ERICSSON

SW signing validation

Description

1 Introduction

This document instructs how to validate SW Packages downloaded from SW Gateway to provide cryptographically verifiable evidence that the software originated from Ericsson and has not been tampered with in transit. To do this both Ericsson SW signing and checksum sha256 should be validated.

Contents

Clarification

< xxxxx > = command where xxxx should be replaced with relevant information

Manifest File = XXX.manifest (File Type: .sha256)

Signature File = XXX.manifest.sha256 (File Type: .sig)

1.1 How to allocate to the instruction document included in the SW Package

This section shows how to allocate to the instructions that are included in the SW Package. There is one instruction document for each file in the SW package. The instruction documents contain the same information except from the commands that are specific for each file. These commands can be used during the validation of each file in the SW package. To ensure that the files in the SW package are secure and origin from Ericsson, each of the files need to be validated separately.

To find the instruction document for each file in the SW Package:

1. Unzip the SW package downloaded from SW Gateway. For an example see Figure 1.

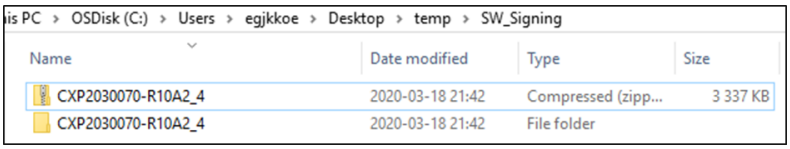

Figure 1. Displays the folder where the SW package has been un-zipped.

2. Open the SW packages and navigate to Signature folder. For an example see Figure 2.

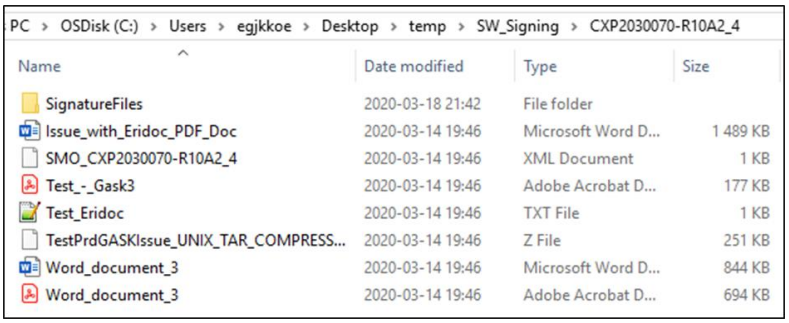

Figure 2. Displays an example of the content in a SW Package.

3. Open the signature folder. In this folder there are two files for each of the file in the SW Package, one document with the instruction and checksum and one signature document. The files that ends with ".manifest" with File Type: .sha256, includes the instructions of how to validate the file. For further understanding see Figure 3.

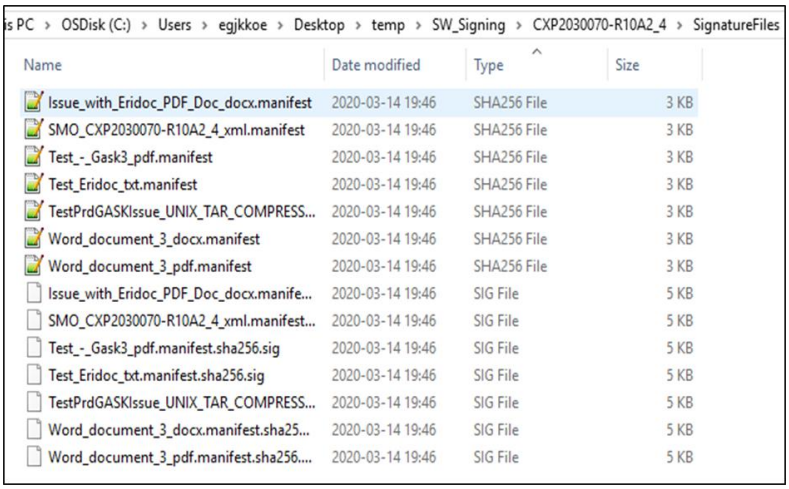

Figure 3. Displays an example of the content in the “SignatureFiles folder”.

Further these documents are referred to as:

Manifest File: XXX.manifest (File Type: .sha256)

Signature File: XXX.manifest.sha256 (File Type: .sig)

2 Instruction for Windows

This section instructs how to validate each file in the software package if you are running on a Windows operative system.

2.1 Prerequisites

To be able to validate the Software packages, CMS (Cryptographic Message Syntax) validation software and the Ericsson Software Gateway Trust Anchor are needed. This section will further explain the prerequisites and give instructions and examples of how to ensure you have all the nessecary software to validate the downloaded Software Package from SW Gateway.

2.1.1 Download the Software Gateway Trust Anchor

All software packages downloaded from Ericsson SW Gateway include a signature that ensures the software originated from Ericsson and has not been tampered with in transit. The Ericsson Software Gateway Trust Anchor is used for validating the software signature. Follow the steps below to download the Ericsson Software Gateway Trust Anchor.

1. Download “software_gateway_integrity_trust_anchor_a1.pem” from:

https://www.ericsson.com/en/about-us/enterprise-security/pki

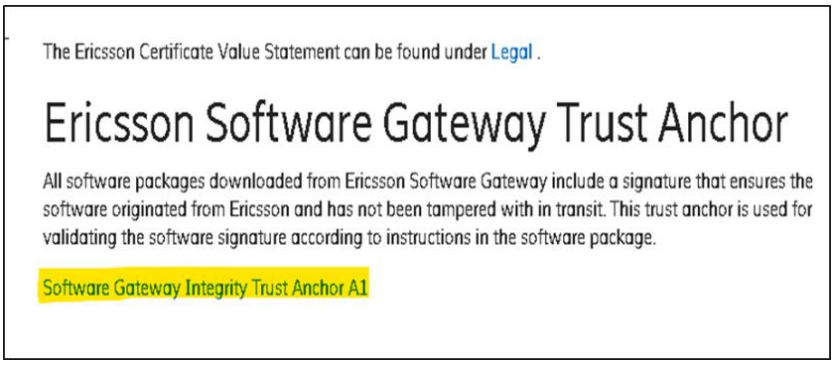

Figure 4. Displays where to download the Ericsson Software Gateway Trust Anchor, click on the highlighted link to download.

2. After downloading the "software_gateway_integrity_trust_anchor_a1.pem" move the pem file to "SignatureFiles" folder. This step is not mandatory but will simplify the verification process. See example in Figure 5.

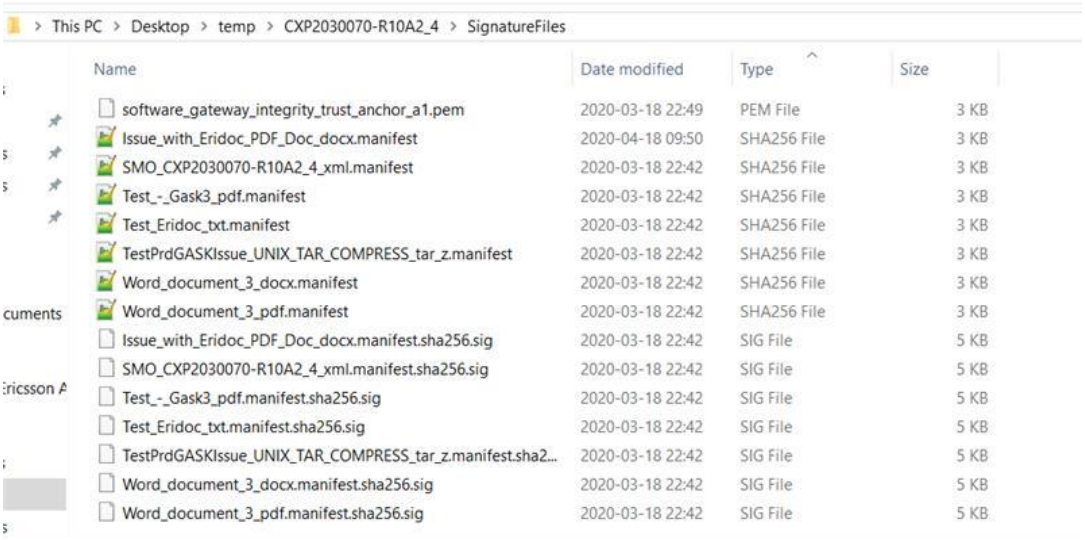

Figure 5. An example of where the Ericsson Software Gateway Trust Anchor could be placed.

2.1.2 Install general software for validating CMS

Install general software for validating CMS (Cryptographic Message Syntax) signatures from a third party of your choice. There are many different third-party software that could be used for this purpose. In this example, GIT will be used since it includes openssl and can be downloaded from Software center if you have and Ericsson computer. There are therefore two options to download GIT, one if you have an Ericsson computer and one if you do not.

Non-Ericsson Computer

If you do not have an Ericsson laptop, GIT could be downloaded from for example:

https://git-scm.com/downloads

Ericsson computer

If you have an Ericsson computer, you can download GIT from the Software Center.

1. Open the Software Center se (Figure 6)

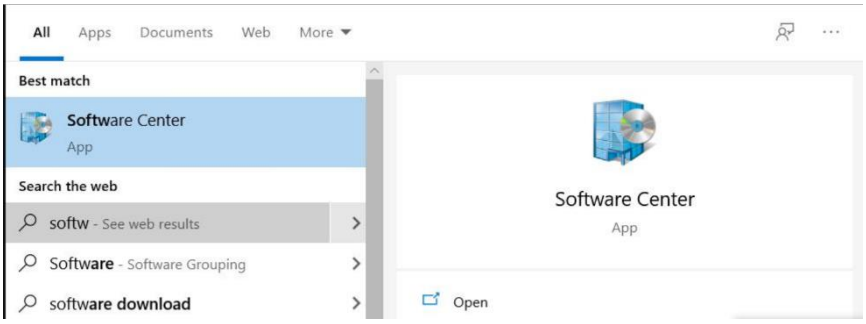

Figure 6. Explains how to open Software center on an Ericsson computer.

2. Search for GIT in Software center and install this on your computer.

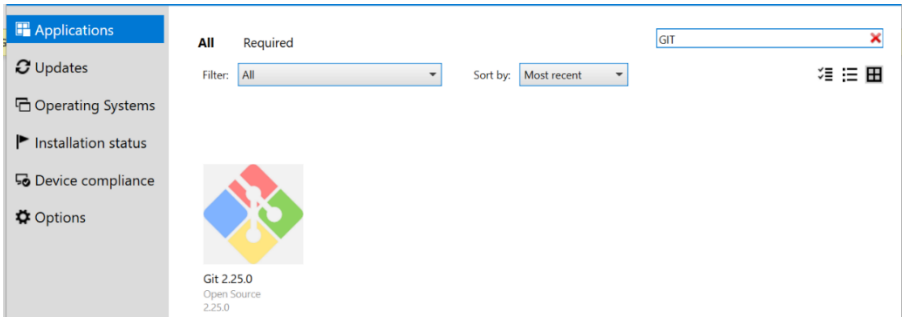

Figure 7. How to allocate to GIT in Software Center.

2.2 Validate the CMS signature

When the Ericsson Software Gateway Trust Anchor and CMS software are installed on the computer, the software signature can be validated.

To validate the signatures, follow the instructions below (These instructions will be for GIT bash, but the same commands can be executed in command prompt):

1. Navigate to the folder where the actual files are residing, by writing in command below:

   cd <pathToSignatureFiles directory>

(complete command and press enter)

2. To validate the chosen file, use command from manifest file or use the command below. (it is of importance to use the manifest file that belongs to the specific file you want to validate, if you want to copy command from manifest file. Because each manifest file contains individual commands). To use command in manifest file copy the highlighted text in manifest file, see example in Figure 8, and remove the # from the command and ensure spaces between each parameter.
   Or use the command below and insert needed data:

   When Trust Anchor are not saved in the signature folder:

   openssl cms -verify -in <Signature file> -binary -inform der -content <Manifest file> -CAfile <PathToPEMFile>/software_gateway_integrity_trust_anchor_a1.pem

   When Trust Anchor are saved in the signature folder:

   openssl cms -verify -in <Signature file> -binary -inform der -content <Manifest file> -CAfile software_gateway_integrity_trust_anchor_a1.pem

   (complete command and press enter)

```
#
#    Example command for OpenSSL:
openssl cms -verify -in Test_Eridoc_txt.manifest.sha256.sig -binary -inform der
-content Test_Eridoc_txt.manifest.sha256 -CAfile
software_gateway_integrity_trust_anchor_a1.pem
#
```

Figure 8. Displays what to copy in the manifest file to validate the signature files.

For an example of the executed commands for validating the software Signature in GIT bash, see Figure 9.

```
egjkkoe@SE-00022696 ~/Desktop/temp/CXP9038909-R4A/SignatureFiles
$ cd /c/Users/egjkkoe/Desktop/temp/CXP9038909-R4A/SignatureFiles

egjkkoe@SE-00022696 ~/Desktop/temp/CXP9038909-R4A/SignatureFiles
$ openssl cms -verify -in Test_Eridoc_txt.manifest.sha256.sig -binary -inform der -content
Test_Eridoc_txt.manifest.sha256 -CAfile software_gateway_integrity_trust_anchor_a1.pem
##############################################################################
```

Figure 9. An example of the command used to verify the signature in the GIT bash.

Upon successful verification the output is: verification successful, see Figure 10.

```
#####################################################################
#
9bdaa65aa52dffa3f0999fa41363f00d2782ddac5d82ed15a898d5adbe8c5540  Test_Eridoc.txt
Verification successful

egjkkoe@SE-00022696 ~/Desktop/temp/CXP9038909-R4A/SignatureFiles
$
```

Figure 10. Illustrates the output when a signature file has been validated successfully.

If you have problem validating the SW Packages please contact Next Level of Maintenance Support.

2.3 Validate the SHA-265 checksum

When the signature has been validated you have ensured the Software Package originates from Ericsson, to ensure the Software Package has not been tampered with, please validate the checksum by following the instructions below:

To validate that the SHA-256 checksum in the signature file match the checksum of the files you received from SW Gateway:

1. Navigate to the folder where the actual files are residing, by writing in the command below:

   cd <pathToDownloadedFiles directory>

   (complete command and press enter)

For an example in GIT bash, see Figure 11.

```
egjkkoe@SE-00022696 ~/Desktop/temp/CXP9038909-R4A
$ cd /c/Users/egjkkoe/Desktop/temp/CXP9038909-R4A

egjkkoe@SE-00022696 ~/Desktop/temp/CXP9038909-R4A
$ ls -l
total 92
-rw-r--r--    1 egjkkoe  Administ      374 May 23 08:31 SMO_CXP9038909-R4A.xml
drwxr-xr-x    1 egjkkoe  Administ     4096 May 24 21:51 SignatureFiles
-rw-r--r--    1 egjkkoe  Administ   180385 May 23 08:31 Test_-_Gask3.pdf
-rw-r--r--    1 egjkkoe  Administ      844 May 23 08:31 Test_Eridoc.txt
```

Figure 11. How to navigate to the folder where the actual files are residing.

2. To validate the chosen file, use command below or from manifest file, for example see Figure 12:

   certUtil -hashfile <downloaded file from the list> SHA256

   (complete command and press enter)

```
# 4) Validate that the SHA-256 hashes below in this file match hashes of
#    the files you received from Software Gateway.
#
#    Example command in Windows (compare hash results to list below):
    certUtil -hashfile <downloaded file from the list> SHA256
```

Figure 12. Shows the command that can be used to calculate the SHA256 checksum.

For an example of how the command should be executed in the GIT bash and the output, see Figure 13.

```
egjkkoe@SE-00022696 ~/Desktop/temp/CXP9038909-R4A
$ certUtil -hashfile Test_Eridoc.txt SHA256
SHA256 hash of Test_Eridoc.txt:
9bdaa65aa52dffa3f0999fa41363f00d2782ddac5d82ed15a898d5adbe8c5540
CertUtil: -hashfile command completed successfully.
```

Figure 13. The commands in GIT prompt for calculating the SHA256 checksum.

The output of checksum should match the checksum from the manifest file (see example from manifest file in Figure 14).

```
#
################################################################################
#
9bdaa65aa52dffa3f0999fa41363f00d2782ddac5d82ed15a898d5adbe8c5540  Test_Eridoc.txt
```

Figure 14. In the bottom of the manifest file the checksum can be found. This picture shows how the SHA256 checksum is displayed in the manifest file.

If you do not want to manually verify that the two checksums match, following command can be executed:

> [[*<CalculatedChecksum>* == *<CheckSumfromManifestFile>*]] && echo " *Successful*" || echo " *Failed*"
>
> (complete command and press enter)

If the checksums match the outcome will be displayed as in Figure 15.

```
$ [[ 2575e1adac98b6d74f09f346570b622b50acd4080b5525641b8988015cb == 2575e1adac98b6d7
4f09f346570b622b50acd4080b5525641b8988015cb ]] && echo "Successful" || echo "Failed"

Successful
```

Figure 15. Displays the output of the command when the checksums match.

If the checksums do not match each other the outcome will be displayed as in Figure 16.

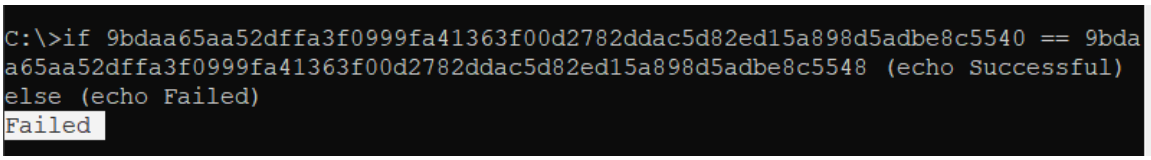

Figure 16. Displays the output of the command when the checksums do not match.

When these steps are successfully executed for each of the files within the Software Package, you have validated that the software originates from Ericsson and have not been tampered with in transit.

3 Instruction for Linux

This section instructs how to validate each file in the software package if you are running on a Linux operative system.

3.1 Prerequisites

To be able to validate the packages, CMS (Cryptographic Message Syntax) validation software and the Ericsson Software Gateway Trust Anchor are needed. This section will further explain what is needed and give instructions and examples of how to ensure you have all the necessary software to validate the downloaded Software Package from SW Gateway. Since Linux already has Openssl available, the only thing needed to download is the Ericsson S Gateway Trust Anchor. To download the Trust Anchor, follow the instructions in this section.

1. Download "software_gateway_integrity_trust_anchor_a1.pem" from:

https://www.ericsson.com/en/about-us/enterprise-security/pki

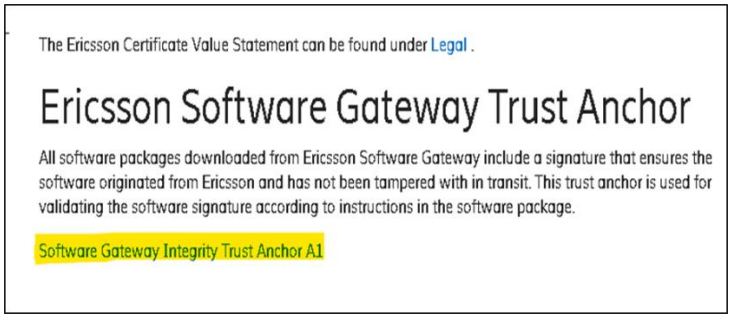

Figure 17. Shows where to download the Ericsson Software Gateway Trust Anchor, click on the highlighted link to download the Ericsson Software Gateway Trust Anchor.

2. After downloading the "software_gateway_integrity_trust_anchor_a1.pem" move the pem file to "SignatureFiles" folder. This step is not mandatory but will simplify the verification process. See an example in Figure 18Figure 17.

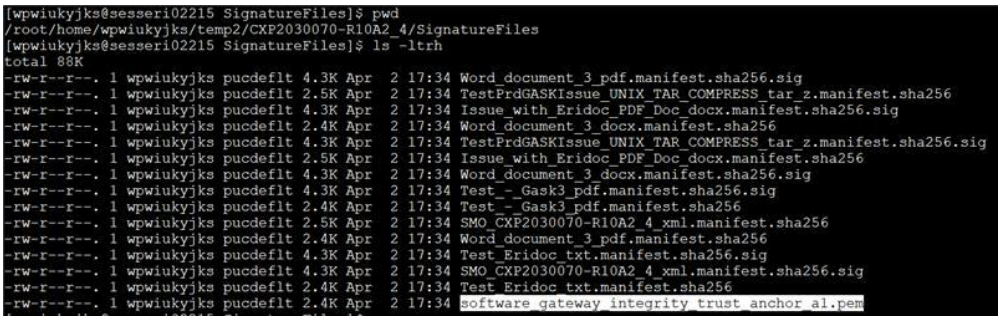

Figure 18. An example of where the Ericsson Software Gateway Trust Anchor could be placed to simplify the validation process.

3.2 Validate the CMS signature

When the Ericsson Software Gateway Trust Anchor is installed on the computer, the software signature can be validated. To validate the signatures, follow the instructions below:

1. Navigate to the folder where the actual files are residing, by writing in the command below:

   cd <pathToSignatureFiles directory>

   (complete command and press enter)

2. To validate the chosen file, use command from manifest file or use the command below. (it is of importance to use the manifest file that belongs to the specific file you want to validate, if you want to copy command from manifest file. Because each manifest file contains individual commands). To use command in manifest file, copy the highlighted text in manifest file, see example in Figure 19, and remove the # from the command and ensure spaces between each parameter.
   Or use the command below and insert needed data:

   When Trust Anchor are not saved in the signature folder:

   openssl cms -verify -in <Signature file> -binary -inform der -content <Manifest file> -CAfile <PathToPEMFile>/software_gateway_integrity_trust_anchor_a1.pem

   When Trust Anchor are saved in the signature folder:

   openssl cms -verify -in <Signature file> -binary -inform der -content <Manifest file> -CAfile software_gateway_integrity_trust_anchor_a1.pem

   (complete command and press enter)

```
# 3) Validate the CMS signature of the manifest file.
#
#    Example command for OpenSSL:
    openssl cms -verify -in Issue_with_Eridoc_PDF_Doc_docx.manifest.sha256.sig -binary -inform der -content
    Issue_with_Eridoc_PDF_Doc_docx.manifest.sha256 -CAfile software_gateway_integrity_trust_anchor_a1.pem
```

Figure 19. Displays what to copy in the manifest file to validate the signature files.

For an example of the executed command, see Figure 20.

```
[wpwiukyjks@sesseri02215 SignatureFiles]$ openssl cms -verify -in Test_Eridoc_txt
.manifest.sha256.sig -binary -inform der -content Test_Eridoc_txt.manifest.sha256
 -CAfile software_gateway_integrity_trust_anchor_a1.pem
##########################################################################
```

Figure 20. An example of the command in Linux.

Upon successful verification, the output is "Verification successful" as in Figure 21.

```
##########################################################################
#
9bdaa65aa52dffa3f0999fa41363f00d2782ddac5d82ed15a898d5adbe8c5540  Test_Eridoc.txt
Verification successful
[wpwiukyjks@sesseri02215 SignatureFiles]$
```

Figure 21. Illustrates the output when a signature file has been validated successfully

If you have problem validating the SW Packages please contact Next Level of Maintenance Support.

3.3 Validate the SHA-256 checksum

When the signature has been validated you have ensured that the software Package originates from Ericsson, to ensure the software Package has not been tampered with, please validate the checksum by follow the instructions below:

To validate that the SHA-256 checksum in the signature file match the checksum of the files you received from SW Gateway:

1. Navigate to the folder where the actual files are residing, by writing in the command below:

   cd <pathToDownloadedPackage>

   (fill in command and press enter)

2. To validate the sha256 checksum insert the needed data in the command below:

   sha256sum -c <pathToSignatureFilesdirectory>/<manifestFile>

   (fill in command and press enter)

For an example of the executed command and the output if the validation is successful, see Figure 22.

```
[wpwiukyjks@sesseri02215 SignatureFiles]$ sha256sum -c /root/home/wpwiukyjks/temp2/
SignatureFiles/SignatureFiles/Test_Eridoc_txt.manifest.sha256
Test_Eridoc.txt: OK
[wpwiukyjks@sesseri02215 SignatureFiles]$
```

Figure 22. Example of commands to calculate the checksum in Linux.

The sha256sum utility automatically validates the hash of any files listed in the manifest file and will warn if those files have been changed or removed. It will not separately warn about additional files in the directory, as their integrity is not tracked. If some of the files in the delivery are not listed in the manifest, their validity will not be verified by this step. Ensure that all expected files show up as "OK" in this check.

When these steps are successfully executed for each of the files within the Software Package, you have validated that the software originates from Ericsson and have not been tampered with.

About Ericsson

Ericsson enables communications service providers to capture the full value of connectivity. The company's portfolio spans Networks, Digital Services, Managed Services, and Emerging Business and is designed to help customers go digital, increase efficiency and find new revenue streams. Ericsson's investments in innovation have delivered the benefits of telephony and mobile broadband to billions of people around the world. The Ericsson stock is listed on Nasdaq Stockholm and on Nasdaq New York. www.ericsson.com