CLI

- Overview
- Trunk
- Trunk Handler
- Phonegroup
- Extensions
- Route
- Class of Service
- Trunk Groups

Overview

The CLI is a powerful tool for viewing and editing your current system configuration.

To use the CLI, you will need to access your phone system's console via a superuser. The system console is available either at the local terminal or via SSH. We recommend the following SSH Clients:

- Windows: PuTTY
- Mac: Terminal, or iTerm2, and ssh
- Linux: ssh or PuTTY

> Warning: With great power comes great responsibility. Most changes done via the CLI are applied Immediately.

> Please be familiar with all commands prior to using the CLI

Test the firezirrah

Commands

Please note that all CLI commands are case-sensitive.

There are two modes available to you when working in the CLI:

- Command

Used for informational purposes, such as viewing current active call status or current device availability status.

Used for controlling real-time aspects of the system, such as rebooting, disconnecting calls, or logging out call center agents

This mode is the default mode when entering the CLI

- Configure

Used for changing configurations for any settings, or viewing existing configurations for specific nodes.

This mode can be accessed from the root of Command Mode by typing **configure**

Command Mode

callqueue

agent

Agent command description

queues

Queues command description

clear

voicemail

box

greeting

configure

terminal

disable

exit

mtr

ip

ipv6

ping

ip

ipv6

arp

reboot

services

restart

stop

start

Show

callqueue

agent

ogins

Show currently logged in agents

assignments

Show current agent assignments by call queue

queues

status

Show current queue statuses

config

Show current queue configurations

polycom

status

Show current device status

config

Show current device configuration

calls

status

List all current active calls

cos

List only CoS

includes

List all CoS as well as any includes

devices

sip

status

iax

status

polycom

status

diag

Show diagnostic information?

extensions

status

interfaces

Show all interfaces and their current configuration/status

ip

interface

route

licenses

Display current available and assigned licenses

phonegroups

Display all current phonegroup information

polycoms

Show only installed polycoms on the system

registrations

sip

iax

all

routes

Show route information

running-config

Show full configuration for all configured nodes

services

Show running services

detail

Show running services actual process details

startup-config

Show startup-config (currently failing)

system-users

Show system users, not sure what they relate to

tenants

Show all current tenants on the active system

trunks

Show all available trunks configured on the system

traceroute

Configure Mode

---

clone

daynight_group

extension

route

core_settings

beep_on_xfer

enable_phone_reprogram_feature

enable_steal_holding_call

externhost

externip

externrefresh

externtcpport

externtlsport

hosted_mode

localnet

pbx2_host

pbx2_port

pbx2_reg_period

pbx2_transport

pbx_ip

pbx2_reg_period

pbx_transport

ringseconds_till_voicemail

server_name

sip_port

core_sip_settings

dtmfmode

qualify

qualifyfreq

rtpholdtimeout

rtptimeout

cos

Class of Service (CoS) takes a second argument which specifies either:

- A new CoS name you wish to create and configure, e.g. "cos foo"
- An existing CoS name you wish to make changes to, e.g. "cos foo" where "foo" is an already existing CoS

Trunk

A **Trunk** is a link to another system, such as a phone carrier, pbx, etc, to send and receive calls. A **Trunk** will need a **Trunk Handler** defined beforehand to function properly.

Adding a Trunk

1) Enter the CLI's configuration mode

2) Ensure you have already made a **Trunk Handler**, while you can create a **Trunk** without defining one, you will still need to define one to ensure that the phone system is properly configured.

3) Enter the name of the **Trunk** you wish to create. You will be prompted with the minimum required settings

```
vbox-markm(config)# trunk building_a
!! Notice: Editing new item: building_a
!! Notice: Minimum requirements needed for insert:
!!   device_type
!!   desc
!!   capacity
vbox-markm(config-trunk building_a)#
```

4) Enter the minimum required settings for adding the **Trunk**, you can use the **"?" key** to get a description of all the available settings.

```
vbox-markm(config-trunk building_a)# capacity 50
vbox-markm(config-trunk building_a)# desc "Main building"
vbox-markm(config-trunk building_a)# device_type
SIP         IAX2        DAHDI       VIRTUAL     SIP-VIRTUAL       ### Used tab to get a list
of the available devices.
vbox-markm(config-trunk building_a)# device_type SIP
```

5) After adding the trunk, use the **show** command to ensure your settings have been saved.

```
vbox-markm(config-trunk building_a)# show
! Generating running-config
```

```
!!!!

! -------------- Trunks ----------------
trunk building_a
  capacity 50
  desc "Main building"
  device_type SIP
  prestart_media no
  record no
  trunk_handler_name default
  sip
    allow ulaw,alaw,adpcm,gsm
    call-limit 50
    canreinvite no
    disallow g723,slin,ilbc,lpc10,g729,speex,g726aal2,g722
    host dynamic
    nat yes
    qualify yes
    rtpholdtimeout 60
    rtptimeout 60
    secret 9a97c729fdce1f0a5d68495f82c5159e
    sendrpid yes
    trustrpid no
    username building_a
    exit
  exit

exit

vbox-markm(config-trunk building_a)#
```

6) Next define a **Trunk Handler** that the **Trunk** will use.

```
vbox-markm(config-trunk building_a)# trunk_handler_name
audiocodes                local_loop                pstn                      nycvoip
branch                    nbs- 9734067200           default                   localyny
chvoip                    Nathans_Test_TrunkHandler
vbox-markm(config-trunk building_a)# trunk_handler_name Nathans_Test_TrunkHandler
```

7) Lastly, do a **show** command to ensure your settings are saved.

```
vbox-markm(config-trunk building_a)# show
! Generating running-config
!!!!

! -------------- Trunks ----------------
trunk building_a
  capacity 50
  desc "Main building"
  device_type SIP
  prestart_media no
  record no
  trunk_handler_name Nathans_Test_TrunkHandler
  sip
    allow ulaw,alaw,adpcm,gsm
    call-limit 50
    canreinvite no
    disallow g723,slin,ilbc,lpc10,g729,speex,g726aal2,g722
    host dynamic
    nat yes
    qualify yes
    rtpholdtimeout 60
    rtptimeout 60
    secret 9a97c729fdce1f0a5d68495f82c5159e
    sendrpid yes
    trustrpid no
    username building_a
    exit
  exit

exit

vbox-markm(config-trunk building_a)#
```

Trunk Settings Description

You can also use the "**?**" key to view the description while in the CLI. * indicates a required setting.

callerid_name	Force this callerid name when receiving calls from this trunk
callerid_number	Force this callerid number when receiving calls from this trunk
capacity	(*) Maximum number of concurrent calls to/from this trunk
desc	(*) Non-functional description
device_type	(*) Protocol that this device will use
do	To run exec commands in config mode
exit	Exit from this level (with notification of incomplete config)
iax	Change IAX-specific trunk settings
incoming_context_label	
no	Clear settings
prestart_media	
record	Whether to record all calls to/from this device
show	Show the running-config for the current item
sip	Change SIP-specific trunk settings
trunk_handler_name	Name of the trunkhandler to use for incoming call processing

Trunk Handler

Defines a context that will handle incoming call routing.

Adding a Trunk Handler

1) Enter the CLI's configuration mode

2) Enter the name of the **Trunk Handler** you wish to create. You will be prompted with the minimum required settings.

```
vbox-markm(config)# trunk_handler chvoip
!! Notice: Editing new item: chvoip
!! Notice: Minimum requirements needed for insert:
!!   desc
vbox-markm(config-trunk_handler chvoip)# desc "Incoming Calls from Chicago"
vbox-markm(config-trunk_handler chvoip)#
```

3) Use the show command to ensure your settings are saved.

```
vbox-markm(config-trunk_handler chvoip)# show
! Generating running-config
!!

! -------------- Trunk Handlers ----------------
trunk_handler chvoip
  desc "Incoming Calls from Chicago"
  exit

exit

vbox-markm(config-trunk_handler chvoip)#
```

Trunk Handler Settings Description

You can also use the **"?" key** to view the description while in the CLI. * indicates a required setting.

Setting	Description
desc	(*) Non-functional description
do	To run exec commands in config mode
exit	Exit from this level (with notification if config was incomplete)
no	Clear settings
show	Show the running-config for the current item

Phonegroup

Phonegroups serve as a collection of phones. **Phonegroups** also serve as a default caller ID when calls are placed outside via a carrier **trunk**. It will also do this on a per-trunk basis for 911 calls.

Adding a Phonegroup

1) Enter the CLI's configuration mode

2) Enter the name of the **Phonegroup** you wish to create. You will be prompted with the minimum required settings.

```
servername(config)# phonegroup kitchen_installs
!! Notice: Editing new item: kitchen_installs
!! Notice: Minimum requirements needed for insert:
!!   callerid_name
!!   callerid_number
!!   desc
servername(config-phonegroup kitchen_installs)#
```

3) Choose settings appropriate to your organization's needs. You can use the **"?" key** on your keyboard to get a description of the various settings.

```
servername(config-phonegroup kitchen_installs)# callerid_name "Bob's Carpet Company"
servername(config-phonegroup kitchen_installs)# callerid_number 5183207755
servername(config-phonegroup kitchen_installs)# desc "Kitchen installs for Bob's Carpet"
servername(config-phonegroup kitchen_installs)# show
```

4) After configuring everything, use the show command to ensure the settings have been saved.

```
servername(config-phonegroup kitchen_installs)# show
! Generating running-config
!!!

! -------------- Phonegroups ----------------
phonegroup kitchen_installs
  callerid_name "Bob's Carpet Company"
```

```
   callerid_number 5183207755
   desc "Kitchen installs for Bob's Carpet"
   record no
   tenant_name default
   exit

 exit
```

Phonegroup Settings Description

You can also use the "**?**" key to view the descriptions while in the CLI. * indicates a required setting.

Setting	**Comments**
callerid_name	Force this callerid name when receiving calls from this trunk
callerid_number	Force this callerid number when receiving calls from this trunk
capacity	(*) Maximum number of concurrent calls to/from this trunk
desc	(*) Non-functional description
device_type	(*) Protocol that this device will use
do	To run exec commands in config mode
exit	Exit from this level (with notification of incomplete config)
iax	Change IAX-specific trunk settings
incoming_context_label	
no	Clear settings
prestart_media	
record	Whether to record all calls to/from this device
show	Show the running-config for the current item

Setting	Comments
sip	Change SIP-specific trunk settings
trunk_handler_name	Name of the trunkhandler to use for incoming call processing

Extensions

Extensions are part of **phonegroups** and serve as end points for calls, or where calls can be placed from. **Extensions** are set to use a **Class of Service** to define how calls from this extension are handled.

Adding an Extension

1) Enter the CLI's configuration mode.

2) Enter the number of the **Extension** you wish to create. You will be prompted with the minimum required settings. An **extension** does not necessarily need to be a number, but in most this cases this is recommended.

```
servername(config)# extension 5000
servername(config-extension 5000)#
```

4) Select a device type. (Press "Tab" after `device_type` to see available options)

```
servername(config-extension 5000)# device_type
SIP         IAX2        DAHDI       VIRTUAL     SIP-VIRTUAL
servername(config-extension 5000)# device_type SIP
servername(config-extension 5000)#
```

Option	Description
SIP	Physical phone (such as Polycom) or a software phone (such as Linphone or X-lite)
IAX2	Inter-Asterisk exchange
DAHDI	Phone cards physically installed in computers, not commonly used
VIRTUAL	Used mainly for mailboxes
SIP-VIRTUAL	Acts as a secondary extension (or tertiary, if using two) for a SIP device, but is still considering a separate extension.

5) By default the extension is created using your system defaults. Use the **show** command to see the default values for the extension.

```
servername(config-extension 5000)# show
! Generating running-config
!!!!!

! -------------- Extensions ----------------
extension 5000
  always_autoanswer no
  callerid_number 5000
  call_screening no
  cos internal+local+ld
  device_type SIP
  in_directory yes
  phonegroup internal
  phonemake_id generic
  phonemodel_id generic
  phone_type Generic
  record no
  tenant_name default
  voicemail_enable no
  sip
    allow ulaw,alaw,adpcm,gsm
    call-limit 4
    canreinvite no
    disallow g723,slin,ilbc,lpc10,g729,speex,g726aal2,g722
    host dynamic
    nat yes
    qualify yes
    rtpholdtimeout 60
    rtptimeout 60
    secret 35712c6fd77554c7d6902ebb5831030f
    sendrpid yes
    trustrpid no
    username 5000
    exit
  exit

exit
```

6) A couple of settings that may need to be changed first are **Phonegroups** and **Class of Service (cos)**. Let's see what options we have for these.

```
servername(config-extension 5000)# phonegroup   ### Use the tab key to a get a list of the
available options
carpet_agents       hard_surface_agents internal
servername(config-extension 5000)# cos
internal+local+ld+intl internal+local+ld      internal+local         internal
```

7) Change the **phonegroup**.

```
servername(config-extension 5000)# phonegroup
carpet_agents       hard_surface_agents internal
servername(config-extension 5000)# phonegroup carpet_agents
```

8) Next change the **Class of Service (cos)**

```
servername(config-extension 5000)# cos
internal+local+ld+intl internal+local+ld      internal+local         internal
servername(config-extension 5000)# cos internal+local
```

9) Afterwards check to ensure the changes were saved.

```
servername(config-extension 5000)# show
! Generating running-config
!!!!!

! -------------- Extensions ----------------
extension 5000
  always_autoanswer no
  callerid_number 5000
  call_screening no
  cos internal+local         ### changes saves successfully
  device_type SIP
  in_directory yes
  phonegroup carpet_agents   ### changes saves successfully
  phonemake_id generic
  phonemodel_id generic
  phone_type Generic
  record no
  tenant_name default
  voicemail_enable no
  sip
```

```
    allow ulaw,alaw,adpcm,gsm
    call-limit 4
    canreinvite no
    disallow g723,slin,ilbc,lpc10,g729,speex,g726aal2,g722
    host dynamic
    nat yes
    qualify yes
    rtpholdtimeout 60
    rtptimeout 60
    secret 35712c6fd77554c7d6902ebb5831030f
    sendrpid yes
    trustrpid no
    username 5000
    exit
  exit

exit
```

Voicemail Setup

Voicemail must be enabled manually on an **extension**. On most systems, voicemail is disabled by default

1) Enter the CLI's configuration mode.

2) Select the **extension** to set up voicemail on.

```
vbox-markm(config)# extension
1000 1001 1005 1234 5000 1050 AB   560
vbox-markm(config)# extension 5000
vbox-markm(config-extension 5000)#
```

3) Enable voicemail using the following command.

```
vbox-markm(config-extension 5000)# voicemail_enable yes
```

4) Use the show command to show voicemail has been enabled.

```
vbox-markm(config-extension 5000)# show
! Generating running-config
!!!!!

! -------------- Extensions ----------------
extension 5000
  always_autoanswer no
  callerid_number 5000
  call_screening no
  cos internal+local+ld
  device_type SIP
  in_directory yes
  phonegroup carpet_agents
  phonemake_id generic
  phonemodel_id generic
  phone_type Generic
  record no
  tenant_name default
  vmail_phonegroup_override internal
  voicemail_enable yes              ### Voicemail has been enabled
  sip
    allow ulaw,alaw,adpcm,gsm
    call-limit 4
    canreinvite no
    disallow g723,slin,ilbc,lpc10,g729,speex,g726aal2,g722
    host dynamic
    nat yes
    qualify yes
    rtpholdtimeout 60
    rtptimeout 60
    secret 35712c6fd77554c7d6902ebb5831030f
    sendrpid yes
    trustrpid no
    username 5000
    exit
  voicemail                         ### Voicemail has been enabled, when it's disabled this
and it's settings will not show.
    format sln
    override_phonegroup_name internal
    pin 5000
```

```
    voicemail yes
    exit
  exit

exit
```

5) There are additional voicemail options, such as setting an email to forward voicemail.

```
vbox-markm(config-extension 5000)# voicemail
vbox-markm(config-extension 5000 voicemail)#                          ### Use ? to see the
available options, it will also show the description as shown below
  attach                        Enable/Disable attach voicemail recording to email
  callback                      Context to call back from
  delete                        Enable/Disable deleting voicemail from the server after
notification
  dialout                       Context to dial out from (option 4 from advanced menu if
enabled)
  do                            To run exec commands in config mode
  email                         Email address to use when sending voicemail email notification
  envelope                      Enable/Disable envelope information playback before message
playback
  exit                          Exit from this level (with notification of incomplete config)
  forcegreetings                Force greeting to be recorded if the mailbox password is the
same as the mailbox number
  forcename                     Force name to be recorded if the mailbox password is the same
as the mailbox number
  format                        Format for the voicemail recording
  fromstring                    The 'From' name when sending a voicemail email notification
  hidefromdir                   Enable/Disable hiding this mailbox from the company directory
  maxgreetingstime              Maximum number of seconds for a voicemail greeting
  maxlogins                     Maximum number of failed login attempts before hanging up on
user
  maxmsgs                       Maximum number of messages that this voicemail box can contain
before it is considered full
  maxsecs                       Maximum number of seconds that a voicemail message can be
  minsecs                       Minimum number of seconds that a voicemail must be in order to
be saved
  nextaftercmd                  Enable/Disable automatically proceeding to the next message
after the user enters a command
  no                            Clear setting
```

```
  operator                      Enable/Disable the ability for the caller to dial 0 from
voicemail
  override_phonegroup_name      Optionally put this mailbox into a different phonegroup than
the one the extension is a part of
  pin                           (*) Voicemail pin number
  portal_voicemail_folders_hide
  review                        Enable/Disable the ability for the caller leaving a message to
review and/or rerecord message before saving it
  saycid                        Enable/Disable say callerid for voicemail message
  sayduration                   Enable/Disable saying the message duration before playing the
message
  saydurationm                  Minimum number of minutes of duration needed when saying the
duration
  sendvoicemail                 Enable/Disable the ability for the user to compose and send a
voicemail while inside voicemail (option 5 from advanced menu if enabled)
  show                          Show the running-config for the current item
  silencethreshold              What level do we consider silence.  Lower numbers mean more
sensitive
  tz                            Timezone to use for this mailbox
  voicemail
```

6) Enable the voicemail to email feature, set the email where voicemail should be forwarded, and then use the **show** command to ensure your settings were saved.

```
vbox-markm(config-extension 5000 voicemail)# attach yes                         ### enables
attachment of voicemail on emails
vbox-markm(config-extension 5000 voicemail)# email notarealemail@zeromail.com   ### email
address voicemail will be sent to.
vbox-markm(config-extension 5000 voicemail)# show
! Generating running-config
!!!!!

! -------------- Extensions ----------------
extension 5000
  always_autoanswer no
  callerid_number 5000
  call_screening no
  cos internal+local+ld
  device_type SIP
  in_directory yes
```

```
  phonegroup carpet_agents
  phonemake_id generic
  phonemodel_id generic
  phone_type Generic
  record no
  tenant_name default
  vmail_phonegroup_override internal
  voicemail_enable yes
  sip
    allow ulaw,alaw,adpcm,gsm
    call-limit 4
    canreinvite no
    disallow g723,slin,ilbc,lpc10,g729,speex,g726aal2,g722
    host dynamic
    nat yes
    qualify yes
    rtpholdtimeout 60
    rtptimeout 60
    secret 35712c6fd77554c7d6902ebb5831030f
    sendrpid yes
    trustrpid no
    username 5000
    exit
  voicemail
    attach yes                          ### saved successfully
    email notarealemail@zeromail.com    #### " "
    format sln
    override_phonegroup_name internal
    pin 5000
    voicemail yes
    exit
  exit

exit
```

> It is important to remember to use the correct email address. That being said, it is NOT recommended to disable saving the voicemail on the extension itself and only have it sent through email since your voicemail could be lost if there is something wrong outside of the Intellasoft platform (transmission issues, email server (SMTP) issues).

7) Lastly it is important to set a PIN for the voicemail, so the user of the extension can access their voicemail securely and prevent unwanted access.

```
vbox-markm(config-extension 5000 voicemail)# pin 0011
```

Extension Settings Description

You can also use the "**?**" key to view the description while in the CLI. * indicates a required setting.

Setting	Description
all_call_forward	Call forwarding
always_autoanswer	Whether or not to send an always answer header when calling this extension
autoanswergroup_id	
busy_call_forward	If the extension is busy or unava
call_screening	
callerid_name	Caller ID name
callerid_number	Caller ID number
cos	Class of service when this extension makes a call
department_name	
device_type	(*) Protocol that this device will use
did	Inbound DID (Direct Inward Dialing)
do	To run exec commands in config mode
exit	Exit from this level (with notification of incomplete config)
external_callerid	What callerid to use when the extension dials out a route
iax	Change IAX-specific extension settings
in_directory	Enable/Disable showing this extension in company directories
no	Clear settings
override_extension	Override the dialable extension
phone_type	Extension type
phonegroup	Phonegroup that this extension is a member of
phonemake_id	

Setting	Description
phonemodel_id	
polycom	MAC Address of the physical phone this extension is assigned to
polycom_reg	Registration number on the physical phone (ie: polycom reg number)
record	Whether to record all calls to/from this device
ringseconds_till_voicemail	
show	Show the running-config for the current item
sip	Change SIP-specific extension settings
tenant_name	Tenant that owns this phone
twinning_number	Twinning number. Ring an external number such as a cell phone.
user_id	
username	Web portal username that is assigned to this extension (Note: a username can be assigned to more than one extension)
vmail_phonegroup_override	Override what phonegroup this extension's mailbox is a member of
vmail_subscribe_override	Override what mailbox this extension subscribes to
voicemail	Change voicemail settings
voicemail_enable	Enable/disable voicemail for this extension

Route

Routes define what **Trunk Group** to use when a certain **Pattern** is dialed.

Adding a Route

1) Enter the CLI's configuration mode

2) Enter a name for the new **Route**. You will then be prompted to enter the minimum options needed to create the new route.

```
vbox-nathan(config)# route      ### use tab to get an available list of routes if you are
unsure
ld11_reg local    ld10     ld11     intl     ld10_reg emerg
vbox-nathan(config)# route A_Route
!! Notice: Editing new item: A_Route
!! Notice: Minimum requirements needed for insert:
!!   pattern                    ### see next step
!!   trunk_group
!!   desc
vbox-nathan(config-route A_Route)# desc "Local and Long Distance Numbers +9"
vbox-nathan(config-route A_Route)# trunk_group  ### use tab to get a list of the trunks you
have made
pbx-sip-t1 default
vbox-nathan(config-route A_Route)# trunk_group pbx-sip-t1
```

3) Next enter a **Pattern** for the route. The **Pattern** represents the combination of numbers that the **Trunk Group** will use when making a call. In many cases, there will be multiple **routes** to define multiple types of **patterns**. Below is an example of what default **Routes** look like on a freshly installed system.

```
-------------------------------------------------------------------------------------------
-------------------------------------
|  Route     |  Desc.                              |  Trunk Group  |  Pattern       |
Replace  |  Remove  |  Prepend  |  Postpend  |
-------------------------------------------------------------------------------------------
-------------------------------------
```

```
|  ld11_reg  |  Long Distance 11 Digit without a 9  |  pbx-sip-t1   |  1XXXXXXXXX
|            |           |           |            |
|  intl      |  International                       |  pbx-sip-t1   |  9011X.
|            |  +1       |           |            |
|  ld11      |  Long Distance 11 Digit              |  pbx-sip-t1   |  91NXXXXXXXX
|            |  +1       |           |            |
|  emerg     |  Emergency                           |  pbx-sip-t1   |  9911
|            |  +1       |           |            |
|  ld10      |  Long Distance 10 Digit              |  pbx-sip-t1   | 9NXXXXXXXX
|            |  +1       |  1        |            |
|  local     |  Local 7 Digit                       |  pbx-sip-t1   |  9XXXXXX
|            |  +1       |           |            |
|  ld10_reg  |  Long Distance 11 Digit without a 9  |  pbx-sip-t1   |  XXXXXXXXX
|            |           |  1        |            |
-------------------------------------------------------------------------------------------
--------------------------------------
```

Pattern

As you can see above **patterns** are not limited to just numbers only, but rather can include specific characters to broaden or narrow down the range of types of calls that the routes will take. You are able to use the following characters to help further define a route that is to be taken:

- `X` matches any digit from 0-9
- `Z` matches any digit from 1-9
- `N` matches any digit from 2-9
- `[1237-9]` matches any digit or letter in the brackets (in this example, 1,2,3,7,8,9)
- `.` wildcard, matches one or more characters
- `!` wildcard, matches zero or more characters immediately
- `[a-z]` matches any lowercase letter (not frequently used/recommended)
- `[A-Z]` matches any UPPER case letter (not frequently used/recommended)

Replace (processed first)

The replace function will replace a **pattern** of numbers completely with the defined number included in the replace function.

- Example: The number 518-268-3769 matches the **pattern** of a **route** with a replace function set to 5000. Instead of using the 518 number it uses the 5000 number instead plus any of the other functions listed thereafter (such as remove, prepend or postpend)

Remove (processed second)

The remove function will remove numbers from the **beginning or end** of a **pattern** depending on how it is configured. For example, a `+1`, will remove the first digit from a number pattern, whereas `-3` would remove the last 3 from a pattern.

- Example 1: If the number "518-268-3769" matches the pattern of a **Route** with a **remove** of "+1", it will remove the first number ("5") leaving "18-268-3769" (just an example, obviously this is an invalid number unless it were an internal extension).
- Example 2: If the same number as above matches the Pattern of a **Route** with a **remove** of "-3". It will remove the last numbers in the sequence leaving the number "5182683" to be dialed (again just an example).

Prepend (processed third)

Prepend will add a specific string of numbers depending on what numbers are defined. In the default **routes** above, many of the numbers are prepended with a 1 to make them valid long-distance phone numbers that can be used on the PSTN (public switched telephone network).

- Example 1: The number "518-268-3769" matches a **route** with a prepend of 1, so it will add the number 1 in front, making the actual number used by the **trunk handler** "1-518-268-3769".
- Example 2: The number "268-3769" matches a **route** with a prepend of 518, so it will add the numbers "518" in front of the original number so the actual number being used by the **trunk handler** would be 518-268-3769.

Postpend (processed last)

Postpend is the exact opposite of prepend, instead of adding numbers to the **beginning** of a **pattern**, it will add numbers to an **end** of a pattern.

- Example: The number "518-268" matches a **route** with a postpend of "3769", it will add the number "3769" to the end of the original number making the actual number dialed out to the **trunk handler** "518-268-3769".

By using a mix of the above settings, you can dial in highly specific **routes** meant to send specific calls to certain places. In the example below, we are going to use this route as a means of calling a long-distance number when someone is using a 9 dial-out number.

```
vbox-nathan(config-route A_Route)# pattern 9NXXXXXXXXX
vbox-nathan(config-route A_Route)# mangle_remove +1
vbox-nathan(config-route A_Route)# mangle_prepend 1
```

4) Lastly use the **show** command to ensure your settings have been saved.

```
vbox-nathan(config-route A_Route)# show
! Generating running-config
!!

! -------------- Routes ----------------
route A_Route
  callerid passthrough
  desc "Local and Long Distance Numbers +9"
  mangle_prepend 1
  mangle_remove +1
  pattern 9NXXXXXXXXX
  record inherit
  trunk_group pbx-sip-t1
  exit

exit
```

Route Settings Description

You can also use the "**?**" key to view the description while in the CLI. * indicates a required setting.

Setting	**Comments**
callerid	Callerid option for when this route is utilized.
desc	(*) Non-functional description of this route.
do	To run exec commands in config mode
exit	Exit from this level (with notification of incomplete config)
mangle_postpend	Number Mangle: Add digits to the end of the dialed destination. This option is processed third.
mangle_prepend	Number Mangle: Add digits to the beginning of the dialed destination. This option is processed second.
mangle_remove	Number Mangle: Remove digits from the dialed destination. This option is processed first.
mange_replace	Number Mangle: Replace the dialed destination with this. (Enabling this option will override all other number-mangling options)
no	Clear settings
override_callerid_name	Callerid name to force, if this route is utilized.

Setting	Comments
override_callerid_number	Callerid number to force, if this route is utilized.
pattern	(*) Pattern to match dialed destination against that will activate this route.
record	Call recording option for when this route is utilized.
show	Show the running-config for the current item
trunk_group	(*) The trunk group that this route will use to dial out.

Class of Service

Class of services define a **route** to take when an **Extension** is placing a call. A **class of service** can contain multiple components to place a call through in order of a priority system. These are explained in greater detail below. **Hunt Groups** as well as **PhoneMap Groups** also must have a class of service defined to function properly. Most systems will have default **classes of services** (as well as routes) however additional ones can be created in the CLI for more advanced routing configurations.

- **Class of Service**: A Class of service can be defined to take another class of service
- **Route**: A Class of Service can take a route which takes a trunk group, and then in turn a trunk.
- **Phonegroup**: A class of service can directly call out to another **phonegroup**.
- **Literal**: ??

Adding an Extension

1) Enter the CLI's configuration mode

2) Enter the name of the **Trunk Group** you wish to create. You will be prompted with the minimum required settings.

```
vbox-nathan(config)# cos billing_reroute
!! Notice: Editing new item: billing_reroute
!! Notice: Minimum requirements needed for insert:
!!   desc
vbox-nathan(config-cos billing_reroute)# desc "Rerouted calls to billing from specific
extensions"
```

3) After this you will need to define the **includes** that the **class of service** will take. This can be done with the following command which takes a number indicating the first **include** that the **class of service** will attempt to take.

```
vbox-nathan(config-cos billing_reroute)# cos_include 1         ### first include that the
class of service will attempt to take
vbox-nathan(config-cos billing_reroute pos-1)# include         ### use tab to get a list of
the potential includes that a class of service can take, remember you are not limited to just
```

```
routes
cos:internal               cos:internal+local         cos:internal+local+ld
cos:internal+local+ld+intl literal:catchall           literal:default
literal:services
phonegroup:Billing         phonegroup:internal        phonegroup:outbound
route:A_Route              route:emerg                route:intl                 route:ld10
route:ld10_reg             route:ld11                 route:ld11_reg             route:local
vbox-nathan(config-cos billing_reroute pos-1)# include route:intl
vbox-nathan(config-cos billing_reroute pos-1)# exit
vbox-nathan(config-cos billing_reroute)# cos_include 2         ### second include a cos will
attempt to take
vbox-nathan(config-cos billing_reroute pos-2)# include route:A_Route
vbox-nathan(config-cos billing_reroute pos-2)# exit
vbox-nathan(config-cos billing_reroute)#
```

4) Use the **show** command to ensure your settings have been saved. In this example, we have only added 2 includes, but you are able to add as many as necessary.

```
vbox-nathan(config-cos billing_reroute)# show
! Generating running-config
!!!

! -------------- Class of Services ----------------
cos billing_reroute
  desc "Rerouted calls to billing from specific extensions"
  cos_include 1
    include route:intl
    exit
  cos_include 2
    include route:A_Route
    exit
  exit

exit
```

Class of Service Settings Description

You can also use the "**?**" key to view the description while in the CLI. * indicates a required setting.

Settings	**Comments**
cos_include	Create or change an include in this class of service
desc	(*) Description
do	To run exec commands in config mode
exit	Exit from this level (with notification of incomplete config)
no	Clear settings
reorder	Change the order of class of service includes
show	Show the running-config for the current item
include	(*) Item to include

Trunk Groups

Defines a list of Trunks to consider when making a call. The system will attempt to call out to trunks in the order they are defined, and if each one fails, the call will not be completed. It is important to have your **Trunks** defined before creating a **Trunk Group**.

Adding a Trunk Group

1) Enter the CLI's configuration mode.

2) Enter the name of the Trunk Group you wish to create. You will be prompted with the minimum required settings.

```
vbox-markm(config)# trunk_group main_office
!! Notice: Editing new item: main_office
!! Notice: Minimum requirements needed for insert:
!!   desc
!!   capacity
vbox-markm(config-trunk-group main_office)# desc "Main Office Trunk Grou"
vbox-markm(config-trunk-group main_office)# capacity 20
vbox-markm(config-trunk-group main_office)#
```

3) Next define the **Trunks** that the **Trunk Group** will call out to. We will assume we are using 3 **Trunks** in this example.

```
vbox-markm(config-trunk-group main_office)# trunk_pos 1
vbox-markm(config-trunk-group main_office pos-1)# trunk_name          ### Use Tab to get a
list of available trunks
AudioCodesFXO1    branch-iax        isoft-sr-in-2     pbx-sip-t1-gw2    building_a
isoft             pbx-sip-t1-gw1    isoft-sr-out-1    Nathan_Test_Trunk isoft-vit-out-1
isoft-sr-in-1
trunk-bob
vbox-markm(config-trunk-group main_office pos-1)# trunk_name building_a
vbox-markm(config-trunk-group main_office pos-1)# exit                ### Returns to main
config options for the trunk group
vbox-markm(config-trunk-group main_office pos-1)# exit force
vbox-markm(config-trunk-group main_office)# show
```

```
! Generating running-config
!!!

! -------------- Trunk Groups ----------------
trunk_group main_office
  callerid passthrough
  capacity 20
  desc "Main Office Trunk Grou"
  record inherit
  trunk_pos 1                       ### This trunk group has been successfully added as
position 1
    trunk_name building_a
    exit
  exit

exit

vbox-markm(config-trunk-group main_office)#
```

4) Next as needed, add additional **Trunks** to the **Trunk Group.**

```
vbox-markm(config-trunk-group main_office)# trunk_pos 2
vbox-markm(config-trunk-group main_office pos-2)# trunk_name
AudioCodesFXO1    branch-iax        isoft-sr-in-2     pbx-sip-t1-gw2    building_a
isoft             pbx-sip-t1-gw1    isoft-sr-out-1    Nathan_Test_Trunk isoft-vit-out-1
isoft-sr-in-1
trunk-bob
vbox-markm(config-trunk-group main_office pos-2)# trunk_name isoft
vbox-markm(config-trunk-group main_office pos-2)# exit
vbox-markm(config-trunk-group main_office pos-2)# exit force
vbox-markm(config-trunk-group main_office)# trunk_pos 3
vbox-markm(config-trunk-group main_office pos-3)# trunk_name
AudioCodesFXO1    branch-iax        isoft-sr-in-2     pbx-sip-t1-gw2    building_a
isoft             pbx-sip-t1-gw1    isoft-sr-out-1    Nathan_Test_Trunk isoft-vit-out-1
isoft-sr-in-1
trunk-bob
vbox-markm(config-trunk-group main_office pos-3)# trunk_name Nathan_Test_Trunk
vbox-markm(config-trunk-group main_office pos-3)# exit
vbox-markm(config-trunk-group main_office pos-3)# exit force
vbox-markm(config-trunk-group main_office)#
```

5) Lastly use the **show** command to ensure your settings have been saved.

```
vbox-markm(config-trunk-group main_office)# show
! Generating running-config
!!!

! -------------- Trunk Groups ----------------
trunk_group main_office
  callerid passthrough
  capacity 20
  desc "Main Office Trunk Grou"
  record inherit
  trunk_pos 1
    trunk_name building_a
    exit
  trunk_pos 2
    trunk_name isoft
    exit
  trunk_pos 3
    trunk_name Nathan_Test_Trunk
    exit
  exit

exit

vbox-markm(config-trunk-group main_office)#
```

Trunk Groups Settings Description

You can also use the "**?**" key to view the option descriptions while in CLI. * indicates a required setting.

Option	Description
callerid	Callerid option for when this trunk group is utilized.
callerid_number_postpend	Append these digits to the callerid when making an outbound call on this trunk group

Option	Description
callerid_number_prepend	Prepend these digits to the callerid on outbound calls after processing removal
callerid_number_remove	Force remove digits from callerid for outbound calls
callerid_number_replace	Force replace callerid with this number for outbound calls on this trunk group
capacity	(*) Maximum number of concurrent calls to/from this trunk group
desc	(*) Non-functional description of this trunk group
do	To run exec commands in config mode
exit	Exit from this level (with notification of incomplete config)
no	Clear settings
record	Whether to record all calls to/from this trunk group
reorder	Change the order of trunk group includes
show	Show the running-config for the current item
trunk_pos	Create or change an include in this trunk group