

Create SSH Public Key

OPERATING INSTRUCTIONS

Copyright

© Ericsson AB 2016. 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
1.1	Prerequisites	1
2	Procedure	3



Create SSH Public Key



1 Introduction

This document describes how to create a Secure Shell (SSH) public key managed object for the local Operation and Maintenance (O&M) user account. The SSH public key is an alternative authentication method for the password authentication. The SSH public key is used to check that the user has the correct private key. O&M users are allowed to create, change, and delete their own SSH key.

1.1 Prerequisites

This section describes the prerequisites, which must be fulfilled before using the procedure.

1.1.1 Conditions

The following conditions must apply:

- The user is familiar with the security policy of the organization.
- The public key is known.
- An Ericsson Command-Line Interface (ECLI) session in Exec mode is in progress.





2 Procedure

To create an SSH public key:

1. Navigate to the `UserAccount` Managed Object (MO), for example:

```
>dn ManagedElement=N0DE06ST, SystemFunctions=1, SecM=1, UserManagement=1, LocalAuthenticationMethod=1, UserAccountM=1, UserAccount=joedoe
```

2. Enter Config mode:

```
(UserAccount=joedoe)>configure
```

3. Create the SSH public key MO, for example:

```
(config-UserAccount=joedoe)>SshPublicKey=1
```

4. Set the attribute `publicKeyContent` to the public SSH key of the user, for example:

```
(config-SshPublicKey=1)>publicKeyContent="ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQCA1ZLKdbq2Ki5n+fMjnx3xLI8YdrgUem
e/HwtU2TK377WoK0VmbF1JwkD+Vw8Kb6yHEyGP8SLYcnHmqbj0qhpq0yxgm76iQ
q3EX1Ueu/5PetBKomVSH3XhxpNg+1WRwg03EQT2+61shy5lm6EHJG2i1+7vc4Q
FOBxrHxC6SW802UjeSIuPFYBZAZ05Wzy2r06X5vLZyokzzcHRqJMzAGxhb+Zg
70WzBFpCj/xCb34Bx6H+DXy1TfYoeV/U8ra5RIBX3Hj0AwcWWA+d8UPUrhEp
kzhJ2b29X4Wk17wJqMiBi69w8wgkyFZLk8GEjMR1hEWVyn5vZ5EH/IRSj6kjN
joedoe@SC-1"
```

Note: The public key content is to be either in OpenSSH key format or entered in Base64 format without line breaks of an RFC 4716, PEM, or PKCS8 formatted key content (use command: `base64 -w 0 content_file`).

Note: The private key must always be protected with a strong password.

5. Commit the setting:

```
(config-SshPublicKey=1)>commit
```

6. Verify the setting:

```
(SshPublicKey=1)>show -v -r
```

The following is an example output:

```
SshPublicKey=1
  publicKeyContent="ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCA1ZLKdbq2Ki5n+fMjnx3xLI8YdrgUeme/HwtU2TK377WoK0VmbF1JwkD+Vw8Kb6yHEyGP8SLYcnHmqbj0qhpq0yxgm76iQq3EX1Ueu/5PetBKomVSH3XhxpNg+1WRwg03EQT2+61shy5lm6EHJG2i1+7vc4QFOBxrHxC6SW802UjeSIuPFYBZAZ05Wzy2r06X5vLZyokzzcHRqJMzAGxhb+Zg70WzBFpCj/xCb34Bx6H+DXy1TfYoeV/U8ra5RIBX3Hj0AwcWWA+d8UPUrhEpkzhJ2b29X4Wk17wJqMiBi69w8wgkyFZLk8GEjMR1hEWVyn5vZ5EH/IRSj6kjNjoedoe@SC-1"
  sshPublicKeyId=1
```



Note: If password authentication is not used, it is recommended to remove the password-based authentication, refer to [Remove Password from User Account](#).