

IPWorks AAA LDAP CUDB Interface

INTERWORK DESCRIPTION

Copyright

© Ericsson AB 2014-2017. All rights reserved. No part of this document may be reproduced in any form without the written permission of the copyright owner.

Disclaimer

The contents of this document are subject to revision without notice due to continued progress in methodology, design and manufacturing. Ericsson shall have no liability for any error or damage of any kind resulting from the use of this document.

Trademark List

Ericsson is the trademark or registered trademark of Telefonaktiebolaget LM Ericsson.

All other product or service names mentioned in this document are trademarks of their respective companies.



Contents

1	Introduction	1
1.1	Prerequisites	1
1.2	Related Information	1
2	Interface Overview	3
2.1	Interface Role	3
2.2	Services	3
2.3	Encapsulation and Addressing	3
3	LDAP Directory Information Tree	5
3.1	LDAP Directory Information Tree for AAA FE (Radius)	5
3.2	LDAP Directory Information Tree for AAA FE (PKI)	8
4	AAA Entries	11
4.1	AAA Entries for AAA FE (Radius)	11
4.2	AA Entries for AAA FE (PKI)	18
5	Procedures	23
5.1	Procedures for AAA FE (Radius)	24
5.2	Procedures for AAA FE (PKI)	38
6	Related Standards	41
	Reference List	43





1 Introduction

This document describes the interface between Authentication, Authorization, Accounting services (AAA) and the Centralized User Database (CUDB).

Scope

This document covers the following topics:

- Describes the LDAP user Database for accessing the CUDB.
- Provides the Ericsson reference DIT for accessing AAA data.
- Specifies the LDAP operations required to be supported by the CUDB.

Target Groups

This document is intended for personnel who needs to understand the logical entity, including interfaces and protocols, of IPWorks.

1.1 Prerequisites

There are no prerequisites for reading this document.

1.2 Related Information

Trademark information, typographic conventions, definition and explanation of acronyms and terminology can be found in the following documents:

- Trademark Information, Reference [2]
- Glossary of Terms and Acronyms, Reference [1]
- Typographic Conventions, Reference [3]

For the standards related to this interface, see Section References.





2 Interface Overview

The CUDB is a network entity in a layered architecture domain that serves as the central storage point for AAA data and other applications (HSS, AuC, and so on). The CUDB is built as an LDAP directory server, containing the necessary entries and attributes according to the defined schema for the different applications. The LDAP server accesses the AAA data, which is then re-structured according to LDAP specifications. That is, the data is displayed in a tree structure format from an external LDAP client. The tree structure is called Directory Information Tree (DIT), where each entry is identified by a Relative Distinguished Name (RDN), referring to the path to the tree root, that is, the Distinguished Name (DN).

2.1 Interface Role

The AAA LDAP CUDB interface uses the LDAP Protocol to access the CUDB and provides the reference Ericsson DIT for accessing the AAA data. The interface also specifies the LDAP operations required to be supported by the CUDB.

2.2 Services

The services offered by the AAA LDAP CUDB interface are presented in Table 1.

Table 1 Offered Services

Offered Service	Description
Search/add/delete/modify	Specifies the LDAP operations required to be supported by the CUDB.

2.3 Encapsulation and Addressing

The following lower-level protocols are used on this interface:

- TCP
- LDAP





3 LDAP Directory Information Tree

LDAP information is displayed in a tree structure format, called DIT. The DIT is composed of entries that have one or more attributes.

The name and value of all the attributes of an entry form the RDN of the entry. The concatenation of the RDNs of the sequence of entries from a particular object to the root entry of the tree forms the DN. The DN uniquely identifies an object within the tree.

This section describes the LDAP DIT for the following service:

- AAA FE (Radius)
- AAA FE (PKI)

3.1 LDAP Directory Information Tree for AAA FE (Radius)

The DIT of the AAA FE (Radius) application is built into the following main structures corresponding to the data it contains:

- Subscriber data
- Subscriber common data

Page 6 presents the container data level distribution offered by the CUDB for the AAA FE (Radius) application.

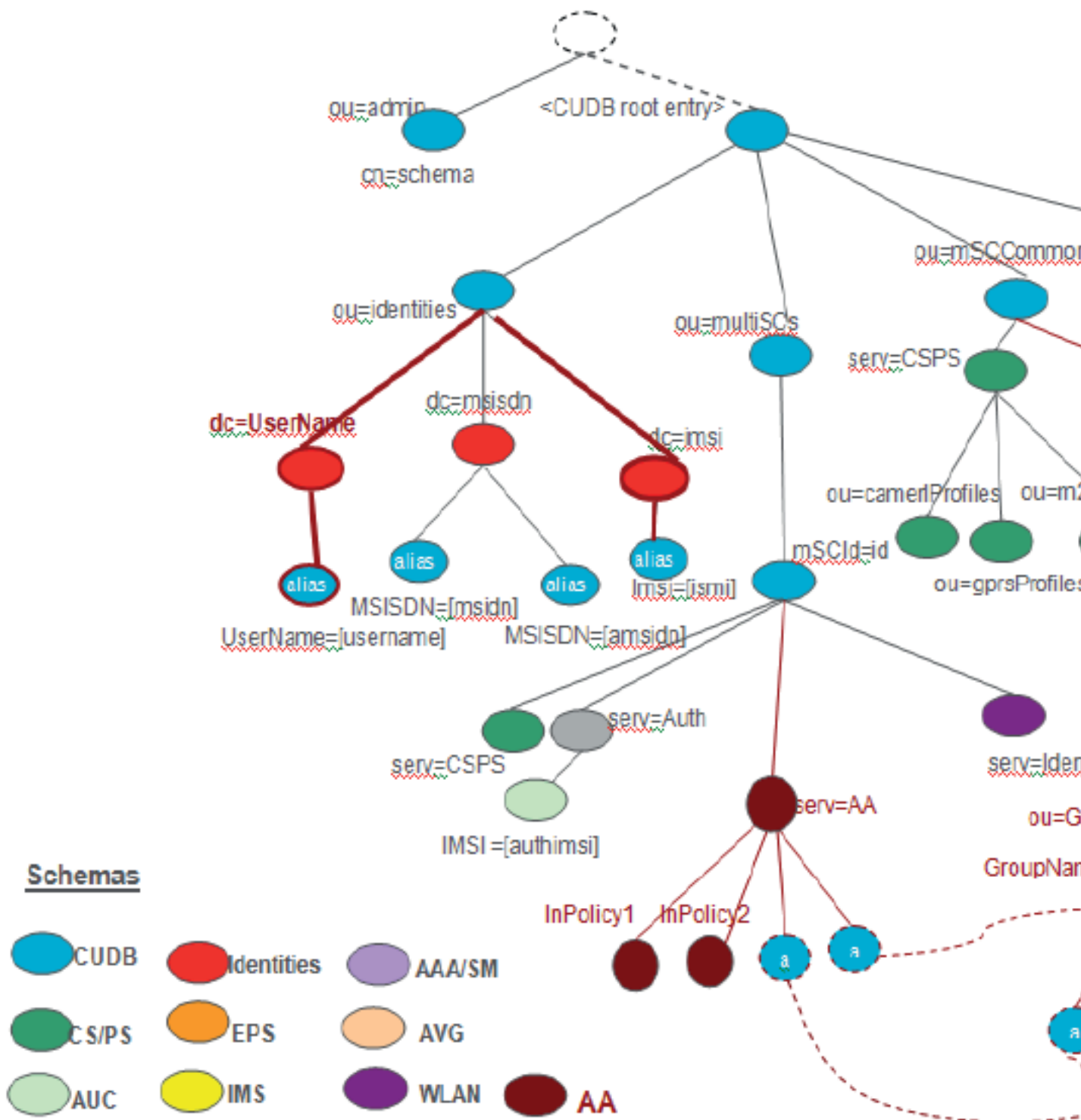


Figure 1 AAA FE (Radius) Directory Information Tree



3.1.1 Subscriber Data

Subscriber data is located under the container level entry `mSCId=<multiserviceconsumer identity>` in the DIT. Within this entry, AAA subscriber data is located as a child under AA services, that is, `serv=AA`.

Entries above the `serv` entry are described in the CUDB data model. For more information, see Reference [4].

The most important piece of information to identify a subscriber in the AAA DIT is the `UserName`.

AAA subscriber data is accessible either through the `multiSCs` branch or through the `identities` branch.

The `identities` branch contains an alias to the corresponding `multiSCs` branch entry. The DN of the associated subscriber identifier entry is obtained by de-referencing the alias.

The `serv` entry contains the individual policies. The individual policies belong to the subscriber.

The `serv` entry contains aliases to the common subscriber data under the `mSCCommonData` branch. The DN of the associated common subscriber data entry is obtained by de-referencing the alias. When common subscriber data is accessed by de-referencing the aliases in the `serv` entry, the data is accessible only for reading purposes.

Subscriber data provisioning operations are performed through the `multiSCs` branch and traffic operations are performed through the `identities` branch.

— Access to AAA Subscriber Data

Access to the AAA subscriber data entries can be provisioned using the following DN:

```
serv=AA, mscId=<multiserviceconsumerId>,
ou=multiSCs, dc=<CUDB root entry>
```

— AAA FE (Radius) Access to AAA Subscriber Data

AAA subscriber data is accessible from the AAA Front End (FE) by means of using the standard LDAP aliases for searching and the Ericsson LDAP aliases for modification. When an LDAP alias is used for searching or modification, the alias is de-referenced when the "search" or "modify" LDAP operation is performed, respectively.

The AAA subscriber data entry is accessible from the AAA FE using the following DN:

```
serv=AA, UserName=<UserName>, dc=UserName,
ou=identities, dc=<CUDB root entry>
```

The DN points to the following entry in the CUDB:



```
serv=AA, msclD=<multiserviceconsumerId>,  
ou=multiSCs
```

3.1.2 Subscriber Common Data

Subscriber common data is located under the container level entry `mscCommonData`. Subscriber common data is located in the tree as a child under AA services, that is, `serv=AA`. Within this entry, AAA subscriber common data is located under Groups or Policies.

Entries above the “serv” entry are described in the CUDB data model. For more information, see Reference [4].

Under the `serv` level, each subscriber entry is accessible based on the group or policy identifier.

— Group Access to Subscriber Common Data

```
groupname=<groupname>, ou=groups, serv=AA,  
ou=mscCommonData, dc=<CUDB root entry>
```

— Policy Access to Subscriber Common Data

```
policyname=<policyname>, ou=policies, serv=AA,  
ou=mscCommonData, dc=<CUDB root entry>
```

Subscriber common data is also accessible for reading purposes through the identities branch by de-referencing the aliases.

```
ei=Policy1, serv=AA, UserName=<UserName>,  
dc=UserName, ou=identities, dc=<CUDB root entry>
```

```
ei=Group1, serv=AA, UserName=<UserName>,  
dc=UserName, ou=identities, dc=<CUDB root entry>
```

3.2 LDAP Directory Information Tree for AAA FE (PKI)

Figure 2 presents the container data level distribution offered by the CUDB for the AAA FE (PKI) application.

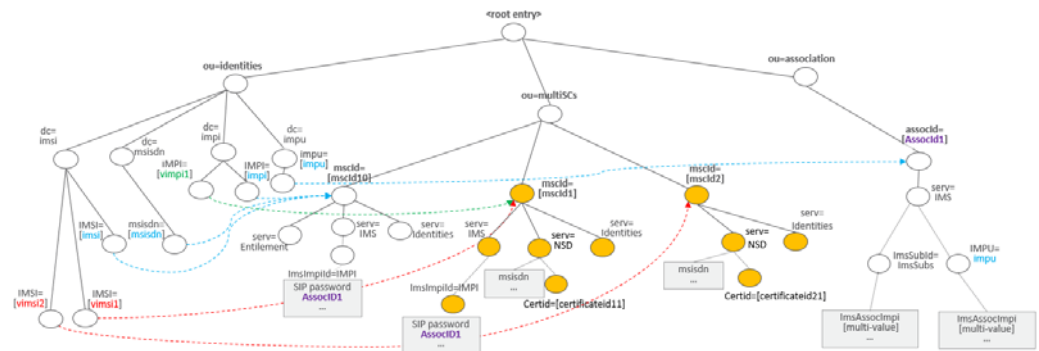


Figure 2 AAA FE (PKI) Directory Information Tree

— AAA PKI Subscriber Data

AAA PKI subscriber data entries can be provisioned using the following DN:

```
serv=NSD,mscId=<multiserviceconsumerId>,ou=multiSCs,dc=<CUDB  
root entry>
```

— AAA FE (PKI) Access to AAA PKI Subscriber Data

The AAA PKI subscriber data entry is accessible from the AAA FE using the following DN:

```
serv=NSD,IMSI=%s,dc=IMSI,ou=identities,dc=<CUDB root entry>
```





4 AAA Entries

The tables in this section specify all the entries for the AAA application. According to LDAP standard RFC 4512 (see Reference [7]), an entry can be initiated with attributes corresponding to several object classes.

The tables contain a header and three columns with the following information:

Table 2 Attribute Description

Entry Name		
For more details, see Reference [5].		
Attributes	Format, Remarks	Example
Brief description of the entry and the OIDs of the entry.	Description of the LDAP syntax characteristics of the entry.	

4.1 AAA Entries for AAA FE (Radius)

4.1.1 AA Profile Entry

This entry contains the authentication and authorization data and it is a child of the mSC entry.

Table 3 Entry Name: AA Profile Entry

Attributes	Description	Format, Remarks	Examples
serv	<p>This attribute identifies the container entry for AA data.</p> <p>Defined in CUDB LDAP Interwork Description included in the object class CUDBS serviceAuxiliary, see Reference [4].</p>	<ul style="list-style-type: none"> • Type: IA5 String • Value range: 1 - 32 characters • Value = "AA" • Required: Mandatory 	"AA"



Attributes	Description	Format, Remarks	Examples
UserName	This attribute identifies the user identity.	<ul style="list-style-type: none">• Type: Directory String• Single-value attribute• Value range: 1 - 255 characters• Required: Mandatory	"User1"
IMSI		<ul style="list-style-type: none">• Type: Directory String• Single-value attribute• Value range: 1 - 15 characters• Required: Optional	
UserPassword	This attribute identifies the user password.	<ul style="list-style-type: none">• Type: Directory String• Single-value attribute• Value range: 1 - 256 characters• Required: Mandatory	"abcdefg"
AuthMethod	This attribute identifies the authentication method.	<ul style="list-style-type: none">• Type: Directory String• Single-value attribute• Value range: 1 - 64 characters• Required: Optional	It supports "NONE", "EAP-MD5", "EAP-SIM", and "EAP-AKA"



Attributes	Description	Format, Remarks	Examples
IPAllocType	This attribute identifies the IP allocation type.	<ul style="list-style-type: none"> • Type: Numeric String • Single-value attribute • Value range: 0- 3 characters • Required: Optional 	It supports "0", "1", "2", and "3"
IPAllocValue	This attribute identifies the IP allocation value.	<ul style="list-style-type: none"> • Type: Directory String • Single-value attribute • Value range: 1 - 64 characters • Required: Optional 	"10.0.0.2"
IPv6PrefixAllocType	This attribute identifies the IPv6 prefix allocation type.	<ul style="list-style-type: none"> • Type: Directory String • Single-value attribute • Value range: 1 - 36 characters • Required: Optional 	
IPv6PrefixAllocValue	This attribute identifies the IPv6 prefix allocation value.	<ul style="list-style-type: none"> • Type: Directory String • Single-value attribute • Value range: 1 - 255 characters • Required: Optional 	



Attributes	Description	Format, Remarks	Examples
GroupNameList	This attribute identifies the list of groups which the subscriber belongs to.	<ul style="list-style-type: none">• Type: Directory String• Multi-value attribute• Value range: 1 - 64 characters• Required: Optional	"GroupA"
PolicyNameList	This attribute identifies the list of shared policies which applies to the user.	<ul style="list-style-type: none">• Type: Directory String• Multi-value attribute• Value range: 1 - 64 characters• Required: Optional	"PolicyA"

4.1.2

AA Individual Policy Entry

This entry contains the individual AA policy data and it is a child of the AA Profile entry.

Table 4 Entry Name: AA Individual Policy

Attributes	Description	Format, Remarks	Examples
PolicyName	This attribute identifies the policy.	<ul style="list-style-type: none">• Type: Directory String• Single-value attribute• Value range: 1 - 64 characters• Required: Mandatory	"Policy1"



Attributes	Description	Format, Remarks	Examples
PolicyChecklist	This attribute identifies the list of policy checklist. This attribute is couple with the AAPolicyReplylist. So value 1 in AAPolicyChecklist is coupled with value 1 in AAPolicyReplylist, and so on.	<ul style="list-style-type: none"> • Type: Directory String • Multi-value attribute • Value range: 1 - 1024 characters • Required: Optional <p>The formula of checklist is this:</p> <pre><![CDATA[expression := condition '(' expression ')' condition := avpname relop value logicalop := '&' ' ' '&&' ' ' relop := '=' '==' '!=' '>' '>=' '<'</pre>	"User-Name ? 1 && User-Name != "IMS-User""
PolicyReplylist	This attribute identifies the list of policy replylist. This attribute is couple with the AAPolicyChecklist. So value 1 in AAPolicyChecklist is coupled with value 1 in AAPolicyReplylist, and so on.	<ul style="list-style-type: none"> • Type: Directory String • Multi-value attribute • Value range: 1 - 1024 characters • Required: Optional <p>The formula of replylist is this:</p> <pre><![CDATA[expression := condition ', ' expression condition := avpname = value value := fixed value \$REQUEST</pre>	"User-Name = \$REQUEST, Login-IP-Host = 10.170.4.169"

4.1.3 AA Shared Policy Alias Entry

This entry contains an alias to the AA Shared Policy entry in the mSCCommon branch and it is a child of the AA entry.



Table 5 Entry Name: AA Shared Policy

Attributes	Format, Remarks	Examples
ei	<ul style="list-style-type: none">Type: Directory StringValue range: 1 - 32 charactersRequired: Mandatory	"Policy1"
aliasedObjectName	<ul style="list-style-type: none">Type: Distinguished NameRequired: Mandatory	PolicyName = <PolicyName>, ou = Policies, serv = AA, ou = mSCCommonData <CUDB root entry >

4.1.4 AA Group Alias Entry

This entry contains an alias to the AA Group entry in the mSCCommon branch and it is a child of the AA entry.

Table 6 Entry Name: AA Group

Attributes	Format, Remarks	Examples
ei	<ul style="list-style-type: none">Type: Directory StringValue range: 1 - 32 charactersRequired: Mandatory	"Group1"
aliasedObjectName	<ul style="list-style-type: none">Type: Distinguished NameRequired: Mandatory	GroupName = <GroupName>, ou = Groups, serv = AA, ou = mSCCommonData <CUDB root entry>

4.1.5 AAA Multiservice Consumers Entry

This entry is the container for the AAA data and for the group and policy alias entries in the CUDB.

Table 7 Entry Name: AA

serv = AA, ou = mscCommonData, <root entry>	
Objectclass top	
Objectclass CUDBService	



serv = AA, ou = mscCommonData, <root entry>		
Attributes	Format, Remarks	Examples
serv	<ul style="list-style-type: none"> • Type: IA5 String • Value range: 1 - 32 characters • Value: AA • Required: Mandatory 	"AA"

4.1.6 AA Shared Policy Domain Entry

This entry is the container for the shared policies within the Policy domain.

Table 8 Entry Name: Policies

Attributes	Format, Remarks	Examples
ou	<ul style="list-style-type: none"> • Type: IA5 String • Value range: 1 - 32 characters • Value: AA • Required: Mandatory 	"Policies"

4.1.7 AA Shared Policy Entry

This entry contains the policy data and it is a child of the AA Shared Policy domain entry.

Table 9 Entry Name: AA Shared Policy

Attributes	Format, Remarks	Examples
PolicyName	<ul style="list-style-type: none"> • Type: Directory String • Single-value attribute • Value range: 1 - 64 characters • Required: Mandatory 	"123456789"

4.1.8 AA Group Domain entry

This entry is the container for the groups within the Group domain.



Table 10 Entry Name: Groups

Attributes	Format, Remarks	Examples
ou	<ul style="list-style-type: none">• Type: IA5 String• Value range: 1 - 32 characters• Value = Groups• Required: Mandatory	“Groups”

4.1.9

AA Group Entry

This entry contains the AA Group data and it is a child of the Group domain entry.

Table 11 Entry Name: AA Group

Attributes	Description	Format, Remarks	Examples
GroupName	This attribute identifies the list of groups which the subscriber belongs to.	<ul style="list-style-type: none">• Type: Directory String• Single-value attribute• Value range 1- 64 characters• Required: Mandatory	“group_a”
PolicyNameList	This attribute identifies the list of shared policies which applies to the user.	<ul style="list-style-type: none">• Type: Directory String• Multi-value attribute• Value range 1- 64 characters• Required: Optional	“123456789”

4.2

AA Entries for AAA FE (PKI)

4.2.1

PKI User Entry

This entry contains the PKI user data and it is a child of the mSC entry.



Table 12 Entry Name: PKI User

Attributes	Description	Format, Remarks	Examples
serv	<p>This attribute identifies the container entry for AA data.</p> <p>Defined in CUDB LDAP Interwork Description included in the object class CUDBServiceAuxiliary, see Reference [4].</p>	<ul style="list-style-type: none"> • Type: IA5 String • Value range: 1 - 32 characters • Value = "NSD" • Required: Mandatory 	"NSD"
UserName		<ul style="list-style-type: none"> • Type: Directory String • Single-value attribute • Value range: 1 - 255 characters • Required: Mandatory 	<p><IMSI>@<domain>:</p> <p>"240994004095@nai.epc.mnc015.mcc234.3gppnetwork.org"</p>
nsduserpwd		<ul style="list-style-type: none"> • Type: Directory String • Single-value attribute • Value range: 1 - 64 characters • Required: Optional 	"123456"
IMSI		<ul style="list-style-type: none"> • Type: Directory String • Single-value attribute • Value range: 1 - 15 characters • Required: Mandatory 	"240994004095"



Attributes	Description	Format, Remarks	Examples
MSISDN		<ul style="list-style-type: none">• Type: Directory String• Single-value attribute• Value range: 1 - 15 characters• Required: Mandatory	"13739944240"
apnlist		<ul style="list-style-type: none">• Type: Directory String• Multi-value attribute• Value range: 1 - 255 characters• Required: Mandatory	"MNC007.Mcc460 .3gppnetworks.org"
userstatus		<ul style="list-style-type: none">• Type: Directory String• Single-value attribute• Value range: 1 - 8 characters• Required: Mandatory	"ENABLE"



Attributes	Description	Format, Remarks	Examples
certificateissuern ame		<ul style="list-style-type: none"> • Type: Directory String • Single-value attribute • Value range: 1 - 255 characters • Required: Optional 	"CN=CA1, OU=A, O=Ericsson, ST=Shanghai, C=CN"
certificateid		<ul style="list-style-type: none"> • Type: Directory String • Single-value attribute • Value range: 1 - 255 characters • Required: Optional 	"91343852333181432387730302044767688728495783940"





5 Procedures

This section describes the operations used by the AAA FE (Radius and PKI) and Provisioning Gateway (PG) to communicate with the CUDB in order to handle AAA data.

The following LDAP operation are used:

- Add
- Delete
- Modify
- Search

These LDAP operations are described in Reference [6] and Reference [9] .

The PG is the client responsible for provisioning the CUDB Server node. It adds, modifies, and deletes entries in the CUDB Server, as well as performs any search operation the PG needs.

The AAA FE is the client responsible for traffic operations. It performs any search operation the AAA FE needs for traffic.

Subscriber-specific data operations from the PG are performed using the branch under `multiSCs` for specific multiple subscription data.

Subscriber common data operations from the PG are performed using the branch under `mscCommonData`.

The PG can perform following operations:

- Add new entries
- Delete entries
- Modify the attributes of an entry (replace, add, delete)
- Search for an entry

All operation from the AAA FE are performed using the branch under `identities` .

The AAA FE can perform the following operation under the branch `multiSCs`:

- Search for an entry



5.1 Procedures for AAA FE (Radius)

5.1.1 Creation and Deletion of Entries

5.1.1.1 Creating the Multiservice Consumer AA Entry

Missing information.

5.1.1.2 Creating the Service Common Data AA Entry

The Service Common Data AA entry is an entry of the AProfile structural object class and the CUDBServiceAuxiliary auxiliary object class defined in the CUDB data model (see Reference [4]).

To create the entry, an LDAP add operation must be performed.

Table 13 Creating the Service Common Data AA Entry

AddRequest	
Entry	dn: serv=AA, ou=mscCommonData, dc=<CUDB root entry>
Attributes	objectclass: top
	objectclass: CUDBService
	serv = "AA"

5.1.1.3 Creating the Service Common Data Groups Entry

To create the entry, an LDAP add operation must be performed.

Table 14 Creating the Service Common Data Groups Entry

AddRequest	
Entry	dn: ou=groups, serv=AA, ou=mscCommonData, dc=<CUDB root entry>
Attributes	objectclass: top
	objectclass: organizationalUnit
	ou = "groups"

5.1.1.4 Creating the Service Common Data Policies Entry

To create the entry, an LDAP add operation must be performed.



Table 15 Creating the Service Common Data Policies Entry

AddRequest	
Entry	dn: ou=policies, serv=AA, ou=mscCommonData, dc=<CUDB root entry>
Attributes	objectclass: top
	objectclass: organizationalUnit
	ou = "policies"

5.1.1.5

Creating the AA Profile Entry

The AA Profile entry is an entry of the AAProfile structural object class and the CUDBServiceAuxiliary auxiliary object class defined in the CUDB data model (see Reference [4]).

To create the entry, an LDAP add operation must be performed.

Table 16 Creating the AA Profile Entry

AddRequest	
Entry	dn:serv = AA, mscId = <multiserviceconsumer identity>, ou = multiSCs, dc=<CUDB root entry>
Attributes	serv=AA
	objectclass: top
	objectclass: CUDBServiceAuxiliary
	Objectclass: AAProfile
	UserName = <UserName>
	UserPassword = <password>
	AuthMethod = <authmethod>
	IPAllocType = <ipalloctype> ⁽¹⁾
	IPAllocValue = <ipallocvalue> ⁽²⁾
	IPv6PrefixAllocType = <ipv6alloctype> ⁽³⁾
	IPv6PrefixAllocValue = <ipv6allocvalue> ⁽⁴⁾
	GroupNameList = <groupname>
	PolicyNameList = <policyname>

(1) The <ipalloctype> can be set in the range of 0~3.

(2) The <ipallocvalue> can be set as a valid pool name or IPv4 address.

(3) The <ipv6alloctype> can be set in the range of 0~3.

(4) The <ipv6allocvalue> can be set as a valid pool name or IPv6 prefix.



5.1.1.6 Creating the AA Individual Policy Entry

The AA Individual Policy entry is an entry of the Policy structural object class defined in the CUDB data model (see Reference [4]).

To create the entry, an LDAP add operation must be performed.

Table 17 Creating the AA Individual Policy Entry

AddRequest	
Entry	dn: polycname = <polycname>, serv = AA, mscId = <multiserviceconsumer identity>, ou= multiSCs, dc = <CUDB root entry>
Attributes	objectclass: top
	objectclass: AAPolicy
	Polycname = <polycname>
	PolicyChecklist = <checklist>
	PolicyReplylist = <replylist>

5.1.1.7 Creating the AA Group Entry

The AA Group entry is an entry of the Group structural object class defined in the CUDB data model (see Reference [4]).

To create the entry, an LDAP add operation must be performed.

Table 18 Creating the AA Group Entry

AddRequest	
Entry	dn: GroupName = <groupname>, ou = Groups, serv = AA, ou = mscCommonData, dc = <CUDB root entry>
Attributes	objectclass: top
	objectclass: AAGroup
	Groupname = <groupname>
	PolicyNameList = <polycnamelist>

5.1.1.8 Creating the AA Group Alias Entry

To create the entry, an LDAP add operation must be performed.



Table 19 Creating the AA Group Alias Entry (1)

AddRequest	
Entry	dn: ei = "Group1", serv = AA, mscId = <multiserviceconsumer identity>, ou = multiSCs, dc = <CUDB root entry>
Attributes	objectclass: top
	objectclass: alias
	objectclass: CUDBExtensibleObject
	ei = "Group1"
	aliasedObjectName: GroupName = <groupname>, ou = Groups, serv = AA, ou=mscCommonData, dc = <CUDB root entry>

At the same time, the group name must be added to the GroupNameList attribute of the AA Profile.

Table 20 Creating the AA Group Alias Entry (2)

ModifyRequest	
Entry	serv = AA, mscId = <multiserviceconsumer identity>, ou = multiSCs, dc = <CUDB root entry>
Operation	Add
Modification	Attributes: GroupNameList = <groupname>

5.1.1.9

Creating the AA Shared Policy Entry

The AA Shared Policy entry is an entry of the Group structural object class defined in the CUDB data model (see Reference [4]).

To create the entry, an LDAP add operation must be performed.

Table 21 Creating the AA Shared Policy Entry

AddRequest	
Entry	dn: PolicyName = <policyname>, ou = Policies, serv=AA, ou=mscCommonData, dc = <CUDB root entry>
Attributes	objectclass: top
	objectclass: AAPolicy
	PolicyName = <policyname>
	PolicyChecklist = <checklist>
	PolicyReplylist = <replylist>



5.1.1.10

Creating the AA Shared Policy Alias Entry for User

To create the entry, an LDAP add operation must be performed.

Table 22 Creating the AA Shared Policy Alias Entry for User (1)

AddRequest	
Entry	dn: ei="Policy1", serv = AA, mscId = <multiserviceconsumer identity>, ou = multiSCs, dc = <CUDB root entry>
Attributes	objectclass: top
	objectclass: alias
	objectclass: CUDBExtensibleObject
	ei = "Policy1"
	aliasedObjectName: PolicyName = <polycname>, ou = Policies, serv=AA, ou = mscCommonData, dc = <CUDB root entry>

At the same time, the policy name must be added to the PolicyNameList attribute of the AAA User Profile.

Table 23 Creating the AA Shared Policy Alias Entry for User (2)

ModifyRequest	
Entry	serv = AA, mscId = <multiserviceconsumer identity>, ou = multiSCs, dc = <CUDB root entry>
Operation	Add
Modification	Attributes: PolicyNameList = <polycname>

5.1.1.11

Creating the AA Shared Policy Alias Entry for Group

To create the entry, an LDAP add operation must be performed.

Table 24 Creating the AA Shared Policy Alias Entry for Group (1)

AddRequest	
Entry	dn: ei= "Policy1", GroupName = <groupname>, ou= Groups, serv = AA, ou = mscCommonData, dc = <CUDB root entry>



Attributes	objectclass: top
	objectclass: alias
	objectclass: CUDBExtensibleObject
	ei = "Policy1"
	aliasedObjectName: PolicyName = <polycname>, ou = Policies, serv = AA, ou = mscCommonData, dc = <CUDB root entry>

At the same time, the policy name must be added to the PolicyNameList attribute of the AAA Group.

Table 25 Creating the AA Shared Policy Alias Entry for Group (2)

ModifyRequest	
Entry	GroupName = <groupname>, ou = Groups, serv=AA, ou = mscCommonData, dc=<CUDB root entry>
Operation	Add
Modification	Attributes: PolicyNameList = <polycname>

5.1.1.12 Deleting the AA User Profile Entry

To delete the entry, an LDAP delete operation must be performed.

Table 26 Deleting the AA User Profile Entry

DelRequest	dn: serv = AA, mscId = <multiserviceconsumer identity>, ou = multiSCs, dc = <CUDB root entry>
------------	---

5.1.1.13 Deleting the AA Group Entry

To delete the entry, an LDAP delete operation must be performed.

Table 27 Deleting the AA Group Entry

DelRequest	dn: GroupName = <groupname>, ou=Groups, serv=AA, ou= mscCommonData, dc = <CUDB root entry>
------------	--

5.1.1.14 Deleting the AA Shared Policy Entry

To delete the entry, an LDAP delete operation must be performed.



Table 28 Deleting the AA Shared Policy Entry

DelRequest	dn: PolicyName = <polycyname>, ou = Policies, serv=AA, ou=mscCommonData, dc = <CUDB root entry>
------------	---

5.1.1.15 Deleting the AA Individual Policy Entry

To delete the entry, an LDAP delete operation must be performed.

Table 29 Deleting the AA Individual Policy Entry

DelRequest	dn: PolicyName = <polycyname>, serv = AA, mscId = <multiserviceconsumer identity>, ou = multiSCs, dc = <CUDB root entry>
------------	--

5.1.1.16 Deleting the AA Group Alias Entry

To delete the entry, an LDAP delete operation must be performed.

Table 30 Deleting the AA Group Alias Entry

DelRequest	dn: ei="Group1", serv = AA, mscId = <multiserviceconsumer identity>, ou = multiSCs, dc = <CUDB root entry>
------------	--

5.1.1.17 Deleting the AA Shared Policy Alias Entry

To delete the entry, an LDAP delete operation must be performed.

Table 31 Deleting the AA Shared Policy Alias Entry

DelRequest	dn: ei="Policy1", serv = AA, mscId = <multiserviceconsumer identity>, ou = multiSCs, dc = <CUDB root entry>
------------	---

5.1.2 AAA Data Modification

This section includes examples of AAA data modification.

5.1.2.1 AAA User Profile Modification (Replace) Using multiSCs Branch

AAA subscriber data is an entry of the AProfile object class.

The following table presents an example of possible AA subscriber data modification, where the authentication method is changed for a subscriber. In this example, an LDAP replace operation is performed.



Table 32 AAA User Profile Modification (Replace) Using multiSCs Branch

ModifyRequest	dn: serv = AA, mscId = <multiserviceconsumer identity>, ou = multiSCs, dc = <CUDB root entry>
Operation	replace
Modification	attributetype: AuthMethod attributevalue: <AuthMethod value>

5.1.2.2

AAA Alias Modification (Replace) Using multiSCs Branch

The AAA alias is an entry of the `alias` object class.

The following table presents an example of possible Group alias modification, where a new group is assigned to an AA subscriber. In this example, an LDAP replace operation is performed.

Table 33 AAA Alias Modification (Replace) Using multiSCs Branch - Example 1

ModifyRequest	dn: ei = "Group1", serv = AA, mscId = <multiserviceconsumer identity>, ou = multiSCs, dc = <CUDB root entry>
Operation	replace
Modification	attributetype: aliasedObjectName attributevalue: GroupName = <groupname>, ou=groups, serv=AA, ou=mscCommonData, dc = <CUDB root entry>

The following table presents an example of possible Policy alias modification, where a new policy is assigned to an AA subscriber. In this example, an LDAP replace operation is performed.

Table 34 AAA Alias Modification (Replace) Using multiSCs Branch - Example 2

ModifyRequest	dn: ei = "Policy1", serv = AA, mscId = <multiserviceconsumer identity>, ou = multiSCs, dc = <CUDB root entry>
Operation	replace
Modification	attributetype: aliasedObjectName attributevalue: PolicyName = <polycyname>, ou=Policies, serv=AA, ou=mscCommonData, dc = <CUDB root entry>

5.1.2.3

AAA Subscriber Data Modification (Delete) Using multiSCs Branch

AAA Profile data is an entry of the `AAProfile` object class.

The following table presents an example of possible AAA Profile data modification, where the `AuthMethod` attribute or one of the shared policies is removed. In this example, an LDAP delete operation is performed.



Table 35 AAA Subscriber Data Modification (Delete) Using multiSCs Branch

ModifyRequest	dn:serv = AA, mscId = <multiserviceconsumer identity>, ou = multiSCs, dc = <CUDB root entry>
Operation	delete
Modification	attributetype: AuthMethod attributevalue: <AuthMethod value>
	attributetype: PolicyNameList attributevalue: <PolicyNameList value>

5.1.2.4

AA Individual Policy Data Modification (Replace)

The following table presents an example of possible AAA Individual Policy data modification, where the PolicyChecklist attribute is updated. In this example, an LDAP replace operation is performed.

Table 36 AA Individual Policy Data Modification (Replace)

ModifyRequest	dn: PolicyName = <policyname>, serv = AA, mscId = <multiserviceconsumer identity>, ou = multiSCs, dc = <CUDB root entry>
Operation	replace
Modification	attributetype: PolicyReplylist attributevalue: <PolicyReplylist value>
	attributetype: PolicyChecklist attributevalue: <PolicyChecklist value>

5.1.2.5

AAA Multiservice Consumer Common Data Modification (Replace)

The following table presents an example of possible AAA Group data modification, where a new Policy is assigned to the group. In this example, an LDAP replace operation is performed.

Table 37 AAA Multiservice Consumer Common Data Modification (Replace) - Example 1

ModifyRequest	dn: GroupName = <groupname>, ou = groups, serv = AA, ou = mscCommonData, dc = <CUDB root entry>
Operation	replace
Modification	attributetype: PolicyNamelist
	attributevalue: <PolicyNamelist value>



The following table presents an example of possible AAA Shared Policy data modification, where the PolicyChecklist attribute is updated. In this example, an LDAP replace operation is performed.

Table 38 AAA Multiservice Consumer Common Data Modification (Replace) - Example 2

ModifyRequest	dn: PolicyName= <polycname>, ou = groups, serv = AA, ou = mscCommonData, dc = <CUDB root entry>
Operation	replace
Modification	attributetype: PolicyReplylist attributevalue: <PolicyReplylist value>
	attributetype: PolicyChecklist attributevalue: <PolicyChecklist value>

5.1.2.6

AAA Multiservice Consumer Common Data Modification (Delete)

The following table presents an example of possible AAA group data modification, where one of the policies is removed from the group. In this example, an LDAP delete operation is performed.

Table 39 AAA Multiservice Consumer Common Data Modification (Delete) - Example 1

ModifyRequest	dn: GroupName = <groupname>, ou = groups, serv=AA, ou=mscCommonData, dc = <CUDB root entry>
Operation	delete
Modification	attributetype: PolicyNamelist
	attributevalue: <PolicyNamelist value>

The following table presents an example of possible policy data modification, where one of the policy checklists is removed. In this example, an LDAP delete operation is performed.

Table 40 AAA Multiservice Consumer Common Data Modification (Delete) - Example 2

ModifyRequest	dn: PolicyName = <polycname>, ou=policies, serv=AA, ou=mscCommonData, dc = <CUDB root entry>
Operation	delete
Modification	attributetype: PolicyChecklist
	attributevalue: <PolicyChecklist value>



5.1.3 AAA Subscriber Data Search

This section describes how to search for AAA subscriber data.

5.1.3.1 Subscriber Common Data Search

Subscriber common data is an entry of the AA object class. To search the subscriber common data, an LDAP search operation must be performed.

It is possible to request the following:

- All AAA Group data
- A specific type of AAA Group data (all attributes in a specific group)
- All AAA Policy data
- A specific type of AAA Policy data (all attributes in a specific policy)

5.1.3.1.1 Searching for All AAA Group Data

AAA Group data is an entry of the AAGroup structural object class. To search for AAA Group data, an LDAP search operation must be performed.

Searching for all AAA Group data can be done from the Groups entry with the scope set to “singleLevel”.

Table 41 Searching for All AAA Group Data

SearchRequest	
BaseObject	dn: ou = groups, serv = AA, ou = mscCommonData, dc = <CUDB root entry>
Scope	singleLevel
DerefAliases	neverDerefAliases
Filter	(objectclass=*)
Attributes	NULL

5.1.3.1.2 Searching for Specific Type of Group Data

Searching for a specific type of Group data can be performed from the GroupName entry with the scope set to “baseObject”.

Table 42 Searching for Specific Type of Group Data

SearchRequest	
BaseObject	dn: GroupName = <groupname>, ou = groups, serv = AA, ou = mscCommonData, dc = <CUDB root entry>



Scope	baseObject
DerefAliases	neverDerefAliases
Filter	(objectclass=*)
Attributes	NULL

5.1.3.1.3 Searching for All AAA Policy Data

AAA Policy data is an entry of the AAPolicy structural object class. To search for AAA Policy data, an LDAP search operation must be performed.

Searching for all AAA Policy data can be performed from the Policies entry with the scope set to “singleLevel”.

Table 43 Searching for All AAA Policy Data

SearchRequest	
BaseObject	dn: ou = policies, serv = AA, ou = mscCommonData, dc = <CUDB root entry>
Scope	singleLevel
DerefAliases	neverDerefAliases
Filter	(objectclass=*)
Attributes	NULL

5.1.3.1.4 Searching for Specific Type of Policy Data

Searching for a specific type of policy data can be performed from the PolicyName entry with the scope set to “baseObject”.

Table 44 Searching for Specific Type of Policy Data

SearchRequest	
BaseObject	dn: PolicyName= <polycynname>, ou = policies, serv = AA, ou = mscCommonData, dc = <CUDB root entry>
Scope	baseObject
DerefAliases	neverDerefAliases
Filter	(objectclass=*)
Attributes	NULL

5.1.3.2 Subscriber Data Search

Subscriber data is an entry of the AA object class. To search the subscriber data, an LDAP search operation must be performed.



It is possible to request the following:

- All data of a subscriber
- Individual subscriber data of a subscriber
- List of subscribers
- List of subscribers with specific data

5.1.3.2.1 Searching for All Data of a Subscriber

AAA subscriber data is an entry of the `AAProfile` structural object classes. To search for AAA subscriber data, an LDAP search operation must be performed.

Searching for all data of a subscriber can be performed either from the `multiSCs` branch or from the `identities` branch with the scope set to “`wholeSubtree`”.

The following table presents how to search for all data of a subscriber from the `multiSCs` entry.

Table 45 Searching for All Data of a Subscriber from `multiSCs` Entry

SearchRequest	
BaseObject	dn: serv = AA, mscId = <multiserviceconsumer identity>, ou = multiSCs, dc = <CUDB root entry>
Scope	wholeSubtree
DerefAliases	derefAlways
Filter	(objectclass=*)
Attributes	NULL

The following table presents how to search for all data of a subscriber from the `identities` entry.

Table 46 Searching for All Data of a Subscriber from `identities` Entry

SearchRequest	
BaseObject	dn: serv=AA, username = <UserName>, dc =UserName, ou = identities, dc = <CUDB root entry>
Scope	wholeSubtree
DerefAliases	derefAlways
Filter	(objectclass=*)
Attributes	NULL



5.1.3.2.2 Searching for Individual Subscriber Data of a Subscriber

AAA individual subscriber data is an entry of the `AAProfile` structural object class. To search for AAA individual subscriber data, an LDAP search operation must be performed.

Searching for the individual subscriber data of a subscriber can be performed either from the `multiSCs` branch or from the `identities` branch with the scope set “`towholeSubtree`”.

The following table presents how to search for the individual subscriber data of a subscriber from the `multiSCs` entry.

Table 47 Searching for Individual Subscriber Data of a Subscriber from “`multiSCs`” Entry

SearchRequest	
BaseObject	dn: serv = AA, mscId = <multiserviceconsumer identity>, ou = multiSCs, dc = <CUDB root entry>
Scope	baseObject
DerefAliases	neverDerefAliases
Filter	(objectclass=*)
Attributes	NULL

The following table presents how to search for the individual subscriber data of a subscriber from the `identities` entry.

Table 48 Searching for Individual Subscriber Data of a Subscriber from “`identities`” Entry

SearchRequest	
BaseObject	dn: serv = AA, username = <UserName>, dc =UserName, ou = identities, dc = <CUDB root entry>
Scope	singleLevel
DerefAliases	derefAlways
Filter	(objectclass=*)
Attributes	NULL

5.1.3.2.3 Searching for a List of Subscribers

To search for a list of AAA subscribers, an LDAP search operation must be performed.

Searching for a list of subscribers can be performed from the `subscribers` branch with the scope set to “`wholeSubtree`”. The filter must be set to “`objectclass=AAProfile`” and include the value range. In the example below,



the value range is “UserName = User*”, where searching is performed for all subscribers whose username begins with “User”.

Table 49 Searching for a List of Subscribers

SearchRequest	
BaseObject	ou = multiSCs, dc = <CUDb root entry>
Scope	wholeSubtree
DerefAliases	neverDerefAliases
Filter	(& (objectclass=AAProfile) (UserName = User*))
Attributes	UserName

5.1.3.2.4

Searching for a List of Subscribers with Specific Data

To search for a list of AAA subscribers with specific data, an LDAP search operation must be performed.

Searching for all subscribers with a specific attribute or a set of attributes can be performed from the multiSCs entry with the scope set “toWholeSubtree”. The attributevalueassertion must be set to the specific attribute for which searching is performed. In the example below, searching is performed for all subscribers provisioned with the EAP-MD5 authentication method.

Table 50 Searching for a List of Subscribers with Specific Data

SearchRequest	
BaseObject	ou = multiSCs, dc=<CUDb root entry>
Scope	wholeSubtree
DerefAliases	neverDerefAliases
Filter	(& (objectclass=AAProfile) (AuthMethod = “eap-md5”))
Attributes	AuthMethod

5.2 Procedures for AAA FE (PKI)

5.2.1 Creation and Deletion of Entries

5.2.1.1 Creating the PKI User Entry

To create the entry, an LDAP add operation must be performed.



Table 51 Creating the PKI User Entry

AddRequest	
Entry	dn:serv = NSD, mscId = <multiserviceconsumer identity>, ou = multiSCs, dc = <CUDB root entry>
Attributes	serv=NSD
	objectclass: top
	objectclass: CUDBServiceAuxiliary
	objectclass: nsduser
	UserName = <UserName>
	nsduserpwd = <password>
	IMSI = <imsi>
	MSISDN = <msisdn>
	apnlist = <apn>
	userstatus = <userstatus>
	certificateissuername = <certificateissuername>
	certificateid = <certificateid>

5.2.1.2 Deleting the PKI User Entry

To delete the entry, an LDAP delete operation must be performed.

Table 52 Deleting the AA User Profile Entry

DelRequest	dn: serv = NSD, mscId = <multiserviceconsumer identity>, ou = multiSCs, dc = <CUDB root entry>
------------	--

5.2.2 PKI Data Modification

This section includes examples of PKI data modification.

PKI subscriber data is an entry of the nsduser object class.

5.2.2.1 PKI User Modification (Replace) Using multiSCs Branch

The following table presents an example of possible PKI subscriber data modification, where the user status is changed for a subscriber. In this example, an LDAP replace operation is performed.



Table 53 PKI User Modification (Replace) Using multiSCs Branch

ModifyRequest	dn: serv = NSD, mscId = <multiserviceconsumer identity>, ou = multiSCs, dc = <CUDB root entry>
Operation	replace
Modification	attributetype: userstatus attributevalue: <userStatus value>

5.2.2.2

PKI Subscriber Data Modification (Delete) Using multiSCs Branch

The following table presents an example of possible PKI subscriber data modification, where the nsduserpwd attribute or one of the apnlist is removed. In this example, an LDAP delete operation is performed.

Table 54 PKI Subscriber Data Modification (Delete) Using multiSCs Branch

ModifyRequest	dn:serv = NSD, mscId = <multiserviceconsumer identity>, ou = multiSCs, dc = <CUDB root entry>
Operation	delete
Modification	attributetype: nsduserpwd attributevalue: <nsduserpwd value>
	attributetype: apnlist attributevalue: <apnlist value>

5.2.3

PKI Subscriber Data Search

The following table presents how to search for the individual subscriber data of a subscriber from the identities entry.

Table 55 Searching for Individual Subscriber Data of a Subscriber from "identities" Entry

SearchRequest	
BaseObject	dn: serv = NSD, IMSI = <IMSI>, dc = IMSI, ou = identities, dc = <CUDB root entry>
Scope	singleLevel
DerefAliases	derefAlways
Filter	(objectclass=*)
Attributes	NULL



6 Related Standards

See section References.





Reference List

IPWorks Library Documents

- [1] Glossary of Terms and Acronyms
- [2] Trademark Information
- [3] Typographic Conventions

PCAT and Other Ericsson Documents

- [4] CUDB LDAP Interwork Description, 1/15519-HDA 104 03/2

Standards

- [5] [Lightweight Directory Access Protocol \(LDAP\): Syntaxes and Matching Rules RFC 4517](#)
- [6] [Lightweight Directory Access Protocol \(LDAP\): The Protocol RFC 4511](#)
- [7] [Lightweight Directory Access Protocol \(LDAP\): Directory Information Models RFC 4512](#)
- [8] [Lightweight Directory Access Protocol \(LDAP\): Schema for User Applications RFC 4519](#)
- [9] [Lightweight Directory Access Protocol \(LDAP\): Modify-Increment Extension RFC 4525](#)