

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "title": "opentelemetry config",
  "properties": {
    "max_length_grpc_log": {
      "description": "max dump length of a otl grpc object",
      "type": "integer",
      "min": -1
    },
    "grpc_json_log": {
      "description": "true if we log otl grpc object to json format",
      "type": "boolean"
    },
    "second_fifo_expiry": {
      "description": "lifetime of data points in fifos",
      "type": "integer",
      "min": 30
    },
    "max_fifo_size": {
      "description": "max number of data points in fifos",
      "type": "integer",
      "min": 1
    },
    "otel_server": {
      "description": "otel grpc config",
      "type": "object"
    },
    "telegraf_conf_server": {
      "description": "http(s) telegraf config server",
      "type": "object"
    },
    "centreon_agent": {
      "description": "config of centreon_agent",
      "type": "object"
    }
  },
  "type": "object"
}
```

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "title": "telegraf config",
  "properties": {
    "http_server": {
      "listen_address": {
        "description": "IP",
        "type": "string",
        "minLength": 5
      },
      "port": {
        "description": "port to listen",
        "type": "integer",
        "minimum": 80,
        "maximum": 65535
      },
      "encryption": {
        "description": "true if https",
        "type": "boolean"
      },
      "keepalive_interval": {
        "description": "delay between 2 keepalive tcp packet, 0 no keepalive packets",
        "type": "integer",
        "minimum": 0,
        "maximum": 3600
      },
      "certificate_path": {
        "description": "path of the certificate file of the server",
        "type": "string",
        "minLength": 5
      },
      "key_path": {
        "description": "path of the key file",
        "type": "string",
        "minLength": 5
      }
    },
    "check_interval": {
      "description": "interval in seconds between two checks (param [agent] interval) ",
      "type": "integer",
      "minimum": 10
    },
    "engine_otel_endpoint": {
      "description": "opentelemetry engine grpc server",
      "type": "string",
      "minLength": 5
    }
  },
  "required": [
    "engine_otel_endpoint"
  ],
  "type": "object"
}
```

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "title": "centreon agent config",
  "properties": {
    "check_interval": {
      "description": "interval in seconds between two checks",
      "type": "integer",
      "minimum": 10
    },
    "max_concurrent_checks": {
      "description": "maximum of running checks at the same time",
      "type": "integer",
      "minimum": 1
    },
    "export_period": {
      "description": "period in second of agent metric export",
      "type": "integer",
      "minimum": 10
    },
    "check_timeout": {
      "description": "check running timeout",
      "type": "integer",
      "minimum": 1
    },
    "reverse_connections": {
      "description": "array of agent endpoints (reverse mode, engine connects to centreon-agent) ",
      "type": "array",
      "items": {
        "type": "object"
      }
    }
  },
  "type": "object"
}
```

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "title": "grpc config",
  "properties": {
    "host": {
      "description": "IP or dns name",
      "type": "string",
      "minLength": 5
    },
    "port": {
      "description": "port to listen",
      "type": "integer",
      "minimum": 1024,
      "maximum": 65535
    },
    "encryption": {
      "description": "true if https",
      "type": "boolean"
    },
    "public_cert": {
      "description": "path of certificate file .crt",
      "type": "string"
    },
    "private_key": {
      "description": "path of certificate file .key",
      "type": "string"
    },
    "ca_certificate": {
      "description": "path of authority certificate file .crt",
      "type": "string"
    },
    "ca_name": {
      "description": "name of authority certificate",
      "type": "string"
    },
    "compression": {
      "description": "activate compression",
      "type": "boolean"
    },
    "keepalive_interval": {
      "description": "delay between 2 keepalive tcp packet",
      "type": "integer",
      "minimum": -1,
      "maximum": 3600
    }
  },
  "required": [
    "host",
    "port"
  ],
  "type": "object"
}
```