Network Working Group   M. Pana, Ed.
Request for Comments: 4104   MetaSolv
Updates: 3703   A. Reyes
Category: Standards Track   Computer Architecture, UPC
                                                               A. Barba
                                                               D. Moron
                                   Technical University of Catalonia
                                                             M. Brunner
                                                                    NEC
                                                              June 2005

Policy Core Extension
Lightweight Directory Access Protocol Schema (PCELS)

Status of This Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Copyright Notice

Abstract

This document defines a number of changes and extensions to the Policy Core Lightweight Directory Access Protocol (LDAP) Schema (RFC 3703) based on the model extensions defined by the Policy Core Information Model (PCIM) Extensions (RFC 3460). These changes and extensions consist of new LDAP object classes and attribute types. Some of the schema items defined in this document re-implement existing concepts in accordance with their new semantics introduced by RFC 3460. The other schema items implement new concepts, not covered by RFC 3703. This document updates RFC 3703.

Table of Contents

1. Introduction 3
   1.1. Specification of Requirements 3
2. Relationship to Other Policy Framework Documents 3
3. Inheritance Hierarchy for PCELS 4

4. General Discussion of Mapping the Policy Core
   Information Model 8
   4.1. Summary of Class Mappings 8
   4.2. Summary of Association Mappings 11
   4.3. Summary of Changes since PCLS 13
   4.4. Relationship to PCLS Classes 15
   4.5. Impact on Existing Implementations of the Policy
        Core LDAP Schema 16
   4.6. The Association of PolicyVariable and PolicyValues 16
   4.7. The Aggregation of PolicyRules and PolicyGroups in
        PolicySets 17
   4.8. The Aggregation of Actions/Conditions in PolicyRules and
        CompoundActions/CompoundConditions 20
5. Class Definitions 25
   5.1. The Abstract Class pcelsPolicySet 26
   5.2. The Structural Class pcelsPolicySetAssociation 29
   5.3. The Three Policy Group Classes 30
   5.4. The Three Policy Rule Classes 31
   5.5. The Structural Class pcelsConditionAssociation 36
   5.6. The Structural Class pcelsActionAssociation 37
   5.7. The Auxiliary Class pcelsSimpleConditionAuxClass 38
   5.8. The Auxiliary Class pcelsCompoundConditionAuxClass 39
   5.9. The Auxiliary Class pcelsCompoundFilterConditionAuxClass ..40
   5.10. The Auxiliary Class pcelsSimpleActionAuxClass 41
   5.11. The Auxiliary Class pcelsCompoundActionAuxClass 42
   5.12. The Abstract Class pcelsVariable 43
   5.13. The Auxiliary Class pcelsExplicitVariableAuxClass 44
   5.14. The Auxiliary Class pcelsImplicitVariableAuxClass 46
   5.15. The Subclasses of pcelsImplicitVariableAuxClass 47
   5.16. The Auxiliary Class pcelsValueAuxClass. 54
   5.17. The Subclasses of pcelsValueAuxClass. 55
   5.18. The Three Reusable Policy Container Classes 60
   5.19. The Structural Class pcelsRoleCollection. 62
   5.20. The Abstract Class pcelsFilterEntryBase 64
   5.21. The Structural Class pcelsIPHeadersFilter 65
   5.22. The Structural Class pcels8021Filter 73
   5.23. The Auxiliary Class pcelsFilterListAuxClass 77
   5.24. The Auxiliary Class pcelsVendorVariableAuxClass 79
   5.25. The Auxiliary Class pcelsVendorValueAuxClass 80
6. Security Considerations 81
7. IANA Considerations 82
   7.1. Object Identifiers 82
   7.2. Object Identifier Descriptors 82
8. Acknowledgements 85
9. Normative References 85
10. Informative References 86

1. Introduction

This document defines a number of changes and extensions to the Policy Core Lightweight Directory Access Protocol (LDAP) Schema [PCLS] based on the model extensions defined by the Policy Core Information Model (PCIM) Extensions [PCIM_EXT]. These changes and extensions consist of new LDAP object classes and attribute types [LDAP]. Some of the schema items defined in this document re-implement existing concepts in accordance with their new semantics introduced by [PCIM_EXT]. The other schema items implement new concepts, not covered by [PCLS]. This document updates RFC 3703 [PCLS].

In addition to the concepts defined by [PCIM_EXT], this document introduces two new classes: pcelsVendorVariableAuxClass and pcelsVendorValueAuxClass. These classes provide a standard extension mechanism for vendor-specific policy variables and policy values that have not been specifically modeled.

Within the context of this document, the term "PCELS" (Policy Core Extension LDAP Schema) is used to refer to the LDAP object class, attribute type definitions and the associated recommendations contained in this document.

1.1. Specification of Requirements

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [KEYWORDS].

2. Relationship to Other Policy Framework Documents

This document contains an LDAP schema mapping for the classes defined in the "Policy Core Information Model (PCIM) Extensions" [PCIM_EXT]. The LDAP schema defined in this document is an extension to the "Policy Core Lightweight Directory Access Protocol (LDAP) Schema" [PCLS], which defines the mapping of the "Policy Core Information Model -- Version 1 Specification" [PCIM] to an LDAP schema.

These three documents ([PCIM], [PCIM_EXT] and [PCLS]) are prerequisites for reading and understanding this document.

Other documents may subsequently be produced with mappings of the same model to other storage or transport technologies.

3. Inheritance Hierarchy for PCELS

The object class and attribute type names defined in this document are prefixed 'pcels'.

The diagram below illustrates the combined class hierarchy for the LDAP object classes defined in the following documents:

- The class names prefixed 'pcels' are defined in this document.
- The class names prefixed 'pcim' are defined in [PCLS].
- The class names prefixed 'dlm1' are defined in [CIM_LDAP].
- The class named 'top' is defined in [LDAP_SCHEMA].

All the new object classes except for pcelsVendorVariableAuxClass and pcelsVendorValueAuxClass, are mapped from concepts defined or modified by [PCIM_EXT]. The pcelsVendorVariableAuxClass and pcelsVendorValueAuxClass classes are not mapped from [PCIM_EXT]. They represent concepts introduced in this document.

```
      top
      |
      +---dlm1ManagedElement (abstract)
      |   |
      |   +---pcimPolicy (abstract)
      |   |   |
      |   |   +---pcelsPolicySet (abstract new)
      |   |   |   |
      |   |   |   +---pcelsGroup (abstract new)
      |   |   |   |   |
      |   |   |   |   +---pcelsGroupAuxClass (auxiliary new)
      |   |   |   |   |
      |   |   |   |   +---pcelsGroupInstance (structural new)
      |   |   |   |
      |   |   |   +---pcelsRule (abstract new)
      |   |   |       |
      |   |   |       +---pcelsRuleAuxClass (auxiliary new)
      |   |   |       |
      |   |   |       +---pcelsRuleInstance (structural new)
      |   |   |
      |   |   +---pcimGroup (abstract)
      |   |   |   |
      |   |   |   +---pcimGroupAuxClass (auxiliary)
      |   |   |   |
      |   |   |   +---pcimGroupInstance (structural)
      |   |   |
      |   |   +---pcimRule (abstract)
      |   |   |   |
      |   |   |   +---pcimRuleAuxClass (auxiliary)
      |   |   |   |
      |   |   |   +---pcimRuleInstance (structural)
      |   |   |
      |   |   +---pcimRuleConditionAssociation (structural)
      |   |   |   |
      |   |   |   +---pcelsConditionAssociation (structural new)
      |   |   |
      |   |   +---pcimRuleValidityAssociation (structural)
      |   |   |
      |   |   +---pcimRuleActionAssociation (structural)
      |   |   |   |
      |   |   |   +---pcelsActionAssociation (structural new)
      |   |   |
      |   |   +---pcelsPolicySetAssociation (structural new)
      |   |   |
      |   |   +---pcimPolicyInstance (structural)
      |   |   |
      |   |   +---pcimElementAuxClass (auxiliary)
      |   |   |
      |   |   +---pcelsRoleCollection (structural new)
      |   |   |
      |   |   +---pcelsFilterEntryBase (abstract new)
      |   |       |
      |   |       +---pcelsIPHeadersFilter (structural new)
      |   |       |
      |   |       +---pcels8021Filter (structural new)
      |   |
      |   +---dlm1ManagedSystemElement (abstract)
      |       |
      |       +---dlm1LogicalElement (abstract)
      |           |
      |           +---dlm1System (abstract)
      |               |
      |               +---dlm1AdminDomain (abstract)
      |                   |
      |                   +---pcimRepository (abstract)
      |                       |
      |                       +---pcimRepositoryAuxClass (auxiliary)
      |                       |
      |                       +---pcimRepositoryInstance (structural)
      |                       |
      |                       +---pcelsReusableContainer (abstract new)
      |                           |
      |                           +---pcelsReusableContainerAuxClass
      |                           |   (auxiliary new)
      |                           |
      |                           +---pcelsReusableContainerInstance
      |                               (structural new)
      |
      |
      +---pcimConditionAuxClass (auxiliary)
      |   |
      |   +---pcimTPCAuxClass (auxiliary)
      |   |
      |   +---pcimConditionVendorAuxClass (auxiliary)
      |   |
      |   +---pcelsSimpleConditionAuxClass (auxiliary new)
      |   |
      |   +---pcelsCompoundConditionAuxClass (auxiliary new)
      |   |   |
      |   |   +---pcelsCompoundFilterConditionAuxClass (auxiliary new)
      |   |
      |   +---pcelsFilterListAuxClass (auxiliary new)
      |
      +---pcimActionAuxClass (auxiliary)
      |   |
      |   +---pcimActionVendorAuxClass (auxiliary)
      |   |
      |   +---pcelsSimpleActionAuxClass (auxiliary new)
      |   |
      |   +---pcelsCompoundActionAuxClass (auxiliary new)
      |
      +---pcelsVariable (abstract new)
      |   |
      |   +---pcelsVendorVariableAuxClass (auxiliary new)
      |   |
      |   +---pcelsExplicitVariableAuxClass (auxiliary new)
      |   |
      |   +---pcelsImplicitVariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsSourceIPv4VariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsSourceIPv6VariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsDestinationIPv4VariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsDestinationIPv6VariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsSourcePortVariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsDestinationPortVariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsIPProtocolVariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsIPVersionVariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsIPToSVariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsDSCPVariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsFlowIdVariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsSourceMACVariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsDestinationMACVariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsVLANVariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsCoSVariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsEthertypeVariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsSourceSAPVariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsDestinationSAPVariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsSNAPOUIVariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsSNAPTypeVariableAuxClass (auxiliary new)
      |       |
      |       +---pcelsFlowDirectionVariableAuxClass (auxiliary new)
      |
      +---pcelsValueAuxClass (auxiliary new)
      |   |
      |   +---pcelsVendorValueAuxClass (auxiliary new)
      |   |
      |   +---pcelsIPv4AddrValueAuxClass (auxiliary new)
      |   |
      |   +---pcelsIPv6AddrValueAuxClass (auxiliary new)
      |   |
      |   +---pcelsMACAddrValueAuxClass (auxiliary new)
      |   |
      |   +---pcelsStringValueAuxClass (auxiliary new)
      |   |
      |   +---pcelsBitStringValueAuxClass (auxiliary new)
      |   |
      |   +---pcelsIntegerValueAuxClass (auxiliary new)
      |   |
      |   +---pcelsBooleanValueAuxClass (auxiliary new)
      |
      +---pcimSubtreesPtrAuxClass (auxiliary)
      |
      +---pcimGroupContainmentAuxClass (auxiliary)
      |
      +---pcimRuleContainmentAuxClass (auxiliary)
```

Figure 1. LDAP Class Inheritance Hierarchy for PCELS

4. General Discussion of Mapping the Policy Core Information Model Extensions to LDAP

The object classes described in this document contain certain optimizations for a directory that uses LDAP as its access protocol. An example is the use of auxiliary class attachment to LDAP entries for the realization of some of the associations defined in the information model. For instance, the aggregation of a specific SimplePolicyCondition to a reusable PolicyRule [PCIM_EXT] may be realized by attaching a pcelsSimpleConditionAuxClass to a pcelsRuleInstance entry.

Note that other data stores might need to implement the associations differently.

4.1. Summary of Class Mappings

The classes and their properties defined in the information model [PCIM_EXT] map directly to LDAP object classes and attribute types.

The details of this mapping are discussed case by case in section 5.

Information Model (PCIM_EXT)	LDAP Class(es)
PolicySet	pcelsPolicySet
PolicyGroup	pcelsGroup pcelsGroupAuxClass pcelsGroupInstance
PolicyRule	pcelsRule pcelsRuleAuxClass pcelsRuleInstance
SimplePolicyCondition	pcelsSimpleConditionAuxClass
CompoundPolicyCondition	pcelsCompoundConditionAuxClass

CompoundFilterCondition	pcelsCompoundFilterConditionAuxClass
SimplePolicyAction	pcelsSimpleActionAuxClass
CompoundPolicyAction	pcelsCompoundActionAuxClass
PolicyVariable	pcelsVariable
--------------	pcelsVendorVariableAuxClass
PolicyExplicitVariable	pcelsExplicitVariableAuxClass
PolicyImplicitVariable	pcelsImplicitVariableAuxClass
PolicySourceIPv4Variable	pcelsSourceIPv4VariableAuxClass
PolicySourceIPv6Variable	pcelsSourceIPv6VariableAuxClass
PolicyDestinationIPv4Variable	pcelsDestinationIPv4VariableAuxClass
PolicyDestinationIPv6Variable	pcelsDestinationIPv6VariableAuxClass
PolicySourcePortVariable	pcelsSourcePortVariableAuxClass
PolicyDestinationPortVariable	pcelsDestinationPortVariableAuxClass
PolicyIPProtocolVariable	pcelsIPProtocolVariableAuxClass
PolicyIPVersionVariable	pcelsIPVersionVariableAuxClass
PolicyIPToSVariable	pcelsIPToSVariableAuxClass
PolicyDSCPVariable	pcelsDSCPVariableAuxClass
PolicyFlowIDVariable	pcelsFlowIDVariableAuxClass
PolicySourceMACVariable	pcelsSourceMACVariableAuxClass
PolicyDestinationMACVariable	pcelsDestinationMACVariableAuxClass
PolicyVLANVariable	pcelsVLANVariableAuxClass
PolicyCoSVariable	pcelsCoSVariableAuxClass
PolicyEthertypeVariable	pcelsEthertypeVariableAuxClass
PolicySourceSAPVariable	pcelsSourceSAPVariableAuxClass

PolicyDestinationSAPVariable	pcelsDestinationSAPVariableAuxClass
PolicySNAPOUIVariable	pcelsSNAPOUIVariableAuxClass
PolicySNAPTypeVariable	pcelsSNAPTypeVariableAuxClass
PolicyFlowDirectionVariable	pcelsFlowDirectionVariableAuxClass
PolicyValue	pcelsValueAuxClass
-------------	pcelsVendorValueAuxClass
PolicyIPv4AddrValue	pcelsIPv4AddrValueAuxClass
PolicyIPv6AddrValue	pcelsIPv6AddrValueAuxClass
PolicyMACAddrValue	pcelsMACAddrValueAuxClass
PolicyStringValue	pcelsStringValueAuxClass
PolicyBitStringValue	pcelsBitStringValueAuxClass
PolicyIntegerValue	pcelsIntegerValueAuxClass
PolicyBooleanValue	pcelsBooleanValueAuxClass
PolicyRoleCollection	pcelsRoleCollection
ReusablePolicyContainer	pcelsReusableContainer pcelsReusableContainerAuxClass pcelsReusableContainerInstance
FilterEntryBase	pcelsFilterEntryBase
IPHeadersFilter	pcelsIPHeadersFilter
8021Filter	pcels8021Filter
FilterList	pcelsFilterListAuxClass

Figure 2. Mapping of Information Model Extension Classes to LDAP

The pcelsVendorVariableAuxClass and pcelsVendorValueAuxClass classes are not mapped from [PCIM_EXT]. These classes are introduced in this document as a new extension mechanism for vendor-specific policy variables and values that have not been specifically modeled. Just like for any other schema elements defined in this document or in

[PCLS], a particular submodel schema generally will not need to use vendor specific variable and value classes. Submodel schemas SHOULD apply the recommendations of section 5.10 of [PCIM_EXT] with regards to the supported and unsupported elements.

4.2. Summary of Association Mappings

The associations in the information model map to one or more of the following options:

1. Attributes that reference DNs (Distinguished Names)
2. Directory Information Tree (DIT) containment (i.e., superior-subordinate relationships) in LDAP
3. Auxiliary class attachment
4. Association object classes and attributes that reference DNs

The details of this mapping are discussed case by case in section 5.

Information Model Association	LDAP Attribute/Class
PolicySetComponent	pcelsPolicySetComponentList in pcelsPolicySet and pcelsPolicySetDN in pcelsPolicySetAsociation
PolicySetInSystem	DIT Containment and pcelsPolicySetDN in pcelsPolicySetAsociation
PolicyGroupInSystem	DIT Containment and pcelsPolicySetDN in pcelsPolicySetAsociation
PolicyRuleInSystem	DIT Containment and pcelsPolicySetDN in pcelsPolicySetAsociation
PolicyConditionStructure	pcimConditionDN in pcelsConditionAssociation
PolicyConditionInPolicyRule	pcelsConditionList in pcelsRule and pcimConditionDN in pcelsConditionAssociation
PolicyConditionInPolicyCondition	pcelsConditionList in pcelsCompoundConditionAuxClass

	and pcimConditionDN in pcelsConditionAssociation
PolicyActionStructure	pcimActionDN in pcelsActionAssociation
PolicyActionInPolicyRule	pcelsActionList in pcelsRule and pcimActionDN in pcelsActionAssociation
PolicyActionInPolicyAction	pcelsActionList in pcelsCompoundActionAuxClass and pcimActionDN in pcelsActionAssociation
PolicyVariableInSimplePolicy Condition	pcelsVariableDN in pcelsSimpleConditionAuxClass
PolicyValueInSimplePolicy Condition	pcelsValueDN in pcelsSimpleConditionAuxClass
PolicyVariableInSimplePolicy Action	pcelsVariableDN in pcelsSimpleActionAuxClass
PolicyValueInSimplePolicyAction	pcelsValueDN in pcelsSimpleActionAuxClass
ReusablePolicy	DIT containment
ExpectedPolicyValuesForVariable	pcelsExpectedValueList in pcelsVariable
ContainedDomain	DIT containment or pcelsReusableContainerList in pcelsReusableContainer
EntriesInFilterList	pcelsFilterEntryList in pcelsFilterListAuxClass
ElementInPolicyRoleCollection	DIT containment or pcelsElementList in pcelsRoleCollection
PolicyRoleCollectionInSystem	DIT Containment

Figure 3. Mapping of Information Model Extension Associations to LDAP

Two [PCIM_EXT] associations are mapped to DIT containment:

- PolicyRoleCollectionInSystem is a weak association and weak associations map well to DIT containment [CIM_LDAP] (without being limited to this mapping). In the absence of additional constraints, DIT containment is chosen here as the optimal association mapping.

- ReusablePolicy is mapped to DIT containment for scalability reasons. It is expected that applications will associate a large number of policy instances to a ReusablePolicyContainer and DIT containment is a type of association that scales well.

4.3. Summary of Changes since PCLS

This section provides an overview of the changes relative to [PCLS] defined in this document:

1. The concept of a set of policies is introduced by two new object classes: pcelsPolicySet and pcelsPolicySetAssociation. These classes enable the aggregation and relative prioritization of policies (rules and/or groups). The attribute pcelsPriority is used by pcelsPolicySetAssociation instances to indicate the priority of a policy relative to the other policies aggregated by the same set. Applications may use this attribute to apply appropriate ordering to the aggregated policies. This new policy aggregation mechanism provides an alternative to the aggregation mechanism defined by [PCLS] (that defines pcimRuleContainmentAuxClass and/or pcimGroupContainmentAuxClass for attaching components to a pcimGroup).

2. The attribute pcimRoles defined by [PCLS] is used here by the pcelsPolicySet object class. Thus, the role based policy selection mechanism is extended to all the subclasses of pcelsPolicySet.

3. A new attribute pcelsDecisionStrategy is added on the pcelsPolicySet class as a mapping from the decision mechanism.

4. A new class pcelsGroup (with two subclasses), implements the modified semantics of the PolicyGroup in accordance with [PCIM_EXT]. This new class inherits from its superclass pcelsPolicySet the ability to aggregate (with relative priority) other policy rules or groups.

5. A new class pcelsRule (with two subclasses), implements the modified semantics of the PolicyRule in accordance with [PCIM_EXT]. It does not include an absolute priority attribute,

but instances of non-abstract subclasses of pcelsRule can be prioritized relative to each other within a System (behavior inherited from its superclass: pcelsPolicySet). The pcelsRule class also inherits from pcelsPolicySet the ability to aggregate other policy rules or groups, and thus, the ability to construct nested rule structures of arbitrary complexity.

6. A new attribute pcelsExecutionStrategy is added to the pcelsRule and pcelsCompoundActionAuxClass classes to allow the specification of the expected behavior in case of multiple actions aggregated by a rule or by a compound action.

7. Compound Conditions: The pcelsCompoundConditionAuxClass class is added in order to map the CompoundPolicyCondition class. A new class, pcelsConditionAssociation is used to aggregate policy conditions in a pcelsCompoundConditionAuxClass. The same class is also used to aggregate policy conditions in a pcelsRule.

8. Compound Actions: The pcelsCompoundActionAuxClass class is added in order to map the CompoundPolicyAction class. A new class, pcelsActionAssociation is used to aggregate policy actions in a pcelsCompoundActionAuxClass. The same class is also used to aggregate policy actions in a pcelsRule.

9. Simple Conditions, Simple Actions, Variables and Values: The simple condition, simple action, variable and value classes defined by [PCIM_EXT] are directly mapped to LDAP object classes. These are: pcelsSimpleConditionAuxClass, pcelsSimpleActionAuxClass, pcelsVariable and its subclasses, and pcelsValueAuxClass and its subclasses.

10. A general extension mechanism is introduced for representing policy variables and values that have not been specifically modeled. The mechanism is intended for vendor-specific extensions.

11. Reusable Policy Repository: A new class (with two subclasses), pcelsReusableContainer is created as a subclass of pcimRepository. While maintaining compatibility with older [PCLS] implementations, the addition of this class acknowledges the intent of [PCIM_EXT] to avoid the potential for confusion with the Policy Framework component named Policy Repository. The new class enables many-to-many associations between reusable policy containers.

12. The ReusablePolicy association defined in [PCIM_EXT] is realized through subordination to an instance of a non-abstract subclass of pcelsReusableContainer. Thus, reusable policy components

(groups, rules, conditions, actions, variables and values) may be defined as stand-alone entries or stand-alone groups of related entries subordinated (DIT contained) to a pcelsReusableContainer.

13. Device level filter classes are added to the schema.

14. The pcelsRoleCollection class is added to the schema to allow the association of policy roles to resources represented as LDAP entries.

4.4. Relationship to PCLS Classes

Several [PCLS] classes are used in this document to derive other classes. If a PCELS application requires a functionality provided by any of derived classes, then the [PCLS] class MUST also be supported by PCELS implementations. These classes are:

```
pcimPolicy
pcimRuleConditionAssociation
pcimRuleActionAssociation
pcimConditionAuxClass
pcimActionAuxClass
pcimRepository
```

Other [PCLS] classes are neither derived to nor superseded by classes defined in this document. If a PCELS application requires a functionality provided by any of these classes, then the [PCLS] class SHOULD be used. These classes are:

```
pcimRuleValidityAssociation
pcimTPCAuxClass
pcimConditionVendorAuxClass
pcimActionVendorAuxClass
pcimPolicyInstance
pcimElementAuxClass
pcimSubtreesPtrAuxClass
```

Among the classes defined in this document some implement concepts that supersede the concepts implemented by similar [PCLS] classes. PCELS implementations MAY support such [PCLS] classes. These classes are:

```
pcimGroup and its subclasses
pcimRule and its subclasses
pcimGroupContainmentAuxClass
pcimRuleContainmentAuxClass
the subclasses of pcimRepository
```

4.5. Impact on Existing Implementations of the Policy Core LDAP Schema

In general, the intent of PCELS is to extend the functionality offered by the Policy Core LDAP Schema. For the most part, the compatibility with [PCLS] is preserved. The few cases in which compatibility cannot be achieved due to fundamental changes imposed by [PCIM_EXT], are defined here as alternatives to the original implementation.

PCELS does not obsolete nor deprecate the concepts implemented by [PCLS]. The new LDAP schema items are defined in this document in a way that avoids, to the extent possible, interference with the normal operation of a reasonably well-executed implementation of [PCLS]. The intent is to permit at least a harmless coexistence of the two models in the same data repository.

However, it should be noted that the PCELS introduces the following changes that may have an impact on some [PCLS] implementations:

1. Some attributes originally used only by pcimRule or pcimGroup are now also used by classes unknown to [PCLS] implementations (pcelsPolicySet, pcelsRule and pcelsGroup). In particular, the attribute pcimRoles is also used by pcelsPolicySet for role based policy selection.

2. Condition and action association classes originally used by only pcimRule are now used (through subclasses) by pcelsRule as well.

3. pcimRepository containers may include entries of types unknown to [PCLS] implementations.

When the choice exists, PCELS implementations SHOULD support the new schema and MAY also support the one defined by [PCLS]. For example, if PolicyRule support is required, an implementation SHOULD be able to read or read-write (as applicable) pcelsRule entries. The same implementation MAY be able to read or read-write pcimRule.

4.6. The Association of PolicyVariable and PolicyValues to PolicySimpleCondition and PolicySimpleAction

A PolicySimpleCondition, as well as a PolicySimpleAction, includes a single PolicyValue and a single PolicyVariable. Each of them can be attached or referenced by a DN.

The attachment helps create compact PolicyCondition and PolicyAction definitions that can be efficiently provisioned and retrieved from the repository. On the other hand, referenced PolicyVariables and PolicyValues instances can be reused in the construction of multiple policies and permit an administrative partitioning of the data and policy definitions.

4.7. The Aggregation of PolicyRules and PolicyGroups in PolicySets

In [PCIM_EXT], the two aggregations PolicyGroupInPolicyGroup and PolicyRuleInPolicyGroup, are combined into a single aggregation PolicySetComponent. This aggregation and the capability of association between a policy and the ReusablePolicyContainer offer new possibilities of reusability. Furthermore, these aggregations introduce new semantics representing the execution of one PolicyRule within the scope of another PolicyRule.

Since PolicySet is defined in [PCIM_EXT], it is mapped in this document to a new class pcelsPolicySet in order to provide an abstraction for a set of policy rules or groups. The aggregation class PolicySetComponent in [PCIM_EXT] is mapped to a multi-value attribute pcelsPolicySetList in the pcelsPolicySet class and the attribute pcelsPolicySetDN in the pcelsPolicySetAssociation. These attributes refer to the nested rules and groups.

It is possible to store a rule/group nested in another rule/group in two ways. The first way is to define the nested rule/group as specific to the nesting rule/group. The second way is to define the nested rules/groups as reusable.

First case: Specific nested sets (rules/groups).

```
                       +----------+
                       |Rule/Group|
                       |          |
                 +-----|-        -|-----+
                 |     +----------+     |
                 |       *      *       |
                 |       *      *       |
                 |    ****      ****    |
                 |    *            *    |
                 v    *            *    v
               +-----------+   +-----------+
               | SA1+Set1  |   | SA2+Set2  |
               +-----------+   +-----------+

              +------------------------------+
              |LEGEND:                       |
              |  ***** DIT containment       |
              |    +   auxiliary attachment  |
              |  ----> DN reference          |
              +------------------------------+
```

#: Number.
Set#: pcelsRuleAuxClass or pcelsGroupAuxClass auxiliary class.
SA#: pcelsPolicySetAssocation structural class.

Figure 4. Policy Set with Specific Components

The nesting pcelsPolicySet refers to instances of pcelsPolicySetAssociation using the attribute pcelsPolicySetList. These structural association classes are subordinated (DIT contained) to an instance of a non-abstract subclass of pcelsPolicySet and represent the association between the PolicySet and its nested rules/groups. The nested instances of auxiliary subclasses of pcelsPolicySet are attached to the association entries.

Second case: Reusable nested sets (rules/groups).

```
          +----------+             +-------------+
          |Rule/Group|             | ContainerX  |
        +-|-        -|--+          |             |
        | +----------+  |          +-------------+
        |   *      *    |             *        *
        | ***      **** |             *        *
        | *           * v             *        *
        | *          +---+            *        *
        | *          |SA2|         +-------+   *
        v *          |  -|-------->|S1+Set2|   *
       +---+         +---+         +-------+   *
       |SA1|                                   +-------+
       |  -|------------------------------>|S2+Set3|
       +---+                                   +-------+

              +------------------------------+
              |LEGEND:                       |
              |  ***** DIT containment       |
              |    +   auxiliary attachment  |
              |  ----> DN reference          |
              +------------------------------+
```

Set#: pcelsRuleAuxClass or pcelsGroupAuxClass auxiliary class.
SA#: PolicySetAssocation structural class.
S#: structural class.

Figure 5. Policy Set with Reusable Components

The nesting pcelsPolicySet refers to instances of pcelsPolicySetAssociation using the attribute pcelsPolicySetList. These structural association classes are subordinated (DIT contained) to an instance of a non-abstract subclass of pcelsPolicySet and represent the association between the PolicySet and its nested rules/groups. The reusable rules/groups are instantiated here as auxiliary classes and attached to pcimPolicyInstance entries in the reusable container. Another option is to use the structural subclasses for defining reusable rules/groups. The association classes belonging to a nesting policy set are reference the reusable rules/groups using the attribute pcelsPolicySetDN.

A combination of both specific and reusable components is also allowed for the same policy set.

4.8. The Aggregation of Actions/Conditions in PolicyRules and CompoundActions/CompoundConditions

[PCIM_EXT] defines two new classes that allow the designer to create more complex conditions and actions. CompoundPolicyCondition and CompoundPolicyAction classes are mapped in this document to pcelsCompoundConditionAuxClass and pcelsCompoundActionAuxClass classes that are subclasses of pcimConditionAuxClass/pcimActionAuxClass. The compound conditions/actions defined in [PCIM_EXT] extend the capability of the rule to associate, group and evaluate conditions or execute actions. The conditions/actions are associated to compounds conditions/actions in the same way as they are associated to the rules.

In this section, how to store instances of these classes in an LDAP Directory is explained. As a general rule, specific conditions/actions are subordinated (DIT contained) to the rule or compound condition/action that aggregates them and are attached to association class instances. Reusable conditions/actions are subordinated to pcelsReusableContainer instances and attached to pcimPolicyInstance instances.

The examples below illustrate the four possible cases combining specific/reusable compound/non-compound condition/action. The rule has two compound conditions, each one has two different conditions. The schemes can be extended in order to store actions.

The examples below are based on and extend those illustrated in the section 4.4 of [PCLS].

First case: Specific compound condition/action with specific conditions/actions.

```
                          +--------------+
                   +------|     Rule     |------+
                   |      +--------------+      |
                   |           *    *           |
                   |   *********    *********   |
                   v   *                    *   v
                  +---------+          +---------+
                +-| CA1+cc1 |-+      +-| CA2+cc2 |-+
                | +---------+ |      | +---------+ |
                |     * *     |      |     * *     |
                |  **** ****  |      |  **** ****  |
                v  *       *  v      v  *       *  v
               +------+ +------+    +------+ +------+
               |CA3+c1| |CA4+c2|    |CA5+c3| |CA6+c4|
               +------+ +------+    +------+ +------+

                  +------------------------------+
                  |LEGEND:                       |
                  |  ***** DIT containment       |
                  |    +   auxiliary attachment  |
                  |  ----> DN reference          |
                  +------------------------------+
```

#: Number.
CA#: pcelsConditionAssociation structural class.
cc#: pcelsCompoundConditionAuxClass auxiliary class.
c#: subclass of pcimConditionAuxClass.

Figure 6. Specific Compound Conditions with Specific Components

Because the compound conditions/actions are specific to the Rule, They are auxiliary attachments to instances of the structural classes pcelsConditionAssociation or pcelsActionAssociation. These structural classes represent the association between the rule and the compound condition/action. The rule specific conditions/actions are therefore subordinated (DIT contained) to the rule entry.

The conditions/actions are tied to the compound conditions/actions in the same way the compound conditions/actions are tied to rules. Association classes realize the association between the aggregating compound conditions/actions and the specific conditions/actions.

Second case: Rule specific compound conditions/actions with
reusable conditions/actions.

```
           +-------------+                   +---------------+
    +------|     Rule    |-----+             |  ContainerX   |
    |      +-------------+     |             +---------------+
    |           *    *         |              *    *    *   *
    |           *    *         |           ****    *    *   *
    |   *********    ********  |           *       *    *   ********
    |   *                   *  v           *       *    *          *
    |   *               +---------+        *       *    ****       *
    |   *             +-| CA2+cc2 |-+      *       *       *       *
    |   *             | +---------+ |      *       *       *       *
    v   *             |    *  *     |      *       *       *       *
   +---------+        | ****  ****  |      *       *       *       *
 +-| CA1+cc1 |-+      | *        *  v      *       *       *       *
 | +---------+ |      | *     +------+  +-----+    *       *       *
 |    *  *     |      v *     |  CA6 |->|S1+c4|    *       *       *
 | ****  ****  |     +------+ +------+  +-----+ +-----+    *       *
 | *        *  v     |  CA5 |------------------>|S2+c3|    *       *
 | *      +------+   +------+                   +-----+ +-----+    *
 v *      |  CA4 |------------------------------------->|S3+c2|    *
 +------+ +------+                                      +-----+ +-----+
 |  CA3 |------------------------------------------------------>|S4+c1|
 +------+                                                       +-----+

                    +------------------------------+
                    |LEGEND:                       |
                    |  ***** DIT containment       |
                    |    +   auxiliary attachment  |
                    |  ----> DN reference          |
                    +------------------------------+
```

#: Number.
CA#: pcelsConditionAssociation structural class.
cc#: pcelsCompoundConditionAuxClass auxiliary class.
c#: subclass of pcimConditionAuxClass.
S#: structural class

Figure 7. Specific Compound Conditions with Reusable Components

This case is similar to the first one. The conditions/actions are reusable and are therefore not attached to the association classes, but rather to structural classes in the reusable container. The association classes tie the conditions/actions in located in a reusable container to their aggregators using DN references.

Third case: Reusable compound condition/action with specific
conditions/actions.

```
        +--------------+                  +--------------+
        |     Rule     |                  |  RepositoryX |
    +---+--------------+----+             +--------------+
    |        *     *        |                  *    *
    |  *******     *******  |           ********    ********
    |  *                 *  v           *                  *
    |  *            +----------+    +---------+            *
    |  *            |   CA2    |--->| S1+cc2  |            *
    |  *            +----------+  +-+---------+-+          *
    |  *                          |     * *     |          *
    |  *                          |  **** ****  |          *
    |  *                          v  *       *  v          *
    |  *                         +------+ +------+         *
    |  *                         |CA5+c3| |CA6+c4|         *
    v  *                         +------+ +------+         *
  +----------+                                          +---------+
  |   CA1    |----------------------------------------->| S2+cc1  |
  +----------+                                        +-+---------+-+
                                                      |     * *     |
                                                      |  **** ****  |
                                                      v  *       *  v
                                                     +------+ +------+
                                                     |CA3+c1| |CA4+c2|
                                                     +------+ +------+

                    +------------------------------+
                    |LEGEND:                       |
                    |  ***** DIT containment       |
                    |    +   auxiliary attachment  |
                    |  ----> DN reference          |
                    +------------------------------+

      #: Number.
      CA#: pcelsConditionAssociation structural class.
      cc#: pcelsCompoundConditionAuxClass auxiliary class.
      c#: subclass of pcimConditionAuxClass.
      S#: structural class
```

Figure 8. Reusable Compound Conditions with Specific Components

Re-usable compound conditions/actions are attached to structural classes and stored in a reusable policy container. They are related to the rule through a DN reference attribute in the association classes.

Specific conditions/actions are attached to association entries and subordinated (DIT contained) to the aggregating compound conditions/actions.

Fourth case: Reusable conditions/actions and compound
conditions/actions.

```
          +------+          +---------------+    +---------------+
    +-----| Rule |-----+    |  ContainerX   |    |  ContainerY   |
    |     +------+     |    +---------------+    +---------------+
    |      *    *      |         *     *           *   *   *   *
    | ******    ****** |       ***     ***       ***   *   *   *****
    | *              * v       *         *       *     *   *       *
    | *          +-------+  +------+     *       *     *   ***     *
    | *          |  CA2  |->|S1+ca1|     *       *     *     *     *
    | *          +-------+  +------+     *       *     *     *     *
    | *                    /  *  *  \    *       *     *     *     *
    | *                    |**   ** |    *       *     *     *     *
    | *                    |*     * v    *       *     *     *     *
    | *                    |*   +---+    *    +-----+  *     *     *
    | *                    |*   |CA6|----*--->|S3+c4|  *     *     *
    | *                    v*   +---+    *    +-----+  *     *     *
    | *                  +---+           *          +-----+  *     *
    | *                  |CA5|-----------*--------->|S4+c3|  *     *
    v *                  +---+           *          +-----+  *     *
  +-------+                           +------+               *     *
  |  CA1  |-------------------------->|S2+cc1|               *     *
  +-------+                           +------+               *     *
                                     /  *  *  \              *     *
                                     | **  ** |              *     *
                                     | *    * v              *     *
                                     | *  +---+           +-----+  *
                                     | *  |CA4|---------->|S5+c2|  *
                                     v *  +---+           +-----+  *
                                    +---+                      +-----+
                                    |CA3|--------------------->|S6+c1|
                                    +---+                      +-----+

                    +------------------------------+
                    |LEGEND:                       |
                    |  ***** DIT containment       |
                    |    +   auxiliary attachment  |
                    |  ----> DN reference          |
                    +------------------------------+
```

#: Number.
CA#: pcelsConditionAssociation structural class.
cc#: pcelsCompoundConditionAuxClass auxiliary class.

```
      c#: subclass of pcimConditionAuxClass.
      S#: structural class
```

Figure 9. Reusable Compound Conditions with Reusable Components

All the conditions/actions are reusable so they are stored in reusable containers. The figure above illustrates two different reusable policy containers, but the number of containers in the system is decided based on administrative reasons. The conditions, actions, etc. may be stored in the same or different containers with no impact on the policy definition semantics.

5. Class Definitions

The semantics for the policy information classes that are to be mapped directly from the information model to an LDAP representation are detailed in [PCIM_EXT]. Consequently, this document presents only a brief reference to those semantics. The focus here is on the mapping from the information model (which is independent of repository type and access protocol) to a form that can be accessed using LDAP. For various reasons including LDAP specific optimization, this mapping is not always 1:1. Some new classes and attributes (that were not part of [PCIM] or [PCIM_EXT]) needed to be created in order to implement the LDAP mapping. These new LDAP-only classes are fully defined in this document.

The following notes apply to this section in its entirety.

Note 1: The formal language for specifying the classes, attributes, and DIT structure and content rules is that defined in [LDAP_SYNTAX]. In the following definitions, the class and attribute definitions follow [LDAP_SYNTAX] but they are line-wrapped to enhance human readability.

Note 2: Even though not explicitly noted in the following class and attribute definitions, implementations may define DIT structure and content rules where applicable and supported by the underlying LDAP infrastructure. In such cases, the DIT structure rule considerations discussed in section 5 of [PCLS] must be applied to PCELS implementations as well. The reasons and details are presented in [X.501].

Note 3: Wherever possible, an equality, a substrings and an ordering matching rule are defined for a particular attribute. This provides additional implementation flexibility. However, in some cases, the LDAP matching semantics may not cover all the application needs. For instance, different values of pcelsIPv4AddrList may be semantically equivalent. The equality matching rule, caseIgnoreMatch, associated to this attribute type is not suitable for detecting this equivalence. Implementers should not rely solely on LDAP syntaxes and matching rules for being consistent with this specification.

Note 4: The following attribute definitions use only LDAP matching rules and syntax definitions from [LDAP_SYNTAX], [LDAP_SCHEMA] and [LDAP_MATCH]. The corresponding X.500 matching rules are defined in [X.520].

Note 5: Some of the following attribute types MUST conform to additional constraints on various data types (e.g., the only valid values for pcelsDecisionStrategy are 1 and 2). Just like the attribute semantics, the definition of the value structures, valid ranges, etc. is covered by [PCIM_EXT] for the corresponding properties while such constraints are only briefly mentioned in this document. In all cases, if a constraint is violated, the entry SHOULD be treated as invalid and the policy rules or groups that refer to it SHOULD be treated as being disabled, meaning that the execution of such policy rules or groups SHOULD be stopped.

Note 6: Some of the object classes defined in this section cannot or should not be directly instantiated because they are either defined as abstract or do not implement stand-alone semantics (e.g., pcelsValueAuxClass). Regarding instances of objects that inherit from such classes, the text refers to "instances of <class_name>" when in fact the strictly correct expression would be "instances of objects that belong to non-abstract subclasses of <class_name>". The omission is intentional; it makes the text easier to read.

5.1. The Abstract Class pcelsPolicySet

The pcelsPolicySet class represents a set of policies with a common decision strategy and a common set of policy roles. This class together with the pcelsPolicySetAssociation class defined in a subsequent section of this document provide sufficient information to allow applications to apply appropriate ordering to a set of policies. The pcelsPolicySet is mapped from the PolicySet class [PCIM_EXT]. The pcelsPolicySet class is an abstract object class and it is derived from the pcimPolicy class [PCLS].

The pcelsPolicySetList attribute of a pcelsPolicySet instance references subordinated pcelsPolicySetAssociation entries. The aggregated pcelsPolicySet instances are either attached to the pcelsPolicySetAssociation entries as auxiliary object classes or referenced by the pcelsPolicySetAssociation entries using the pcelsPolicySetDN attribute.

The pcelsPolicySet class is defined as follows:

```
( 1.3.6.1.1.9.1.1
  NAME 'pcelsPolicySet'
  DESC 'Set of policies'
  SUP pcimPolicy
  ABSTRACT
  MAY ( pcelsPolicySetName
      $ pcelsDecisionStrategy
      $ pcimRoles
      $ pcelsPolicySetList )
)
```

One of the attributes of the pcelsPolicySet class, pcimRoles is defined in the section 5.3 of [PCLS]. In the pcelsPolicySet class the pcimRole attribute preserves its syntax and semantics as defined by [PCLS] and [PCIM].

The pcelsPolicySetName attribute type may be used as naming attribute for pcelsPolicySet entries. This attribute type is of syntax Directory String [LDAP_SYNTAX]. It has an equality matching rule of caseIgnoreMatch, an ordering matching rule of caseIgnoreOrderingMatch and a substrings matching rule of caseIgnoreSubstringsMatch [LDAP_SYNTAX]. Attributes of this type can only have a single value.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.1
  NAME 'pcelsPolicySetName'
  DESC 'User-friendly name of a policy set'
  EQUALITY caseIgnoreMatch
  ORDERING caseIgnoreOrderingMatch
  SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)
```

The pcelsDecisionStrategy attribute type indicates the evaluation method for the policies aggregated in the policy set. It is mapped from the PolicySet.PolicyDecisionStrategy property [PCIM_EXT]. This attribute type is of syntax Integer [LDAP_SYNTAX]. It has an equality matching rule of integerMatch [LDAP_SYNTAX] and an ordering matching rule of integerOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are 1 (FirstMatching) and 2 (AllMatching). If this attribute is missing from a pcelsPolicySet instance, applications MUST assume a FirstMatching decision strategy for the policy set.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.2
  NAME 'pcelsDecisionStrategy'
  DESC 'Evaluation method for the components of a pcelsPolicySet'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
```

The pcelsPolicySetList attribute type is used in the realization of the PolicySetComponent association [PCIM_EXT]. This attribute type is of syntax DN [LDAP_SYNTAX]. It has an equality matching rule of distinguishedNameMatch [LDAP_SYNTAX]. Attributes of this type can have multiple values. The only allowed values for pcelsPolicySetList attributes are DNs of pcelsPolicySetAssociation entries. In a pcelsPolicySet, the pcelsPolicySetList attribute represents the associations between this policy set and its components.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.3
  NAME 'pcelsPolicySetList'
  DESC 'Unordered set of DNs of pcelsPolicySetAssociation entries'
  EQUALITY distinguishedNameMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
)
```

Note: A pcelsPolicySet instance aggregates other pcelsPolicySet instances using pcelsPolicySetAssociation entries (defined in the next section). Applications can sort the components of a pcelsPolicySet using attributes of the pcelsPolicySetAssociation entries. However, implementations should not expect the LDAP data store to provide a useful ordering of the pcelsPolicySetList values in a pcelsPolicySet instance or to return sets of matching pcelsPolicySetAssociation entries in a meaningful order. Instead, applications SHOULD implement their own means for post-retrieval ordering of policy rules/groups based on pcelsPolicySetAssociation.pcelsPriority values.

5.2. The Structural Class pcelsPolicySetAssociation

The pcelsPolicySetAssociation class is used to associate PolicySet instances [PCIM_EXT] to other entries. pcelsPolicySetAssociation entries are always subordinated to the aggregating entry. When subordinated to an instance of pcelsPolicySet, pcelsPolicySetAssociation realizes a PolicySetComponent association [PCIM_EXT]. When subordinated to an instance of dlm1System [CIM_LDAP], pcelsPolicySetAssociation realizes a PolicySetInSystem association [PCIM_EXT].

The pcelsPolicySetAssociation class is a structural object class and it is derived from the pcimPolicy class [PCLS].

The aggregation of a reusable pcelsPolicySet instance is realized via the pcelsPolicySetDN attribute. A non-reusable pcelsPolicySet instance is attached (as auxiliary subclass of pcelsPolicySet) directly to the pcelsPolicySetAssociation entry.

When reading a pcelsPolicySetAssociation instance that has a pcelsPolicySet attached, the attribute pcelsPolicySetDN MUST be ignored. Applications SHOULD remove the pcelsPolicySetDN value from a pcelsPolicySetAssociation upon attachment of a pcelsPolicySet to the entry.

The pcelsPolicySetAssociation class is defined as follows:

```
( 1.3.6.1.1.9.1.2
  NAME 'pcelsPolicySetAssociation'
  DESC 'Associates a policy set to an aggregating entry'
  SUP pcimPolicy
  STRUCTURAL
  MUST ( pcelsPriority )
  MAY ( pcelsPolicySetName
      $ pcelsPolicySetDN )
)
```

The pcelsPriority attribute type indicates the priority of a policy set component. This attribute type is of syntax Integer [LDAP_SYNTAX]. It has an equality matching rule of integerMatch [LDAP_SYNTAX] and an ordering matching rule of integerOrderingMatch [LDAP_MATCH]. Attributes of this type can only have single values. The only allowed values for attributes of this type are non-negative integers. Within the set of pcelsPolicySetAssociation entries directly subordinated to a pcelsPolicySet or a dlm1System [CIM_LDAP], the pcelsPriority values MUST be unique.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.4
  NAME ’pcelsPriority’
  DESC ’Priority of a component’
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
```

The pcelsPolicySetDN attribute type is used in the aggregation of PolicySet instances [PCIM_EXT]. This attribute type is of syntax DN [LDAP_SYNTAX]. It has an equality matching rule of distinguishedNameMatch [LDAP_SYNTAX]. Attributes of this type can only have single values. The only allowed values for pcelsPolicySetDN attributes are DNs of pcelsPolicySet entries.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.5
  NAME ’pcelsPolicySetDN’
  DESC ’DN of a pcelsPolicySet entry’
  EQUALITY distinguishedNameMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
  SINGLE-VALUE
)
```

5.3. The Three Policy Group Classes

The pcelsGroup class is the base class for representing a policy group. It is mapped from the modified PolicyGroup class [PCIM_EXT]. The pcelsGroup class is derived from the pcelsPolicySet class. To maximize flexibility, the pcelsGroup class is defined as abstract. An auxiliary subclass pcelsGroupAuxClass enables the attachment of a policy group to an existing entry, while a structural subclass pcelsGroupInstance permits the representation of a policy group as a standalone entry.

The pcelsGroup class is defined as follows:

```
( 1.3.6.1.1.9.1.3
  NAME ’pcelsGroup’
  DESC ’Base class for representing a policy group’
  SUP pcelsPolicySet
  ABSTRACT
  MAY ( pcimGroupName )
)
```

The pcelsGroupAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.4
  NAME 'pcelsGroupAuxClass'
  DESC 'Auxiliary class for representing a policy group'
  SUP pcelsGroup
  AUXILIARY
)
```

The pcelsGroupInstance class is defined as follows:

```
( 1.3.6.1.1.9.1.5
  NAME 'pcelsGroupInstance'
  DESC 'Structural class for representing a policy group'
  SUP pcelsGroup
  STRUCTURAL
)
```

The pcimGroupName attribute type used by the pcelsGroup class is defined in the section 5.2 of [PCLS]. In the pcelsGroup object class, this attribute preserves its syntax and semantics as defined by [PCLS] and [PCIM].

Note: PCELS implementations SHOULD support pcelsGroup and its two subclasses and MAY also support pcimGroup and its two subclasses [PCLS]. Applications that choose to support pcelsGroup and its two subclasses MUST use the aggregation mechanism provided by pcelsPolicySetAssociation for aggregating policy groups or policy rules in policy groups represented as instances of pcelsGroup.

5.4. The Three Policy Rule Classes

The pcelsRule class is the base class for representing a policy rule. It is mapped from the modified PolicyRule class [PCIM_EXT]. The pcelsRule class is derived from the pcelsPolicySet class. To maximize flexibility, the pcelsRule class is defined as abstract. An auxiliary subclass pcelsRuleAuxClass enables the attachment of a policy rule to an existing entry, while a structural subclass pcelsRuleInstance permits the representation of a policy rule as a standalone entry.

When reading a pcelsRule instance that has a pcimConditionAuxClass attached, from the policy rule perspective the attribute pcelsConditionList MUST be ignored. For example, if present, the attribute MUST NOT be considered an association between this policy rule and a policy condition. Such situations may occur, for example, when a pcelsCompoundConditionAuxClass is attached to a pcelsRule instance.

When reading a pcelsRule instance that has a pcimActionAuxClass attached, from the policy rule perspective the attribute pcelsActionList MUST be ignored. For example, if present, the attribute MUST NOT be considered an association between this policy rule and a policy action. Such situations may occur, for example, when a pcelsCompoundActionAuxClass is attached to a pcelsRule instance.

The pcelsRule class is defined as follows:

```
( 1.3.6.1.1.9.1.6
  NAME 'pcelsRule'
  DESC 'Base class for representing a policy rule'
  SUP pcelsPolicySet
  ABSTRACT
  MAY ( pcimRuleName
      $ pcimRuleEnabled
      $ pcimRuleUsage
      $ pcimRuleMandatory
      $ pcelsRuleValidityPeriodList
      $ pcelsConditionListType
      $ pcelsConditionList
      $ pcelsActionList
      $ pcelsSequencedActions
      $ pcelsExecutionStrategy )
)
```

The pcelsRuleAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.7
  NAME 'pcelsRuleAuxClass'
  DESC 'Auxiliary class for representing a policy rule'
  SUP pcelsRule
  AUXILIARY
)
```

The pcelsRuleInstance class is defined as follows:

```
( 1.3.6.1.1.9.1.8
  NAME 'pcelsRuleInstance'
  DESC 'Structural class for representing a policy rule'
  SUP pcelsRule
  STRUCTURAL
)
```

Four of the attributes used by the pcelsRule class are defined in the section 5.3 of [PCLS]. These attributes are: pcimRuleName, pcimRuleEnabled, pcimRuleUsage and pcimRuleMandatory. In the pcelsRule object class, these attributes preserve their syntax and semantics as defined by [PCLS] and [PCIM].

The attributes pcimRuleValidityPeriodList, pcimRuleConditionListType, pcimRuleConditionList, pcimRuleActionList and pcimRuleSequencedActions defined in [PCLS] are not used by pcelsRule. Instead, this class uses the new attributes pcelsRuleValidityPeriodList, pcelsConditionListType, pcelsConditionList, pcelsActionList and pcelsSequencedActions. Except for pcelsRuleValidityPeriodList, the new attributes are also used for similar purpose by either pcelsCompoundConditionAuxClass or pcelsCompoundActionAuxClass.

The pcelsRuleValidityPeriodList attribute type is used in the realization of the PolicyRuleValidityPeriod association ([PCIM_EXT] and [PCIM]). This attribute type is of syntax DN [LDAP_SYNTAX]. It has an equality matching rule of distinguishedNameMatch [LDAP_SYNTAX]. Attributes of this type can have multiple values. The only allowed values for pcelsRuleValidityPeriodList attributes are DNs of pcimRuleValidityAssociation entries. In a pcelsRule, the pcelsRuleValidityPeriodList attribute represents the associations between this policy rule and its time period conditions.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.62
  NAME 'pcelsRuleValidityPeriodList'
  DESC 'Unordered set of DNs of pcimRuleValidityAssociation entries'
  EQUALITY distinguishedNameMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
)
```

The pcelsConditionListType attribute type indicates whether the set of aggregated conditions is in disjunctive or conjunctive normal form. It is mapped from the PolicyRule.ConditionListType property [PCIM] (identical to the CompoundPolicyCondition.ConditionListType property defined in [PCIM_EXT]). This attribute type is of syntax Integer [LDAP_SYNTAX]. It has an equality matching rule of integerMatch [LDAP_SYNTAX] and an ordering matching rule of integerOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are 1 (Disjunctive) and 2 (Conjunctive). If this attribute is missing from a pcelsRule instance, applications MUST assume that the set of aggregated conditions is in disjunctive normal form.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.6
  NAME 'pcelsConditionListType'
  DESC 'Indicates the type of condition aggregation'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
```

The pcelsConditionList attribute type is used in the realization of the PolicyConditionStructure association [PCIM_EXT]. This attribute type is of syntax DN [LDAP_SYNTAX]. It has an equality matching rule of distinguishedNameMatch [LDAP_SYNTAX]. Attributes of this type can have multiple values. The only allowed values for pcelsConditionList attributes are DNs of pcelsConditionAssociation entries. In a pcelsRule, the pcelsConditionList attribute represents the associations between this policy rule and its conditions.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.7
  NAME 'pcelsConditionList'
  DESC 'Unordered set of DNs of pcelsConditionAssociation entries'
  EQUALITY distinguishedNameMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
)
```

The pcelsActionList attribute type is used in the realization of the PolicyActionStructure association [PCIM_EXT]. This attribute type is of syntax DN [LDAP_SYNTAX]. It has an equality matching rule of distinguishedNameMatch [LDAP_SYNTAX]. Attributes of this type can have multiple values. The only allowed values for pcelsActionList attributes are DNs of pcelsActionAssociation entries. In a pcelsRule, the pcelsActionList attribute represents the associations between this policy rule and its actions.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.8
  NAME 'pcelsActionList'
  DESC 'Unordered set of DNs of pcelsActionAssociation entries'
  EQUALITY distinguishedNameMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
)
```

The pcelsSequencedActions attribute type indicates whether the ordered execution of actions in an aggregate is Mandatory, Recommended or DontCare. It is mapped from the PolicyRule.SequencedActions property [PCIM] (identical to the CompoundPolicyAction.SequencedActions property defined in [PCIM_EXT]). This attribute type is of syntax Integer [LDAP_SYNTAX]. It has an equality matching rule of integerMatch [LDAP_SYNTAX] and an ordering matching rule of integerOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are 1 (Mandatory), 2 (Recommended) and 3 (DontCare). If this attribute is missing from a pcelsRule instance, applications MUST assume that the ordered execution of actions in this rule is not important (DontCare).

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.9
  NAME 'pcelsSequencedActions'
  DESC 'Indicates the importance of action sequencing'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
```

The pcelsExecutionStrategy attribute type indicates whether the actions in an aggregate are to be executed until success, all (independent of their outcome) or until failure. It is mapped from the PolicyRule.ExecutionStrategy property [PCIM_EXT] (identical to the CompoundPolicyAction.ExecutionStrategy property). This attribute type is of syntax Integer [LDAP_SYNTAX]. It has an equality matching rule of integerMatch [LDAP_SYNTAX] and an ordering matching rule of integerOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are 1 (Do until success), 2 (Do all) and 3 (Do until failure). If this attribute is missing from a pcelsRule instance, applications MUST assume that all the actions are to be executed (Do all).

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.10
  NAME 'pcelsExecutionStrategy'
  DESC 'Indicates the action execution strategy'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
```

Note 1: Rule validity periods for an instance of pcelsRule are realized using the attribute pcelsRuleValidityPeriodList and pcimRuleValidityAssociation [PCLS] entries subordinated to the rule.

If DIT structure rules and name forms are written for a PCELS implementation (as suggested in section 5.5 of [PCLS]), they would require that an instance of the pcimRuleValidityAssociation class have as its superior an instance of the pcelsRule class or, if applicable, an instance of the pcimRule class. Any structure rules and name forms that require an instance of the pcimRuleValidityAssociation class to have as its superior only an instance of the pcimRule class, are in conflict and MUST be removed.

Note 2: PCELS implementations SHOULD support pcelsRule and its two subclasses and MAY also support pcimRule and its two subclasses [PCLS]. Applications that choose to support pcelsRule and its two subclasses MUST use the aggregation mechanism provided by pcelsPolicySetAssociation for aggregating policy groups or policy rules in policy rules represented as instances of pcelsRule.

5.5. The Structural Class pcelsConditionAssociation

The pcelsConditionAssociation class is used in the aggregation of PolicyCondition instances [PCIM]. pcelsConditionAssociation entries are always subordinated to the aggregating entry. When subordinated to an instance of pcelsRule, the pcelsConditionAssociation entry realizes the PolicyConditionInPolicyRule association [PCIM_EXT]. When subordinated to an instance of pcelsCompoundConditionAuxClass, the pcelsConditionAssociation entry realizes the PolicyConditionInPolicyCondition association [PCIM_EXT].

The pcelsConditionAssociation class is a structural object class and it is derived from the pcimRuleConditionAssociation class [PCLS].

The aggregation of a reusable instance of pcimConditionAuxClass is realized via the pcimConditionDN attribute. A non-reusable instance of pcimConditionAuxClass is attached directly to the pcelsConditionAssociation entry.

When reading a pcelsConditionAssociation entry that has a pcimConditionAuxClass instance attached, the attribute pcimConditionDN MUST be ignored. Applications SHOULD remove the pcimConditionDN value from a pcelsConditionAssociation upon attachment of a pcimConditionAuxClass to the entry.

The pcelsConditionAssociation class is defined as follows:

```
( 1.3.6.1.1.9.1.9
  NAME 'pcelsConditionAssociation'
  DESC 'Associates a policy conditions to an aggregating entry'
  SUP pcimRuleConditionAssociation
  STRUCTURAL
)
```

This class extends the semantics of the pcimRuleConditionAssociation object class without using any new attributes. All its attributes are inherited from the pcimRuleConditionAssociation that is defined in section 5.4 of [PCLS].

5.6. The Structural Class pcelsActionAssociation

The pcelsActionAssociation class is used in the aggregation of PolicyAction instances [PCIM]. pcelsActionAssociation entries are always subordinated to the aggregating entry. When subordinated to a pcelsRule instance, the pcelsActionAssociation entry realizes the PolicyActionInPolicyRule association [PCIM_EXT]. When subordinated to an instance of pcelsCompoundActionAuxClass, the pcelsActionAssociation entry realizes the PolicyActionInPolicyAction association [PCIM_EXT].

The pcelsActionAssociation class is a structural object class and it is derived from the pcimRuleActionAssociation class [PCLS].

The aggregation of a reusable instance of pcimActionAuxClass is realized via the pcimActionDN attribute. A non-reusable instance of pcimActionAuxClass is attached directly to the pcelsActionAssociation entry.

When reading a pcelsActionAssociation entry that has a pcimActionAuxClass instance attached, the attribute pcimActionDN MUST be ignored. Applications SHOULD remove the pcimActionDN value from a pcelsActionAssociation upon attachment of a pcimActionAuxClass to the entry.

The pcelsActionAssociation class is defined as follows:

```
( 1.3.6.1.1.9.1.10
  NAME 'pcelsActionAssociation'
  DESC 'Associates a policy conditions to an aggregating entry'
  SUP pcimRuleActionAssociation
  STRUCTURAL
)
```

This class extends the semantics of the pcimRuleActionAssociation object class without using any new attributes. All its attributes are inherited from the pcimRuleActionAssociation that is defined in section 5.6 of [PCLS].

5.7. The Auxiliary Class pcelsSimpleConditionAuxClass

The pcelsSimpleConditionAuxClass class implements a Value matching condition for a Variable. It is mapped from the SimplePolicyCondition class [PCIM_EXT]. The pcelsSimpleConditionAuxClass class is an auxiliary object class and it is derived from the pcimConditionAuxClass class [PCLS].

A reusable variable/value is associated to a pcelsSimpleConditionAuxClass via the pcelsVariableDN/pcelsValueDN reference from the simple condition instance. A non-reusable variable/value is associated directly as auxiliary object class to the same entry as the pcelsSimpleConditionAuxClass instance.

When reading a pcelsSimpleConditionAuxClass instance that has an instance of pcelsVariable attached, the attribute pcelsVariableDN MUST be ignored. Applications SHOULD remove the pcelsVariableDN value from a pcelsSimpleConditionAuxClass instance upon attachment of a pcelsVariable instance to the same entry.

When reading a pcelsSimpleConditionAuxClass instance that has an instance of pcelsValue attached, the attribute pcelsValueDN MUST be ignored. Applications SHOULD remove the pcelsValueDN value from a pcelsSimpleConditionAuxClass instance upon attachment of a pcelsValue instance to the same entry.

The pcelsSimpleConditionAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.11
  NAME 'pcelsSimpleConditionAuxClass'
  DESC 'Value matching condition for a policy variable'
  SUP pcimConditionAuxClass
  AUXILIARY
  MAY ( pcelsVariableDN
      $ pcelsValueDN )
)
```

The pcelsVariableDN attribute type realizes the PolicyVariableInSimplePolicyCondition association [PCIM_EXT]. This attribute type is of syntax DN [LDAP_SYNTAX]. It has an equality matching rule of distinguishedNameMatch [LDAP_SYNTAX]. Attributes of this type can only have a single value. The only allowed values for pcelsVariableDN attributes are DNs of pcelsVariable entries. In a pcelsSimpleConditionAuxClass, the pcelsVariableDN attribute represents the association between this simple policy condition and its policy variable.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.11
  NAME 'pcelsVariableDN'
  DESC 'DN of a pcelsVariable entry'
  EQUALITY distinguishedNameMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
  SINGLE-VALUE
)
```

The pcelsValueDN attribute type realizes the PolicyValueInSimplePolicyCondition association [PCIM_EXT]. This attribute type is of syntax DN [LDAP_SYNTAX]. It has an equality matching rule of distinguishedNameMatch [LDAP_SYNTAX]. Attributes of this type can only have a single value. The only allowed values for pcelsValueDN attributes are DNs of pcelsValueAuxClass entries. In a pcelsSimpleConditionAuxClass, the pcelsValueDN attribute represents the association between this simple policy condition and its policy value.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.12
  NAME 'pcelsValueDN'
  DESC 'DN of a pcelsValueAuxClass entry'
  EQUALITY distinguishedNameMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
  SINGLE-VALUE
)
```

Note: An instance of pcelsSimpleActionAuxClass and an instance of pcelsSimpleConditionAuxClass MUST NOT be attached to the same entry. Because the two classes use the same mechanisms to associate Variables and Values, this restriction is necessary in order to avoid ambiguities.

5.8. The Auxiliary Class pcelsCompoundConditionAuxClass

The pcelsCompoundConditionAuxClass class represents a compound policy condition formed by the aggregation of other policy conditions. It is mapped from the CompoundPolicyCondition class [PCIM_EXT]. The pcelsCompoundConditionAuxClass class is an auxiliary object class and it is derived from the pcimConditionAuxClass class [PCLS].

The pcelsCompoundConditionAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.12
  NAME 'pcelsCompoundConditionAuxClass'
  DESC 'Boolean combination of simpler conditions'
  SUP pcimConditionAuxClass
  AUXILIARY
  MAY ( pcelsConditionListType
      $ pcelsConditionList )
)
```

If the pcelsConditionListType attribute is missing from a pcelsCompoundConditionAuxClass instance, applications MUST assume that the set of aggregated conditions is in disjunctive normal form.

In a pcelsCompoundConditionAuxClass instance, the pcelsConditionList attribute represents the associations between this compound policy condition and the compounded conditions.

These attribute types are defined in section 5.4.

Like pcelsRule, instances of pcelsCompoundConditionAuxClass use pcelsConditionList values and subordinated pcelsConditionAssociation entries to aggregate policy conditions.

5.9. The Auxiliary Class pcelsCompoundFilterConditionAuxClass

The pcelsCompoundFilterConditionAuxClass class represents a domain-level filter. It is mapped from the CompoundFilterCondition class [PCIM_EXT]. The pcelsCompoundFilterConditionAuxClass class is an auxiliary object class and it is derived from the pcelsCompoundConditionAuxClass class.

The pcelsCompoundFilterConditionAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.13
  NAME 'pcelsCompoundFilterConditionAuxClass'
  DESC 'A compound condition with mirroring capabilities'
  SUP pcelsCompoundConditionAuxClass
  AUXILIARY
  MAY ( pcelsIsMirrored )
)
```

The pcelsIsMirrored attribute type indicates whether the traffic that mirrors the specified filter is to be treated as matching the filter. It is mapped from the CompoundFilterCondition.IsMirrored property [PCIM_EXT]. This attribute type is of syntax Boolean [LDAP_SYNTAX]. It has an equality matching rule of booleanMatch [LDAP_MATCH].

Attributes of this type can only have a single value. If this attribute is missing from a pcelsCompoundFilterConditionAuxClass instance, applications MUST assume that the filter is not mirrored.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.13
  NAME 'pcelsIsMirrored'
  DESC 'Indicates whether the mirrored traffic matches'
  EQUALITY booleanMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
  SINGLE-VALUE
)
```

5.10. The Auxiliary Class pcelsSimpleActionAuxClass

The pcelsSimpleActionAuxClass class implements the action of assigning a Value to a Variable. It is mapped from the SimplePolicyAction class [PCIM_EXT]. The pcelsSimpleActionAuxClass class is an auxiliary object class and it is derived from the pcimActionAuxClass class [PCLS].

A reusable variable/value is associated to a pcelsSimpleActionAuxClass via the pcelsVariableDN/pcelsValueDN reference from the simple action instance. A non-reusable variable/value is associated directly as auxiliary object class to the same entry as the pcelsSimpleActionAuxClass instance.

When reading a pcelsSimpleActionAuxClass instance that has an instance of pcelsVariable attached, the attribute pcelsVariableDN MUST be ignored. Applications SHOULD remove the pcelsVariableDN value from a pcelsSimpleActionAuxClass instance upon attachment of a pcelsVariable instance to the same entry.

When reading a pcelsSimpleActionAuxClass instance that has an instance of pcelsValue attached, the attribute pcelsValueDN MUST be ignored. Applications SHOULD remove the pcelsValueDN value from a pcelsSimpleActionAuxClass instance upon attachment of a pcelsValue instance to the same entry.

The pcelsSimpleActionAuxClass class is defined as follows:

```
   ( 1.3.6.1.1.9.1.14
     NAME 'pcelsSimpleActionAuxClass'
     DESC 'Value assignment action for a policy variable'
     SUP pcimActionAuxClass
     AUXILIARY
     MAY ( pcelsVariableDN
         $ pcelsValueDN )
   )
```

In a pcelsSimpleActionAuxClass, the pcelsVariableDN attribute represents the association between this simple policy action and its policy variable. It realizes the PolicyVariableInSimplePolicyAction association [PCIM_EXT].

In a pcelsSimpleActionAuxClass, the pcelsValueDN attribute represents the association between this simple policy action and its policy value. It realizes the PolicyValueInSimplePolicyAction association [PCIM_EXT].

These attributes are defined in section 5.7.

Note: An instance of pcelsSimpleActionAuxClass and an instance of pcelsSimpleConditionAuxClass MUST NOT be attached to the same entry. Because the two classes use the same mechanisms to associate Variables and Values, this restriction is necessary in order to avoid ambiguities.

5.11. The Auxiliary Class pcelsCompoundActionAuxClass

The pcelsCompoundActionAuxClass class represents a compound policy action formed by the aggregation of other policy actions. It is mapped from the CompoundPolicyCondition class [PCIM_EXT]. The pcelsCompoundActionAuxClass class is an auxiliary object class and it is derived from the pcimActionAuxClass class [PCLS].

The pcelsCompoundActionAuxClass class is defined as follows:

```
   ( 1.3.6.1.1.9.1.15
     NAME 'pcelsCompoundActionAuxClass'
     DESC 'Sequence of actions with specific execution strategy'
     SUP pcimActionAuxClass
     AUXILIARY
     MAY ( pcelsActionList
         $ pcelsSequencedActions
         $ pcelsExecutionStrategy )
   )
```

In a pcelsCompoundActionAuxClass instance, the pcelsActionList attribute represents the associations between this policy rule and its actions.

If the pcelsSequencedActions attribute is missing from a pcelsCompoundActionAuxClass instance, applications MUST assume that the ordered execution of actions in this compound policy action is not important (DontCare).

If the pcelsExecutionStrategy attribute is missing from a pcelsCompoundActionAuxClass instance, applications MUST assume that all the actions are to be executed (Do all).

These attribute types are defined in section 5.4.

Like pcelsRule, instances of pcelsCompoundActionAuxClass use pcelsActionList values and subordinated pcelsActionAssociation entries to aggregate policy actions.

5.12. The Abstract Class pcelsVariable

The pcelsVariable class is mapped from the PolicyVariable class [PCIM_EXT]. The pcelsVariable is an abstract object class and it is derived directly from the 'top' object class [LDAP_SCHEMA].

A pcelsVariable instance may be associated to a set of pcelsValueAuxClass instances that represent its expected values. The expected values for a variable may be indicated by:

(1) pcelsExpectedValueList references to reusable instances of pcelsValueAuxClass, or
(2) pcelsExpectedValueList references to subordinated non-reusable instances of pcelsValueAuxClass

The pcelsVariable class is defined as follows:

```
( 1.3.6.1.1.9.1.16
  NAME 'pcelsVariable'
  DESC 'Base class for representing a policy variable'
  SUP top
  ABSTRACT
  MAY ( pcelsVariableName
      $ pcelsExpectedValueList )
)
```

The pcelsVariableName attribute type may be used as naming attribute for pcelsVariable entries. This attribute type is of syntax Directory String [LDAP_SYNTAX]. It has an equality matching rule of caseIgnoreMatch, an ordering matching rule of caseIgnoreOrderingMatch and a substrings matching rule of caseIgnoreSubstringsMatch [LDAP_SYNTAX]. Attributes of this type can only have a single value.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.14
  NAME 'pcelsVariableName'
  DESC 'The user-friendly name of a variable.'
  EQUALITY caseIgnoreMatch
  ORDERING caseIgnoreOrderingMatch
  SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)
```

The pcelsExpectedValueList attribute type realizes the ExpectedPolicyValuesForVariable association [PCIM_EXT]. This attribute type is of syntax DN [LDAP_SYNTAX]. It has an equality matching rule of distinguishedNameMatch [LDAP_SYNTAX]. Attributes of this type can have multiple values. The only allowed values for pcelsExpectedValueList attributes are DNs of pcelsValueAuxClass entries. In a pcelsVariable, the pcelsExpectedValueList attribute represents the associations between this policy variable and its expected values.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.15
  NAME 'pcelsExpectedValueList'
  DESC 'Unordered set of DNs of pcelsValueAuxClass entries
        representing expected values for a policy variable'
  EQUALITY distinguishedNameMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
)
```

5.13. The Auxiliary Class pcelsExplicitVariableAuxClass

The pcelsExplicitVariableAuxClass class is mapped from the PolicyExplicitVariable class [PCIM_EXT]. The pcelsExplicitVariableAuxClass is an auxiliary object class and it is derived from the pcelsVariable class.

The pcelsExplicitVariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.17
  NAME 'pcelsExplicitVariableAuxClass'
  DESC 'Explicitly defined policy variable'
  SUP pcelsVariable
  AUXILIARY
  MUST ( pcelsVariableModelClass
       $ pcelsVariableModelProperty )
)
```

The pcelsVariableModelClass attribute type identifies a [CIM] class whose property is evaluated or set as a variable. It is mapped from the PolicyExplicitVariable.ModelClass property [PCIM_EXT]. This attribute type is of syntax Directory String [LDAP_SYNTAX]. It has an equality matching rule of caseIgnoreMatch [LDAP_SYNTAX]. Attributes of this type can only have a single value.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.16
  NAME 'pcelsVariableModelClass'
  DESC 'Identifies a CIM class'
  EQUALITY caseIgnoreMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)
```

The pcelsVariableModelProperty attribute type identifies the attribute of a [CIM] class, which is evaluated or set as a variable. It is mapped from the PolicyExplicitVariable.ModelProperty property [PCIM_EXT]. This attribute type is of syntax Directory String [LDAP_SYNTAX]. It has an equality matching rule of caseIgnoreMatch [LDAP_SYNTAX]. Attributes of this type can only have a single value.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.17
  NAME 'pcelsVariableModelProperty'
  DESC 'Identifies the property of a CIM class.'
  EQUALITY caseIgnoreMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)
```

5.14. The Auxiliary Class pcelsImplicitVariableAuxClass

The pcelsImplicitVariableAuxClass class is mapped from the PolicyImplicitVariable class [PCIM_EXT]. The pcelsImplicitVariableAuxClass is an auxiliary object class and it is derived from the pcelsVariable class.

The pcelsImplicitVariableAuxClass class does not represent actual variables; these are introduced by its subclasses. pcelsImplicitVariableAuxClass introduces the semantics of being an implicitly defined policy variable and these semantics are inherited by all its subclasses. These semantics include those inherited from pcelsVariable that possibly represent either rule-specific or reusable policy variables.

In order to preserve the ability to represent rule-specific or reusable variables, all the subclasses of pcelsImplicitVariableAuxClass MUST also be auxiliary classes.

The pcelsImplicitVariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.18
  NAME 'pcelsImplicitVariableAuxClass'
  DESC 'Implicitly defined policy variable'
  SUP pcelsVariable
  AUXILIARY
  MAY ( pcelsExpectedValueTypes )
)
```

The pcelsExpectedValueTypes attribute type represents the set of policy value types that may be used with this policy variable. It is mapped from the PolicyImplicitVariable.ValueTypes property [PCIM_EXT]. This attribute type is of syntax Directory String [LDAP_SYNTAX]. It has an equality matching rule of caseIgnoreMatch [LDAP_SYNTAX]. Attributes of this type can have multiple values.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.18
  NAME 'pcelsExpectedValueTypes'
  DESC 'Identifies subclasses of pcelsValueAuxClass by name'
  EQUALITY caseIgnoreMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)
```

5.15. The Subclasses of pcelsImplicitVariableAuxClass

The following classes are derived from the pcelsImplicitVariableAuxClass class. They are mapped from the corresponding subclasses of the PolicyImplicitVariable class [PCIM_EXT]. All the classes defined below are auxiliary object classes.

Each one of the classes defined in this section introduces specific restrictions for the values of the pcelsExpectedValueTypes attribute. If this attribute is missing, applications MUST assume that all allowed value types are expected for the policy variable.

Some of these classes have additional restrictions on the actual values of the associated policy value instances (e.g., only integers in the range 0..65535 must be used with a SourcePort variable). The association between a pcelsImplicitVariableAuxClass instance and a pcelsValueAuxClass instance that contains values outside the valid range or set for that variable SHOULD be considered invalid. The entry that realizes such association SHOULD be treated as invalid and the policy rules or groups that refer to it SHOULD be treated as being disabled, meaning that the execution of such policy rules or groups SHOULD be stopped.

The pcelsSourceIPv4VariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.19
  NAME 'pcelsSourceIPv4VariableAuxClass'
  DESC 'Source IP v4 address'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsSourceIPv4VariableAuxClass instance, the only allowed value for the pcelsExpectedValueTypes attribute is 'pcelsIPv4AddrValueAuxClass'.

The pcelsSourceIPv6VariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.20
  NAME 'pcelsSourceIPv6VariableAuxClass'
  DESC 'Source IP v6 address'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsSourceIPv6VariableAuxClass instance, the only allowed value for the pcelsExpectedValueTypes attribute is 'pcelsIPv6AddrValueAuxClass'.

The pcelsDestinationIPv4VariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.21
  NAME 'pcelsDestinationIPv4VariableAuxClass'
  DESC 'Destination IP v4 address'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsDestinationIPv4VariableAuxClass instance, the only allowed value for the pcelsExpectedValueTypes attribute is 'pcelsIPv4AddrValueAuxClass'.

The pcelsDestinationIPv6VariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.22
  NAME 'pcelsDestinationIPv6VariableAuxClass'
  DESC 'Destination IP v6 address'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsDestinationIPv6VariableAuxClass instance, the only allowed value for the pcelsExpectedValueTypes attribute is 'pcelsIPv6AddrValueAuxClass'.

The pcelsSourcePortVariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.23
  NAME 'pcelsSourcePortVariableAuxClass'
  DESC 'Source port'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsSourcePortVariableAuxClass instance, the only allowed value for the pcelsExpectedValueTypes attribute is 'pcelsIntegerValueAuxClass'. Additionally, only policy values that represent integers in the range 0..65535 (inclusive) SHOULD be used with pcelsSourcePortVariableAuxClass instances.

The pcelsDestinationPortVariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.24
  NAME 'pcelsDestinationPortVariableAuxClass'
  DESC 'Destination port'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsDestinationPortVariableAuxClass instance, the only allowed value for the pcelsExpectedValueTypes attribute is 'pcelsIntegerValueAuxClass'. Additionally, only policy values that represent integers in the range 0..65535 (inclusive) SHOULD be used with pcelsDestinationPortVariableAuxClass instances.

The pcelsIPProtocolVariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.25
  NAME 'pcelsIPProtocolVariableAuxClass'
  DESC 'IP protocol number'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsIPProtocolVariableAuxClass instance, the only allowed value for the pcelsExpectedValueTypes attribute is 'pcelsIntegerValueAuxClass'. Additionally, only policy values that represent integers in the range 0..255 (inclusive) SHOULD be used with pcelsIPProtocolVariableAuxClass instances.

The pcelsIPVersionVariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.26
  NAME 'pcelsIPVersionVariableAuxClass'
  DESC 'IP version number'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsIPVersionVariableAuxClass instance, the only allowed value for the pcelsExpectedValueTypes attribute is 'pcelsIntegerValueAuxClass'. Additionally, only policy values that represent integers in the range 0..15 (inclusive) SHOULD be used with pcelsIPVersionVariableAuxClass instances.

The pcelsIPToSVariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.27
  NAME 'pcelsIPToSVariableAuxClass'
  DESC 'IP ToS octet'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsIPToSVariableAuxClass instance, the only allowed values for the pcelsExpectedValueTypes attribute are 'pcelsIntegerValueAuxClass' and 'pcelsBitStringValueAuxClass'. Additionally, only policy values that represent integers in the range 0..255 (inclusive) or 8-bit bitStrings SHOULD be used with pcelsIPToSVariableAuxClass instances.

The pcelsDSCPVariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.28
  NAME 'pcelsDSCPVariableAuxClass'
  DESC 'DiffServ code point'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsDSCPVariableAuxClass instance, the only allowed values for the pcelsExpectedValueTypes attribute are 'pcelsIntegerValueAuxClass' and 'pcelsBitStringValueAuxClass'. Additionally, only policy values that represent integers in the range 0..63 (inclusive) or 6-bit bitStrings SHOULD be used with pcelsDSCPVariableAuxClass instances.

The pcelsFlowIdVariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.29
  NAME 'pcelsFlowIdVariableAuxClass'
  DESC 'Flow Identifier'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsFlowIdVariableAuxClass instance, the only allowed values for the pcelsExpectedValueTypes attribute are 'pcelsIntegerValueAuxClass' and 'pcelsBitStringValueAuxClass'. Additionally, only policy values that represent integers in the range 0..1048575 (inclusive) or 20-bit bitStrings SHOULD be used with pcelsFlowIdVariableAuxClass instances.

The pcelsSourceMACVariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.30
  NAME 'pcelsSourceMACVariableAuxClass'
  DESC 'Source MAC address'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsSourceMACVariableAuxClass instance, the only allowed value for the pcelsExpectedValueTypes attribute is 'pcelsMACAddrValueAuxClass'.

The pcelsDestinationMACVariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.31
  NAME 'pcelsDestinationMACVariableAuxClass'
  DESC 'Destination MAC address'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsDestinationMACVariableAuxClass instance, the only allowed value for the pcelsExpectedValueTypes attribute is 'pcelsMACAddrValueAuxClass'.

The pcelsVLANVariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.32
  NAME 'pcelsVLANVariableAuxClass'
  DESC 'VLAN'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsVLANVariableAuxClass instance, the only allowed values for the pcelsExpectedValueTypes attribute are 'pcelsIntegerValueAuxClass' and 'pcelsBitStringValueAuxClass'. Additionally, only policy values that represent integers in the range 0..4095 (inclusive) or 12-bit bitStrings SHOULD be used with pcelsVLANVariableAuxClass instances.

The pcelsCoSVariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.33
  NAME 'pcelsCoSVariableAuxClass'
  DESC 'Class of service'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsCoSVariableAuxClass instance, the only allowed values for the pcelsExpectedValueTypes attribute are 'pcelsIntegerValueAuxClass' and 'pcelsBitStringValueAuxClass'. Additionally, only policy values that represent integers in the range 0..7 (inclusive) or 3-bit bitStrings SHOULD be used with pcelsCoSVariableAuxClass instances.

The pcelsEthertypeVariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.34
  NAME 'pcelsEthertypeVariableAuxClass'
  DESC 'Ethertype'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsEthertypeVariableAuxClass instance, the only allowed values for the pcelsExpectedValueTypes attribute are 'pcelsIntegerValueAuxClass' and 'pcelsBitStringValueAuxClass'. Additionally, only policy values that represent integers in the range 0..65535 (inclusive) or 16-bit bitStrings SHOULD be used with pcelsEthertypeVariableAuxClass instances.

The pcelsSourceSAPVariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.35
  NAME 'pcelsSourceSAPVariableAuxClass'
  DESC 'Source SAP'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsSourceSAPVariableAuxClass instance, the only allowed values for the pcelsExpectedValueTypes attribute are 'pcelsIntegerValueAuxClass' and 'pcelsBitStringValueAuxClass'. Additionally, only policy values that represent integers in the range 0..255 (inclusive) or 8-bit bitStrings SHOULD be used with pcelsSourceSAPVariableAuxClass instances.

The pcelsDestinationSAPVariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.36
  NAME 'pcelsDestinationSAPVariableAuxClass'
  DESC 'Destination SAP'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsDestinationSAPVariableAuxClass instance, the only allowed values for the pcelsExpectedValueTypes attribute are 'pcelsIntegerValueAuxClass' and 'pcelsBitStringValueAuxClass'. Additionally, only policy values that represent integers in the range 0..255 (inclusive) or 8-bit bitStrings SHOULD be used with pcelsDestinationSAPVariableAuxClass instances.

The pcelsSNAPOUIVariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.37
  NAME 'pcelsSNAPOUIVariableAuxClass'
  DESC 'SNAP OUI'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsSNAPOUIVariableAuxClass instance, the only allowed values for the pcelsExpectedValueTypes attribute are 'pcelsIntegerValueAuxClass' and 'pcelsBitStringValueAuxClass'. Additionally, only policy values that represent integers in the range 0..16777215 (inclusive) or 24-bit bitStrings SHOULD be used with pcelsSNAPOUIVariableAuxClass instances.

The pcelsSNAPTypeVariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.38
  NAME 'pcelsSNAPTypeVariableAuxClass'
  DESC 'SNAP type'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsSNAPTypeVariableAuxClass instance, the only allowed values for the pcelsExpectedValueTypes attribute are 'pcelsIntegerValueAuxClass' and 'pcelsBitStringValueAuxClass'. Additionally, only policy values that represent integers in the range 0..65535 (inclusive) or 16-bit bitStrings SHOULD be used with pcelsSNAPTypeVariableAuxClass instances.

The pcelsFlowDirectionVariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.39
  NAME 'pcelsFlowDirectionVariableAuxClass'
  DESC 'Flow direction'
  SUP pcelsImplicitVariableAuxClass
  AUXILIARY
)
```

In a pcelsFlowDirectionVariableAuxClass instance, the only allowed value for the pcelsExpectedValueTypes attribute is 'pcelsStringValueAuxClass'. Additionally, only policy values that represent the strings 'IN' and 'OUT' SHOULD be used with pcelsFlowDirectionVariableAuxClass instances.

5.16. The Auxiliary Class pcelsValueAuxClass

The pcelsValueAuxClass class is the base class for representing a policy value. It is mapped from the PolicyValue class [PCIM_EXT]. The pcelsValueAuxClass is an auxiliary object class and it is derived directly from the 'top' object class [LDAP_SCHEMA].

The pcelsValueAuxClass class does not represent actual values; these are introduced by its subclasses. pcelsValueAuxClass introduces the semantics of being a policy value that are inherited by all its subclasses. Among these semantics are those of representing either rule-specific or reusable policy values.

In order to preserve the ability to represent rule-specific or reusable values, all the subclasses of pcelsValueAuxClass MUST also be auxiliary classes.

The pcelsValueAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.40
  NAME 'pcelsValueAuxClass'
  DESC 'Base class for representing a policy value'
  SUP top
  AUXILIARY
  MAY ( pcelsValueName )
)
```

The pcelsValueName attribute type may be used as naming attribute for pcelsValueAuxClass entries. This attribute type is of syntax Directory String [LDAP_SYNTAX]. It has an equality matching rule of caseIgnoreMatch, an ordering matching rule of caseIgnoreOrderingMatch and a substrings matching rule of caseIgnoreSubstringsMatch [LDAP_SYNTAX]. Attributes of this type can only have a single value.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.19
  NAME 'pcelsValueName'
  DESC 'The user-friendly name of a value'
  EQUALITY caseIgnoreMatch
  ORDERING caseIgnoreOrderingMatch
  SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)
```

5.17. The Subclasses of pcelsValueAuxClass

The following classes are derived from the pcelsValueAuxClass class. They are mapped from the corresponding subclasses of the PolicyValue class [PCIM_EXT]. All the classes defined below are auxiliary object classes.

The pcelsIPv4AddrValueAuxClass class represents a policy value that provides an unordered set of IPv4 addresses, IPv4 address ranges or hosts. It is mapped from the PolicyIPv4AddrValue class [PCIM_EXT].

The pcelsIPv4AddrValueAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.41
  NAME 'pcelsIPv4AddrValueAuxClass'
  DESC 'Provides IPv4 addresses'
  SUP pcelsValueAuxClass
  AUXILIARY
  MUST ( pcelsIPv4AddrList )
)
```

The pcelsIPv4AddrList attribute type represents an unordered set of IPv4 addresses, IPv4 address ranges or hosts. It is mapped from the PolicyIPv4AddrValue.IPv4AddrList property [PCIM_EXT]. This attribute type is of syntax Directory String [LDAP_SYNTAX]. It has an equality matching rule of caseIgnoreMatch, an ordering matching rule of caseIgnoreOrderingMatch and a substrings matching rule of caseIgnoreSubstringsMatch [LDAP_SYNTAX]. Attributes of this type can have multiple values. The only allowed values for attributes of this type are strings conforming to any of the formats defined for the IPv4AddrList property [PCIM_EXT].

This attribute type is defined as follows:

```
   ( 1.3.6.1.1.9.2.20
     NAME 'pcelsIPv4AddrList'
     DESC 'Unordered set of IPv4 addresses, IPv4 address ranges or
           hosts'
     EQUALITY caseIgnoreMatch
     ORDERING caseIgnoreOrderingMatch
     SUBSTR caseIgnoreSubstringsMatch
     SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
   )
```

The pcelsIPv6AddrValueAuxClass class represents a policy value that provides an unordered set of IPv6 addresses, IPv6 address ranges or hosts. It is mapped from the PolicyIPv6AddrValue class [PCIM_EXT].

The pcelsIPv6AddrValueAuxClass class is defined as follows:

```
   ( 1.3.6.1.1.9.1.42
     NAME 'pcelsIPv6AddrValueAuxClass'
     DESC 'Provides IPv6 addresses'
     SUP pcelsValueAuxClass
     AUXILIARY
     MUST ( pcelsIPv6AddrList )
   )
```

The pcelsIPv6AddrList attribute type represents an unordered set of IPv6 addresses, IPv6 address ranges or hosts. It is mapped from the PolicyIPv6AddrValue.IPv6AddrList property [PCIM_EXT]. This attribute type is of syntax Directory String [LDAP_SYNTAX]. It has an equality matching rule of caseIgnoreMatch, an ordering matching rule of caseIgnoreOrderingMatch and a substrings matching rule of caseIgnoreSubstringsMatch [LDAP_SYNTAX]. Attributes of this type can have multiple values. The only allowed values for attributes of this type are strings conforming to any of the formats defined for the IPv6AddrList property [PCIM_EXT].

This attribute type is defined as follows:

```
   ( 1.3.6.1.1.9.2.21
     NAME 'pcelsIPv6AddrList'
     DESC 'Unordered set of IPv6 addresses, IPv6 address ranges or
           hosts'
     EQUALITY caseIgnoreMatch
     ORDERING caseIgnoreOrderingMatch
     SUBSTR caseIgnoreSubstringsMatch
     SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
   )
```

The pcelsMACAddrValueAuxClass class represents a policy value that provides an unordered set of MAC addresses or MAC address ranges. It is mapped from the PolicyMACAddrValue class [PCIM_EXT].

The pcelsMACAddrValueAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.43
  NAME 'pcelsMACAddrValueAuxClass'
  DESC 'Provides MAC addresses'
  SUP pcelsValueAuxClass
  AUXILIARY
  MUST ( pcelsMACAddrList )
)
```

The pcelsMACAddrList attribute type represents an unordered set of MAC addresses or MAC address ranges. It is mapped from the PolicyMACAddrValue.MACAddrList property [PCIM_EXT]. This attribute type is of syntax Directory String [LDAP_SYNTAX]. It has an equality matching rule of caseIgnoreMatch, an ordering matching rule of caseIgnoreOrderingMatch and a substrings matching rule of caseIgnoreSubstringsMatch [LDAP_SYNTAX]. Attributes of this type can have multiple values. The only allowed values for attributes of this type are strings conforming to any of the formats defined for the MACAddrList property [PCIM_EXT].

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.22
  NAME 'pcelsMACAddrList'
  DESC 'Unordered set of MAC addresses or MAC address ranges'
  EQUALITY caseIgnoreMatch
  ORDERING caseIgnoreOrderingMatch
  SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)
```

The pcelsStringValueAuxClass class represents a policy value that provides an unordered set of strings with wildcards. It is mapped from the PolicyStringValue class [PCIM_EXT].

The pcelsStringValueAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.44
  NAME 'pcelsStringValueAuxClass'
  DESC 'Provides string values'
  SUP pcelsValueAuxClass
  AUXILIARY
  MUST ( pcelsStringList )
```

```
   )
```

The pcelsStringList attribute type represents an unordered set of strings with wildcards. It is mapped from the PolicyStringValue.StringList property [PCIM_EXT]. This attribute type is of syntax Directory String [LDAP_SYNTAX]. It has an equality matching rule of caseIgnoreMatch, an ordering matching rule of caseIgnoreOrderingMatch and a substrings matching rule of caseIgnoreSubstringsMatch [LDAP_SYNTAX]. Attributes of this type can have multiple values. The only allowed values for attributes of this type are strings conforming to the format defined for the StringList property [PCIM_EXT].

This attribute type is defined as follows:

```
   ( 1.3.6.1.1.9.2.23
     NAME 'pcelsStringList'
     DESC 'Unordered set of strings with wildcards'
     EQUALITY caseIgnoreMatch
     ORDERING caseIgnoreOrderingMatch
     SUBSTR caseIgnoreSubstringsMatch
     SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
   )
```

The pcelsBitStringValueAuxClass class represents a policy value that provides an unordered set of bit strings or bit string ranges. It is mapped from the PolicyBitStringValue class [PCIM_EXT].

The pcelsBitStringValueAuxClass class is defined as follows:

```
   ( 1.3.6.1.1.9.1.45
     NAME 'pcelsBitStringValueAuxClass'
     DESC 'Provides bit strings'
     SUP pcelsValueAuxClass
     AUXILIARY
     MUST ( pcelsBitStringList )
   )
```

The pcelsBitStringList attribute type represents an unordered set of bit strings or bit string ranges. It is mapped from the PolicyBitStringValue.BitStringList property [PCIM_EXT]. This attribute type is of syntax Directory String [LDAP_SYNTAX]. It has an equality matching rule of caseIgnoreMatch, an ordering matching rule of caseIgnoreOrderingMatch and a substrings matching rule of caseIgnoreSubstringsMatch [LDAP_SYNTAX]. Attributes of this type can have multiple values. The only allowed values for attributes of this type are strings conforming to any of the formats defined for the BitStringList property [PCIM_EXT].

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.24
  NAME 'pcelsBitStringList'
  DESC 'Unordered set of bit strings or bit string ranges'
  EQUALITY caseIgnoreMatch
  ORDERING caseIgnoreOrderingMatch
  SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)
```

The pcelsIntegerValueAuxClass class represents a policy value that provides an unordered set of integers or integer ranges. It is mapped from the PolicyIntegerValue class [PCIM_EXT].

The pcelsIntegerValueAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.46
  NAME 'pcelsIntegerValueAuxClass'
  DESC 'Provides integer values'
  SUP pcelsValueAuxClass
  AUXILIARY
  MUST ( pcelsIntegerList )
)
```

The pcelsIntegerList attribute type represents an unordered set of integers or integer ranges. It is mapped from the PolicyIntegerValue.IntegerList property [PCIM_EXT]. This attribute type is of syntax Directory String [LDAP_SYNTAX]. It has an equality matching rule of caseIgnoreMatch, an ordering matching rule of caseIgnoreOrderingMatch and a substrings matching rule of caseIgnoreSubstringsMatch [LDAP_SYNTAX]. Attributes of this type can have multiple values. The only allowed values for attributes of this type are strings conforming to the format defined for the IntegerList property [PCIM_EXT].

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.25
  NAME 'pcelsIntegerList'
  DESC 'Unordered set of integers or integer ranges'
  EQUALITY caseIgnoreMatch
  ORDERING caseIgnoreOrderingMatch
  SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)
```

The pcelsBooleanValueAuxClass class represents a policy value that provides a boolean. It is mapped from the PolicyIntegerValue class [PCIM_EXT].

The pcelsBooleanValueAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.47
  NAME 'pcelsBooleanValueAuxClass'
  DESC 'Provides a boolean value.'
  SUP pcelsValueAuxClass
  AUXILIARY
  MUST ( pcelsBoolean )
)
```

The pcelsBoolean attribute type represents a boolean. It is mapped from the PolicyBooleanValue.BooleanValue property [PCIM_EXT]. This attribute type is of syntax Boolean [LDAP_SYNTAX]. It has an equality matching rule of booleanMatch [LDAP_MATCH]. Attributes of this type can only have a single value.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.26
  NAME 'pcelsBoolean'
  DESC 'Boolean value'
  EQUALITY booleanMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
  SINGLE-VALUE
)
```

5.18. The Three Reusable Policy Container Classes

The pcelsReusableContainer class represents a container of reusable policy elements. It is mapped from the ReusablePolicyContainer class [PCIM_EXT]. The pcelsReusableContainer class is derived from the pcimRepository class [PCLS]. To maximize flexibility, the pcelsReusableContainer class is defined as abstract. An auxiliary subclass pcelsReusableContainerAuxClass enables the attachment of a reusable policy container to an existing entry, while a structural subclass pcelsReusableContainerInstance permits the representation of a reusable policy container as a standalone entry.

The elements contained in a reusable policy container are aggregated via subordination to a pcelsReusableContainer instance (DIT containment). A reusable policy container can include the elements of another reusable policy container by aggregating the container itself. This is realized by DIT containment when the policy containers are subordinated to one another, or by reference when the aggregating policy container references the aggregated one using the attribute pcelsReusableContainerList.

The pcelsReusableContainer class is defined as follows:

```
( 1.3.6.1.1.9.1.48
  NAME 'pcelsReusableContainer'
  DESC 'Container for reusable policy information'
  SUP pcimRepository
  ABSTRACT
  MAY ( pcelsReusableContainerName
$ pcelsReusableContainerList )
)
```

The pcelsReusableContainerAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.49
  NAME 'pcelsReusableContainerAuxClass '
  DESC 'Container for reusable policy information'
  SUP pcelsReusableContainer
  AUXILIARY
)
```

The pcelsReusableContainerInstance class is defined as follows:

```
( 1.3.6.1.1.9.1.50
  NAME 'pcelsReusableContainerInstance'
  DESC 'Container for reusable policy information'
  SUP pcelsReusableContainer
  STRUCTURAL
)
```

The pcelsReusableContainerName attribute type may be used as naming attribute for pcelsReusableContainer entries. This attribute type is of syntax Directory String [LDAP_SYNTAX]. It has an equality matching rule of caseIgnoreMatch, an ordering matching rule of caseIgnoreOrderingMatch and a substrings matching rule of caseIgnoreSubstringsMatch [LDAP_SYNTAX]. Attributes of this type can only have a single value.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.27
  NAME 'pcelsReusableContainerName'
  DESC 'User-friendly name of a reusable policy container'
  EQUALITY caseIgnoreMatch
  ORDERING caseIgnoreOrderingMatch
  SUBSTR caseIgnoreSubstringsMatch
```

```
     SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
     SINGLE-VALUE
   )
```

The pcelsReusableContainerList attribute type realizes the ContainedDomain association [PCIM_EXT]. This attribute type is of syntax DN [LDAP_SYNTAX]. It has an equality matching rule of distinguishedNameMatch [LDAP_SYNTAX]. Attributes of this type can have multiple values. The only allowed values for pcelsReusableContainerList attributes are DNs of pcelsReusableContainer entries. In a pcelsReusableContainer, the pcelsReusableContainerList attribute represents the associations between this reusable policy container and others for the purpose of including them as nested containers.

This attribute type is defined as follows:

```
   ( 1.3.6.1.1.9.2.28
     NAME 'pcelsReusableContainerList'
     DESC 'Unordered set of DNs of pcelsReusableContainer entries'
     EQUALITY distinguishedNameMatch
     SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
   )
```

Note: PCELS implementations SHOULD support pcelsReusableContainer and its two subclasses and MAY also support the two subclasses of pcimRepository [PCLS].

5.19. The Structural Class pcelsRoleCollection

The pcelsRoleCollection class represents a collection of managed elements that share a common role. It is mapped from the PolicyRoleCollection class [PCIM_EXT]. The pcelsRoleCollection class is a structural object class and it is derived from the pcimPolicy class [PCLS].

The pcelsRoleCollection class is defined as follows:

```
   ( 1.3.6.1.1.9.1.51
     NAME 'pcelsRoleCollection'
     DESC 'Collection of managed elements that share a common role'
     SUP pcimPolicy
     STRUCTURAL
     MUST ( pcelsRole )
     MAY ( pcelsRoleCollectionName
   $ pcelsElementList )
   )
```

The pcelsRole attribute type represents the role associated with a collection of managed elements. It is mapped from the PolicyRoleCollection.PolicyRole property [PCIM_EXT]. This attribute type is of syntax Directory String [LDAP_SYNTAX]. It has an equality matching rule of caseIgnoreMatch, an ordering matching rule of caseIgnoreOrderingMatch and a substrings matching rule of caseIgnoreSubstringsMatch [LDAP_SYNTAX]. Attributes of this type can only have a single value.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.29
  NAME 'pcelsRole'
  DESC 'String representing a role.'
  EQUALITY caseIgnoreMatch
  ORDERING caseIgnoreOrderingMatch
  SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)
```

The pcelsRoleCollectionName attribute type may be used as naming attribute for pcelsRoleCollection entries. This attribute type is of syntax Directory String [LDAP_SYNTAX]. It has an equality matching rule of caseIgnoreMatch, an ordering matching rule of caseIgnoreOrderingMatch and a substrings matching rule of caseIgnoreSubstringsMatch [LDAP_SYNTAX]. Attributes of this type can only have a single value.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.30
  NAME 'pcelsRoleCollectionName'
  DESC 'User-friendly name of a role collection'
  EQUALITY caseIgnoreMatch
  ORDERING caseIgnoreOrderingMatch
  SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)
```

The pcelsElementList attribute type realizes the ElementInPolicyRoleCollection association [PCIM_EXT]. This attribute type is of syntax DN [LDAP_SYNTAX]. It has an equality matching rule of distinguishedNameMatch [LDAP_SYNTAX]. Attributes of this type can have multiple values. In a pcelsRoleCollection, the pcelsElementList attribute represents the associations between this role collection and its members.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.31
  NAME 'pcelsElementList'
  DESC 'Unordered set of managed elements'
  EQUALITY distinguishedNameMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
)
```

5.20. The Abstract Class pcelsFilterEntryBase

The pcelsFilterEntryBase class is the base class for defining message or packet filters. It is mapped from the FilterEntryBase class [PCIM_EXT]. The pcelsFilterEntryBase class is an abstract object class and it is derived from the pcimPolicy class [PCLS].

The pcelsFilterEntryBase class is defined as follows:

```
( 1.3.6.1.1.9.1.52
  NAME 'pcelsFilterEntryBase'
  DESC 'Base class for message or packet filters'
  SUP pcimPolicy
  ABSTRACT
  MAY ( pcelsFilterName
      $ pcelsFilterIsNegated )
)
```

The pcelsFilterName attribute type may be used as naming attribute for pcelsFilterEntryBase entries. This attribute type is of syntax Directory String [LDAP_SYNTAX]. It has an equality matching rule of caseIgnoreMatch, an ordering matching rule of caseIgnoreOrderingMatch and a substrings matching rule of caseIgnoreSubstringsMatch [LDAP_SYNTAX]. Attributes of this type can only have a single value.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.32
  NAME 'pcelsFilterName'
  DESC 'User-friendly name of a filter entry'
  EQUALITY caseIgnoreMatch
  ORDERING caseIgnoreOrderingMatch
  SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)
```

The pcelsFilterIsNegated attribute type indicates whether the match information specified in a pcelsFilterEntryBase is negated or not.

It is mapped from the FilterEntryBase.IsNegated property [PCIM_EXT]. This attribute type is of syntax Boolean [LDAP_SYNTAX]. It has an equality matching rule of booleanMatch [LDAP_MATCH]. Attributes of this type can only have a single value. If this attribute is missing from a pcelsFilterEntryBase instance, applications MUST assume that the filter is not negated.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.33
  NAME 'pcelsFilterIsNegated'
  DESC 'Indicates whether the filter is negated'
  EQUALITY booleanMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
  SINGLE-VALUE
)
```

5.21. The Structural Class pcelsIPHeadersFilter

The pcelsIPHeadersFilter class provides the most commonly required attributes for performing filtering on IP, TCP or UDP headers. It is mapped from the IpHeadersFilter class [PCIM_EXT]. It is a structural object class derived from the pcelsFilterEntryBase class.

The pcelsIPHeadersFilter class is defined as follows:

```
( 1.3.6.1.1.9.1.53
  NAME 'pcelsIPHeadersFilter'
  DESC 'IP header filter'
  SUP pcelsFilterEntryBase
  STRUCTURAL
  MAY ( pcelsIPHdrVersion
      $ pcelsIPHdrSourceAddress
      $ pcelsIPHdrSourceAddressEndOfRange
      $ pcelsIPHdrSourceMask
      $ pcelsIPHdrDestAddress
      $ pcelsIPHdrDestAddressEndOfRange
      $ pcelsIPHdrDestMask
      $ pcelsIPHdrProtocolID
      $ pcelsIPHdrSourcePortStart
      $ pcelsIPHdrSourcePortEnd
      $ pcelsIPHdrDestPortStart
      $ pcelsIPHdrDestPortEnd
      $ pcelsIPHdrDSCPList
      $ pcelsIPHdrFlowLabel )
)
```

Applications MUST assume 'all values' for optional (MAY) attributes not present in a pcelsIPHeadersFilter entry.

[PCIM_EXT] defines several constraints for the IpHeadersFilter class and its properties. All these constraints (even those that, for brevity, are not reiterated in this document) apply to the pcelsIPHeadersFilter class and its attributes. A pcelsIPHeadersFilter entry that violates any of these constraints SHOULD be treated as invalid and the policy rules or groups associated to this entry SHOULD be treated as being disabled, meaning that the execution of such policy rules or groups SHOULD be stopped.

The pcelsIPHdrVersion attribute type indicates the version of the IP addresses to be filtered on. It is mapped from the IpHeadersFilter.HdrIpVersion property [PCIM_EXT]. This attribute type is of syntax Integer [LDAP_SYNTAX]. It has an equality matching rule of integerMatch [LDAP_SYNTAX] and an ordering matching rule of integerOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are 4 and 6.

In a pcelsIPHeadersFilter entry, the pcelsIPHdrVersion attribute type determines the size for the IP version dependent attribute values. These attributes are: pcelsIPHdrSourceAddress, pcelsIPHdrSourceAddressEndOfRange, pcelsIPHdrSourceMask, pcelsIPHdrDestAddress, pcelsIPHdrDestAddressEndOfRange and pcelsIPHdrDestMask. Their valid values are as follows:

```
   for IPv4: OctetStrings with a size of 4
   for IPv6: OctetStrings with a size of 16 or 20
```

If the pcelsIPHdrVersion attribute is missing from a pcelsFilterEntryBase instance, then the filter does not consider IP version in selecting matching packets. In this case, the IP version dependent attributes (listed above) must not be present in the filter entry.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.34
  NAME 'pcelsIPHdrVersion'
  DESC 'IP version'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
```

The pcelsIPHdrSourceAddress attribute type represents a source IP address. It is mapped from the IpHeadersFilter.HdrSrcAddress property [PCIM_EXT]. This attribute type is of syntax OctetString [LDAP_SYNTAX]. It has an equality matching rule of octetStringMatch [LDAP_SCHEMA] and an ordering matching rule of octetStringOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are octet strings with a size of 4, 16, or 20.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.35
  NAME 'pcelsIPHdrSourceAddress'
  DESC 'Source IP address'
  EQUALITY octetStringMatch
  ORDERING octetStringOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.40
  SINGLE-VALUE
)
```

The pcelsIPHdrSourceAddressEndOfRange attribute type represents the end of a range of source IP addresses. It is mapped from the IpHeadersFilter.HdrSrcAddressEndOfRange property [PCIM_EXT]. This attribute type is of syntax OctetString [LDAP_SYNTAX]. It has an equality matching rule of octetStringMatch [LDAP_SCHEMA] and an ordering matching rule of octetStringOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are octet strings with a size of 4, 16, or 20.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.36
  NAME 'pcelsIPHdrSourceAddressEndOfRange'
  DESC 'End of a range of source IP addresses'
  EQUALITY octetStringMatch
  ORDERING octetStringOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.40
  SINGLE-VALUE
)
```

The pcelsIPHdrSourceMask attribute type represents the mask to be used in comparing the source IP address. It is mapped from the IpHeadersFilter.HdrSrcMask property [PCIM_EXT]. This attribute type is of syntax OctetString [LDAP_SYNTAX]. It has an equality matching rule of octetStringMatch [LDAP_SCHEMA] and an ordering matching rule of octetStringOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are octet strings with a size of 4, 16, or 20.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.37
  NAME 'pcelsIPHdrSourceMask'
  DESC 'Mask to be used in comparing the source IP address'
  EQUALITY octetStringMatch
  ORDERING octetStringOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.40
  SINGLE-VALUE
)
```

The pcelsIPHdrDestAddress attribute type represents a destination IP address. It is mapped from the IpHeadersFilter.HdrDestAddress property [PCIM_EXT]. This attribute type is of syntax OctetString [LDAP_SYNTAX]. It has an equality matching rule of octetStringMatch [LDAP_SCHEMA] and an ordering matching rule of octetStringOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are octet strings with a size of 4, 16, or 20.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.38
  NAME 'pcelsIPHdrDestAddress'
  DESC 'Destination IP address'
  EQUALITY octetStringMatch
  ORDERING octetStringOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.40
  SINGLE-VALUE
)
```

The pcelsIPHdrDestAddressEndOfRange attribute type represents the end of a range of destination IP addresses. It is mapped from the IpHeadersFilter.HdrDestAddressEndOfRange property [PCIM_EXT]. This attribute type is of syntax OctetString [LDAP_SYNTAX]. It has an equality matching rule of octetStringMatch [LDAP_SCHEMA] and an ordering matching rule of octetStringOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are octet strings with a size of 4, 16, or 20.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.39
  NAME 'pcelsIPHdrDestAddressEndOfRange'
  DESC 'End of a range of destination IP addresses'
  EQUALITY octetStringMatch
  ORDERING octetStringOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.40
  SINGLE-VALUE
)
```

The pcelsIPHdrDestMask attribute type represents a mask to be used in comparing the destination IP address. It is mapped from the IpHeadersFilter.HdrDestMask property [PCIM_EXT]. This attribute type is of syntax OctetString [LDAP_SYNTAX]. It has an equality matching rule of octetStringMatch [LDAP_SCHEMA] and an ordering matching rule of octetStringOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are octet strings with a size of 4, 16, or 20.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.40
  NAME 'pcelsIPHdrDestMask'
  DESC 'Mask to be used in comparing the destination IP address'
  EQUALITY octetStringMatch
  ORDERING octetStringOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.40
  SINGLE-VALUE
)
```

The pcelsIPHdrProtocolID attribute type indicates an IP protocol type. It is mapped from the IpHeadersFilter.HdrProtocolID property [PCIM_EXT]. This attribute type is of syntax Integer [LDAP_SYNTAX]. It has an equality matching rule of integerMatch [LDAP_SYNTAX] and an ordering matching rule of integerOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are integers in the range 0..255 (inclusive).

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.41
  NAME 'pcelsIPHdrProtocolID'
  DESC 'IP protocol type'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
```

The pcelsIPHdrSourcePortStart attribute type represents the lower end of a range of UDP or TCP source ports. It is mapped from the IpHeadersFilter.HdrSrcPortStart property [PCIM_EXT]. This attribute type is of syntax Integer [LDAP_SYNTAX]. It has an equality matching rule of integerMatch [LDAP_SYNTAX] and an ordering matching rule of integerOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are integers in the range 0..65535 (inclusive).

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.42
  NAME 'pcelsIPHdrSourcePortStart'
  DESC 'Lower end of a range of UDP or TCP source ports'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
```

The pcelsIPHdrSourcePortEnd attribute type represents the upper end of a range of UDP or TCP source ports. It is mapped from the IpHeadersFilter.HdrSrcPortEnd property [PCIM_EXT]. This attribute type is of syntax Integer [LDAP_SYNTAX]. It has an equality matching rule of integerMatch [LDAP_SYNTAX] and an ordering matching rule of integerOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are integers in the range 0..65535 (inclusive).

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.43
  NAME 'pcelsIPHdrSourcePortEnd'
  DESC 'Upper end of a range of UDP or TCP source ports'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
```

The pcelsIPHdrDestPortStart attribute type represents the lower end of a range of UDP or TCP destination ports. It is mapped from the IpHeadersFilter.HdrDestPortStart property [PCIM_EXT]. This attribute type is of syntax Integer [LDAP_SYNTAX]. It has an equality matching rule of integerMatch [LDAP_SYNTAX] and an ordering matching rule of integerOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are integers in the range 0..65535 (inclusive).

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.44
  NAME 'pcelsIPHdrDestPortStart'
  DESC 'Lower end of a range of UDP or TCP destination ports'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
```

The pcelsIPHdrDestPortEnd attribute type represents the upper end of a range of UDP or TCP destination ports. It is mapped from the IpHeadersFilter.HdrDestPortEnd property [PCIM_EXT]. This attribute type is of syntax Integer [LDAP_SYNTAX]. It has an equality matching rule of integerMatch [LDAP_SYNTAX] and an ordering matching rule of integerOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are integers in the range 0..65535 (inclusive).

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.45
  NAME ’pcelsIPHdrDestPortEnd’
  DESC ’Upper end of a range of UDP or TCP destination ports’
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
```

The pcelsIPHdrDSCPList attribute type is mapped from the IpHeadersFilter.HdrDSCP property [PCIM_EXT]. This attribute type is of syntax Integer [LDAP_SYNTAX]. It has an equality matching rule of integerMatch [LDAP_SYNTAX] and an ordering matching rule of integerOrderingMatch [LDAP_MATCH]. Attributes of this type can have multiple values. The only allowed values for attributes of this type are integers in the range 0..63 (inclusive).

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.46
  NAME ’pcelsIPHdrDSCPList’
  DESC ’DSCP values’
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
)
```

The pcelsIPHdrFlowLabel attribute type is mapped from the IpHeadersFilter.HdrFlowLabel property [PCIM_EXT]. This attribute type is of syntax OctetString [LDAP_SYNTAX]. It has an equality matching rule of octetStringMatch [LDAP_SCHEMA] and an ordering matching rule of octetStringOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are octet strings of size 3 (that is, 24 bits) that contain a Flow Label value in the rightmost 20 bits padded on the left with b’0000’.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.47
  NAME ’pcelsIPHdrFlowLabel’
  DESC ’IP flow label’
  EQUALITY octetStringMatch
  ORDERING octetStringOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.40
  SINGLE-VALUE
```

```
   )
```

5.22. The Structural Class pcels8021Filter

The pcels8021Filter class provides 802.1 attributes for performing filtering on 802.1 headers. It is mapped from the 8021Filter class [PCIM_EXT]. The pcels8021Filter class is a structural object class and it is derived from the pcelsFilterEntryBase class.

The pcels8021Filter class is defined as follows:

```
   ( 1.3.6.1.1.9.1.54
     NAME 'pcels8021Filter'
     DESC '802.1 header filter'
     SUP pcelsFilterEntryBase
     STRUCTURAL
     MAY ( pcels8021HdrSourceMACAddress
         $ pcels8021HdrSourceMACMask
         $ pcels8021HdrDestMACAddress
         $ pcels8021HdrDestMACMask
         $ pcels8021HdrProtocolID
         $ pcels8021HdrPriority
         $ pcels8021HdrVLANID )
   )
```

Applications MUST assume 'all values' for optional (MAY) attributes not present in a pcels8021Filter entry.

[PCIM_EXT] defines several constraints for the 8021Filter class and its properties. All these constraints (even those that, for brevity, are not reiterated in this document) apply to the pcels8021Filter class and its attributes. A pcels8021Filter entry that violates any of these constraints SHOULD be treated as invalid and the policy rules or groups associated to this entry SHOULD be treated as being disabled, meaning that the execution of such policy rules or groups SHOULD be stopped.

The pcels8021HdrSourceMACAddress attribute type represents a source MAC address. It is mapped from the 8021Filter.8021HdrSrcMACAddr property [PCIM_EXT]. This attribute type is of syntax OctetString [LDAP_SYNTAX]. It has an equality matching rule of octetStringMatch [LDAP_SCHEMA] and an ordering matching rule of octetStringOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are octet strings with a size of 6.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.48
  NAME 'pcels8021HdrSourceMACAddress'
  DESC 'Source MAC address'
  EQUALITY octetStringMatch
  ORDERING octetStringOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.40
  SINGLE-VALUE
)
```

The pcels8021HdrSourceMACMask attribute type represents the a mask to be used in comparing the source MAC address. It is mapped from the 8021Filter.8021HdrSrcMACMask property [PCIM_EXT]. This attribute type is of syntax OctetString [LDAP_SYNTAX]. It has an equality matching rule of octetStringMatch [LDAP_SCHEMA] and an ordering matching rule of octetStringOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are octet strings with a size of 6.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.49
  NAME 'pcels8021HdrSourceMACMask'
  DESC 'Source MAC address mask'
  EQUALITY octetStringMatch
  ORDERING octetStringOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.40
  SINGLE-VALUE
)
```

The pcels8021HdrDestMACAddress attribute type represents a destination MAC address. It is mapped from the 8021Filter.8021HdrDestMACAddr property [PCIM_EXT]. This attribute type is of syntax OctetString [LDAP_SYNTAX]. It has an equality matching rule of octetStringMatch [LDAP_SCHEMA] and an ordering matching rule of octetStringOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are octet strings with a size of 6.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.50
  NAME 'pcels8021HdrDestMACAddress'
  DESC 'Destination MAC address'
  EQUALITY octetStringMatch
  ORDERING octetStringOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.40
  SINGLE-VALUE
)
```

The pcels8021HdrDestMACMask attribute type represents the a mask to be used in comparing the destination MAC address. It is mapped from the 8021Filter.8021HdrDestMACMask property [PCIM_EXT]. This attribute type is of syntax OctetString [LDAP_SYNTAX]. It has an equality matching rule of octetStringMatch [LDAP_SCHEMA] and an ordering matching rule of octetStringOrderingMatch [LDAP_MATCH]. Attributes of this type can only have a single value. The only allowed values for attributes of this type are octet strings with a size of 6.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.51
  NAME 'pcels8021HdrDestMACMask'
  DESC 'Destination MAC address mask'
  EQUALITY octetStringMatch
  ORDERING octetStringOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.40
  SINGLE-VALUE
)
```

The pcels8021HdrProtocolID attribute type indicates an Ethernet protocol type. It is mapped from the 8021Filter.8021HdrProtocolID property [PCIM_EXT]. This attribute type is of syntax Integer [LDAP_SYNTAX]. It has an equality matching rule of integerMatch [LDAP_SYNTAX] and an ordering matching rule of integerOrderingMatch [LDAP_MATCH]. Attributes of this type can have multiple values. No order is implied. The only allowed values for attributes of this type are integers in the range 0..65535 (inclusive).

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.52
  NAME 'pcels8021HdrProtocolID'
  DESC 'Ethernet protocol ID'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
)
```

The pcels8021HdrPriority attribute type indicates an 802.1Q priority. It is mapped from the 8021Filter.8021HdrPriorityValue property [PCIM_EXT]. This attribute type is of syntax Integer [LDAP_SYNTAX]. It has an equality matching rule of integerMatch [LDAP_SYNTAX] and an ordering matching rule of integerOrderingMatch [LDAP_MATCH]. Attributes of this type can have multiple values. No order is implied. The only allowed values for attributes of this type are integers in the range 0..7 (inclusive).

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.53
  NAME 'pcels8021HdrPriority'
  DESC '802.1Q priority'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
)
```

The pcels8021HdrVLANID attribute type indicates an 802.1Q VLAN Identifier. It is mapped from the 8021Filter.8021HdrVLANID property [PCIM_EXT]. This attribute type is of syntax Integer [LDAP_SYNTAX]. It has an equality matching rule of integerMatch [LDAP_SYNTAX] and an ordering matching rule of integerOrderingMatch [LDAP_MATCH]. Attributes of this type can have multiple values. The only allowed values for attributes of this type are integers in the range 0..4095 (inclusive).

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.54
  NAME 'pcels8021HdrVLANID'
  DESC '802.1Q VLAN ID'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
)
```

5.23. The Auxiliary Class pcelsFilterListAuxClass

The pcelsFilterListAuxClass class represents a collection of device-level filters aggregated in a policy condition. It is mapped from the FilterList class [PCIM_EXT]. pcelsFilterListAuxClass instances can be used as conditions in policy rules or as components in compound conditions. The pcelsFilterListAuxClass class is an auxiliary object class and it is derived from the pcimConditionAuxClass class [PCLS].

The pcelsFilterListAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.55
  NAME 'pcelsFilterListAuxClass'
  DESC 'Collection of pcelsFilterEntryBase filters'
  SUP pcimConditionAuxClass
  AUXILIARY
  MAY ( pcelsFilterListName
      $ pcelsFilterDirection
      $ pcelsFilterEntryList )
)
```

The pcelsFilterListName attribute type may be used as naming attribute for pcelsFilterListAuxClass entries. This attribute type is of syntax Directory String [LDAP_SYNTAX]. It has an equality matching rule of caseIgnoreMatch, an ordering matching rule of caseIgnoreOrderingMatch and a substrings matching rule of caseIgnoreSubstringsMatch [LDAP_SYNTAX]. Attributes of this type can only have a single value.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.55
  NAME 'pcelsFilterListName'
  DESC 'User-friendly name of a FilterList'
  EQUALITY caseIgnoreMatch
  ORDERING caseIgnoreOrderingMatch
  SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)
```

The pcelsFilterDirection attribute type indicates the direction of the packets or messages relative to the interface where the filter is applied. It is mapped from the FilterList.Direction property [PCIM_EXT]. This attribute type is of syntax Integer [LDAP_SYNTAX]. It has an equality matching rule of integerMatch [LDAP_SYNTAX] and an ordering matching rule of integerOrderingMatch [LDAP_MATCH].

Attributes of this type can only have a single value. The only allowed values for attributes of this type are 0 (NotApplicable), 1 (Input), 2 (Output), 3 (Both) and 4 (Mirrored). If this attribute is missing from a pcelsFilterListAuxClass instance, applications MUST assume that a direction is not applicable.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.56
  NAME 'pcelsFilterDirection'
  DESC 'Direction to which this filter is applied'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
```

The pcelsFilterEntryList attribute type realizes the EntriesInFilterList association [PCIM_EXT]. This attribute type is of syntax DN [LDAP_SYNTAX]. It has an equality matching rule of distinguishedNameMatch [LDAP_SYNTAX]. Attributes of this type can have multiple values. The only allowed values for pcelsFilterEntryList attributes are DNs of pcelsFilterEntryBase entries. In a pcelsFilterListAuxClass, the pcelsFilterEntryList attribute represents the associations between this filter collection and its components.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.57
  NAME 'pcelsFilterEntryList'
  DESC 'Unordered set of DNs of pcelsFilterEntryBase entries'
  EQUALITY distinguishedNameMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
)
```

The EntrySequence property of the association EntriesInFilterList is restricted to a single value ('0') [PCIM_EXT] which makes it redundant. Therefore, its mapping to an LDAP schema element is unnecessary.

5.24. The Auxiliary Class pcelsVendorVariableAuxClass

The pcelsVendorVariableAuxClass class provides a general extension mechanism for representing policy variables that have not been specifically modeled. Instead, its two properties are used to define the content and format of the variable, as explained below. This class is intended for vendor-specific extensions that are not amenable to using pcelsVariable; standardized extensions SHOULD NOT use this class.

The pcelsVendorVariableAuxClass class is an auxiliary object class and it is derived from the pcelsVariable class.

The pcelsVendorVariableAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.56
  NAME 'pcelsVendorVariableAuxClass'
  DESC 'Defines registered means to describe a policy variable'
  SUP pcelsVariable
  AUXILIARY
  MAY ( pcelsVendorVariableData $
        pcelsVendorVariableEncoding )
)
```

The pcelsVendorVariableData attribute provides a general mechanism for representing policy variables that have not been specifically modeled. This attribute type is of syntax OctetString [LDAP_SYNTAX]. It has an equality matching rule of octetStringMatch [LDAP_SCHEMA] and an ordering matching rule of octetStringOrderingMatch [LDAP_MATCH]. Attributes of this type can have multiple values. In pcelsVendorVariableAuxClass instances, the format of the values for attributes of this type is identified by the OID stored in the pcelsVendorVariableEncoding attribute.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.58
  NAME 'pcelsVendorVariableData'
  DESC 'Mechanism for representing variables that have not
        been specifically modeled'
  EQUALITY octetStringMatch
  ORDERING octetStringOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.40
)
```

The pcelsVendorVariableEncoding attribute identifies the format for representing policy variables that have not been specifically modeled. This attribute type is of syntax OID [LDAP_SYNTAX]. It has an equality matching rule of objectIdentifierMatch [LDAP_SYNTAX]. Attributes of this type can only have a single value. In pcelsVendorVariableAuxClass instances, the pcelsVendorVariableEncoding attribute is used to identify the format and semantics for the pcelsVendorVariableData attribute values.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.59
  NAME 'pcelsVendorVariableEncoding'
  DESC 'Identifies the format and semantics for policy variables'
  EQUALITY objectIdentifierMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.38
  SINGLE-VALUE
)
```

5.25. The Auxiliary Class pcelsVendorValueAuxClass

The pcelsVendorValueAuxClass class provides a general extension mechanism for representing policy values that have not been specifically modeled. Instead, its two properties are used to define the content and format of the policy value, as explained below. This class is intended for vendor-specific extensions that are not amenable to using pcelsValueAuxClass; standardized extensions SHOULD NOT use this class.

The pcelsVendorValueAuxClass class is an auxiliary object class and it is derived from the pcelsValueAuxClass class.

The pcelsVendorValueAuxClass class is defined as follows:

```
( 1.3.6.1.1.9.1.57
  NAME 'pcelsVendorValueAuxClass'
  DESC 'Defines registered means to describe a policy value'
  SUP pcelsValueAuxClass
  AUXILIARY
  MAY ( pcelsVendorValueData $
        pcelsVendorValueEncoding )
)
```

The pcelsVendorValueData attribute provides a general mechanism for representing policy values that have not been specifically modeled. This attribute type is of syntax OctetString [LDAP_SYNTAX]. It has an equality matching rule of octetStringMatch [LDAP_SCHEMA] and an ordering matching rule of octetStringOrderingMatch [LDAP_MATCH]. Attributes of this type can have multiple values. In pcelsVendorValueAuxClass instances, the format of the values for attributes of this type is identified by the OID stored in the pcelsVendorValueEncoding attribute.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.60
  NAME 'pcelsVendorValueData'
  DESC 'Mechanism for representing values that have not been
        specifically modeled'
  EQUALITY octetStringMatch
  ORDERING octetStringOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.40
)
```

The pcelsVendorValueEncoding attribute identifies the format for representing policy values that have not been specifically modeled. This attribute type is of syntax OID [LDAP_SYNTAX]. It has an equality matching rule of objectIdentifierMatch [LDAP_SYNTAX]. Attributes of this type can only have a single value. In pcelsVendorVarlueAuxClass instances, the pcelsVendorValueEncoding attribute is used to identify the format and semantics for the pcelsVendorValueData attribute values.

This attribute type is defined as follows:

```
( 1.3.6.1.1.9.2.61
  NAME 'pcelsVendorValueEncoding'
  DESC 'Identifies the format and semantics for policy values'
  EQUALITY objectIdentifierMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.38
  SINGLE-VALUE
)
```

6. Security Considerations

The Policy Core LDAP Schema [PCLS] describes the general security considerations related to the general core policy schema. The extensions defined in this document do not introduce any additional considerations related to security.

7. IANA Considerations

Refer to RFC 3383, "Internet Assigned Numbers Authority (IANA) Considerations for the Lightweight Directory Access Protocol (LDAP)" [LDAP-IANA].

7.1. Object Identifiers

The IANA has registered an LDAP Object Identifier for use in this technical specification according to the following template:

```
Subject: Request for LDAP OID Registration
Person & e-mail address to contact for further information:
Mircea Pana (mpana@metasolv.com)
Specification: RFC 4104
Author/Change Controller: IESG
Comments:
   The assigned OID is used as a base for identifying
   a number of schema elements defined in this document.
```

IANA has assigned an OID of 1.3.6.1.1.9 with the name of pcelsSchema to this registration as recorded in the following registry:

http://www.iana.org/assignments/smi-numbers

7.2. Object Identifier Descriptors

The IANA has registered the LDAP Descriptors used in this technical specification as detailed in the following template:

```
Subject: Request for LDAP Descriptor Registration Update
Descriptor (short name): see comment
Object Identifier: see comment
Person & e-mail address to contact for further information:
   Mircea Pana (mpana@metasolv.com)
Usage: see comment
Specification: RFC 4104
Author/Change Controller: IESG
Comments:
```

The following descriptors have been added:

NAME	Type	OID
pcelsPolicySet	O	1.3.6.1.1.9.1.1
pcelsPolicySetAssociation	O	1.3.6.1.1.9.1.2
pcelsGroup	O	1.3.6.1.1.9.1.3
pcelsGroupAuxClass	O	1.3.6.1.1.9.1.4

pcelsGroupInstance	O	1.3.6.1.1.9.1.5
pcelsRule	O	1.3.6.1.1.9.1.6
pcelsRuleAuxClass	O	1.3.6.1.1.9.1.7
pcelsRuleInstance	O	1.3.6.1.1.9.1.8
pcelsConditionAssociation	O	1.3.6.1.1.9.1.9
pcelsActionAssociation	O	1.3.6.1.1.9.1.10
pcelsSimpleConditionAuxClass	O	1.3.6.1.1.9.1.11
pcelsCompoundConditionAuxClass	O	1.3.6.1.1.9.1.12
pcelsCompoundFilterConditionAuxClass	O	1.3.6.1.1.9.1.13
pcelsSimpleActionAuxClass	O	1.3.6.1.1.9.1.14
pcelsCompoundActionAuxClass	O	1.3.6.1.1.9.1.15
pcelsVariable	O	1.3.6.1.1.9.1.16
pcelsExplicitVariableAuxClass	O	1.3.6.1.1.9.1.17
pcelsImplicitVariableAuxClass	O	1.3.6.1.1.9.1.18
pcelsSourceIPv4VariableAuxClass	O	1.3.6.1.1.9.1.19
pcelsSourceIPv6VariableAuxClass	O	1.3.6.1.1.9.1.20
pcelsDestinationIPv4VariableAuxClass	O	1.3.6.1.1.9.1.21
pcelsDestinationIPv6VariableAuxClass	O	1.3.6.1.1.9.1.22
pcelsSourcePortVariableAuxClass	O	1.3.6.1.1.9.1.23
pcelsDestinationPortVariableAuxClass	O	1.3.6.1.1.9.1.24
pcelsIPProtocolVariableAuxClass	O	1.3.6.1.1.9.1.25
pcelsIPVersionVariableAuxClass	O	1.3.6.1.1.9.1.26
pcelsIPToSVariableAuxClass	O	1.3.6.1.1.9.1.27
pcelsDSCPVariableAuxClass	O	1.3.6.1.1.9.1.28
pcelsFlowIdVariableAuxClass	O	1.3.6.1.1.9.1.29
pcelsSourceMACVariableAuxClass	O	1.3.6.1.1.9.1.30
pcelsDestinationMACVariableAuxClass	O	1.3.6.1.1.9.1.31
pcelsVLANVariableAuxClass	O	1.3.6.1.1.9.1.32
pcelsCoSVariableAuxClass	O	1.3.6.1.1.9.1.33
pcelsEthertypeVariableAuxClass	O	1.3.6.1.1.9.1.34
pcelsSourceSAPVariableAuxClass	O	1.3.6.1.1.9.1.35
pcelsDestinationSAPVariableAuxClass	O	1.3.6.1.1.9.1.36
pcelsSNAPOUIVariableAuxClass	O	1.3.6.1.1.9.1.37
pcelsSNAPTypeVariableAuxClass	O	1.3.6.1.1.9.1.38
pcelsFlowDirectionVariableAuxClass	O	1.3.6.1.1.9.1.39
pcelsValueAuxClass	O	1.3.6.1.1.9.1.40
pcelsIPv4AddrValueAuxClass	O	1.3.6.1.1.9.1.41
pcelsIPv6AddrValueAuxClass	O	1.3.6.1.1.9.1.42
pcelsMACAddrValueAuxClass	O	1.3.6.1.1.9.1.43
pcelsStringValueAuxClass	O	1.3.6.1.1.9.1.44
pcelsBitStringValueAuxClass	O	1.3.6.1.1.9.1.45
pcelsIntegerValueAuxClass	O	1.3.6.1.1.9.1.46
pcelsBooleanValueAuxClass	O	1.3.6.1.1.9.1.47
pcelsReusableContainer	O	1.3.6.1.1.9.1.48
pcelsReusableContainerAuxClass	O	1.3.6.1.1.9.1.49
pcelsReusableContainerInstance	O	1.3.6.1.1.9.1.50
pcelsRoleCollection	O	1.3.6.1.1.9.1.51
pcelsFilterEntryBase	O	1.3.6.1.1.9.1.52

pcelsIPHeadersFilter	O	1.3.6.1.1.9.1.53
pcels8021Filter	O	1.3.6.1.1.9.1.54
pcelsFilterListAuxClass	O	1.3.6.1.1.9.1.55
pcelsVendorVariableAuxClass	O	1.3.6.1.1.9.1.56
pcelsVendorValueAuxClass	O	1.3.6.1.1.9.1.57
pcelsPolicySetName	A	1.3.6.1.1.9.2.1
pcelsDecisionStrategy	A	1.3.6.1.1.9.2.2
pcelsPolicySetList	A	1.3.6.1.1.9.2.3
pcelsPriority	A	1.3.6.1.1.9.2.4
pcelsPolicySetDN	A	1.3.6.1.1.9.2.5
pcelsConditionListType	A	1.3.6.1.1.9.2.6
pcelsConditionList	A	1.3.6.1.1.9.2.7
pcelsActionList	A	1.3.6.1.1.9.2.8
pcelsSequencedActions	A	1.3.6.1.1.9.2.9
pcelsExecutionStrategy	A	1.3.6.1.1.9.2.10
pcelsVariableDN	A	1.3.6.1.1.9.2.11
pcelsValueDN	A	1.3.6.1.1.9.2.12
pcelsIsMirrored	A	1.3.6.1.1.9.2.13
pcelsVariableName	A	1.3.6.1.1.9.2.14
pcelsExpectedValueList	A	1.3.6.1.1.9.2.15
pcelsVariableModelClass	A	1.3.6.1.1.9.2.16
pcelsVariableModelProperty	A	1.3.6.1.1.9.2.17
pcelsExpectedValueTypes	A	1.3.6.1.1.9.2.18
pcelsValueName	A	1.3.6.1.1.9.2.19
pcelsIPv4AddrList	A	1.3.6.1.1.9.2.20
pcelsIPv6AddrList	A	1.3.6.1.1.9.2.21
pcelsMACAddrList	A	1.3.6.1.1.9.2.22
pcelsStringList	A	1.3.6.1.1.9.2.23
pcelsBitStringList	A	1.3.6.1.1.9.2.24
pcelsIntegerList	A	1.3.6.1.1.9.2.25
pcelsBoolean	A	1.3.6.1.1.9.2.26
pcelsReusableContainerName	A	1.3.6.1.1.9.2.27
pcelsReusableContainerList	A	1.3.6.1.1.9.2.28
pcelsRole	A	1.3.6.1.1.9.2.29
pcelsRoleCollectionName	A	1.3.6.1.1.9.2.30
pcelsElementList	A	1.3.6.1.1.9.2.31
pcelsFilterName	A	1.3.6.1.1.9.2.32
pcelsFilterIsNegated	A	1.3.6.1.1.9.2.33
pcelsIPHdrVersion	A	1.3.6.1.1.9.2.34
pcelsIPHdrSourceAddress	A	1.3.6.1.1.9.2.35
pcelsIPHdrSourceAddressEndOfRange	A	1.3.6.1.1.9.2.36
pcelsIPHdrSourceMask	A	1.3.6.1.1.9.2.37
pcelsIPHdrDestAddress	A	1.3.6.1.1.9.2.38
pcelsIPHdrDestAddressEndOfRange	A	1.3.6.1.1.9.2.39
pcelsIPHdrDestMask	A	1.3.6.1.1.9.2.40
pcelsIPHdrProtocolID	A	1.3.6.1.1.9.2.41
pcelsIPHdrSourcePortStart	A	1.3.6.1.1.9.2.42
pcelsIPHdrSourcePortEnd	A	1.3.6.1.1.9.2.43

pcelsIPHdrDestPortStart	A	1.3.6.1.1.9.2.44
pcelsIPHdrDestPortEnd	A	1.3.6.1.1.9.2.45
pcelsIPHdrDSCPList	A	1.3.6.1.1.9.2.46
pcelsIPHdrFlowLabel	A	1.3.6.1.1.9.2.47
pcels8021HdrSourceMACAddress	A	1.3.6.1.1.9.2.48
pcels8021HdrSourceMACMask	A	1.3.6.1.1.9.2.49
pcels8021HdrDestMACAddress	A	1.3.6.1.1.9.2.50
pcels8021HdrDestMACMask	A	1.3.6.1.1.9.2.51
pcels8021HdrProtocolID	A	1.3.6.1.1.9.2.52
pcels8021HdrPriority	A	1.3.6.1.1.9.2.53
pcels8021HdrVLANID	A	1.3.6.1.1.9.2.54
pcelsFilterListName	A	1.3.6.1.1.9.2.55
pcelsFilterDirection	A	1.3.6.1.1.9.2.56
pcelsFilterEntryList	A	1.3.6.1.1.9.2.57
pcelsVendorVariableData	A	1.3.6.1.1.9.2.58
pcelsVendorVariableEncoding	A	1.3.6.1.1.9.2.59
pcelsVendorValueData	A	1.3.6.1.1.9.2.60
pcelsVendorValueEncoding	A	1.3.6.1.1.9.2.61
pcelsRuleValidityPeriodList	A	1.3.6.1.1.9.2.62

where Type A is Attribute, Type O is ObjectClass

These assignments are recorded in the following registry:

http://www.iana.org/assignments/ldap-parameters

8. Acknowledgements

We would like to thank Kurt Zeilenga, Bert Wijnen, Ryan Moats, John Strassner, David McTavish, Larry Bartz and all the other members of the Policy Framework WG for reviewing this document and making many helpful suggestions and corrections.

We would also like to thank Joel Halpern (co-chair of the Policy Framework WG) for his support, for bringing this document to the attention of the Policy Framework WG and for moderating the resulting interactions.

9. Normative References

[KEYWORDS] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[CIM] Distributed Management Task Force, Inc., "Common Information Model (CIM) Specification", Version 2.2, June 14, 1999, http://www.dmtf.org/standards/documents/CIM/DSP0004.pdf

[CIM_LDAP] Distributed Management Task Force, Inc., "DMTF LDAP Schema for the CIM v2.5 Core Information Model", April 15, 2002, http://www.dmtf.org/standards/documents/DEN/DSP0123.pdf

[PCIM] Moore, B., Ellesson, E., Strassner, J., and A. Westerinen, "Policy Core Information Model -- Version 1 Specification", RFC 3060, February 2001.

[PCIM_EXT] Moore, B., "Policy Core Information Model (PCIM) Extensions", RFC 3460, January 2003.

[PCLS] Strassner, J., Moore, B., Moats, R., and E. Ellesson, "Policy Core Lightweight Directory Access Protocol (LDAP) Schema", RFC 3703, February 2004.

[LDAP] Hodges, J. and R. Morgan, "Lightweight Directory Access Protocol (v3): Technical Specification", RFC 3377, September 2002.

[LDAP_SYNTAX] Wahl, M., Coulbeck, A., Howes, T., and S. Kille, "Lightweight Directory Access Protocol (v3): Attribute Syntax Definitions", RFC 2252, December 1997.

[LDAP_SCHEMA] Wahl, M., "A Summary of the X.500(96) User Schema for use with LDAPv3", RFC 2256, December 1997.

[LDAP_MATCH] Zeilenga, K., "Lightweight Directory Access Protocol (LDAP): Additional Matching Rules", RFC 3698, February 2004.

[X.501] The Directory: Models. ITU-T Recommendation X.501, 2001.

[X.520] The Directory: Selected Attribute Types. ITU-T Recommendation X.520, 2001.

10. Informative References

[LDAP-IANA] Zeilenga, K., "Internet Assigned Numbers Authority (IANA) Considerations for the Lightweight Directory Access Protocol (LDAP)", BCP 64, RFC 3383, September 2002.

Authors' Addresses

Mircea Pana
MetaSolv Software Inc.
360 Legget Drive
Ottawa, Ontario, Canada
K2K 3N1

EMail: mpana@metasolv.com

Angelica Reyes
Department of Computer Architecture
Technical University of Catalonia
Campus Castelldefels
Spain

EMail: mreyes@ac.upc.edu

Antoni Barba
Technical University of Catalonia
Jordi-Girona 1-3
08034 Barcelona
Spain

EMail: telabm@mat.upc.es

David Moron
Technical University of Catalonia
Jordi-Girona 1-3
08034 Barcelona
Spain

EMail: dmor4477@hotmail.com

Marcus Brunner
NEC Europe Ltd.
Kurfuersten-Anlage 36
D-69115 Heidelberg
Germany

EMail: brunner@netlab.nec.de

Full Copyright Statement

Intellectual Property

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.