



Netify Agent JSON Interface v1.90

Consuming JSON Structures Generated by the Netify Agent

Overview

The Netify Agent can be configured to generate two distinct JSON sources:

1. Periodic application-wide status updates

The Agent can be configured to output it's current state, including flow and DPI metadata. These JSON payloads are usually HTTPS POSTed to a "sink server", however the Agent can also be configured to save this data to a file on every update.

See: *json_save* in *netifyd.conf(5)*.

Default JSON output location: `/var/run/netifyd/sink-request.json`

2. Real-time status, flow, and DPI metadata

This data stream is available via a TCP/IP or UNIX socket when enabled. Clients can connect to this socket in a read-only manner to receive near real-time updates. This is a read-only socket and any data written to it will result in the connection being terminated. Flow and DPI metadata of a new flow will be available after 8-10 packets or less.

See: *[socket]* in *netifyd.conf(5)*.

Application-wide Status Updates

When enabled (see *enable_sink* in *netifyd.conf(5)*), the entire state of the Agent plus the active flow and DPI metadata for each configured interface is encoded and POSTed via HTTPS to either the default or designated URL. These payloads contain the complete set of an Agent's status since the previous update (the default update interval is 15 seconds).

Optionally, when configured (via *json_save*) this same payload will be written to a local file for other applications to process.

Real-time Flow Data

For real-time Agent status and flow/DPI detection data, applications can make use of the Agent Socket Interface (TCP/IP and UNIX sockets). The Agent will, in near real-time, send JSON

encoded payloads to any connected readers. This is the fastest way to react to specific DPI classifications. If, for example, the administrator wishes to log, block, or QoS-prioritize specific protocol/application detections, a reader application can be written to read and react to such events from an Agent.

See the following GitLab project, Netify Console, for an example (PHP) application that reads and displays real-time flow data from an Agent:

<https://gitlab.com/netify.ai/public/netify-console>

See the following GitLab project, Netify FWA (Firewall Agent), for an example (Python) application that processes real-time DPI detections from an Agent and applies basic firewall blocking according to a configurable ruleset.

<https://gitlab.com/netify.ai/public/netify-fwa>

Appendix

Structural examples of the various JSON payloads can be found here. The structures should be self-documenting but if any further explanation or details are desired, please feel free to contact team@netify.ai and we can provide additional information.

Example JSON Structure of Application-wide Status Updates

```
{
  "version": 1.9,
  "timestamp": 1574357103,
  "uptime": 603535,
  "maxrss_kb": 91324,
  "tcm_kb": 37655,
  "interfaces": {
    "eth0": {
      "role": "WAN",
      "mac": "00:90:fb:26:27:66"
    },
    "eth1": {
      "role": "WAN",
      "mac": "00:90:fb:26:27:68"
    },
    "eth2": {
      "role": "WAN",
```

```
    "mac": "00:90:fb:26:27:6a"
  },
  "eth3": {
    "role": "LAN",
    "mac": "00:90:fb:26:27:6b"
  }
},
"devices": {
  "00:90:fb:26:27:6b": [
    "192.168.242.1"
  ],
  "bc:ae:c5:8a:40:c5": [
    "192.168.242.3"
  ],
  "88:b6:ee:0e:ea:d7": [
    "192.168.242.63"
  ],
  "e4:f0:42:4d:df:93": [
    "192.168.242.39"
  ],
  "a4:77:33:cf:eb:c1": [
    "192.168.242.38"
  ],
  "00:e0:4c:36:aa:86": [
    "192.168.242.35"
  ],
  "20:df:b9:b4:ba:b1": [
    "192.168.242.41"
  ],
  "00:3e:e1:c0:2a:ba": [
    "192.168.242.14"
  ],
  "38:8b:59:23:16:54": [
    "192.168.242.33"
  ],
  "7c:2e:bd:2f:7a:b4": [
    "192.168.242.32"
  ],
  "40:4e:36:e6:db:8f": [
    "192.168.242.57"
  ],
  "e4:f0:42:5d:dd:3a": [
```

```
    "192.168.242.40"
  ]
},
"stats": {
  "eth0": {
    "raw": 260,
    "ethernet": 260,
    "mpls": 0,
    "pppoe": 74,
    "vlan": 0,
    "fragmented": 0,
    "discarded": 16,
    "discarded_bytes": 930,
    "largest_bytes": 1474,
    "ip": 244,
    "tcp": 34,
    "udp": 210,
    "icmp": 0,
    "igmp": 0,
    "ip_bytes": 68696,
    "wire_bytes": 74552,
    "pcap_recv": 260,
    "pcap_drop": 0,
    "pcap_ifdrop": 0,
    "queue_dropped": 0
  },
  "eth1": {
    "raw": 18874,
    "ethernet": 18874,
    "mpls": 0,
    "pppoe": 0,
    "vlan": 0,
    "fragmented": 0,
    "discarded": 1077,
    "discarded_bytes": 64620,
    "largest_bytes": 26130,
    "ip": 17797,
    "tcp": 17431,
    "udp": 362,
    "icmp": 4,
    "igmp": 0,
    "ip_bytes": 23634871,
  }
}
```

```
    "wire_bytes": 24061999,
    "pcap_recv": 18874,
    "pcap_drop": 0,
    "pcap_ifdrop": 0,
    "queue_dropped": 0
  },
  "eth2": {
    "raw": 47,
    "ethernet": 47,
    "mpls": 0,
    "pppoe": 47,
    "vlan": 0,
    "fragmented": 0,
    "discarded": 2,
    "discarded_bytes": 90,
    "largest_bytes": 572,
    "ip": 45,
    "tcp": 20,
    "udp": 23,
    "icmp": 2,
    "igmp": 0,
    "ip_bytes": 7207,
    "wire_bytes": 8287,
    "pcap_recv": 47,
    "pcap_drop": 0,
    "pcap_ifdrop": 0,
    "queue_dropped": 0
  },
  "eth3": {
    "raw": 26025,
    "ethernet": 26025,
    "mpls": 0,
    "pppoe": 0,
    "vlan": 0,
    "fragmented": 0,
    "discarded": 37,
    "discarded_bytes": 2022,
    "largest_bytes": 1514,
    "ip": 25988,
    "tcp": 25204,
    "udp": 722,
    "icmp": 62,
```

```
"igmp": 0,
"ip_bytes": 24223339,
"wire_bytes": 24847051,
"pcap_recv": 26025,
"pcap_drop": 0,
"pcap_ifdrop": 0,
"queue_dropped": 0
}
},
"flows": {
  "eth0": [
    {
      "digest": "55a66d117e08521800bddc3347e1ee6d86b1fd0c",
      "ip_nat": false,
      "ip_version": 4,
      "ip_protocol": 17,
      "vlan_id": 0,
      "other_type": "multicast",
      "local_origin": true,
      "local_mac": "00:3e:e1:c0:2a:ba",
      "other_mac": "01:00:5e:7f:ff:fa",
      "local_ip": "192.168.242.14",
      "other_ip": "239.255.255.250",
      "local_port": 58086,
      "other_port": 1900,
      "local_bytes": 648,
      "other_bytes": 0,
      "local_packets": 3,
      "other_packets": 0,
      "total_packets": 3,
      "total_bytes": 648,
      "detected_protocol": 12,
      "detected_protocol_name": "SSDP",
      "detected_application": 0,
      "detected_application_name": "Unknown",
      "detection_guessed": 0,
      "first_seen_at": 1574357100716,
      "first_update_at": 1574357100716,
      "last_seen_at": 1574357102718
    },
    {
      "digest": "bb1856f73208a6dca13d241c555d33fbf74341ff",
```

```
"ip_nat": false,
"ip_version": 4,
"ip_protocol": 17,
"vlan_id": 0,
"other_type": "multicast",
"local_origin": true,
"local_mac": "7c:2e:bd:2f:7a:b4",
"other_mac": "01:00:5e:00:00:fb",
"local_ip": "192.168.242.32",
"other_ip": "224.0.0.251",
"local_port": 5353,
"other_port": 5353,
"local_bytes": 427,
"other_bytes": 0,
"local_packets": 1,
"other_packets": 0,
"total_packets": 1,
"total_bytes": 427,
"detected_protocol": 8,
"detected_protocol_name": "MDNS",
"detected_application": 586,
"detected_application_name": "10036.netify.internal-network",
"detection_guessed": 0,
"mdns": {
  "answer": "_googlecast._tcp.local"
},
"first_seen_at": 1574357091462,
"first_update_at": 1574357091462,
"last_seen_at": 1574357091462
},
{
  "digest": "9a8b8685182adba57115e71eb18d0c3e376fafe7",
  "ip_nat": true,
  "ip_version": 4,
  "ip_protocol": 6,
  "vlan_id": 0,
  "other_type": "remote",
  "local_origin": true,
  "local_mac": "00:90:fb:26:27:66",
  "other_mac": "0c:a4:02:48:2d:4d",
  "local_ip": "127.255.0.2",
  "other_ip": "172.217.10.14",
```

```
"local_port": 54630,
"other_port": 443,
"local_bytes": 2612,
"other_bytes": 2939,
"local_packets": 15,
"other_packets": 12,
"total_packets": 27,
"total_bytes": 5551,
"detected_protocol": 196,
"detected_protocol_name": "HTTPS",
"detected_application": 294,
"detected_application_name": "126.netify.google",
"detection_guessed": 0,
"ssl": {
  "version": "0x0303",
  "cipher_suite": "0x1303",
  "client_sni": "clients4.google.com",
  "client_ja3": "d5a82b7e86091b9fc3bd24e2c9ae2919",
  "server_ja3": "0736b4d1842f2fe89b288890613f4baf"
},
"first_seen_at": 1574357088846,
"first_update_at": 1574357088846,
"last_seen_at": 1574357089068
},
{
  "digest": "a172d5d284de22811523843d222d7c04ede6fc99",
  "ip_nat": true,
  "ip_version": 4,
  "ip_protocol": 6,
  "vlan_id": 0,
  "other_type": "remote",
  "local_origin": true,
  "other_mac": "0c:a4:02:48:2d:4d",
  "local_mac": "00:90:fb:26:27:66",
  "other_ip": "34.234.23.242",
  "local_ip": "127.255.0.2",
  "other_port": 443,
  "local_port": 45812,
  "other_bytes": 728,
  "local_bytes": 1654,
  "other_packets": 2,
  "local_packets": 3,
```

```

    "total_packets": 25,
    "total_bytes": 11320,
    "detected_protocol": 196,
    "detected_protocol_name": "HTTPS",
    "detected_application": 371,
    "detected_application_name": "10119.netify.adobe",
    "detection_guessed": 0,
    "ssl": {
      "version": "0x0303",
      "cipher_suite": "0x1301",
      "client_sni": "bellmedia.sc.omtrdc.net",
      "client_ja3": "c67e9dc27d283f1f89b4ebb4b4670c21",
      "server_ja3": "eb1d94daa7e0344597e756a1fb6e7054"
    },
    "first_seen_at": 1574357056854,
    "first_update_at": 1574357089373,
    "last_seen_at": 1574357089451
  }
]
}
}

```

Example JSON Structure of Real-time Flow Data

Length Header

Each real-time message is preceded by a “length” JSON message which informs the reader how many bytes the following JSON payload will be. JSON messages are always terminated with a trailing CR-LF pair. The reader should read one text line up to 1,024 characters in length to consume the length header. The next read should be exactly the number of bytes indicated by the value in the length header.

```

{"length": 607}

```

The subsequent JSON payload will always contain a *type* field which indicates the structure of the message.

Agent Hello, type: agent_hello

The first payload the Agent will write is an “agent_hello” containing a few version fields, such as:

```
{
  "type": "agent_hello",
  "build_version": "Netify Agent/2.97-master-1143-bc2cb48 (clearos; x86_64;
  contrack; netlink; dns-cache; plugins; tcmmalloc; ssl-no-verify; inotify)
  nDPI/2.9.0 JSON/1.90",
  "agent_version": 2.97,
  "json_version": 1.9
}
```

Agent Status, type: agent_status

Periodically the Agent will write an “agent_status” payload containing a few general runtime metrics. Not all fields are available for all platforms (*tcm_xxx* for example).

```
{
  "type": "agent_status",
  "timestamp": 1574357585,
  "uptime": 604015,
  "flows": 325,
  "flows_prev": 339,
  "maxrss_kb": 91324,
  "maxrss_kb_prev": 91324,
  "tcm_kb": 37423,
  "tcm_kb_prev": 37481,
  "dhc_status": true,
  "dhc_size": 334,
  "sink_status": true,
  "sink_queue_size_kb": 0,
  "sink_queue_max_size_kb": 2048,
  "sink_resp_code": 1
}
```

Active Protocol and Application Definitions

Upon each connection, the Agent will send its active protocol and application definitions. This payload may be sent again in the event that the Agent received an updated configuration from a sink server.

```

{
  "type": "protocols",
  "protocols": [
    {
      "id": 0,
      "tag": "Unknown"
    },
    {
      "id": 1,
      "tag": "FTP_CONTROL"
    },
    {
      "id": 2,
      "tag": "POP3"
    },
    {
      "id": 3,
      "tag": "SMTP"
    },
    ...
  ]
}

```

Flow Data (Metadata and DPI Detection), type: flow

The “flow” JSON payload matches the same structures seen in the “flows” section of the Application-wide Status structure. An example HTTP request would be similar to the following:

```

{
  "type": "flow",
  "interface": "ens34",
  "internal": false,
  "established": false,
  "flow": {
    "digest": "37f76f1a8cb84623bac5895f73a9547e6f2717b0",
    "ip_nat": true,
    "ip_version": 4,
    "ip_protocol": 6,
    "vlan_id": 0,
    "other_type": "remote",
    "local_origin": true,
    "local_mac": "00:90:fb:26:27:68",
    "other_mac": "00:01:5c:97:5c:46",
  }
}

```

```
"local_ip": "127.255.0.1",
"other_ip": "172.217.6.227",
"local_port": 52323,
"other_port": 80,
"detected_protocol": 7,
"detected_protocol_name": "HTTP",
"detected_application": 297,
"detected_application_name": "126.netify.google",
"detection_guessed": 0,
"host_server_name": "connectivitycheck.gstatic.com",
"http": {
  "user_agent": "Mozilla/5.0 (X11; Linux armv7l) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/76.0.3809.81 Safari/537.36 CrKey/1.42.180518",
  "url": "http://connectivitycheck.gstatic.com/generate_204"
},
"first_seen_at": 1574357506135,
"first_update_at": 1574357506135,
"last_seen_at": 1574357521127
}
}
```