



emSFP Devices REST API

emSFP REST API control and statuses
description.

Document version: D008

25/02/2015
SViengkhou

TABLE OF CONTENT

1	GENERAL	3
1.1	Scope.....	3
1.2	Revision History	3
1.3	References	3
1.4	Abbreviation and Acronyms list.....	4
2	Embrionix REST API description	5
2.1	Implementation	5
2.2	Limitations.....	5
2.3	Representation of a emSFP.....	6
2.3.1	2022-6 Gateways	6
2.4	Resources descriptions and methods.	8
2.5	Resource details and example	10
3	HTTP response descriptions.....	17
4	Annexe 1	17
5	Annexe 2	18

LIST OF TABLES

Table 1-1 : Revision History.....	3
Table 1-2	3

1 GENERAL

1.1 Scope

Description of Embrionix's emSFP REST API. This document describes many common resources of the API supporting control parameters of various SFP.

This document focus on encapsulator and decapsulator SMPTE 2022-6 and NAT type modules and will be complemented for other modules later.

1.2 Revision History

Table 1-1 : Revision History

Revision	Author	Date	Changes - DOCUMENT STILL UNDER DEVELOPMENT
D001	SV	05/10/2015	Initial version
D002	SV	09/11/2015	Added example for gateway type SFP.
D004	SV	10/11/2015	Change representation layout. Added FPGA load in self. Added NAT example.
D005	SV	07/02/2016	Revised documentation for 2022-6
D006	SV	12/02/16	Changed table 2.3.1. Added comments on flow difference ENCAP/DECAP Added notes for not implemented resources/field.
D007	SV	22/02/16	Added HTTP Requester too example in annexe.

1.3 References

Table 1-2

Standard	Org.	Description
[292]	SMPTE	HD-SDI
[INF-8074]	MSA	SFP (Small Form Factor Pluggable) Transceiver
[SFF-8472]	MSA	Diagnostic Monitoring Interface for Optical Transceivers
[348M]	SMPTE	HD-SDTI
[305M]	SMPTE	SDTI
[802.3]	IEEE	Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications", IEEE Std 802.3TM-2002, revision of IEEE Std 802.3, 2000 Edition, that is also known as ISO/IEC 8802-3:2000(E)
[G.826]	ITU	End-to-end performance parameters and objectives for international, constant bit-rate digital paths and connections
RFC-2616	IETF	Hypertext Transfer Protocol (HTTP) Status Code Registry (HTTP)
RFC-4122	IETF	Universally unique identifier

1.4 Abbreviation and Acronyms list

SFP	Small Form Factor
MSA	Multi Source Agreement
SDI	Serial Digital Interface
GE	Gigabit Ethernet

2 Embrionix REST API description

2.1 Implementation

Embrionix's API implementation follows the JTNM model resource description (node/device/source/flow).

For some cases, given resource will be optional, since there is no logical representation. If missing resource is queried, then a HTTP 404 "Not Found" code shall be reported.

Data serialization is done based on JSON (JavaScript Object Notation). XML serialization is not supported.

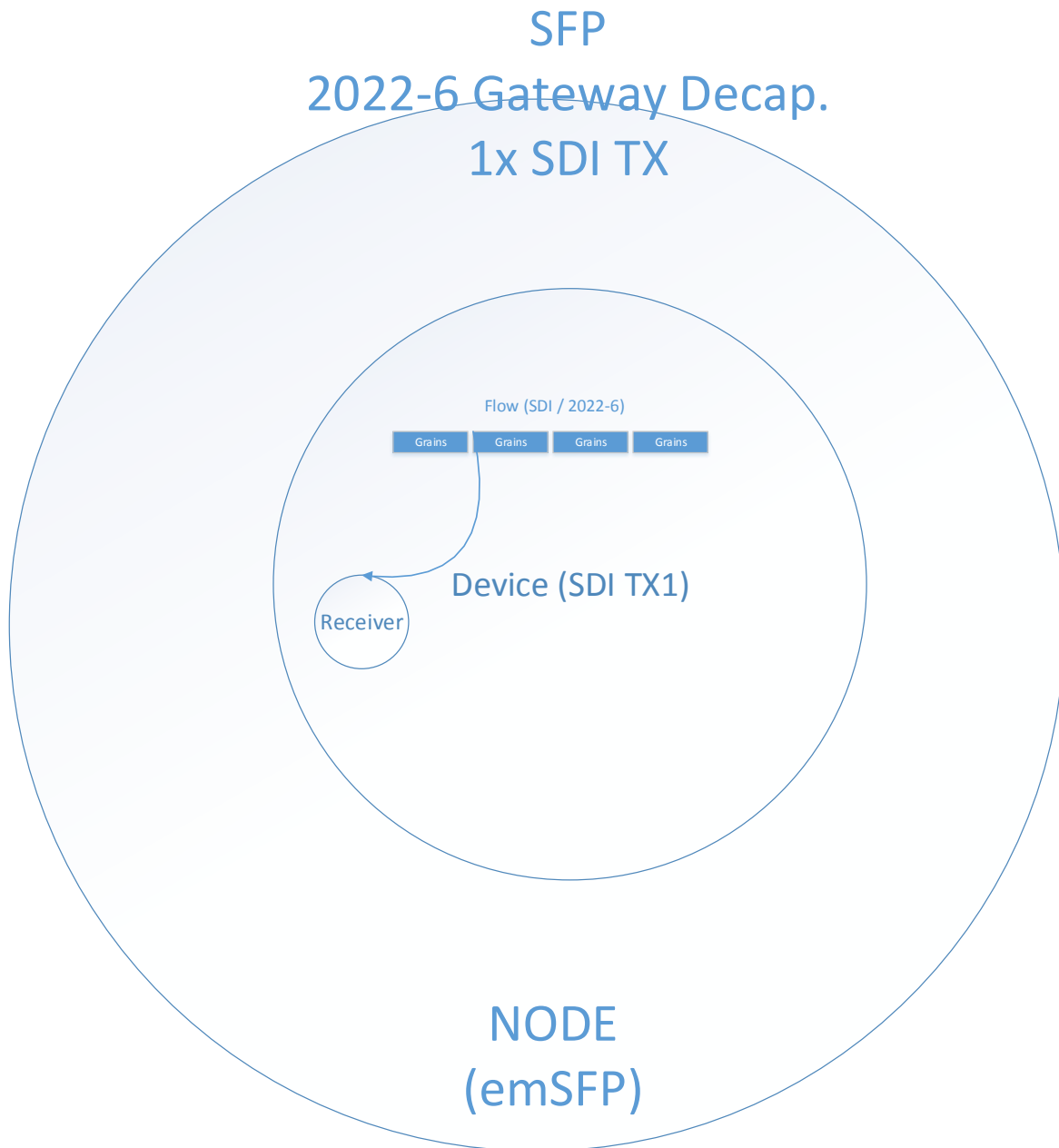
2.2 Limitations

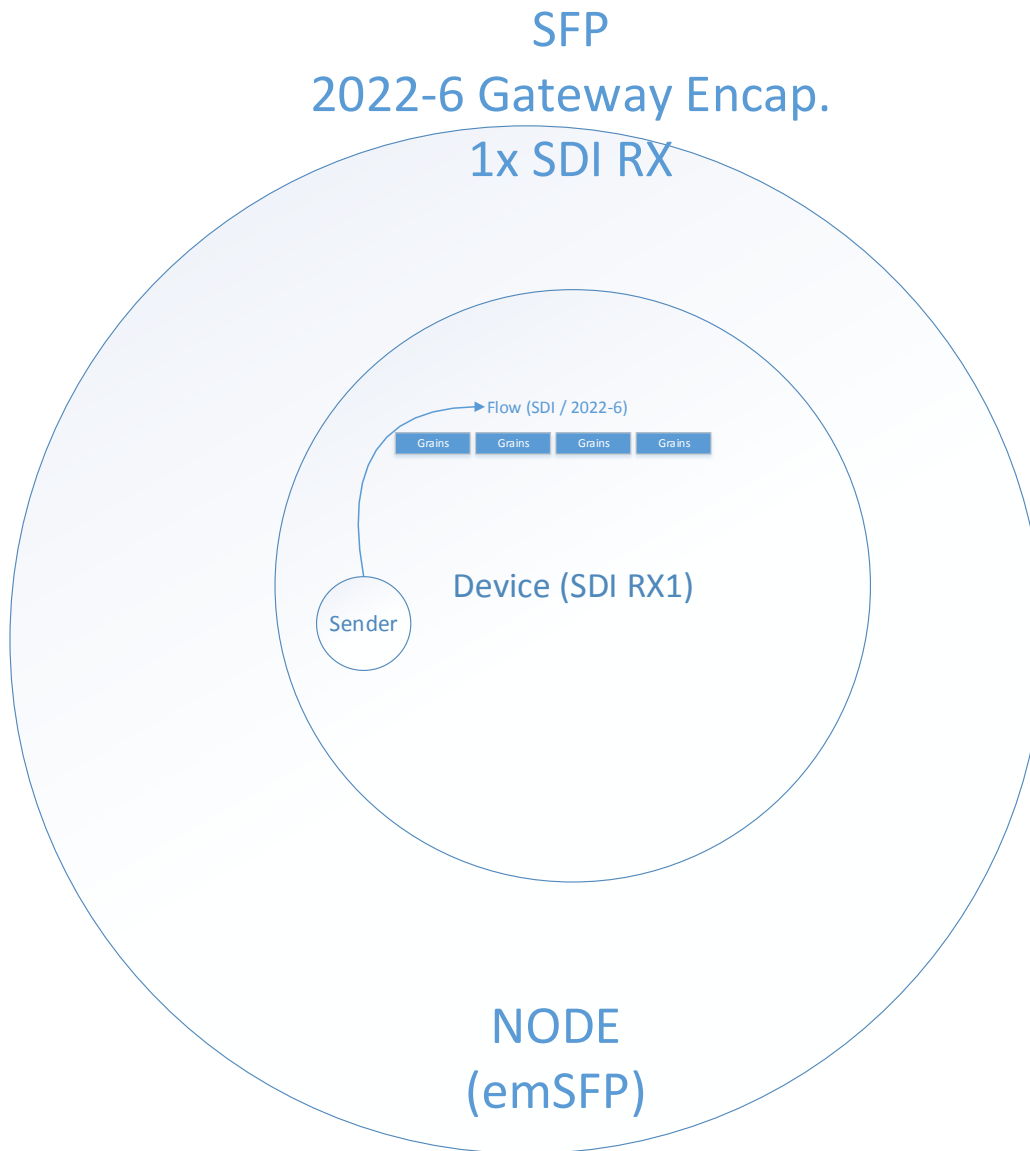
Embrionix REST service support only a subset of HTTP methods of such as PUT, GET, POST. Status codes has also been limited to "200 OK", "400 Bad Request" and "404 Not Found".

Redirection is not supported.

2.3 Representation of a emSFP

2.3.1 2022-6 Gateways





2.4 Resources descriptions and methods.

All queries shall be done under via http://ip_address:port/emsfp/node/vxx/ or <http://emsfp-ax-xx-xx:port/emsfp/node/vxx/> (hostname formed by last 3 bytes of MAC address)

Flow information will depend SFP function.

Resources sff, debug and information are provisioned and not implemented at this point.

Field SDI format, RX/TX packets counts have been provisioned and not implemented at this point.

Resource	URI	Description
Node	emsfp/node/v1	Description: List elements available under for current node.
		Get: returns a list of resource in the node.
		Put: not supported.
		Post: not supported.
Self	emsfp/node/v1/self	Description: Lists the elements under self.
		Get: returns information about this node.
		Put: not supported.
		Post: not supported.
information	emsfp/node/v1/self/information	Description : information on Node component versions
		Get: returns information about this resource.
		Put: not supported.
		Post: not supported.
ipconfig	emsfp/node/v1/self/ipconfig	Description: information on node local ip configuration
		Get: returns information about this resource.
		Put: Modify elements of resource.
		Post: not supported.
sff	emsfp/node/v1/self/sff	Description: Node's SFF-8472 related information.
		Get: Return SFP's A0h, B0h, A2h and B2h information.
		Put: not supported.
		Post: not supported.
debug	emsfp/node/v1/self/debug	Description: Resource to debug module.
		Get: returns information about this resource.
		Put: not supported.
		Post: not supported.
Sources	emsfp /v1/node/sources	Information about the source(s)
		Get: returns information about this node.
		Put: not supported.
		Post: not supported.
Flows	emsfp/node/v1 /flows	Information of the flow
		Get: returns information about this node.
		Put: not supported.
		Post: not supported.
Senders	emsfp/ v1/node/ senders	Virtual output port.
		Get: returns information about this node.
		Put: not supported.
		Post: not supported.

emSFP API Description

Receivers	emsfp /v1/node/ receivers	Virtual input port.
		Get: returns information about this node.
		Put: not supported.
		Post: not supported.

2.5 Resource details and example

Resource	example	Description
Node	["self/", "sources/", "flows/", "devices/", "senders/", "receivers/"]	List of resources available under Node/Vx/

Resource	example	Description
self	["information/", "ipconfig/", "sff/", "debug/"]	List of resources available under self

Resource	example	Description
self/information	{ "current_version" : "1234", "emsfp_version": "A101", "fpag_version" : "f123", "type" : "1", "fpga_slot_00" : "f120", "fpga_slot_01" : "f121", "fpga_slot_02" : "f122", "fpga_slot_03" : "f123", "hw_version" : "A100" }	current_version: sub module current version
		emsfp_version : Vendor revision of the SFP module
		fpag_version: Current version of FPGA load.
		type: Current load type.
		fpga_slot_00: Flash slot 00 FPGA load version.
		fpga_slot_01: Flash slot 01 FPGA load version.
		fpga_slot_02: Flash slot 02 FPGA load version.
		fpga_slot_03: Flash slot 03 FPGA load version.
		hw_version: Hardware revision.

emSFP API Description

Resource	example	Description
self/ipconfig	<pre>{ "version" : "1234", "local_mac" : "00:01:02:03:04:06", "ip_addr" : "192.168.1.2", "subnet_mask" : "255.255.255.0", "gateway" : "192.168.1.0", "hostname" : "emsfp-03-04-05", "port" : "80", "dhcp_enable" : "1", "ctl_vlan_id": "12345", "ctl_vlan_enable" : "1", "data_vlan_id" : "123", "data_vlan_enable" : "1", "bootstatus1" : "255", "bootstatus2" : "255" }</pre>	version: Current resource version. local_mac: Device local MAC address. DO NOT MODIFY. ip_addr: Current device IP address. If DHCP is disabled, device will boot with this ip address. If DHCP is enabled and IP address has been acquired, returns the IP address that has been granted to device. subnet_mask: Current device subnet mask. (see ip_addr) gateway: Current device default gateway. (see ip_addr) hostname: Device hostname. port: Port to access REST API and web page. dhcp_enable: User configurable DHCP flag. Set to 1 to enable, 0 to disable. ctl_vlan_id: Control plane VLAN number (not implemented yet) ctl_vlan_enable: Flag to enable/disable control plane vlan. 1 = control plane VLAN enabled. 0 = Control plane VLAN disable. data_vlan_id: Data plane VLAN number (not implemented yet) data_vlan_enable: Flag to enable/disable data plane vlan. 1 = data plane VLAN enabled. 0 = data plane VLAN disable. bootstatus1: Current boot status. bootstatus2: Previous boot status.

Resource	example	Description
self/sff	<pre>{ "a0": "00,01,02,...", "b0": "00,01,02,...", "a2": "00,01,02,...", "b2": "00,01,02,..." }</pre>	a0: comma separated hexadecimal dump of SFP's EEPROM (A0h) b0: comma separated hexadecimal dump of SFP's EEPROM (B0h) A2: comma separated hexadecimal dump of SFP's digital diagnostic (A2h). B2: comma separated hexadecimal dump of SFP's digital diagnostic (B2h).

emSFP API Description

Resource	example	Description
self/debug	<pre>{ "operation" : "read", "device": "a2", "register" : "1234", "value" : "1234" }</pre>	Reserved for DEBUG PURPOSE ONLY

Resource	example	Description
node/v1/devices	<pre>{ "label" : "2022-6-ecap-01", "version" : "1234", "id" : "67c25159-ce25-4000-a66c-f31fff890265", "node_id" : "3b8be755-08ff-452b-b217-c9151eb21193", "senders" : [], "receivers" : ["d7aa5a30-681d-4e72-92fb-f0ba0f6f4c3e"] }</pre>	label: User defined label (not available yet)
		version: devices register version
		id: device UUID.
		node_id: Node's UUID.
		senders: list of senders UUID available for current device.
		receivers: List of receivers UUID available for current devices.

Resource	example	Description
node/v1/sources	<pre>{ "version" : "1234", "label" : "2022-6-ecap-01", "id" : "4569cea2-ab63-4f97-8dd1-bad4669ea5e4", "device_id" : "67c25159-ce25-4000-a66c-f31fff890265" }</pre>	label: User defined label (not available yet)
		version: devices register version
		id: source UUID.
		device_id: associated device's UUID.

Resource	example	Description
Node/v1/senders	<pre>{ "label" : "2022-6-ecap-01", "version" : "1234", "id" : "d7aa5a30-681d-4e72-92fb-f0ba0f6f4c3e", "flow_id" : "b3bb5be7-9fe9-4324-a5bb-4c70e1084448", "device_id" : "67c25159-ce25-4000-a66c-f31fff890265" }</pre>	label: User defined label (not available yet)
		version: devices register version
		id: sender's UUID.
		flow_id: associated flow's UUID.
		device_id: associated device's UUID.

emSFP API Description

Resource	example	Description
node/v1/receivers	{ "label" : "2022-6-ecap-01", "version" : "1234", "id" : "d7aa5a30-681d-4e72-92fb-f0ba0f6f4c3e", "flow_id" : "b3bb5be7-9fe9-4324-a5bb-4c70e1084448", "device_id" : "67c25159-ce25-4000-a66c-f31fff890265" }	label: User defined label (not available yet)
		version: devices register version
		id: receiver's UUID.
		flow_id: associated flow's UUID.
		device_id: associated device's UUID.

emSFP API Description

Resource	example	Description 2022-DECAP FLOW
node/v1/flows	<pre>{ "version" : "1234", "label" : "2022-6-dec-01", "id" : "b3bb5be7-9fe9-4324-a5bb-4c70e1084448", "source_id" : "b3bb5be7-9fe9-4324-a5bb-4c70e1084449", "type" : "1", "name" : "chan1", "network":{ "src_ip_addr" : "192.168.1.1", "src_udp_port" : "2000", "dst_ip_addr" : "192.168.1.2", "dst_udp_port" : "2001", "dst_mac" : "00:01:02:03:04:05", "vid_clk_rate" : "12345678", "vlan_tag" : "1234", "ssrc" : "1234", "pkt_filter_src_ip" : "0", "pkt_filter_src_udp" : "0", "pkt_filter_src_mac" : "0", "pkt_filter_dst_ip" : "0", "pkt_filter_dst_udp" : "0", "pkt_filter_dst_mac" : "0", "pkt_filter_vlan" : "1", "pkt_filter_ssrc" : "0", "rx_pkt_cnt" : "12345", "rx_pkt_good_cnt" : "12345", "rx_pkt_filtld_cnt" : "12345" }, "format_code_valid" : "1", "format_code_t_scan" : "1", "format_code_p_scan" : "1", "format_code_mode" : "1", "format_code_format" : "1", "format_code_rate" : "123", "format_code_sampling" : "1" }</pre>	version: current flow version. label: User defined label (not available yet) id: flow's UUID. source_id: associated source UUID. type: Flow type. 1 = 2022-6 decapsulator. 2 = 2022-6 encapsulator. name: Flow name. network: IP network related configuration and statuses. src_ip_addr: Source IP address. src_udp_port: Source UDP port. dst_ip_addr: Destination IP address. dst_udp_port: Destination UDP port. dst_mac: Destination MAC address vid_clk_rate: Video clock rate. vlan_tag: Current VLAN tag. Ssrc – Secondary source pkt_filter_src_ip: Packet filter flag for source IP. 1 = Packet will be filtered on source IP address field. 0 = Filter disabled. pkt_filter_src_udp: Packet filter flag for source UDP. 1 = Packet will be filtered on source UDP address field. 0 = Filter disabled. pkt_filter_src_mac: Not implemented. pkt_filter_dst_ip: Packet filter flag for destination IP. 4 = Packet will be filtered on source IP address field. 0 = Filter disabled. pkt_filter_dst_udp: Packet filter flag for destination UDP. 1 = Packet will be filtered on source UDP address field. 0 = Filter disabled. pkt_filter_dst_mac: : Packet filter flag for destination MAC. 1 = Packet will be filtered on for destination MAC address field. 0 = Filter disabled. pkt_filter_vlan: Packet filter flag for VLAN tag. 1 = Packet will be filtered on VLAN tag field. 0 = Filter disabled.

emSFP API Description

		pkt_filter_ssrc: Packet filter flag forSSRC. 1 = Packet will be filtered on SSRC field. 0 = Filter disabled.
		rx_pkt_cnt: Received packet counter (max = 2^32-1).
		rx_pkt_good_cnt: Valid received packet counter (max = 2^32-1).
		rx_pkt_filtd_cnt: Number of received packet discarded by the filter (max = 2^32-1)
		format_code_valid: Flag that indicated that the SDI signal is valid. 1 = valid. 0 = invalid.
		format_code_t_scan: TBD
		format_code_p_scan: TBD
		format_code_mode: TBD
		format_code_format: TBD
		format_code_rate: TBD
		format_code_sampling:TBD

emSFP API Description

Resource	example	Description 2022-ENCAP FLOW
node/v1/flows	<pre>{ "version" : "1234", "label" : "2022-6-ecap-01", "id" : "b3bb5be7-9fe9-4324-a5bb-4c70e1084448", "source_id" : "b3bb5be7-9fe9-4324-a5bb-4c70e1084449", "type" : "1", "name" : "chan1", "network":{ "src_ip_addr" : "192.168.1.1", "src_udp_port" : "2000", "dst_ip_addr" : "192.168.1.2", "dst_udp_port" : "2001", "dst_mac" : "00:01:02:03:04:05", "vid_clk_rate" : "12345678", "vlan_tag" : "1234", "ssrc" : "12345", "tx_pkt_cnt" : "12345" }, "format_code_valid" : "1", "format_code_t_scan" : "1", "format_code_p_scan" : "1", "format_code_mode" : "1", "format_code_format" : "1", "format_code_rate" : "1", "format_code_sampling" : "1" }</pre>	version: current flow version. label: User defined label (not available yet) id: flow's UUID. source_id: associated source UUID. type: Flow type. 1 = 2022-6 decapsulator. 2 = 2022-6 encapsulator. name: Flow name. network: IP network related configuration and statuses. src_ip_addr: Source IP address. src_udp_port: Source UDP port. dst_ip_addr: Destination IP address. dst_udp_port: Destination UDP port. dst_mac: Destination MAC address vid_clk_rate: Video clock rate. vlan_tag: Current VLAN tag. ssrc: TBD tx_pkt_cnt: Transmitted packet counter (max = 2^32-1).

3 HTTP response descriptions

Code	Name	Description
200	OK 200	The request has succeeded. The information returned with the response is dependent of the method used.
400	400 Bad request	The request could not be understood due to malformed syntax.
404	404 Not found	Not matching request URI.

4 Annexe 1

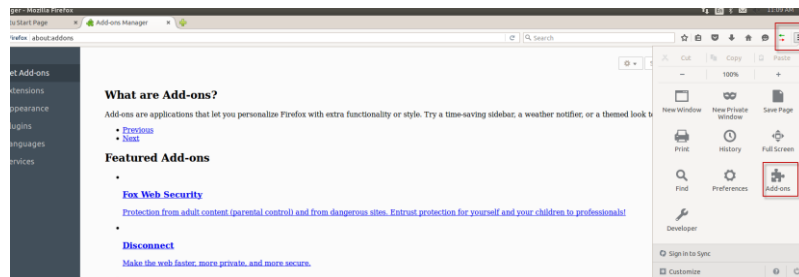
JSON schema and example related to this document can be found on folder that comes with this document.

5 Annexe 2

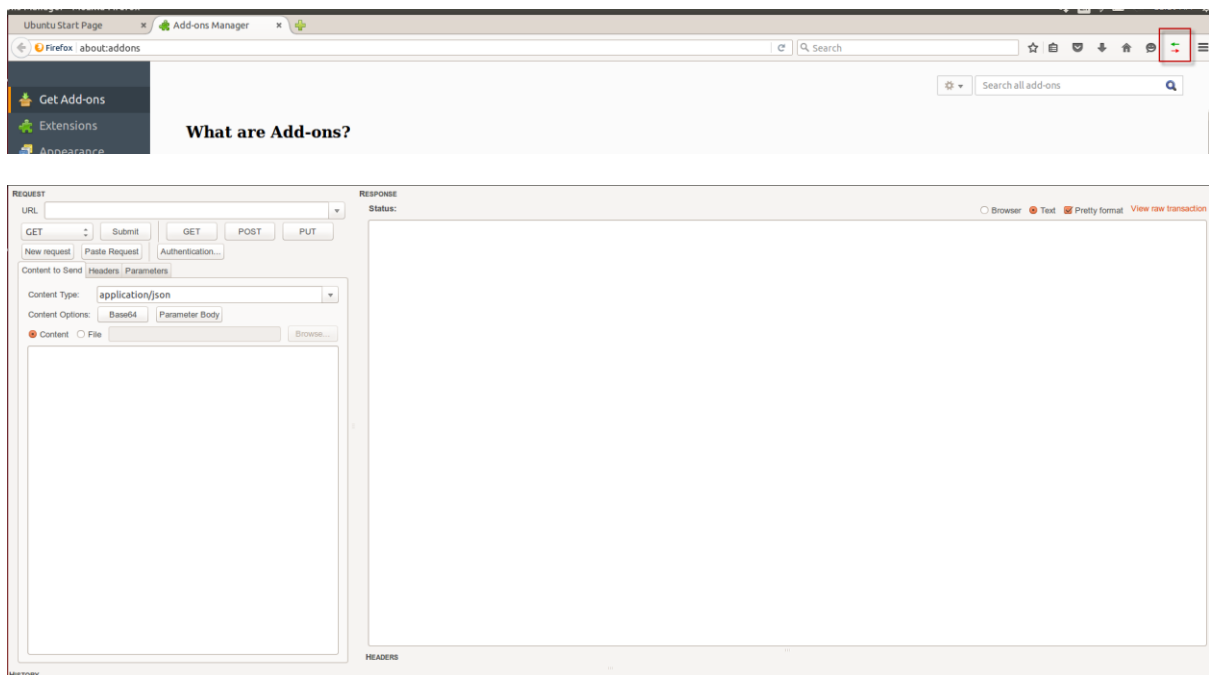
API Example with Firefox HTTP Requester.

1 – Download Firefox

2 – Download HttpRequester Addon

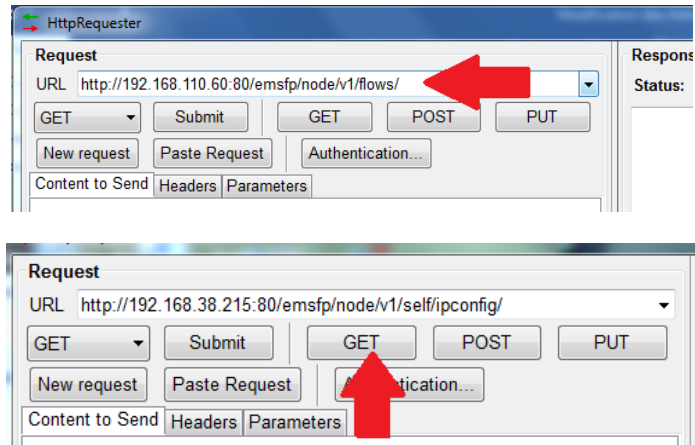


3 – Start HttpRequester Addon

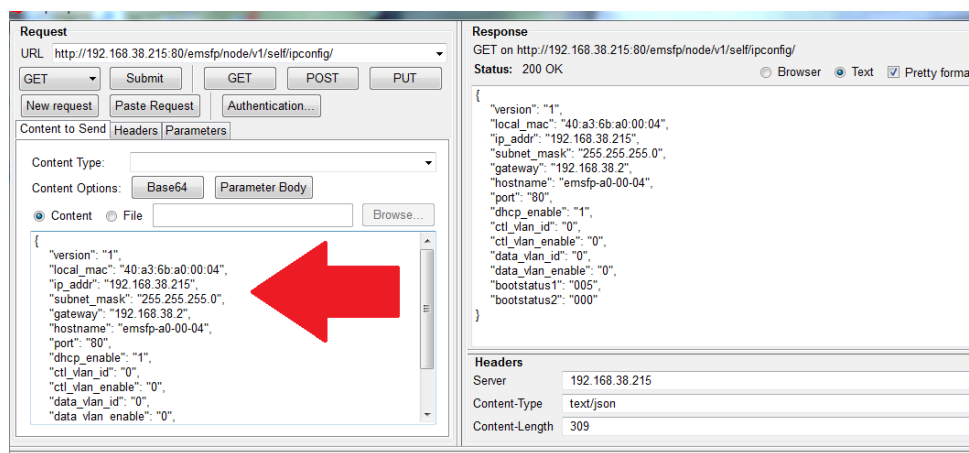


4 – Changing Video flows information enter the device flow url in the URL field of HTTP request example "http://<deviceip>:<port>/emsfp/node/v1/flows/" then execute the GET.

emSFP API Description

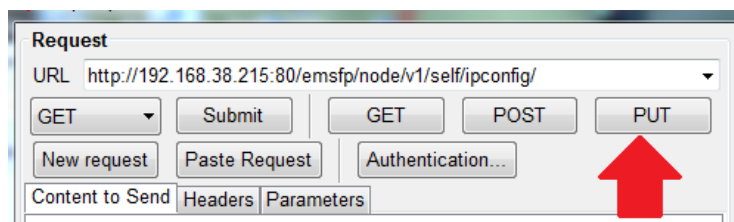


The device should then return the flow in Response field.



Now you can just copy the response string and paste it into the “Content to send” Tab and edit the field you want to change.

Once the change made you can execute a PUT command.



The same process can be done for the device local IP configuration under “http://<deviceip>:<port>/emsfp/node/v1/self/ipconfig”