

Configure EPC AAA

IPWorks

OPERATING INSTRUCTIONS

Copyright

© Ericsson AB 2017, 2018. 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
1.2	Relation Information	2
2	EPC AAA Configuration	3
2.1	Configuring Diameter Session Control	3
2.2	Configuring EPC Behavior Control	5
2.3	Configuring EAP AKA/AKA' Authentication	10
2.4	Configuring EPC AAA PKI Authentication	13
2.5	Configuring AAA Front End (PKI)	22
2.6	Configuring Wi-Fi Mobility Management	28
2.7	Configuring Dynamic Info Query Service	34
2.8	Configuring SES Support	34
2.9	Configuring IMEI Check Support	35
2.10	Configuring limit SWx Message Throttling	38
2.11	Configuring Emergency Service Support	39
3	EPC AAA Operations	41
3.1	Restarting EPC AAA Server	41
3.2	Viewing Server Logs	41
3.3	EPC AAA Session Operation	42
4	Appendix	45
4.1	Configuring SS7Stack Parameters	45
4.2	Examples: Listing and Detaching EPC Sessions	46
	Reference List	49





1 Introduction

This document describes how to configure IPWorks EPC AAA.

1.1 Prerequisites

This section states the prerequisites that must be fulfilled.

- Intermediate Linux and UNIX skills
- Concepts, terminologies, and telecommunication abbreviations, such as TCP/IP, packet data networks, and SC/PL node.
- An Ericsson Command-Line Interface (ECLI) session in Exec mode is in progress.

1.1.1 Documents

Before starting this procedure, ensure that the following documents are available:

- For more information about the basics and concepts regarding the configuration management of IPWorks, refer to *IPWorks Configuration Management*.
- For more information about the objects configured through IPWorks CLI (ipwcli), refer to *IPWorks AAA Parameter Description*.
- This document only introduces the most commonly used configuration scenarios, for complete information about the objects configured through ECLI, refer to *Managed Object Model (MOM)*.

1.1.2 Tools

Not applicable.

1.1.3 Conditions

Before starting this procedure, the following conditions must apply:

- IPWorks installation is completed.
- Each functionality, associated with a specific license, must be valid and running normally in the license server.



For more information about IPWorks license related information, refer to *License Management*.

- Storage Server is started.
- EPC AAA must be initially configured.
- Diameter Stack must be configured.

Refer to *Diameter Stack Configuration Guide*.

- SS7 Stack must be configured for the following functions:
 - CS Location Lookup of Wi-Fi Mobility Management
 - SES Support
 - Diameter Over SCTP

For more information about how to configure SS7 Stack, refer to *Configure SS7 for AAA*.

1.2 Relation Information

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

- *Trademark Information*
- *Typographic Conventions*
- *Glossary of Terms and Acronyms*



2 EPC AAA Configuration

This section provides the following topics to guide the configuration personnel how to configure IPWorks EPC AAA:

Note: For most of the configurations in ECLI, you need to restart EPC AAA server to take effect for the change. See Section 3.1 Restarting EPC AAA Server on page 41 for details.

Basic configurations (for each EPC AAA function):

- Section 2.1 Configuring Diameter Session Control on page 3
- Section 2.2 Configuring EPC Behavior Control on page 5

Function configurations:

- Section 2.3 Configuring EAP AKA/AKA' Authentication on page 10
- Section 2.4 Configuring EPC AAA PKI Authentication on page 13
- Section 2.6 Configuring Wi-Fi Mobility Management on page 28
- Section 2.7 Configuring Dynamic Info Query Service on page 34
- Section 2.8 Configuring SES Support on page 34

2.1 Configuring Diameter Session Control

This section provides the procedure to configure the Diameter Session Control.

Prerequisites:

The prerequisites in Section 1.1 on page 1 must be fulfilled first.

2.1.1 Configuring Diameter Session Time-out

1. Set the value of the attribute *diameterSessionTimeout* in the MO *EPCSessionControl*.

```
# ssh <username>@<MIP_OAM_IP> -t -s cli
```

```
>configure
```

```
(config)> dn ManagedElement=<Node Name>,IpworksFunction=
1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAA
Service=1,EPCAAAService=1,EPCSessionControl=1
```



```
(config-EPCSessionControl=1) > diameterSessionTimeout=2880
```

2. Enable the STa session time-out function if needed.

```
(config-EPCSessionControl=1) > enableSTaSessionTimeout=true
```

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

Note: The configuration takes effect immediately.

```
(EPCSessionControl=1) > exit
```

2.1.2 Configuring Session Authorization Life Time Control

Do the following to control the Non-3GPP IP Access Gateway for re-authorization:

1. Modify the value of attribute *enable_AuthGracePeriod_AuthorizationLifetime* in the MO *EPCSessionControl*.

```
# ssh <username>@<MIP_OAM_IP> -t -s cli
```

```
>configure
```

```
(config) > dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAService=1,EPCAAAService=1,EPCSessionControl=1
```

```
(config-EPCSessionControl=1) > enable_AuthGracePeriod_AuthorizationLifetime=true
```

2. Set the value of attribute *authGracePeriod* and *authorizationLifetime* in the MO *EPCSessionControl* if needed.

```
(config-EPCSessionControl=1) > authGracePeriod=200
```

```
(config-EPCSessionControl=1) > authorizationLifetime=1440
```

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

Note: The configuration takes effect immediately.

```
(EPCSessionControl=1) > exit
```

2.1.3 Configuring EPC AAA Session Capacity License Type

Do the following steps to configure which EPC AAA session capacity License type is used, Classic or Layered:

1. Log on to the ECLI and enter the configuration mode.



```
# ssh <username>@<MIP_OAM_IP> -t -s cli
```

```
>configure
```

2. Configure EPC AAA session capacity License type.

- If AAA Server is deployed in classic architecture, AAA Base - Classic Session Capacity License is used to control the capacity of the diameter session. Configure the `sessionCapacityLicenseType` as Classic.

```
(config) > dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAService=1,EPCAAAService=1,EPCSessionControl=1
```

```
(config-EPCSessionControl=1) > sessionCapacityLicenseType=Classic
```

- If AAA Server is deployed in data-layered architecture, AAA Base - Layered Session Capacity License is used to control the capacity of diameter session. Configure the `sessionCapacityLicenseType` as Layered.

```
(config) > dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAService=1,EPCAAAService=1,EPCSessionControl=1
```

```
(config-EPCSessionControl=1) > sessionCapacityLicenseType=Layered
```

3. Commit the change.

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

```
(EPCSessionControl=1) > exit
```

2.2 Configuring EPC Behavior Control

This section provides the procedure to configure the EPC Behavior Control.

Prerequisites:

The prerequisites in Section 1.1 on page 1 must be fulfilled first.

2.2.1 Configuring ASR RAR with Prefix

Enable ASR/RAR command with a prefix digit prepending to AVP User-Name.

Modify the value of attribute `ASR_RARwithPrefix` in the MO `EPCBehaviorControl`.

```
# ssh <username>@<MIP_OAM_IP> -t -s cli
```



```
>configure

(config)> dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAService=1,EPCAAAService=1,EPCBehaviorControl=1

(config-EPCBehaviorControl=1)> ASR_RARwithPrefix=true

(config-EPCBehaviorControl=1)>commit

(EPCBehaviorControl=1)> exit
```

2.2.2 Configuring Authentication Vector Number (SIP-Number-Auth-Items)

This section provides the procedure to configure the value of SIP-Number-Auth-Items in MAR. However, the value falls back to 1 once the Wi-Fi Mobility Management function is enabled.

Modify the value of attribute *authVectorNumber* in the MO *EPCBehaviorControl*.

```
# ssh <username>@<MIP_OAM_IP> -t -s cli

>configure

(config)> dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAService=1,EPCAAAService=1,EPCBehaviorControl=1

(config-EPCBehaviorControl=1)> authVectorNumber=5

(config-EPCBehaviorControl=1)>commit

(EPCBehaviorControl=1)> exit
```

2.2.3 Configuring Get Latest User Profile

This section provides the procedure to delete the local cached user profile and fetch the latest user profile in each SWm/STa authentication.

Modify the value of attribute *getLatestUserProfile* in the MO *EPCBehaviorControl*.

```
# ssh <username>@<MIP_OAM_IP> -t -s cli

>configure

(config)> dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAService=1,EPCAAAService=1,EPCBehaviorControl=1
```



```
(config-EPCBehaviorControl=1) > getLatestUserProfile=true

(config-EPCBehaviorControl=1) > commit

(EPCBehaviorControl=1) > exit
```

2.2.3.1

Configuring AVP Server-Assignment-Type of SAR for authenticated user (Optional)

This section provides the procedure to configure AVP Server-Assignment-Type (SAT) of SAR for authenticated user (the user profile had been cached in local database).

Modify the value of attribute `SARSATypeforAuthenticatedUser` in the MO `EPCBehaviorControl`.

```
# ssh <username>@<MIP_OAM_IP> -t -s cli

> configure

(config) > dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAService=1,EPCAAAService=1,EPCBehaviorControl=1

(config-EPCBehaviorControl=1) > SARSATypeforAuthenticatedUser=<SAT_option>
```

Where:

`<SAT_option>` determines the AVP SAT of SAR. The value can be one of the following:

- `AAA_USER_DATA_REQUEST`

It means that SAR includes AVP SAT with `AAA_USER_DATA_REQUEST`.

In this case, AAA sends SAR to HSS to get user profile.

- `REGISTRATION`

It means that SAR includes AVP SAT with `REGISTRATION`.

In this case, AAA sends SAR to HSS to register and get user profile.

The default value is `AAA_USER_DATA_REQUEST`.

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

(EPCBehaviorControl=1) > exit
```

Note: The configuration of the attribute `SARSATypeforAuthenticatedUser` takes effect only if `getLatestUserProfile=true`.



2.2.4 Configuring S6b RAR Support

Modify the value of attribute *supportRARInS6b* in the MO *EPCBehaviorControl*.

```
# ssh <username>@<MIP_OAM_IP> -t -s cli

>configure

(config)> dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAService=1,EPCAAAService=1,EPCBehaviorControl=1

(config-EPCBehaviorControl=1)> supportRARInS6b=true

(config-EPCBehaviorControl=1)>commit

(EPCBehaviorControl=1)> exit
```

2.2.5 Configuring Enable S6b Authentication without Profile (Optional)

To configure Enable S6b Authentication without Profile, do the following:

Modify the value of attribute *enableS6bAuthzWithoutProfile* in the MO *EPCBehaviorControl*.

```
# ssh <username>@<MIP_OAM_IP> -t -s cli

>configure

(config)> dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAService=1,EPCAAAService=1,EPCBehaviorControl=1

(config-EPCBehaviorControl=1)>enableS6bAuthzWithoutProfile=true

(config-EPCBehaviorControl=1)>commit

(EPCBehaviorControl=1)> exit
```

2.2.6 Configuring Redirect Host Usage (Optional)

This section describes to the client about how the cached route table entry in client, which is created from the Redirect-Host AVP, is to be used. The configuration takes effect when IPWorks receives the redirect information from HSS. Based on the configuration, the AVPs (Redirect-Host-Usage, Redirect-Max-Cache-Time) will be included by IPWorks AAA Server in the reply message to client.



Configure Redirect Host Usage by modifying the value of attribute `redirectHostUsage` and `redirectMaxCacheTime` in the MO `EPCBehaviorControl`.

```
#ssh <username>@<MIP_OAM_IP> -t -s cli

>configure

(config)> dn ManagedElement=<Node Name>,IpworksFunction=1
,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAServ
ice=1,EPCAAAService=1,EPCBehaviorControl=1

(config-EPCBehaviorControl=1)>redirectHostUsage=<redirect
_host_usage>

(config-EPCBehaviorControl=1)>redirectMaxCacheTime=<redir
ect_max_cache_time>

(config-EPCBehaviorControl=1)>commit

(EPCBehaviorControl=1)>exit
```

Where:

- `<redirect_host_usage>` represents the usage of Redirect-Host in client. The default value is `DONT_CACHE`. The value can be `DONT_CACHE`, `ALL_SESSION`, `ALL_REALM`, `EALM_AND_APPLICATION`, `ALL_APPLICATION`, `ALL_HOST` and `ALL_USER`.
- `<redirect_max_cache_time>` represents the maximum number of minutes that the peer and route table entries will be cached. The default value is 65535. The scope of values is 0-71582788 minutes.

2.2.7

Configuring includingHSSHostOption for SWx Requests (Optional)

According to 3GPP standard, AAA server shall include Destination-Host AVP (HSS name) in the SWx requests in case of the AAA server already has HSS name stored.

In HSS-FE deployment scenario, IPWorks AAA provides a configurable switch to set if Destination-Host AVP (HSS name) is included in SWx requests in case of IPWorks AAA already has HSS name stored.

```
# ssh <username>@<MIP_OAM_IP> -t -s cli

> configure

(config)> dn ManagedElement=<Node Name>,IpworksFunction=1,I
PWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAServ
ice=1,EPCAAAService=1,EPCBehaviorControl=1
```



```
(config-EPCBehaviorControl=1) > includingHSSHostOption=<including_HSS_host_option>
```

Where:

<including_HSS_host_option> determines how the HSS host is included in the SWx request. The value can be one of the following:

- **ALL_SWX_REQUEST_WITH_DESTHOST_EXCEPT_MAR:** It means that AAA includes Destination-Host AVP in all subsequent SWx requests.

In this case, the subsequent SWx requests for the same subscriber will be routed to the HSS node which AAA server has registered the subscriber on. If the target HSS server is down, all the subsequent authentications for the same subscriber will be failed.

- **ALL_SWX_REQUEST_WITHOUT_DESTHOST:** It means that no Destination-Host AVP included in all the SWx requests.

In this case, the subsequent SWx requests for the same subscriber will be routed to the node which has highest priority in all of reachable HSS nodes.

- **RESEND_SWX_REQUEST_WITHOUT_DESTHOST_WHEN_HSS_DOWN:** The resend SWx request does not include Destination-Host AVP when HSS host is down.

In this case, the subsequent SWx requests for the same subscriber will be routed to the HSS node which AAA server has registered the subscriber on. If the target HSS server is down, the subsequent SWx requests for the same subscriber fail to send, IPWorks removes Destination-Host AVP of the requests and resend the request to the node which has highest priority in all of reachable HSS nodes.

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

Note: The configuration takes effect immediately.

```
(EPCBehaviorControl=1) > exit
```

2.3 Configuring EAP AKA/AKA' Authentication

This section provides the procedure to configure the EAP AKA/AKA' Authentication.

Prerequisites:

The prerequisites in Section 1.1 on page 1 must be fulfilled first.



2.3.1 Configuring Authentication Using Pseudonym Identity

1. Modify the value of attribute *enablePseudonymAuthentication* in the MO *EapAkaConfig*.

```
# ssh <username>@<MIP_OAM_IP> -t -s cli

>configure

(config)> dn ManagedElement=<Node Name>,IpworksFunction=
1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAA
Service=1,EPCAAAService=1,EapAkaConfig=1

(config-EapAkaConfig=1)> enablePseudonymAuthenticatio
n=true
```

2. Set the value of attribute *pseudonymIdentityExpirationTime* in the MO *EapAkaConfig* if needed.

```
(config-EapAkaConfig=1)>pseudonymIdentityExpirationT
ime=2880

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

Note: The configuration takes effect immediately.

```
(EapAkaConfig=1)> exit
```

2.3.2 Configuring Fast Re-authentication after Full Authentication

1. Modify the value of attribute *maxFastReauthNum* to a non-zero value in the MO *EapAkaConfig*.

```
# ssh <username>@<MIP_OAM_IP> -t -s cli

>configure

(config)> dn ManagedElement=<Node Name>,IpworksFunction=
1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAA
Service=1,EPCAAAService=1,EapAkaConfig=1

(config-EapAkaConfig=1)> maxFastReauthNum=1
```

2. Set the value of attribute *fastIdentityExpirationTime* in the MO *EapAkaConfig* if needed.

```
(config-EapAkaConfig=1)> fastIdentityExpirationTime=300

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

Note: The configuration takes effect immediately.

```
(EapAkaConfig=1)> exit
```



2.3.3 Configuring Authentication Success Indication

Modify the value of attribute *enableAuthSuccessIndication* in the MO *EapAkaConfig*.

```
# ssh <username>@<MIP_OAM_IP> -t -s cli

>configure

(config)> dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAService=1,EPCAAAService=1,EapAkaConfig=1

(config-EapAkaConfig=1)> enableAuthSuccessIndication=true

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

Note: The configuration takes effect immediately.

```
(EapAkaConfig=1)> exit
```

2.3.4 Configuring Mobile Network Code (MNC) Length

Modify the value of attribute *mncLength* in the MO *EapAkaConfig*.

```
# ssh <username>@<MIP_OAM_IP> -t -s cli

>configure

(config)> dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAService=1,EPCAAAService=1,EapAkaConfig=1

(config-EapAkaConfig=1)> mncLength=2

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

Note: The configuration takes effect immediately.

```
(EapAkaConfig=1)> exit
```

2.3.5 Configuring Identity with MAC Address

Modify the value of attribute *identityWithMacAddress* in the MO *EapAkaConfig*

```
# ssh <username>@<MIP_OAM_IP> -t -s cli

>configure
```



```
(config)> dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAService=1,EPCAAAService=1,EapAkaConfig=1
```

```
(config-EapAkaConfig=1)> identityWithMacAddress=true
```

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

Note: The configuration takes effect immediately.

```
(EapAkaConfig=1)> exit
```

2.4 Configuring EPC AAA PKI Authentication

This section provides the procedure to configure the EPC AAA PKI authentication.

Note: Configuration changes take effect after AAA restarts. For information on how to restart EPC AAA, see Section 3.1 Restarting EPC AAA Server on page 41.

Prerequisites:

The prerequisites in Section 1.1 on page 1 must be fulfilled first.

2.4.1 Uploading Certificate Files

Upload all certificate files to the designated directories on SC-1 according to the following table.

Certificate File	Directory	Format
CA Root Certificate ⁽¹⁾	/etc/ipworks/aaa_diameter/pki/ca_cert_path	PEM
Certificate For The Server	/etc/ipworks/aaa_diameter/pki/serv_cert	PEM
Private Key of The Server	/etc/ipworks/aaa_diameter/pki/serv_key	

(1) Multiple certificates can be saved in this directory.

Note: It is mandatory to set the certificate/key files correctly. Incorrect setting might cause AAA server unable to start.



2.4.2 Enabling EPC AAA PKI Authentication

Enable EPC AAA PKI Authentication and configure certificate related attributes in the MO *AAAPKIService*.

```
# ssh <username>@<MIP_OAM_IP> -t -s cli

>configure

(config)>dn ManagedElement=<Node Name>,IpworksFunction=1,IP
WorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAServi
ce=1,AAAPKIService=1

(config-AAAPKIService=1)>enablePki=true

(config-AAAPKIService=1)>pkiUsernamePrefix=<user_name_pref
ix>

(config-AAAPKIService=1)>serverCertificateFileName="<cer
t_file_name>"

(config-AAAPKIService=1)>serverPrivateKeyFileName="<privat
e_key_file_name>"

(config-AAAPKIService=1)>serverPrivateKeyPassword="<pass
word>" cleartext

(config-AAAPKIService=1)>commit

(AAAPKIService=1)>exit
```

Where:

- *<user_name_prefix>* represents the prefix of the user name for PKI authentication.
- *<cert_file_name>* represents the name of the server certificate file. The default value is *server.pem*.
- *<private_key_file_name>* represents the server private key file. The default value is *server.key*.
- *<password>* represents the server private password. When the option *cleartext* is appended, the *<password>* can be plaintext. The default value is whatever.

Note: It is mandatory to set the certificate/key file name and password correctly. Incorrect setting might cause AAA server unable to start.

2.4.3 Configuring Optional Certificate Verification

IPWorks EPC AAA PKI certificate supports optional ways to check the revocation status of user certificate during PKI authentication procedure.



Use the one of the following methods according to actual use scenario:

- Configure the timely information about user certificate revocation status by OCSP (Online Certificate Status Protocol).
- Configure the received user certificate ID in DB.

Note: The CRL (Certificate Revocation List) checking is no more supported in IPWorks EPC AAA.

OCSP

1. Enable the OCSP Check function.

```
# ssh <username>@<MIP_OAM_IP> -t -s cli

>configure

(config)>dn ManagedElement=<Node Name>,IpworksFunction=1
,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAS
ervice=1,AAAPKIService=1,OCSPMgr=1

(config-OCSPMgr=1)>enableOcspCheck=true
```

2. Get CA server name.

```
# /usr/bin/openssl x509 -in /etc/ipworks/aaa_diameter/pk
i/ca_cert_path/<CA Root Certificate File> -noout -subject
-nameopt compat 2>/dev/null
```

For example:

```
# /usr/bin/openssl x509 -in /etc/ipworks/aaa_diameter/p
ki/ca_cert_path/ca.pem -noout -subject -nameopt compat
2>/dev/null
```

```
subject= /C=AU/ST=Some-State/O=Internet Widgits Pty
Ltd/CN=CA
```

Note: CA server name is the content behind "subject= ", in this example, it is:

```
/C=AU/ST=Some-State/O=Internet Widgits Pty
Ltd/CN=CA
```

3. Get CA server responder URL.

Contact network design to get it.

4. Configure OCSP CA server name and responder URL, which are got from Step 2 and Step 3 respectively.

```
(config-OCSPMgr=1)>OCSPServer=1
```



```
(config-OCSPServer=1)>name="/C=AU/ST=Some-State/O=Internet Widgits Pty Ltd/CN=CA"

(config-OCSPServer=1)>responderUrl="http://127.0.0.1:12345"

(config-OCSPServer=1)>commit

(OCSPServer=1)>exit
```

Certificate Information Check in DB

1. Enable the Check Certificate Info function in ECLI.

```
# ssh <username>@<MIP_OAM_IP> -t -s cli

>configure

(config)>dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAService=1,AAAPKIService=1

(config-AAAPKIService=1)>checkCertificateInDB=true

(config-AAAPKIService=1)>commit

(AAAPKIService=1)>exit
```

2. Get CA issuer for client.

```
# /usr/bin/openssl x509 -in <Client Certificate File> -noout -issuer -nameopt sep_comma_plus_space 2>/dev/null
```

For example:

```
# /usr/bin/openssl x509 -in /tmp/client_certificate_1.pem -noout -issuer -nameopt sep_comma_plus_space 2>/dev/null
```

```
issuer= C=AU, ST=Some-State, O=Internet Widgits Pty Ltd, CN=CA
```

3. Get CA serial number for client.

```
# /usr/bin/openssl x509 -in <CA Root Certificate File> -noout -serial 2>/dev/null
```

For example:

```
# /usr/bin/openssl x509 -in /tmp/client_certificate_1.pem -noout -serial 2>/dev/null
```



- ```
serial=01
```
4. Log on to the active SC.
 

```
ssh <Username>@<MIP_OAM_IP>
```

```
Password: <Password>
```
  5. Log on to IPWorks CLI on the Storage Server.
 

```
ipwcli
```

```
IPWorks> Login: <Username>
```

```
IPWorks> Password: <Password>
```
  6. Create an EPC AAA PKI user via IPWorks CLI on the SS, set the issuer and serial number got from Step 2 and Step 3.

```
ipwcli
```

```
IPWorks> create AAANSUser -set name=<username>;IMSI=
<imsi>;MSISDN=<msisdn>;APN="<apnlist>" ;userStatus=<dis
able/enable>;Certificateissuename=<Client Certificate
Issuer>;certificateid=<Client Certificate Serial Number>
```

Example:

```
create AAANSUser -set name=240994004097@nai.epc.m
nc015.mcc234.3gppnetwork.org;IMSI=240994004095;MSI
SDN=13739944240;APN="example-apn1,server.alibaba,m
ail.com.org";userStatus=enable;Certificateissuerna
me="C=AU,ST=Some-State,O=Internet Widgits Pty Ltd,
CN=CA";certificateid="01"
```

## 2.4.4 Configuring PKI User

1. Start the AAA Server Manager.
 

```
ipw-ctr start aaasm <PL hostname>
```

For example:

```
ipw-ctr start aaasm PL-3
```
2. Log on to the active SC.
 

```
ssh <Username>@<MIP_OAM_IP>
```

```
Password: <Password>
```
3. Log on to IPWorks CLI on the Storage Server.



```
ipwcli
```

```
IPWorks> Login: <Username>
```

```
IPWorks> Password: <Password>
```

4. Create an EPC AAA PKI User via IPWorks CLI on the SS.

```
ipwcli
```

```
IPWorks> create AAANSUser -set name=<username>;IMSI=<imsi>;MSISDN=<msisdn>;APN="<apnlist>";userStatus=<disable/enable>;
```

5. Add APNs for EPC AAA PKI User.

**Note:** The following is just an example. How many APNs to be configured is dependent on the actual requirements of the operators. The authentication and authorization request related to the specific APN of the specific user will be rejected, if the specific APN is not listed for the user.

```
ssh <username>@<MIP_OAM_IP> -t -s cli
```

```
>configure
```

```
(config)>dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorksAAARoot=1, IPWorksDiameterAAARoot=1,DiameterAAAService=1,AAAPKIService=1,AAAPKIAPNList=1
```

```
(config-AAAPKIAPNList=1)>APN=example-apn1
```

```
(config-APN=example-apn1)>apnName="example-apn1"
```

```
(config-APN=example-apn1)>contextIdentifier=1
```

```
(config-APN=example-apn1)>up
```

```
(config-AAAPKIAPNList=1)>APN=example-apn2
```

```
(config-APN=example-apn2)>apnName="example-apn2"
```

```
(config-APN=example-apn2)>contextIdentifier=2
```

```
(config-APN=example-apn2)>up
```

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

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

```
AAAPKIAPNList=1
 pkiAPNListId="1" <default>
 APN=example-apn2
```



```

 apnId="example-apn2"
 apnName="example-apn2"
 contextIdentifier=2
 pdnType=0 <default>
 threegppChargingCharact="0A00" <default>
 ambr=[] <empty>
 APN=example-apn1
 apnId="example-apn1"
 apnName="example-apn1"
 contextIdentifier=1 <default>
 pdnType=0 <default>
 threegppChargingCharact="0A00" <default>
 ambr=[] <empty>
 APN=1
 apnId="1" <default>
 apnName="example.apn" <default>
 contextIdentifier=1 <default>
 pdnType=0 <default>
 threegppChargingCharact="0A00" <default>
 ambr=[] <empty>

 (AAPKIAPNList=1)>exit

```

## 2.4.5 Configuring IMSI Mask Handling (Optional)

### 2.4.5.1 Managing Private Key

You can use the tool `aaaimsimaskkeyTool` to manage the private key.

#### #aaaimsimaskkeyTool

##### Usages:

##### NAME

`aaaimsimaskkeyTool` - import, list and delete the record

##### SYNOPSIS

`aaaimsimaskkeyTool import <FILE>`

`aaaimsimaskkeyTool list status`

`aaaimsimaskkeyTool [OPTION] keyid <keyid>`

##### DESCRIPTION

`aaaimsimaskkeyTool` can import the key group file to ipworks NDB. In addition, it can list and delete the record.

`list status` ----- print the count of records in the table `aaaimsi`

##### OPTION

##### Matcher Selection

`list, --list`

Print the record via a key identity.

`delete, --delete`

Delete the record via a key identity



The file name must be the absolute path name, such as "/tmp/mrd47782/private\_key\_group.txt".

The format of the provision file is a text file as following sequence:

```
Labelname
Keyidentityname
Privatekey
Labelname
Keyidentityname
Privatekey
```

For example:

```
labelwith200lengthcharacters0
keyidentifier1
-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEAodcHS7rwt0z4MJM6z5IegdxVZG57v6mxfaVykCpJDg102J6r
cJvI0rzpz9RGbKuiO+ZzUd8prMM+SmlDZMIAvgY2V2keJ5ZECybtGWfOo7mXHcAa
...
-----END RSA PRIVATE KEY-----
labelwith200lengthcharacters1
keyidentifier2
-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEAodcHS7rwt0z4MJM6z5IegdxVZG57v6mxfaVykCpJDg102J6r
cJvI0rzpz9RGbKuiO+ZzUd8prMM+SmlDZMIAvgY2V2keJ5ZECybtGWfOo7mXHcAa
...
-----END RSA PRIVATE KEY-----
```

**Note:** The private key must be 2048 bits PEM (PKCS#1) format.

- Import private key from provision file.

For example:

```
#aaaimsimaskkeyTool import private_key_group.txt
```

There are 2 records are loaded to the table aaaimsimaskkey!

- List private keys.
  - List the count of all the keys.

For example:

```
#aaaimsimaskkeyTool list status
```



There are 2 records in the table `aaaimsimaskkey`.

- List keyid and related information through the specific keyid.

For example:

```
#aaaimsimaskkeyTool list keyid keyidentifier1

[keyidentifier1]
keyidentity: keyidentifier1
label: labelwith200lengthcharacters1
```

- Delete private keys.
  - Delete all private keys and related information.

For example:

```
aaaimsimaskkeyTool delete all

All 100 records are deleted.
```

- Delete the keys information through the specific keyid.

For example:

```
aaaimsimaskkeyTool delete keyid keyidentifier1

The record keyidentifier1 is deleted.
```

#### 2.4.5.2 Configuring Default RSA Key Identity

To configure default RSA key identity, modify the value of attribute `IMSIMaskDefaultKeyID` in the MO `EPCBehaviorControl`:

```
ssh <username>@<MIP_OAM_IP> -t -s cli
```

```
>configure
(config)> dn ManagedElement=<Node Name>,IpworksFunction=1,
IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAService=1,
EPCAAAService=1,EPCBehaviorControl=1
(config-EPCBehaviorControl=1)>IMSIMaskDefaultKeyID ="<key identity>"
(config-EPCBehaviorControl=1)>commit
(EPCBehaviorControl=1)>exit
```

**Note:** The value of `<key identity>` can be fetched from one of the `Keyidentityname` in the provision file, such as `/tmp/mrd47782/private_key_group.txt`.



### 2.4.5.3 Configuring Enable IMSI Mask Handling

To enable IMSI Mask, modify the value of attribute `enableIMSIMask` in the MO `EPCBehaviorControl`:

```
ssh <username>@<MIP_OAM_IP> -t -s cli

>configure
(config)> dn ManagedElement=<Node Name>,IpworksFunction=1,
IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAService=1,
EPCAAAService=1,EPCBehaviorControl=1
(config-EPCBehaviorControl=1)>enableIMSIMask=true
(config-EPCBehaviorControl=1)>commit
(EPCBehaviorControl=1)>exit
```

**Note:** The `IMSIMaskDefaultKeyID` must be configured when `enableIMSIMask=true`.

## 2.5 Configuring AAA Front End (PKI)

In this section, it describes how to configure AAA Front End (PKI) by following topics:

- Section 2.4.1 Uploading Certificate Files on page 13
- Section 2.4.2 Enabling EPC AAA PKI Authentication on page 14
- Section 2.4.3 Configuring Optional Certificate Verification on page 14
- Section 2.4.5 Configuring IMSI Mask Handling (Optional) on page 19
- Section 2.5.1 Enabling AAA Front End (PKI) Feature on page 22
- Section 2.5.2 Configuring CUDB Connection Pool on page 23
- Section 2.5.3 Configuring LDAP Dictionary on page 27
- Section 2.5.4 Configuring AAA FE (PKI) Graceful Handling for CUDB Overload Protection on page 27
- Section 2.5.5 Configuring IPsec Tunnel for CUDB on page 28
- Section 2.5.6 Configuring AAA Front End (PKI) Counter in CUDB on page 28

### 2.5.1 Enabling AAA Front End (PKI) Feature

To enable PKI Front End feature, do the following command:

```
ssh <username>@<MIP_OAM_IP> -t -s cli
```



```
>ManagedElement=<Node name>,IpworksFunction.ipworksRootId
=1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAA
AService=1,AAAPKIService=1
```

```
(config-AAAPKIService=1) >configure
```

```
(config-AAAPKIService=1) >enablePki=true
```

```
(config-AAAPKIService=1) >enablePkiFE=true
```

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

**Note:** The configuration takes effect after EPC AAA server restarts.

## 2.5.2

### Configuring CUDB Connection Pool

This section guides how to configure CUDB connection pool for AAA FE (PKI) feature.

#### Prerequisite:

If route is required for the CUDB connection, follow the examples described in *Configure Route for IPWorks PL Node*.

Table 1 lists the presupposition values that are used as an example for the CUDB connection configuration. CUDB connection configuration varies based on the actual environment. Both IPv4 and IPv6 are supported to work with CUDB.

*Table 1 Example: CUDB Node Parameters Values*

| CUDB Site Name | CUDB Node Name <sup>(1)</sup> | CUDB Node Parameters                                               |
|----------------|-------------------------------|--------------------------------------------------------------------|
| site1          | node1                         | Address = "192.168.20.11"                                          |
|                |                               | distinguishedName = ""                                             |
|                |                               | Password = ""                                                      |
|                |                               | poolSize = 16                                                      |
|                |                               | Port = 389                                                         |
|                | node2                         | Address = "192.168.20.12"                                          |
|                |                               | distinguishedName = "cudbUser=AAAUser,ou=admin,dc=ericsson,dc=com" |
|                |                               | Password = "secret"                                                |
|                |                               | poolSize = 16                                                      |
|                |                               | Port = 389                                                         |



| CUDB Site Name | CUDB Node Name <sup>(1)</sup> | CUDB Node Parameters      |
|----------------|-------------------------------|---------------------------|
| site2          | node1                         | Address = "192.168.20.13" |
|                |                               | distinguishedName = ""    |
|                |                               | Password = ""             |
|                |                               | poolSize = 16             |
|                |                               | Port = 389                |
|                | node2                         | Address = "192.168.20.14" |
|                |                               | distinguishedName = ""    |
|                |                               | Password = ""             |
|                |                               | poolSize = 16             |
|                |                               | Port = 389                |
| site3          | node1                         | Address = "192.168.20.15" |
|                |                               | distinguishedName = ""    |
|                |                               | Password = ""             |
|                |                               | poolSize = 16             |
|                |                               | Port = 389                |
|                | node2                         | Address = "192.168.20.16" |
|                |                               | distinguishedName = ""    |
|                |                               | Password = ""             |
|                |                               | poolSize = 16             |
|                |                               | Port = 389                |

(1) Only CUDB AD Node can be used for connection.

Follow the following example to configure the other CUDB sites and CUDB nodes.



```

>dn ManagedElement=PKI,IpworksFunction=1,
IpworksCommonRoot=1,DataBaseInfo=1,CudbManager=1,
CudbServiceSite=PKI,CudbSiteManager=1
(CudbSiteManager=1)>configure
(config-CudbSiteManager=1)>CudbSite=site1
(config-CudbSite=site1)>CudbNode=node1
(config-CudbNode=node1)>address=192.168.20.11
(config-CudbNode=node1)>poolSize=16
(config-CudbNode=node1)>show -v
CudbNode=node1
 address="192.168.20.11"
 cudbNodeId="node1"
 distinguishedName=[] <empty>
 password=[] <empty>
 poolSize=16
 port=389 <default>

(config-CudbNode=node1)>up
(config-CudbSite=site1)>CudbNode=node2
(config-CudbNode=node2)>address=192.168.20.12
(config-CudbNode=node2)>distinguishedName="cudbUser=AAAUser,ou=admin,dc=com"
(config-CudbNode=node2)>password="secret" cleartext
(config-CudbNode=node2)>poolSize=16
(config-CudbNode=node2)>show -v
CudbNode=node2
 address="192.168.20.12"
 cudbNodeId="node2"
 distinguishedName="cudbUser=AAAUser,ou=admin,dc=ericsson,dc=com"
 password="1:k8d2jPCL2Qa76V1jmjN6+CLUQIbQreeg"
 poolSize=16
 port=389 <default>
(config-CudbNode=node2)>commit

```

**Note:** The configuration takes effect after EPC AAA server restarts.

- *<Node Name>* represents the node name for IPWorks.
- The username and password of the `cudbUser` are created by the CUDB. If they are used, make sure that they have been created in CUDB before you restart EPC AAA server. If not, the related configuration is unnecessary and keeps it empty.

The final results are as below:

```

>dn ManagedElement=<Node Name>,IpworksFunction=
1,IpworksCommonRoot=1,
DataBaseInfo=1,CudbManager=1,CudbServiceSite=PKI,CudbSiteManager=1
(CudbSiteManager=1)>show -v CudbSite=site1
CudbSite=site1
 cudbSiteId="site1"
 CudbNode=node1

```



```

CudbNode=node2
(CudbSiteManager=1)>show -v CudbSite=site2
CudbSite=site2
 cudbSiteId="site2"
 CudbNode=node1
 CudbNode=node2
(CudbSiteManager=1)>show -v CudbSite=site3
CudbSite=site3
 cudbSiteId="site3"
 CudbNode=node1
 CudbNode=node2

```

**Note:** The total pool size is recommended to be smaller than 1,000 according to IPWorks environment and CUDB configuration.

$$\begin{aligned} \text{total\_pool\_size} &= \text{site1\_pool\_size} + \text{site2\_pool\_size} + \\ &\text{site3\_pool\_size} = 200 + 200 + 400 = 800 < 1000 \end{aligned}$$

### CUDB Site Priority

In the scenario of CUDB connection pool with multiple sites, for example, three CUDB sites (site1, site2, and site3), the site priority is as below:

```
site1> site2> site3
```

Which means:

- AAA FE (PKI) connects to the nodes in `site1` by default.
- If `site1` is unreachable, then it connects to `site2`.
- If both `site1` and `site2` are unreachable, then it connects to `site3`.
- If `site1` has recovered, then it switches back to `site1`.

AAA FE (PKI) does not connect to a lower priority site if a higher priority site (like `site1`) is available or recovered.

### CUDB Site Priority Strategy

The CUDB site with a lower string name has a higher priority.

String name `<X>` is lower than string name `<Y>` in the following cases:

- Both string names are compared character by character. The value of the first unmatched character in string name `<X>` is lower than the character in string name `<Y>`. For example, `site1 > site2`.
- All compared characters match but string name `<X>` is shorter.



### 2.5.3 Configuring LDAP Dictionary

```
vi /etc/ipworks/ldapschema/ldap_pki_fe_dictionary.xml
```

```
...
<service name="PKI">
 <cudbRootEntry name="dc=o,dc=com"/>
 <bindDn name="ou=identities,"/>
 <searchDn name="serv=NSD,IMSI=%s,dc=IMSI,ou=identities,"/>
 <entryList>
 <entry name="nsduser">
 <dn name="serv=NSD,mscId=%s,ou=multiSCs,"/>
 <attr name="nsduserpwd" alias="password"/>
 <attr name="IMSI" alias="imsi"/>
 <attr name="MSISDN" alias="msisdn"/>
 <attr name="apnlist" alias="apn"/>
 <attr name="userstatus" alias="userstatus"/>
 <attr name="certificateissuername" alias="certificateissu"
 <attr name="certificateid" alias="certificateid"/>
 </entry>
 </entryList>
...

```

Make sure that the content is aligned with customer LDAP server configuration: cudbRootEntry, searchDn and dn.

### 2.5.4 Configuring AAA FE (PKI) Graceful Handling for CUDB Overload Protection

---

#### Attention!

The configuration of AAA FE (PKI) graceful handling also applies to AAA FE (Radius).

---

To configure AAA FE (PKI) graceful handling for CUDB Overload Protection, execute the following command:

```
ssh <username>@<MIP_OAM_IP> -t -s cli
```

```
>dn ManagedElement=<Node Name>,IpworksFunction=1,IpworksCommonRoot=1,DataBaseInfo=1,CudbManager=1,CudbFunction=1
```

*Table 2 Parameter List of MO CudbFunction*

| Parameter     | Default Value | Description                                                                                   |
|---------------|---------------|-----------------------------------------------------------------------------------------------|
| maxRejectRate | 95            | The Max rate of rejecting or discarding Access-Request in CUDB overload protection situation. |



| Parameter          | Default Value | Description                                                                                                                                                                                                                                                                                                                                       |
|--------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| busyRateThreshold  | 2             | The threshold value in percentage of busy response numbers (from CUDBnode) and ldap requests number (from IPWorks). When the real value exceeds this threshold value, IPWorks starts to discard Access-Request.<br><br>The threshold value in percentage of the rate: number of LDAP_BUSY(received from CUDB) / number of queries (sent to CUDB). |
| rejectRateUpStep   | 5             | The step value in percentage used in recovery procedure from CUDB overload protection. The next rejection rate is the previous rejection rate minus the step value.                                                                                                                                                                               |
| rejectRateDownStep | 10            | The step value in percentage used in continuous CUDB overload protection situation. The next rejection rate is the step value plus the previous rejection rate.                                                                                                                                                                                   |

**Note:**

- Do not change the value of the other parameters. If the default value does not meet the requirements of user, contact the site engineer for support.
- The configuration takes effect after EPC AAA server restarts.

### 2.5.5 Configuring IPSec Tunnel for CUDB

If the operator wants to apply IPSec function, refer to *eVIP Management Guide* for more information.

### 2.5.6 Configuring AAA Front End (PKI) Counter in CUDB

If needed, configure the counter NSDSUSERCNT in CUDB.

For more information about the counter and how to do the configuration, refer to *IPWorks Application Counters in CUDB*.

## 2.6 Configuring Wi-Fi Mobility Management

This section provides the procedure to configure the Wi-Fi Mobility Management.

**Prerequisites:**

The prerequisites in Section 1.1 on page 1 must be fulfilled first.

The Wi-Fi Mobility Management function has five exclusive options which are shown in following:

- WIFIMM\_DISABLE: Disable Wi-Fi Mobility Management function.
- GEOIP\_ONLY: Get user location from Geography IP database. The following list shows the operations:



- Section 2.6.2 Configuring Wi-Fi Mobility Management Common Parameters on page 29
- Section 2.6.3 Configuring Wi-Fi Mobility Management GEOIP Parameters on page 30
- Section 2.6.5 Configuring Geography IP Data on page 31
- Section 2.6.6 Configuring ISOCC to MCC Mapping Dictionary on page 33
- **CS\_LOCATION\_ONLY:** Get user location from 3GPP CS Network. The following list shows the operations:
  - Section 2.6.1 Configure SS7Stack Parameters on page 29
  - Section 2.6.2 Configuring Wi-Fi Mobility Management Common Parameters on page 29
  - Section 2.6.4 Configuring Wi-Fi Mobility Management 3GPP CS Parameters on page 31
  - Section 2.6.7 Configuring E164CC to MCC Mapping Dictionary on page 33

For the configuration information of the following two options, configurations for both **GEOIP\_ONLY** and **CS\_LOCATION\_ONLY** are required.

- **PREFER\_GEOIP:** Get user location from both Geography IP database and 3GPP CS Network.

The result from Geography IP database is preferred.

- **PREFER\_3GPP\_CS\_LOCATION:** Get user location from both Geography IP database and 3GPP CS Network.

The result from 3GPP CS Network is preferred.

## 2.6.1 Configure SS7Stack Parameters

Configure SS7Stack MO related parameters for the function Wi-Fi Mobility Management, see Section 4.1 Configuring SS7Stack Parameters on page 45 for details.

## 2.6.2 Configuring Wi-Fi Mobility Management Common Parameters

1. Modify the value of attribute *option* in the MO *Wi-FiMMService*.

```
ssh <username>@<MIP_OAM_IP> -t -s cli
>configure
```



```
(config)> dn ManagedElement=<Node Name>,IpworksFunction=
1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAA
Service=1,WiFiMMSERVICE=1
```

```
(config-WiFiMMSERVICE=1)> option=GEOIP_ONLY
```

**Note:** option has the following choices:

```
WIFIMM_DISABLE
GEOIP_ONLY
CS_LOCATION_ONLY
PREFER_GEOIP
PREFER_3GPP_CS_LOCATION
```

2. Set the special MNC for Wi-Fi Mobility Management by modifying the value of attribute *specialMNC* in the MO *WiFiMMSERVICE*, which negotiated with HSS, to indicate this function. Its default value is 999, you can modify it according to HSS server configuration.

```
(config-WiFiMMSERVICE=1)> specialMNC=<MNC_VALUE>
```

3. Set the MCC for unknown user location by modifying the value of attribute *mcc4UnknownUserLocation* in the MO *WiFiMMSERVICE*. It indicates that there is no related MCC from Geography IP database or from 3GPP CS network or from both, .

```
(config-WiFiMMSERVICE=1)> mcc4UnknownUserLocation=888
```

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

```
(WiFiMMSERVICE=1)> exit
```

### 2.6.3 Configuring Wi-Fi Mobility Management GEOIP Parameters

If use the function of anonymous proxy checking, configure the related parameters as follows. If do not use the function of anonymous proxy checking, skip this section.

1. Modify the value of attribute *checkAnonymousProxy* in the MO *WiFiMMSERVICE*.

```
ssh <username>@<MIP_OAM_IP> -t -s cli
```

```
> configure
```

```
(config)> dn ManagedElement=<Node Name>,IpworksFunction=
1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAA
Service=1,WiFiMMSERVICE=1
```

```
(config-WiFiMMSERVICE=1)> checkAnonymousProxy=true
```



2. Set the specific MCC for anonymous proxy. When the function is used, it indicates the IP in UE-Local-IP-Address is the anonymous proxy IP. Set the value of attribute *mcc4AnonymousProxy* in the MO *Wi-Fi-MMS-Service*.

```
(config-Wi-Fi-MMS-Service=1) > mcc4AnonymousProxy=777
```

```
(config-Wi-Fi-MMS-Service=1) > commit
```

```
(Wi-Fi-MMS-Service=1) > exit
```

## 2.6.4 Configuring Wi-Fi Mobility Management 3GPP CS Parameters

1. Configure SS7 Stack by using Signaling Manager.

Refer to *Configure SS7 for AAA*.

2. Enter the dn of MO *Wi-Fi-MMS-Service*.

```
ssh <username>@<MIP_OAM_IP> -t -s cli
```

```
> configure
```

```
(config) > dn ManagedElement=<Node Name>, IpworksFunction=
1, IPWorksAAARoot=1, IPWorksDiameterAAARoot=1, DiameterAAA
Service=1, Wi-Fi-MMS-Service=1
```

3. Configure the following parameters with proper values according to MO *Wi-Fi-MMS-Service*.

The following lists the default values:

**Note:** The value range of maxAgeOfLocation is 0-32767. When “32767” is set, AAA considers user location returned from HLR as always valid. Do not set a value greater than “32767”, otherwise AAA will use the value of maxAgeOfLocation as “32767”.

| Parameter        | Default value |
|------------------|---------------|
| maxAgeOfLocation | 120           |
| mcc4HPLMN        | 999           |

4. Commit the change.

```
(config-Wi-Fi-MMS-Service=1) > commit
```

```
(Wi-Fi-MMS-Service=1) > exit
```

## 2.6.5 Configuring Geography IP Data

This section describes how to configure the geography IP data according to the defined CSV format for Wi-Fi Mobility Management.



1. Prepare the geography data according to *IPWorks 3GPP AAA Server Geography Data Description*.
2. Upload the geography data to the active Storage Server.

- a. Upload the geography IP data.

```
scp <ipworks_aaageoipv4_csv_file> <user>@<MIP_OAM_IP>:/tmp/<ipworks_aaageoipv4_csv_file>
```

- b. Upload the anonymous proxy data if the function of checking anonymous proxy in WiFi Mobility Management is enabled.

```
scp <ipworks_aaageoipv4_csv_file> <user>@<MIP_OAM_IP>:/tmp/<ipworks_aaageoipv4_csv_file>
```

3. Log on to the active Storage Server.
4. Load the geography data.

**Note:** For the tool usage, refer to the Section *Tool to Import IPWorks Server Geography Data* in *IPWorks 3GPP AAA Server Geography Data Description*.

- a. Load the geography IP data.

```
/opt/ipworks/common/scripts/geoip/load_geoip_data.py -f /tmp/<ipworks_aaageoipv4_csv_file> -t aaageoipv4
```

```
Step 1: Prepare : [Done]
Step 2: Load New Data From CSV File : [Done]
Step 3: Dump Old Data From MySQL : [Done]
Step 4: Calculate Delta : [Done]
Step 5: Import Delta Data into MySQL : [Done]
[summary]
Add 0 records
Modify 10 records
Delete 168315 records
```

- b. Load the anonymous proxy data if the function of checking anonymous proxy in WiFi Mobility Management is enabled.

```
/opt/ipworks/common/scripts/geoip/load_geoip_data.py -f /tmp/<ipworks_aaaaanonproxyipv4_csv_file> -t aaaaanonproxyipv4
```

```
Step 1: Prepare : [Done]
Step 2: Load New Data From CSV File : [Done]
Step 3: Dump Old Data From MySQL : [Done]
Step 4: Calculate Delta : [Done]
Step 5: Import Delta Data into MySQL : [Done]
[summary]
Add 1048576 records
Modify 0 records
Delete 11 records
```



## 2.6.6 Configuring ISOCC to MCC Mapping Dictionary

The data of the dictionary is from the official online web page. You can get the valid latest information from the web page shown in following:

- ISOCC: ISO 3166-1 alpha-2 codes are two-letter country codes defined in ISO 3166-1. It is a part of the ISO 3166 standard published by the International Organization for Standardization. For details, refer to [Country Codes - ISO 3166](#).
- MCC: ITU-T E.212 Mobile Country Code. For details, refer to [ITU-T E.212 Mobile Country Code](#).

You can configure the mapping dictionary by editing the dictionary file which is stored under `/etc/ipworks/aaa_diameter/` directory.

- `isocc_to_mcc.csv`

For configuring ISOCC to MCC mapping dictionary, the format must be `ISOCC,MCC,Country/Geographical Area`. For example, `AF,412,Afghanistan`.

If one ISOCC corresponds to more than one MCC, only one MCC is selected. For example, `CN,460,China`.

## 2.6.7 Configuring E164CC to MCC Mapping Dictionary

The data of the dictionary is from the official online web page. You can get the valid latest information from the web page shown in following:

- E164CC: ITU-T E.164 Country Code. For details, refer to [ITU-T E.164 Country Code](#).
- MCC: ITU-T E.212 Mobile Country Code. For details, refer to [ITU-T E.212 Mobile Country Code](#).
- Area codes of North America countries. For details, refer to [North American Numbering Plan Administration](#).

You can configure the mapping dictionary by editing the dictionary file which is stored under `/etc/ipworks/aaa_diameter/` directory.

- `e164cc_to_mcc.csv`

For configuring E164CC to MCC mapping dictionary, the format must be `E164CC,MCC,Country/Geographical Area`. For example, `93,412,Afghanistan`.

If one E164CC corresponds to more than one MCC, only one MCC is selected. For example, `86,460,China`.



In North America countries, the same E164CC can be used by the different countries. For this condition, if one country keeps using this country code, the E164CC field must be set as E164CC+Area Codes for the other countries.

For example, both USA and Canada use 1 as their E164CC. For distinguishing them, USA keeps using 1 as its E164CC, and Canada must set the E164CC field as E164CC+Area Codes.

In Canada, 867, 807 and 709 are the area codes, and 1 is the original E164CC. The new E164CC must be set as 1867, 1807 and 1709.

```
1867,302,Canada
1807,302,Canada
1709,302,Canada
1,310,United States of America
```

**Note:** Both Italy and Vatican City State use 39 as E164CC now, but they do not distinguish them. They both use the same data 39,222,Italy from Italy.

## 2.7 Configuring Dynamic Info Query Service

This section provides the procedure to configure the Dynamic Info Query Service.

### Prerequisites:

The prerequisites in Section 1.1 on page 1 must be fulfilled first.

Enable Dynamic Info Query function and configure authorized client.

```
ssh <username>@<MIP_OAM_IP> -t -s cli
>configure
(config)>dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorksAAARoo
IPWorksDiameterAAARoot=1,DiameterAAAService=1,DynamicInfoQueryService
(config-DynamicInfoQueryService=1)>enable=true
(config-DynamicInfoQueryService=1)>authorizedHost="gw.ericsson.se"
(config-DynamicInfoQueryService=1)>commit
(DynamicInfoQueryService=1)>exit
```

## 2.8 Configuring SES Support

This section describes how to configure SES parameters by using ECLI.

### Prerequisites:

The prerequisites in Section 1.1 on page 1 must be fulfilled first.



1. Configure SS7 Stack by using Signaling Manager.

Refer to *Configure SS7 for AAA*.

2. Configure SS7Stack MO related parameters in ECLI for the function *SES Support*.

See Section 4.1 Configuring SS7Stack Parameters on page 45 for details.

3. Enable the SES support in the MO *SES*.

```
ssh <username>@<MIP_OAM_IP> -t -s cli
```

```
>configure
```

```
(config-AppInstanceIdMgr=1) > dn ManagedElement=<Node
Name>,IpworksFunction=1,IPWorksAAARoot=1,IPWorksDiamete
rAAARoot=1,DiameterAAAService=1,SES=1
```

```
(config-SES=1) > enableSES=true
```

4. Configure the criteria of authentication requests from SES.

- a. Check the special authentication request type. The default value is 1 (AUTHENTICATE ONLY),

```
(config-SES=1) > show -v specialAuthRequestType
```

**Note:** SpecialAuthRequestType MUST not be set with a value other than 1.

- b. Set the SES origin host list.

```
(config-SES=1) > sesOriginHost=[<SES-SERVER1-ORIGIN-
HOST>, ..., <SES-SERVERn-ORIGIN-HOST>]
```

5. Commit the change.

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

```
(SES=1) > exit
```

## 2.9 Configuring IMEI Check Support

This section describes how to configure **IMEI Check** feature by ECLI.

### Prerequisites:

Diameter stack should be configured for EIR server, refer to *Diameter Stack Configuration Guide*.

Table 3 shows the default value for the parameter of **IMEI Check** feature:



Table 3 Default Value for IMEI Check

| Parameter name   | Default value   |
|------------------|-----------------|
| AllowAbsentIMEI  | TRUE            |
| AllowGreyList    | TRUE            |
| AllowUnknownIMEI | FALSE           |
| Enable           | FALSE           |
| eirRealm         | eir.ericsson.se |

1. Enable **IMEI Check** feature.

```
>dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorks
AAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAService=1
,EPCAAAService=1,IMEICheck=1
```

```
(IMEICheck=1)>configure
```

```
(config-IMEICheck=1)>Enable=true
```

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

```
(IMEICheck=1)>
```

2. Configure the destination realm for EIR server.

For example: the EIR server realm is eir.ericsson.se.

```
>dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorks
AAARoot=1,IPWorksDiameterAAARoot=1,DiameterStack=1,Eir
Domain=1
```

```
(EirDomain=1)>configure
```

```
(config-EirDomain=1)>eirRealm=eir.ericsson.se
```

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

3. Configure any of the following parameter according to the specific requirement:

a. Configure AllowAbsentIMEI.

With the default value `true`, if ME identity is not carried in DER message from ePDG, AAA will bypass IMEI check and continue the authentication and authorization procedure. If the value is set to `false`, and ME identity is not carried in DER message from ePDG, AAA will reject the authentication request.

If you want to disable AllowAbsentIMEI, do the following:



```
>dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAService=1,EPCAAAService=1,IMEICheck=1
```

```
(IMEICheck=1)>configure
```

```
(config-IMEICheck=1)>AllowAbsentIMEI=false
```

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

```
(IMEICheck=1)>
```

b. Configure AllowGreyList.

With the default value `true`, if EIR returns ECA to AAA and indicates the ME identity is in grey list, AAA will continue the authentication and authorization procedure. If the value is set to `false`, and ME identity is in grey list, AAA will reject the authentication request.

If you want to disable AllowGreyList, do the following:

```
>dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAService=1,EPCAAAService=1,IMEICheck=1
```

```
(IMEICheck=1)>configure
```

```
(config-IMEICheck=1)>AllowGreyList=false
```

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

c. Configure AllowUnknownIMEI.

With the default value `false`, if EIR returns ECA to AAA and indicates the ME identity is unknown, AAA will reject the authentication request. If the value is set to `true`, and ME identity is unknown to EIR, AAA will continue the authentication and authorization procedure.

If you want to enable AllowUnknownIMEI, do the following:

```
>dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,DiameterAAAService=1,EPCAAAService=1,IMEICheck=1
```

```
(IMEICheck=1)>configure
```

```
(config-IMEICheck=1)>AllowUnknownIMEI=true
```

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



## 2.10 Configuring limit SWx Message Throttling

This section describes how to configure limit SWx message throttling by using ECLI.

### 1. Enable function limit SWx message throttling.

- a. Log on to the ECLI and enter the configuration mode.

```
ssh <username>@<MIP_OAM_IP> -t -s cli

>configure
```

- b. Enable EPC SWx throttling function flag, and set rate Limit value.

EpcSWxThrottlingEnabled is the function flag of EPC SWx throttling. The default value is false.

```
>dn ManagedElement= <Node Name>,IpworksFunction=1
,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,Dia
meterAAAService=1,EPCAAAService=1,EPCSWxThrottlin
gControl=1
```

```
(config-EPCSWxThrottlingControl=1)>epcSWxThrottli
ngEnabled=true
```

```
(config-EPCSWxThrottlingControl=1)>rateLimitHigh
Priority=<rateLimitHighThreshold>
```

```
(config-EPCSWxThrottlingControl=1)>rateLimitLowPr
iority=<rateLimitLowThreshold>
```

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

**Note:** The configuration takes effect immediately.

rateLimitLowPriority specifies the low priority threshold for SWx Throttling function.

If the current SWx QPS is more than this value, IPWorks will throttle the MAR message for new user initial session for this PL board.

rateLimitHighPriority specifies the high priority threshold for SWx Throttling function.

If the current SWx QPS is more than this value, IPWorks will throttle all SWx messages for this PL board.

Recommend: rateLimitLowThreshold = rateLimitHighThreshold \* 80%

rateLimitHighThreshold = the maximal QPS allowed by HSS / the number of AAA Diameter PLs



rateLimitHighThreshold value is set as the QPS of one PL board.

rateLimitLowPriority value can be set as different percentage of rateLimitHighThreshold value based on the user requirement.

## 2. Disable throttling to limit SWx message function.

- a. Log on to the ECLI and enter the configuration mode.

```
ssh <username>@<MIP_OAM_IP> -t -s cli

>configure
```

- b. Disable EPC SWx throttling function flag.

```
>dn ManagedElement= <Node Name>,IpworksFunction=1
,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,Dia
meterAAAService=1,EPCAAAService=1,EPCSWxThrottlin
gControl=1

(config-EPCSWxThrottlingControl=1)>epcSWxThrott
lingEnabled=false

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

**Note:** The configuration takes effect immediately.

## 2.11 Configuring Emergency Service Support

This section describes how to configure Emergency Service feature by using ECLI.

- If customers need IMEI check function for Emergency Service, set skipIMEICheckForEC to false.

```
>dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorksAAARoot=1,IPWorksDiameterAAARoot
(IMEICheck=1)>configure
(config-IMEICheck=1)>skipIMEICheckForEC=false
(config-IMEICheck=1)>commit
(IMEICheck=1)>
```

**Note:** The default value of skipIMEICheckForEC is true. It means IPWorks will skip IMEI check for emergency service.

- If customers need roaming check function for Emergency Service, make sure that mcc4HPLMN is set to home MCC, and set skipRoamingCheck4EC to false.



```
>dn ManagedElement==<Node Name>,IpworksFunction=1,IPWorksAAARoot=1,IPWo
rksDiameterAAARoot=1,DiameterAAAService=1,WiFiMMService=1
(WiFiMMService=1)>show -v
WiFiMMService=1
 mcc4HPLMN="<home MCC>" <default>
(WiFiMMService=1)>configure
(config-WiFiMMService=1)>skipRoamingCheck4EC=false
(config-WiFiMMService=1)>commit
(WiFiMMService=1)>
```

**Note:** The default value of `skipRoamingCheck4EC` is `true`, it means that the Emergency Service roaming check function is disabled.



## 3 EPC AAA Operations

This section provides the procedure for common EPC AAA operations.

### 3.1 Restarting EPC AAA Server

IPWorks provides mechanisms for controlling the AAA server once it is installed and operating. You can use the `ipw-ctr` command to start or stop the server directly from the system where the server is in operation.

1. Log on to SC-1 or SC-2.

```
ssh <Username>@<SC-1 or SC-2 IP Address>
```

```
Password:<Password>
```

2. Stop the EPC AAA Server.

```
ipw-ctr stop aaa_diameter <PL hostname>
```

```
Stop aaa_diameter ==> success.
```

3. Start the EPC AAA Server.

```
ipw-ctr start aaa_diameter <PL hostname>
```

```
Start aaa_diameter ==> success.
```

### 3.2 Viewing Server Logs

The EPC AAA server allows the viewing of logs.

The Diameter server log is configured by changing the attribute level of the MO *IPWorksLog*.

1. Log on to the ECLI.

```
ssh <username>@<MIP_OAM_IP> -t -s cli
```

2. Enter the configuration mode.

```
>configure
```

3. Set the value of attribute level in the MO *IPWorksLog*. Choose the right server and process. The following is the example to change the EPC AAA Server log level on PL-3.



```
> dn ManagedElement=<Node Name>,IpworksFunction=1,IPWork
sAAARoot=1,IPWorksAAACCommonRoot=1,AAAServer=PL-3,LogMan
agement=1,IPWorksLog=AAA_DIAMETER_SERVER
```

```
(config-IPWorksLog=AAA_DIAMETER_SERVER)> level=LOG_LEV
EL_DEBUG
```

4. Commit the change and exit ECLI.

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

**Note:** The configuration takes effect immediately.

```
(IPWorksLog=1)>exit
```

5. View the logs in the related log file.

All logs are stored in the directory `/cluster/storage/no-backup/ipworks/logs`. The EPC AAA related log file is `/cluster/storage/no-backup/ipworks/logs/<hostname>/aaa_diameter_server.log`.

**Where:**

The `<hostname>` represents the PL hostname that holds the EPC AAA server. For example, PL-3. The user can see the EPC AAA server log by using the following command:

```
tailf /cluster/storage/no-backup/ipworks/logs/PL-3/aa
a_diameter_server.log
```

## 3.3 EPC AAA Session Operation

This section guides users how to manually list or detach EPC sessions on AAA server.

It includes the following topics:

- Section 3.3.1 Listing EPC Sessions on page 42
- Section 3.3.2 Detaching EPC Sessions on page 43

### Prerequisites

Before listing or detaching the existing diameter sessions according to the specified filter using the `send` command in CLI, ensure both EPC AAA server and AAA Server Manager (SM) are started.

### 3.3.1 Listing EPC Sessions

Use the following `send` command syntax to list the session according to the filter specified by "query" condition:



```
IPWorks> send aaaserver [<aaaservername>] -message="show
stasession" -query="<AVPEXpression_1>[&&<AVPEXpression_2>&&
<AVPEXpression_3>...]"
```

**Where:**

- *<aaaservername>*: the name of the AAA server object created on storage server.
- *<AVPEXpression\_n>*: the format is AVP<relop>'value', where <relop> can be one of the following: =, >=, <=.

**Note:**

- This command lists at most 50 sessions.
- The syntax is also applicable to *s6bsession*, *swmsession*, *swmplussession*, and *staplussession* by replacing *stasession* in the syntax.
- An example of listing EPC sessions is provided in Section 4.2 Examples: Listing and Detaching EPC Sessions on page 46.

### 3.3.2 Detaching EPC Sessions

Before detaching the EPC sessions, it is recommended for users to understand the detachment handling process from the following documents:

- For STaSession and STaPlusSession, refer to *IPWorks 3GPP AAA Server-Non-3GPP Access GW STa Interface*.
- For SWmSession and SWmPlusSession, refer to *IPWorks 3GPP AAA Server-ePDG SWm and SWm+ Interface*.

**Note:** SAR message and SAA message are optional for the SWm Session detachment.

Use the following `send` command syntax to detach the session according to the filter specified by "query" condition:

```
IPWorks> send aaaserver [<aaaservername>] -message="detach
stasession" -query="<AVPEXpression_1>[&&<AVPEXpression_2>&&<A
VPEXpression_3>...]" [-force] [-nowait]
```

**Where:**

- *<aaaservername>*: the name of the AAA server object created on storage server.
- *<AVPEXpression\_n>*: the format is AVP<relop>'value', where <relop> can be one of the following: =, >=, <=.
- `-force`: Using this qualifier forces the session termination.



For STa session termination with this qualifier, the STa session is cleared directly from MySQL NDB without sending the abort session request to IP Access GW.

- `-nowait`: A reply is needed immediately after AAA receives the session detaching commands.

**Note:**

- This command detaches at most 50 sessions. Repeat this command for more than 50 matched sessions.
- The syntax is also applicable to `swmsession`, `swmplussession`, and `staplussession` by replacing `stasession` in the syntax.
- An example of detaching EPC sessions is provided in Section 4.2 Examples: Listing and Detaching EPC Sessions on page 46.



## 4 Appendix

### 4.1 Configuring SS7Stack Parameters

*SS7Stack* configuration is applied to both Wi-Fi Mobility Management and SES Support. However, these two functions have different calling GT address:

- The parameters *isdnNoaOfgsmSCF* and *isdnOfgsmSCF* are used by W-Fi Mobility Management.
- The parameters *isdnNoaOfVLR* and *isdnOfVLR* are used by SES Support.

1. Enter the dn of MO *SS7Stack*.

```
ssh <username>@<MIP_OAM_IP> -t -s cli
```

```
> configure
```

```
(config)> dn ManagedElement=<Node Name>,IpworksFunction=1,IPWorksAAARoot=1,IPWorksDiameterAAARoot=1,SS7Stack=1
```

2. Configure the following *SS7Stack* parameters with proper values.

| Parameter                 | Default Value            | Comment                                                         |
|---------------------------|--------------------------|-----------------------------------------------------------------|
| cpmAddress                | ss7cafcpmaddress:6669    |                                                                 |
| origPointCode             | 100                      |                                                                 |
| destPointCode             | 200                      |                                                                 |
| destSSN                   | 6                        |                                                                 |
| mapInvokeTimer            | 3                        |                                                                 |
| isdnNoaOfgsmSCF           | NOA_INTERNATIONAL_NUMBER | It is the calling GT address used by Wi-Fi Mobility Management. |
| isdnOfgsmSCF              | 123456                   |                                                                 |
| isdnNoaOfVLR              | NOA_INTERNATIONAL_NUMBER | It is the calling GT address used by SES.                       |
| isdnOfVLR                 | 234567                   |                                                                 |
| useGT4CallingPartyAddress | FALSE                    |                                                                 |

The modification of these parameters take effect after EPC AAA restarts.

For details about the parameters, refer to the MO *SS7Stack* in *Managed Object Model (MOM)*.

3. Commit the change.

```
(config-SS7Stack=1)>commit
```



```
(SS7Stack=1) >exit
```

## 4.2 Examples: Listing and Detaching EPC Sessions

When AAA initiates the operation and traffic in a non-single mode configuration, one AAA server is automatically selected to handle the request. As the examples shown in this section, the users can also specify which AAA server to handle the requests.

### 4.2.1 AVP Expressions

| AVP            | relop     | Value Type | Value               |
|----------------|-----------|------------|---------------------|
| UserName       | =         | string     | *                   |
|                |           |            | [User-Name]         |
| IMSI           | =         | string     | *                   |
|                |           |            | [IMSI]              |
| SessionID      | =         | string     | *                   |
|                |           |            | [Session-Id]        |
| OriginHost     | =         | string     | *                   |
|                |           |            | [Origin-Host]       |
| LastUpdateTime | =, >=, <= | unit       | yyyy:MM:dd:HH:mm:ss |
| DiameterState  | =         | enum       | Open                |
|                |           |            | Discon              |

### 4.2.2 Examples: Listing EPC Sessions

```
IPWorks> send aaaserver [aaaserver_name] -message="show
swmsession" -query="UserName=* && DiameterState=open"
```

**Note:** It is recommended to add [aaaserver\_name] when using the send command.

```
IPWorks> send aaaserver server1 -message\
="show swmplussession" -query=" UserName='test_user' "
```

The records are under the control of other AAAServer(active), please send the message to the active server.

```
IPWorks> send aaaserver server2 -message\
"show swmplussession" -query="UserName='test_user' "
```

Received the request, start to handle it now,  
please check the result later.



In the above examples, `server1` is specified to handle the request, however the printout indicates that the server is not active. Then `server2` (assume that it is active) is specified to handle the request, the printout indicates the AAA server is selected correctly.

### 4.2.3 Examples: Detaching EPC Sessions

```
IPWorks> send aaaserver [aaaserver_name] -message="detach
swmplusession" -query="SessionID=* &&UserName='<user_name>'"
```

**Note:** It is recommended to add `[aaaserver_name]` when using the `send` command.

```
IPWorks> send aaaserver server1 -message="detach
swmplusession"
-query="SessionID=* &&UserName='test_user'" -nowait
```

The records are under the control of other AAAServer(active), please send the message to the active server.

```
IPWorks> send aaaserver -message="detach swmplusession" \
-query="SessionID=* &&UserName='test_user'" -nowait
```

Received the request, start to handle it now,  
please check the result later.

In the above examples, `server1` is specified to handle the request, however the printout indicates that the server is not active. Then AAA server name is not specified in the command, the active AAA server is selected automatically.





## Reference List

### Ericsson Documents

- [1] *Trademark Information*
- [2] *Typographic Conventions*
- [3] *Glossary of Terms and Acronyms*
- [4] *License Management*
- [5] *IPWorks Configuration Management*
- [6] *IPWorks AAA Parameter Description*
- [7] *Managed Object Model (MOM)*
- [8] *Command Line Interface User Guide for IPWorks SS*
- [9] *Ericsson Command-Line Interface User Guide*
- [10] *Configure SS7 for AAA*
- [11] *Diameter Stack Configuration Guide*
- [12] *IPWorks 3GPP AAA Server Geography Data Description*
- [13] *IPWorks 3GPP AAA Server-Non-3GPP Access GW STa Interface*
- [14] *IPWorks 3GPP AAA Server-ePDG SWm and SWm+ Interface*

### Online References

- [15] <http://www.ipv6.com/articles/general/ipv6-the-next-generation-internet.htm>
- [16] <http://www.3gpp.org/DynaReport/29329.htm>