRLSign  
0000 0000 1000 0000 = encipherOnly  
0000 0000 0100 0000 = decipherOnly  

Devices using asymmetric key material not adhering to RFC 5280 (X.509 format) may still use an applicable value for the Usage, or may use 'other'.

```plaintext
::= { cAsymKeyEntry 13 }
```

**cAsymKeyClassification** OBJECT-TYPE  
SYNTAX BITS { unclassified(0), restricted(1), confidential(2), secret(3), topSecret(4) }  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"The supported classification level supported by the cAsymKeySubject used by this key material.  
Bit value translation:  
1000 0000 = unclassified,  
0100 0000 = restricted,  
0010 0000 = confidential,  
0001 0000 = secret,  
0000 1000 = topSecret.  

This column does not exist for devices that do not have the concept of classification."

```plaintext
::= { cAsymKeyEntry 14 }
```

**cAsymKeySource** OBJECT-TYPE  
SYNTAX OCTET STRING (SIZE(1..255))  
MAX-ACCESS read-write  
STATUS current  
DESCRIPTION  
"The source of the key material. This can be the URI of a key source entity. Keys developed by the device should contain the string DEVICE-GENERATED. If the key was filled locally then this column should begin with the word FILL followed by the fill protocol. If the source is unknown, this column should be blank."

```plaintext
::= { cAsymKeyEntry 15 }
```
MAX-ACCESS read-write  
STATUS current  
DESCRIPTION  
"The status of this row by which existing entries may be deleted from this table. Deleting a row in this table will also delete analogous rows in the cCertSubAltNameTable that are referenced by the cAsymKeySubjectAltName.

Setting this column to destroy is synonymous with zeroizing the key material. Any reference(s) to this object, upon setting this RowStatus to destroy, should be destroyed as well. At a minimum, implementations must support active and destroy management functions. Support for notInService and notReady management functions is optional. Implementations must not support createAndWait and createAndGo management functions for this object."

::= { cAsymKeyEntry 16 }

cAsymKeyVersion OBJECT-TYPE  
SYNTAX INTEGER  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"The version of the asymmetric key material. For example, X.509 Version 3 certificates would have a value of '2', as defined in RFC 5280 - Section 4.1.2.1.

When this object does not apply for the key material, this column will not exist."

::= { cAsymKeyEntry 17 }

cAsymKeyRekey OBJECT-TYPE  
SYNTAX TruthValue  
MAX-ACCESS read-create  
STATUS current  
DESCRIPTION  
"Setting this object to 'true' initiates a rekey operation for the asymmetric key material. Note, additional configurations will likely be required based on the supported key management protocol.

Note after being set to true, an agent should reset this object to false once the rekey operation has completed."

::= { cAsymKeyEntry 18 }

cAsymKeyType OBJECT-TYPE  
SYNTAX OCTET STRING (SIZE(1..255))  
MAX-ACCESS read-only
This column describes the type of asymmetric key material. Note, this is a free form OCTET STRING column. Implementations are expected to utilize definition of string values that apply to their specific nomenclature supported. If no such nomenclature exists, this column should not be populated or be set to an empty string (i.e., '').

 ::= { cAsymKeyEntry 19 }

cAsymKeyAutoRekeyEnable OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"Controls the automatic rekey settings for this PKC.
[true] Enables automatic rekey.
[false] Disables automatic rekey.

This column is optional to support."
DEFVAL { false }
 ::= { cAsymKeyEntry 20 }

-- *****************************************************************
-- CC MIB cTrustAnchorTable
-- *****************************************************************

-- ***************************************************************************
--  CC MIB cTrustAnchorTable
-- ***************************************************************************

cTrustAnchorTableCount OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The number of rows in the cTrustAnchorTable."
 ::= { cTrustAnchorInfo 1 }

cTrustAnchorTableLastChanged OBJECT-TYPE
SYNTAX TimeStamp
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The last time any entry in the table was modified, created, or deleted by either SNMP, agent, or other management method (e.g., via an HMI). Managers can use this object to ensure that no changes to configuration of this table have happened since the last time it examined the table. A value of 0
indicates that no entry has been changed since the agent initialized. The value in CC-DEVICE-INFO-MIB cSystemUpTime should be used to populate this column.

::= { cTrustAnchorInfo 2 }

cTrustAnchorTable  OBJECT-TYPE
SYNTAX      SEQUENCE OF CTrustAnchorEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
    "The table containing the Trust Anchors (TAs) in this device."
::= { cTrustAnchorInfo 3 }

cTrustAnchorEntry  OBJECT-TYPE
SYNTAX      CTrustAnchorEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
    "A row containing information about a Trust Anchor (TA) that has been loaded into the device."
INDEX      { cTrustAnchorFingerprint }
::= { cTrustAnchorTable 1 }

CTrustAnchorEntry  ::= SEQUENCE {
    cTrustAnchorFingerprint         SnmpTLSFingerprint,
    cTrustAnchorFormatType          INTEGER,  
    cTrustAnchorName                OCTET STRING,
    cTrustAnchorUsageType           INTEGER,  
    cTrustAnchorKeyIdentifier       OCTET STRING,
    cTrustAnchorPublicKeyAlgorithm  OCTET STRING,  
    cTrustAnchorContingencyAvail    TruthValue,  
    cTrustAnchorRowStatus           RowStatus, 
    cTrustAnchorVersion             OCTET STRING  
}

cTrustAnchorFingerprint  OBJECT-TYPE
SYNTAX      SnmpTLSFingerprint
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "An inherent identification of the trust anchor and the primary index to the cTrustAnchorTable."
::= { cTrustAnchorEntry 1 }

cTrustAnchorFormatType  OBJECT-TYPE
SYNTAX      INTEGER { x509v3(1), trustAnchorFormat(2),  
                         tbsCertificate(3)  
}
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION  "The type/format of the trust anchor.

 ::= { cTrustAnchorEntry 2 }

cTrustAnchorName  OBJECT-TYPE
SYNTAX      OCTET STRING (SIZE(0..255))
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION  "The name of the Trust Anchor. When available, this is the
X.500 distinguished name (DN) associated with the Trust
Anchor (TA) used to construct and validate an X.509
certification path. When the value of cTrustAnchorFormatType
is ‘trustAnchorFormat’, this column is populated with the
value from the taTitle field of the TrustAnchorInfo
structure defined in RFC 5914, which is a human-readable
name for the trust anchor. Otherwise, this column should be
blank."
 ::= { cTrustAnchorEntry 3 }

cTrustAnchorUsageType  OBJECT-TYPE
SYNTAX      INTEGER { other(1), apex(2), management(3),
                      identity(4), firmware(5), crl(6) }
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION  "The usage type for the Trust Anchor (TA). Note, crl(6) also
applies to compromised key lists."
 ::= { cTrustAnchorEntry 4 }

cTrustAnchorKeyIdentifier  OBJECT-TYPE
SYNTAX      OCTET STRING (SIZE(1..255))
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION  "The identifier of the Trust Anchor’s (TA’s) public key."
 ::= { cTrustAnchorEntry 5 }

cTrustAnchorPublicKeyAlgorithm  OBJECT-TYPE
SYNTAX      OCTET STRING
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"Public key algorithm with which the public key is used (as associated with the trust anchor).

Note, this is a free form OCTET STRING column, meaning implementations may utilize a standardized definition of string values or use a proprietary definition of string values for supported public key algorithms."

::= { cTrustAnchorEntry 6 }

cTrustAnchorContingencyAvail OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"An indication of the availability of a contingency key for an Apex Trust Anchor. When set to ’True’, a contingency key is available."

::= { cTrustAnchorEntry 7 }

cTrustAnchorRowStatus OBJECT-TYPE
SYNTAX RowStatus
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"The status of this row by which existing entries may be deleted from this table. Setting this column to destroy is synonymous with zeroizing the Trust Anchor (TA). Any reference(s) to this object, upon setting this RowStatus to destroy, should be destroyed as well.

At a minimum, implementations must support active and destroy management functions. Support for notInService and notReady management functions is optional. Implementations must not support createAndWait and createAndGo management functions for this object.

Some implementations may restrict the deletion of Trust Anchors to specific protocols (e.g., TAMP)."

::= { cTrustAnchorEntry 8 }

cTrustAnchorVersion OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The version of the Trust Anchor."

::= { cTrustAnchorEntry 9 }
cCKLTableCount OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
  "The number of rows in the cCKLTable."
::= { cCKLInfo 1 }

cCKLLastChanged OBJECT-TYPE
SYNTAX      TimeStamp
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
  "The last time any entry in the table was modified, created,
  or deleted by either SNMP, agent, or other management method
  (e.g., via an HMI). Managers can use this object to ensure
  that no changes to configuration of this table have happened
  since the last time it examined the table. A value of 0
  indicates that no entry has been changed since the agent
  initialized. The value in CC-DEVICE-INFO-MIB cSystemUpTime
  should be used to populate this column."
::= { cCKLInfo 2 }

cCKLTable OBJECT-TYPE
SYNTAX      SEQUENCE OF CCKLEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
  "The table containing the Compromised Key Lists and
  Certificate Revocation Lists (CRLs) used by the device. This
  table is used both for CRLs as defined in RFC 5280 and for
  other formats of revocation lists (such as Compromised Key
  Lists.)."
::= { cCKLInfo 3 }

cCKLEntry OBJECT-TYPE
SYNTAX      CCKLEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
  "A row containing information about a Compromised Key List
  or Certificate Revocation List (CRL) used by the device."
INDEX      { cCKLIndex, cCKLIssuer }
::= { cCKLTable 1 }
CCKLEntry ::= SEQUENCE {
  cCKLIndex  Unsigned32,
  cCKLIssuer  OCTET STRING,
  cCKLSerialNumber  OCTET STRING,
  cCKLIssueDate  DateAndTime,
  cCKLNextUpdate  DateAndTime,
  cCKLRowStatus  RowStatus,
  cCKLVersion  INTEGER,
  cCKLLastUpdate  DateAndTime
}

CCKLIndex  OBJECT-TYPE
SYNTAX   Unsigned32
MAX-ACCESS  read-only
STATUS   current
DESCRIPTION
  "An ID that uniquely identifies the Compromised Key List (CKL) in this table."
 ::= { cCKLEntry 1 }

CCKLIssuer  OBJECT-TYPE
SYNTAX   OCTET STRING (SIZE(0..255))
MAX-ACCESS  read-only
STATUS   current
DESCRIPTION
  "For devices adhering to RFC 5280 this is the X.500 distinguished name (DN) of the entity that has signed and issued the Certificate Revocation List (CRL).

Other CRL/CKL issuers may use proprietary naming conventions or formats.

If the source is unknown, this column should not be populated or be set to an empty string, ''."
 ::= { cCKLEntry 2 }

CCKLSerialNumber  OBJECT-TYPE
SYNTAX   OCTET STRING (SIZE(0..255))
MAX-ACCESS  read-only
STATUS   current
DESCRIPTION
  "A Serial Number for this CRL or CKL.

For CRLs adhering to RFC 5280, this will be a monotonically increasing sequence number for a given Certificate Revocation List (CRL) scope and CRL issuer. The CRL Number allows users to easily determine when a particular CKL/CRL supersedes another CKL/CRL."
::= { cCKLEntry 3 }

cCKLIssueDate OBJECT-TYPE
SYNTAX DateAndTime
MAX-ACCESS read-only
STATUS current
DESCRIPTION "The issue date of this CRL/CKL."
::= { cCKLEntry 4 }

cCKLNextUpdate OBJECT-TYPE
SYNTAX DateAndTime
MAX-ACCESS read-only
STATUS current
DESCRIPTION "The date by which the next CKL/CRL issued. The next CRL could be issued before the indicated date, but it will not be issued any later than the indicated date.

If this value is unknown, this column should not be populated or be set to an empty string, ''."
::= { cCKLEntry 5 }

cCKLRowStatus OBJECT-TYPE
SYNTAX RowStatus
MAX-ACCESS read-write
STATUS current
DESCRIPTION "The status of this row by which existing entries may be deleted from this table.

At a minimum, implementations must support active and destroy management functions. Support for notInService and notReady management functions is optional. Implementations must not support createAndWait and createAndGo management functions for this object."
::= { cCKLEntry 6 }

cCKLVersion OBJECT-TYPE
SYNTAX INTEGER
MAX-ACCESS read-only
STATUS current
DESCRIPTION "The version of the CKL/CRL. For example, X.509 Version 2 CRLs would have a value of ‘1’, as defined in RFC 5280 - Section 5.1.2.1.

When this object does not apply for the CKL/CRL, this column
cCKLLastUpdate  OBJECT-TYPE
SYNTAX      DateAndTime
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
  "The date this CKL/CRL was last updated."
 ::= { cCKLEntry 8 }

-- *****************************************************************
-- CC MIB cCDMStoreTable
-- *****************************************************************

cCDMStoreTableCount  OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
  "The number of rows in the cCDMStoreTable."
 ::= { cCDMStoreInfo 1 }

cCDMStoreTableLastChanged  OBJECT-TYPE
SYNTAX      TimeStamp
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
  "The last time any entry in the table was modified, created,
or deleted by either SNMP, agent, or other management method
(e.g., via an HMI). Managers can use this object to ensure
that no changes to configuration of this table have happened
since the last time it examined the table. A value of 0
indicates that no entry has been changed since the agent
initialized. The value in CC-DEVICE-INFO-MIB cSystemUpTime
should be used to populate this column."
 ::= { cCDMStoreInfo 2 }

cCDMStoreTable  OBJECT-TYPE
SYNTAX      SEQUENCE OF CCDMStoreEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
  "The table containing various types of stored Crypto Device
  Material (CDM) that are destined for this device and/or
  destined for another device. When sending CDM to a destined
device, the cCDMTransferPkgLocatorRowPtr from the
CC-KEY-TRANSFER-PUSH-MIB can be used to point to the rows in
this table.
::= { cCDMStoreInfo 3 }

cCDMStoreEntry OBJECT-TYPE
SYNTAX CCDMStoreEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"A row containing information about stored Crypto Device Material (CDM)."
INDEX { cCDMStoreIndex }
::= { cCDMStoreTable 1 }

CCDMStoreEntry ::= SEQUENCE {
    cCDMStoreIndex          Unsigned32,
    cCDMStoreType           INTEGER { symKey(1), asymKey(2), trustAnchor(3),
                                         crl(4), ckl(5), firmware(6),
                                         storeAndForwardWrappedPkg(7),
                                         storeAndForwardPkg(8) },
    cCDMStoreSource         SnmpAdminString,
    cCDMStoreID             OCTET STRING,
    cCDMStoreFriendlyName   SnmpAdminString,
    cCDMStoreControl        INTEGER,
    cCDMStoreRowStatus      RowStatus
}

cCDMStoreIndex OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"A numeric index that identifies a unique location in this table."
::= { cCDMStoreEntry 1 }

cCDMStoreType OBJECT-TYPE
SYNTAX INTEGER { symKey(1), asymKey(2), trustAnchor(3),
                          crl(4), ckl(5), firmware(6),
                          storeAndForwardWrappedPkg(7),
                          storeAndForwardPkg(8) }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The type of Crypto Device Material (CDM) populated in this row.

(1) symKey - This row contains information about a stored symmetric key.
(2) asymKey - This row contains information about a stored asymmetric key.
(3) trustAnchor - This row contains information about a
stored Trust Anchor (TA).

(4) crl - This row contains information about a stored Certificate Revocation List (CRL).

(5) ckl - This row contains information about a stored Compromised Key List (CKL).

(6) firmware - This row contains information about stored firmware.

(7) storeAndForwardWrappedPkg - This row contains information about a stored encrypted wrapped package, typically meant to be forwarded to another device.

(8) storeAndForwardPkg - This row contains information about a stored unencrypted, typically meant to be forwarded to another device.

::= { cCDMStoreEntry 2 }

cCDMStoreSource OBJECT-TYPE
SYNTAX SnmpAdminString
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"An administrative name that identifies the source of this Crypto Device Material (CDM). This could be the URI used when downloaded from the Secure Object Management System (SOMS) server or a physical port designator for CDM downloaded via HMI."

::= { cCDMStoreEntry 3 }

cCDMStoreID OBJECT-TYPE
SYNTAX OCTET STRING (SIZE(1..255))
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"Represents a unique identifier assigned to this Crypto Device Material (CDM). This would typically be an identifier inherent to the CDM, such as a serial number or other form of identifier derived from a tag or other CDM wrapper. This object differs from cCDMStoreFriendlyName which is a user-defined ID."

::= { cCDMStoreEntry 4 }

cCDMStoreFriendlyName OBJECT-TYPE
SYNTAX SnmpAdminString
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"A human readable label of this Crypto Device Material (CDM) for easier reference. It is used only for helpful or informational purposes."
::= { cCDMStoreEntry 5 }

cCDMStoreControl OBJECT-TYPE
SYNTAX INTEGER { readyForInstall(1), install(2),
                         installAndDiscard(3), other (4) }
MAX-ACCESS read-write
STATUS current
DESCRIPTION
   "A means to control what happens to the Crypto Device
    Material (CDM) stored in this table.
   (1) readyForInstall - The CDM is ready for installation.
   (2) install - The CDM will be installed in the appropriate
                  table based on the cCDMStoreType.
   (3) installAndDiscard - The CDM will be installed in the
                              appropriate table based on the cCDMStoreType and
                              discarded from this table after the install operation
                              is complete.
   (4) other - The CDM will be processed based on family
                extension specific action.

    Note, setting the cCDMStoreRowStatus object to 'destroy'
    will discard the CDM."
::= { cCDMStoreEntry 6 }

cCDMStoreRowStatus OBJECT-TYPE
SYNTAX RowStatus
MAX-ACCESS read-write
STATUS current
DESCRIPTION
   "The status of this row by which existing entries may be
    deleted from this table.

    At a minimum, implementations must support active and
    destroy management functions. Support for notInService and
    notReady management functions is optional. Implementations
    must not support createAndWait and createAndGo management
    functions for this object."
::= { cCDMStoreEntry 7 }

-- *****************************************************************
-- CCM MIB cCertSubAltNameTable
-- *****************************************************************

cCertSubAltNameTableCount OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The number of rows in the cCertSubAltNameTable."
::= { cCertSubAltNameInfo 1 }

cCertSubAltNameTableLastChanged OBJECT-TYPE
SYNTAX      TimeStamp
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
  "The last time any entry in the table was modified, created,
   or deleted by either SNMP, agent, or other management method
   (e.g., via an HMI). Managers can use this object to ensure
   that no changes to configuration of this table have happened
   since the last time it examined the table. A value of 0
   indicates that no entry has been changed since the agent
   initialized. The value in CC-DEVICE-INFO-MIB cSystemUpTime
   should be used to populate this column."
::= { cCertSubAltNameInfo 2 }

cCertSubAltNameTable OBJECT-TYPE
SYNTAX     SEQUENCE OF CcCertSubAltNameTableEntry
MAX-ACCESS not-accessible
STATUS     current
DESCRIPTION
  "The table containing a list of Subject Alternative Names
   associated with the certificate."
::= { cCertSubAltNameInfo 3 }

CcCertSubAltNameTableEntry OBJECT-TYPE
SYNTAX     CcCertSubAltNameTableEntry
MAX-ACCESS not-accessible
STATUS     current
DESCRIPTION
  "A row containing information about a Subject Alternative
   Name and its type."
INDEX  { cCertSubAltNameList, cCertSubAltNameListIndex }  ::= { cCertSubAltNameTable 1 }  
CcCertSubAltNameTableEntry ::= SEQUENCE {
  cCertSubAltNameList         SnmpAdminString,
  cCertSubAltNameListIndex    Unsigned32,
  cCertSubAltNameType         INTEGER,
  cCertSubAltNameValue1       OCTET STRING,
  cCertSubAltNameValue2       OCTET STRING,
  cCertSubAltNameRowStatus    RowStatus
}

cCertSubAltNameList OBJECT-TYPE
SYNTAX     SnmpAdminString (SIZE(1..32))
MAX-ACCESS not-accessible
STATUS      current
DESCRIPTION
"The administrative name defining the set of Subject
Alternative Names that are associated with the certificate.
Multiple Subject Alternative Names may use the same
administrative name, implying a group. It is the combination
of cCertSubAltNameList and cCertSubAltNameListIndex that
uniquely identifies each row or set of Subject Alternative
Names."
 ::= { cCertSubAltNameTableEntry 1 }

cCertSubAltNameListIndex OBJECT-TYPE
SYNTAX     Unsigned32
MAX-ACCESS not-accessible
STATUS      current
DESCRIPTION
"A unique numeric index for rows, or sets of Subject
Alternative Names, with the same cCertSubAltNameList value.
This value, in combination with cCertSubAltNameList,
uniquely identifies each row, or set of Subject Alternative
Names."
 ::= { cCertSubAltNameTableEntry 2 }

cCertSubAltNameType OBJECT-TYPE
SYNTAX     INTEGER { otherName(0), rfc822Name(1), dNSName(2),
x400Address(3), directoryName(4),
ediPartyName(5),
uniformResourceIdentifier(6), ipAddress(7),
registeredID(8) }
MAX-ACCESS read-only
STATUS      current
DESCRIPTION
"The type of the Subject Alternative Name as defined in RFC
5280, Section 4.2.1.6. Specifically, the value of this
object determines the format of cCertSubAltNameValue1 and
cCertSubAltNameValue2."
 ::= { cCertSubAltNameTableEntry 3 }

cCertSubAltNameValue1 OBJECT-TYPE
SYNTAX     OCTET STRING
MAX-ACCESS read-only
STATUS      current
DESCRIPTION
"The main value of the Subject Alternative Name. The format
of the value must match its Type as defined in RFC 5280,
Section 4.2.1.6."
This column is the main value and is used for all
cCertSubAltNameType types. For otherName(0), this column
provides the value of the ‘value’ field. For
ediPartyName(5), this column provides the value of the
‘partyName’. For all other types, this column provides the
value as defined in RFC 5280, Section 4.2.1.6.”

::= { cCertSubAltNameTableEntry 4 }

cCertSubAltNameValue2 OBJECT-TYPE
SYNTAX     OCTET STRING
MAX-ACCESS read-only
STATUS     current
DESCRIPTION
"This column is a supplement to the main value
cCertSubAltNameValue1 and may only be used when the
cCertSubAltNameType is either otherName(0) or
ediPartyName(5). For otherName(0), this column provides the
value of the ‘type-id’ as defined in RFC 5280, Section
4.2.1.6. For ediPartyName(5), this column provides the value
of the ‘nameAssigner’ as defined in RFC 5280, Section
4.2.1.6.

For all other values of cCertSubAltNameType or when the
‘nameAssigner’ is not used for ediPartyName(5), this column
will not exist.

Note: Support for multiple otherName(0) or ediPartyName(5)
alternate names is provided by allowing multiple rows of the
same cCertSubAltNameType and cCertSubAltNameList but with a
unique cCertSubAltNameListIndex.”

::= { cCertSubAltNameTableEntry 5 }

cCertSubAltNameRowStatus OBJECT-TYPE
SYNTAX     RowStatus
MAX-ACCESS read-create
STATUS     current
DESCRIPTION
"The status of this row by which existing entries may be
deleted from this table.

At a minimum, implementations must support active and
destroy management functions. Support for notInService and
notReady management functions is optional. Implementations
must not support createAndWait and createAndGo management
functions for this object.”

::= { cCertSubAltNameTableEntry 6 }

-- ------------------------------------------------------------------
cCertPathCtrlsTableCount OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
   "The number of rows in the cCertPathCtrlsTable."
 ::= { cCertPathCtrlsInfo 1 }

cCertPathCtrlsTableLastChanged OBJECT-TYPE
SYNTAX      TimeStamp
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
   "The last time any entry in the table was modified, created,
    or deleted by either SNMP, agent, or other management method
    (e.g., via an HMI). Managers can use this object to ensure
    that no changes to configuration of this table have happened
    since the last time it examined the table. A value of 0
    indicates that no entry has been changed since the agent
    initialized. The value in CC-DEVICE-INFO-MIB cSystemUpTime
    should be used to populate this column."
 ::= { cCertPathCtrlsInfo 2 }

cCertPathCtrlsTable OBJECT-TYPE
SYNTAX      SEQUENCE OF CCertPathCtrlsEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
   "The table containing the controls and constraints applied
    to a certificate in order to process certificate trust
    paths."
 ::= { cCertPathCtrlsInfo 3 }

CCertPathCtrlsEntry OBJECT-TYPE
SYNTAX      CCertPathCtrlsEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
   "A row containing information about certificate path
    controls and constraints."
INDEX  { cCertPathCtrlsKeyFingerprint }
 ::= { cCertPathCtrlsTable 1 }

CCertPathCtrlsEntry ::= SEQUENCE {
   cCertPathCtrlsKeyFingerprint    SnmpTLSFingerprint,
cCertPathCtrlsCertificate OBJECT-TYPE
SYNTAX RowPointer,
cCertPathCtrlsCertPolicies OCTET STRING,
cCertPathCtrlsPolicyMappings OCTET STRING,
cCertPathCtrlsPolicyFlags BITS,
cCertPathCtrlsNamesPermitted OCTET STRING,
cCertPathCtrlsNamesExcluded OCTET STRING,
cCertPathCtrlsMaxPathLength Unsigned32

}

cCertPathCtrlsKeyFingerprint OBJECT-TYPE
SYNTAX SnmpTLSFingerprint
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"Identifies a trust anchor in the cTrustAnchorTable or a certificate in the cAsymKeyTable. This column is the primary index to the cCertPathCtrlsTable."
::= {cCertPathCtrlsEntry 1}

cCertPathCtrlsCertificate OBJECT-TYPE
SYNTAX RowPointer
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Optional reference to an X.509 certificate defined in the cAsymKeyTable to assist with certification path development and validation."
::= {cCertPathCtrlsEntry 2}

cCertPathCtrlsCertPolicies OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Indicates a grouping of one or more policies for this certificate. The value of this column corresponds to the cCertPolicyInformation column in the cCertPolicyTable."

When this object does not apply for the key material, this column will not exist."
::= {cCertPathCtrlsEntry 3}

cCertPathCtrlsPolicyMappings OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"For a Certification Authority (CA) certificate, this
indicates a grouping of policy mappings between a certificate issuer CA domain policy and a domain policy of the subject certificate CA. The value of this column corresponds to the cPolicyMappingGroup column of the cPolicyMappingTable.

For non-X.509 based key material, or when this object does not apply for the key material, this column will not exist.

::= { cCertPathCtrlsEntry 4 }

cCertPathCtrlsPolicyFlags  OBJECT-TYPE
SYNTAX      BITS { inhibitPolicyMapping(0),
                    requireExplicitPolicy(1),
                    inhibitAnyPolicy(2) }
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"Optional certificate path policy flags consisting of the following: inhibitPolicyMapping, requireExplicitPolicy, and inhibitAnyPolicy.

inhibitPolicyMapping: Indicates if policy mapping is allowed in the certification path.

requireExplicitPolicy: Indicates if the certification path must be valid for at least one of the certificate policies in cCertPathCtrlsCertPolicies.

inhibitAnyPolicy: Indicates whether the special anyPolicy policy identifier is considered an explicit match for other certificate policies.

Bit value translation:
1000 = inhibitPolicyMapping
0100 = requireExplicitPolicy
0010 = inhibitAnyPolicy"

::= { cCertPathCtrlsEntry 5 }

cCertPathCtrlsNamesPermitted  OBJECT-TYPE
SYNTAX      OCTET STRING
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"Indicates a subtree of names that are permitted for certificate path validation. The value of this column corresponds to the cNameConstraintGenSubtree column in the cNameConstraintTable."
When this object does not apply for the key material, this column will not exist.

::= { cCertPathCtrlsEntry 6 }

cCertPathCtrlsNamesExcluded OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Indicates a subtree of names that are excluded from certificate path validation, regardless of information appearing in the cCertPathCtrlsNamesPermitted subtree. The value of this column corresponds to the cNameConstraintGenSubtree column in the cNameConstraintTable.

When this object does not apply for the key material, this column will not exist.

::= { cCertPathCtrlsEntry 7 }

cCertPathCtrlsMaxPathLength OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Optional indication of the maximum number of non-self-issued intermediate certificates that may follow this certificate in a valid certification path."

::= { cCertPathCtrlsEntry 8 }

-- *****************************************************************
-- CC MIB cCertPolicyTable
-- *****************************************************************

cCertPolicyTableCount OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The number of rows in the cCertPolicyTable."

::= { cCertPolicyInfo 1 }

cCertPolicyTableLastChanged OBJECT-TYPE
SYNTAX TimeStamp
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The last time any entry in the table was modified, created,
or deleted by either SNMP, agent, or other management method (e.g., via an HMI). Managers can use this object to ensure that no changes to configuration of this table have happened since the last time it examined the table. A value of 0 indicates that no entry has been changed since the agent initialized. The value in CC-DEVICE-INFO-MIB cSystemUpTime should be used to populate this column.

::= { cCertPolicyInfo 2 }

cCertPolicyTable OBJECT-TYPE
SYNTAX SEQUENCE OF CCertPolicyEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"The table containing certificate policy information to be provided as input to the certificate path validation algorithm. For an end entity certificate, this information indicates under which policy this certificate has been issued and the purposes for which the certificate may be used. For a Certification Authority (CA) certificate, this information limits the set of policies for certification paths that include this certificate."

::= { cCertPolicyInfo 3 }

CCertPolicyEntry OBJECT-TYPE
SYNTAX CCertPolicyEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"A row containing information about a certificate policy."
INDEX { cCertPolicyInformation, cCertPolicyInformationIndex }

 ::= { cCertPolicyTable 1 }

CCertPolicyEntry ::= SEQUENCE {
cCertPolicyInformation  OCTET STRING,
cCertPolicyInformationIndex Unsigned32,
cCertPolicyIdentifier  OBJECT IDENTIFIER,
cCertPolicyQualifierID INTEGER,
cCertPolicyQualifier  OCTET STRING
}

CcCertPolicyInformation OBJECT-TYPE
SYNTAX OCTET STRING (SIZE(1..255))
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"Identifies a grouping of policies that are applicable to a certificate. When used in conjunction with
cCertPolicyInformationIndex, a unique policy and qualifier set is defined.
::= { cCertPolicyEntry 1 }

cCertPolicyInformationIndex  OBJECT-TYPE
SYNTAX       Unsigned32
MAX-ACCESS   not-accessible
STATUS       current
DESCRIPTION  "A numerical index that is unique for a specific
cCertPolicyInformation value. This index allows multiple
qualifiers to be defined for a particular policy. When used
in conjunction with cCertPolicyInformation, a unique policy
and qualifier set is defined."
::= { cCertPolicyEntry 2 }

cCertPolicyIdentifier  OBJECT-TYPE
SYNTAX       OBJECT IDENTIFIER
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION  "For end entity certificates, this is an identifier for the
policy under which the certificate has been issued. For
Certification Authority (CA) certificates, this is an
identifier for a certification path policy that includes
this certificate."
::= { cCertPolicyEntry 3 }

cCertPolicyQualifierID  OBJECT-TYPE
SYNTAX       INTEGER { cpsPointer(0), userNotice(1) }
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION  "Indicates the type of qualifier per RFC 5280,
Section 4.2.1.4."
::= { cCertPolicyEntry 4 }

cCertPolicyQualifier  OBJECT-TYPE
SYNTAX       OCTET STRING
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION  "Qualifier information with type based on
cCertPolicyQualifierID."
::= { cCertPolicyEntry 5 }

-- *****************************************************************
-- CC MIB cPolicyMappingTable

Internet-Draft                    CCMIB                        July 2019

-- *****************************************************************

cPolicyMappingTableCount OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION  "The number of rows in the cPolicyMappingTable."
::= { cPolicyMappingInfo 1 }

cPolicyMappingTableLastChanged OBJECT-TYPE
SYNTAX      TimeStamp
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION  "The last time any entry in the table was modified, created,
or deleted by either SNMP, agent, or other management method
(e.g., via an HMI). Managers can use this object to ensure
that no changes to configuration of this table have happened
since the last time it examined the table. A value of 0
indicates that no entry has been changed since the agent
initialized. The value in CC-DEVICE-INFO-MIB cSystemUpTime
should be used to populate this column."
::= { cPolicyMappingInfo 2 }

cPolicyMappingTable OBJECT-TYPE
SYNTAX      SEQUENCE OF CPolicyMappingEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION  "The table listing mappings between policies that a
certificate issuing Certification Authority (CA) considers
as equivalent or comparable to the domain policies of the
subject certificate’s CA."
::= { cPolicyMappingInfo 3 }

CPolicyMappingEntry OBJECT-TYPE
SYNTAX      CPolicyMappingEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION  "A row containing a mapping between the domain policy of an
issuing Certification Authority (CA) and an equivalent
domain policy of the subject certificate’s CA."
INDEX  { cPolicyMappingGroup, cPolicyMappingIndex }
::= { cPolicyMappingTable 1 }

CPolicyMappingEntry ::= SEQUENCE {

cPolicyMappingGroup OCTET STRING,
cPolicyMappingIndex Unsigned32,
cPolicyMappingSubjectPolicy OBJECT IDENTIFIER,
cPolicyMappingIssuerPolicy OBJECT IDENTIFIER
}

cPolicyMappingGroup OBJECT-TYPE
SYNTAX OCTET STRING (SIZE(1..255))
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION "Identifies a grouping of policy mappings that are
applicable to a certificate. When used in conjunction with
cPolicyMappingIndex, a unique policy mapping is defined."
::= { cPolicyMappingEntry 1 }

cPolicyMappingIndex OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION "A numerical index that is unique for a specific
cPolicyMappingGroup value. When used in conjunction with
cPolicyMappingGroup, a unique policy mapping is defined."
::= { cPolicyMappingEntry 2 }

cPolicyMappingSubjectPolicy OBJECT-TYPE
SYNTAX OBJECT IDENTIFIER
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Indicates the subject Certification Authority’s domain policy."
::= { cPolicyMappingEntry 3 }

cPolicyMappingIssuerPolicy OBJECT-TYPE
SYNTAX OBJECT IDENTIFIER
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Indicates the issuer domain policy that the issuer
Certification Authority (CA) considers equivalent to the
subject CA domain policy."
::= { cPolicyMappingEntry 4 }

-- *****************************************************************
-- CC MIB cNameConstraintTable
-- *****************************************************************
cNameConstraintTableCount OBJECT-TYPE
SYNTAX    Unsigned32
MAX-ACCESS read-only
STATUS    current
DESCRIPTION
   "The number of rows in the cNameConstraintTable."
::= { cNameConstraintInfo 1 }

cNameConstraintTableLastChanged OBJECT-TYPE
SYNTAX    TimeStamp
MAX-ACCESS read-only
STATUS    current
DESCRIPTION
   "The last time any entry in the table was modified, created,
   or deleted by either SNMP, agent, or other management method
   (e.g., via an HMI). Managers can use this object to ensure
   that no changes to configuration of this table have happened
   since the last time it examined the table. A value of 0
   indicates that no entry has been changed since the agent
   initialized. The value in CC-DEVICE-INFO-MIB cSystemUpTime
   should be used to populate this column."
::= { cNameConstraintInfo 2 }

cNameConstraintTable OBJECT-TYPE
SYNTAX    SEQUENCE OF CNameConstraintEntry
MAX-ACCESS not-accessible
STATUS    current
DESCRIPTION
   "The table listing designated name spaces within which
   subject names in subsequent certificates in a certification
   path can be stored."
::= { cNameConstraintInfo 3 }

cNameConstraintEntry OBJECT-TYPE
SYNTAX    CNameConstraintEntry
MAX-ACCESS not-accessible
STATUS    current
DESCRIPTION
   "A row designating an entity’s distinguished name to a name
   space."
INDEX    { cNameConstraintGenSubtree,
                  cNameConstraintSubtreeIndex }
::= { cNameConstraintTable 1 }

CNameConstraintEntry ::= SEQUENCE {
   cNameConstraintGenSubtree   OCTET STRING,
   cNameConstraintSubtreeIndex Unsigned32,
   cNameConstraintBaseName     SnmpAdminString}
{ cNameConstraintGenSubtree  OBJECT-TYPE
SYNTAX      OCTET STRING (SIZE(1..255))
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
"Identifies a permitted or excluded name constraint subtree.
When used with cNameConstraintSubtreeIndex, a unique subject
name constraint entry is defined."
::= { cNameConstraintEntry 1 }

cNameConstraintSubtreeIndex  OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
"A numerical index used to specify a name constraint within
a permitted or excluded name constraint subtree. When used
with a specific value of cNameConstraintGenSubtree, a unique
subject name constraint entry is defined."
::= { cNameConstraintEntry 2 }

cNameConstraintBaseName  OBJECT-TYPE
SYNTAX      SnmpAdminString
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"The distinguished name of the subject that is permitted or
excluded."
::= { cNameConstraintEntry 3 }

-- *****************************************************************
-- CC MIB cRemoteKeyMaterialTable
-- *****************************************************************

cRemoteKeyMaterialTableCount OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"The number of rows in the cRemoteKeyMaterialTable."
::= { cRemoteKeyMaterialInfo 1 }

cRemoteKeyMaterialTableLastChanged OBJECT-TYPE
SYNTAX      TimeStamp
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"The last time any entry in the table was modified, created, or deleted by either SNMP, agent, or other management method (e.g., via an HMI) Managers can use this object to ensure that no changes to configuration of this table have happened since the last time it examined the table. A value of 0 indicates that no entry has been changed since the agent initialized. The value in CC-DEVICE-INFO-MIB cSystemUpTime should be used to populate this column."

::= { cRemoteKeyMaterialInfo 2 }

cRemoteKeyMaterialTable OBJECT-TYPE
SYNTAX       SEQUENCE OF CRemoteKeyMaterialTableEntry
MAX-ACCESS   not-accessible
STATUS       current
DESCRIPTION
"The table containing remote key material information - namely, key material used to help establish the secure connection."

::= { cRemoteKeyMaterialInfo 3 }

cRemoteKeyMaterialTableEntry OBJECT-TYPE
SYNTAX       CRemoteKeyMaterialTableEntry
MAX-ACCESS   not-accessible
STATUS       current
DESCRIPTION
"A row describing the remote key material information used to establish the secure connection."

INDEX  { cRemoteKeyMaterialID }

::= { cRemoteKeyMaterialTable 1 }

CRemoteKeyMaterialTableEntry ::= SEQUENCE {
  cRemoteKeyMaterialID             OCTET STRING,
cRemoteKeyMatFriendlyName        SnmpAdminString,
cRemoteKeyMatSerialNumber        OCTET STRING,
cRemoteKeyMaterialKeyType        OCTET STRING,
cRemoteKeyMatExpirationDate      DateAndTime,
cRemoteKeyMatClassification      BITS
}

cRemoteKeyMaterialID OBJECT-TYPE
SYNTAX       OCTET STRING (SIZE(1..255))
MAX-ACCESS   not-accessible
STATUS       current
DESCRIPTION
"Represents a unique identifier assigned to this key material. This would typically be an identifier inherent to

the key material, such as a serial number or other form of identifier derived from a tag or other key wrapper. This object differs from cRemoteKeyMatFriendlyName which is a user-defined ID.

::= { cRemoteKeyMaterialTableEntry 1 }

cRemoteKeyMatFriendlyName OBJECT-TYPE
SYNTAX       SnmpAdminString
MAX-ACCESS   read-write
STATUS       current
DESCRIPTION   "A human readable label of the key for easier reference. It is used only for helpful or informational purposes."

::= { cRemoteKeyMaterialTableEntry 2 }

cRemoteKeyMatSerialNumber OBJECT-TYPE
SYNTAX       OCTET STRING
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION   "The unique positive integer assigned to the remote key material. Note, this information may not be available in some key material types."

::= { cRemoteKeyMaterialTableEntry 3 }

cRemoteKeyMaterialKeyType OBJECT-TYPE
SYNTAX       OCTET STRING
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION   "This column describes the type of remote key material. Note, this is a free form OCTET STRING column. Implementations are expected to utilize definition of string values that apply to their specific nomenclature supported. If no such nomenclature exists, this column should not be populated or be set to an empty string (i.e., '')."

::= { cRemoteKeyMaterialTableEntry 4 }

cRemoteKeyMatExpirationDate OBJECT-TYPE
SYNTAX       DateAndTime
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION   "The expiration date of the key."

::= { cRemoteKeyMaterialTableEntry 5 }
cRemoteKeyMatClassification OBJECT-TYPE
    SYNTAX        BITS { unclassified(0), restricted(1),
                    confidential(2), secret(3), topSecret(4) }
    MAX-ACCESS    read-only
    STATUS        current
    DESCRIPTION
    "The classification of the key.
    Bit value translation:
    1000 0000 = unclassified
    0100 0000 = restricted
    0010 0000 = confidential
    0001 0000 = secret
    0000 1000 = topSecret

    This column does not exist for devices that do not have
    the concept of classification."
    ::= { cRemoteKeyMaterialTableEntry 6 }

-- *****************************************************************
-- Module Conformance Information
-- *****************************************************************

-- ****************************
cKeyManagementCompliances          OBJECT IDENTIFIER
    ::= { cKeyManagementConformance 1}

cKeyManagementGroups               OBJECT IDENTIFIER
    ::= { cKeyManagementConformance 2}

cKeyManSymKeyCompliance MODULE-COMPLIANCE
    STATUS    current
    DESCRIPTION
    "Compliance levels for symmetric key information."
    MODULE
    MANDATORY-GROUPS { cKeyManSymKeyGroup, cKeyManRemoteKeyGroup }

GROUP cKeyManSymKeyNotifyScalars
    DESCRIPTION
    "This symmetric key notification scalar group is optional
    for implementation."

GROUP cKeyManSymKeyNotifyGroup
    DESCRIPTION
    "This notification group is optional for implementation."
    ::= { cKeyManagementCompliances 1 }

cKeyManAsymKeyCompliance MODULE-COMPLIANCE
    STATUS    current
    DESCRIPTION
"Compliance levels for asymmetric key information."

MODULE
MANDATORY-GROUPS { cKeyManAsymKeyGroup, cKeyManRemoteKeyGroup }

GROUP cKeyManCertSubAltNameGroup
DESCRIPTION
"Certificate Subject Alternative Name group is optional for implementation."

GROUP cKeyManCertPathCtrlsGroup
DESCRIPTION
"Certificate Path Controls group is optional for implementation."

GROUP cKeyManCertPolicyGroup
DESCRIPTION
"Certificate Policy group is optional for implementation."

GROUP cKeyManPolicyMappingGroup
DESCRIPTION
"Policy Mapping group is optional for implementation."

GROUP cKeyManNameConstraintGroup
DESCRIPTION
"Name Constraint group is optional for implementation."

GROUP cKeyManTrustAnchorGroup
DESCRIPTION
"Trust Anchor group is optional for implementation."

GROUP cKeyManAsymKeyNotifyScalars
DESCRIPTION
"This asymmetric key notification scalar group is optional for implementation."

GROUP cKeyManAsymKeyNotifyGroup
DESCRIPTION
"This notification group is optional for implementation."

GROUP cKeyManTrustAnchorNotifyGroup
DESCRIPTION
"This notification group is optional for implementation."

OBJECT cCertPathCtrlsCertificate
MIN-ACCESS not-accessible
DESCRIPTION
"Implementation of this object is optional."
OBJECT cCertPathCtrlsPolicyFlags
MIN-ACCESS not-accessible
DESCRIPTION
"Implementation of this object is optional."

OBJECT cCertPathCtrlsMaxPathLength
MIN-ACCESS not-accessible
DESCRIPTION
"Implementation of this object is optional."
::= { cKeyManagementCompliances  2 }

cKeyManTrustAnchorCompliance MODULE-COMPLIANCE
STATUS current
DESCRIPTION
"Compliance levels for trust anchor information."
MODULE
MANDATORY-GROUPS { cKeyManTrustAnchorGroup }

GROUP cKeyManCertPathCtrlsGroup
DESCRIPTION
"Certificate Path Controls group is optional for implementation."

GROUP cKeyManCertPolicyGroup
DESCRIPTION
"Certificate Policy group is optional for implementation."

GROUP cKeyManPolicyMappingGroup
DESCRIPTION
"Policy Mapping group is optional for implementation."

GROUP cKeyManNameConstraintGroup
DESCRIPTION
"Name Constraint group is optional for implementation."

GROUP cKeyManTrustAnchorNotifyGroup
DESCRIPTION
"This notification group is optional for implementation."

OBJECT cCertPathCtrlsCertificate
MIN-ACCESS not-accessible
DESCRIPTION
"Implementation of this object is optional."

OBJECT cCertPathCtrlsPolicyFlags
MIN-ACCESS not-accessible
DESCRIPTION
"Implementation of this object is optional."
OBJECT cCertPathCtrlsMaxPathLength
MIN-ACCESS not-accessible
DESCRIPTION
"Implementation of this object is optional."
::= { cKeyManagementCompliances  3 }

cKeyManCKLCompliance MODULE-COMPLIANCE
STATUS    current
DESCRIPTION
"Compliance levels for CKL information."
MODULE
MANDATORY-GROUPS { cKeyManCKLGroup }

GROUP cKeyManCKLNotifyGroup
DESCRIPTION
"This notification group is optional for implementation."
::= { cKeyManagementCompliances  4 }

cKeyManCDMStoreCompliance MODULE-COMPLIANCE
STATUS    current
DESCRIPTION
"Compliance levels for CDM Store information."
MODULE
MANDATORY-GROUPS { cKeyManCDMStoreGroup }

GROUP cKeyManCDMStoreNotifyGroup
DESCRIPTION
"This notification group is optional for implementation."
::= { cKeyManagementCompliances  5 }

cKeyManSymKeyGroup OBJECT-GROUP
OBJECTS {
cZeroizeAllKeys,
cZeroizeSymmetricKeyTable,
cSymmetricKeyTableCount,
cSymmetricKeyTableLastChanged,
cSymKeyUsage,
cSymKeyID,
cSymKeyIssuer,
cSymKeyEffectiveDate,
cSymKeyExpirationDate,
cSymKeyExpiryWarning,
cSymKeyNumberOfTransactions,
cSymKeyFriendlyName,
cSymKeyClassification,
cSymKeySource,
cSymKeyRowStatus
}

STATUS current
DESCRIPTION
"This group is composed of objects related to symmetric key information."
::= { cKeyManagementGroups 1 }

cKeyManAsymKeyGroup OBJECT-GROUP
OBJECTS {
cZeroizeAllKeys,
cZeroizeAsymKeyTable,
cAsymKeyTableCount,
cAsymKeyTableLastChanged,
cAsymKeyFingerprint,
cAsymKeyFriendlyName,
cAsymKeySerialNumber,
cAsymKeyIssuer,
cAsymKeySignatureAlgorithm,
cAsymKeyPublicKeyAlgorithm,
cAsymKeyEffectiveDate,
cAsymKeyExpirationDate,
cAsymKeyExpiryWarning,
cAsymKeySubject,
cAsymKeySubjectType,
cAsymKeyUsage,
cAsymKeyClassification,
cAsymKeySource,
cAsymKeyRowStatus,
cAsymKeyVersion,
cAsymKeyRekey,
cAsymKeyType,
cAsymKeyAutoRekeyEnable
}
STATUS current
DESCRIPTION
"This group is composed of objects related to asymmetric key information."
::= { cKeyManagementGroups 2 }

cKeyManCertSubAltNameGroup OBJECT-GROUP
OBJECTS {
cAsymKeySubjectAltName,
cCertSubAltNameTableCount,
cCertSubAltNameTableLastChanged,
cCertSubAltNameType,
cCertSubAltNameValue1,
cCertSubAltNameValue2,
cCertSubAltNameRowStatus
}
STATUS current
DESCRIPTION
"This group is composed of objects related to certificate subject alternative name information."
::= { cKeyManagementGroups 3 }

cKeyManCertPathCtrlsGroup OBJECT-GROUP
OBJECTS {
  cCertPathCtrlsTableCount,
cCertPathCtrlsTableLastChanged,
cCertPathCtrlsCertificate,
cCertPathCtrlsPolicyFlags,
cCertPathCtrlsMaxPathLength
}
STATUS current
DESCRIPTION
"This group is composed of objects related to certificate path controls information."
::= { cKeyManagementGroups 4 }

cKeyManCertPolicyGroup OBJECT-GROUP
OBJECTS {
  cCertPathCtrlsCertPolicies,
cCertPolicyTableCount,
cCertPolicyTableLastChanged,
cCertPolicyIdentifier,
cCertPolicyQualifierID,
cCertPolicyQualifier
}
STATUS current
DESCRIPTION
"This group is composed of objects related to certificate policy information."
::= { cKeyManagementGroups 5 }

cKeyManPolicyMappingGroup OBJECT-GROUP
OBJECTS {
  cCertPathCtrlsPolicyMappings,
cPolicyMappingTableCount,
cPolicyMappingTableLastChanged,
cPolicyMappingSubjectPolicy,
cPolicyMappingIssuerPolicy
}
STATUS current
DESCRIPTION
"This group is composed of objects related to policy mapping information."
::= { cKeyManagementGroups 6 }

cKeyManNameConstraintGroup OBJECT-GROUP
  OBJECTS {
    cCertPathCtrlsNamesPermitted,
    cCertPathCtrlsNamesExcluded,
    cNameConstraintTableCount,
    cNameConstraintTableLastChanged,
    cNameConstraintBaseName
  }
STATUS current
DESCRIPTION
   "This group is composed of objects related to name
    constraint information."
::= { cKeyManagementGroups 7 }

cKeyManTrustAnchorGroup OBJECT-GROUP
  OBJECTS {
    cZeroizeAllKeys,
    cZeroizeTrustAnchorTable,
    cTrustAnchorTableCount,
    cTrustAnchorTableLastChanged,
    cTrustAnchorFingerprint,
    cTrustAnchorFormatType,
    cTrustAnchorName,
    cTrustAnchorUsageType,
    cTrustAnchorKeyIdentifier,
    cTrustAnchorPublicKeyAlgorithm,
    cTrustAnchorContingencyAvail,
    cTrustAnchorRowStatus,
    cTrustAnchorVersion
  }
STATUS current
DESCRIPTION
   "This group is composed of objects related to trust anchor
    information."
::= { cKeyManagementGroups 8 }

cKeyManCKLGroup OBJECT-GROUP
  OBJECTS {
    cCKLTableCount,
    cCKLLastChanged,
    cCKLIndex,
    cCKLIssuer,
    cCKLSerialNumber,
    cCKLIssueDate,
    cCKLNextUpdate,
    cCKLRowStatus,
    cCKLVersion,
    cCKLLastUpdate
  }

cKeyManCDMStoreGroup OBJECT-GROUP
OBJECTS {
    cZeroizeAllKeys,
    cZeroizeCDMStoreTable,
    cCDMStoreTableCount,
    cCDMStoreTableLastChanged,
    cCDMStoreIndex,
    cCDMStoreType,
    cCDMStoreSource,
    cCDMStoreID,
    cCDMStoreFriendlyName,
    cCDMStoreControl,
    cCDMStoreRowStatus
}

DESCRIPTION
"This group is composed of objects related to Crypto
Device Material store information."
::= { cKeyManagementGroups 10 }

cKeyManAsymKeyNotifyScalars OBJECT-GROUP
OBJECTS {
    cKeyMaterialTableOID,
    cKeyMaterialFingerprint,
    cAsymKeyGlobalExpiryWarning
}

DESCRIPTION
"This group is composed of objects related to asymmetric key
notations."
::= { cKeyManagementGroups 11 }
notifications."
 ::= { cKeyManagementGroups 12 }

cKeyManSymKeyNotifyGroup NOTIFICATION-GROUP
 NOTIFICATIONS {
   cKeyMaterialLoadSuccess,
   cKeyMaterialLoadFail,
   cKeyMaterialExpiring,
   cKeyMaterialExpired,
   cKeyMaterialExpirationChanged,
   cKeyMaterialZeroized
 }
 STATUS current
 DESCRIPTION
 "This group is composed of notifications related to
 symmetric key information."
 ::= { cKeyManagementGroups 13 }

cKeyManAsymKeyNotifyGroup NOTIFICATION-GROUP
 NOTIFICATIONS {
   cKeyMaterialLoadSuccess,
   cKeyMaterialLoadFail,
   cKeyMaterialExpiring,
   cKeyMaterialExpired,
   cKeyMaterialExpirationChanged,
   cKeyMaterialZeroized
 }
 STATUS current
 DESCRIPTION
 "This group is composed of notifications related to
 asymmetric key information."
 ::= { cKeyManagementGroups 14 }

cKeyManTrustAnchorNotifyGroup NOTIFICATION-GROUP
 NOTIFICATIONS {
   cTrustAnchorAdded,
   cTrustAnchorUpdated,
   cTrustAnchorRemoved
 }
 STATUS current
 DESCRIPTION
 "This group is composed of notifications related to trust
 anchor information."
 ::= { cKeyManagementGroups 15 }

cKeyManCKLNotifyGroup NOTIFICATION-GROUP
 NOTIFICATIONS {
   cCKLLoadSuccess,
This MIB module makes reference to the following documents: [RFC2578], [RFC2579], [RFC2580], and [RFC3411].

CC-KEY-TRANSFER-PULL-MIB DEFINITIONS ::= BEGIN

IMPORTS
cKeyTransferPull
FROM CC-FEATURE-HIERARCHY-MIB -- FROM Sec 5.2
MODULE-COMPLIANCE, OBJECT-GROUP,
NOTIFICATION-GROUP
FROM SNMPv2-CONF -- FROM RFC 2580
OBJECT-TYPE, Unsigned32, NOTIFICATION-TYPE,
MODULE-IDENTITY
FROM SNMPv2-SMI -- FROM RFC 2578
SnmpAdminString
FROM SNMP-FRAMEWORK-MIB -- FROM RFC 3411
RowStatus, TimeStamp
FROM SNMPv2-TC; -- FROM RFC 2579

ccKeyTransferPullMIB MODULE-IDENTITY
LAST-UPDATED "201609302154Z"
ORGANIZATION "CCMIB CCB"
CONTACT-INFO
"CC MIB Configuration Control Board
  Email: CCMIB.CCB@us.af.mil"
DESCRIPTION
"This MIB defines the CC MIB Key Transfer Pull objects.
Copyright (c) 2019 IETF Trust and the persons
identified as authors of the code. All rights reserved.
Redistribution and use in source and binary forms, with
or without modification, is permitted pursuant to, and
subject to the license terms contained in, the Simplified
BSD License set forth in Section 4.c of the IETF Trust’s
Legal Provisions Relating to IETF Documents
(http://trustee.ietf.org/license-info).
This version of this MIB module is part of RFC xxxx;
see the RFC itself for full legal notices."
-- RFC Ed.: RFC-editor please fill in xxxx.
REVISION "201609302154Z"
DESCRIPTION "CC MIB 1.0.5 FINAL. Published as RFC xxxx."
-- RFC Ed.: RFC-editor please fill in xxxx.
::= { ccKeyTransferPull 1 }

-- ********************************************************************************
-- Key Transfer Pull Information Segments
-- ********************************************************************************
cKeyTransferPullConformance OBJECT IDENTIFIER
::= { ccKeyTransferPullMIB 1 }
cKeyTransferPullScalars OBJECT IDENTIFIER
::= { ccKeyTransferPullMIB 2 }
cKeyTransferPullNotify OBJECT IDENTIFIER
::= { ccKeyTransferPullMIB 3 }
cCDMServerInfo OBJECT IDENTIFIER
cCDMDeliveryInfo OBJECT IDENTIFIER ::= { ccKeyTransferPullMIB 4 }

-- *****************************************************************
-- Key Transfer Pull Scalars
-- *****************************************************************

-- Key Transfer Pull Scalars

--- Key Transfer Pull Scalars

--- *****************************************************************
--- Key Transfer Pull Scalars
--- *****************************************************************

-- Key Transfer Pull Scalars

--- Key Transfer Pull Scalars

--- Key Transfer Pull Scalars

cCDMServerRetryDelay OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"The amount of time to wait after a download attempt to the
Cryptographic Device Material (CDM) server fails before
attempting to retry the operation. Note, this scalar applies
to the download of any type of item from the CDM server
(e.g., CDMs, CDMLs)."
 ::= { ccKeyTransferPullScalars 1 }

-- *****************************************************************
-- Key Transfer Pull Scalars
-- *****************************************************************

-- *****************************************************************
-- Key Transfer Pull Scalars
-- *****************************************************************

-- Key Transfer Pull Scalars

--- Key Transfer Pull Scalars

--- Key Transfer Pull Scalars

--- Key Transfer Pull Scalars

cCDMServerRetryMaxAttempts OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"The amount of retries attempted before the download attempt
to the Cryptographic Device Material (CDM) server is
considered a failure. Note, this scalar applies to the
download of any type of item from the CDM server (e.g.,
CDMs, CDMLs).
 ::= { ccKeyTransferPullScalars 2 }

-- *****************************************************************
-- Key Transfer Pull Scalars
-- *****************************************************************

-- Key Transfer Pull Scalars

--- Key Transfer Pull Scalars

--- Key Transfer Pull Scalars

--- Key Transfer Pull Scalars

cCDMPullRetrievalPriorities OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"An indication of which cryptographic device materials
(CDMs) to retrieve based on this value and a configured
cCDMDeliveryPriority in a cCDMDeliveryTable entry. This
value identifies an upper bound. A value of '5' for example,
implies that only cCDMDeliveryTable entries with a
cCDMDeliveryPriority value of '5' or less can be acted upon
(i.e., retrieved).

Different types of ECUs may have different values for this
scalar. Bandwidth-limited ECUs, for example, may configure
lower values for only retrieving high-priority CDMs.

A value of 0, also a default value for this scalar, indicates that all cCDMDeliveryTable entries can be acted upon regardless of the configured cCDMDeliveryPriority value.

DEFVAL {0}
::= { cKeyTransferPullScalars 3 }

cCDMLDeliveryRequest OBJECT-TYPE
SYNTAX INTEGER { readyForDownload(1), downloadAndParse(2), discard(3) }
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"This scalar controls the server’s CDML download process - server information is stored in the cCDMServerTable. When read, it will return ‘readyForDownload’ if the last action succeeded. If the last action is in progress or failed, it will return the last requested action.

The values which may be set depend on the current value of this object and the cCDMLDeliveryStatus object.

In order to initiate a new download, this object must contain the value ‘readyForDownload’, and the cCDMLDeliveryStatus must contain the value ‘complete’. At which point, setting this object to to ‘downloadAndParse’ initiates the CDML download process. Note, the cCDMLDeliveryStatus should transition to ‘inProgress’ at the device begins the CDML download process from the server(s) and URI(s) listed in the cCDMLServerTable (as ordered by the cCDMLServerPriority index).

If the CDML download fails, the next highest priority URI will be tried, and so on.

While a CDML download is in progress, or if the CDML download fails for all possible servers and URIs (indicated by a cCDMLDeliveryStatus value of ‘downloadFailed’), this object will return an inconsistentValue error for any new value except ‘discard’ (which will cancel the current download).

If the CDML download succeeded, the cCMDLDeliveryStatus value remains inProgress and the device attempts to parse the download immediately. During the parsing of the CDML, all new values will return inconsistentValue error (i.e., the parse process can not be aborted). If the parse fails, the cCDMLDeliveryStatus will transition to ‘parseFailed’,
and this object must be set to ‘discard’ before a new CDML download is attempted."
::= { cKeyTransferPullScalars 4 }

ccCDMLDeliveryStatus OBJECT-TYPE
SYNTAX INTEGER { complete(1), inProgress(2),
downloadFailed(3),
parseFailed(4) }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"This indicates the current state of a CDML download.
‘complete’ indicates that the last requested
cCDMLDeliveryRequest action was successful.
‘inProgress’ indicates that a CDML download or CDML parse is underway.
‘downloadFailed’ indicates that the last attempted CDML
download failed.
‘parseFailed’ indicates that the last attempted CDML parse failed.
The relationship between this object and
cCDMLDeliveryRequest is detailed in the following table. The table indicates values of cCDMLDeliveryRequest that are allowed depending on the current value of this object.

<table>
<thead>
<tr>
<th>cCDMLDeliveryRequest</th>
<th>cCDMLDeliveryStatus</th>
</tr>
</thead>
<tbody>
<tr>
<td>--------------------------------+---------------------</td>
<td></td>
</tr>
<tr>
<td>readyForDownload</td>
<td>allowed error error error error !</td>
</tr>
<tr>
<td>downloadAndParse</td>
<td>allowed error error error error !</td>
</tr>
<tr>
<td>discard</td>
<td>error allowed allowed allowed allowed !</td>
</tr>
</tbody>
</table>

As described cCDMLDeliveryRequest description, an inconsistentValue error is returned."
DEFVAL { complete }
::= { cKeyTransferPullScalars 5 }

-- **********************************************************

-- Key Transfer Pull Notifications
-- ************************************************************************
cCDMLPullReceiveSuccess  NOTIFICATION-TYPE
OBJECTS    { cCDMServerURI }
STATUS    current
DESCRIPTION
"An attempt to receive a cryptographic device material
list (CDML) succeeded. The CDM server URI is provided with
this notification."
::= { cKeyTransferPullNotify 1 }
cCDMLPullReceiveFailed   NOTIFICATION-TYPE
OBJECTS    {
               cCDMServerURI,
               cCDMLDeliveryStatus
               }
STATUS    current
DESCRIPTION
"An attempt to receive a cryptographic device material
list (CDML) has failed. The CDM server URI and CDML Delivery
Status are provided with this notification. Note, the
expected values for the CDML Delivery Status are:
'downloadFailed' and 'parseFailed'."
::= { cKeyTransferPullNotify 2 }
cCDMPullReceiveSuccess   NOTIFICATION-TYPE
OBJECTS    {
               cCDMType,
               cCDMURI
               }
STATUS    current
DESCRIPTION
"An attempt to receive a cryptographic device material (CDM)
has succeeded. The CDM Type and CDM URI are provided with
this notification."
::= { cKeyTransferPullNotify 3 }
cCDMPullReceiveFailed    NOTIFICATION-TYPE
OBJECTS    {
               cCDMType,
               cCDMURI
               }
STATUS    current
DESCRIPTION
"An attempt to receive a cryptographic device material (CDM)
has failed. The CDM Type and CDM URI are provided with this
notification."
::= { cKeyTransferPullNotify 4 }

-- *****************************************************************
-- CC MIB cCDMServerTable
-- *****************************************************************


cCDMServerTableCount OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The number of rows in the cCDMServerTable."
::= { cCDMServerInfo 1 }

cCDMServerTableLastChanged OBJECT-TYPE
SYNTAX TimeStamp
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The last time any entry in the table was modified, created, or deleted by either SNMP, agent, or other management method (e.g., via an HMI). Managers can use this object to ensure that no changes to configuration of this table have happened since the last time it examined the table. A value of 0 indicates that no entry has been changed since the agent initialized. The value in CC-DEVICE-INFO-MIB cSystemUpTime should be used to populate this column."
::= { cCDMServerInfo 2 }

cCDMServerTable OBJECT-TYPE
SYNTAX SEQUENCE OF CCDMServerEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"The table containing a list of servers that will be queried for available cryptographic device materials (CDMs), such as keys and firmware packages. This table is also used to obtain the cryptographic device material list (CDML), which is a list detailing available CDMs and their associated location for obtainment."
::= { cCDMServerInfo 3 }

cCDMServerEntry OBJECT-TYPE
SYNTAX CCDMServerEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"A row containing information about a server that has
available CDMLs/CDMs for download."
INDEX { cCDMServerPriority }
 ::= { cCDMServerTable 1 }

CCDMServerEntry ::= SEQUENCE {
  cCDMServerPriority          Unsigned32,
  cCDMServerURI               OCTET STRING,
  cCDMServerAdditionalInfo    SnmpAdminString,
  cCDMServerRowStatus         RowStatus
}

CCDMServerPriority  OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
"A unique numeric index that identifies a server that has
available CDMLs/CDMs for download. This index also provides
server prioritization functionality - lower values have a
higher priority. For example, the server with the lowest
value will be the first server for CDML/CDM downloads. In
the event of failure, the next lowest value server will be
tried, and so on.

This column is the sole index to the cCDMServerTable."
 ::= ( cCDMServerEntry 1 )

CCDMServerURI  OBJECT-TYPE
SYNTAX      OCTET STRING (SIZE(1..255))
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
"The location of the server that has available CDMLs/CDMs
for download. The value in this column is represented as a
URI.

Note, download of a CDML will typically result in the
population of new CDM entries in the cCDMDeliveryTable."
 ::= ( cCDMServerEntry 2 )

CCDMServerAdditionalInfo  OBJECT-TYPE
SYNTAX      SnmpAdminString
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
"Additional information about the CDM server. This
information is manually configured by the manager both at or
after row creation."
::= { cCDMServerEntry 3 }

cCDMServerRowStatus OBJECT-TYPE
SYNTAX      RowStatus
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
 "The status of the row, by which new entries may be created or old entries deleted from this table.

Entries created within this table may not become active unless all read-create columns in this column have valid values, as detailed by each individual column’s description.

At a minimum, implementations must support createAndGo, active, and destroy management functions. Support for createAndWait, notInService, and notReady management functions is optional."
::= { cCDMServerEntry 4 }

-- *****************************************************************
-- CC MIB cCDMDeliveryTable
-- *****************************************************************

ccMDeliveryTableCount  OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
 "The number of rows in the cCDMDeliveryTable."
::= { cCDMDeliveryInfo 1 }

ccMDeliveryTableLastChanged  OBJECT-TYPE
SYNTAX      TimeStamp
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
 "The last time any entry in the table was modified, created, or deleted by either SNMP, agent, or other management method (e.g., via an HMI). Managers can use this object to ensure that no changes to configuration of this table have happened since the last time it examined the table. A value of 0 indicates that no entry has been changed since the agent initialized. The value in CC-DEVICE-INFO-MIB cSystemUpTime should be used to populate this column."
::= { cCDMDeliveryInfo 2 }

ccMDeliveryTable OBJECT-TYPE
SYNTAX      SEQUENCE OF CCDMDeliveryEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
"The table storing information about cryptographic device
materials (CDMs) that are ready/available for retrieval.
Entries in this table are typically automatically configured
by the device after a server query. Entries can also be
manually configured by a manager if the location of the CDM
is predetermined."
::= { cCDMDeliveryInfo 3 }

CCDMDeliveryEntry OBJECT-TYPE
SYNTAX      CCDMDeliveryEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
"A row containing information about a specific cryptographic
device material (CDM) available for download."
INDEX      { cCDMType, cCDMURI }
::= { cCDMDeliveryTable 1 }

CCDMDeliveryEntry ::= SEQUENCE {
  cCDMType                INTEGER,
  cCDMURI                 OCTET STRING,
  cCDMPackageSize         Unsigned32,
  cCDMAdditionalInfo      SnmpAdminString,
  cCDMLastDownloadDate    OCTET STRING,
  cCDMDeliveryPriority    Unsigned32,
  cCDMDeliveryRequest     INTEGER,
  cCDMDeliveryStatus      INTEGER,
  cCDMDeliveryRowStatus   RowStatus
}

cCDMType  OBJECT-TYPE
SYNTAX      INTEGER { notification(1), symmetricKey(2),
               asymmetricKey(3), certificate(4),
               cklOrCrl(5), firmware(6) }  
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"The type of the cryptographic device material (CDM) that
can be retrieved from a CDM server:

[notification] = CDM is a notification providing
status/information for a particular
(other) CDM

[symmetricKey] = CDM is a symmetric key
[asymmetricKey] = CDM is a non-certificate asymmetric key
[certificate] = CDM is a certificate
[cklOrCrl] = CDM is a compromised key list or certificate revocation list
[firmware] = CDM is a firmware package

::= { cCDMDeliveryEntry 1 }

cCDMURI OBJECT-TYPE
SYNTAX          OCTET STRING (SIZE(1..255))
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
"The location of the cryptographic device material (CDM), represented in a URI format. Because of its type, the associated URI of the CDM Server can easily be derived.

This column is typically populated by an agent upon querying a CDM Server (e.g., downloading and parsing a cryptographic device material list (CDML) from a CDM Server (entry in the cCDMServerTable)). However, a manager can also configure an entry in this table with predetermined knowledge of the CDM location."

::= { cCDMDeliveryEntry 2 }

cCDMPackageSize OBJECT-TYPE
SYNTAX          Unsigned32
UNITS           "bytes"
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
"The package size, in bytes, of the cryptographic device material (CDM). This information is retrieved from a cryptographic device material list (CDML) or a server’s product availability response following a query. This column does not apply to notifications found in CDMLs."

::= { cCDMDeliveryEntry 3 }

cCDMAdditionalInfo OBJECT-TYPE
SYNTAX          SnmpAdminString
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
"Additional information about the cryptographic device material (CDM). This information can be retrieved from the downloaded cryptographic device material list (CDML) or manually configured by the manager both at or after row creation."

::= { cCDMDeliveryEntry 4 }
cCDMLastDownloadDate  OBJECT-TYPE
SYNTAX      OCTET STRING (SIZE(14))
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
   "This is a 14 character field that will be populated with
   the following values depending on the state of the download
   and the CDM type.

   1. The date and time (expressed as Generalized Time) when
   the device last successfully downloaded the CDM from the
   CDM Server. The format follows: ‘yyyyymmddhhmmss’ where
   ‘yyyy’ - year
   ‘mm’ - month (first ‘mm’s from left to right)
   ‘dd’ - day
   ‘hh’ - hour
   ‘mm’ - minutes (second ‘mm’s from left to right)
   ‘ss’ - seconds

   2. All zero characters for the following cases.
   a. No indication that device has successfully downloaded
      the CDM.
   b. The cCDMType is a notification."
 ::= { cCDMDeliveryEntry 5 }

cCDMDeliveryPriority  OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
   "A configurable priority value on the cryptographic device
   material (CDM). This column is a means to allow certain key
   products to be downloaded before others. Lower values have a
   higher priority (e.g., a value of 1 will be processed before
   a value of 2)."
 ::= { cCDMDeliveryEntry 6 }

cCDMDeliveryRequest  OBJECT-TYPE
SYNTAX     INTEGER { downloadAndInstall(1), downloadAndStore(2),
               discard(3) }
MAX-ACCESS read-create
STATUS      current
DESCRIPTION
   "This object signals the local device to perform actions on
   the available cryptographic device materials (CDMs) from a
   CDM server. The following types of actions are supported:

   [downloadAndInstall] = Initiates a download of a CDM. After
a successful download, the CDM will be installed for local consumption and an entry is to be configured in the appropriate MIB table based on cCDMType:

<table>
<thead>
<tr>
<th>cCDMType</th>
<th>MIB Table Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) notification</td>
<td>N/A</td>
</tr>
<tr>
<td>(2) symmetricKey</td>
<td>cSymmetricKeyTable</td>
</tr>
<tr>
<td>(3) asymmetricKey</td>
<td>cAsymKeyTable</td>
</tr>
<tr>
<td>(4) certificate</td>
<td>cAsymKeyTable</td>
</tr>
<tr>
<td>(5) cklOrCrl</td>
<td>cCKLTable</td>
</tr>
<tr>
<td>(6) firmware</td>
<td>cFirmwareInformationTable</td>
</tr>
</tbody>
</table>

[downloadAndStore] = Initiates a download of the CDM. After a successful download, an entry is created in the cCDMStoreTable to store the CDM.

[discard] = Stops the current CDM delivery request and discards the CDM if potentially downloaded; this reverts the current value of the cCDMDeliveryStatus to ‘complete’. If entries are created in the aforementioned tables for the install and store operations, these newly configured entries will be removed.

The enumeration value of ‘downloadAndStore’ does not apply when cCDMType is set to ‘notification’. ‘downloadAndInstall’ is used for a cCDMType of ‘notification’.

If this column is configured to any value except ‘discard’ while the value of cCDMDeliveryStatus is any value except ‘complete’, the SNMP set operation must result in an inconsistentValue exception. The same applies if ‘discard’ is configured while the value cCDMDeliveryStatus is ‘complete’.

::= { cCDMDeliveryEntry 7 }

cCDMDeliveryStatus OBJECT-TYPE
SYNTAX INTEGER { complete(1), inProgress(2),
downloadFailed(3), installFailed(4),
storeFailed(5) }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The status of the cryptographic device material (CDM) delivery operation. The following status values are supported:

[complete] = The default state where the local device is
ready to start a delivery request for the CDM. Between requests this state can only be reached after successful operations or if cCDMDeliveryRequest is set to ‘discard’ during an operation.

[inProgress] = This state is reached when the device is either currently performing a download of the CDM or configuring appropriate MIB tables conveying installation or storage of key material.

[downloadFailed] = This state is reached after a failure occurs during a download of a CDM when cCDMDeliveryRequest was configured to either ‘downloadAndStore’ or ‘downloadAndInstall’.

[installFailed] = This state is reached after a failure occurs during the install of the downloaded CDM when cCDMDeliveryRequest was configured to ‘downloadAndInstall’.

[storeFailed] = This state is reached after a failure occurs during the store of the downloaded CDM when cCDMDeliveryRequest was configured to ‘downloadAndStore’.

::= { cCDMDeliveryEntry 8 }

cCDMDeliveryRowStatus OBJECT-TYPE
SYNTAX RowStatus
MAX-ACCESS read-create
STATUS current
DESCRIPTION "The status of the row, by which new entries may be created or old entries deleted from this table.

Entries created within this table may not become active unless all read-create columns in this column have valid values, as detailed by each individual column’s description.

At a minimum, implementations must support createAndGo, active, and destroy management functions. Support for createAndWait, notInService, and notReady management functions is optional."
::= { cCDMDeliveryEntry 9 }

-- *****************************************************************
-- Module Conformance Information
-- *****************************************************************

cKeyTransferPullCompliances OBJECT IDENTIFIER
::= { cKeyTransferPullConformance 1}
cKeyTransferPullGroups OBJECT IDENTIFIER ::= { cKeyTransferPullConformance 2}

cKeyTransferPullCompliance MODULE-COMPLIANCE
  STATUS current
  DESCRIPTION "Compliance levels for key transfer pull information."
  MODULE
  MANDATORY-GROUPS {
    cKeyTransferPullServerGroup,
    cKeyTransferPullDeliveryGroup
  }

GROUP cKeyTransferPullDeliveryNotifyGroup
  DESCRIPTION "This notification group is optional for implementation."

OBJECT cCDMDeliveryRequest
  SYNTAX INTEGER { downloadAndInstall(1), discard(3) }
  DESCRIPTION "Implementation of this enumeration value(s) is mandatory -
  enumeration values not listed here are optional."

OBJECT cCDMDeliveryStatus
  SYNTAX INTEGER { complete(1), inProgress(2), downloadFailed(3),
    installFailed(4) }
  DESCRIPTION "Implementation of this enumeration value(s) is mandatory -
  enumeration values not listed here are optional."
  ::= { cKeyTransferPullCompliances 1 }

cKeyTransferPullServerGroup OBJECT-GROUP
  OBJECTS {
    cCDMServerRetryDelay,
    cCDMServerRetryMaxAttempts,
    cCDMServerTableCount,
    cCDMServerTableLastChanged,
    cCDMServerURI,
    cCDMServerAdditionalInfo,
    cCDMServerRowStatus
  }
  STATUS current
  DESCRIPTION "This group is composed of objects related to server
  information."
  ::= { cKeyTransferPullGroups 1 }

cKeyTransferPullDeliveryGroup OBJECT-GROUP
OBJECTS {
  cCDMPullRetrievalPriorities,
  cCDSLDeliveryRequest,
  cCDSLDeliveryStatus,
  cCDMDeliveryTableCount,
  cCDMDeliveryTableLastChanged,
  cCDMDeliveryTableLastChanged,
  cCDMType,
  cCDMURI,
  cCDMPackageSize,
  cCDMAdditionalInfo,
  cCDMLastDownloadDate,
  cCDMDeliveryPriority,
  cCDMDeliveryRequest,
  cCDMDeliveryStatus,
  cCDMDeliveryRowStatus
}

STATUS current
DESCRIPTION
"This group is composed of objects related to delivery information."
::= { cKeyTransferPullGroups 2 }

cKeyTransferPullDeliveryNotifyGroup NOTIFICATION-GROUP
NOTIFICATIONS {
  cCDSLPullReceiveSuccess,
  cCDSLPullReceiveFailed,
  cCDMPullReceiveSuccess,
  cCDMPullReceiveFailed
}

STATUS current
DESCRIPTION
"This group is composed of notifications related to delivery information."
::= { cKeyTransferPullGroups 3 }

END

5.6. Key Transfer Push

This MIB module makes reference to following documents: [RFC2578],
[RFC2579], [RFC2580], and [RFC3411].

CC-KEY-TRANSFER-PUSH-MIB DEFINITIONS ::= BEGIN

IMPORTS
  ccKeyTransferPush
  FROM CC-FEATURE-HIERARCHY-MIB

END
OBJECT-TYPE, Unsigned32, NOTIFICATION-TYPE,
MODULE-IDENTITY
FROM SNMPv2-SMI
FROM SNMP-FRAMEWORK-MIB
RowPointer, RowStatus, DateAndTime,
TimeStamp
FROM SNMPv2-TC
FROM SNMPv2-SM
SnmpAdminString
FROM SNMP-FRAMEWORK-MIB
ORGANIZATION  "CCMIB CCB"
CONTACT-INFO
"CC MIB Configuration Control Board
Email: CCMIB.CCB@us.af.mil"
DESCRIPTION
"This MIB defines the CC MIB Key Transfer Push object.

Copyright (c) 2019 IETF Trust and the persons
identified as authors of the code. All rights reserved.

Redistribution and use in source and binary forms, with
or without modification, is permitted pursuant to, and
subject to the license terms contained in, the Simplified
BSD License set forth in Section 4.c of the IETF Trust's
Legal Provisions Relating to IETF Documents
(http://trustee.ietf.org/license-info).

This version of this MIB module is part of RFC xxxx;
see the RFC itself for full legal notices."
-- RFC Ed.: RFC-editor please fill in xxxx.
REVISION      "201609302154Z"
DESCRIPTION   "CC MIB 1.0.5 FINAL. Published as RFC xxxx."
-- RFC Ed.: RFC-editor please fill in xxxx.
::= { ccKeyTransferPushMIB 1 }

-- ******************************************************
-- Key Transfer Push Information Segments
-- ******************************************************
cCDMPushDestInfo OBJECT IDENTIFIER
::= { ccKeyTransferPushMIB 1 }
cCDMTransferPkgInfo OBJECT IDENTIFIER
::= { ccKeyTransferPushMIB 2 }
cCDMPushSrcInfo OBJECT IDENTIFIER
cKeyTransferPushMIB ::= { ccKeyTransferPushMIB 3 }
cKeyTransferPushScalars OBJECT IDENTIFIER ::= { ccKeyTransferPushMIB 4 }
cKeyTransferPushNotify OBJECT IDENTIFIER ::= { ccKeyTransferPushMIB 5 }
cKeyTransferPushConformance OBJECT IDENTIFIER ::= { ccKeyTransferPushMIB 6 }

-- *****************************************************************
-- Key Transfer Push Scalars
-- *****************************************************************
cCDMTransferDelay OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-write
STATUS current
DESCRIPTION "The number of seconds to wait after a Cryptographic Device Material (CDM) transfer attempt initiated by the sender fails before attempting to retry the operation."
::= { cKeyTransferPushScalars 1 }
cCDMTransferMaxAttempts OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-write
STATUS current
DESCRIPTION "The amount of retries attempted before giving up on a device due to consecutive Cryptographic Device Material (CDM) transfer failures."
::= { cKeyTransferPushScalars 2 }

-- *****************************************************************
-- Key Transfer Push Notifications
-- *****************************************************************
cCDMPushSendSuccess NOTIFICATION-TYPE
OBJECTS { cCDMPushDestAddressLocationType, cCDMPushDestAddressLocation, cCDMPushDestTransferType, cCDMPushDestPackageSelection }
STATUS current
DESCRIPTION "An attempt to send CDM, identified by CDM push transfer information (cCDMPushDestTable row data), has succeeded."
::= { cKeyTransferPushNotify 1 }
cCDMPushReceiveSuccess NOTIFICATION-TYPE
OBJECTS {
    cCDMPushSrcAddrLocationType,
    cCDMPushSrcAddrLocation,
    cCDMPushSrcTransferType
}
STATUS current
DESCRIPTION
"An attempt to receive key material, identified by CDM push
transfer information (cCDMPushSrcTable row data), has
succeeded."

::= { cKeyTransferPushNotify 2 }

-- *****************************************************************
-- CC MIB cCDMPushDestTable
-- *****************************************************************

-- **********************
-- CC MIB cCDMPushDestTable
-- **********************

cCDMPushDestTableCount OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"An attempt to send key material, identified by the
Recipient Address and Transfer Type, has failed."

::= { cKeyTransferPushNotify 4 }
"The number of rows in the cCDMPushDestTable."
::= { cCDMPushDestInfo 1 }

cCDMPushDestTableLastChanged  OBJECT-TYPE
SYNTAX      TimeStamp
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"The last time any entry in the table was modified, created, or deleted by either SNMP, agent, or other management method (e.g., via an HMI). Managers can use this object to ensure that no changes to configuration of this table have happened since the last time it examined the table. A value of 0 indicates that no entry has been changed since the agent initialized. The value in CC-DEVICE-INFO-MIB cSystemUpTime should be used to populate this column."
::= { cCDMPushDestInfo 2 }

cCDMPushDestTable  OBJECT-TYPE
SYNTAX      SEQUENCE OF CCDMPushDestEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
"The table that provides the necessary information a sender needs to initiate a Cryptographic Device Material (CDM) send to a receiving device."
::= { cCDMPushDestInfo 3 }

cCDMPushDestEntry  OBJECT-TYPE
SYNTAX      CCDMPushDestEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
"A row containing information for a Cryptographic Device Material (CDM) transfer to a receiving device."
INDEX      { cCDMPushDestIndex }
::= { cCDMPushDestTable 1 }

CCDMPushDestEntry  ::= SEQUENCE {
  cCDMPushDestIndex               Unsigned32,
  cCDMPushDestTransferType        INTEGER,
  cCDMPushDestAddressLocationType INTEGER,
  cCDMPushDestAddressLocation     OCTET STRING,
  cCDMPushDestTransferTime        DateAndTime,
  cCDMPushDestPackageSelection    SnmpAdminString,
  cCDMPushDestRowStatus           RowStatus
}
cCDMPushDestIndex OBJECT-TYPE
SYNTAX   Unsigned32
MAX-ACCESS not-accessible
STATUS   current
DESCRIPTION
   "A numeric index that identifies a unique location in this
table."
::= { cCDMPushDestEntry 1 }

cCDMPushDestTransferType OBJECT-TYPE
SYNTAX   SnmpAdminString (SIZE(1..32))
MAX-ACCESS read-create
STATUS   current
DESCRIPTION
   "The transfer mechanism or protocol used by the sender to
execute the Cryptographic Device Material (CDM) transfer."
::= { cCDMPushDestEntry 2 }

cCDMPushDestAddressLocationType OBJECT-TYPE
SYNTAX   INTEGER { ipv4(1), ipv6(2), uri(3), other(4) }
MAX-ACCESS read-create
STATUS   current
DESCRIPTION
   "Enumeration indicating the type of address location."
::= { cCDMPushDestEntry 3 }

cCDMPushDestAddressLocation OBJECT-TYPE
SYNTAX   OCTET STRING
MAX-ACCESS read-create
STATUS   current
DESCRIPTION
   "Location of the receiver. The syntax allows a URI or an IP
address to be configured."
::= { cCDMPushDestEntry 4 }

cCDMPushDestTransferTime OBJECT-TYPE
SYNTAX   DateAndTime
MAX-ACCESS read-create
STATUS   current
DESCRIPTION
   "A valid date and time value populated in this object will
automatically initiate the transfer at the value specified.

To initiate an immediate transfer the following
configuration is used: '0' for the year field, '1' for the
month field, '1' for the day field, '-' for the direction
from UTC field, and '0' for all other fields. This
configuration is displayed as '0-1-1,00:00:00.0,-0:0'. Note
that if the timezone fields are not used then the displayed value is as follows: '0-1-1,00:00:00.0'. The timezone fields are the direction from UTC, hours from UTC, and minutes from UTC.

::= { cCDMPushDestEntry 5 }

cCDMPushDestPackageSelection OBJECT-TYPE
SYNTAX      SnmpAdminString
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
"A reference string that points to the key material(s) to transfer. This column may reference one entry (e.g., an entry in the cCDMStoreTable) or multiple entries (e.g., multiple entries in the cCDMTransferPkgTable). This object defines all the items in the package that will be sent."

::= { cCDMPushDestEntry 6 }

cCDMPushDestRowStatus  OBJECT-TYPE
SYNTAX      RowStatus
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
"The status of the row, by which new entries may be created or old entries deleted from this table.

Entries created within this table may not become active unless all read-create columns in this column have valid values, as detailed by each individual column’s description.

At a minimum, implementations must support createAndGo, active, and destroy management functions. Support for createAndWait, notInService, and notReady management functions is optional."

::= { cCDMPushDestEntry 7 }

-- *****************************************************************
-- CC MIB cCDMTransferPkgTable
-- *****************************************************************

cCDMTransferPkgTableCount  OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"The number of rows in the cCDMTransferPkgTable."

::= { cCDMTransferPkgInfo 1 }
cCDMTransferPkgTableLastChanged  OBJECT-TYPE
SYNTAX      TimeStamp
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"The last time any entry in the table was modified, created, or deleted by either SNMP, agent, or other management method (e.g., via an HMI). Managers can use this object to ensure that no changes to configuration of this table have happened since the last time it examined the table. A value of 0 indicates that no entry has been changed since the agent initialized. The value in CC-DEVICE-INFO-MIB cSystemUpTime should be used to populate this column."
::= { cCDMTransferPkgInfo 2 }

cCDMTransferPkgTable  OBJECT-TYPE
SYNTAX      SEQUENCE OF CCDMTransferPkgEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
"The table for configuring single or multiple Cryptographic Device Material (CDM) in a package that can be transferred on a send operation. Entries in this table are referenced by the cCDMPushDestPackageSelection column."
::= { cCDMTransferPkgInfo 3 }

cCDMTransferPkgEntry  OBJECT-TYPE
SYNTAX      CCDMTransferPkgEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
"A row containing information about a package used on a send operation."
INDEX      { cCDMTransferPkgLabel, cCDMTransferPkgIndex }
::= { cCDMTransferPkgTable 1 }

CCDMTransferPkgEntry ::= SEQUENCE {
  cCDMTransferPkgLabel            SnmpAdminString,
  cCDMTransferPkgIndex            Unsigned32,
  cCDMTransferPkgLocatorRowPtr    RowPointer,
  cCDMTransferPkgRowStatus        RowStatus
}

cCDMTransferPkgLabel  OBJECT-TYPE
SYNTAX      SnmpAdminString
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
"An administrative name that identifies a package within this table. cCDMTransferPkgLabel and cCDMTransferPkgIndex serve as indexes of this table."
::= { cCDMTransferPkgEntry 1 }

cCDMTransferPkgIndex OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"An administrative way of creating a unique row within this table. This value shows the position of a given item within this package designated by cCDMTransferPkgLabel. cCDMTransferPkgLabel and cCDMTransferPkgIndex serve as indexes of this table."
::= { cCDMTransferPkgEntry 2 }

cCDMTransferPkgLocatorRowPtr OBJECT-TYPE
SYNTAX RowPointer
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"A RowPointer that points to a unique entry in the table containing the necessary Cryptographic Device Material (CDM) for transfer. For example, referencing a key in the cSymmetricKeyTable, the value in this column contains the pointer to the appropriate row in the cSymmetricKeyTable."
::= { cCDMTransferPkgEntry 3 }

cCDMTransferPkgRowStatus OBJECT-TYPE
SYNTAX RowStatus
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The status of the row, by which new entries may be created or old entries deleted from this table. Entries created within this table may not become active unless all read-create columns in this column have valid values, as detailed by each individual column’s description. At a minimum, implementations must support createAndGo, active, and destroy management functions. Support for createAndWait, notInService, and notReady management functions is optional."
::= { cCDMTransferPkgEntry 4 }

-- *****************************************************************
-- CC MIB cCDMPushSrcTable
-- *****************************************************************

cCDMPushSrcTableCount OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The number of rows in the cCDMPushSrcTable."
::= { cCDMPushSrcInfo 1 }

cCDMPushSrcTableLastChanged OBJECT-TYPE
SYNTAX TimeStamp
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The last time any entry in the table was modified, created,
or deleted by either SNMP, agent, or other management method
(e.g., via an HMI). Managers can use this object to ensure
that no changes to configuration of this table have happened
since the last time it examined the table. A value of 0
indicates that no entry has been changed since the agent
initialized. The value in CC-DEVICE-INFO-MIB cSystemUpTime
should be used to populate this column."
::= { cCDMPushSrcInfo 2 }

cCDMPushSrcTable OBJECT-TYPE
SYNTAX SEQUENCE OF CCDMPushSrcEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"This table provides the list of authorized senders that
this receiving device will accept Cryptographic Device
Material (CDM) transfers from. Servers for the
cCDMServerTable are not listed in this table since this
table is specific for the Push Model."
::= { cCDMPushSrcInfo 3 }

cCDMPushSrcEntry OBJECT-TYPE
SYNTAX CCDMPushSrcEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"A row containing information about an authorized sender
that this receiving device will accept."
INDEX { cCDMPushSrcSenderName, cCDMPushSrcTransferType }
::= { cCDMPushSrcTable 1 }
CCDMPushSrcEntry ::= SEQUENCE {
  cCDMPushSrcSenderName SnmpAdminString,
  cCDMPushSrcTransferType SnmpAdminString,
  cCDMPushSrcAddrLocationType INTEGER,
  cCDMPushSrcAddrLocation OCTET STRING,
  cCDMPushSrcRowStatus RowStatus
}


cCDMPushSrcSenderName OBJECT-TYPE
SYNTAX SnmpAdminString
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION "An administrative string for an authorized sender. cCDMPushSrcSenderName and cCDMPushSrcTransferType serve as indexes of this table."
::= { cCDMPushSrcEntry 1 }

cCDMPushSrcTransferType OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE(1..32))
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Analogous to cCDMPushDestTransferType. The transfer mechanism or protocol used by the receiver to receive the Cryptographic Device Material (CDM) transfer. cCDMPushSrcSenderName and cCDMPushSrcTransferType serve as indexes of this table."
::= { cCDMPushSrcEntry 2 }

cCDMPushSrcAddrLocationType OBJECT-TYPE
SYNTAX INTEGER { ipv4(1), ipv6(2), uri(3), other(4) }
MAX-ACCESS read-create
STATUS current
DESCRIPTION "Enumeration indicating the type of address location (values: ipv4, ipv6 or uri)."
::= { cCDMPushSrcEntry 3 }

cCDMPushSrcAddrLocation OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-create
STATUS current
DESCRIPTION "Location of the authorized sender."
::= { cCDMPushSrcEntry 4 }
cCDMPushSrcRowStatus  OBJECT-TYPE
SYNTAX      RowStatus
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
 "The status of the row, by which new entries may be created
or old entries deleted from this table.

Entries created within this table may not become active
unless all read-create columns in this column have valid
values, as detailed by each individual column’s description.

At a minimum, implementations must support createAndGo,
active, and destroy management functions. Support for
createAndWait, notInService, and notReady management
functions is optional."
 ::= { cCDMPushSrcEntry 5 }

-- *****************************************************************
-- Module Conformance Information
-- *****************************************************************
cKeyTransferPushCompliances  OBJECT IDENTIFIER
 ::= { cKeyTransferPushConformance 1}
cKeyTransferPushGroups    OBJECT IDENTIFIER
 ::= { cKeyTransferPushConformance 2}
cKeyTransferPushSenderCompliance  MODULE-COMPLIANCE
 STATUS    current
DESCRIPTION
 "Compliance levels for sender information."
MODULE
MANDATORY-GROUPS { cKeyTransferPushSenderGroup }
GROUP cKeyTransferPushSenderNotifyGroup
DESCRIPTION
 "This notification group is optional for implementation."
OBJECT cCDMTransferDelay
MIN-ACCESS not-accessible
DESCRIPTION
 "Implementation of this object is optional."
OBJECT cCDMTransferMaxAttempts
MIN-ACCESS not-accessible
DESCRIPTION
 "Implementation of this object is optional."
::= { cKeyTransferPushCompliances 1 }

cKeyTransferPushReceiverCompliance MODULE-COMPLIANCE
STATUS current
DESCRIPTION "Compliance levels for receiver information."
MODULE
MANDATORY-GROUPS { cKeyTransferPushReceiverGroup }

GROUP cKeyTransferPushReceiverNotifyGroup
DESCRIPTION "This notification group is optional for implementation."
::= { cKeyTransferPushCompliances 2 }

cKeyTransferPushSenderGroup OBJECT-GROUP OBJECTS {
    cCDMTransferDelay,
    cCDMTransferMaxAttempts,
    cCDMPushDestTableCount,
    cCDMPushDestTableLastChanged,
    cCDMPushDestTransferType,
    cCDMPushDestAddressLocationType,
    cCDMPushDestAddressLocation,
    cCDMPushDestTransferTime,
    cCDMPushDestPackageSelection,
    cCDMPushDestRowStatus,
    cCDMTransferPkgTableCount,
    cCDMTransferPkgTableLastChanged,
    cCDMTransferPkgLocatorRowPtr,
    cCDMTransferPkgRowStatus
}
STATUS current
DESCRIPTION "This group is composed of objects related to sender
information."
::= { cKeyTransferPushGroups 1 }

cKeyTransferPushReceiverGroup OBJECT-GROUP OBJECTS {
    cCDMPushSrcTableCount,
    cCDMPushSrcTableLastChanged,
    cCDMPushSrcTransferType,
    cCDMPushSrcAddrLocationType,
    cCDMPushSrcAddrLocation,
    cCDMPushSrcRowStatus
}
STATUS current
DESCRIPTION
"This group is composed of objects related to receiver information."
::= { cKeyTransferPushGroups 2 }

cKeyTransferPushSenderNotifyGroup NOTIFICATION-GROUP
NOTIFICATIONS {
cCDMPushSendSuccess,
cCDMPushSendFail
}
STATUS current
DESCRIPTION
"This group is composed of notifications related to sender information."
::= { cKeyTransferPushGroups 3 }

cKeyTransferPushReceiverNotifyGroup NOTIFICATION-GROUP
NOTIFICATIONS {
cCDMPushReceiveSuccess,
cCDMPushReceiveFail
}
STATUS current
DESCRIPTION
"This group is composed of notifications related to receiver information."
::= { cKeyTransferPushGroups 4 }

END

5.7. Security Policy Information

This module makes reference to: Section 5.2, [RFC2578], [RFC2579], [RFC2580], and {RFC3411}.

CC-SECURE-POLICY-INFO-MIB  DEFINITIONS  ::=  BEGIN
IMPORTS
ccSecurePolicyInfo
FROM CC-FEATURE-HIERARCHY-MIB -- FROM Sec 5.2
OBJECT-TYPE, Unsigned32, NOTIFICATION-TYPE,
MODULE-IDENTITY
FROM SNMPv2-SMI -- FROM RFC 2578
MODULE-COMPLIANCE, OBJECT-GROUP,
NOTIFICATION-GROUP
FROM SNMPv2-CONF -- FROM RFC 2580
SnmpAdminString
FROM SNMP-FRAMEWORK-MIB -- FROM RFC 3411
RowStatus, TimeStamp
FROM SNMPv2-TC; -- FROM RFC 2579
ccSecurePolicyInfoMIB  MODULE-IDENTITY
LAST-UPDATED "201609302154Z"
ORGANIZATION "CCMIB CCB"
CONTACT-INFO
  "CC MIB Configuration Control Board
  Email: CCMIB.CCB@us.af.mil"
DESCRIPTION
  "This MIB defines the CC MIB Security Policy Information
  objects.

Copyright (c) 2019 IETF Trust and the persons
identified as authors of the code. All rights reserved.

Redistribution and use in source and binary forms, with
or without modification, is permitted pursuant to, and
subject to the license terms contained in, the Simplified
BSD License set forth in Section 4.c of the IETF Trust’s
Legal Provisions Relating to IETF Documents
(http://trustee.ietf.org/license-info).

This version of this MIB module is part of RFC xxxx;
see the RFC itself for full legal notices."
-- RFC Ed.: RFC-editor please fill in xxxx.
REVISION  "201609302154Z"
DESCRIPTION  "CC MIB 1.0.5 FINAL. Published as RFC xxxx."
-- RFC Ed.: RFC-editor please fill in xxxx.
 ::= { ccSecurePolicyInfo 1 }

-- ************************************************************************
-- Secure Policy Info Information Segments
-- ************************************************************************
cSecurePolicyConformance  OBJECT IDENTIFIER
  ::= { ccSecurePolicyInfoMIB 1 }
cSecPolicyRuleInfo  OBJECT IDENTIFIER
  ::= { ccSecurePolicyInfoMIB 2 }
cSecurePolicyInfoScalars  OBJECT IDENTIFIER
  ::= { ccSecurePolicyInfoMIB 3 }
cSecurePolicyInfoNotify OBJECT IDENTIFIER
  ::= { ccSecurePolicyInfoMIB 4 }

-- ************************************************************************
-- Secure Policy Info Scalars
-- ************************************************************************

-- ************************************************************************
-- Secure Policy Info Notifications
-- ************************************************************************
cSecPolicyChanged NOTIFICATION-TYPE
OBJECTS {
    cSecPolicyRulePriorityID,
    cSecPolicyRuleDescription
}
STATUS current
DESCRIPTION
"A notification indicating that an existent Security Policy
entry in the cSecPolicyRuleTable in has changed."
::= { cSecurePolicyInfoNotify 1 }

-- *****************************************************************
-- CC MIB cSecPolicyRuleTable
-- *****************************************************************

cSecPolicyRuleTableCount OBJECT-TYPE
SYNTAX   Unsigned32
MAX-ACCESS read-only
STATUS   current
DESCRIPTION
"The number of rows in the cSecPolicyRuleTable."
::= { cSecPolicyRuleInfo 1 }

cSecPolicyRuleTableLastChanged OBJECT-TYPE
SYNTAX    TimeStamp
MAX-ACCESS read-only
STATUS    current
DESCRIPTION
"The last time any entry in the table was modified, created,
or deleted by either SNMP, agent, or other management method
(e.g., via an HMI). Managers can use this object to ensure
that no changes to configuration of this table have happened
since the last time it examined the table. A value of 0
indicates that no entry has been changed since the agent
initialized. The value in CC-DEVICE-INFO-MIB cSystemUpTime
should be used to populate this column."
::= { cSecPolicyRuleInfo 2 }

cSecPolicyRuleTable OBJECT-TYPE
SYNTAX    SEQUENCE OF CSecPolicyRuleEntry
MAX-ACCESS not-accessible
STATUS    current
DESCRIPTION
"The cSecPolicyRuleTable stores the Security Policy Rules
that are compared against inbound and outbound data traffic
flow. These Security Policy Rules define the actions (e.g.,
protect, bypass, discard) on how the data traffic flow
should be treated."
::= { cSecPolicyRuleInfo 3 }

cSecPolicyRuleEntry  OBJECT-TYPE
SYNTAX   CSecPolicyRuleEntry
MAX-ACCESS not-accessible
STATUS   current
DESCRIPTION
   "A row containing general information about a Security
   Policy rule."
INDEX    { cSecPolicyRulePriorityID }
::= { cSecPolicyRuleTable 1 }

CSecPolicyRuleEntry ::= SEQUENCE {
cSecPolicyRulePriorityID        Unsigned32,
cSecPolicyRuleDescription       OCTET STRING,
cSecPolicyRuleType              INTEGER,
cSecPolicyRuleFilterReference   SnmpAdminString,
cSecPolicyRuleAction            INTEGER,
cSecPolicyRuleRowStatus         RowStatus
}

cSecPolicyRulePriorityID  OBJECT-TYPE
SYNTAX   Unsigned32
MAX-ACCESS read-only
STATUS   current
DESCRIPTION
   "Local unique index that identifies the priority at which
   this Security Policy rule is applied. Lower values have a
   higher priority (e.g., a value of 1 will be processed before
   a value of 2). This column is the primary index to the
   cSecPolicyRuleTable."
::= { cSecPolicyRuleEntry 1 }

cSecPolicyRuleDescription  OBJECT-TYPE
SYNTAX   OCTET STRING
MAX-ACCESS read-create
STATUS   current
DESCRIPTION
   "An administrative string describing the Security Policy
   rule. Note, this is a free form OCTET STRING that provides
   the user a store for any form of description/documentation
   for the given entry."
::= { cSecPolicyRuleEntry 2 }

cSecPolicyRuleType  OBJECT-TYPE
SYNTAX   INTEGER { ipsec(1), tls(2), macsec(3) }
MAX-ACCESS read-create
STATUS   current
DESCRIPTION
"Optional column that defines the related protocol type of the Security Policy rule. Depending on this column’s set value, entries will vary in respect to which other columns/tables (if at all) must be populated to fully configure the Security Policy rule."
::= { cSecPolicyRuleEntry 3 }

cSecPolicyRuleFilterReference OBJECT-TYPE
SYNTAX SnmpAdminString
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"A string that references the associated filter for the Security Policy rule. Data traffic flow (inbound/outbound) comparison against the associated filter provide the basis in which a Security Policy rule is applied to the given data traffic flow."
::= { cSecPolicyRuleEntry 4 }

cSecPolicyRuleAction OBJECT-TYPE
SYNTAX INTEGER { protect(1), bypass(10), discard(20),
discardInbound(21), discardOutbound(22) }
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"This object indicates what action the ECU should take on matching a data traffic flow against a filter (as defined by cSecPolicyRuleFilterReference). The value of this column can take one of four enumeration values.

[1] protect: The ‘protect’ enumeration value indicates that the data traffic flow should be protected by a Secure Connection with attributes defined by the associated filter (cSecPolicyRuleFilterReference).

[10] bypass: The ‘bypass’ enumeration value indicates that the data traffic flow should be bypassed with no cryptographic protection/services provided.

[20] discard: The ‘discard’ enumeration value indicates that the data traffic flow, agnostic of their direction, should be discarded.

[21] discardInbound: The ‘discardInbound’ enumeration value indicates that an inbound data traffic flow should be discarded."
discardOutbound: The 'discardOutbound' enumeration value indicates that an outbound data traffic flow should be discarded.

Implementations that do not support the 'discardInbound' and 'discardOutbound' enumeration values should return a wrongValue exception during a SET to the cSecPolicyRuleAction object.

A valid enumeration value must be specified in order for cSecPolicyRuleRowStatus to be 'active'.

::= { cSecPolicyRuleEntry 5 }

cSecPolicyRuleRowStatus OBJECT-TYPE
SYNTAX RowStatus
MAX-ACCESS read-create
STATUS current
DESCRIPTION "The status of the row, by which new entries may be created, or old entries deleted from this table.

Entries created within this table may not become active unless all read-create columns in this table have valid values, as detailed by each individual column's description.

At a minimum, implementations must support createAndGo and destroy management functions. Support for createAndWait, active, notInService, and notReady management functions is optional."

::= { cSecPolicyRuleEntry 6 }

-- *****************************************************************
-- Module Conformance Information
-- *****************************************************************

cSecurePolicyCompliances OBJECT IDENTIFIER ::= { cSecurePolicyConformance 1 }

cSecurePolicyGroups OBJECT IDENTIFIER ::= { cSecurePolicyConformance 2 }

cSecurePolicyCompliance MODULE-COMPLIANCE
STATUS current
DESCRIPTION "Compliance levels for secure policy information."

MODULE
MANDATORY-GROUPS { cSecurePolicyGroup }
GROUP cSecurePolicyNotifyGroup
DESCRIPTION "This notification group is optional for implementation."
 ::= { cSecurePolicyCompliances 1 }

cSecurePolicyGroup OBJECT-GROUP
OBJECTS {
  cSecPolicyRuleTableCount,
  cSecPolicyRuleTableLastChanged,
  cSecPolicyRulePriorityID,
  cSecPolicyRuleDescription,
  cSecPolicyRuleType,
  cSecPolicyRuleFilterReference,
  cSecPolicyRuleAction,
  cSecPolicyRuleRowStatus
}
STATUS current
DESCRIPTION "This group is composed of objects related to secure policy
information."
 ::= { cSecurePolicyGroups 1 }

5.8. Secure Connection Information

This module makes reference to: Section 5.2, [RFC2578], [RFC2579], [RFC2580], [RFC3411], and [RFC4303].

CC-SECURE-CONNECTION-INFO-MIB DEFINITIONS ::= BEGIN

IMPORTS
  ccSecureConnectionInfo
    FROM CC-FEATURE-HIERARCHY-MIB             -- FROM Sec 5.2
OBJECT-TYPE, Unsigned32, NOTIFICATION-TYPE,
MODULE-IDENTITY
  FROM SNMPv2-SMI                           -- FROM RFC 2578
MODULE-COMPLIANCE, OBJECT-GROUP,
ccSecureConnectionInfoMIB  MODULE-IDENTITY
LAST-UPDATED  "201609302154Z"
ORGANIZATION  "CCMIB CCB"
CONTACT-INFO
   "CC MIB Configuration Control Board
   Email: CCMIB.CCB@us.af.mil"
DESCRIPTION
 "This MIB defines the CC MIB Secure Connection Information
   objects.

Copyright (c) 2019 IETF Trust and the persons
   identified as authors of the code. All rights reserved.

Redistribution and use in source and binary forms, with
   or without modification, is permitted pursuant to, and
   subject to the license terms contained in, the Simplified
   BSD License set forth in Section 4.c of the IETF Trust’s
   Legal Provisions Relating to IETF Documents
   (http://trustee.ietf.org/license-info).

This version of this MIB module is part of RFC xxxx;
   see the RFC itself for full legal notices."
   -- RFC Ed.: RFC-editor please fill in xxxx.
REVISION      "201609302154Z"
DESCRIPTION   "CC MIB 1.0.5 FINAL. Published as RFC xxxx."
   -- RFC Ed.: RFC-editor please fill in xxxx.
   ::= { ccSecureConnectionInfo 1 }
Secure Connection Info Scalars

Secure Connection Info Notifications

cSecConnectionEstablished  NOTIFICATION-TYPE
OBJECTS     { cSecConTableID }
STATUS      current
DESCRIPTION
"A notification indicating that a new Secure Connection was
successfully established."
::= { cSecureConnectionInfoNotify 1 }

cSecConnectionDeleted  NOTIFICATION-TYPE
OBJECTS     { cSecConTableID }
STATUS      current
DESCRIPTION
"A notification indicating that an existent Secure
Connection was successfully deleted."
::= { cSecureConnectionInfoNotify 2 }

CC MIB cSecConTable

cSecConTableCount  OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"The number of rows in the cSecConTable."
::= { cSecureConnectionInfo 1 }

cSecConTableLastChanged  OBJECT-TYPE
SYNTAX      TimeStamp
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"The last time any entry in the table was modified, created,
or deleted by either SNMP, agent, or other management method
(e.g., via an HMI). Managers can use this object to ensure
that no changes to configuration of this table have happened
since the last time it examined the table. A value of 0
indicates that no entry has been changed since the agent
The table provides the base/common information for Secure Connections.

::= { cSecureConnectionInfo 3 }

cSecConEntry OBJECT-TYPE
SYNTAX CSecConEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION "A row containing general information about an active/inactive Secure Connection."
INDEX { cSecConTableID }
::= { cSecConTable 1 }

CSecConEntry ::= SEQUENCE {
    cSecConTableID              Unsigned32,
    cSecConType                 OCTET STRING,
    cSecConDataPlaneID          OCTET STRING,
    cSecConDirection            INTEGER,
    cSecConKeyReference         OCTET STRING,
    cSecConCryptographicSuite   OCTET STRING,
    cSecConEstablishmentTime    DateAndTime,
    cSecConStatus               OCTET STRING,
    cSecConRowStatus            RowStatus,
    cSecConRemoteKeyReference   OCTET STRING
}

cSecConTableID OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Local unique index that identifies a Secure Connection. This column is the primary index to the cSecConTable."
::= { cSecConEntry 1 }

cSecConType OBJECT-TYPE
Optional column that defines the related protocol type of the Secure Connection. Depending on this column’s populated value, entries will vary in respect to which other columns/tables (if at all) are applicable to the Secure Connection. Example of values for this column are: ‘ipsec’ for Internet Protocol Security secure connections and ‘tls’ for Transport Layer Security/Secure Socket Layer secure connections.

 ::= { cSecConEntry 2 }

cSecConDataPlaneID OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The unique identifier associated with the Secure Connection, based on the Secure Connection protocol. Note, this is a free form OCTET STRING column where meaningful values/format are defined per Secure Connection protocol type basis. For instance, in an IPsec context (i.e., cSecConType value is set to ‘ipsec’), this column would store the Security Parameter Index (SPI) for a given Encapsulating Security Payload Version 3 Security Association (RFC 4303 - Section 2.1)."

 ::= { cSecConEntry 3 }

cSecConDirection OBJECT-TYPE
SYNTAX INTEGER { inbound(1), outbound(2), bidirectional(3) }
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The data plane traffic flow direction for the Secure Connection.


cSecConKeyReference  OBJECT-TYPE
SYNTAX    OCTET STRING (SIZE(0..255))
MAX-ACCESS read-create
STATUS    current
DESCRIPTION
"Administrative string that references key material
associated with the Secure Connection. This column
references an entry (via table index value) in a key-related
table in the CC-KEY-MANAGEMENT-MIB.

If there is no appropriate value to populate with, this
column would be populated with an empty string, ''."

::= { cSecConEntry 5 }

cSecConCryptographicSuite  OBJECT-TYPE
SYNTAX    OCTET STRING
MAX-ACCESS read-create
STATUS    current
DESCRIPTION
"The set of cryptographic attributes (e.g. Encryption
Algorithm, Integrity Algorithm) respective to the Secure
Connection. Note, this is a free form OCTET STRING column,
meaning implementations may utilize a standardized
definition of string values that describe a set of
cryptographic suites or use a proprietary definition of
string values for supported cryptographic suites."

::= { cSecConEntry 6 }

cSecConEstablishmentTime  OBJECT-TYPE
SYNTAX    DateAndTime
MAX-ACCESS read-create
STATUS    current
DESCRIPTION
"The local date and time when the Secure Connection was or
will be established. The value in this column may be
manually set to a date and time prior to the effective date
of the key material (if associated) as referenced by the
cSecConKeyReference column. If this column value is not
manually configured with a date and time then the value will
be automatically populated with the current cSystemDate
value in respect to when the cSecConRowStatus column is
first set to Active.

Note, implementations may treat this column as an alpha date
for the Secure Connection, and thus ascertain other Secure
Connection-related values based on this time."
::= { cSecConEntry 7 }

cSecConStatus OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"Column that provides the current status of the Secure Connection. Note, this is a free form OCTET STRING column where meaningful values are defined per Secure Connection protocol type basis (i.e., as defined by the cSecConType value) or per implementation basis.

If there is no appropriate value to populate with, this column would be populated with an empty string, "."
::= { cSecConEntry 8 }

cSecConRowStatus OBJECT-TYPE
SYNTAX RowStatus
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The status of the row, by which new entries may be created, or old entries deleted from this table.

Entries created within this table may not become active unless all read-create columns in this table have valid values, as detailed by each individual column's description.

The set of RowStatus enumerations that must be supported is dependent on the type of secure connection. At a minimum, implementations must support createAndGo and destroy if the secure connection can be created and destroyed by the manager. Implementations must support active and notInService if the secure connection can be enabled/disabled by the manager."
::= { cSecConEntry 9 }

cSecConRemoteKeyReference OBJECT-TYPE
SYNTAX OCTET STRING (SIZE(0..255))
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"Administrative string that references remote key material associated with the Secure Connection (i.e., the remote key material used by the peer to establish the Secure Connection. This column references an entry (via table index value) in cRemoteKeyMaterialTable (CC-KEY-MANAGEMENT-MIB)."
If there is no appropriate value to populate with, this column would be populated with an empty string, ""
::= {cSecConEntry 10}

-- Module Conformance Information
-- *****************************************************************

-- *****************************************************************

-- Module Conformance Information

-- *****************************************************************


cSecureConnectionCompliances OBJECT IDENTIFIER
::= {cSecureConnectionConformance 1}

cSecureConnectionGroups OBJECT IDENTIFIER
::= {cSecureConnectionConformance 2}

cSecureConnectionCompliance MODULE-COMPLIANCE
STATUS current
DESCRIPTION
"Compliance levels for secure connection information."
MODULE
MANDATORY-GROUPS {cSecureConnectionGroup}

GROUP cSecureConnectionNotifyGroup
DESCRIPTION
"This notification group is optional for implementation."

OBJECT cSecConType
MIN-ACCESS not-accessible
DESCRIPTION
"Implementation of this object is optional."
::= {cSecureConnectionCompliances 1}

cSecureConnectionGroup OBJECT-GROUP
OBJECTS {
cSecConTableCount,
cSecConTableLastChanged,
cSecConTableID,
cSecConType,
cSecConDataPlaneID,
cSecConDirection,
cSecConKeyReference,
cSecConCryptographicSuite,
cSecConEstablishmentTime,
cSecConStatus,
cSecConRowStatus,
cSecConRemoteKeyReference
}
STATUS current
DESCRIPTION
"This group is composed of objects related to secure connection information." 
::= { cSecureConnectionGroups 1 }

cSecureConnectionNotifyGroup NOTIFICATION-GROUP
  NOTIFICATIONS { 
    cSecConnectionEstablished, 
    cSecConnectionDeleted 
  }

STATUS current
DESCRIPTION
"This group is composed of notifications related to secure connection information." 
::= { cSecureConnectionGroups 2 }

END

6. IANA Considerations

This document makes no requests of IANA. All of the object identifiers used in the document are defined in the IANA Private Enterprise Number (PEN) ccmib arc (34493).

RFC EDITOR: Please delete the following note prior to publication

NOTE: "cpsg" is undergoing a name change to "ccmib".

7. Security Considerations

SNMP versions prior to SNMPv3 did not include adequate security. Even if the network itself is secure (for example by using IPsec), there is no control as to who on the secure network is allowed to access and GET/SET (read/change/create/delete) the objects in this MIB module.

Implementations SHOULD provide the security features described by the SNMPv3 framework (see [RFC3410]), and implementations claiming compliance to the SNMPv3 standard MUST include full support for authentication and privacy via the User-based Security Model (USM) [RFC3414] with the AES cipher algorithm [RFC3826]. Implementations MAY also provide support for the Transport Security Model (TSM) [RFC5591] in combination with a secure transport such as SSH [RFC5592] or TLS/DTLS [RFC6353].

Further, deployment of SNMP versions prior to SNMPv3 is NOT RECOMMENDED. Instead, it is RECOMMENDED to deploy SNMPv3 and to enable cryptographic security. It is then a customer/operator responsibility to ensure that the SNMP entity giving access to an
instance of this MIB module is properly configured to give access to
the objects only to those principals (users) that have legitimate
rights to indeed GET or SET (change/create/delete) them.

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
Requirement Levels", BCP 14, RFC 2119,
editor.org/info/rfc2119>.

Schoenwaelder, Ed., "Structure of Management Information
Version 2 (SMIv2)", STD 58, RFC 2578,
DOI 10.17487/RFC2578, April 1999, <https://www.rfc-
editor.org/info/rfc2578>.

Schoenwaelder, Ed., "Textual Conventions for SMIv2",
STD 58, RFC 2579, DOI 10.17487/RFC2579, April 1999,

Schoenwaelder, Ed., "Conformance Statements for SMIv2",
STD 58, RFC 2580, DOI 10.17487/RFC2580, April 1999,

Architecture for Describing Simple Network Management
Protocol (SNMP) Management Frameworks", STD 62, RFC 3411,
editor.org/info/rfc3411>.

(USM) for version 3 of the Simple Network Management
Protocol (SNMPv3)", STD 62, RFC 3414,
editor.org/info/rfc3414>.

Advanced Encryption Standard (AES) Cipher Algorithm in the
SNMP User-based Security Model", RFC 3826,
editor.org/info/rfc3826>.
8.2. Informative References


[RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)",
RFC 4303, DOI 10.17487/RFC4303, December 2005,

Appendix A.  Contributors

The following people made technical contributions to this specification:

- Shadi Azoum
  Naval Information Warfare Center Pacific
  shadi.azoum@navy.mil

- Elliott Jones
  Naval Information Warfare Center Pacific
  elliott.jones@navy.mil

- Lily Sun
  Naval Information Warfare Center Pacific
  lily.sun@navy.mil

Authors’ Addresses

Jeffrey Sun
Naval Information Warfare Center Pacific
Email: sunjeff@spawar.navy.mil

Mike Irani
Naval Information Warfare Center Pacific
Email: irani@spawar.navy.mil

Tom Nguyen
Naval Information Warfare Center Pacific
Email: tmnguyen@spawar.navy.mil

Ray Purvis
The MITRE Corporation
Email: rpurvis@mitre.org