LOGZILLA DOCUMENTATION

Kubernetes Ingest Module

LogZilla Ingest module Kubernetes manifest with syslog-ng, ParserModule, and HTTP receiver containers for BSD, RFC5424, JSON, and /incoming event intake

Kubernetes Deployment Overview · Generated April 27, 2026 · logzilla.ai/docs/kubernetes-deployment/ingest-module

© LogZilla Corporation · logzilla.ai

Ingest module manifest

The Ingest module provides syslog receivers (TCP/UDP 514), JSON (TCP 515), RFC5424 (TCP 601), and an HTTP receiver for `/incoming`. It runs three containers: `syslogng`, `parsermodule`, and `httpreceiver`.

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: syslogng-conf
data:
  config.yaml: |
    config_version: '6.32'
    custom_conf_dir: /etc/logzilla/syslog-ng/conf.d/
    default_max_connections: 50
    default_log_iw_size: 1000000
    default_flush_lines: 100
    default_log_fetch_limit: 100

    extra_log_rules: ''
    flow_control: true

    destinations:
    - enabled: true
      type: logzilla-http
      ingest_token: replace_me
      url: http://localhost/
      mem_buf_length: 50000
      batch_timeout: 500
      batch_lines: 10000
      retries: 999
      send_timeout: 1000
      reply_timeout: 5000
      connect_timeout: 5000
      time_reopen: 1
      name: logzilla
      port: 11412
      reliable: false
      workers: 2
      buffer_dir: /var/lib/syslog-ng/
      disk_buf_size: 1048576
      disk_buff_enabled: false
      mem_buf_size: 4194304
      qout_size: 10000

    sources:
    - enabled: true
      flags: [syslog-protocol]
      name: bsd
      port: 514
      type: network
```

```
    - enabled: true
      flags: [syslog-protocol]
      name: bsd_udp
      port: 514
      transport: udp
      type: network
    - enabled: true
      name: rfc5424
      port: 601
      type: syslog
    - enabled: true
      flags: [no-parse]
      name: json
      port: 515
      program_override: _JSON
      type: network
    - enabled: false
      flags: [syslog-protocol]
      name: tls
      port: 6514
      tls_cert_file: /etc/logzilla/syslog-ng/tls.crt
      tls_key_file: /etc/logzilla/syslog-ng/tls.key
      transport: tls
      type: network
---
apiVersion: apps/v1
kind: StatefulSet
metadata:
  name: ingest
spec:
  serviceName: ingest
  podManagementPolicy: Parallel
  replicas: 4
  selector:
    matchLabels:
      name: ingest
  template:
    metadata:
      labels:
        name: ingest
    spec:
      enableServiceLinks: false
      containers:
      - name: syslogng
        image: logzilla/syslogng:stable
        terminationMessagePolicy: FallbackToLogsOnError
        imagePullPolicy: Always
        command: ["/usr/local/bin/docker-entrypoint.sh"]
        args: ["--worker-threads=8", "--no-caps"]
        resources:
          requests:
            memory: "200Mi"
            cpu: "500m"
          limits:
```

```
            memory: "1Gi"
            cpu: "2000m"
        ports:
        - containerPort: 514
          protocol: TCP
          name: bsd-tcp
        - containerPort: 514
          protocol: UDP
          name: bsd-udp
        - containerPort: 515
          protocol: TCP
          name: json
        - containerPort: 601
          protocol: TCP
          name: rfc5424
        - containerPort: 8080
          protocol: TCP
          name: stats
        livenessProbe:
          exec:
            command: ["syslog-ng-ctl", "stats"]
          failureThreshold: 3
          initialDelaySeconds: 30
          periodSeconds: 10
          successThreshold: 1
          timeoutSeconds: 5
        readinessProbe:
          exec:
            command: ["syslog-ng-ctl", "stats"]
          failureThreshold: 3
          initialDelaySeconds: 5
          periodSeconds: 10
          successThreshold: 1
          timeoutSeconds: 1
        volumeMounts:
        - name: syslogng-conf
          mountPath: /etc/logzilla/syslog-ng/config.yaml
          subPath: config.yaml
          readOnly: true
      - name: parsermodule
        image: logzilla/runtime:stable
        imagePullPolicy: Always
        terminationMessagePolicy: FallbackToLogsOnError
        resources:
          requests:
            memory: "200Mi"
            cpu: "200m"
          limits:
            cpu: "1000m"
            memory: "500Mi"
        command: ["/usr/lib/logzilla/bin/module_run", "ParserModule"]
        ports:
        - containerPort: 11412
          name: zmq-ingest
```

```
      protocol: TCP
    - containerPort: 81
      name: module-api
      protocol: TCP
    livenessProbe:
      httpGet:
        path: /liveness
        port: module-api
        scheme: HTTP
      failureThreshold: 3
      initialDelaySeconds: 30
      periodSeconds: 10
      successThreshold: 1
      timeoutSeconds: 5
    readinessProbe:
      httpGet:
        path: /readiness
        port: module-api
        scheme: HTTP
      failureThreshold: 3
      initialDelaySeconds: 5
      periodSeconds: 10
      successThreshold: 1
      timeoutSeconds: 1
    envFrom:
      - configMapRef:
          name: log-multimap
      - configMapRef:
          name: module-multimap
      - secretRef:
          name: internal-api-secret
    env:
    - name: PARSER_WORKERS
      value: "8"
    - name: LOG_MAX_LEVEL
      value: INFO
    - name: DOWNLOAD_API_RULES
      value: "1"
    - name: LOG_LEVEL
      value: "INFO"
    - name: SINGLE_SM_MODE
      value: "1"
    - name: SM_INGEST_URLS
      value: http://storage-{{1-4}}.storage:81/ingest
  - name: httpreceiver
    image: logzilla/runtime:stable
    terminationMessagePolicy: FallbackToLogsOnError
    imagePullPolicy: Always
    command: ["/usr/lib/logzilla/bin/http_receiver"]
    resources:
      requests:
        memory: "250Mi"
        cpu: "250m"
      limits:
```

```
            memory: "1000Mi"
            cpu: "500m"
        ports:
        - containerPort: 80
          protocol: TCP
          name: http-recv-port
        livenessProbe:
          httpGet:
            path: /ping
            port: http-recv-port
            scheme: HTTP
          failureThreshold: 3
          initialDelaySeconds: 30
          periodSeconds: 10
          successThreshold: 1
          timeoutSeconds: 5
        readinessProbe:
          httpGet:
            path: /ping
            port: http-recv-port
            scheme: HTTP
          failureThreshold: 3
          initialDelaySeconds: 5
          periodSeconds: 10
          successThreshold: 1
          timeoutSeconds: 1
        env:
        - name: LZ_TARGET_BASE_URL
          value: http://localhost:81/
        - name: IA_BASE_URL
          value: http://api/api/
        - name: WORKERS_NUM
          value: "8"
        envFrom:
          - secretRef:
              name: internal-api-secret
          - secretRef:
              name: http-ingest-token-secret
      volumes:
      - name: syslogng-conf
        configMap:
          name: syslogng-conf
          items:
          - key: config.yaml
            path: config.yaml
---
apiVersion: v1
kind: Service
metadata:
  name: syslog
spec:
  clusterIP: None
  ports:
    - port: 514
```

```
      protocol: TCP
      targetPort: bsd-tcp
      name: bsd-tcp-svc
    - port: 514
      protocol: UDP
      targetPort: bsd-udp
      name: bsd-udp-svc
    - port: 515
      protocol: TCP
      targetPort: json
      name: json-svc
    - port: 601
      protocol: TCP
      targetPort: rfc5424
      name: rfc5424-svc
  type: ClusterIP
  selector:
    name: ingest
---
apiVersion: v1
kind: Service
metadata:
  name: httpreceiver
spec:
  clusterIP: None
  ports:
    - port: 80
      protocol: TCP
      targetPort: http-recv-port
      name: http-recv-port
  type: ClusterIP
  selector:
    name: ingest
```

Notes

- Replace `ingest_token` in `syslogng-conf` with a valid token.
- The syslog Service exposes TCP/UDP 514, JSON 515, RFC5424 601.
- `httpreceiver` exposes port 80 for `/incoming` HTTP ingest.