Shell - Command Line Interface

Table of Contents

Command Reference 1
Linkset Management Commands 1
- Create Linkset 1
- Remove Linkset 2
- Activate Linkset 2
- Deactivate Linkset 3
- Create Link 3
- Remove Link 4
- Activate Link 4
- Deactivate Link 5
- Show status 5

SCCP Management Commands 6
- Service access points and destinations management 7
- Rule Management 10
- Address Management 13
- Remote Signaling Point Management 16
- Remote Sub-System Management 18
- Long message rules Management 20
- Concerned signaling point codes Management 21
- General parameters 23

M3UA Management Commands 24
- M3UA Management - SCTP 24
- M3UA Management 29

Command Reference

This section describes the Shell commands, their syntax and provide examples of usage. This information is also available at run-time in the Shell and can be referenced by issuing any command using the `--help` option.

Linkset Management Commands

You can manage Linksets using the `linkset`command. It allows you to perform the following operations:

- create linkset
- delete linkset
- activate linkset
- deactivate linkset
- create link
- delete link
- activate link
- deactivate link
- list state of linksets and present links

Create Linkset

Linkset can be created by issuing the `linkset` command with following structure:

```
linkset create <linkset-type> opc <point-code> apc <point-code> ni <network-id>
<linkset-name>
```

or in case of dialogic:

```
linkset create dialogic opc <point-code> apc <point-code> ni <network-id> srcmod <src-
mode> destmod <dest-mode> <linkset-name>
```

or in case of M3UA:

```
linkset create m3ua opc <point-code> apc <point-code> ni <network-id> as <as-name>
<linkset-name>
```

Where:

linkset-type

refers to type of linkset to be created, ie. `dahdi`, `dialogic`or `m3ua`. Correct values depend on which linkset factories have been deployed.

point-code

is simply `MTP`point - either local(`opc`) or remote(`dpc`)

ni

is simply network identifier. It can have following values:

linkset-name

simple string name, which identifies linkset

as-name

Name of AS that M3UALinkset wrapps. Make sure that AS is already created as explained in Create AS

Example 1. Linkset creation

```
mobicents(10.65.208.215:3435)>linkset create dahdi opc 1 apc 2 ni 0 linkset1
LinkSet successfully added
mobicents(10.65.208.215:3435)>linkset create dialogic opc 3 apc 4 ni 3 srcmod 1
destmod 2 linkset2
LinkSet successfully added
```

Remove Linkset

Linkset can be deleted by issuing command with following structure:

```
linkset delete  <linkset-name>
```

Where:

linkset-name

is name set during link creation

Example 2. Linkset Removal

```
mobicents(10.65.208.215:3435)>linkset delete linkset1
LinkSet successfully deleted
```

Activate Linkset

Linkset can be activated by issuing command with following structure:

```
linkset activate  <linkset-name>
```

Where:

linkset-name
: is name set during link creation

Example 3. Linkset Activation

```
mobicents(10.65.208.215)>linkset activate linkset1
LinkSet activated successfully
```

Deactivate Linkset

Linkset can be deactivated by issuing command with following structure:

```
linkset deactivate  <linkset-name>
```

Where:

linkset-name
: is name set during link creation

Example 4. Linkset Deactivateion

```
mobicents(10.65.208.215)>linkset deactivate linkset1
LinkSet deactivated successfully
```

Create Link

`Link`can be created in `Linkset`by issuing command with following structure:

```
linkset link create span <span-num> code <code-num> channel <channel-num> <linkset-
name> <link-name>
```

Where:

span-num
: integer number. It represents port number in card(indexed from 0).

code-num
: link code(sls assigned to this link).

channel-num
: integer number indicating time slot number(TDM time slot).

linkset-name
: is name set during link creation.

link-name
: name which identifies link in linkset.

.

```
mobicents(10.65.208.215:3435)>linkset link create span 1 code 1 channel 1 linkset1
link1
Link successfully added
```

Remove Link

`Link`can be removed from in `Linkset`by issuing command with following structure:

```
linkset link delete <linkset-name> <link-name>
```

Where:

linkset-name
: is name set during link creation

link-name
: name which identifies link in linkset

Example 5. Link Removal

```
mobicents(10.65.208.215:3435)>linkset link delete linkset1 link1
Link successfully deleted
```

Activate Link

Link can be activated by issuing command with following structure:

```
linkset link activate <linkset-name> <link-name>
```

Where:

linkset-name
: is name set during link creation

link-name
: name which identifies link in linkset

Example 6. Link Activation

```
mobicents(10.65.208.215:3435)>linkset link activate linkset1 link1
Link activated successfully
```

Deactivate Link

Link can be deactivated by issuing command with following structure:

```
linkset link deactivate <linkset-name> <link-name>
```

Where:

linkset-name
: is name set during link creation

link-name
: name which identifies link in linkset

Example 7. Link Deactivateion

```
mobicents(10.65.208.215:3435)>linkset link deactivate linkset1 link1
Link deactivated successfully
```

Show status

Linkset and Link's status can be viewed by issuing command with following structure:

```
linkset show
```

Example 8. Linkset Status

```
mobicents(10.65.208.215:3435)>linkset show
linkset1      dahdi    opc=1           apc=2           ni=0    state=UNAVAILABLE
    link1       span=1   channelId=1   code=1  state=UNAVAILABLE
```

The possible state of Linkset are

- UNAVAILABLE : Indicates the linkset does not have any “available” links and cannot transport traffic
- SHUTDOWN : Indicates the linkset has been shutdown in the configuration
- AVAILABLE : Indicates the linkset has at least one available link and can carry traffic

The possible state of Link are

- UNAVAILABLE : Indicates the link is not available to carry traffic. This can occur if the link is remotely or locally inhibited by a user. It can also be unavailable if MTP2 has not been able to successfully activate the link connection.
- SHUTDOWN : Indicates the link has been shutdown in the configuration.
- AVAILABLE : Indicates the link is active and able to transport traffic
- FAILED : A link is FAILED when the link is not shutdown but is unavailable at layer2 for some reason. For example Initial Alignment failed or the link test messages sent by MTP3 are not being acknowledged.

SCCP Management Commands

SCCP provides connectionless and connection-oriented network services. This includes address(GTT) translation and routing, flow control segmentation and reassembly.

A global title is an address (e.g., a dialed 800 number, calling card number, or mobile subscriber identification number) which is translated by SCCP into a destination point code and subsystem number. A subsystem number uniquely identifies an application at the destination signaling point. is used as the transport layer for -based services

The first step for SCCP configuring is a service access points (sap) definition. This step is mandatory. Each SCCP stack can use one or more Mtp3UserPart (Refer [_configuring_sccp] about Mtp3UserPart setting). A sap is a logical definition of the Mtp3UserPart (corresponded local SPC, network indicator (NI) and a set of destinations (remote SPC list)). The index of a sap must correspont to the index of the sap.

The second step is a definition of a list of available remote signaling pointcodes (SPC - rsp) and a list of available remote Sub-Systems (SNN - rss). This step is also mandatory. If routing only by GlobalTytle is used the remote Sub-Systems configuring is not required.

As acts as message router, it requires means to configure routing information. Rules (rule),

primary (primary_add) and backup (backup_add) (if backup addresses are available) addresses should be configured.

If XUDT and LUDT messages are availavle in the SS7 network, user should config a set of long message rules (lmr) that will allow long messages. This step is not mandatory. If no long message rule is configured only UDT messages will be used.

The last step is also optional. A user can configure a set of concerned signaling point codes (csp). Each point code will be announced when local SCCP user becomes (un)available.

User can also configure general SCCP parameters (Refer General parameters about general parameters setting).

Service access points and destinations management

SCCP service access points (sap) and destinations (dest) are managed by `sccp sap` and `sccp dest` commands. They allows to perform following:

- `sccp sap create`
- `sccp sap modify`
- `sccp sap delete`
- `sccp sap show`
- `sccp dest create`
- `sccp dest modify`
- `sccp dest delete`
- `sccp dest show`

Create service access point

Sap can be created by issuing command with following structure:

```
sccp sap create <id> <mtp3-id> <opc> <ni>
```

<id>

A unique number to identify this sap

<mtp3-id>

A Mtp3UserPart index that is used as an index of mtp3UserPart property for SccpStack bean. Refer to [_configuring_sccp] for details. For each Mtp3UserPart a sap must be configured.

<opc>

Specifies the local signaling point code.

<ni>

Specifies the network indicator that forms the part of service information octet (SIO)

For each of sap one or more destinations should be configured. Refer to Create destination for

service access point for details.

Example 9. SCCP sap creation

```
telscale(10.65.208.215:3435)>sccp sap create 1 1 101 2
telscale(10.65.208.215:3435)>sccp sap create 2 2 102 2
```

Modify existing sap

Sap can be modified by issuing command with following structure:

```
sccp sap modify <id> <mtp3-id> <opc> <ni>
```

Meaning of parameters is the same.

Delete SCCP service access point

SCCP sap can be deleted by issuing command with following structure:

```
sccp sap delete <id>
```

Where:

<id>
: is id set during sap creation

Example 10. SCCP sap Removal

```
telscale(10.65.208.215:3435)>sccp sap delete 1
Service access point successfully removed
```

Show SCCP service access point

Saps can be viewed by issuing command with following structure:

```
sccp sap show <id>
```

Where:

<id>
: id is optional. If passed only sap matching the id will be shown, else all the saps will be shown

Create destination for service access point

Destination can be created by issuing command with following structure:

```
sccp dest create <sap-id> <id> <first-dpc> <last-dpc> <first-sls> <last-sls> <sls-
mask>
```

<*sap-id*>

An identifier of the sap for which the destination is being created

<*id*>

A number to identify this destination. The number must be unique for each sap.

<*first-dpc*>

Specifies the first value of the remote signaling point codes range.

<*last-dpc*>

Specifies the last value of the remote signaling point codes range. If destination specifies a single signaling point code, this value must be equal first-dpc

<*first-sls*>

Specifies the first value of the SLS range.

<*last-sls*>

Specifies the last value of the SLS range.

<*sls-mask*>

Specifies the mask value. SLS af a message will be exposed by bitwise AND operation with this mask before comparing with first-sls and last-sls valuers.

SLS value range is from 0 to 255. If the destination cover all possible SLS's use first-sls=0, last-sls=255, sls-mask=255

Example 11. SCCP destination creation

```
telscale(10.65.208.215:3435)>sccp dest create 1 1 201 201 0 7 7
telscale(10.65.208.215:3435)>sccp dest create 2 1 300 399 0 255 255
```

Modify existing destination for service access point

Destination can be modified by issuing command with following structure:

```
sccp dest modify <sap-id> <id> <first-dpc> <last-dpc> <first-sls> <last-sls> <sls-
mask>
```

Meaning of parameters is the same.

Delete SCCP destination for service access point

SCCP destination can be deleted by issuing command with following structure:

```
sccp dest delete <sap-id> <id>
```

Where:

<sap-id>
: An identifier of the sap for which the destination has been created

<id>
: is id set during destination creation

Example 12. SCCP destination Removal

```
telscale(10.65.208.215:3435)>sccp destination delete 1 1
Destination definition successfully deleted
```

Show SCCP destination for service access point

Destinations can be viewed by issuing command with following structure:

```
sccp dest show <sap-id> <id>
```

Where:

<sap-id>
: An identifier of the sap for which the destination has been created

<id>
: id is optional. If passed only destination matching the id will be shown, else all destinations of the saps will be shown

Rule Management

SCCP routing rules are managed by `sccp rule` command. It allows to perform following:

- `sccp rule create`
- `sccp rule modify`
- `sccp rule delete`
- `sccp rule show`

Create Rule

A Rule can be created by issuing command with following structure:

```
sccp rule create <id> <mask> <address-indicator> <point-code> <subsystem-number>
<translation-type> <numbering-plan>
<nature-of-address-indicator> <digits> <ruleType> <primary-address-id> <backup-
address-id> <loadsharing-algorithm>
```

This command should be specified after `primary_add` and `backup_add` are configured. Please refer [_address_management] on how to configure `primary_add` and `backup_add`

<id>

A unique number to identify this rule

<mask>

mask defines which part of the originally dialed digits remains in the translated digits and which part is replaced by the digits from primary or backup address. mask is divided into sections by separator /. The number of sections in mask should be equal to sections in digits passed in this command and sections in primary or backup address

<address-indicator>

The address indicator is the first field in SCCP Party Address(called/calling) and is one octet in length. Its function is to indicate which information elements are present so that the address can be interpreted, in other words, it indicates the type of addressing information that is to be found in the address field. The addressing information from original global title is then compared with passed address information to match the rule.

<point-code>

Point code. This is ignored if bit 0 of address-indicator is not set.

<subsystem-number>

Subsystem Number. This is ignored if bit 1 of address-indicator is not set.

<translation-type>

Translation type. This is ignored if GT Indicator is 0000 or 0001

<numbering-plan>

The Number Plan (NP) field specifies the numbering plan that the address information follows. This is ignored if GT Indicator is 0000, 0001 or 0010

<nature-of-address-indicator>

The Nature of Address Indicator (NAI) field defines the address range for a specific numbering plan. This is only used if GT Indicator is 0100

<digits>

Specifies the string of digits divided into subsections using separator '/' depending on if mask contains separator. The dialed digits should match with theses digits as per rule specified bellow

<ruleType>

Rule type. One of following values is possible.

<primary-address-id>

Identifies the SCCP Address used as the primary translation

<backup-address-id>

Identifies the SCCP Address used as the backup translation in case if pointcode specified by primary address is not available. Backup address is optional and is used only with dominant and loadshared address types

<loadsharing-algorithm>

This parameter is mandatory only if <ruleType> parameter is "loadshared". Loadsharing algorithm is configured here. Possible values of the parameter:

Example 13. SCCP Rule creation

```
telscale(10.65.208.215:3435)>sccp rule create 1 R 71 2 8 0 0 3 123456789 solitary
1
telscale(10.65.208.215:3435)>sccp rule create 2 R 71 2 8 0 0 3 123456789 dominant
1 1
telscale(10.65.208.215:3435)>sccp rule create 2 R 71 2 8 0 0 3 123456789
loadshared 1 1 bit4
```

Modify existing Rule

Rule can be modified by issuing command with following structure:

```
sccp rule modify <id> <mask> <address-indicator> <point-code> <subsystem-number>
<translation-type> <numbering-plan>
<nature-of-address-indicator> <digits> <ruleType> <primary-address-id> <backup-
address-id>
```

Meaning of parameters is the same.

Delete SCCP Rule

SCCP Rule can be deleted by issuing command with following structure:

```
sccp rule delete  <id>
```

Where:

<id>

is id set during rule creation

Example 14. SCCP Rule Removal

```
telscale(10.65.208.215:3435)>sccp rule delete 1
Rule successfully removed
```

Show SCCP Rule

Rule's can be viewed by issuing command with following structure:

```
sccp rule show <id>
```

Where:

<id>
: id is optional. If passed only rule matching the id will be shown, else all the rules will be shown

Address Management

The command is used to define primary or backup address of translation. The global title address information of this command is combined with the global title being translated by examining the mask provided in the `sccp rule create` command.The syntax remains same except for primary address `sccp primary_add` is used and for backup address `sccp backup_add` is used

- `sccp primary_add create`

  `sccp backup_add create`
- `sccp primary_add modify`

  `sccp backup_add modify`
- `sccp primary_add delete`

  `sccp backup_add delete`
- `sccp primary_add show`

  `sccp backup_add show`

Create Address

Address can be created by issuing command with following structure:

- For primary address

```
sccp primary_add create <id> <address-indicator> <point-code> <subsystem-number>
<translation-type> <numbering-plan>
<nature-of-address-indicator> <digits>
```

- For backup address

```
sccp backup_add create <id> <address-indicator> <point-code> <subsystem-number>
<translation-type> <numbering-plan>
<nature-of-address-indicator> <digits>
```

<id>

A unique number to identify this address

<address-indicator>

The address indicator is the first field in SCCP Party Address(called/calling) and is one octet in length. Its function is to indicate which information elements are present so that the address can be interpreted, in other words, it indicates the type of addressing information that is to be found in the address field. The addressing information from original global title is then compared with passed address information to match the rule.

<point-code>

Point code. This is ignored if bit 0 of address-indicator is not set.

<subsystem-number>

Subsystem Number. This is ignored if bit 1 of address-indicator is not set.

<translation-type>

Translation type. This is ignored if GT Indicator is 0000 or 0001

<numbering-plan>

The Number Plan (NP) field specifies the numbering plan that the address information follows. This is ignored if GT Indicator is 0000, 0001 or 0010

<nature-of-address-indicator>

The Nature of Address Indicator (NAI) field defines the address range for a specific numbering plan. This is only used if GT Indicator is 0100

<digits>

The global title address information to translate to, specified as string of digits divided into subsections using separator '/' depending on if mask contains separator.

Example 15. SCCP Primary Address creation

```
telscale(10.65.208.215:3435)>sccp primary_add create 1 71 2 8 0 0 3 123456789
```

Example 16. SCCP Backup Address creation

```
telscale(10.65.208.215:3435)>sccp backup_add create 1 71 3 8 0 0 3 123456789
```

Modify existing Address

Address can be modified by issuing command with following structure:

- For primary address

```
sccp primary_add modify <id> <address-indicator> <point-code> <subsystem-number>
<translation-type> <numbering-plan>
<nature-of-address-indicator> <digits>
```

- For backup address

```
sccp backup_add modify <id> <address-indicator> <point-code> <subsystem-number>
<translation-type> <numbering-plan>
<nature-of-address-indicator> <digits>
```

Meaning of parameters is the same.

Delete Address

- For primary address

```
sccp primary_add delete <id>
```

- For backup address

```
sccp backup_add delete <id>
```

Where:

<id>
: is id set during address creation

Example 17. Primary Address Removal

```
telscale(10.65.208.215:3435)>sccp primary_add delete 1
Rule successfully removed
```

Example 18. Backup Address Removal

```
telscale(10.65.208.215:3435)>sccp backup_add delete 1
Rule successfully removed
```

Show Address

Address's can be viewed by issuing command with following structure:

- For primary address

  ```
  sccp primary_add show <id>
  ```

- For backup address

  ```
  sccp backup_add show <id>
  ```

Where:

<id>
: id is optional. If passed only address matching the id will be shown, else all the addresses will be shown

Remote Signaling Point Management

SCCP resources includes remote signaling point and remote subsystem. Each remote signaling point that SCCP can communicate with must be configured using `sccp rsp` command

- `sccp rsp create`
- `sccp rsp modify`
- `sccp rsp delete`
- `sccp rsp show`

Create Remote Signaling Point

Remote signaling point can be create by issuing command with following structure:

```
sccp rsp create <id> <remote-spc> <rspc-flag> <mask>
```

<id>

A unique number to identify this remote signaling point

<remote-spc>

The remote signaling point

<rspc-flag>

32 bit value. Not used for now. Reserved for future

<mask>

32 bit value. Not used for now. Reserved for future

Example 19. Remote Signalin Point creation

```
telscale(10.65.208.215:3435)>sccp rsp create 1 6477 0 0
```

Modify existing Remote Signaling Point

Remote Signaling Point can be modified by issuing command with following structure:

```
sccp rsp modify <id> <remote-spc> <rspc-flag> <mask>
```

Meaning of parameters is the same.

Delete Remote Signaling Point

```
sccp rsp delete <id>
```

Where:

<id>

is id set during remote signaling point creation

Example 20. Remote Signaling Point removal

```
telscale(10.65.208.215:3435)>sccp rsp delete 1
```

Show Remote Signaling Point/s

Remote signaling point can be viewed by issuing command with following structure:

```
sccp rsp show <id>
```

Where:

<id>

id is optional. If passed only remote signaling point matching the id will be shown, else all the addresses will be shown

Remote Sub-System Management

SCCP resources includes remote signaling point and remote subsytem. Each remote subsystem that SCCP can communicate with must be configured using `sccp rss` command

- `sccp rss create`
- `sccp rss modify`
- `sccp rss delete`
- `sccp rss show`

This command should be specified after remote signaling point is configured. Please refer Remote Signaling Point Management on how to configure remote signaling point

Create Remote Sub-System

Remote subsystem can be created by issuing command with following structure:

```
sccp rss create <id> <remote-spc> <remote-ssn> <rss-flag> <mark-prohibited-when-spc-
resuming>
```

<id>

A unique number to identify this remote subsystem

<remote-spc>

The remote signaling point where this remote susbsytem is deployed

<remote-ssn>

The remote subsystem number

<rss-flag>

32 bit value. Not used for now. Reserved for future

<mark-prohibited-when-spc-resuming>

This parameter is optional, its possible value is: prohibitedWhenSpcResuming. When this

parameter is present configured subsystem is marked as prohibited when its corresponded signaling point code has been resumed.

Example 21. Remote Sub-System creation

```
telscale(10.65.208.215:3435)>sccp rss create 1 6477 8 0 prohibitedWhenSpcResuming
```

Modify existing Remote Sub-System

Remote Sub-System can be modified by issuing command with following structure:

```
sccp rss modify <id> <remote-spc> <remote-ssn> <rss-flag> <mark-prohibited-when-spc-resuming>
```

Meaning of parameters is the same.

Delete Remote Sub-System

```
sccp rss delete <id>
```

Where:

<id>
: is id set during remote subsystem creation

Example 22. Remote Sub-System removal

```
telscale(10.65.208.215:3435)>sccp rss delete 1
```

Show Remote Sub-System/s

Remote subsystem can be viewed by issuing command with following structure:

```
sccp rss show <id>
```

Where:

<id>
: id is optional. If passed only remote subsystem matching the id will be shown, else all will be shown

Long message rules Management

Long message rules describe which message types (UDT/XUDT/LUDT) will be used for outgoing message encoding depends on dpc. If no long message rules is configured only UDT messages will be used.

- `sccp lmr create`
- `sccp lmr modify`
- `sccp lmr delete`
- `sccp lmr show`

Create Long message rule

Long message rule can be created by issuing command with following structure:

```
sccp lmr create <id> <first-spc> <last-spc> <long-message-rule-type>
```

<id>

A unique number to identify this Long message rule

<first-spc>

Specifies the first value of the remote signaling point codes range. (for which Long message rule will apply)

<last-spc>

Specifies the last value of the remote signaling point codes range. If Long message rule specifies a single signaling point code, this value must be equal first-spc.

<long-message-rule-type>

Specifies which message types will be used for the remote signaling point codes range. Possible values are: udt, xudt and ludt.

Example 23. Long message rule creation

```
telscale(10.65.208.215:3435)>sccp lmr create 1 201 201 xudt
telscale(10.65.208.215:3435)>sccp lmr create 2 230 239 udt
```

Modify existing Long message rule

Long message rule can be modified by issuing command with following structure:

```
sccp lmr modify <id> <first-spc> <last-spc> <long-message-rule-type>
```

Meaning of parameters is the same.

Delete Modify existing Long

```
sccp lmr delete <id>
```

Where:

<id>
: is id set during Long message rule creation

Example 24. Long message rule removal

```
telscale(10.65.208.215:3435)>sccp lmr delete 1
```

Show Long message rule/s

Long message rule can be viewed by issuing command with following structure:

```
sccp lmr show <id>
```

Where:

<id>
: id is optional. If passed only Long message rule matching the id will be shown, else all the rules will be shown

Concerned signaling point codes Management

Concerned signaling point codes define a DPC list which will be noticed when local SSN is registered (SSA messages) or unregistered (SSP messages).

- `sccp csp create`
- `sccp csp modify`
- `sccp csp delete`
- `sccp csp show`

Create Concerned signaling point code

Concerned signaling point codes can be created by issuing command with following structure:

```
sccp csp create <id> <spc>
```

<id>
: A unique number to identify this Concerned signaling point code

<spc>

Specifies the value of the remote signaling point code, which will be noticed.

Example 25. Concerned signaling point code creation

```
telscale(10.65.208.215:3435)>sccp csp create 1 201
telscale(10.65.208.215:3435)>sccp csp create 2 202
```

Modify existing Concerned signaling point code

Concerned signaling point code can be modified by issuing command with following structure:

```
sccp csp modify <id> <spc>
```

Meaning of parameters is the same.

Delete Concerned signaling point code

```
sccp csp delete <id>
```

Where:

<id>

is id set during Concerned signaling point code creation

Example 26. Concerned signaling point code removal

```
telscale(10.65.208.215:3435)>sccp csp delete 1
```

Show Concerned signaling point code/s

Concerned signaling point code can be viewed by issuing command with following structure:

```
sccp csp show <id>
```

Where:

<id>

id is optional. If passed only Concerned signaling point code matching the id will be shown, else all the codes will be shown

General parameters

User can set several general parameters that influence the whole SCCP stack.

Table 1. SCCP general parameters

Mnemonic name	Function	Value range	Default value

General parameters setting

General parameter can be set by issuing command with following structure:

```
sccp set <parameter-name> <parameter-value>
```

<parameter-name>

A mnemonic name of a parameter.

<parameter-value>

A value of a parameter.

Example 27. General parameters setting

```
telscale(10.65.208.215:3435)>sccp set zMarginXudtMessage 230
telscale(10.65.208.215:3435)>sccp set removeSpc false
```

General parameters getting

General parameter can be got by issuing command with following structure:

```
sccp get <parameter-name>
```

<parameter-name>

A mnemonic name of a parameter. This parameter is optional. If a mnemonic name is absent all parameter values will be returned.

Example 28. General parameters getting

```
telscale(10.65.208.215:3435)>sccp get zMarginXudtMessage
telscale(10.65.208.215:3435)>sccp get
```

M3UA Management Commands

M3UA stack is also responsible to manage the SCTP Associations.

M3UA Management - SCTP

M3UA - SCTP is managed by `sctp` command. It allows to perform following:

- `sctp server create`
- `sctp server destroy`
- `sctp server start`
- `sctp server stop`
- `sctp server show`
- `sctp association create`
- `sctp association destroy`
- `sctp association show`

Create SCTP Server

SCTP Server can be created by issuing command with following structure:

```
sctp server create <server-name> <host-ip> <host-port> <socket-type>
```

Where:

server-name

Unique name assigned to the server.

host-ip

The host ip address where underlying SCTP server socket will bind. For SCTP multi-home support, you can pass comma separated ip addresses that this server socket will bind to. If the primary ip address becomes unavailable, it will automatically fall back to secondary address. For socket-type TCP, comma separated values will be ignored and only first value (primary address) will be used

host-port

The host port where underlying SCTP server socket will bind

socket-type

This is optional. If not passed default is SCTP else specify as TCP.

Example 29. SCTP Server creation

```
mobicents(127.0.0.1:3436)>sctp server create TestServer 127.0.0.1 2905
Successfully added Server=TestServer
```

Example 30. SCTP Server creation with multi-home

```
mobicents(127.0.0.1:3436)>sctp server create TestServer 10.2.50.145,10.2.50.146
2905
Successfully added Server=TestServer
```

Example 31. TCP Server creation

```
mobicents(127.0.0.1:3436)>sctp server create TestServerTCP 10.2.50.145 2906 TCP
Successfully added Server=TestServerTCP
```

Destroy SCTP Server

SCTP Server can be destroyed by issuing command with following structure:

```
sctp server destroy <server-name>
```

Where:

server-name
: Unique name of the server to be destroyed. Make sure server is stopped before destroying.

Example 32. Destroy SCTP Server

```
mobicents(127.0.0.1:3436)>sctp server destroy TestServer
Successfully removed Server=TestServer
```

Start SCTP Server

SCTP Server can be started by issuing command with following structure:

```
sctp server start <server-name>
```

Where:

server-name

Unique name of the server to be started. The underlying SCTP server socket is bound to ip:port configured at creation time.

Example 33. Start SCTP Server

```
mobicents(127.0.0.1:3436)>sctp server start TestServer
Successfully started Server=TestServer
```

Stop SCTP Server

SCTP Server can be stopped by issuing command with following structure:

```
sctp server stop <server-name>
```

Where:

server-name

Unique name of the server to be stopped. The underlying socket is closed at this point and all resource are released.

Example 34. Stop SCTP Server

```
mobicents(127.0.0.1:3436)>sctp server stop TestServer
Successfully stopped Server=TestServer
```

Show SCTP Server

SCTP Server's configuration can be viewed by issuing command with following structure:

```
sctp server show
```

Example 35. Show SCTP Server

```
mobicents(local)>sctp server show

SERVER TCP name=TestServerTCP started=false hostIp=10.2.50.145 hostPort=2906
Associations:

SERVER SCTP name=TestServer started=false hostIp=10.2.50.145 hostPort=2905
secondaryHost=10.2.50.146
Associations:
```

Create SCTP Association

Association can be created by issuing command with following structure:

```
sctp association create <assoc-name> <CLIENT | SERVER> <server-name> <peer-ip> <peer-
port> <host-ip> <host-port> <socket-type>
```

Where:

assoc-name

Unique name of the association

CLIENT | SERVER

If this association is client side or server side. If its client side, it will initiate the connection to peer and bind's to host-ip:host-port trying to connect to peer-ip:peer-port. For SCTP multi-home support, you can pass comma separated ip addresses that this association will bind to. If the primary ip address becomes unavailable, it will automatically fall back to secondary address. For socket-type TCP, comma separated values will be ignored and only first value (primary address) will be used

server-name

If this association is server side, server-name must be passed to associate with server. Server with server-name should have already been created by using command Create SCTP Server

socket-type

This is optional. If not passed default is SCTP else specify as TCP. If association is of SERVER type, the socket-type should match with one specified while creating server.

Example 36. Create CLIENT SCTP Association

```
mobicents(local)>sctp association create Assoc1 CLIENT 192.168.56.101 2905
192.168.56.1,192.168.56.1 2905
Successfully added client Associtaion=Assoc1
```

Example 37. Create SERVER SCTP Association

```
mobicents(192.168.56.1:3436)>sctp association create Assoc2 SERVER TestServer
192.168.56.1 2905
Successfully added server Association=TestServer
```

Destroy SCTP Association

Association can be destroyed by issuing command with following structure:

```
sctp association destroy <assoc-name>
```

Where:

assoc-name

Unique name of the association to be destroyed

Example 38. Destroy SCTP Association

```
mobicents(192.168.56.1:3436)>sctp association destroy Assoc1
Successfully removed association=Assoc1
```

Show SCTP Association

Configuration of Association can be viewed by issuing command with following structure:

```
sctp association show
```

Example 39. Show SCTP Association

```
(local)>sctp association show

ASSOCIATION SCTP name=Assoc1 started=false peerIp=192.168.56.101 peerPort=2905
hostIp=192.168.56.1 hostPort2905 type=CLIENT secondaryHost=192.168.56.1

ASSOCIATION SCTP name=Assoc2 started=false peerIp=192.168.56.1 peerPort=2905
server=TestServer type=SERVER
```

M3UA Management

M3UA is managed by `m3ua` command. It allows to perform following:

- `m3ua as create`
- `m3ua as destroy`
- `m3ua as show`
- `m3ua asp create`
- `m3ua asp destroy`
- `m3ua asp show`
- `m3ua asp start`
- `m3ua asp stop`
- `m3ua as add`
- `m3ua as remove`
- `m3ua route add`
- `m3ua route remove`
- `m3ua route show`

Create AS

Application Server (AS) can be created by issuing command with following structure:

```
m3ua as create <as-name> <AS | SGW | IPSP> mode <SE | DE> ipspType <client | server>
rc <routing-context> traffic-mode <traffic mode> min-asp <minimum asp active for
TrafficModeType.Loadshare> network-appearance <network appearance>
```

Where:

as-name
: simple string name, which identifies AS. Make sure this is unique. This is mandatory parameter

AS | *SGW* | *IPSP*
: Specify if this is of type AS or SGW or IPSP. This is mandatory parameter

SE | *DE*
: Specify if the single or double exchange of ASP State Maintenance (ASPSM) and ASP Traffic Maintenance (ASPTM) messages should be performed. This is mandatory parameter.

client | *server*
: If As if of type IPSP, speicfy here if its client or server type.

routing-context
: refers to Routing Context already configured on M3UA stack on SGW side. This is optional parameter. If no Routing Context is passed, Application Server Process (assigned to this AS) may not be configured to process signaling traffic related to more than one Application Server, over a single SCTP Association

traffic-mode

Traffic mode for ASP's. By default its loadshare. Mobicents M3UA only supports loadshare and override, broadcast is not supported.

min-asp

The minumum number of active ASPs needed (if the traffic mode is 'loadshare') before the payload starts flowing (AS goes into ACTIVE state). This is an optional parameter and if not specified the default value is 1. Also if the traffic-mode is not set as 'loadshare', this parameter has no effect.

network-appearance

The Network Appearance is a M3UA local reference shared by SG and AS (typically an integer) that, together with an Signaling Point Code, uniquely identifies an SS7 node by indicating the specific SS7 network to which it belongs. It can be used to distinguish between signaling traffic associated with different networks being sent between the SG and the ASP over a common SCTP association. This is optional.

Example 40. AS (IPSP) creation

```
mobicents(127.0.0.1:3435)>m3ua as create AS1 IPSP mode DE ipspType server rc 1
traffic-mode loadshare
Successfully created AS name=AS1
```

Example 41. AS creation

```
mobicents(127.0.0.1:3435)>m3ua as create AS2 AS mode SE rc 100 traffic-mode
loadshare network-appearance 12
Successfully created AS name=AS2
```

Destroy AS

Application Server (AS) can be destroyed by issuing command with following structure:

```
m3ua as destroy <as-name>
```

Where:

as-name

Simple string name, which identifies AS. Make sure AS is in state INACTIVE and all the ASP's are unassigned before destroying

Example 42. Destroy AS

```
mobicents(127.0.0.1:3435)>m3ua as destroy AS1
Successfully destroyed AS name=AS1
```

Show AS

Application Server configured can viewed by issuing command with following structure:

```
m3ua as show
```

Example 43. Show AS

```
mobicents(local)>m3ua as show

AS name=AS2 functionality=AS mode=SE rc=[100] trMode=2 defaultTrMode=2 na=12
peerFSMState=DOWN
Assigned to :


AS name=AS1 functionality=IPSP mode=DE ipspType=SERVER rc=[1] trMode=2
defaultTrMode=2 localFSMState=DOWN peerFSMState=DOWN
Assigned to :
```

Create ASP

Application Server Process (ASP) can be created by issuing command with following structure:

```
m3ua asp create <asp-name> <sctp-association> aspid <aspid>
```

Where:

asp-name
: Name of this ASP. It should be unique

sctp-association
: name of SCTP Association

aspid
: Identifier for this newly created Application Server Process. If this is not passed, next available aspid will be used.

Example 44. ASP creation

```
mobicents(127.0.0.1:3435)>m3ua asp create ASP1 Assoc1
Successfully created AS name=ASP1
```

Destroy ASP

ASP can be destroyed by issuing command with following structure:

```
m3ua asp destroy <asp-name>
```

Where:

asp-name
: Name of this ASP to be destroyed. Make sure ASP is stopped before destroying

Example 45. Destroy ASP

```
mobicents(127.0.0.1:3435)>m3ua asp destroy ASP1
Successfully destroyed ASP name=ASP1
```

Show ASP

ASP configured can be viewed by issuing command with following structure:

```
m3ua asp show
```

Example 46. Show ASP

```
mobicents(local)>m3ua asp show

ASP name=ASP1 sctpAssoc=Assoc1 started=false
Assigned to :
```

Start ASP

Application Server Process (ASP) can be started with following structure

```
m3ua asp start <asp-name>
```

Where:

asp name
: name of ASP created earlier. Make sure ASP you are trying to start is assigned to at least one AS

Example 47. Start ASP

```
mobicents(127.0.0.1:3435)>m3ua asp start ASP1
Successfully started ASP name=ASP1
```

Stop ASP

Application Server Process (ASP) can be stopped with following structure

```
m3ua asp stop <asp-name>
```

Where:

asp name
: name of ASP started earlier.

Example 48. Stop ASP

```
mobicents(127.0.0.1:3435)>m3ua asp stop ASP1
Successfully stopped ASP name=ASP1
```

Add ASP to AS

Application Server Process (ASP) can be assigned to Application Server (AS) with following structure

```
m3ua as add <as-name> <asp-name>
```

Where:

as name
: name of AS created earlier

asp name
: name of ASP created earlier

Note

Mobicents M3UA supports configuring ASP to process signaling traffic related to more than one Application Server, over a single SCTP Association. However you need to make sure that all the AS's that ASP is shared with has Routing Context (unique) configured.

Example 49. Add ASP to AS

```
mobicents(127.0.0.1:3435)>m3ua as add AS1 ASP1
Successfully added ASP name=ASP1 to AS name=AS1
```

Remove ASP from AS

Application Server Process (ASP) can be unassigned from Application Server (AS) with following structure

```
m3ua as remove <as-name> <asp-name>
```

Where:

as name
: name of AS

asp name
: name of ASP

Example 50. Remove ASP from AS

```
mobicents(127.0.0.1:3435)>m3ua as remove AS1 ASP1
Successfully removed ASP name=ASP1 from AS name=AS1
```

Add Route

Configure the destination point code that message will be routed to

```
m3ua route add <as-name> <dpc> <opc> <si>
```

Where:

as name
: name of AS created earlier

dpc
: Destination point code

opc
: Originating point code

si
: Service Indicator

Example 51. Add Route

```
mobicents(127.0.0.1:3435)>m3ua route add AS1 2 -1 -1
```

Remove Route

Remove the As configured for the destination point code

```
m3ua route remove <as-name> <dpc> <opc> <si>
```

Where:

as name
: name of AS assigned to route message for this dpc

dpc
: Destination point code

opc
: Originating point code

si
: Service Indicator

Example 52. Remove Route

```
mobicents(127.0.0.1:3435)>m3ua route remove AS1 2 -1 -1
```

Show Route

Show all the routes configured

```
m3ua route show
```

Example 53. Show Route

```
mobicents(local)>m3ua route show

2:-1:-1        AS1,AS2,
```