

Show Supported and Enabled TLS Ciphers

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 |



Show Supported and Enabled TLS Ciphers



1 Introduction

This document describes how to show the supported Transport Layer Security (TLS) and enabled cipher suites for the Managed System (MS).

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 has the System Security Administrator or System Administrator role.
- An Ericsson Command-Line Interface (ECLI) session in Exec mode is in progress.



Show Supported and Enabled TLS Ciphers



2 Procedure

To show the supported and enabled TLS cipher suites for the MS:

1. Navigate to *Tls* managed object, for example:

```
>dn ManagedElement=NODE06ST,SystemFunctions=1,SecM=1,Tls=1
```

2. Show the supported ciphers, for example:

```
(Tls=1)>show supportedCiphers
```

The following is an example output:

```
...
supportedCiphers="DES-CBC-MD5"
  authentication="aRSA"
  encryption="DES"
  export=""
  keyExchange="kRSA"
  mac="MD5"
  protocolVersion="SSLv2"
...
```

3. Show the enabled ciphers, for example:

```
(Tls=1)>show enabledCiphers
```

The following is an example output:

```
enabledCiphers="PSK-AES256-CBC-SHA"
  authentication="aPSK"
  encryption="AES"
  export=""
  keyExchange="kPSK"
  mac="SHA1"
  protocolVersion="SSLv3"
enabledCiphers="DES-CBC3-SHA"
  authentication="aRSA"
  encryption="3DES"
  export=""
  keyExchange="kRSA"
  mac="SHA1"
  protocolVersion="SSLv3"
```

Note: The result depends on configuration of *cipherFilters* and can vary depending on systems