Available as of v1.2.0

The monitoring feature is now implemented with an addon and is disabled by default in new installations.

Users can enable/disable `rancher-monitoring` addon from the Harvester WebUI after installation.

Users can also enable/disable the `rancher-monitoring` addon in their Harvester installation by customizing the harvester-configuration file.

For Harvester clusters upgraded from version v1.1.x, the monitoring feature is converted to an addon automatically and kept enabled as before.

Dashboard Metrics

Harvester has provided a built-in monitoring integration using Prometheus. Monitoring is automatically enabled during the Harvester installations.

From the `Dashboard` page, users can view the cluster metrics and top 10 most used VM metrics respectively. Also, users can click the Grafana dashboard link to view more dashboards on the Grafana UI.

:::note

Only admin users are able to view the cluster dashboard metrics.

Additionally, Grafana is provided by `rancher-monitoring`, so the default admin password is: prom-operator

Reference: values.yaml :::

VM Detail Metrics

For VMs, you can view VM metrics by clicking on the `VM details page > VM Metrics`.

:::note

The current `Memory Usage` is calculated based on `(1 - free/total) * 100%`, not `(used/total) * 100%`.

:::

For example, in a Linux OS, the `free -h` command outputs the current memory statistics as follows

```
$ free -h
               total        used        free      shared  buff/cache   available
Mem:           7.7Gi       166Mi       4.6Gi       1.0Mi       2.9Gi       7.2Gi
Swap:             0B          0B          0B
```

The corresponding `Memory Usage` is `(1 - 4.6/7.7) * 100%`, roughly `40%`.

How to Configure Monitoring Settings

Available as of v1.0.2

Monitoring has several components that help to collect and aggregate metric data from all Nodes/Pods/VMs. The resources required for monitoring depend on your workloads and hardware resources. Harvester sets defaults based on general use cases, and you can change them accordingly.

Currently, `Resources Settings` can be configured for the following components:

- Prometheus
- Prometheus Node Exporter

From UI

On the **Advanced** page, you can view and change the resource settings as follows:

1. Go to the **Advanced** > **Addons** page and select the **rancher-monitoring** page.
2. From the **Prometheus** tab, change the resource requests and limits.
3. Select **Save** when finished configuring the settings for the **rancher-monitoring** addon. The **Monitoring** deployments restart within a few seconds. Please be aware that the reboot can take time to reload previous data.

:::note

The UI configuration is only visible when the **rancher-monitoring** addon is enabled.

:::

The most frequently used option is the memory setting:

- The `Requested Memory` is the minimum memory required by the `Monitoring` resource. The recommended value is about 5% to 10% of the system memory of one single management node. A value less than 500Mi will be denied.
- The `Memory Limit` is the maximum memory that can be allocated to a `Monitoring` resource. The recommended value is about 30% of the system's memory for one single management node. When the `Monitoring` reaches this threshold, it will automatically restart.

Depending on the available hardware resources and system loads, you may change the above settings accordingly.

:::note

If you have multiple management nodes with different hardware resources, please set the value of Prometheus based on the smaller one.

:::

:::caution

When an increasing number of VMs get deployed on one node, the `prometheus-node-exporter` pod might get killed due to OOM (out of memory). In that case, you should increase the value of `limits.memory`.

:::

From CLI

You can use the following `kubectl` command to change resource configurations for the `rancher-monitoring` addon: `kubectl edit addons.harvesterhci.io -n cattle-monitoring-system rancher-monitoring`.

The resource path and default values are as follows:

```
apiVersion: harvesterhci.io/v1beta1
kind: Addon
metadata:
  name: rancher-monitoring
  namespace: cattle-monitoring-system
spec:
  valuesContent: |
    prometheus:
      prometheusSpec:
        resources:
          limits:
            cpu: 1000m
            memory: 2500Mi
          requests:
            cpu: 850m
            memory: 1750Mi
```

:::note

You can still make configuration adjustments when the addon is disabled. However, these changes only take effect when you re-enable the addon.

:::

Alertmanager

`Harvester` uses `Alertmanager` to collect and manage all the alerts that happened/happening in the cluster.

Alertmanager Config

Enable/Disable Alertmanager

`Alertmanager` is enabled by default. You may disable it from the following config path.

Change Resource Setting

You can also change the resource settings of `Alertmanager` as shown in the picture above.

Configure AlertmanagerConfig from WebUI

To send the alerts to third-party servers, you need to config `AlertmanagerConfig`.

On the WebUI, navigate to `Monitoring & Logging` -> `Monitoring` -> `Alertmanager Configs`.

On the `Alertmanager Config: Create` page, click `Namespace` to select the target namespace from the drop-down list and set the `Name`. After this, click `Create` in the lower right corner.

Click the `Alertmanager Configs` you just created to continue the configuration.

Click `Add Receiver`.

Set the `Name` for the receiver. After this, select the receiver type, for example, `Webhook`, and click `Add Webhook`.

Fill in the required parameters and click `Create`.

To set up Microsoft Teams or SMS webhooks, first install the rancher-alerting-drivers app using the following commands:

```
helm repo add rancher-charts https://charts.rancher.io/
helm repo update
helm install rancher-charts/rancher-alerting-drivers \
  --set sachet.enabled=false \  # Set to true if you want to use SMS Webhook
  --set prom2teams.enabled=true \ # Set to true if you want to use MS Teams Webhook
  --namespace cattle-monitoring-system \
  --generate-name
```

For detailed configuration instructions, see Rancher Monitoring Receiver Configuration in the Rancher documentation.

If your environment does not have direct internet access (air-gapped), you must manually download the Helm chart and related container images, and then upload them to the Harvester cluster.

1. Download the rancher-alerting-drivers Helm chart and package it.

   ```
   helm pull rancher-charts/rancher-alerting-drivers --version <VERSION>
   ```

2. Download the required images.

   ```
   docker save -o sachet.tar rancher/mirrored-messagebird-sachet:<VERSION>
   docker save -o prom2teams.tar rancher/mirrored-idealista-prom2teams:<VERSION>
   ```

3. Upload the chart and images to the Harvester cluster.

4. Load the images on all Harvester nodes.

```
docker load -i sachet.tar
docker load -i prom2teams.tar
```

5. Install rancher-alerting-drivers on the Harvester cluster.

:::info important

Harvester does not manage upgrades of the rancher-alerting-drivers app, which is not part of the Harvester project. You must upgrade the app manually.

:::

Configure AlertmanagerConfig from CLI

You can also add `AlertmanagerConfig` from the CLI.

Exampe: a Webhook receiver in the `default` namespace.

```
cat << EOF > a-single-receiver.yaml
apiVersion: monitoring.coreos.com/v1alpha1
kind: AlertmanagerConfig
metadata:
  name: amc-example
  # namespace: your value
  labels:
    alertmanagerConfig: example
spec:
  route:
    continue: true
    groupBy:
    - cluster
    - alertname
    receiver: "amc-webhook-receiver"
  receivers:
  - name: "amc-webhook-receiver"
    webhookConfigs:
    - sendResolved: true
      url: "http://192.168.122.159:8090/"
EOF

# kubectl apply -f a-single-receiver.yaml
alertmanagerconfig.monitoring.coreos.com/amc-example created

# kubectl get alertmanagerconfig -A
NAMESPACE   NAME          AGE
default     amc-example   27s
```

Example of an Alert Received by Webhook

Alerts sent to the webhook server will be in the following format:

```
{
'receiver': 'longhorn-system-amc-example-amc-webhook-receiver',
'status': 'firing',
```

```
'alerts': [],
'groupLabels': {},
'commonLabels': {'alertname': 'LonghornVolumeStatusWarning', 'container': 'longhorn-
manager', 'endpoint': 'manager', 'instance': '10.52.0.83:9500', 'issue': 'Longhorn
volume is Degraded.',
'job': 'longhorn-backend', 'namespace': 'longhorn-system', 'node': 'harv2', 'pod':
'longhorn-manager-r5bgm', 'prometheus': 'cattle-monitoring-system/rancher-monitoring-
prometheus',
'service': 'longhorn-backend', 'severity': 'warning'},
'commonAnnotations': {'description': 'Longhorn volume is Degraded for more than 5
minutes.', 'runbook_url': 'https://longhorn.io/docs/1.3.0/monitoring/metrics/',
'summary': 'Longhorn volume is Degraded'},
'externalURL': 'https://192.168.122.200/api/v1/namespaces/cattle-monitoring-
system/services/http:rancher-monitoring-alertmanager:9093/proxy',
'version': '4',
'groupKey': '{}/{namespace="longhorn-system"}:{}',
'truncatedAlerts': 0
}
```


:::note

Different receivers may present the alerts in different formats. For details, please refer to the related documents.

:::

Known Limitation

The `AlertmanagerConfig` is enforced by the `namespace`. Gloabl-level `AlertmanagerConfig` without a namespace is not supported.

We have already created a GithHb issue to track upstream changes. Once the feature is available, `Harvester` will adopt it.

View and Manage Alerts

From Alertmanager Dashboard

You can visit the original dashboard of `Alertmanager` from the link below. Note that you need to replace `the-cluster-vip` with the actual cluster-vip:

> *https://the-cluster-vip/api/v1/namespaces/cattle-monitoring-system/services/http:rancher-monitoring-alertmanager:9093/proxy/#/alerts*

The overall view of the `Alertmanager` dashboard is as follows.

You can view the details of an alert:

From Prometheus Dashboard

You can visit the original dashboard of `Prometheus` from the link below. Note that you need to replace `the-cluster-vip` with the actual cluster-vip:

> *https://the-cluster-vip/api/v1/namespaces/cattle-monitoring-system/services/http:rancher-monitoring-prometheus:9090/proxy/*

The `Alerts` menu in the top navigation bar shows all defined rules in Prometheus. You can use the filters `Inactive`, `Pending`, and `Firing` to quickly find the information that you need.

Troubleshooting

For Monitoring support and troubleshooting, please refer to the troubleshooting page .