

User Management

DESCRIPTION

Copyright

© Ericsson AB 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

All trademarks mentioned herein are the property of their respective owners. These are shown in the document Trademark Information.



Contents

1	Introduction	1
2	Functions and Concepts	3
2.1	User Authentication	4
2.2	User Authorization	5
2.3	Backup and Restore of Local User Data	7
2.4	Types of Operation	7
3	Managed Object Model	13
3.1	Rules for Default Roles	14
4	Configuration Management	21
5	MTAS Roles and Rules	23





1 Introduction

This document provides an overview of the management model and concepts associated with the User Management managed area.

A managed area is represented by a group of Managed Object Classes (MOCs) within the Managed Object Model (MOM).





2 Functions and Concepts

User Management provides a management interface to configure the following on the Managed Element (ME):

- Local user authentication
- Lightweight Directory Access Protocol (LDAP) authentication
- Local authorization for maintaining local Policy Information Point (PIP)

An overview of User Management is shown in Figure 1.

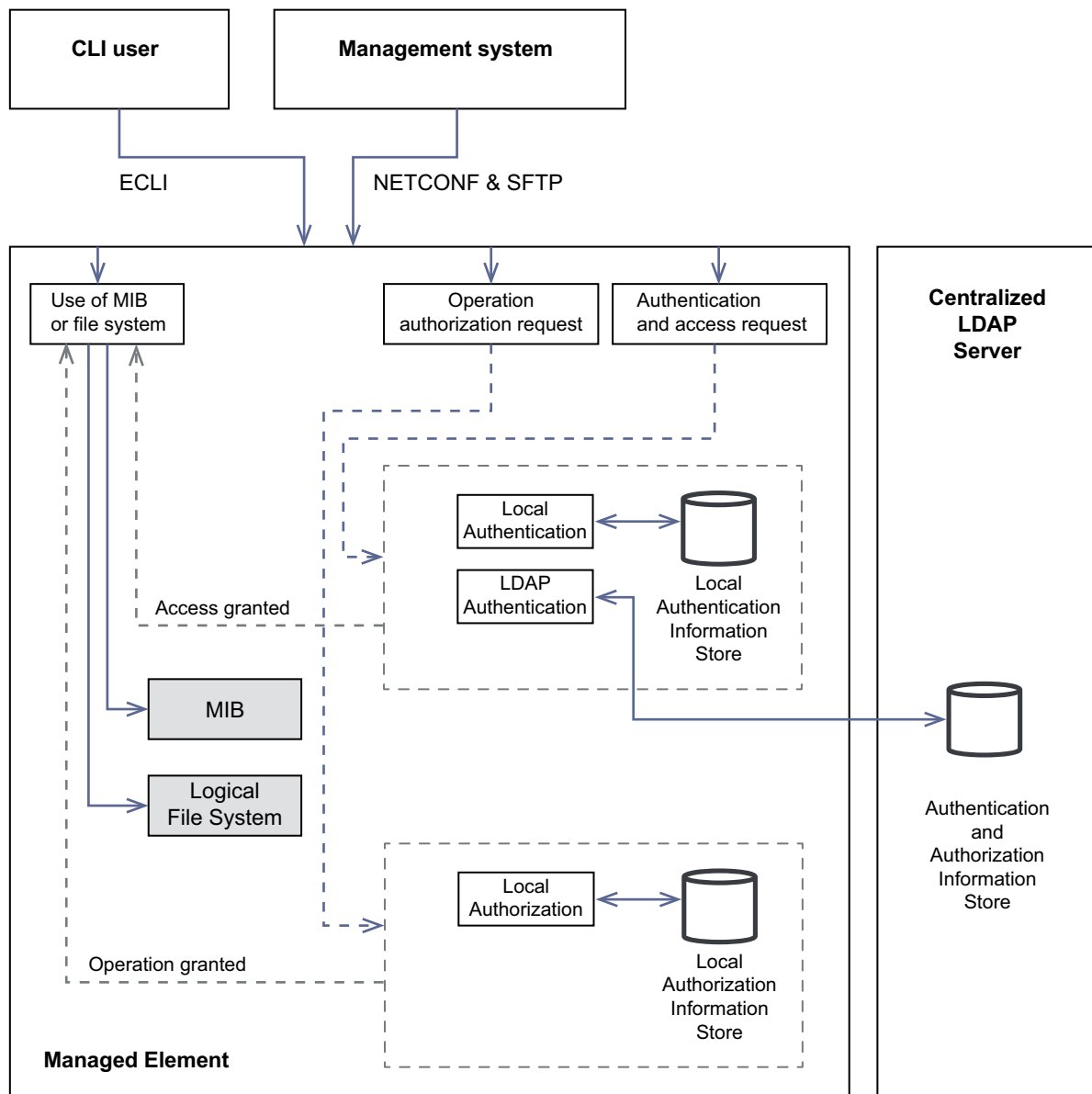


Figure 1 User Management Overview



This document assumes that the ME has already been installed and initially configured. The initial configuration includes the necessary settings for the authentication and authorization of users.

Authentication is used for checking user credentials and user access. Role-Based Access Control (RBAC) authorization is used to ensure correct user access privileges. The ME supports management of local users and authentication and supports the LDAP protocol for centralized user authentication. For centralized authentication, Target-Based Access Control (TBAC) can be applied over RBAC. Authentication and authorization are performed according to the organization authorization policy.

The local authentication method is always available to ensure that the operator cannot be inadvertently denied access to the managed element. It is recommended to create enough local accounts to mitigate connectivity issues to centralized authentication. The managed element supports centralized authentication by the LDAP protocol. Centralized authentication is preferred for daily operations to keep a consistent user base over a network of managed elements.

The local authentication method is always performed. If local authentication fails to find a user, the authentication continues with centralized LDAP authentication. The order of authentication methods cannot be changed.

For more information on the LDAP interface, refer to *LDAP-Based Authentication and Authorization Interface*.

2.1 User Authentication

The user initiates a session which triggers user authentication. For the authentication to be successful, a user account must be configured either locally by Local Authentication, or centrally in an external LDAP server. The first configured account is the Local Authentication administrator, which is defined at site deployment.

When adding user accounts, naming must serve as a unique identity. Naming collisions can result in unexpected authentication behavior, as the user trying to authenticate with that name is mapped to the account first found with that name. The operator must ensure that usernames are globally unique, in the scope of both local and central authentication, to match expected authentication behavior.

In centralized LDAP authentication, a primary and a secondary LDAP server is supported. The LDAP authentication first tries against the primary server and then the secondary server.

All authentication attempts, whether successful or not, are recorded in the ME security log.

For more information, refer to *Audit Information*.



A successful user authentication triggers a user authorization.

2.2 User Authorization

Before user authorization occurs, the ME queries the roles of the users.

For local users, the roles are stored in the user account configuration.

For LDAP, the ME performs additional checks whether the user can access the ME based on POSIX parameters (uidNumber) and TBAC attribute of the user account. For roles, three different LDAP profile filters are supported for search; flexible filter, POSIX groups filter, and Ericsson roles filter. The Ericsson roles profile filter is used together with the Ericsson Operations Support System (OSS) solution. When the Ericsson roles profile filter is used, the authorization can be selective based on the target type. In some networks, it can be required to let a user have different management roles on different MEs. For example, the network can span several countries, and it can be needed to let a user act as “admin” in one country, but only as “operator” in another. This function is part of the TBAC functionality.

The user access rights depend on defined authorization rules that specify the permissions to a set of resources within the ME. The authorization rules are grouped into roles. A role is equivalent to the user occupation within an organization, for example, system administrator. A user can have one or more roles.

The ME supports some predefined roles, see Section 2.2.2 Default Roles on page 6. Custom roles can also be configured over the Northbound Interface (NBI).

The authorization rules are all defined locally on the ME. Therefore, the user authorization is a local authorization. Custom rules corresponding to customer roles can be configured over the NBI.

Authorization rules provide different access levels to the MIB and the ECLI commands. Authorization rules are defined by permission types, see Section 2.2.1 Permission Types on page 5.

2.2.1 Permission Types

Rules for access can be specified for Managed Objects (MOs), their attributes and actions. The execution of the ECLI commands and the NETCONF operations is not subject to authorization. However, the rules affect the result of the ECLI commands and the NETCONF operations that operate on MOs.

Permission types define different levels of access to the MIB according to Table 1.



Table 1 Permission Types

Permission Type	Description
No access (NO_ACCESS)	The user has no read, write, or execute rights to the MOs, attributes, or actions
Execute (X)	The user can execute all actions in the MOM
Read (R)	The user can read MOs and get attribute values
Read and execute (RX)	The user can read MOs, get attribute values, and execute all actions in the MOM
Read and write (RW)	The user can create and delete MOs as well as get and set attribute values
Read, write, and execute (RWX)	The user can create and delete MOs, set, and get attribute values, as well as execute all actions in the MOM

When a user with an authorization profile wants to access resources of the ME, the access request is authorized against matching security rules. The rules are checked in the following order:

- 1 All negative rules (with the NO_ACCESS permission) are evaluated. If a match is found, access is denied.
- 2 All positive rules (with X, R, RX, RW, and RWX permissions) are evaluated until a match is found; the corresponding access is granted. If no match is found, access is denied.

2.2.2 Default Roles

The ME supports the predefined default roles described in Table 2. These roles and the corresponding rules cannot be modified. The detailed permissions for each role are described in Section 3.1 Rules for Default Roles on page 14.

Default permissions to the ME are granted automatically to all users and are expressed through the role named “Self”.

Table 2 Default Roles

Default Role	Description
Self	Default authorization
System Administrator	Responsible for the administration of all non-security-related attributes and capabilities of an ME, including features, configuration parameters, and monitoring
Local Authentication Administrator	Responsible for the administration of the local user accounts. Dedicated to the Administrator Account to limit its use



Table 2 Default Roles

Default Role	Description
System Security Administrator	Responsible for the administration of all security-related attributes and capabilities of an ME, including user accounts and authorizations
System Read Only	Can view most non-security-related attributes and capabilities of an ME, including features, configuration parameters, and monitoring
Managed Function Application Administrator	Responsible for the administration of all non-security-related attributes and capabilities of the Managed Function, including features, configuration parameters, and monitoring
Managed Function Application Security Administrator	Responsible for the administration of all security-related attributes and capabilities of the Managed Function, including user accounts and authorizations
Managed Function Application Operator	Can view some non-security-related attributes and capabilities of the Managed Function, including features, configuration parameters, and monitoring

2.3 Backup and Restore of Local User Data

Information related to user accounts expires over time. Backup taken of configuration data could have been done at a time when different accounts were used. The users can also have changed their passwords after the configuration data backup. This can result in a state after restoration where the valid users before the backup restoration cannot access the ME after the restoration, because of expired passwords or invalid accounts, or can access the ME with different authorization.

It is strongly recommended to create a backup immediately after any change in the password of the administrator account. It is also recommended to create regular backups of User Account information frequently.

User account information classifies as system data in backup and restore procedures.

2.4 Types of Operation

User Management supports the following operations for an administrator with the System Security Administrator role. The local administrator account defined in Local Authentication can access the local authentication operations.



General

- Configure legal notice

The legal notice presented before user authentication on certain interfaces can be changed to comply to domestic legal requirements. The procedure *Configure Legal Notice* provides further details on how to perform this operation. Also refer to the appropriate documentation of the interface to learn if the legal notice is applicable.

- Change Login Failure Delay

The delay after a failed password logon attempt can be changed. The procedure *Configure Login Failure Delay* provides further details on how to perform this operation.

Local Authentication

Local authentication operations must be used if the system does not support centralized authentication, or to configure centralized authentication and to define fallback accounts, in case the centralized user management service becomes inaccessible.

- Create, change, and delete user account

An O&M user account can be created and modified to give access to the system. It includes a username and a password or an SSH public key used for identification and authorization. The procedures in *Create User Account*, *Change User Account*, and *Delete User Account* provide further details on how to perform these operations. SSH public key management is described by procedures in *Create SSH Public Key*, *Change SSH Public Key*, and *Delete SSH Public Key*.

- Create, change, and delete account policy

The purpose of an account policy is to limit the accessibility of unused accounts. Account policies can be created and modified. The account policy setting locks an account if the account dormant time is set to be measured and the account dormant time runs out. All non-password related properties of user account are associated with account policy. The procedures in *Create Account Policy*, *Change Account Policy*, and *Delete Account Policy* provide further details on how to perform these operations.

- Create, change, and delete password policy

Security and usability with passwords are achieved by password management policies and the possibility to enforce strong passwords. The procedures in *Create Password Policy*, *Change Password Policy*, and *Delete Password Policy* provide further details on how to perform these operations. Strong passwords must be chosen to prevent brute-force password attacks. The procedure in *Change Password Quality Configuration* provides further details on how to perform this operation.



- Reset password for user account

A reset password operation must be performed by the administrator when the user account is locked because of the password expiry. The procedure in *Reset Password for User Account* provides further details on how to perform this operation.

- Remove password from user account

The password can be removed if the user account is configured to use key based authentication. This removal of the password liberates the user account from password management requirements. The procedure *Remove Password from User Account* provides further details on how to perform this operation.

- Set user roles for user account

A user account is assigned one or several roles to provide the access to control the node resources. For instance, the node resources can be the MO tree, CLI commands, or NETCONF operations. The procedure in *Set User Roles for User Account* provides further details on how to perform this operation.

- Lock and unlock user account

The administrator can lock and unlock a user account. In managing the user access, the user can be locked out by the administrator, for example, if the user for some reason no longer is approved for having access. The procedures in *Lock User Account Administratively*, and *Unlock Administrative Lock for User Account* provide further details on how to perform these operations. A user account can also be locked by system, which can be unlocked by administrator. The reasons for a user account to be locked by the system could be, for example, because of too long user inactivity. The procedure in *Unlock Operational Lock for User Account* provides further details on how to perform this operation.

- Change the alarm configuration for the administrator account

The specific Administrator account cannot be locked. As a measure to detect irregular logon activity to this account, the account can emit an alarm if the alarming threshold is reached. The number of failure attempts as a threshold can be configured. The procedure in *Change Administrator Account* provides further details on how to perform this operation.

LDAP Authentication

LDAP authentication must be configured if there is a centralized user management service accessible with the LDAP protocol. For security, deploying it with TLS is highly recommended.

- View LDAP configuration

The administrator can check the current LDAP configuration. The understanding of the LDAP configuration is a prerequisite for solving any authentication issues. The procedure in *View LDAP Configuration* provides further details on how to perform this operation.

- Unlock/lock LDAP authentication method

In maintenance situations, the administrator can lock the LDAP authentication to prevent users from accessing the ME, when it is not fully operational. When the LDAP authentication method is locked, only local authentication and emergency access to the MIB is possible. The procedure in *Lock LDAP Authentication Method* provides further details on how to perform this operation.

The administrator unlocks the LDAP authentication to enable user LDAP authentication when the ME is operational or to test the proper execution of LDAP authentication. The procedure in *Unlock LDAP Authentication Method* provides further details on how to perform this operation.

- Configure LDAP basic connection

To get a clear text unsecure connection to an LDAP authentication server, the IP address and the port number of the server must be configured. Search operations to the server require a base Distinguished Name. All LDAP user object must be accessible from this DN. Optionally a fallback IP address can be configured.

The procedure in *Configure LDAP Basic Connection* provides further details on how to perform this operation.

It is strongly recommended to secure the LDAP connection by using TLS.

- Configure bind name and password for LDAP authentication

The administrator can configure the bind name and password required for password-based simple bind LDAP authentication. The change of bind name and password can also be triggered by the organization security policy. The procedure in *Configure LDAP Simple Bind* provides further details on how to perform this operation.

- Configure LDAP authorization filter

LDAP authorization to get roles needs an authorization filter to be set up.

The ME supports the following authorization filter types:

- Ericsson filter, built-in LDAP filter that allows for RBAC and TBAC.
- POSIX filter, standard POSIX group filter which treats groups as RBAC roles.
- Flexible filter, which allows for interpreting an arbitrary attribute of an arbitrary object as RBAC role.



Only one filter type can be selected, and the recommended alternative is the Ericsson filter.

The procedures in *Configure Ericsson LDAP Filter*, *Configure POSIX LDAP Filter*, and *Configure Flexible LDAP Filter* provide further for performing these operations.

- **Configure TLS for LDAP**

The administrator needs to install certificates for TLS. For server only authentication, a trust category is required; for mutual authentication, a node credential must also be deployed. For the information on how to deploy certificates, refer to *Certificate Management*.

The administrator needs to change the certificate settings for LDAP TLS in the following situations:

- The ME node credential for LDAP TLS has been reinstalled by certificate management.
- Another trust category for LDAP TLS must be used.

The administrator may need to change the default cipher suite for changed security requirements or compatibility with network peers.

The procedures in *Configure TLS for LDAP* and *SSH and TLS Protocol Management* provide further details on how to perform this operation.

- **Configure Target-Based Access Control (TBAC)**

The administrator needs to change the TBAC settings when the current settings no longer match the operator organization needs, for example, in the following situations:

- The ME needs to become part of a different geographical domain.
- The ME needs to become part of a different functional domain.
- The ME needs to become part of a different competence domain.

The procedure in *Configure Target-Based Access Control* provides further details on how to perform this operation.

Local Authorization

Local authorization must be used to understand the default roles the product delivers, and using roles in assigning authorization for users. Customization of roles and rules are possible by adding extra roles over the default ones.

- **View roles and rules**

The administrator can view the roles retrieved from the LDAP server and the rules defined in the ME. The understanding of the roles and rules is a



prerequisite for solving any authorization issues. The procedure in *View Roles and Rules* provides further details on how to perform this operation.

- Lock/unlock local authorization method

The administrator locks the local authorization to give full access to all resources to all users authenticated by LDAP. Locking can be done in maintenance situations. The procedure in *Lock Local Authorization Method* provides further details on how to perform this operation.

The administrator unlocks the local authorization to enable the local authorization based on defined rules and roles when the ME is operational or to test the proper execution of local authorization. The procedure in *Unlock Local Authorization Method* provides further details on how to perform this operation.

- Create, change, and delete custom roles and custom rules

The administrator can create or change custom roles and custom rules when the predefined roles and rules do not match the needs of the organization authorization policy. The procedures in *Create Custom Role*, *Change Custom Role*, *Create Custom Rule*, and *Change Custom Rule* provide further details on how to perform these operations.

The administrator can delete custom roles and custom rules when they are no longer needed by the organization authorization policy. The procedures in *Delete Custom Role* and *Delete Custom Rule* provide further details on how to perform these operations.

3 Managed Object Model

The User Management managed area is represented in the *Managed Object Model (MOM)* as follows:

```
ManagedElement
+-SystemFunctions
+-SecM
+-UserManagement
+-AuthenticationMethodOrder
+-LdapAuthenticationMethod
+-Ldap
+-EricssonFilter
+-Filter
+-LocalAuthenticationMethod
+-AccountPolicy
+-AdministratorAccount
+-SshPublicKey
+-PasswordPolicy
+-PasswordQuality
+-UserAccountM
+-UserAccount
+-SshPublicKey
+-LocalAuthorizationMethod
+-CustomRole
+-CustomRule
+-Role
+-Rule
```

For general information about the MOM, MOCs, MOs, cardinality, and related concepts, refer to *Managed Object Model User Guide*.

The User Management MOCs are described in Table 3.

Table 3 User Management Managed Object Class Descriptions

Managed Object Class	Description
<i>SecM</i>	The root of the Security Management model
<i>UserManagement</i>	Describes the ME target types for TBAC
<i>LdapAuthenticationMethod</i>	Handles the LDAP authentication method used to verify user credentials when attempting to log on to an ME
<i>Ldap</i>	Handles the primary and secondary LDAP servers
<i>EricssonFilter</i>	Defines the configuration used for the Ericsson filter (applicable when the value of <i>profileFilter</i> is ERICSSON_FILTER)



Table 3 User Management Managed Object Class Descriptions

Managed Object Class	Description
<i>Filter</i>	Defines the configuration used for the flexible filter (applicable when the value of <i>profileFilter</i> is FLEXIBLE)
<i>LocalAuthenticationMethod</i>	Handles the local authentication method used to verify user credentials when attempting to log on to an ME
<i>AccountPolicy</i>	Handles the account policies
<i>AdministratorAccount</i>	Handles the administrator account
<i>PasswordPolicy</i>	Handles the password policies
<i>PasswordQuality</i>	Handles the password quality parameters
<i>UserAccountM</i>	Contains all the local user accounts created by an administrator
<i>UserAccount</i>	Describes an individual user account, and related policies, credentials, and relevant password monitoring
<i>LocalAuthorizationMethod</i>	Handles the local authorization method used to verify the user access to the ME resources
<i>CustomRole</i>	Handles the authorization roles that can be assigned to users
<i>CustomRule</i>	Handles the rules that define the user access control of MOs
<i>Role</i>	Describes the authorization roles that can be assigned to users
<i>Rule</i>	Describes the authorization rules that define the user access control to MOs

3.1 Rules for Default Roles

The detailed permissions for the default roles are described in Table 4, Table 5, Table 6, Table 7, and Table 8.

“Deny” indicates the default behavior when no permission rule is defined.



Table 4 Self-Permissions

MOM Fragment						Permission	Scope	
Managed Element						R	Only the MO but not the attributes (enables navigation in the ECLI)	
	System Functions							
		Backup and Restore Management					Deny	Not applicable
		Fault Management						
		File Management						
		License Management						
		Performance Management						
		Security Management					R	Only the MO but not the attributes (enables navigation in the ECLI)
			User Management					
				LocalAuthenticationMethod				
		AdministratorAccount						
				SshPublicKey				
		UserAccountM						
				UserAccount				
				SshPublicKey				
		Software Inventory Management						Deny
		Software Management						
		System Management						
		Transport						
Equipment								



Table 5 LocalAuthenticationAdministrator Permissions

MOM Fragment				Permission	Scope			
Managed Element				R	Only the MO but not the attributes (enables navigation in the ECLI)			
	System Functions							
		Backup and Restore Management		Deny	Not applicable			
		Fault Management						
		File Management						
		License Management						
		Performance Management						
		Security Management						
			User Management		R	Only the MO but not the attributes (enables navigation in the ECLI)		
				LocalAuthenticationMethod			RWX	The MO, its attributes, actions, and child MOs
				LocalAuthorizationMethod			R	The MO, its attributes, and child MOs
		Software Inventory Management		Deny	Not applicable			
	Software Management							
	System Management							
	Transport							
Equipment								



Table 6 System Administrator Permissions for Default Roles

MOM Fragment			Permission	Scope		
Managed Element			RWX	The MO, its attributes, and actions		
	System Functions			The MO, its attributes, actions, and child MOs		
		Backup and Restore Management			FileGroup=InServicePerformance : R FileGroup=SoftwareManagement : RWX	
		Fault Management				
		License Management				
		Performance Management				
	File Management		Only the MO but not the attributes (enables navigation in the ECLI)			
	Security Management			R	The MO, its attributes, actions, and child MOs	
		Certificate Management	R			
	Software Inventory Management		RW			
	Software Management		RWX			
	System Management					
	Transport		Deny	The MO, its attributes, and actions		
	Equipment			Not applicable		



Table 7 System Security Administrator Permissions for Default Roles

MOM Fragment			Permission	Scope
Managed Element			R	Only the MO but not the attributes (enables navigation in the ECLI)
	System Functions			
		Backup and Restore Management	Deny	Not applicable
		Fault Management	R	The MO, its attributes, actions, and child MOs
		File Management	Deny	Not applicable
		License Management		
		Performance Management		
		Security Management	RWX	The MO, its attributes, actions, and child MOs
		Certificate Management		
		Software Inventory Management	R	Not applicable
		Software Management		
		System Management		
	Transport			
	Equipment			



Table 8 System Read Only for Default Roles

MOM Fragment			Permission	Scope	
Managed Element			R	The MO, its attributes, and actions	
	System Functions			The MO, its attributes, actions, and child MOs	
		Backup and Restore Management			Deny
		Fault Management			
		File Management	R	The MO, its attributes, actions, and child MOs	
		License Management			
		Performance Management			
		Security Management	Deny	Not applicable	
		Software Inventory Management	R	The MO, its attributes, actions, and child MOs	
		Software Management	R		
		System Management			
	Transport		Deny	The MO, its attributes, and actions	
	Equipment			Not applicable	





4 Configuration Management

User Management is accessed using NETCONF or the ECLI to manipulate the MIB.

The following operations, described in Operating Instructions using the ECLI, can be performed by an administrator with the System Security Administrator role:

General

- *Configure Legal Notice*
- *Configure Login Failure Delay*
- *Show Authentication Method Order*

Local Authentication

- *Create User Account*
- *Change User Account*
- *Delete User Account*
- *Create Account Policy*
- *Change Account Policy*
- *Delete Account Policy*
- *Create Password Policy*
- *Change Password Policy*
- *Delete Password Policy*
- *Change Password Quality Configuration*
- *Reset Password for User Account*
- *Set User Roles for User Account*
- *Lock User Account Administratively*
- *Unlock Administrative Lock for User Account*
- *Unlock Operational Lock for User Account*
- *Change Administrator Account*



- *Create Ssh Public Key*
- *Change Ssh Public Key*
- *Delete Ssh Public Key*
- *Remove Password from User Account*

LDAP Authentication

- *View LDAP Configuration*
- *Unlock LDAP Authentication Method*
- *Lock LDAP Authentication Method*
- *Configure LDAP Basic Connection*
- *Configure Ericsson LDAP Filter*
- *Configure Flexible LDAP Filter*
- *Configure LDAP Simple Bind*
- *Configure POSIX LDAP Filter*
- *Configure Target-Based Access Control*
- *Configure TLS for LDAP*

Local Authorization

- *View Roles and Rules*
- *Unlock Local Authorization Method*
- *Lock Local Authorization Method*
- *Create Custom Role*
- *Change Custom Role*
- *Delete Custom Role*
- *Create Custom Rule*
- *Change Custom Rule*
- *Delete Custom Rule*



5 MTAS Roles and Rules

The MTAS roles and rules are listed in Table 9.

Table 9 MTAS Roles and Rules

Managed Area	Mtas_Application_Administrator Role	Mtas_Application_Security_Administrator Role	Mtas_Application_Operator Role	Mtas_Application_Operator_OSSUser Role
Backup and Restore Management	Mtas_AA_BrM 'RWX' permission on the 'ManagedElement, SystemFunctions, BrM,*' resource			
Fault Management	Mtas_AA_Fm 'R' permission on the 'ManagedElement, SystemFunctions, Fm,*' resource	Mtas_ASA_Fm 'R' permission on the 'ManagedElement, SystemFunctions, Fm,*' resource	Mtas_AO_Fm 'R' permission on the 'ManagedElement, SystemFunctions, Fm,*' resource	



Table 9 MTAS Roles and Rules

Managed Area	Mtas_Application_Administrator Role	Mtas_Application_Security_Administrat or Role	Mtas_Application_Operator Role	Mtas_Application_Op erator_OSSUser Role
File Management	Mtas_AA_FileM_FGP 'RWX' permission on the 'ManagedElement, SystemFunctions, FileM,FileGroupPolicy, *' resource	Mtas_ASA_FileM_FGP 'RWX' permission on the 'ManagedElement, SystemFunctions, FileM,FileGroupPolicy, *' resource	Mtas_AO_FileM_FGP 'R' permission on the 'ManagedElement, SystemFunctions, FileM,FileGroupPolicy, *' resource	
	Mtas_AA_FileM_FG_Alarm 'RWX' permission on the 'ManagedElement=1, SystemFunctions=1, FileM=1,LogicalFs=1, FileGroup=AlarmLogs, *' resource	Mtas_ASA_FileM_FG_Alarm 'R' permission on the 'ManagedElement=1, SystemFunctions=1, FileM=1,LogicalFs=1, FileGroup=AlarmLogs, *' resource	Mtas_AO_FileM_FG_Alarm 'R' permission on the 'ManagedElement=1, SystemFunctions=1, FileM=1,LogicalFs=1, FileGroup=AlarmLogs, *' resource	Mtas_AO_FileM_FG_Alarm 'R' permission on the 'ManagedElement=1, SystemFunctions=1, FileM=1, LogicalFs=1, FileGroup=AlarmLogs, *' resource
	Mtas_AA_FileM_FG_Alert 'RWX' permission on the 'ManagedElement=1, SystemFunctions=1, FileM=1,LogicalFs=1, FileGroup=AlertLogs, *' resource	Mtas_ASA_FileM_FG_Alert 'R' permission on the 'ManagedElement=1, SystemFunctions=1, FileM=1,LogicalFs=1, FileGroup=AlertLogs, *' resource	Mtas_AO_FileM_FG_Alert 'R' permission on the 'ManagedElement=1, SystemFunctions=1, FileM=1,LogicalFs=1, FileGroup=AlertLogs, *' resource	Mtas_AO_FileM_FG_Alert 'R' permission on the 'ManagedElement=1, SystemFunctions=1, FileM=1, LogicalFs=1, FileGroup=AlertLogs, *' resource
	Mtas_AA_FileM_FG_DC 'RWX' permission on the 'ManagedElement=1, SystemFunctions=1, FileM=1,LogicalFs=1, FileGroup=DataCollecti on, *' resource		Mtas_AO_FileM_FG_DC 'R' permission on the 'ManagedElement=1, SystemFunctions=1, FileM=1,LogicalFs=1, FileGroup=DataCollecti on, *' resource	
	Mtas_AA_FileM_FG_Mtas 'RWX' permission on the 'ManagedElement=1, SystemFunctions=1, FileM=1,LogicalFs=1, FileGroup=Mtas, *' resource		Mtas_AO_FileM_FG_Mtas 'R' permission on the 'ManagedElement=1, SystemFunctions=1, FileM=1,LogicalFs=1, FileGroup=Mtas, *' resource	
	Mtas_AA_FileM_FG_PM 'RWX' permission on the 'ManagedElement=1, SystemFunctions=1, FileM=1,LogicalFs=1, FileGroup=Performanc eManagementReportFi les, *' resource			
Performance Management	Mtas_AA_PM 'RWX' permission on the 'ManagedElement, SystemFunctions, Pm,*' resource		Mtas_AO_FileM_FG_PM 'R' permission on the 'ManagedElement=1, SystemFunctions=1, FileM=1,LogicalFs=1, FileGroup=Performanc eManagementReportFi les, *' resource	Mtas_AO_FileM_FG_PM 'R' permission on the 'ManagedElement=1, SystemFunctions=1, FileM=1, LogicalFs=1, FileGroup=P erformanceM anagementReportFiles, *' resource



Table 9 MTAS Roles and Rules

Managed Area	Mtas_Application_Administrator Role	Mtas_Application_Security_Administrator Role	Mtas_Application_Operator Role	Mtas_Application_Operator_OSSUser Role
Security Management	Mtas_AA_CertM 'R' permission on the 'ManagedElement, SystemFunctions, SecM, CertM,*' resource	Mtas_ASA_CertM 'R' permission on the 'ManagedElement, SystemFunctions, SecM,CertM,*' resource	Mtas_AO_CertM 'R' permission on the 'ManagedElement, SystemFunctions, SecM,CertM,*' resource	
		Mtas_ASA_MAA_Role_Rule 'RWX' permission on the 'ManagedElement, SystemFunctions, SecM,UserManagement, LocalAuthorizationMethod,Role=Mtas_Application_Administrator,*' resource		
		Mtas_ASA_MASA_Role_Rule 'RWX' permission on the 'ManagedElement, SystemFunctions, SecM,UserManagement, LocalAuthorizationMethod, Role=Mtas_Application_Security_Administrator,*' resource		
		Mtas_ASA_UserManagement 'R' permission on the 'ManagedElement, SystemFunctions, SecM,UserManagement,*' resource		
Software Inventory Management	Mtas_AA_SwIM 'RW' permission on the 'ManagedElement, SystemFunctions, SwInventory,*' resource	Mtas_ASA_SwIM 'R' permission on the 'ManagedElement, SystemFunctions, SwInventory,*' resource	Mtas_AO_SwIM 'R' permission on the 'ManagedElement, SystemFunctions, SwInventory,*' resource	
Software Management	Mtas_AA_SwM 'RWX' permission on the 'ManagedElement, SystemFunctions, SwM,*' resource			
	Mtas_AA_SwM_UP 'RWX' permission on the 'ManagedElement, SystemFunctions, SwM, UpgradePackage.*' resource			



Table 9 MTAS Roles and Rules

Managed Area	Mtas_Application_Administrator Role	Mtas_Application_Security_Administrator Role	Mtas_Application_Operator Role	Mtas_Application_Operator_OSSUser Role
Application	Mtas_AA_MtasFunction 'RWX' permission on the 'ManagedElement, MtasFunction, ** resource	Mtas_ASA_MtasFunction 'R' permission on the 'ManagedElement, MtasFunction, ** resource	Mtas_AO_MtasFunction 'RWX' permission on the 'ManagedElement, MtasFunction, ** resource	
	Mtas_AA_MtasXdmsCai3gUser 'NO_ACCESS' permission on the 'managed element, MtasFunction, MtasXdms, MtasXdmsCai3gUser, ** resource	Mtas_ASA_MtasXdmsCai3gUser 'RWX' permission on the 'ManagedElement, MtasFunction, MtasXdms, MtasXdmsCai3gUser, ** resource	Mtas_AO_MtasXdmsCai3gUser 'NO_ACCESS' permission on the 'ManagedElement, MtasFunction, MtasXdms, MtasXdmsCai3gUser, ** resource	