Flowmon
Networks

FlowMon DDoS Defender 4.05.02

RESTful API Reference Guide

January 31, 2020

Contents

1 Introduction

This document describes usage of REST API for Flowmon DDoS Defender plugin which is available in version 4.02.00 and newer.

The next chapter describes authorization process for REST API access. Following chapters contains description and usage of data types and methods.

In this documentation is used CodeList, which is more specific for certain methods. Its basic form is as follows:

CodeList

```
CodeList = {
    id: Integer|String("Item id"),
    name: String("Item name")
}
```

The items with null values are also shown in this document (during element creating and editing). These values do not have to be sent, they are displayed only as a reference.

2 Authorization

The API for interception management uses the Oauth 2.0 authorization. An application, which want to use API methods, has to request for granting access token using following HTTPS request (change only the values of the username and password properties).

2.1 Obtaining authentication token

New access token request

```
POST /resources/oauth/token
FORM-DATA grant_type=password
          client_id=invea-tech
          username=<user_login>
          password=<user_password>
```

The server will verify username and password and returns following response:

```
{
    "access_token": "AnlSCIWYbchsCc5sdc5ac4caca8a2",
    "token_type": "bearer",
    "expires_in": 3600,
    "refresh_token": "DS6SA512ADCVa51adc54VDS51VD5"
}
```

Now the application has to add this access token to every call of API method. The token has to be inserted in the **Authorization** header field of HTTPS requests and value of this field has to be "**<token_type> <access_token>**" from **/resources/oauth/token** response. So for the example the **Authentication** header value will look like "**bearer AnlSCIWYbchsCc5sdc5ac4caca8a2**". The access token is valid for **<expires_in>** seconds. Then the application can request new access token using previous HTTPS request or it can use **<refresh_token>** field to acquire new token without supplying of username and password.

Example of refresh token request:

```
POST /resources/oauth/token
FORM-DATA grant_type=refresh_token
          refresh_token=DS6SA512ADCVa51adc54VDS51VD5
          client_id=invea-tech
```

2.2 Getting Rest API information

There is new posibility of getting REST API available methods since Invea OS version 7.2.0.

2.2.1 Getting list of avalable methods

- Returns list of avalable methods

Example Usage

```
GET /rest/methods

RETURNS:
[
    {
        "url": "/rest/fcc/device",
        "method": "GET"
    },
    {
        "url": "/rest/fcc/device/info",
        "method": "GET"
    },
    {
        "url": "/rest/fcc/device/cpu",
        "method": "GET"
    },
    {
        "url": "/rest/fcc/device/ram",
        "method": "GET"
    },
    ...
]
```

2.2.2 Getting current version of FlowMon

- Returns Current version of FlowMon device.
- @returns 200 { "version": String("Version") }

Example Usage

```
GET /rest/version

RETURNS:
[
    {
        "version": "7.01.00"
    }
]
```

3 Attack

3.1 Data Types

Attacks Search

```
AttacksSearch = {
   ?start: {
      ?from: String("Timestamp"),
      ?to: String("Timestamp"),
   },

   ?end: {
      ?from: String("Timestamp"),
      ?to: String("Timestamp"),
   },

   ?segment: [Integer(ProtectedSegment.id)],
   ?status: [String(Status.id)],
   ?actions: [String("Action label - fulltext search")],
   ?activeMittigation: Integer(Boolean)
   ?limit: Integer("Default 500")
   ?order: String("asc or desc")
}
```

Attack

```
Attack = {
   id: Integer,
   segment: {
      id: Integer,
      name: String
   },

   start: String(Timestamp),
   end: String(Timestamp),

   status: AttackStatus,
   activeMittigation: Integer(Boolean),

   actions: ActionStatus
}
```

Attack Status

```
AttackStatus = {
   id: String,
```

```
   name: String
} << [
   {
      id: "STARTING",
      name: "New about to be automatically mitigated attack"
   },
   {
      id: "ACTIVE",
      name: "Ongoing attack"
   },
   {
      id: "NOT ACTIVE",
      name: "Inactive attack"
   },
   {
      id: "ENDED",
      name: "Finished attack"
   }
]


DetailedAttack = {
   attackTraffic: AttackSummary("Summary of traffic during attack"),
   normalTraffic: TrafficSummary("Summary of normal traffic"),
   chartData: [ChartSerie]
} extends Attack

ActionStatus = {
    action: String,
    performed: String or Null,
    date: String(Y-m-d H:i:s),
    attackDetail: Integer or null,
    info: [DetectionMethodOutput]
}

DetectionMethodOutput = {
    method: Methods,
    flags: [Flag]
}

Flag = {
    name: Flags,
    isSuspected: Boolean
}

Methods extends CodeList << [
  {id: in_adaptive_baseline_rate, name: "Adaptive threshold"},
  {id: in_out_rate, name: "Incoming / Outgoing Ratio"},
```

```
  {id: min_rate, name: "Minimal traffic"},
  {id: in_baseline_rate, name: "Manual threshold"}
]

Flags extends CodeList << [
  {id: in_adaptive_baseline_rate.tcp, name: "TCP"},
  {id: in_adaptive_baseline_rate.tcp_r, name: "TCP Rst"},
  {id: in_adaptive_baseline_rate.tcp_s, name: "TCP Syn"},
  {id: in_adaptive_baseline_rate.tcp_sa, name: "TCP Syn Ack"},
  {id: in_adaptive_baseline_rate.tcp_af, name: "TCP Ask Fin"},
  {id: in_adaptive_baseline_rate.udp, name: "UDP"},
  {id: in_adaptive_baseline_rate.icmp, name: "ICMP"},
  {id: in_out_rate.in_out, name: "Incoming / Outgoing Ratio"},
  {id: min_rate.in, name: "Packed per second"},
  {id: min_rate.in_bps, name: "Bits per second"},
  {id: in_baseline_rate.in, name: "Manual threshold"}
]
```

Action Status

```
ActionStatus = {
    action: String,
    performed: String or Null,
    date: String(Y-m-d H:i:s),
    attackDetail: Integer or null,
    info: [DetectionMethodOutput]
}

DetectionMethodOutput = {
    method: Methods,
    flags: [Flag]
}

Flag = {
    name: Flags,
    isSuspected: Boolean
}

Methods extends CodeList << [
  {id: in_adaptive_baseline_rate, name: "Adaptive threshold"},
  {id: in_out_rate, name: "Incoming / Outgoing Ratio"},
  {id: min_rate, name: "Minimal traffic"},
  {id: in_baseline_rate, name: "Manual threshold"}
]

Flags extends CodeList << [
  {id: in_adaptive_baseline_rate.tcp, name: "TCP"},
  {id: in_adaptive_baseline_rate.tcp_r, name: "TCP Rst"},
```

```
  {id: in_adaptive_baseline_rate.tcp_s, name: "TCP Syn"},
  {id: in_adaptive_baseline_rate.tcp_sa, name: "TCP Syn Ack"},
  {id: in_adaptive_baseline_rate.tcp_af, name: "TCP Ask Fin"},
  {id: in_adaptive_baseline_rate.udp, name: "UDP"},
  {id: in_adaptive_baseline_rate.icmp, name: "ICMP"},
  {id: in_out_rate.in_out, name: "Incoming / Outgoing Ratio"},
  {id: min_rate.in, name: "Packed per second"},
  {id: min_rate.in_bps, name: "Bits per second"},
  {id: in_baseline_rate.in, name: "Manual threshold"}
]
```

3.2 Endpoints

3.2.1 Methods

Get all detection methods.

GET /rest/iad/attacks/methods

- Return CodeList Methods
- @return 200 CodeList Methods (3)

```
GET /rest/iad/attacks/methods

Return 200:
[
    {
        "id": "in_adaptive_baseline_rate",
        "name": "Adaptive threshold"
    },
    {
        "id": "in_out_rate",
        "name": "Incoming / Outgoing Ratio"
    },
    {
        "id": "min_rate",
        "name": "Minimal traffic"
    },
    {
```

```
        "id": "in_baseline_rate",
        "name": "Manual threshold"
    }
]
```

3.2.2 Flags

Get all detection flags.

GET /rest/iad/attacks/flags

- Return CodeList Flags
- @return 200 CodeList Flags (3)

```
GET /rest/iad/attacks/flags

Return 200:
[
    {
        "id": "in_adaptive_baseline_rate.tcp",
        "name": "TCP"
    },
    {
        "id": "in_adaptive_baseline_rate.tcp_r",
        "name": "TCP Rst"
    },
    {
        "id": "in_adaptive_baseline_rate.tcp_s",
        "name": "TCP Syn"
    },
    {
        "id": "in_adaptive_baseline_rate.tcp_sa",
        "name": "TCP Syn Ack"
    },
    {
        "id": "in_adaptive_baseline_rate.tcp_af",
        "name": "TCP Ack Fin"
    },
    {
```

```
        "id": "in_adaptive_baseline_rate.udp",
        "name": "UDP"
    },
    {
        "id": "in_adaptive_baseline_rate.icmp",
        "name": "ICMP"
    },
    {
        "id": "in_out_rate.in_out",
        "name": "Incoming / Outgoing Ratio"
    },
    {
        "id": "min_rate.in",
        "name": "Packed per second"
    },
    {
        "id": "min_rate.in_bps",
        "name": "Bits per second"
    },
    {
        "id": "in_baseline_rate.in",
        "name": "Manual threshold"
    }
]
```

3.2.3 Attacks

Method for obtaining all Attacks.

GET /rest/iad/attacks/

- Return Attacks
- @return 200 Attack (3)

```
GET /rest/iad/attacks/

Return 200:
[
    {
```

```
"id": 380,
"segment": {
    "id": 22,
    "name": "Update by API"
},
"suspect": false,
"falsePositive": true,
"userComment": "",
"start": "2017-12-01 14:27:30",
"end": "2017-12-01 14:54:55",
"status": {
    "id": "ENDED",
    "name": "Finished attack"
},
"activeMitigation": 0,
"actions": [
    {
        "action": "Detected",
        "perfomed": null,
        "date": "2017-12-01 14:28:33",
        "actionId": null,
        "info": [
            {
                "method": {
                    "id": "in_adaptive_baseline_rate",
                    "name": "Adaptive threshold"
                },
                "flags": [
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.icmp",
                            "name": "ICMP"
                        },
                        "isSuspected": false
                    }
                ]
            }
        ]
    },
    {
        "action": "Detected",
        "perfomed": null,
        "date": "2017-12-01 14:49:33",
        "actionId": null,
        "info": [
            {
                "method": {
                    "id": "in_adaptive_baseline_rate",
```

```
                    "name": "Adaptive threshold"
                },
                "flags": [
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.icmp",
                            "name": "ICMP"
                        },
                        "isSuspected": false
                    },
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.tcp_r",
                            "name": "TCP Rst"
                        },
                        "isSuspected": false
                    },
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.tcp_s",
                            "name": "TCP Syn"
                        },
                        "isSuspected": false
                    },
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.tcp_af",
                            "name": "TCP Ack Fin"
                        },
                        "isSuspected": false
                    },
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.tcp_sa",
                            "name": "TCP Syn Ack"
                        },
                        "isSuspected": false
                    }
                ]
            }
        ]
    },
    {
        "action": "Detected",
        "perfomed": null,
        "date": "2017-12-01 14:54:33",
        "actionId": null,
        "info": [
```

```
                {
                    "method": {
                        "id": "in_adaptive_baseline_rate",
                        "name": "Adaptive threshold"
                    },
                    "flags": [
                        {
                            "name": {
                                "id": "in_adaptive_baseline_rate.icmp",
                                "name": "ICMP"
                            },
                            "isSuspected": false
                        },
                        {
                            "name": {
                                "id": "in_adaptive_baseline_rate.udp",
                                "name": "UDP"
                            },
                            "isSuspected": false
                        },
                        {
                            "name": {
                                "id": "in_adaptive_baseline_rate.tcp_r",
                                "name": "TCP Rst"
                            },
                            "isSuspected": false
                        },
                        {
                            "name": {
                                "id": "in_adaptive_baseline_rate.tcp_s",
                                "name": "TCP Syn"
                            },
                            "isSuspected": false
                        },
                        {
                            "name": {
                                "id": "in_adaptive_baseline_rate.tcp_af",
                                "name": "TCP Ack Fin"
                            },
                            "isSuspected": false
                        },
                        {
                            "name": {
                                "id": "in_adaptive_baseline_rate.tcp_sa",
                                "name": "TCP Syn Ack"
                            },
                            "isSuspected": false
                        }
```

```
                    ]
                }
            ]
        },
        {
            "action": "Ended",
            "perfomed": "manually",
            "date": "2017-12-01 14:54:55",
            "actionId": null,
            "info": []
        },
        {
            "action": "Mitigation Start (reverted)",
            "perfomed": "manually",
            "date": "2017-12-01 18:09:06",
            "actionId": 123,
            "info": []
        }
    ]
}
]
```

3.2.4 Attack info

A method for acquisition of details about a specific attack searched by id.

GET /rest/iad/attacks/<id>

- Return attack information
- @return 200 Attack (3)

```
GET /rest/iad/attacks/380

Returns 200
{
    "id": 380,
    "segment": {
        "id": 22,
        "name": "Update by API"
```

```
},
"suspect": false,
"falsePositive": true,
"userComment": "",
"start": "2017-12-01 14:27:30",
"end": "2017-12-01 14:54:55",
"status": {
    "id": "ENDED",
    "name": "Finished attack"
},
"activeMitigation": 0,
"actions": [
    {
        "action": "Detected",
        "perfomed": null,
        "date": "2017-12-01 14:28:33",
        "actionId": null,
        "info": [
            {
                "method": {
                    "id": "in_adaptive_baseline_rate",
                    "name": "Adaptive threshold"
                },
                "flags": [
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.icmp",
                            "name": "ICMP"
                        },
                        "isSuspected": false
                    }
                ]
            }
        ]
    },
    {
        "action": "Detected",
        "perfomed": null,
        "date": "2017-12-01 14:49:33",
        "actionId": null,
        "info": [
            {
                "method": {
                    "id": "in_adaptive_baseline_rate",
                    "name": "Adaptive threshold"
                },
                "flags": [
                    {
```

```
                    "name": {
                        "id": "in_adaptive_baseline_rate.icmp",
                        "name": "ICMP"
                    },
                    "isSuspected": false
                },
                {
                    "name": {
                        "id": "in_adaptive_baseline_rate.tcp_r",
                        "name": "TCP Rst"
                    },
                    "isSuspected": false
                },
                {
                    "name": {
                        "id": "in_adaptive_baseline_rate.tcp_s",
                        "name": "TCP Syn"
                    },
                    "isSuspected": false
                },
                {
                    "name": {
                        "id": "in_adaptive_baseline_rate.tcp_af",
                        "name": "TCP Ack Fin"
                    },
                    "isSuspected": false
                },
                {
                    "name": {
                        "id": "in_adaptive_baseline_rate.tcp_sa",
                        "name": "TCP Syn Ack"
                    },
                    "isSuspected": false
                }
            ]
        }
    ]
},
{
    "action": "Detected",
    "perfomed": null,
    "date": "2017-12-01 14:54:33",
    "actionId": null,
    "info": [
        {
            "method": {
                "id": "in_adaptive_baseline_rate",
                "name": "Adaptive threshold"
```

```
                },
                "flags": [
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.icmp",
                            "name": "ICMP"
                        },
                        "isSuspected": false
                    },
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.udp",
                            "name": "UDP"
                        },
                        "isSuspected": false
                    },
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.tcp_r",
                            "name": "TCP Rst"
                        },
                        "isSuspected": false
                    },
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.tcp_s",
                            "name": "TCP Syn"
                        },
                        "isSuspected": false
                    },
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.tcp_af",
                            "name": "TCP Ack Fin"
                        },
                        "isSuspected": false
                    },
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.tcp_sa",
                            "name": "TCP Syn Ack"
                        },
                        "isSuspected": false
                    }
                ]
            }
        ]
    },
```

```
        {
            "action": "Ended",
            "perfomed": "manually",
            "date": "2017-12-01 14:54:55",
            "actionId": null,
            "info": []
        },
        {
            "action": "Mitigation Start (reverted)",
            "perfomed": "manually",
            "date": "2017-12-01 18:09:06",
            "actionId": 123,
            "info": []
        }
    ]
}
```

3.2.5 Attacks search

Method for search Attacks.

GET /rest/iad/attacks/

- Return Attacks
- @return 200 Attack (3)

```
GET /rest/iad/attacks/?search={
    "segment":[22],
    "status":"ENDED",
    "start":{
        "from":"2017-12-01 00:00:00",
        "to": "2017-12-01 23:59:59"
    }
}&limit=1&order=desc

Return 200:
[
    {
        "id": 380,
```

```
"segment": {
    "id": 22,
    "name": "Update by API"
},
"suspect": false,
"falsePositive": true,
"userComment": "",
"start": "2017-12-01 14:27:30",
"end": "2017-12-01 14:54:55",
"status": {
    "id": "ENDED",
    "name": "Finished attack"
},
"activeMitigation": 0,
"actions": [
    {
        "action": "Detected",
        "perfomed": null,
        "date": "2017-12-01 14:28:33",
        "actionId": null,
        "info": [
            {
                "method": {
                    "id": "in_adaptive_baseline_rate",
                    "name": "Adaptive threshold"
                },
                "flags": [
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.icmp",
                            "name": "ICMP"
                        },
                        "isSuspected": false
                    }
                ]
            }
        ]
    },
    {
        "action": "Detected",
        "perfomed": null,
        "date": "2017-12-01 14:49:33",
        "actionId": null,
        "info": [
            {
                "method": {
                    "id": "in_adaptive_baseline_rate",
                    "name": "Adaptive threshold"
```

```
                },
                "flags": [
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.icmp",
                            "name": "ICMP"
                        },
                        "isSuspected": false
                    },
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.tcp_r",
                            "name": "TCP Rst"
                        },
                        "isSuspected": false
                    },
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.tcp_s",
                            "name": "TCP Syn"
                        },
                        "isSuspected": false
                    },
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.tcp_af",
                            "name": "TCP Ack Fin"
                        },
                        "isSuspected": false
                    },
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.tcp_sa",
                            "name": "TCP Syn Ack"
                        },
                        "isSuspected": false
                    }
                ]
            }
        ]
    },
    {
        "action": "Detected",
        "perfomed": null,
        "date": "2017-12-01 14:54:33",
        "actionId": null,
        "info": [
            {
```

```
                "method": {
                    "id": "in_adaptive_baseline_rate",
                    "name": "Adaptive threshold"
                },
                "flags": [
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.icmp",
                            "name": "ICMP"
                        },
                        "isSuspected": false
                    },
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.udp",
                            "name": "UDP"
                        },
                        "isSuspected": false
                    },
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.tcp_r",
                            "name": "TCP Rst"
                        },
                        "isSuspected": false
                    },
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.tcp_s",
                            "name": "TCP Syn"
                        },
                        "isSuspected": false
                    },
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.tcp_af",
                            "name": "TCP Ack Fin"
                        },
                        "isSuspected": false
                    },
                    {
                        "name": {
                            "id": "in_adaptive_baseline_rate.tcp_sa",
                            "name": "TCP Syn Ack"
                        },
                        "isSuspected": false
                    }
                ]
```

```
                }
            ]
        },
        {
            "action": "Ended",
            "perfomed": "manually",
            "date": "2017-12-01 14:54:55",
            "actionId": null,
            "info": []
        },
        {
            "action": "Mitigation Start (reverted)",
            "perfomed": "manually",
            "date": "2017-12-01 18:09:06",
            "actionId": 123,
            "info": []
        }
    ]
}
]
```

4 Alert

4.1 Data Types

Alert

```
Templates = {
    id: String("Alert id"),
    name: String("Alert name")
}

Alert = {
    id: Integer,
    name: String,
    sendSyslog: Boolean,
    sendSnmp: Boolean,
    sendEmail: Boolean,
    email: String,
    temaplte: Templates("Available email tempaltes"),
    emailEvents: {
        detected: Boolean,
        statistics: Boolean,
        ended: Boolean
    }
    ipTranslate: Boolean,
    runScript: Boolean,
    scriptEvents: {
        detected: Boolean,
        statistics: Boolean,
        ended: Boolean
    },
    scriptParam: String,
    scriptContent: String("base64")
    lang: Language("Language of template")
}
```

4.2 Endpoints

4.2.1 Templates

A templates consists of dials (arrays) of available Alerts.

GET /rest/iad/alerts/templates

- Return CodeList
- @return 200 Template[]

```
GET /rest/iad/alerts/templates

Returns 200
    [
       {
          id: 1,
          name: "Default template"
       },
       {
          id: 2,
          name: "New Template by API"
       }
    ]
```

4.2.2 Alerts

4.2.2.1 Get all Alerts

This method serves for obtaining all alerts. The return is an array of available alerts.

GET /rest/iad/alerts/

- Return all Alerts
- @return 200 Alerts[] (4)

```
GET /rest/iad/alerts/

Returns 200
    [
```

```
    {
        "id": 1,
        "name": "Alert!",
        "sendSyslog": false,
        "sendSnmp": false,
        "sendEmail": true,
        "email": "test@test.cz",
        "emailEvents": {
            "detected": true,
            "statistics": false,
            "ended": true
        },
        "runScript": false,
        "scriptEvents": {
            "detected": false,
            "statistics": false,
            "ended": false
        },
        "template": {
            "id": 1,
            "name": "Default template"
        },
        "ipTranslations": false,
        "scriptParam": "",
        "scriptContent": ""
    },
    {
        "id": 2,
        "name": "Novy alert",
        "sendSyslog": false,
        "sendSnmp": false,
        "sendEmail": true,
        "email": "test@test.cz",
        "emailEvents": {
            "detected": true,
            "statistics": false,
            "ended": true
        },
        "runScript": true,
        "scriptEvents": {
            "detected": false,
            "statistics": false,
            "ended": true
        },
        "template": {
            "id": 19,
            "name": "New Template by API"
        },
```

```
            "ipTranslations": false,
            "scriptParam": "",
            "scriptContent": "IyEvYmluL2Jhc2gKZWNobyAiSGVsbG8sIFdvcmxkISIKZWNobyBEQVRFPWBkYXRlICcrJVkt="
        }
    ]
```

4.2.2.2 Get one specific Alert

A method for acquisition information about specific Alert searched by id.

GET /rest/iad/alerts/<id>

- Return Alert
- @return 200 Alert (4)

```
GET /rest/iad/alerts/1

Returns 200
        {
            "id": 1,
            "name": "Alert!",
            "sendSyslog": false,
            "sendSnmp": false,
            "sendEmail": true,
            "email": "test@test.cz",
            "emailEvents": {
                "detected": true,
                "statistics": false,
                "ended": true
            },
            "runScript": false,
            "scriptEvents": {
                "detected": false,
                "statistics": false,
                "ended": false
            },
            "template": {
                "id": 1,
                "name": "Default template"
            },
```

```
          "ipTranslations": false,
          "scriptParam": "",
          "scriptContent": ""
      }
```

4.2.2.3 Add new Alert

A method for adding a new Alert. New alert must have an empty id.

POST /rest/iad/alerts/

- Return new Alert
- @return 200 Alert (4)

```
POST /rest/iad/alerts/
BODY:
  {
        "entity":
    {
      "id": null,
      "name": "New Alert by API",
      "sendSyslog": true,
      "sendSnmp": false,
      "sendEmail": true,
      "email": "flowmon@flowmon.com",
      "emailEvents": {
          "detected": true,
          "statistics": false,
          "ended": true
      },
      "runScript": true,
      "scriptEvents": {
          "detected": false,
          "statistics": false,
          "ended": true
      },
      "template": {
          "id": 19,
          "name": "New Template by API"
      },
```

```
    "ipTranslations": false,
    "scriptParam": "",
    "scriptContent": "IyEvYmluL2Jhc2gKZWNobyAiSGVsbG8sIFdvcmxkISIKZWNobyBEQVRFPWBkYXRlICcrJVkt="
  }
}

Returns 200
{
    "id": 4,
    "name": "New Alert by API",
    "sendSyslog": true,
    "sendSnmp": false,
    "sendEmail": true,
    "email": "flowmon@flowmon.com",
    "emailEvents": {
        "detected": true,
        "statistics": false,
        "ended": true
    },
    "runScript": true,
    "scriptEvents": {
        "detected": false,
        "statistics": false,
        "ended": true
    },
    "template": {
        "id": 19,
        "name": "New Template by API"
    },
    "ipTranslations": false,
    "scriptParam": "",
    "scriptContent": "IyEvYmluL2Jhc2gKZWNobyAiSGVsbG8sIFdvcmxkISIKZWNobyBEQVRFPWBkYXRlICcrJVkt="
}
```

4.2.2.4 Update existing Alert

A method for updating Alert. ScriptContent is not an obligatory item. The content is overwritten only when you type.

PUT /rest/iad/alerts/

- Return updated Alert

- @return 200 Alert (4)

```
PUT /rest/iad/alerts/
BODY:
  {
        "entity":
    {
      "id": 4,
      "name": "Update Alert by API",
      "sendSyslog": true,
      "sendSnmp": false,
      "sendEmail": false,
      "email": "flowmon@flowmon.com",
      "emailEvents": {
          "detected": true,
          "statistics": false,
          "ended": true
      },
      "runScript": true,
      "scriptEvents": {
          "detected": false,
          "statistics": false,
          "ended": true
      },
      "template": {
          "id": 19,
          "name": "New Template by API"
      },
      "ipTranslations": false,
      "scriptParam": "",
      "scriptContent": "IyEvYmluL2Jhc2gKZWNobyAiSGVsbG8sIFdvcmxkISIKZWNobyBEQVRFPWBkYXRlICcrJVkt="
    }
  }

Returns 200
  {
      "id": 4,
      "name": "Update Alert by API",
      "sendSyslog": true,
      "sendSnmp": false,
      "sendEmail": false,
      "email": "flowmon@flowmon.com",
      "emailEvents": {
          "detected": true,
          "statistics": false,
          "ended": true
```

```
    },
    "runScript": true,
    "scriptEvents": {
        "detected": false,
        "statistics": false,
        "ended": true
    },
    "template": {
        "id": 19,
        "name": "New Template by API"
    },
    "ipTranslations": false,
    "scriptParam": "",
    "scriptContent": "IyEvYmluL2Jhc2gKZWNobyAiSGVsbG8sIFdvcmxkISIKZWNobyBEQVRFPWBkYXRlICcrJVkt="
}
```


4.2.2.5 Delete specific Alert

Deletes specific Alert by id. Return code is 204 (No data).

DELETE /rest/iad/alerts/<id>

- Return empty answer
- @return 204 No data

```
DELETE /rest/iad/alerts/4

Returns 204
```

5 Router

5.1 Data Types

Router

```
Router = {
    id: Integer,
    name: String,
    ip: String(IPv4),
    plugin: String,
    osci: String << ["ACL", "BGP"],
    aclName: String,
    login: String,
    password: String,
    protocol: Strong,
    locked: Boolean,
    acl6Name: String,
    community: String,
    asNum: Integer,
    asNumIad: Integer,
    bgpMode: String,
    nextHopIp: String(IPv4),
    nextHopIp6: String(IPv6),
    ip6: String(IPv6),
    interface: String,
    bgpNeighbour: bgpNeighbour("Only with PLUGIN = bgp and OSCI = defensepro")
}

bgpNeighBour = {
    ip: String,
    asNum: Integer,
    activated: Boolean
}
```

5.2 Endpoints

5.2.1 Routers

5.2.1.1 Get all Routers

This method serves for obtaining all Routers. The return is an array of available Routers.

GET /rest/iad/routers/

- Return all Routers
- @return 200 Routers[] (5)

```
GET /rest/iad/routers/

Returns 200
    [
       {
           "id": 1,
           "name": "First",
           "ip": "10.11.12.13",
           "aclName": "",
           "login": "test",
           "password": "password",
           "protocol": "ssh",
           "locked": false,
           "acl6Name": "",
           "community": "",
           "asNum": 1,
           "asNumIad": 1,
           "bgpMode": "ebgp",
           "nextHopIp": null,
           "nextHopIp6": null,
           "ip6": null,
           "interface": "auto",
           "bgpNeighbour": {
              "ip": "12.2.2.2",
              "asNum": 2,
              "activated": true
```

```
        },
        "osci": "exabgp",
        "plugin": "exabgp"
    },
    {
        "id": 6,
        "name": "test",
        "ip": "1.2.3.4",
        "aclName": "",
        "login": "",
        "password": "",
        "protocol": "ssh",
        "locked": false,
        "acl6Name": "",
        "community": "",
        "asNum": 1,
        "asNumIad": 1,
        "bgpMode": "ebgp",
        "nextHopIp": null,
        "nextHopIp6": null,
        "ip6": null,
        "interface": "auto",
        "bgpNeighbour": null,
        "osci": "exabgp",
        "plugin": "exabgp"
    }
]
```


5.2.1.2 Get one specific Router

A method for acquisition of informations about a specific Router searched by id.

GET /rest/iad/routers/<id>

- Return Router
- @return 200 Router (5)

```
GET /rest/iad/routers/1

Returns 200
```

```
  {
      "id": 6,
      "name": "test",
      "ip": "1.2.3.4",
      "aclName": "",
      "login": "",
      "password": "",
      "protocol": "ssh",
      "locked": false,
      "acl6Name": "",
      "community": "",
      "asNum": 1,
      "asNumIad": 1,
      "bgpMode": "ebgp",
      "nextHopIp": null,
      "nextHopIp6": null,
      "ip6": null,
      "interface": "auto",
      "bgpNeighbour": null,
      "osci": "exabgp",
      "plugin": "exabgp"
  }
```


5.2.1.3 Add new Router

A method for adding a new Router. New Router must have empty id.

POST /rest/iad/routers/

- Return new Router
- @return 200 Router (5)


```
POST /rest/iad/routers/
BODY:
  {
        "entity":
        {
              "id": null,
              "name": "First",
              "ip": "10.11.12.15",
```

```
            "aclName": "",
            "login": "test",
            "password": "password",
            "protocol": "ssh",
            "acl6Name": "",
            "community": "",
            "asNum": 1,
            "asNumIad": 1,
            "bgpMode": "ebgp",
            "nextHopIp": null,
            "nextHopIp6": null,
            "ip6": null,
            "interface": "auto",
            "bgpNeighbour": {
                "ip": "12.2.2.2",
                "asNum": 2,
                "activated": true
            },
            "osci": "exabgp",
            "plugin": "exabgp"
    }
 }

Returns 200
 {
    "id": 8,
    "name": "First",
    "ip": "10.11.12.15",
    "aclName": "",
    "login": "test",
    "password": "password",
    "protocol": "ssh",
    "locked": false,
    "acl6Name": "",
    "community": "",
    "asNum": 1,
    "asNumIad": 1,
    "bgpMode": "ebgp",
    "nextHopIp": null,
    "nextHopIp6": null,
    "ip6": null,
    "interface": "auto",
    "bgpNeighbour": {
        "ip": "12.2.2.2",
        "asNum": 2,
        "activated": true
    },
    "osci": "exabgp",
```

```
    "plugin": "exabgp"
}
```

5.2.1.4 Update existing Router

A method for updating a Router.

PUT /rest/iad/routers/

- Return updated Router
- @return 200 Router (5)

```
PUT /rest/iad/routers/
BODY:
  {
        "entity":
        {
              "id": 8,
              "name": "First",
              "ip": "10.11.12.15",
              "aclName": "",
              "login": "",
              "password": "",
              "protocol": "ssh",
              "acl6Name": "",
              "community": "",
              "asNum": 1,
              "asNumIad": 1,
              "bgpMode": "ebgp",
              "nextHopIp": null,
              "nextHopIp6": null,
              "ip6": null,
              "interface": "auto",
              "bgpNeighbour": {
                  "ip": "12.2.2.3",
                  "asNum": 2,
                  "activated": true
              },
              "osci": "defensepro",
              "plugin": "exabgp"
```

```
    }
  }

Returns 200
  {
    "id": 8,
    "name": "First",
    "ip": "10.11.12.15",
    "plugin": "exabgp",
    "osci": "defensepro",
    "aclName": "",
    "login": "",
    "password": "",
    "protocol": "ssh",
    "locked": false,
    "acl6Name": "",
    "community": "",
    "asNum": 1,
    "asNumIad": 1,
    "bgpMode": "ebgp",
    "nextHopIp": null,
    "nextHopIp6": null,
    "ip6": null,
    "interface": "auto",
    "bgpNeighbour": {
        "ip": "12.2.2.3",
        "asNum": 2,
        "activated": true
    }
  }
```

5.2.1.5 Delete specific Router

Deletes specific Router by id. Return code is 204 (No data).

DELETE /rest/iad/routers/<id>

- Return empty answer
- @return 204 No data

```
DELETE /rest/iad/routers/8
```

```
Returns 204
```

6 Scrubbing Center

6.1 Data Types

Scrubbing Center

```
ScrubbingCenter = {
    id: Integer,
    name: String,
    ip: String(IPv4|IPv6),
    osci: String,
    login: String,
    password: String,
    locked: Boolean(ReadOnly),
    options: String
}
```

6.2 Endpoints

6.2.1 Scrubbing Centers

6.2.1.1 Get all Scrubbing Centers

This method serves for obtaining all Scrubbing Centers. The return is a array of available Scrubbing Centers.

GET /rest/iad/scrubbing-centers/

- Return all Scrubbing Centers
- @return 200 ScrubbingCenters[] (6)

```
GET /rest/iad/scrubbing-centers/

Returns 200
    [
        {
```

```
        "id": 1,
        "name": "Moje AbsoluteVision3",
        "ip": "192.168.50.122",
        "osci": "APSolute Vision",
        "login": "aaaa",
        "password": "aaaa",
        "locked": false,
        "options": "keep_rtemplate"
    },
    {
        "id": 3,
        "name": "scrub c to fail",
        "ip": "192.168.55.58",
        "osci": "APSolute Vision",
        "login": "aaaa",
        "password": "aaaa",
        "locked": false,
        "options": "keep_rtemplate"
    }
]
```

6.2.1.2 Get one specific Scrubbing Center

Method for acquisition of informations about specific Scrubbing Center searched by id.

GET /rest/iad/scrubbing-centers/<id>

- Return Scrubbing Center
- @return 200 ScrubbingCenter (6)

```
GET /rest/iad/scrubbing-centers/1

Returns 200
[
    {
        "id": 1,
        "name": "Moje AbsoluteVision3",
        "ip": "192.168.50.122",
        "osci": "APSolute Vision",
        "login": "aaaa",
        "password": "aaaa",
```

```
      "locked": false,
      "options": "keep_rtemplate"
  }
```

6.2.1.3 Add new Scrubbing Center

Method for adding a new Scrubbing Center. A new Scrubbing Center must have an empty id.

POST /rest/iad/scrubbing-centers/

- Return new Scrubbing Center
- @return 200 ScrubbingCenter (6)

```
POST /rest/iad/scrubbing-centers/
BODY:
  {
      "entity":
      {
      "id": null,
      "name": "Add by Api",
      "ip": "192.168.50.207",
      "osci": "APSolute Vision",
      "login": "aaaa",
      "password": "bbbb",
      "options": "keep_rtemplate"
    }
  }

Returns 200
  {
    "id": 6,
    "name": "Add by Api",
    "ip": "192.168.50.207",
    "osci": "APSolute Vision",
    "login": "aaaa",
    "password": "bbbb",
    "locked": false,
    "options": "keep_rtemplate"
  }
```

6.2.1.4 Update existing Scrubbing Center

A method for update a Scrubbing Center. "Locked" is read-only attribute.

PUT /rest/iad/scrubbing-centers/

- Return updated Scrubbing Center
- @return 200 ScrubbingCenter (6)

```
PUT /rest/iad/scrubbing-centers/
BODY:
  {
        "entity":
        {
        "id": 6,
        "name": "Update by API",
        "ip": "192.168.51.207",
        "osci": "APSolute Vision",
        "login": "aaaa",
        "password": "bbbb",
        "options": "keep_rtemplate"
      }
  }

Returns 200
  {
      "id": 6,
      "name": "Update by API",
      "ip": "192.168.51.207",
      "osci": "APSolute Vision",
      "login": "aaaa",
      "password": "bbbb",
      "locked": false,
      "options": "keep_rtemplate"
  }
```

6.2.1.5 Delete specific Scrubbing Center

Deletes specific Scrubbing Center by id. Return code is 204 (No data).

DELETE /rest/iad/scrubbing-centers/<id>

- Return empty answer
- @return 204 No data

```
DELETE /rest/iad/scrubbing-centers/6

Returns 204
```

7 Scrubbing Center Parameter

7.1 Data Types

Scrubbing Center Parameter

```
ScrubbingCenterParameter = {
    option: String("Key of option"),
    value: String
}
```

7.2 Endpoints

7.2.1 Scrubbing Center Parameters

7.2.1.1 Get all defaults Scrubbing Center Parameters

This method serve forobtaining all default Scrubbing Center Parameters. Return is an array of available Scrubbing Center Parameters.

GET /rest/iad/scrubbing-center-parameters

- Return all Scrubbing Center Parameters
- @return 200 ScrubbingCenterParameters[] (7)

```
GET /rest/iad/scrubbing-center-parameters

Returns 200
    [
        {
            "option": "dns_a_in_ipv4",
            "value": "72"
        },
        {
            "option": "dns_a_in_ipv6",
            "value": "72"
```

```
        },
        {
            "option": "dns_a_out_ipv4",
            "value": "0"
        },
        {
            "option": "dns_a_out_ipv6",
            "value": "0"
        },
        {
            "option": "dns_mx_in_ipv4",
            "value": "42"
        },
        {
            "option": "dns_mx_in_ipv6",
            "value": "42"
        },

        ...

        {
            "option": "con_limit_udp",
            "value": "30000"
        }
    ]
```

7.2.1.2 Update existing default Scrubbing Center Parameters

Method for updating Scrubbing Center Parameters. It is possible to update one or more parameters at the same time.

PUT /rest/iad/scrubbing-center-parameters

- Return updated Scrubbing Center Parameters
- @return 200 ScrubbingCenterParameters[] (7)

```
PUT /rest/iad/scrubbing-center-parameters
BODY:
  {
        "entity":
```

```
      [
    {
        "option": "dns_a_in_ipv4",
        "value": "42"
    },
    {
        "option": "dns_ptr_in_ipv6",
        "value": "72"
    }
  ]
}

Returns 200
 [
    {
        "option": "dns_a_in_ipv4",
        "value": "72"
    },
    {
        "option": "dns_a_in_ipv6",
        "value": "72"
    },
    {
        "option": "dns_a_out_ipv4",
        "value": "0"
    },
    {
        "option": "dns_a_out_ipv6",
        "value": "0"
    },
    {
        "option": "dns_mx_in_ipv4",
        "value": "42"
    },
    {
        "option": "dns_mx_in_ipv6",
        "value": "42"
    },
    {
        "option": "dns_mx_out_ipv4",
        "value": "0"
    },
    {
        "option": "dns_mx_out_ipv6",
        "value": "0"
    },
    {
        "option": "dns_ptr_in_ipv4",
```

```
        "value": "42"
    },
    {
        "option": "dns_ptr_in_ipv6",
        "value": "42"
    },

    ...

    {
        "option": "con_limit_udp",
        "value": "30000"
    }
]
```

8 Report Chapters

8.1 Data Types

CodeList

```
CodeList = {
    id: Integer|String("Item id"),
    name: String("Item name")
}
```

Chapter

```
Chapter = {
    id: Integer,
    name: String,
    description: String,
    scope: ChapterScope("Whether it is overall or segment based chapter"),

    //TODO - better name
    template: ChapterTemplate("What (predefined) data is displayed")
    parameters: ChapterParameters("Parameters this chapter is defined with, differ for various scopes")
}

ChapterParameters = OverallChapterParameters | PerSegmentChapterParameters

OverallChapterParameters = {
    segments: ProtectedSegment.{id, name}[]("Array of selected segments this chapter reports attacks from"),
    attackedOnly: Integer(Boolean("If set to true, only attacked segments will be shown"))
}

PerSegmentChapterParameters = {
    segment: ProtectedSegment.{id,name}("Explicit segment this chapter will display data for"),
    nthTop: Integer("Index of segment this chapter will display data for")
}

ChapterScope extends CodeList << [
  {id: 1, name: "Overall chapters"},
  {id: 2, name: "Per-segment chapters"}
]

ReportedMetric extends CodeList << [
  {id: "overall.attacksOverview", name: "Attacks overview"},
  {id: "overall.ongoingAttacks", name: "Ongoing attacks"},
  {id: "overall.overallStats", name: "Overall stats"},
  {id: "overall.topAttacksPerSegment", name: "Top attacks per segment"},
  {id: "overall.topAttacksPerType", name: "Top attacks per type"},
```

```
  {id: "segment.attacksOverview", name: "Attacks overview for segment XXX"},
  {id: "segment.overallStats", name: "Overall stats for segment XXX"},
  {id: "segment.topAttacksPerType", name: "Top attacks per type for segment XXX"}
]
```

8.2 Endpoints

8.2.1 Code List

Code Lists are consist of dials (arrays), where id can be integer or string.

GET /rest/iad/report-chapter/scopes

or

GET /rest/iad/report-chapter/templates

- Return Code List
- @return 200 CodeList[] (8.1)

```
GET /rest/iad/report-chapter/scopes

Returns 200
[
    {
        "id": 1,
        "name": "Overall chapters"
    },
    {
        "id": 2,
        "name": "Per-segment chapters"
    }
]

GET /rest/iad/report-chapter/templates

Returns 200
[
```

```
    {
        "id": "overall.attacksOverview",
        "name": "Attacks overview"
    },
    {
        "id": "overall.ongoingAttacks",
        "name": "Ongoing attacks"
    },
    {
        "id": "overall.overallStats",
        "name": "Overall stats"
    },
    {
        "id": "overall.topAttacksPerSegment",
        "name": "Top attacks per segment"
    },
    {
        "id": "overall.topAttacksPerType",
        "name": "Top attacks per type"
    },
    {
        "id": "segment.attacksOverview",
        "name": "Attacks overview for segment XXX"
    },
    {
        "id": "segment.overallStats",
        "name": "Overall stats for segment XXX"
    },
    {
        "id": "segment.topAttacksPerType",
        "name": "Top attacks per type for segment XXX"
    }
]
```


8.2.2 Chapters

8.2.2.1 Get all Chapters

This method returns list od all Report Chapters.

GET /rest/iad/report-chapter/

- Return all Chapters
- @return 200 Chapter[] (8.1)
- @return 204 No data

```
GET /rest/iad/report-chapter/

Returns 200
[
    {
        "id": 1,
        "name": "Chapter",
        "description": "Description asd",
        "parameters": {
            "segments": [
                {
                    "id": 7,
                    "name": "Prvn"
                }
            ],
            "attackedOnly": true
        },
        "scope": {
            "id": 1,
            "name": "Overall chapters"
        },
        "template": {
            "id": "overall.overallStats",
            "name": "Overall stats"
        }
    },
    {
        "id": 2,
        "name": "Chapter no. 2",
        "description": "Description new",
        "parameters": {
            "segment": {
                "id": 7,
                "name": "Prvn"
            },
            "nthBest": null
        },
        "scope": {
            "id": 2,
            "name": "Per-segment chapters"
```

```
        },
        "template": {
            "id": "segment.overallStats",
            "name": "Overall stats for segment XXX"
        }
    }
]
```

8.2.2.2 Get one specific Chapter

This method returns details of single Report Chapter given by its ID.

GET /rest/iad/report-chapter/<id>

- @param <id> ReportChapter ID
- Return Chapter
- @return 200 Chapter (8.1)

```
GET /rest/iad/report-chapter/1

Returns 200
{
    "id": 1,
    "name": "Chapter",
    "description": "Popis",
    "parameters": {
        "segments": [
            {
                "id": 7,
                "name": "Prvn"
            }
        ],
        "attackedOnly": true
    },
    "scope": {
        "id": 1,
        "name": "Overall chapters"
    },
    "template": {
```

```
        "id": "overall.overallStats",
        "name": "Overall stats"
    }
}
```

8.2.2.3 Add new Chapter

This method creates a new Report Chapter. New Report Chapter must have unique combination of name, type and selected Segments.

POST /rest/iad/report-chapter/

- Return new Chapter
- @return 405 ValidationResult[]
- @return 201 Chapter (8.1)

```
POST /rest/iad/report-chapter/
BODY:
{
        "entity":
        {
            "id": null,
            "name": "Chapter new REST",
            "description": "Description of chapter",
            "scope": {
                "id": 1,
                "name": "Overall chapters"
            },
            "template": {
                "id": "overall.overallStats",
                "name": "Overall stats"
            },
            "parameters": {
                "segments": [
                    {
                        "id": 7,
                        "name": "Prvn"
                    }
                ],
```

```
                "attackedOnly": true
            }
        }
}

Returns 201
{
    "id": 1,
    "name": "Chapter new REST",
    "description": "Description of chapter",
    "parameters": {
        "segments": [
            {
                "id": 7,
                "name": "Prvn"
            }
        ],
        "attackedOnly": true
    },
    "scope": {
        "id": 1,
        "name": "Overall chapters"
    },
    "template": {
        "id": "overall.overallStats",
        "name": "Overall stats"
    }
}
```

8.2.2.4 Update existing Chapter

This method updates an existing Report Chapter.

PUT /rest/iad/report-chapter/

- Return an updated Chapter
- @return 404 Not found
- @return 405 ValidationResult[]
- @return 200 Chapter (8.1)

```
PUT /rest/iad/report-chapter/
BODY:
{
        "entity":
        {
            "id": 1,
            "name": "Chapter new REST",
            "description": "New description",
            "scope": {
                "id": 1,
                "name": "Overall chapters"
            },
            "template": {
                "id": "overall.overallStats",
                "name": "Overall stats"
            },
            "parameters": {
                "segments": [
                    {
                        "id": 7,
                        "name": "Prvn"
                    }
                ],
                "attackedOnly": true
            }
        }
}

Returns 200
{
    "id": 1,
    "name": "Chapter new REST",
    "description": "New description",
    "parameters": {
        "segments": [
            {
                "id": 7,
                "name": "Prvn"
            }
        ],
        "attackedOnly": true
    },
    "scope": {
        "id": 1,
        "name": "Overall chapters"
    },
    "template": {
```

```
        "id": "overall.overallStats",
        "name": "Overall stats"
    }
}
```

8.2.2.5 Delete specific Chapter

Delete specific Report Chapter by its id. Return code is 204 (No data).

DELETE /rest/iad/report-chapter/<id>

- Return empty answer
- @return 204 No data

```
DELETE /rest/iad/report-chapter/1

Returns 204
```

9 Email Templates

9.1 Data Types

Languages

```
Language = {
    id: String("Language id"),
    name: String("Language name")
}
```

EmailTemplate

```
EmailTemplate = {
    id: Integer,
    name: String,
    subject: String,
    body: String,
    lang: Language("Language of template")
}
```

9.2 Endpoints

9.2.1 Languages

This method returns a list of available languages.

GET /rest/iad/email-templates/languages

- @return 200 Language[] (9.1)

```
GET /rest/iad/email-templates/languages

Returns 200
    [
       {
          id: "cz",
          name: "Cesky"
```

```
      },
      {
         id: "en",
         name: "English"
      },
      {
         id: "jp",
         name: "Japan"
      },
   ]
```

9.2.2 EmailTemplates

9.2.2.1 Get all Email Templates

This method serve for obtaining all Email Templates. The return is an array of available templates.

GET /rest/iad/email-templates/

- Return all EmailTemplates
- @return 200 EmailTemplate[] (9)

```
GET /rest/iad/email-templates/

Returns 200
    [
      {
          "id": 1,
          "name": "Default template",
          "subject": "DDD Alert on %DEVICE: %TIME - %SEGMENT - %EVENT",
          "body": "Hi,

Segment: %SEGMENT (subnets: %SUBNETS)
Time: %TIME
Event type: %EVENT
Triggered detection methods: %METHODS
Mitigation configuration: %MITIGATION_TYPE
Mitigation status: %MITIGATION_STATUS
```

```
Flowmon device: %DEVICE

You can check the event here: %LINK

Your Flowmon DDoS Defender",
           "lang": {
               "id": "en",
               "name": "English"
           }
       },
       {
           "id": 2,
           "name": "Second template",
           "subject": "DDD Alert on %DEVICE: %TIME - %SEGMENT - %EVENT",
           "body": "Hi,

Segment: %SEGMENT (subnets: %SUBNETS)
Time: %TIME
Event type: %EVENT
Triggered detection methods: %METHODS
Mitigation configuration: %MITIGATION_TYPE
Mitigation status: %MITIGATION_STATUS
Flowmon device: %DEVICE

You can check the event here: %LINK

Your Flowmon DDoS Defender",
           "lang": {
               "id": "en",
               "name": "English"
           }
       }
    ]
```

9.2.2.2 Get one specific Email Template

A method for acquisition of informations about a specific Email Template searched by its id.

GET /rest/iad/email-templates/<id>

- Return Email Template
- @return 200 EmailTemplate (9)

```
GET /rest/iad/email-templates/1

Returns 200
    {
        "id": 1,
        "name": "Default template",
        "subject": "DDD Alert on %DEVICE: %TIME - %SEGMENT - %EVENT",
        "body": "Hi,

Segment: %SEGMENT (subnets: %SUBNETS)
Time: %TIME
Event type: %EVENT
Triggered detection methods: %METHODS
Mitigation configuration: %MITIGATION_TYPE
Mitigation status: %MITIGATION_STATUS
Flowmon device: %DEVICE
You can check the event here: %LINK
Your Flowmon DDoS Defender",
        "lang": {
            "id": "en",
            "name": "English"
        }
    }
```

9.2.2.3 Add new Email Template

A method for adding a new Email Template. A new template must have empty id.

POST /rest/iad/email-template/

- Return new Email Template
- @return 200 EmailTemplate (9.1)

```
POST /rest/iad/email-template/
BODY:
  {
        "entity":
        {
            "id": null,
```

```
           "name": "New Template by API",
           "subject": "DDD Alert on %DEVICE: %TIME - %SEGMENT - %EVENT",
           "body": "Hi,

Segment: %SEGMENT (subnets: %SUBNETS)
Time: %TIME
Event type: %EVENT
Triggered detection methods: %METHODS
Mitigation configuration: %MITIGATION_TYPE",
       "lang": {
              "id": "en",
              "name": "English"
       }
       }
  }

Returns 200
       {
           "id": 3,
           "name": "New Template by API",
           "subject": "DDD Alert on %DEVICE: %TIME - %SEGMENT - %EVENT",
           "body": "Hi,

Segment: %SEGMENT (subnets: %SUBNETS)
Time: %TIME
Event type: %EVENT
Triggered detection methods: %METHODS
Mitigation configuration: %MITIGATION_TYPE",
       "lang": {
              "id": "en",
              "name": "English"
       }
       }
```

9.2.2.4 Update existing Email Template

A method for updating an Email Template.

PUT /rest/iad/email-template/

- Return updated Email Template
- @return 200 EmailTemplate (9.1)

```
PUT /rest/iad/email-template/
BODY:
  {
        "entity":
        {
            "id": 3,
            "name": "Edit Template by API",
            "subject": "DDD Alert on %DEVICE: %TIME - %SEGMENT - %EVENT",
            "body": "Hi,

Segment: %SEGMENT (subnets: %SUBNETS)
Time: %TIME
Event type: %EVENT
Triggered detection methods: %METHODS
Mitigation configuration: %MITIGATION_TYPE",
        "lang": {
                "id": "en",
                "name": "English"
        }
        }
  }

Returns 200
        {
            "id": 3,
            "name": "Edit Template by API",
            "subject": "DDD Alert on %DEVICE: %TIME - %SEGMENT - %EVENT",
            "body": "Hi,
Segment: %SEGMENT (subnets: %SUBNETS)
Time: %TIME
Event type: %EVENT
Triggered detection methods: %METHODS
Mitigation configuration: %MITIGATION_TYPE",
        "lang": {
                "id": "en",
                "name": "English"
        }
        }
```

9.2.2.5 Delete specific Email Template

Delete a specific Email Template by its id. Return code is 204 (No data). Email Template with id 1 cannot be deleted.

DELETE /rest/iad/email-template/<id>

- Return empty answer
- @return 204 No data

```
DELETE /rest/iad/email-template/3

Returns 204
```

10 Rule

10.1 Data Types

Rule

```
Rule = {
    id: Integer,
    name: String,
    inOutRatio: Integer("Incoming / Outgoing packets ratio"),
    manualThreshold: Integer("Number of incoming packets is M\% of baseline"),
    baselineInMinutes: Integer("Lenght of learning period"),
    rulesEnabled: String[],
    minimalTraffic: {
      bps: Integer("Minimal bits per second to activate rule") or null,
      pps: Integer("Minimal packets per second to activatre rule") or null
   } or null,
    slots: Integer("This option can be used to avoid false attack detections for short data bursts.
                    If you have selected 1 (0:30), check whether active timeout on your flow exporting
                    device is < 30s to avoid false attack detections")
}
```

10.2 Endpoints

10.2.1 Rules

10.2.1.1 Get all Rules

This method serves for obtaining all Rules. Return is an array of available Rules.

GET /rest/iad/rules/

- Return all Rules
- @return 200 Rules[] (10)

```
GET /rest/iad/rules/
```

```
Returns 200
    [
        {
            "id": 1,
            "name": "Default rule",
            "inOutRatio": 1000,
            "manualThreshold": 200,
            "baseline": 2016,
            "baselineInMinutes": 10080,
            "rulesEnabled": [
                "in_out_rate",
                "in_baseline_rate",
                "in_adaptive_baseline_rate"
            ],
            "slots": 2,
            "flowspecRoutes": null,
            "minimalTraffic": {
                "bps": 20000000,
                "pps": 2000
            }
        },
        {
            "id": 3,
            "name": "test",
            "inOutRatio": 0,
            "manualThreshold": 1200,
            "baseline": 2,
            "baselineInMinutes": 10,
            "rulesEnabled": [
                "in_baseline_rate"
            ],
            "slots": 2,
            "flowspecRoutes": null,
            "minimalTraffic": {
                "bps": 10000000,
                "pps": 2000
            }
        }
    ]
```

10.2.1.2 Get one specific Rule

A method for acquisition of informations about a specific Rule searched by its id.

GET /rest/iad/rules/<id>

- Return Rule
- @return 200 Rule (10)

```
GET /rest/iad/rules/3

Returns 200
    {
        "id": 3,
        "name": "test",
        "inOutRatio": 0,
        "manualThreshold": 1200,
        "baseline": 2,
        "baselineInMinutes": 10,
        "rulesEnabled": [
            "in_baseline_rate"
        ],
        "slots": 2,
        "flowspecRoutes": null,
        "minimalTraffic": {
            "bps": 10000000,
            "pps": 2000
        }
    }
```

10.2.1.3 Add new Rule

A method for adding a new Rule. A new Rule must have empty id.

POST /rest/iad/rules/

- Return new Rule
- @return 201 Rule (10)

```
POST /rest/iad/rules/
```

```
BODY:
  {
        "entity":
    {
        "id": null,
        "name": "New Rule by API",
        "inOutRatio": 1000,
        "manualThreshold": 200,
        "baseline": 2016,
        "baselineInMinutes": 10080,
        "rulesEnabled": [
            "in_out_rate",
            "in_baseline_rate",
            "in_adaptive_baseline_rate"
        ],
        "slots": 2,
        "flowspecRoutes": null,
        "minimalTraffic": {
            "bps": 20000000,
            "pps": 2000
        }
    }
  }

Returns 201
        {
            "id": 6,
            "name": "New Rule by API",
            "inOutRatio": 1000,
            "manualThreshold": 200,
            "baselineInMinutes": 10080,
            "rulesEnabled": [
                "in_out_rate",
                "in_baseline_rate",
                "in_adaptive_baseline_rate"
            ],
            "slots": 2,
            "minimalTraffic": {
                "bps": 20000000,
                "pps": 2000
            }
        }
```

10.2.1.4 Update existing Rule

A method for updating a Rule.

PUT /rest/iad/rules/

- Return updated Rule
- @return 200 Rule (10)

```
PUT /rest/iad/rules/
BODY:
  {
        "entity":
    {
        "id": 6,
        "name": "Updated Rule by API",
        "inOutRatio": 1000,
        "manualThreshold": 200,
        "baseline": 5,
        "baselineInMinutes": 25,
        "rulesEnabled": [
            "in_out_rate",
            "in_baseline_rate",
            "in_adaptive_baseline_rate"
        ],
        "slots": 2,
        "flowspecRoutes": null,
        "minimalTraffic": {
            "bps": 20000000,
            "pps": 2000
        }
    }
  }

Returns 200
{
    "id": 6,
    "name": "Updated Rule by API",
    "inOutRatio": 1000,
    "manualThreshold": 200,
    "baselineInMinutes": 25,
    "rulesEnabled": [
```

```
        "in_out_rate",
        "in_baseline_rate",
        "in_adaptive_baseline_rate"
    ],
    "slots": 2,
    "minimalTraffic": {
        "bps": 20000000,
        "pps": 2000
    }
}
```

10.2.1.5 Delete specific Rule

Deletes specific Rule by its id. Return code is 204 (No data).

DELETE /rest/iad/rules/<id>

- Return empty answer
- @return 204 No data

```
DELETE /rest/iad/rules/4

Returns 204
```

11 Segment

11.1 Data Types

Segment

```
Segment = {
   id: Integer,
   name: String,
   filterIn: String("input filter"),
   filterOut: String("output filter"),
   profileName: String("Name of profile generated by this segment"),
   rule: Rule({id, name}) or null (for allways on),
   timeout: Integer("Positive integer, count of seconds, use -1 as "Never""),
   delay: Integer("Delay of flow data beyond the Scrubbing Center"),
   autoSubnet: Integer("minimal IPv4 subnet mask"),
   autoSubnetIpv6: Integer("minimal IPv6 subnet mask"),
   bandwidthDay: Integer("Prumerny bandwidth za predchozi den bez utoku TRANSLATE"),
   trafficSeen: Timestamp("First segment traffic"),
   status: String << [succ, error, warning],
   bandwidth: Integer("max bandwidth in bps"),
   bandwidthAuto: Integer(Boolean)("is bandwidth determined automatically"),
   actionSubnet: ActionSubnets << ["subnets", "preferred", "bgp_community", "automatic"],
   type: SegmentTypes("subnets or as numbers") << [1, 2],
   asNumbers: [Integer]("AS numbers"),
   subnets: [String("Masked IP")],
   preferredSubnets: String[]("Masked IP"),
   parentProfile: ProfileAndChannels("Parent profile and selected channels"),

   //measures to be taken to suppress attack
   measures: {
      mode: String("How actions should be triggered") << ["manual", "auto"],

      alert: Alert({id, name}, "Alert to be triggered") or null,
      reroute: [SegmentRouter, "Router to change route")] or null,
      mitigate: {
         scrubbingCenter: ScrubbingCenter({id,name}, "Scrubbings center to mittigate attack") or null
         *parameters: ScrubbingCenterParameter[]( {"option", "value"} "Array of scrubbing center parameters,
                                                                      available only if scrubbing center is
                                                                      one of Vission scrubinng centers."),
      }
   }
}

SegmentRouter = {
  routerId: Integer,
```

```
  routerName: String,
  community: String,
  bgpMode: String or null
}

ProfileAndChannels = {
    profile: String("Profile id"),
    channels: String[]("Array of selected channel ids or "*" for all parent channels.")
}

CodeList = {
    id: Integer|String("Item id"),
    name: String("Item name")
}

SegmentTypes extends CodeList << [
  {id: 1, name: "Subnets"},
  {id: 2, name: "As numbers"}
]

ActionSubnets extends CodeList << [
  {id: "subnet", name: "Subnets"},
  {id: "preferred", name: "Preferred subnets"},
  {id: "bgp_community", name: "By BGP Community"},
  {id: "automatic", name: "Autodetected subnets"}
]
```

11.2 Endpoints

11.2.1 Segments

11.2.1.1 Get all Segments

This method serve for obtaining all Segments. The return is an array of available Segments.

GET /rest/iad/segments/

- Return all Segments
- @return 200 Segments[] (11)

```
GET /rest/iad/segments/

Returns 200
     [
        {
            "id": 10,
            "name": "druhy",
            "filterIn": "(( not (src net 192.168.50.207/21 and ipv4)))
                            and
                        (( dst net 192.168.50.207/21 and ipv4))",
            "filterOut": "(( src net 192.168.50.207/21 and ipv4))
                              and
                            (( not (dst net 192.168.50.207/21 and ipv4)))",
            "profileName": "ddosdefender/druhy-10097e1",
            "rule": {
                "id": 1,
                "name": "Default rule"
            },
            "timeout": 120,
            "delay": 1,
            "autoSubnet": null,
            "autoSubnetIpV6": null,
            "bandwidthDay": null,
            "trafficSeen": "2017-11-29 13:24:30",
            "status": null,
            "bandwidth": 0,
            "bandwidthAuto": 0,
            "actionSubnet": {
                "id": "subnets",
                "name": "Subnets"
            },
            "type": {
                "id": 1,
                "name": "Subnets"
            },
            "asNumbers": [],
            "subnets": [
                "192.168.50.207/21"
            ],
            "preferredSubnets": [
                "12.15.16.10/32",
                "12.15.16.11/32"
            ],
            "parentProfile": {
                "profile": "icmp",
                "channels": [
                    "*"
```

```
            ]
        },
        "measures": {
            "mode": "auto",
            "alert": null,
            "reroute": [
                {
                    "routerId": 6,
                    "routerName": "First",
                    "community": null,
                    "bgpMode": null
                }
            ],
            "mitigate": {
                "scrubbingCenter": {
                    "id": 3,
                    "name": "my APSolute Vision 3.00"
                },
                "parameters": [
                    {
                        "option": "dns_a_in_ipv4",
                        "value": "72"
                    },
                    {
                        "option": "dns_a_in_ipv6",
                        "value": "72"
                    }

                     ...

                    {
                        "option": "con_limit_udp",
                        "value": "30000"
                    }
                ]
            }
        }
    },
    {
        "id": 11,
        "name": "VivaLasVegas",
        "filterIn": "(( not (src net 12.2.2.2/32 and ipv4))) and (( dst net 12.2.2.2/32 and ipv4))",
        "filterOut": "(( src net 12.2.2.2/32 and ipv4)) and (( not (dst net 12.2.2.2/32 and ipv4)))",
        "profileName": "ddosdefender/vija-83728376",
        "rule": {
            "id": 4,
            "name": "Baseline"
        },
```

```
            "timeout": 120,
            "delay": 1,
            "autoSubnet": null,
            "autoSubnetIpV6": null,
            "bandwidthDay": null,
            "trafficSeen": "2017-11-24 11:04:30",
            "status": null,
            "bandwidth": 0,
            "bandwidthAuto": 0,
            "actionSubnet": {
                "id": "subnets",
                "name": "Subnets"
            },
            "type": {
                "id": 1,
                "name": "Subnets"
            },
            "asNumbers": [],
            "subnets": [
                "12.2.2.2/32"
            ],
            "preferredSubnets": null,
            "parentProfile": {
                "profile": "QoS_ToS",
                "channels": [
                    "Networkcontrol",
                    "Critical",
                    "Flash"
                ]
            },
            "measures": {
                "mode": "manual",
                "alert": null,
                "reroute": null,
                "mitigate": {
                    "scrubbingCenter": null,
                    "parameters": null
                }
            }
        }
    ]
```


11.2.1.2 Get one specific Segment

A method for acquisition of informations about a specific Segment searched by its id.

GET /rest/iad/segments/<id>

- Return Segment
- @return 200 Segment (11)

```
GET /rest/iad/segments/10

Returns 200
     {
         "id": 10,
          "name": "druhy",
            "filterIn": "(( not (src net 192.168.50.207/21 and ipv4)))
                            and
                          (( dst net 192.168.50.207/21 and ipv4))",
            "filterOut": "(( src net 192.168.50.207/21 and ipv4))
                              and
                            (( not (dst net 192.168.50.207/21 and ipv4)))",
            "profileName": "ddosdefender/druhy-10097e1",
         "rule": {
            "id": 1,
            "name": "Default rule"
         },
         "timeout": 120,
         "delay": 1,
         "autoSubnet": null,
         "autoSubnetIpV6": null,
         "bandwidthDay": null,
         "trafficSeen": "2017-11-29 13:24:30",
         "status": null,
         "bandwidth": 0,
         "bandwidthAuto": 0,
         "actionSubnet": {
            "id": "subnets",
            "name": "Subnets"
         },
         "type": {
            "id": 1,
            "name": "Subnets"
         },
         "asNumbers": [],
         "subnets": [
            "192.168.50.207/21"
         ],
         "preferredSubnets": [
```

```
            "12.15.16.10/32",
            "12.15.16.11/32"
        ],
        "parentProfile": {
            "profile": "icmp",
            "channels": [
                "*"
            ]
        },
        "measures": {
            "mode": "auto",
            "alert": null,
            "reroute": [
                {
                    "routerId": 6,
                    "routerName": "First",
                    "community": null,
                    "bgpMode": null
                }
            ],
            "mitigate": {
                "scrubbingCenter": {
                    "id": 3,
                    "name": "scrub c to faile"
                },
                "parameters": [
                    {
                        "option": "dns_a_in_ipv4",
                        "value": "72"
                    },
                    {
                        "option": "dns_a_in_ipv6",
                        "value": "72"
                    }

                    ...

                    {
                        "option": "con_limit_udp",
                        "value": "30000"
                    }
                ]
            }
        }
    }
```

11.2.1.3 Add new Segment

A method for adding new Segment. A new Segment must have without ID (i.e. must be empty or unused).

POST /rest/iad/segments/

- Return new Segment
- @return 201 Segment (11)

```
POST /rest/iad/segments/
BODY:
  {
        "entity":
      {
          "name": "Segment by API",
          "rule": {
              "id": 4,
              "name": "Baseline"
          },
          "timeout": 120,
          "delay": 1,
          "autoSubnet": null,
          "autoSubnetIpV6": null,
          "bandwidthDay": null,
          "status": null,
          "bandwidth": 0,
          "bandwidthAuto": 0,
          "actionSubnet": "automatic",
          "type": 1,
          "asNumbers": [],
          "subnets": [
              "12.15.16.32/21"
          ],
          "preferredSubnets": [
              "12.15.16.10/32",
              "12.15.16.11/32"
          ],
          "parentProfile": {
              "profile": "icmp",
              "channels": [
                  "othericmp",
```

```
                "timeexceeded",
                "echoreply"
            ]
        },
        "measures": {
            "mode": "manual",
            "alert": null,
            "reroute": null,
            "mitigate": null
        }
    }
}

Returns 201
   {
       "id": 22,
       "name": "Segment by API",
       "filterIn": "(( not (src net 12.15.16.32/21 and ipv4))) and (( dst net 12.15.16.32/21 and ipv4))",
       "filterOut": "(( src net 12.15.16.32/21 and ipv4)) and (( not (dst net 12.15.16.32/21 and ipv4)))",
       "profileName": "ddosdefender/Segmen-f9667f",
       "rule": {
           "id": 4,
           "name": "Baseline"
       },
       "timeout": 120,
       "delay": 1,
       "autoSubnet": null,
       "autoSubnetIpV6": null,
       "bandwidthDay": null,
       "trafficSeen": null,
       "status": null,
       "bandwidth": 0,
       "bandwidthAuto": 0,
       "actionSubnet": {
           "id": "automatic",
           "name": "Autodetected subnets"
       },
       "type": {
           "id": 1,
           "name": "Subnets"
       },
       "asNumbers": [],
       "subnets": [
           "12.15.16.32/21"
       ],
       "preferredSubnets": [
           "12.15.16.10/32",
           "12.15.16.11/32"
```

```
        ],
        "parentProfile": {
            "profile": "icmp",
            "channels": [
                "othericmp",
                "timeexceeded",
                "echoreply"
            ]
        },
        "measures": {
            "mode": "manual",
            "alert": null,
            "reroute": null,
            "mitigate": {
                "scrubbingCenter": null,
                "parameters": null
            }
        }
    }
```

11.2.1.4 Update existing Segment

A method for updating a Segment.

PUT /rest/iad/segments/

- Return updated Segment
- @return 200 Segment (11)

```
PUT /rest/iad/segments/
BODY:
  {
        "entity":
      {
          "id": 22,
          "name": "Update by API",
          "rule": {
              "id": 4,
              "name": "Baseline"
          },
```

```
        "timeout": 120,
        "delay": 1,
        "autoSubnet": null,
        "autoSubnetIpV6": null,
        "bandwidthDay": null,
        "trafficSeen": "2017-11-27 14:12:00",
        "status": null,
        "bandwidth": 0,
        "bandwidthAuto": 0,
        "actionSubnet": {
            "id": "subnets",
            "name": "Subnets"
        },
        "type": {
            "id": 1,
            "name": "Subnets"
        },
        "asNumbers": [],
        "subnets": [
            "200.3.33.14/32",
            "201.1.1.15/32"
        ],
        "preferredSubnets": null,
        "parentProfile": {
            "profile": "live",
            "channels": "*"
        },
        "measures": {
            "mode": "auto",
            "alert": {
                "id": 31,
                "name": "New Alert by API"
            },
            "reroute": [
                {
                    "routerId": 6,
                    "routerName": "First",
                    "community": "test",
                    "bgpMode": null
                }
            ],
            "mitigate": {
                "scrubbingCenter": {
                    "id": 1,
                    "name": "Moje AbsoluteVision3"
                },
                "parameters": []
            }
```

```
        }
    }
}

Returns 200
{
    "id": 22,
    "name": "Update by API",
    "filterIn": "(( not (src net 200.3.33.14/32 and ipv4)) and ( not (src net 201.1.1.15/32 and ipv4)))
                  and
                (( dst net 200.3.33.14/32 and ipv4) or ( dst net 201.1.1.15/32 and ipv4))",
    "filterOut": "(( src net 200.3.33.14/32 and ipv4) or ( src net 201.1.1.15/32 and ipv4))
                    and
                  (( not (dst net 200.3.33.14/32 and ipv4)) and ( not (dst net 201.1.1.15/32 and ipv4)))",
    "profileName": "ddosdefender/Update-195c02",
    "rule": {
        "id": 4,
        "name": "Baseline"
    },
    "timeout": 120,
    "delay": 1,
    "autoSubnet": null,
    "autoSubnetIpV6": null,
    "bandwidthDay": null,
    "trafficSeen": "2017-11-27 14:12:00",
    "status": null,
    "bandwidth": 0,
    "bandwidthAuto": 0,
    "actionSubnet": {
           "id": "subnets",
           "name": "Subnets"
        },
        "type": {
            "id": 1,
            "name": "Subnets"
        },
    "asNumbers": [],
    "subnets": [
        "200.3.33.14/32",
        "201.1.1.15/32"
    ],
    "preferredSubnets": null,
    "parentProfile": {
        "profile": "live",
        "channels": "*"
    },
    "measures": {
        "mode": "auto",
```

```
        "alert": {
            "id": 31,
            "name": "New Alert by API"
        },
        "reroute": [
            {
                "routerId": 6,
                "routerName": "First",
                "community": "test",
                "bgpMode": null
            }
        ],
        "mitigate": {
            "scrubbingCenter": {
                "id": 1,
                "name": "Moje AbsoluteVision3"
            },
            "parameters": [
                {
                    "option": "dns_a_in_ipv4",
                    "value": "72"
                },
                {
                    "option": "dns_a_in_ipv6",
                    "value": "72"
                }

                ...

                {
                    "option": "con_limit_udp",
                    "value": "30000"
                }
            ]
        }
    }
}
```


11.2.1.5 Delete specific Segment

Delete a specific Segment by its id. Return code is 204 (No data).

DELETE /rest/iad/segments/<id>

- Return empty answer
- @return 204 No data

```
DELETE /rest/iad/segments/22

Returns 204
```

12 REST API changes

Version 4.0

- added Attacks methods (3)
- added Alerts methods (4)
- added Routers methods (**??**)
- added Scrubbing Center methods (**??**)
- added Report Chapters methods (8)
- added EmailTempaltes methods (9)
- added Rules methods (10)
- added Segments methods (11)

Contacts

FLOWMON NETWORKS a.s.
Sochorova 3232/34
Brno 61600

Web: `www.flowmon.com`
Email:`info@flowmon.com`
Tel.: +420 530 510 600

Feedback

We would be pleased if you tell us your comments to this text (typing errors, incomplete or unclear information). Please, contact us via email `support@flowmon.com`.

Copyright