LOGZILLA DOCUMENTATION

Downstream Trap Receivers

Forward LogZilla events as SNMP traps to downstream trap receivers with custom OID mapping, trap_oid selection, and severity-based pre-match filters

Forwarding To Downstream Receivers · Generated May 3, 2026 · logzilla.ai/docs/forwarding-module/downstream-snmp-receivers

SNMP Trap

The SNMP TRAP forwarder allows forwarding of all or specific matched events to a downstream trap receiver. Deduplication occurs at ingest to reduce repeated events before forwarding.

For configuration detail on each section other than the actual `forwarders` section below, see Downstream Syslog Receivers (https://www.logzilla.ai/docs/forwarding-module/downstream-syslog-receivers).

Sample Forwarder Config File

```
window_size: 60
pre_match:
- field: message
  op: "~*"
  value: duplex mismatch discovered on
- field: severity
  value:
  - 1
  - 2
  - 5
forwarders:
- type: snmp
  target: 10.10.1.200:162
  trap_oid: 1.3.6.1.4.1.2021.991
  oid_prefix: 1.3.6.1.4.1.9.9.41.1.2.3
  oid_map:
  - type: s
    oid: ".1.2.0"
    src: facility
  - type: i
    oid: ".1.3.0"
    src: severity
  - type: s
    oid: ".1.4.0"
    src: cisco_mnemonic
  - type: s
    oid: ".1.5.0"
    src: message
  - type: i
    oid: ".1.99.0"
    src: counter
```

This forwarder sends the specified SNMP Trap for every matching event after deduplication.

Place the configuration file (for example, `fwd-snmp.yaml`) in `/etc/logzilla/forwarder.d/`, then verify and reload:

```
logzilla forwarder print
logzilla forwarder reload
```

NOTE: OIDs can defined here based on the needs, LogZilla does not limit which OIDs you are permitted to send.

trap_oid

Used to set the type of outgoing SNMP trap. In the case of `1.3.6.1.4.1.2021.991`, it specifies that it is from the `UCD-SNMP-MIB`, Specifically, `NOTIFICATION-TEST-MIB`.

oid_prefix

The base OID for all subsequent fields in the `oid_map`.

oid_map

The list of variables to be added to the trap:

- type: Only `i` (32 bit integer) and `s` (string) are supported
- oid: Object id of this variable; if it starts with dot then it's prefixed with `oid_prefix`
- src: Name of event field from LogZilla placed in this variable
- value: If no `src` macro is defined, you may use this to add extra information to each outgoing event.

For example, in a Service Provider Network, this would allow the addition of a customer's BGP AS Number:

```
{ "type" : "i", "oid" : ".1.70.0", "value" : "64512" },
{ "type" : "s", "oid" : ".1.70.1", "value" : "BGP_AS_ID" }
```