



GFN SDN Controller User Manual

TAP Service

Release 2.4.0

Gateflow.net and gateflow.net logo are trademarks of 3A alliance, LLC. All other trademarks may be property of their respective owners.

3A alliance, LLC is an enterprise registered and doing business under the law of Republic of Latvia, reg.#40103868526, address: 10-40 Lielezeres street, Riga, Latvia, LV-1007, <http://3a-alliance.com>

3A alliance, LLC assumes no responsibility for any inaccuracies in this document. The content of this document is subject to change without any notice.

The software described in this document is subject to be delivered "AS IS" without warranty of any kind.

Table of Contents

1. Acronyms.....	3
2. TAP Service Review.....	6
TAP Rule.....	6
TAP Rule Match.....	6
TAP Rule Actions.....	8
TAP Rule Statistics.....	9
TAP Rule Template.....	9
3. Prerequisites.....	10
4. Operation of TAP Service.....	11
CLI.....	11
REST API JSON Descriptors' format.....	11
REST API Service URLs.....	15
Web UI.....	17

1. Acronyms

- AD – Administrative Domain
- AI – Artificial intelligence
- ASIC – Application Specific Integrated Circuit
- BGP – Border Gateway Protocol
- BNG – Border Network Gateway
- BRAS – Broadband Remote Access Server
- BSS – Business Support System
- CBS – Committed Bust Size
- CEN -Carrier Ethernet Network
- CG-NAT – Carrier Grade Network Address Translation
- CIR – Committed Information Rate
- CLI - Command Line Interface
- CPU – Central Processing Unit
- CRM – Customer Relationship Management
- CRUD - Create, Read, Update, Delete
- DB - Database
- DC – Datacenter
- DPI – Deep Packet Inspection
- DPID – Data Path Identifier
- E2E – End-to-End (services)
- E-Access - OVC-based service with at least one UNI OVC End Point and one ENNI End Point
- EBS – Excess Burst Size
- EIR - Excess Information Rate
- E-LAN – multipoint-to-multipoint EVC
- E-Line – point-to-point EVC accordingly to MEF
- EMS – Element Managements System
- ENNI – External Network-to-Network Interface

EP-LAN – Ethernet Private LAN

EPL – Ethernet Private Line

E-Transit - OVC-based Carrier Ethernet service in which all OVC End Points are at ENNIs

E-Tree – point-to-multipoint EVC

ETSI – European Telecommunications Standards Institute

EVC – Ethernet Virtual Circuit

IGMP – Internet Group Management Protocol

LAN – Local Area Network

LPM - Longest Prefix Match

MEF – Metro Ethernet Forum

MPLS – Multiprotocol Label Switching

NAT – Network Address Translation

NBI – North Bound Interface

NE – Network Element

NPU – Network Processing Unit

NVF – Network Functions Virtualization

NFVI – Network Functions Virtualization Infrastructure

OAM – Operations, Administration and Management

OF – OpenFlow protocol

OF-DPA – OpenFlow Data Plane Abstraction

ONF – Open Networking Foundation

OSS – Operation Support System

OVC – Operator Virtual Connection

OVS – Open vSwitch

PNE – Physical Network element

PNF – Physical Network Element

PoP – Point of Presence, see also Datacenter

QinQ – IEEE 802.1ad standard

QoS – Quality of Service

RFC – Request for Comments
SBI – South Bound Interface
SDN – Software Defined Network
SLA – Service Level Agreement
SQL – Structured Query Language
SR – Segment Routing
SRAM – Static Random Access Memory
TAP – Terminal Access Point
TCAM – Ternary Content Addressable Memory
TE – Traffic Engineering
T/T – Troubleshooting
TTP – Table Type Pattern
UDF - User-Defined Field
UNI – User Network Interface
VLAN – Virtual Local Area Network
VIM – Virtual Infrastructure Manager
VM – Virtual Machine
VNE – Virtual Network Element
VNF – Virtual Network Function
VNFD – Virtual Network Function Descriptor
VNFM – Virtual Network Function Manager
VPLS – Virtual Private Area Network
WAN – Wide Area Network
ZTP – Zero Touch Provisioning

2. TAP Service Review

TAP Service is a policy based network packet filtering and distribution engine. TAP filters and distributes a mirrored copy of network traffic to consumers - performance monitoring applications thus providing End-to-End network visibility for networking operations team.

TAP Rule

All TAP rules are organized in a list of rules. Rules are implemented concurrently in device TCAM memory. For selecting a best suiting one TAP implements an LPM (Longest Prefix Match) algorithm.



Note that if two or more rules suit to some packet on the same priority level (prefix length) then TAP will choose that one which is located closer to the beginning of the rules' list.

Each rule consists of three sections:

- Policy ID
- Match section
- Actions section
- Statistics (read only)

Rule ID is a unique string Identifier of TAP policy. Other sections are described in more details in the corresponding sections of this manual.

TAP Rule Match

A match section of TAP rule contains bit-masked fields values of the following network packet headers:

- L1 header:
 - OF Switch DPID
 - Ingress port
- L2 header:

- Source MAC address and optional bit-mask
- Destination MAC address and optional bit-mask
- Ethertype
- VLAN VID
- VLAN PCP
- L3 - IPv4, ARP, IPv6:
 - Source IPv4 address and optional bit-mask
 - Destination IPv4 address and optional bit-mask
 - IPv4 Protocol
 - IPv4 DSCP
 - IPv4 ECN
 - Source IPv6 address and optional bit-mask
 - Destination IPv6 address and optional bit-mask
 - IPv6 NextHeader
 - IPv6 ToS
 - IPv6 ECN
 - ARP SPA
 - ARP TPA
 - ARP SHA
 - ARP THA
 - ARP OP
- L4 - TCP, UDP, ICMP, ICMPv6
 - TCP source port
 - TCP destination port
 - UDP source port
 - UDP destination port
 - ICMP type
 - ICMP code
 - ICMPv6 type

- ICMPv6 code
- UDF (User-Defined) fields
 - 4 fields of 4-byte value/mask in HEX format with a pre-set shift in bytes shift from the beginning of L2, L3 or L4 header; UDF fields are useful for parsing and checking of non-ISO headers' fields (e.g. MPLS, VxLAN etc.)

If any field value is not used in the exact rule then its value has to be set to 0 with a mask 0. Web UI for fields without value automatically sets 0/0 value during REST API call. So any unused fields in Web UI can be left unfilled.



Note that due to hardware limitations of Broadcom Strata XGS family of ASICs a concurrent use of UDF fields and L2-L3-L4 fields is not possible in the same TAP rule.

TAP Rule Actions

TAP rule actions can include the following actions:

- Set output port
- Push VLAN
- Pop VLAN
- Set VLAN
- Set VLAN PCP
- Set egress queue
- Set source MAC address
- Set destination MAC address
- Set source IP address
- Set destination IP address
- Set TCP source port
- Set TCP destination port
- Set UDP source port
- Set UDP destination port

TAP Rule Statistics

TAP rule statistics section provides 64-bit counters' values of amount of packets processed by this very policy:

- Packets
- Bytes

Statistics counters are a good tool for troubleshooting a TAP policy.




Note that due to hardware limitations of some ASICs statistics counters can be reset to zero not on all hardware switch platforms.

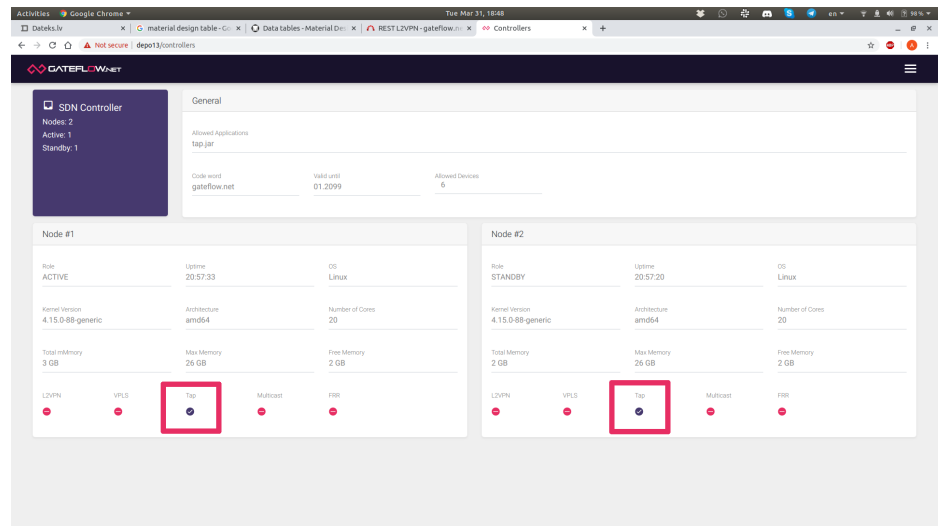
TAP Rule Template

TAP Rule Template is very much like TAP Rule in its structure. But unlike TAP Rule it is not pushed into the switch and is used for fast creating of generic TAP Rules by means of web interface.

3. Prerequisites

To use TAP service tap application has to be running on SDN Controller. Check TAP application is running can be done in two ways:

- In Command Line Interface of the controller with a command *app show*, TAP application status has to be *running*.
- In Web UI Menu->Inventory->Controller page TAP application should be marked with icon 



4. Operation of TAP Service

There are three generic ways to operate by policies in TAP service:

CLI

REST API

Web UI

These interfaces are described in details in sections below.

CLI

CLI provides a limited set of commands of TAP policy management. They are targeted for viewing and deleting of existing policies only:

- tap show
- tap delete

For creating and updating of TAP policies please use either Web UI or REST API call.

REST API JSON Descriptors' format

REST API of TAP service implements a standard CRUD (Create, Read, Update, Delete) data manipulation paradigm. Any REST API call operates with data in JSON format. Below is an example of JSON file for REST API call for TAP rule:

```
{
  "id": "tap_example",
  "description": "Example of TAP rule",
  "ingress_dpid": "00:00:00:00:00:00:02",
  "egress_dpid": "00:00:00:00:00:00:03",
  "priority": 300,
  "cir": 10000,
  "match_fields": {
```

```
"in_port": 10,  
"eth_dst": "00:00:00:00:00:01",  
"eth_src": "00:00:00:00:00:02",  
"eth_type": 2048,  
"vlan_vid": 2000,  
"vlan_pcp": 7,  
"ip_dscp": 48,  
"ip_ecn": 0,  
"ip_proto": 4,  
"ipv4_src": "127.0.0.2/24",  
"ipv4_dst": "128.0.0.2/24",  
"tcp_src": 65535,  
"tcp_dst": 65535,  
"udp_src": 65535,  
"udp_dst": 65535,  
"sctp_src": 65535,  
"sctp_dst": 65535,  
"icmpv4_type": 0,  
"icmpv4_code": 255,  
"arp_op": 65535,  
"arp_spa": "127.0.0.1/24",  
"arp_tpa": "127.0.0.2/24",  
"arp_sha": "00:00:00:00:00:03",  
"arp_tha": "00:00:00:00:00:04",  
"ipv6_src": "2001:0db8:85a3:0000:0000:8a2e:0370:7334",  
"ipv6_dst": "2001:0db8:85a3:0000:0000:8a2e:0370:7334",  
"ipv6_flabel": 0,  
"icmpv6_type": 0,  
"icmpv6_code": 0,  
"ipv6_nd_target": "2001:0db8:85a3:0000:0000:8a2e:0370:7334",  
"ipv6_nd_sll": "00:00:00:00:00:03",  
"ipv6_nd_tll": "00:00:00:00:00:03",  
"mpls_label": 0,  
"mpls_tc": 0,  
"mpls_bos": true,  
"udf0": "1000/65535",  
"udf1": "1000/65535",  
"udf2": "1000/65535",  
"udf3": "1000/65535"
```

```
    },  
    "action_fields": {  
      "out_ports": [3, 4],  
      "push_vlan": true,  
      "pop_vlan": true,  
      "set_vlan_vid": 200,  
      "set_vlan_pcp": 10,  
      "eth_dst": "00:00:00:00:00:01",  
      "eth_src": "00:00:00:00:00:02",  
      "eth_type": 2048,  
      "tcp_src": 65535,  
      "tcp_dst": 65535,  
      "udp_src": 65535,  
      "udp_dst": 65535,  
      "set_queue": 9803,  
      "ipv4_src": "127.0.0.2/24",  
      "ipv4_dst": "128.0.0.2/24"  
    }  
  }  
}
```



Only necessary packet header fields have to be included into JSON TAP Rule descriptor. All unnecessary fields can be omitted.

This is a basic template of JSON file for creating of TAP template:

```
{  
  "id": "tap_example",  
  "description": "Example of TAP rule template",  
  "priority": 300,  
  "cir": 10000,  
  "match_fields": {  
    "eth_dst": "00:00:00:00:00:01",  
    "eth_src": "00:00:00:00:00:02",  
    "eth_type": 2048,  
    "vlan_vid": 2000,  
  }  
}
```

```
"vlan_pcp": 7,  
"ip_dscp": 48,  
"ip_ecn": 0,  
"ip_proto": 4,  
"ipv4_src": "127.0.0.2/24",  
"ipv4_dst": "128.0.0.2/24",  
"tcp_src": 65535,  
"tcp_dst": 65535,  
"udp_src": 65535,  
"udp_dst": 65535,  
"sctp_src": 65535,  
"sctp_dst": 65535,  
"icmpv4_type": 0,  
"icmpv4_code": 255,  
"arp_op": 65535,  
"arp_spa": "127.0.0.1/24",  
"arp_tpa": "127.0.0.2/24",  
"arp_sha": "00:00:00:00:00:03",  
"arp_tha": "00:00:00:00:00:04",  
"ipv6_src": "2001:0db8:85a3:0000:0000:8a2e:0370:7334",  
"ipv6_dst": "2001:0db8:85a3:0000:0000:8a2e:0370:7334",  
"ipv6_flabel": 0,  
"icmpv6_type": 0,  
"icmpv6_code": 0,  
"ipv6_nd_target": "2001:0db8:85a3:0000:0000:8a2e:0370:7334",  
"ipv6_nd_sll": "00:00:00:00:00:03",  
"ipv6_nd_tll": "00:00:00:00:00:03",  
"mpls_label": 0,  
"mpls_tc": 0,  
"mpls_bos": true,  
"udf0": "1000/65535",  
"udf1": "1000/65535",  
"udf2": "1000/65535",  
"udf3": "1000/65535",  
},  
"action_fields": {
```

```
"out_port": 2,  
"push_vlan": true,  
"pop_vlan": true,  
"set_vlan_vid": 200,  
"set_vlan_pcp": 10,  
"eth_dst": "00:00:00:00:00:01",  
"eth_src": "00:00:00:00:00:02",  
"eth_type": 2048,  
"tcp_src": 65535,  
"tcp_dst": 65535,  
"udp_src": 65535,  
"udp_dst": 65535,  
"set_queue": 9803,  
"ipv4_src": "127.0.0.2/24",  
"ipv4_dst": "128.0.0.2/24"  
}  
}
```



Like for the TAP Rule JSON descriptor for TAP Template has to include only necessary fields while all unnecessary fields have to be omitted.

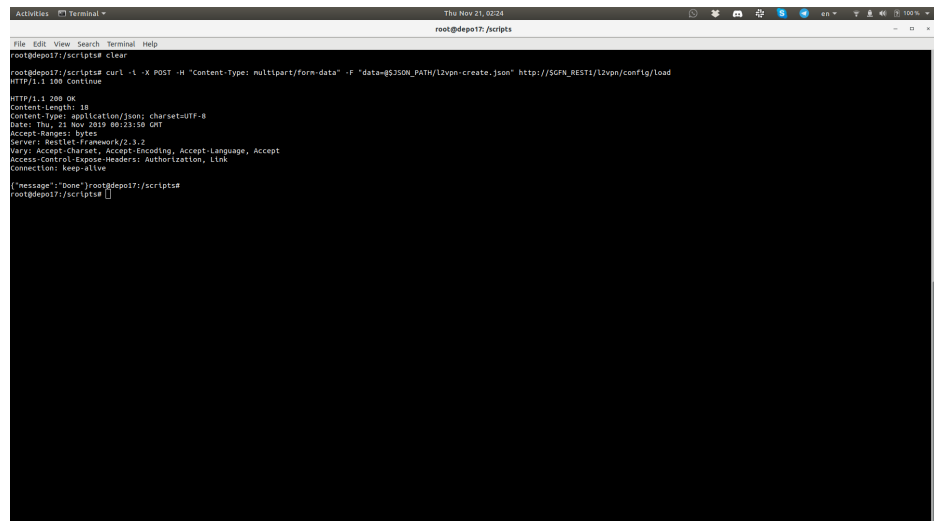
REST API Service URLs

There are several REST API URLs for manipulating of TAP service application:

- Create a single rule - <http://sdn-node:8084/tap/rules>
- Get all TAP rules - <http://sdn-node:8084/tap/rules>
- Get TAP rule by id - <http://sdn-node:8084/tap/rules/{id}>
- Get TAP rule's path in monitoring fabric - <http://sdn-node:8084/tap/rules/{id}/path>
- Update TAP rule by id - <http://sdn-node:8084/tap/rules/{id}>
- Delete TAP rule by id - <http://sdn-node:8084/tap/rules/{id}>

- Create a batch of TAP rules -
`http://sdn-node:8084/tap/rulesbatch`
- Create TAP template - `http://sdn-node:8084/tap/templates`
- Get all TAP template - `http://sdn-node:8084/tap/templates`
- Get TAP template by id -
`http://sdn-node:8084/tap/templates/{id}`
- Update TAP template - `http://sdn-node:8084/tap/templates/{id}`
- Delete TAP template -
`http://sdn-node:8084/tap/templates/{id}`
- Get all available match fields for TAP rule or TAP template -
`http://sdn-node:8084/tap?matchfields`
- Get all available actions fields for TAP rule or TAP template -
`http://sdn-node:8084/tap?actionfields`

For invoking of REST API call from Linux command line a curl utility can be used as show below:



```
root@depo17:/scripts# curl -i -X POST -H "Content-Type: multipart/form-data" -F "data=@$SSON_PATH/l2vpn:create.json" http://$GFN_RESTI/l2vpn/config/load
HTTP/1.1 200 OK
Content-Length: 18
Content-Type: application/json; charset=utf-8
Date: Thu, 21 Nov 2019 08:23:50 GMT
Server: Restlet-Framework/2.3.2
Access-Control-Allow-Origin: *
Access-Control-Expose-Headers: Authorization, Link
Connection: keep-alive

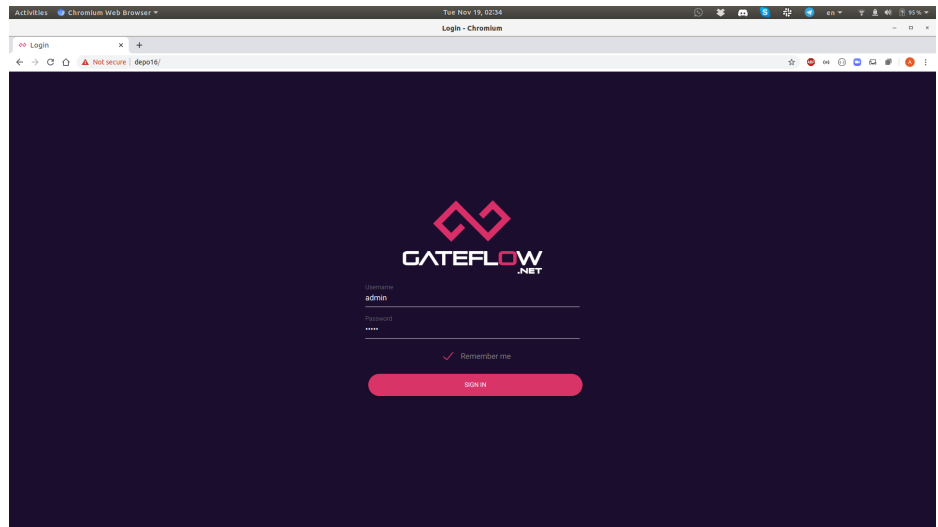
{"message": "Done"}root@depo17:/scripts#
```



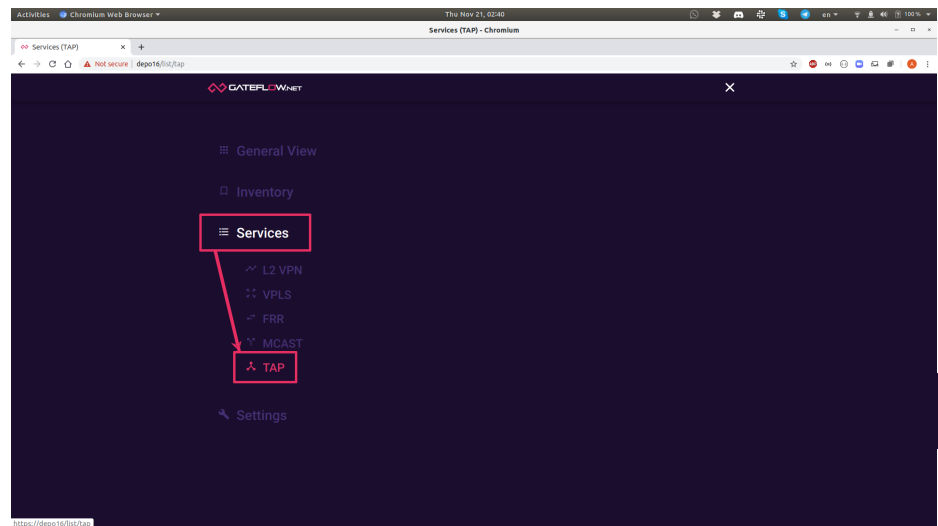
Please note that only actions fields of TAP Rule or TAP Template can be updated. If you need to update match fields or general fields then you have to delete TAP Rule/Template first and then recreate it with new values.

Web UI

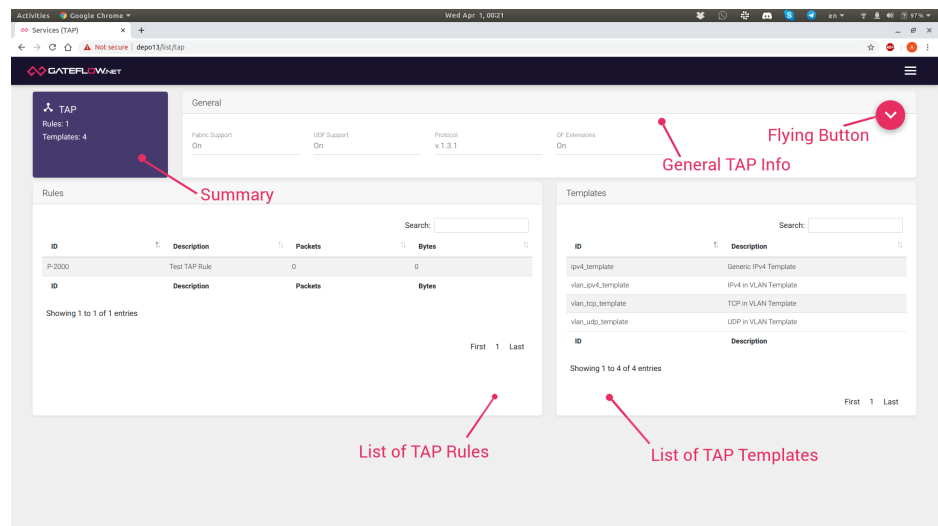
For access of TAP management graphic interface via web browser one has to logon to GFN SDN Controller Web UI first as show in the example below:



After logon one has to open a full screen menu using icon the top bar and choose drop-down Menu->Services->TAP section as show below:



A TAP list screen will open:



This screen contains five major elements:

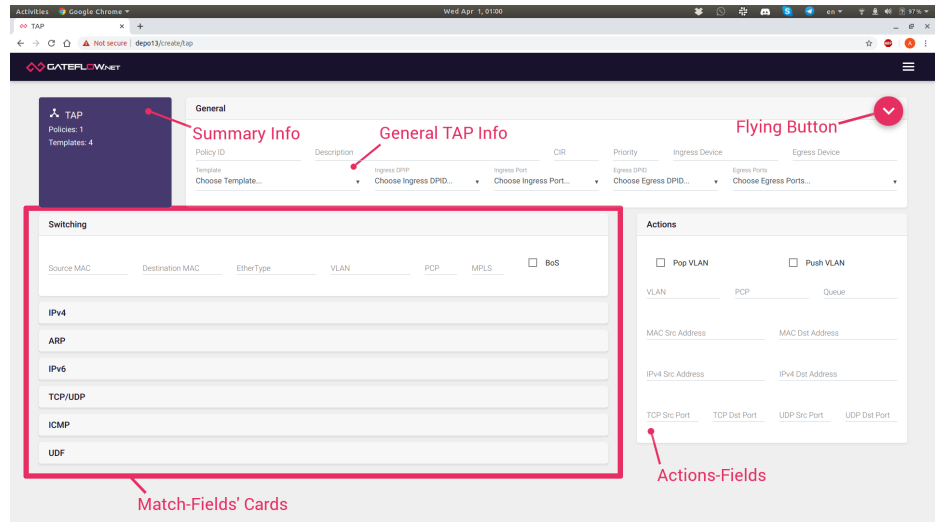
- Summary info card
- General information about TAP service application
- List of TAP Rules
- List of TAP Templates
- Flying button

Summary Info card contains information about total number of TAP Rules and Templates configured. General information card displays major TAP service application capabilities.

List of TAP Rules and List of TAP Templates cards display lists of rules and templates accordingly. They will split a long lists on many tabs with 8 items per tab. Besides this these lists have per column ordering capability and search by any column feature.

Pressing Flying button will invoke context menu which has "Add Rule" button. Pressing this button one can invoke a TAP form screen in create mode. Also TAP form screen in view mode can be invoked by pressing corresponding TAP Rule inside List of TAP Rules List card.

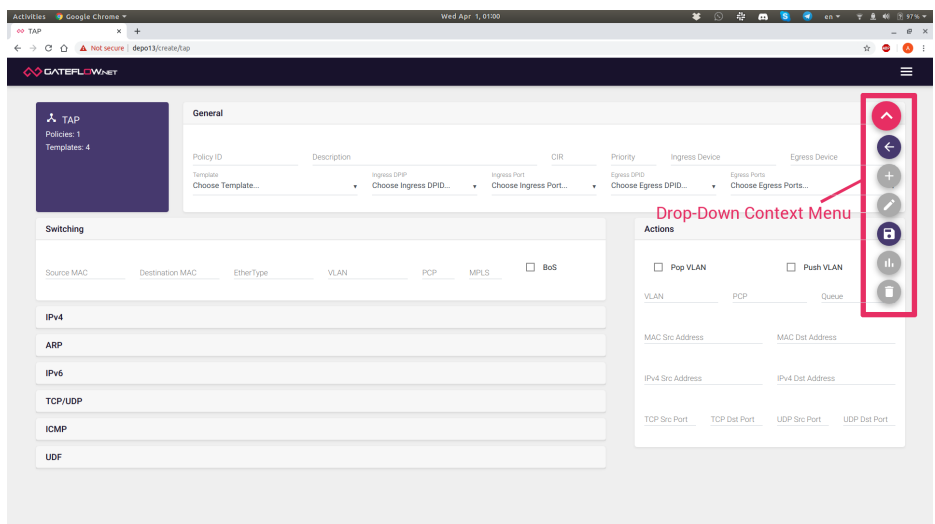
Entering TAP form screen you will see the following:



There are following elements on this screen:

- TAP Service summary information
- General information about TAP Rule
- Match-Fields accordion cards
- Actions-Fields card
- Flying button

Pressing Flying Button will invoke a drop-down context menu.in top right side of the screen:



Drop-down menu contains following buttons:

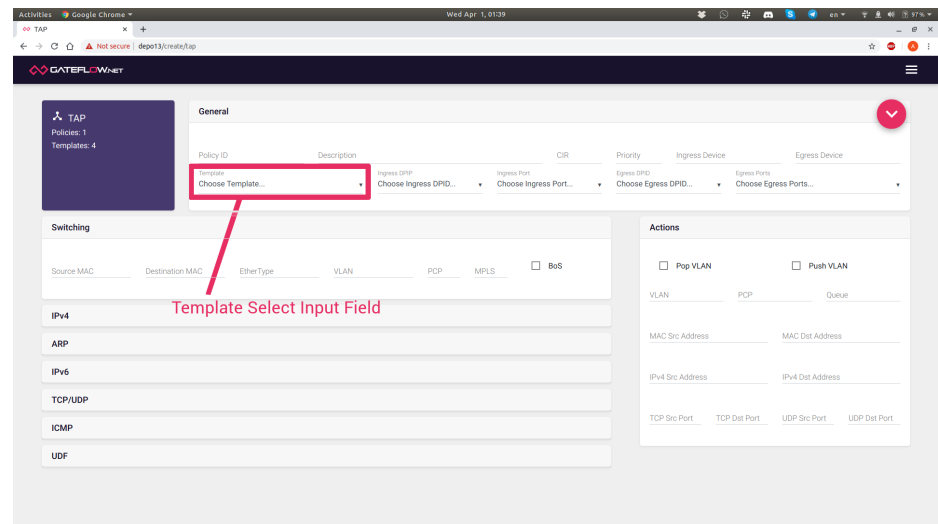
- Back to TAP list screen
- Add TAP Rule

- Edit TAP Rule
- Save TAP Rule
- Open Grafana
- Delete TAP Rule

Availability of buttons above is context sensitive and depends on TAP form screen mode. Active buttons look blue while inactive ones are gray. The only exclusion is Delete button which is pink when active.

When entering TAP form screen in create mode all input fields in general info, match-fields and actions-fields cards will be empty and available for entering parameters. Only Back to list and Save buttons will be active in this mode. After entering all necessary TAP Rule parameters the button Save has to be pressed. Than TAP form will save the new TAP Rule and will enter in view mode.

To improve productivity when entering a lot of generic TAP Rules you can you TAP Template choosing a necessary one from the select list:



If TAP form screen is invoked in view mode then some or all general, match-fields and actions-fields inputs will display parameters of selected TAP Rule. They will be grayed and will not be available for editing. In view mode only Back to list , Edit and Grafana buttons will be active.

Pressing Grafana button you can open Grafana dashboard displaying load charts of selected TAP Rule.



Please note that Grafana is a separate open source software product. Grafana has to be deployed and configured separately to become available from GFN SDN Controller Web UI. Deployment and configuration of Grafana is out of the scope of this manual.

For entering edit mode the button Edit has to be pressed. In edit mode some fields of general card and all fields of actions-fields card will be open for editing values. To save edited TAP Rule please press the button Save. If you would like to cancel editing then press either the button Add or the button Back to list.



Please note that in current version of TAP service application only Description fields of general card and actions-fields can be edited. If you would like to edit match-fields then you have to delete selected TAP Rule and then recreate it with new parameters.

Besides the above selected TAP Rule can be deleted in edit mode by pressing the button Delete. After deleting Web UI will enter TAP list screen.



Please note that not all fields of the TAP policy form can be used simultaneously. E.g. when filling IPv4 fields one cannot use IPv6 fields, concurrent Push VLAN and Pop VLAN in actions make no sense etc. Using fields of some upper layer protocol may require mandatory filling of some fields in lower layer protocol. E.g. while using IPv4 fields one has to set a value 0x0800 to the Ethertype field of Ethernet header etc. These constraints are caused by switch ASIC hardware limitations. For more information about constraints please refer to GFN SDN Controller Admin Manual.
