

REST API

© 2023 Imagine Communications Corp. Proprietary and Confidential.

Imagine Communications considers this document and its contents to be proprietary and confidential. Except for making a reasonable number of copies for your own internal use, you may not reproduce this publication, or any part thereof, in any form, by any method, for any purpose, or in any language other than English without the written consent of Imagine Communications. All other uses are illegal.

This publication is designed to assist in the use of the product as it exists on the date of publication of this manual and may not reflect the product at the current time or an unknown time in the future. This publication does not in any way warrant description accuracy or guarantee the use for the product to which it refers. Imagine Communications reserves the right, without notice to make such changes in equipment, design, specifications, components, or documentation as progress may warrant to improve the performance of the product.

Setup:

```
# examples and defaults
# For Remote SMM:
$smmUrl="https://10.127.1.151"
$fme_ip="10.127.1.12"
$fme_name=<added_by_rest_api>

# For Local SMM (self-hosted):
$fme_ip="172.25.35.187"
$smmUrl="https://$fme_ip:9089"
# or
$smmUrl="https://172.25.35.187:9089"
$fme_ip="127.0.0.1"

$smm_user="admin"
$smm_pass="password"

$remote_user="remote"
$remote_pass="remote"
```

Auth

API Num	Restful API operation	Restful API examples	Comments	Available Since
1	Login and refresh endpoint	curl -k -X POST -H "Content-Type:application/json" -d "{\"username\": \"\$smm_user\", \"password\": \"\$smm_pass\"}" \$smmUrl/api/auth	required by all other APIs	GA1.2
2	Logout endpoint	curl -k -X DELETE -H "Authorization: ./login.sh" \$smmUrl/api/auth		GA1.2
3	Verify remote user credentials	curl -k -X POST -H "Content-Type:application/json" -d "{\"username\": \"\$smm_user\", \"password\": \"\$smm_pass\"}" \$smmUrl/api/authRemoteUser		GA1.2

Presets

API Num	Restful API operation	Restful API examples	Comments	Available Since
4	Get preset list	curl -k -X GET -H "Authorization: ./login.sh" \$smmUrl/api/presets	names returned by query are required by other preset APIs	GA1.2
5	Get the named preset	curl -k -X GET -H "Authorization: ./login.sh" \$smmUrl/api/presets/mypreset >mypreset.prst	use "name" from API # 4	GA1.2
6	Remove a preset	curl -k -X DELETE -H "Authorization: ./login.sh" \$smmUrl/api/presets/mypreset	use "name" from API # 4	GA1.2
7	Create preset from current configuration	preset_name="test_preset" curl -X POST \$smmUrl/api/presets/\$preset_name -k -H "Authorization: ./login.sh" -H "Content-Type: application/json" -d "{\"fme_ip_address\": \"\$fme_ip\", \"name\": \"\$preset_name\", \"element_type\": \"Selenio Network Processor\", \"interface_version\": \"\"}"	will create a new "name"ed preset	GA1.3
8	Import a preset	curl -k -X PUT -H "Authorization: ./login.sh" \$smmUrl/api/presets/preset_name -H "Content-Type: application/json" -d @./preset_name.prst	uses local file named <preset_name.prst>	GA1.3

9	Load preset using preset name	<pre>preset_file="test_preset" reboot=true # (recommended) curl -k -X POST -H "Authorization: ./login.sh" \$smmUrl/api/presets/loadpreset -H "Content-Type: application/json" -d "{\"fme_ip\":\"\$fme_ip\", \"preset_file\":\"\$preset_name\", \"reboot_after\": \"\$reboot\"}"</pre>	use "name" from API # 4	GA1.2
---	-------------------------------	--	-------------------------	-------

Elements

API Num	Restful API operation	Restful API examples	Comments	Available Since
10	Add SNP to SNP Manager (Remote SMM only)	<pre>curl -k -X POST \$smmUrl/api/elements -H "Authorization: ./login.sh" -H "Content-Type:application/json" -d "{\"fme_ip_address\":\"\$fme_ip\", \"fme_name\": \"\$fme_name\", \"remote_user\": \"\$remote_user\", \"remote_pass\": \"\$remote_pass\"}"</pre>	<remote_user> and <remote_pass> should be passed in when adding an element	GA1.2
11	Remove SNP from SNP Manager (Remote SMM only)	<pre>curl -k -X DELETE -H "Authorization: ./login.sh" \$smmUrl/api/elements/\$fme_ip</pre>		GA1.2
12	Rename SNP in SNP Manager (Remote SMM only)	<pre>curl -k -X PUT \$smmUrl/api/elements -H "Authorization: ./login.sh" -H "Content-Type:application/json" -d "{\"fme_ip_address\":\"\$fme_ip\", \"fme_name\":\"\$fme_new_name\"}"</pre>		GA1.2
13	Get list of SNPs in SNP Manager (Remote SMM only)	<pre>curl -k -X GET -H "Authorization: ./login.sh" \$smmUrl/api/elements</pre>		GA1.2
14	Get SNP with given IP (Remote SMM only)	<pre>curl -k -X GET -H "Authorization: ./login.sh" \$smmUrl/api/elements/\$fme_ip</pre>		GA1.2
15	Get 'system up' status	<pre>curl -k -X GET -H "Authorization: ./login.sh" \$smmUrl/api/elements/[{\$fme_ip}]/isInitialized</pre>		GA1.2
16	Query SNP System Properties	<pre>curl -k -H "Authorization: ./login.sh" \$smmUrl/api/elements/\$fme_ip</pre>		GA1.2
17	Query Configuration from Object	<pre>curl -k -X GET -H "Authorization: ./login.sh" \$smmUrl/api/elements/\$fme_ip/config /ipAudRx > ipAudRx_config.json cat ipAudRx_config.json</pre>		GA1.2
18	Configure element and remove all existing config	<pre>curl -k -X PUT -H "Authorization: ./login.sh" -H "Content-Type:application/json" -d @./ipAudRx_config.json \$smmUrl/api/elements/\$fme_ip/config/</pre>		GA1.2
19	Configure Object	<pre>curl -k -X PUT -H "Authorization: ./login.sh" -H "Content-Type:application/json" -d @./new_config.json \$smmUrl/api/elements/\$fme_ip/config/ipAudRx</pre>		GA1.2
20	Remove all config from this element	<pre>curl -k -X DELETE -H "Authorization: ./login.sh" \$smmUrl/api/elements/\$fme_ip /config/</pre>		GA1.2
21	Removes a specific config from this element	<pre>curl -k -X DELETE -H "Authorization: ./login.sh" \$smmUrl/api/elements/\$fme_ip /config/ipAudRx</pre>		GA1.2
22	Get Object Status	<pre>curl -k -X GET -H "Authorization: ./login.sh" \$smmUrl/api/elements/\$fme_ip/status /ipAudRx</pre>		GA1.2
23	Get Alarm configuration	<pre>curl -k -X GET -H "Authorization: ./login.sh" \$smmUrl/api/elements/\$fme_ip/alarms /config</pre>		GA1.2
24	Get active alarms	<pre>curl -k -X GET -H "Authorization: ./login.sh" \$smmUrl/api/elements/\$fme_ip/alarms</pre>		GA1.2
25	Configure alarm	<pre>curl -k -X PUT -H "Authorization: ./login.sh" \$smmUrl/api/elements/\$fme_ip/alarms /config -d [{"number": 0, "priority": 10, "enable": true, "raiseDelay": 0, "clearDelay": 0}]</pre>		GA1.2
26	List all alarm logs	<pre>curl -k -X GET -H "Authorization: ./login.sh" \$smmUrl/api/elements/alarm_logs/all</pre>		GA1.2

27	Get alarm logs csv	curl -k -X GET -H "Authorization: `./login.sh`" \$smmUrl/api/elements/alarm_logs/csv > alarm_logs.csv cat alarm_logs.csv		GA1.2
28	Delete all alarm logs	curl -k -X DELETE -H "Authorization: `./login.sh`" \$smmUrl/api/elements/alarm_logs /delAll		GA1.2
29	Delete alarm logs by fme_ip	curl -k -X DELETE -H "Authorization: `./login.sh`" \$smmUrl/api/elements/alarm_logs /\$fme_ip		GA1.2
30	Get license data	curl -k -X GET -H "Authorization: `./login.sh`" \$smmUrl/api/elements/\$fme_ip /licenses		GA1.3
31	Update license	curl -k -X POST -H "Authorization: `./login.sh`" \$smmUrl/api/elements/\$fme_ip/licenses -H "Content-Type:application/json" -d "{\"license_key\":\"32YAI-AX4G9-O65R2-2BQZV-QN\"}"		GA1.3
32	Delete License	curl -k -X DELETE -H "Authorization: `./login.sh`" \$smmUrl/api/elements/\$fme_ip /licenses		GA1.3
33	Get display layouts for MV mode	curl -k -X GET -H "Authorization: `./login.sh`" \$smmUrl/api/elements/\$fme_ip/layouts		GA1.5 (LTD1.4.1)
34	Check preset status	curl -k -X GET -H "Authorization: `./login.sh`" \$smmUrl/api/elements/\$fme_ip /presetstatus	This call checks if the SMM is available for login. Response will contain the state i.e. { "presetLoaded": true/false } If "presetLoaded" is false then the preset is in the loading process, otherwise it is already loaded.	GA1.5
35	Upload display layouts for MV mode	curl -k -X POST -H "Authorization: `./login.sh`" \$smmUrl/api/elements/\$fme_ip /layouts -H "Content-Type:multipart/form-data" -F 'file=@/.../layout.xml'		GA1.6
36	Get list of 3D LUTs	curl -k -X GET -H "Authorization: `./login.sh`" \$smmUrl/api/elements/luts		GA1.7
37	Get list of 3D LUTs on specified SNP	curl -k -X GET -H "Authorization: `./login.sh`" \$smmUrl/api/elements/\$fme_ip/luts		GA1.7
38	Import lut cube file	curl -k -X PUT -H "Authorization: `./login.sh`" -H Content-Type:application/json --data-binary @./myluts.cube \$smmUrl/api/elements/luts/myluts.cube		GA1.7
39	Import lut metadata	curl -k -X PUT -H "Authorization: `./login.sh`" -H Content-Type:application/json --data-binary @./myluts.json \$smmUrl/api/elements/luts/myluts.json		GA1.7
40	Update lut metadata	curl -k -X PUT -H "Authorization: `./login.sh`" -H Content-Type:application/json --data-binary @./myluts.json \$smmUrl/api/elements/luts/myluts		GA1.7
41	Delete the named lut	curl -k -X DELETE -H "Authorization: `./login.sh`" \$smmUrl/api/elements/luts/myluts		GA1.7
??	List all graphics files for MCL mode	curl -k -X GET -H "Authorization: `./login.sh`" \$smmUrl/api/elements/gfx		GA2.2
??	Retrieve MCL graphics file	curl -k -X GET -H "Authorization: `./login.sh`" \$smmUrl/api/elements/gfx/mygfx.mg3		GA2.2
??	Upload graphics file for MCL mode	curl -k -X PUT -H "Authorization: `./login.sh`" \$smmUrl/api/elements/gfx/mygfx.mg3 -H "Content-Type:multipart/form-data" -F 'graphicData=@/.../graphicsFileName.mg3'	This command is subject to disk usage limits. The limit is 2GB on the 32GB SSD and 30GB on the 64GB SSD.	GA2.2
??	Get disk space usage percent of MCL graphics	curl -k -X GET -H "Authorization: `./login.sh`" \$smmUrl/api/elements/gfx/space		GA2.2
??	Delete MCL graphics file	curl -k -X DELETE -H "Authorization: `./login.sh`" \$smmUrl/api/elements/gfx/mygfx.mg3		GA2.2

elements/luts/myluts.json example

myluts.json

```
{
  "description": "3dlut.cube metadata",
  "filename": "mylut(cube",
  "md5sum": "0b06c6b8a1ef84a547ee9efa54cdbb9d",
  "lut_input_colorimetry_tcs": 0,
  "lut_output_colorimetry_tcs": 2,
  "lut_input_range": 0,
  "lut_output_range": 1
}
```

Console

API Num	Restful API operation	Restful API examples	Comments	Available Since
42	Toggle SSH on /off	curl -k -X POST -H "Authorization: ./login.sh" \$smmUrl/api/console/togglesh	works for self-host only	GA1.6
43	Toggle Telnet on/off	curl -k -X POST -H "Authorization: ./login.sh" \$smmUrl/api/console/toggletelnet	blocked by firewall	GA1.6
44	Toggle inetd on /off	curl -k -X POST -H "Authorization: ./login.sh" \$smmUrl/api/console/toggleinetd	blocked by firewall	GA1.6
45	Get system status	curl -k -X GET -H "Authorization: ./login.sh" \$smmUrl/api/console/status	works for self-host only	GA1.3
46	Reboot SMM	curl -k -X POST -H "Authorization: ./login.sh" \$smmUrl/api/console/reboot	works for self-host only	GA1.6
47	Config reset + reboot	curl -k -X POST -H "Authorization: ./login.sh" \$smmUrl/api/console/confreset	works for self-host only	GA1.6
48	Factory reset	curl -k -X POST -H "Authorization: ./login.sh" \$smmUrl/api/console/factreset	works for self-host only	GA1.6
49	Upload new firmware	curl -k -X POST -H "Authorization: ./login.sh" \$smmUrl/api/console/upgrade -F firmwareUpload=@/.../firmware.tgz	works for self-host only	GA1.3
50	Collect logs & download logs	curl -k -X GET -H "Authorization: ./login.sh" \$smmUrl/api/console/logs curl -k -X GET -H "Authorization: ./login.sh" \$smmUrl/api/console/logfile	works for self-host only	GA1.3
51	Update Console Setting *See table below	curl -k -X POST -H "Authorization: ./login.sh" \$smmUrl/api/console/setconfig -H "Content-Type:application/json" -d "{\"operation\":\"set\", \"data\":{\"netMode\":\"dual\"}}"	use with caution since changing the control interface may require new <fm1_ip> or a physical change in the network wiring	GA1.3
52	Set User Password	curl -k -X POST \$smmUrl/api/console/userpassword -H "Authorization: ./login.sh" -H "Content-Type:application/json" -d "{\"old_password\":\"myoldpassword\", \"password\":\"mynewpassword\"}"	works for self-host only	GA1.3
53	Set Root Password	curl -k -X POST \$smmUrl/api/console/rootpassword -H "Authorization: ./login.sh" -H "Content-Type:application/json" -d "{\"old_password\":\"myoldpassword\", \"password\":\"mynewpassword\"}"	works for self-host only	GA1.3
54	Reset Password	curl -k -X POST \$smmUrl/api/console/resetpass -H "Authorization: ./login.sh"	works for self-host only	GA1.6
55	Get current log level	curl -k -X GET \$smmUrl/api/console/getloglevel -H "Authorization: ./login.sh"	works for self-host only	GA2.0
56	Set the log level	curl -k -X POST \$smmUrl/api/console/setloglevel -H "Authorization: ./login.sh" -H "Content-Type:application/json" -d "{\"loglevel\":\"8\"}"	works for self-host only direct alternative to console/setconfig	GA2.0
57	Get remote logger	curl -k -v -X GET \$smmUrl/api/console/getremotelogger -H "Authorization: ./login.sh"	works for self-host only	GA2.0
58	Set the remote logger	curl -k -v -X POST \$smmUrl/api/console/setremotelogger -H "Authorization: ./login.sh" -H "Content-Type:application/json" -d "{\"remotelogger\":\"172.25.20.163:514\"}"	works for self-host only direct alternative to console/setconfig	GA2.0
59	Get the minimum firmware version allowed	curl -k -v -X POST \$smmUrl/api/console/getminfwversion -H "Authorization: ./login.sh"	works for self-host only	GA2.2

*List of settings to change with api/console/setconfig:

Setting	Parameter	Possible Values	Description
Control Link Bonding	"netMode"	"bondedActiveBackup"	Bonded Active/Backup
		"bondedLACP"	Bonded LACP
		"dual"	Dual Addresses
System Logging Control: Log Level	"logLevel"	"1"	EMERGENCY
		"2"	ALERT
		"3"	CRITICAL
		"4"	ERROR
		"5"	WARNING
		"6"	NOTICE
		"7"	INFORMATION
		"8"	DEBUG
System Logging Control: New Remote Logger	"remoteLogger"	"IP:PORT"	ex: 0.0.0.0:0
Control Link Configuration	"interfaces": [{ "gateway": "0.0.0.0", "id": "ctrl1", "ip": "172.25.75.12", "mask": "255.255.0.0", "dhcp": false }]	See table below	Parameters of "interfaces" (gateway, ip, etc.) can be excluded to remain unchanged (apart from id which must be included)

Control Link Configuration "interfaces" parameters:

"interfaces" parameter	Possible Values	Description
"gateway"	"IP"	Gateway IP Address (ex "0.0.0.0")
"id"	"ctrl1"	Control A
	"ctrl2"	Control B
"ip"	"IP"	SNP Control Link IP Address (ex "172.25.75.12")
"mask"	"IP"	Subnet Mask (ex "255.255.0.0")
"dhcp"	true	DHCP on
	false	DHCP off

Users

API Num	Restful API operation	Restful API examples	Comments	Available Since
59	Get users list	curl -k -X GET -H "Authorization: `./login.sh`" \$smmUrl/api/users/all		GA1.3
60	Get specific user by username	username="admin" curl -k -X GET -H "Authorization: `./login.sh`" \$smmUrl/api/users/\$username		GA1.3
61	Create user	curl -k -X POST \$smmUrl/api/users -H "Authorization: `./login.sh`" -H "Content-Type:application/json" -d @./create_user.json		GA1.3
62	Update user	user_id=2 curl -k -X PUT \$smmUrl/api/users/\$User_id -H "Authorization: `./login.sh`" -H "Content-Type:application/json" -d @./create_user.json		GA1.3

63	Delete user	username="test_name" curl -k -X DELETE -H "Authorization: ./login.sh` \$smmUrl/api/users/\$username		GA1.3
64	Get group list	curl -k -X GET -H "Authorization: ./login.sh` \$smmUrl/api/users/groups		GA1.3

Logos

API Num	Restful API operation	Restful API examples	Comments	Available Since
65	Get known logos and metadata	curl -k -X GET -H "Authorization: ./login.sh` \$smmUrl/api/logos/		
66	About info	curl -k -X GET -H "Authorization: ./login.sh` \$smmUrl/api/logos/about		
67	Get logo parameters	curl -k -X GET -H "Authorization: ./login.sh` \$smmUrl/api/logos/params		
68	Get a logo	curl -k -X GET -H "Authorization: ./login.sh` \$smmUrl/api/logos/logo.jpg >logo.jpg		
69	Remove a logo	curl -k -X DELETE -H "Authorization: ./login.sh` \$smmUrl/api/logos/logo.jpg		

Advanced Audio Processing

API Num	Restful API operation	Restful API examples	Comments	Available Since
70	Add new AAP session	curl -k -X PUT "\$smmUrl/api/elements/\$SNP_IP_ADDRESS/config/system?partial=true" -H 'Content-Type: application/json' -H "Authorization: \$authCode" -d '{ "type": "system", "object_ID": "system", "fme_ip": "\$SNP_IP_ADDRESS", "config": { \"AAP_Algorithms\": [{ \"session\": \"\", \"algorithm\": 1, \"DSP\": 0, \"program\": 5, \"enable\": true, \"delete\": false }] } }'	Creates a new AAP session on the specified program with an algorithm.	LTD 2.0.3
71	Delete existing AAP session	curl -k -X PUT "\$smmUrl/api/elements/\$SNP_IP_ADDRESS/config/system?partial=true" -H 'Content-Type: application/json' -H "Authorization: \$authCode" -d '{ "type": "system", "object_ID": "system", "fme_ip": "\$SNP_IP_ADDRESS", "config": { \"AAP_Algorithms\": [{ \"session\": \"A-2\", \"delete\": true }] } }'		LTD 2.0.3
72	Set AAP session's input gain	curl -k -X PUT "\$smmUrl/api/elements/\$SNP_IP_ADDRESS/config/A-procAudio?partial=true" -H 'Content-Type: application/json' -H "Authorization: \$authCode" -d '{ "type": "procAudio", "object_ID": "A-procAudio", "fme_ip": "\$SNP_IP_ADDRESS", "config": { \"aapSessionConfig\": [{ \"session\": \"A-1\", \"aapSessionAudInConfig\": [{ \"idx\": 1, \"AudChGain\": 360 }] }] } }'	Set the gain on a channel that is an input into the AAP in units of 0.1dB (i.e. 10 x desired dB value)	LTD 2.0.3
73	AAP Input Channel Routing	curl -k -X PUT "https://\$SMM_IP_ADDRESS/api/elements/\$SNP_IP_ADDRESS/config/A-procAudio?partial=true" -H 'Content-Type: application/json' -H "Authorization: \$authCode" -d '{ "type": "procAudio", "object_ID": "A-procAudio", "fme_ip": "\$SNP_IP_ADDRESS", "config": { \"aapSessionConfig\": [{ \"session\": \"A-1\", \"aapSessionAudInConfig\": [{ \"idx\": 1, \"sourceInputType\": 0, \"sourceInputChannel\": 1 }] }] } }'		LTD 2.0.3
74	AAP Input Mute/Invert Toggles	curl -k -X PUT "https://\$SMM_IP_ADDRESS/api/elements/\$SNP_IP_ADDRESS/config/A-procAudio?partial=true" -H 'Content-Type: application/json' -H "Authorization: \$authCode" -d '{ "type": "procAudio", "object_ID": "A-procAudio", "fme_ip": "\$SNP_IP_ADDRESS", "config": { \"aapSessionConfig\": [{ \"session\": \"A-1\", \"aapSessionAudInConfig\": [{ \"idx\": 1, \"AudChMute\": true, \"AudChInvert\": false }] }] } }'		LTD 2.0.3
75	AAP Algorithm Parameter	Similar to above, but passing in one of the codec configs for the config field. Note that the " in the config has to be escaped. This can be done using JSON.stringify.		LTD 2.0.3

NB: The <processor>-procAudio endpoints must reflect the should also be changed between A, B, C, or D based on baseband processor within the SNP.

AAP Session settings	Parameter	Possible Values	Description
Session Identifier	"session"	<processor>-<numerical ID>	<processor> = A, B, C, or D representing the SNP's baseband processors <numerical ID> = 1 based unique identifier within a given processor
AAP Algorithm	"algorithm"	1 2 3 4 5 6 7	Upmix Downmix Multimerge Loudness UM+Loudness DM+Loudness MM+Loudness
DSP selection	"DSP"	0	for on-board DSP

Video Program assignment	"program"	"<processor> <section>-<program number>"	<processor> = A, B, C, or D representing the SNP's baseband processors <section> - the processing section within the SNP's baseband processor <program> - the program number within the processor's section
Session enable	"enable"	"true"	boolean value to enable the session
		"false"	boolean value to disable the session
Delete session	"delete"	"true"	boolean value to delete the session
		"false"	boolean value default value when modifying as session

AAP Configuration Parameters

```
{"aapSessionConfig":[{"session": "A-1", "aapSessionAudInConfig":[{"idx": 1, "AudChGain": 0}]}]}
```

aapSessionAudInConfig: array containing information for all channels

idx: integer (eg: 1)

input channel number to the algorithm. (Number of inputs depends on algorithm). 1-indexed.

AudChGain: integer (eg: 200)

Gain between -360 (-36 dB) to 360 (36 dB) to set the channel to

```
{"aapSessionConfig":[{"session": "A-1", "aapSessionAudInConfig":[{"idx": 1, "sourceInputType": 0, "sourceInputChannel": 1}]}]}
```

aapSessionAudInConfig: array containing info for input channels

idx: integer (eg: 1)

input channel number to the algorithm.

sourceInputType: integer (eg: 0)

Selects the input type to route, where 0 = APIN (DMB and IP), 1 = IP Expansion and MADI, 3 = Test tones.

sourceInputChannel: integer (eg: 1)

Selects the input channel from the selected type to route. APIN and EXP are 1-128, tones is 1-4.

```
{"aapSessionConfig":[{"session": "A-1", "aapSessionAudInConfig":[{"idx": 1, "AudChInvert": false, "AudChMute": false}]}]}
```

aapSessionAudInConfig: array containing info for input channels

idx: integer (eg: 1)

input channel number to the algorithm.

AudChInvert: boolean

whether the channel signal should be inverted (ie: opposite phase)

AudChMute: boolean

whether the incoming signal should be muted.

```
{"aapSessionConfig": [ { "session": "A-1",
```

```
    "aapSessionCodecUpMixConfig": {"implementation": 0, "ChanConfig": 6, "Latency": 1, "DICEProcessLevel": 50, "Depth": 0, "FrontWidth": 0, "SurroundWidth": 100, "LFECutoff": 80, "FinalLimiterCeiling": 0},
```

or

```
    "aapSessionCodecDownMixConfig": {"implementation": 0, "L_R_EncodingMode": 1, "LFECutoff": 80, "FinalLimiterCeiling": 0, "ActiveCorrection": true},  
    "aapSessionCodecMultiMergeConfig": {"InputSelectMode": 1, "InputNoiseFloorThreshold": -60, "InputCrossFadeTime": 250, "MMergeMode": 0, "NoiseFloorThreshold": -60, "CrossFadeTime": 200, "Latency": 1, "UpmixDepth": 0, "UpmixFrontWidth": 0, "UpmixSurroundWidth": 0, "UpmixLFECutoff": 80, "DownmixLREncodingMode": 1, "DownmixLFECutoff": 0, "MultiFinalLimiterCeiling": 0, "StereoFinalLimiterCeiling": 0}
```

or,

```
    "aapSessionCodecLoudnessConfig": {"implementation": 0, "ImsType": 0, "targetLoudness_dBFS": -240, "target_Ratio": 10, "upperLoudnessThresh_dBFS": 10, "lowerLoudnessThresh_dBFS": -10, "freezeWindow_dB": 10, "quietThreshold_dBFS": -400, "attack_msec": 40, "release_msec": 120, "compressorThresh_dB": 20, "compressor_Ratio": 5, "loudnessShaping": 5, "finalLimiterCeiling_dBFS": 0, "bypass": false, "runFinalLimiters": true, "meterAlg": false},
```

```
} ] }
```

Send one of Upmix or Downmix or Loudness configurations as per the original session's algorithm at the time that it was created.

login.sh

login.sh file should be created with execute permissions in the directory in which the example curl commands are run.

login.sh

```
smm_ip="10.127.1.151"
smm_user="admin"
smm_pass="password"
curl -k -X POST -H "Content-Type:application/json" -d "{\"username\": \"$smm_user\", \"password\": \"$smm_pass\"}" $smmUrl/api/auth
```

examples for console/setconfig:

```
{
    "operation": "set"
    "data":
    {
        "logLevel": "4"
    }
}
```

```
{
    "operation": "set"
    "data":
    {
        "interfaces":
        [
            {
                "gateway": "0.0.0.0",
                "id": "ctrl1",
                "ip": "172.25.75.12",
                "mask": "255.255.0.0",
                "dhcp": true
            }
        ]
    }
}
```

preset file template:

preset_name.prst

```
{
    "element_type": "Selenio Network Processor",
    "fme_ip_address": "10.127.1.12",
    "interface_version": "",
    "name": "preset_name",
    "objects": [ ... ],
    "alarms": [ ... ]
}
```

create/edit user template:

create_user.json

```
{  
    "activated": true,  
    "first_name": "FirstName",  
    "last_name": "LastName",  
    "group_id": "2",  
    "password": "password",  
    "username": "test_name"  
}
```