Federator.ai Release v4.5.1
Installation Guide
Overview

Federator.ai

ProphetStor Federator.ai is an AI-based solution that helps enterprises manage and optimize resources for applications on Kubernetes and virtual machines (VMs) in VMware clusters. Using advanced machine learning algorithms to predict application workloads, Federator.ai offers:

- AI-based workload prediction for containerized applications in Kubernetes clusters and VMs in VMware clusters
- Resource recommendations based on workload prediction, application, Kubernetes, and other related metrics
- Automatic scaling of application containers
- Multicloud cost analysis and recommendations based on workload predictions for Kubernetes clusters and VM clusters
- Actual cost and potential savings based on recommendations for clusters, Kubernetes applications, VMs, and Kubernetes namespaces

Supported Metrics Data Sources

There are four different types of metrics data sources supported in release 4.5.1: Prometheus, Datadog, Sysdig and VMware vCenter.

Prometheus (Kubernetes, Rancher, RedHat OpenShift)

Prometheus is a free and open-source event monitoring tool for containers or microservices. It uses the principle of scraping to collect numerical data based on time series. Metrics are collected in regular timestamps and stored locally. Federator.ai supports using Prometheus gathering Kubernetes cluster metrics, and leverage collected data for workload predictions, recommendations for resource planning, autoscaling containers/pods, and cost analysis for clusters deployed in a multicloud environment.

The following diagram shows how the metrics are collected from Prometheus by Federator.ai in a Kubernetes environment.
Datadog

Federator.ai has integrated with Datadog and utilizes the metrics collected by Datadog Agent for workload predictions. The following diagram shows how application metrics are used by Federator.ai to predict workload and to automatically scale applications for better performance. Specifically,

- Datadog Agent sends cluster/applications metrics to Datadog Services
- Federator.ai's Data-adapter queries cluster/applications metrics from Datadog Services and forwards to Federator.ai AI engine
- Data-adapter posts the prediction/recommendation/plan created by Federator.ai to Datadog Services
- Datadog Cluster Agent gets prediction/recommendation/plan from Datadog Services
- WPA applies plans and auto-scales applications
- Datadog Dashboard displays cluster/applications metrics and prediction/recommendation/plan by Federator.ai

![Datadog Diagram](image)

Sysdig

Federator.ai has integrated with Sysdig and utilizes the metrics collected by Sysdig Agent for workload predictions. The following diagram shows how application metrics are used by Federator.ai to predict workload and to automatically scale applications for better performance.
VMWare vCenter

VMware vCenter Server provides integrated management of all hosts and virtual machines in the data center from a single console, allowing IT administrators to improve control, simplify daily work, and reduce the complexity and cost of managing the IT environment. Federator.ai data adapter connects to VMware vCenter servers via VMware SDK to retrieve all of VMs workload metrics data for predictions, recommendations and cost analysis for VM clusters.
Requirements and Recommended Resource Configuration

Platforms
- OpenShift: 3.11, 4.x
- Kubernetes: 1.11 ~ 1.20
- Rancher v2.4.8
- EKS/AKS/GKE

Data Source
- Datadog
- Sysdig
- Prometheus (Kubernetes, OpenShift)
- VMWare vCenter 5.5/6.0/6.5/6.7/7.0

Federator.ai Resource Requirements
- Total Resource Requirements
  - Request: 5.1 CPU cores (Limit: 22 cores)
  - Request: 5.0 GB Memory (Limit: 28GB)
  - StorageClass: 168GB (require ReadWriteOnce access mode)

- Resource requirements for AI Engine
  - There must be at least one worker node with at least 2 CPU (Limit: 8 cores) cores and 1 GB of memory available
  - The 2 CPU cores and 1 GB memory are included in the total 5.1 CPU cores and 5.0 GB memory requirements

Federator.ai Version
- Version: Release v4.5.1
- Tag: v4.5.1-ga

Datadog Agent Version (reference)
- Datadog Agent helm chart version: v2.4.24
- Datadog Agent version: v7.21.1
- Datadog Cluster Agent version: v1.8.0
- Datadog Watermark Pod Autoscaler version: v0.1.0

Prometheus Version (reference)
- OpenShift
  - Default installed Prometheus
• Kubernetes
  - prometheus-operator-8.5.11
  - Rancher v2.4.8 kube-prometheus-stack-12.3.0
  - kube-prometheus-release-0.6
  - kube-prometheus-stack-12.5.0
  - kube-prometheus-stack-12.3.0

Sysdig Agent Version (reference)
• Sysdig agent: 10.8.0

Persistent Volumes
• The StorageClass that provides the persistent volumes must support RWO (ReadWriteOnce) access mode.
• It is recommended to use persistent volumes instead of using ephemeral storage to store the data in the production environment.

Kafka
• For Federator.ai’s application-aware Kafka consumer resource/performance optimization feature, the following version of Kafka is supported:

  Kafka operator version (Reference) : Strimzi/kafka:0.17.0-kafka-2.4.0
Federator.ai Installation and Configuration

Summary of Installation Steps

Step 0: Review pre-installation checklist items, make sure the environment and required information are ready.

Step 1: Preparation
- For Datadog, obtain API Key, Application Key of Datadog Cloud Service account. Instructions are provided below.
- For Prometheus, obtain Prometheus service URL (ex: http://<prometheus_svc_name>.<namespace>:9090)
- For Sysdig, obtain Sysdig API URL and Token.
- For VMware vCenter, obtain administrator login credential and vCenter IP or FQDN.

Step 2:
- For Datadog, install and configure Datadog Agent/Cluster Agent if they have not been installed. Please follow the Datadog documentation on how to install Datadog Agent and Cluster Agent.
- For Sysdig, install and configure Sysdig Agent. Please follow Sysdig documentation on how to install Sysdig Agent.

Step 3: Install Federator.ai.

Step 4: Configure Federator.ai Data Adapter for the external metrics data source via Federator.ai Initial Setup Wizard.

Step 5: Optionally install Datadog WPA and apply WPA autoscaling CR if using Datadog WPA for autoscaling.

Step 6: Review installation result on Datadog/Sysdig Cloud Dashboard.

Pre-installation Check List

Kubernetes:

<table>
<thead>
<tr>
<th>#</th>
<th>Checklist Item</th>
<th>Requirement</th>
<th>Details</th>
</tr>
</thead>
</table>
| 1  | What is the Kubernetes version?                     | 1.11~1.20   | Use the command below to get the Kubernetes version: $ kubectl version ...
| 2  | Does installation on this Kubernetes cluster require a private image repository? | If a private image repository is required, the following information is needed during installation | Input the URL and credential when the Federator.ai installation script asks for the information. |
### 3 StorageClass and Persistent Volumes requirement
- **StorageClass** supports ReadWriteOnce access mode. Available storage size is larger than 168GB.
- The minimum storage size for Federator.ai Release v4.5.1 is 168GB, including database, data, and logs.

### 4 Kubernetes cluster CPU/memory requirement
- **Minimum CPU/mem/storage:**
  - CPU: 5,100 (mcores)
  - Memory: 5.0 (GB)
  - Storage Class Capacity: 168GB
- **At least one worker node with:**
  - CPU: 2 Cores
  - Memory: 1GB
- To be able to run the AI Engine pod, there must be at least one worker node that has more than 2 CPU cores and 1 GB of memory available.
- 2 CPU Cores and 1GB for AI Engine are included in the total 5.1 CPU Cores and 5 GB memory requirements.

### 5 Is this Kubernetes cluster allowed for NodePort configuration?
- Federator.ai creates two NodePorts for GUI and REST API by default
  - REST API - https://<server>:31011
  - GUI - https://<server>:31012
- If NodePort is not allowed, answer ‘N’ when the installation script prompts for creating NodePorts.
- Users need to expose Federator.ai GUI and REST API service manually.

### 6 Will there be a resource quota imposed for the namespace where Federator.ai is installed?
- **CPU/mem request quota should be more than the minimum resource requirement**
  - CPU: 5.1 Cores
  - Memory: 5.0 GB
- The CPU/memory required for Federator.ai depends on the number of clusters and applications being monitored/managed.
- **Suggestion for initial namespace quota is**
  - CPU 8 cores
  - Memory 12GB
- The quota could be adjusted if the number of managed clusters/applications increases.
- **Use the command to get namespace resource quota**
  `kubectl get resourcequota --all-namespaces`

### 7 Does this deployment requires resource request/limit specified?
- By default, Federator.ai deployments do not specify resource requests/limits. It can be done by setting up an environment variable before installation starts.
- **To turn on resource request/limit settings for all Federator.ai deployments, manually export environment variable before running ‘federatorai-launcher.sh’**
  ```bash
  $ export ENABLE_RESOURCE_REQUIREMENT=y
  $ ./federatorai-launcher.sh
  ```

### Prometheus:

<table>
<thead>
<tr>
<th>#</th>
<th>Checklist Item</th>
<th>Requirement</th>
<th>Details</th>
</tr>
</thead>
</table>
| 1 | What is the Prometheus version? (for Kubernetes) | Recommended version-Prometheus operator helm chart version: 8.5.11-Prometheus operator version: 0.34.0-Prometheus server version: 2.13.1 | Use the command below to get Prometheus version: `helm ls -A |grep -i prometheus`
  prometheus-operator monitoring 1 2020-03-13 15:35:05.28963154 +0800 CST deployed
  prometheus-operator-2.1.3 v0.6.0 prometheus-operator monitoring 1 2020-03-13 14:34:16.132479221 +0800 CST deployed
  prometheus-operator-8.12.1 0.37.0 |
### Datadog Agent:

<table>
<thead>
<tr>
<th>#</th>
<th>Checklist Item</th>
<th>Requirement</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Is Datadog Agent installed?</td>
<td>Datadog Agent is mandatory</td>
<td>Kubernetes resources and workload metrics are collected by Datadog Agent.</td>
</tr>
<tr>
<td>2</td>
<td>Is Datadog Cluster Agent installed?</td>
<td>Cluster Agent is mandatory for the HPA autoscaling feature</td>
<td>Cluster Agent provides metrics to HPA Autoscaler for autoscaling.</td>
</tr>
<tr>
<td>3</td>
<td>Is Datadog WPA controller installed?</td>
<td>Datadog WPA is required if autoscaling is done by WPA</td>
<td>Datadog WPA is the HPA Autoscaler developed by Datadog. Users can use Datadog WPA or Kubernetes native HPA to do autoscaling.</td>
</tr>
</tbody>
</table>
| 4  | Datadog Kafka Consumer integration is enabled?      | Datadog Kafka Consumer integration is mandatory if user wants to use Kafka optimization feature | Use the command to confirm Kafka integration is enabled:  
$ kubectl exec <datadog-agent-pod> -n <datadog-agent-namespace> -- agent integration show datadog-kafka-consumer  
Refer to [https://www.datadoghq.com/blog/monitor-kafka-with-datadog/](https://www.datadoghq.com/blog/monitor-kafka-with-datadog/) for Kafka Consumer integration installation |
| 5  | Datadog account API key                             | An API key is mandatory for connecting Datadog Service | Follow the steps described in the “Before You Start” session to obtain the API key. |
| 6  | Datadog account Application key                     | An application key is mandatory for connecting Datadog Service | Follow the steps described in the “Before You Start” session to obtain the Application key. |
| 7  | Is one of cluster name is configured for the Datadog agent/cluster agent?  
1.>DD_TAGS with value ="kube_cluster:<cluster_name >" in values.yaml or  
2.>"cluster_name" in values.yaml, or  
3.>"DD_CLUSTER_NAME" in Datadog cluster agent deployment | "kube_cluster","cluster_name","kube_cluster_name(DD_CLSUTER_NAME)" one of them is required for Federator.ai to identify Kubernetes cluster. | Case 1.>New Datadog Agent installation:  
Install Datadog agent and cluster agent by "helm install -f values.yaml", in values.yaml.  
...  
clusterName: <cluster-name>  
...  
clusterAgent:  
enabled: false true  

Case 2.> In Datadog Agent installed environment, with no Cluster Agent and no cluster_name setting  
Update Datadog Agent to enable Cluster agent by "helm upgrade -f values.yaml", in values.yaml  
- assign a cluster name  
...  
datadog:  
- clusterName: <cluster-name>  
- enable cluster agent  
...  
clusterAgent:  
- enabled: false true  
...  

<table>
<thead>
<tr>
<th>8</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
</table>
Before You Start (Datadog)

- The admin role for installing Federator.ai is "Cluster Admin."
- Datadog agent must be ready if Federator.ai runs in the same Kubernetes cluster that is being monitored.
- Obtain Datadog account API Key, Application Key.
  1. A Datadog account is required for connecting and using Datadog Cloud Service. If you don’t have an account, visit Datadog website and sign up for a free trial account. [https://www.datadoghq.com/](https://www.datadoghq.com/)
  2. Log in Datadog Cloud Service with your account and get an API Key and Application Key for using Datadog API
Copy the API Key and Application Key for Federator.ai metrics data source configuration

Before You Start (Sysdig)

- Different Sysdig API URL is needed for different regions:
  - For US East, Sysdig API URL is [https://app.sysdigcloud.com](https://app.sysdigcloud.com)
  - For US West, Sysdig API URL is [https://us2.app.sysdig.com](https://us2.app.sysdig.com)
  - For European Union, Sysdig API URL is [https://eu1.app.sysdig.com](https://eu1.app.sysdig.com)
- Copy Sysdig Monitor API Token for Federator.ai metrics data source configuration
Before You Start (VMware vCenter)

VM Cluster

- You can define a VM cluster from any VMs under the same cluster path. See below for an example of cluster path on vCenter.

```
*cluster_path:*

"Datacenter/cluster1"

"my-vm-cluster1"

"my-vm-cluster2"
```
New Installation

(For upgrading from previous Federator.ai, please refer to the next section.)

1. Log into Kubernetes cluster

2. Install the Federator.ai for Kubernetes by using the following command

   $ curl https://raw.githubusercontent.com/containers-ai/prophetstor/master/deploy/federatorai-launcher.sh | bash

~/# curl https://raw.githubusercontent.com/containers-ai/prophetstor/master/deploy/federatorai-launcher.sh | bash

100 17260  100 17260  0  0 2020 0:00:00  0:00:00  0:00:00  0:00:00

Please input Federator.ai version tag (e.g., v4.4.1): v4.5.1-ga

Please input Federator.ai files save path [default: /opt]:

Downloading v4.5.1-ga tgz file ...
Done
Do you want to use a private repository URL? [default: n]:
Do you want to launch the Federator.ai installation script? [default: y]:

Executing install.sh ...
Checking environment version... ...Passed
Enter the namespace you want to install Federator.ai [default: federatorai]:

--------------------------------------------------------
tag_number = v4.5.1-ga
install_namespace = federatorai
--------------------------------------------------------
Is the above information correct? [default: y]:

Downloading v4.5.1-ga tgz file ...
Done

Applying Federator.ai operator yaml files...
Applying 00-namespace.yaml...
namespace/federatorai created
Applying 01-serviceaccount.yaml...
serviceaccount/federatorai-operator created
Applying 02-alamedaservice.crd.yaml...
customresourcedefinition.apiextensions.k8s.io/alamedaservices.federatorai.containers.ai created
Applying 03-federatorai-operator.deployment.yaml...
deployment.apps/federatorai-operator created
Applying 04-clusterrole.yaml...
clusterrole.rbac.authorization.k8s.io/federatorai-operator created
clusterrole.rbac.authorization.k8s.io/alameda-gc created
Applying 05-clusterrolebinding.yaml...
clusterrolebinding.rbac.authorization.k8s.io/federatorai-operator created
Applying 06-role.yaml...
role.rbac.authorization.k8s.io/federatorai-operator created
Applying 07-rolebinding.yaml...
rolebinding.rbac.authorization.k8s.io/federatorai-operator created
Applying 08-service.yaml...
  service/federator.ai-operator-service created
Applying 09-secret.yaml...
  secret/federator.ai-operator-service-cert created
Applying 10-mutatingwebhook.yaml...
  mutatingwebhookconfiguration.admissionregistration.k8s.io/federator.ai-operator-servicesmutation created
Applying 11-validatingwebhook.yaml...
  validatingwebhookconfiguration.admissionregistration.k8s.io/federator.ai-operator-servicesvalidation created

Checking pods...
Waiting for pod federator.ai-operator-747fb5558-8fbgm in namespace federator.ai to be ready. phase: [Running]
Waiting for pods in namespace federator.ai to be ready...
Waiting for pod federator.ai-operator-747fb5558-8fbgm in namespace federator.ai to be ready. phase: [Running]
Waiting for pods in namespace federator.ai to be ready...

All pods under namespace(federator.ai) are ready.

Install Federator.ai operator v4.5.1-ga successfully

Downloading Federator.ai alamedaservice sample file ... done

Downloading Federator.ai alamedascaler sample files ... done

Which storage type you would like to use? ephemeral or persistent? [default: persistent]:
Specify log storage size [e.g., 2 for 2GB, default: 2]:
Specify AI engine storage size [e.g., 10 for 10GB, default: 10]:
Specify InfluxDB storage size [e.g., 100 for 100GB, default: 100]:
Specify storage class name: managed-nfs-storage
Do you want to expose dashboard and REST API services for external access? [default: y]:

install_namespace = federator.ai
storage_type = persistent
log storage size = 2 GB
AI engine storage size = 10 GB
InfluxDB storage size = 100 GB
storage class name = managed-nfs-storage
expose service = y

Is the above information correct [default: y]:
Processing...
Waiting for datahub(v4.5.1-ga) pod to appear ...

datahub pod is present.

Checking pods...
Waiting for pod alameda-ai-69864dd766-kkjgg in namespace federator.ai to be ready. phase: [Pending]
Waiting for pods in namespace federator.ai to be ready...
Waiting for pod alameda-ai-69864dd766-kkjgg in namespace federator.ai to be ready. phase: [Pending]
Waiting for pods in namespace federatorai to be ready...
Waiting for pod alameda-ai-69864dd766-kkjgg in namespace federatorai to be ready. phase: [Pending]
Waiting for pods in namespace federatorai to be ready...
Waiting for pod alameda-ai-69864dd766-kkjgg in namespace federatorai to be ready. phase: [Pending]
Waiting for pods in namespace federatorai to be ready...
Waiting for pod alameda-ai-69864dd766-kkjgg in namespace federatorai to be ready. phase: [Running]
Waiting for pods in namespace federatorai to be ready...
All pods under namespace(federatorai) are ready.
The default alamedaorganization under namespace federatorai is ready.

You can now access GUI through https://<YOUR IP>:31012
The default login credential is admin/admin
Also, you can start to apply alamedascaler CR for the target you would like to monitor. Review the administration guide for further details.

You can now access Federatorai REST API through https://<YOUR IP>:31011
The default login credential is admin/admin
The REST API online document can be found in https://<YOUR IP>:31011/apis/v1/swagger/index.html

Install Federator.ai v4.5.1-ga successfully
Downloaded YAML files are located under /opt/federatorai/installation
Downloaded files are located under /opt/federatorai/repo/v4.5.1-ga

3. Verify Federator.ai pods are running properly

<table>
<thead>
<tr>
<th>NAME</th>
<th>READY</th>
<th>STATUS</th>
<th>RESTARTS</th>
<th>AGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>alameda-ai-5dc4b9dc5f-kpcn6</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>5m34s</td>
</tr>
<tr>
<td>alameda-ai-dispatcher-5b5dcbc8f7-txwgs</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>5m34s</td>
</tr>
<tr>
<td>alameda-analyzer-5b9677dbb6-lvcv6</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>5m34s</td>
</tr>
<tr>
<td>alameda-datahub-7dcdcbcc8-vzrhg</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>5m34s</td>
</tr>
<tr>
<td>alameda-executor-8645fd8f94-xmm6f</td>
<td>1/1</td>
<td>Running</td>
<td>4</td>
<td>5m33s</td>
</tr>
<tr>
<td>alameda-influxdb-0</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>5m33s</td>
</tr>
<tr>
<td>alameda-notifier-9d5774447-hrgrnw</td>
<td>1/1</td>
<td>Running</td>
<td>4</td>
<td>5m34s</td>
</tr>
<tr>
<td>alameda-operator-57dd69f46c-tid2m9</td>
<td>1/1</td>
<td>Running</td>
<td>4</td>
<td>5m34s</td>
</tr>
<tr>
<td>alameda-rabbitmq-5f8446fd5f-4rkfg</td>
<td>1/1</td>
<td>Running</td>
<td>4</td>
<td>5m34s</td>
</tr>
<tr>
<td>fedemeter-api-7c88976bbf-2h6sh</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>5m34s</td>
</tr>
<tr>
<td>fedemeter-influxdb-0</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>5m33s</td>
</tr>
<tr>
<td>federatorai-agent-867544c5f7-2hh9r</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>5m34s</td>
</tr>
<tr>
<td>federatorai-dashboard-backend-6bd97b7744-s659w</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>5m33s</td>
</tr>
<tr>
<td>federatorai-dashboard-frontend-85f665cd96-52c2h</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>5m33s</td>
</tr>
<tr>
<td>federatorai-data-adapter-6dcb878676-xh2pn</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>5m33s</td>
</tr>
<tr>
<td>federatorai-operator-7f7c468999d-hrls6</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>5m33s</td>
</tr>
<tr>
<td>federatorai-recommender-dispatcher-77996f7586-bthl7</td>
<td>1/1</td>
<td>Running</td>
<td>1</td>
<td>5m32s</td>
</tr>
<tr>
<td>federatorai-recommender-worker-6f74fc6fc-tsgkl</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>5m32s</td>
</tr>
</tbody>
</table>
4. Log on Federator.ai GUI and finish installation through the Initial Setup Wizard. For more information on Initial Setup Wizard, please see Federator.ai 4.5.1 User Guide.

https://<master_node_IP>:31012
Login ID: admin
Password: admin
Upgrade from a previous version

Federator.ai v4.5.1 supports upgrade from previous versions. The Federator.ai installation script automatically detects previously installed Federator.ai. When the installation script prompts if a backup of the previous configuration is needed, just enter yes to save a copy of the configuration if a rollback to the previous version is needed.

Prerequisite

1. Federator.ai version is 4.2 or later.
2. Federator.ai installed and running with Persistent Volume.

Upgrade

1. Log into Kubernetes cluster

2. Install the Federator.ai for Kubernetes by using the following command

```
$ curl https://raw.githubusercontent.com/containers-ai/prophetstor/master/deploy/federatorai-launcher.sh | bash
```

```
$ curl https://raw.githubusercontent.com/containers-ai/prophetstor/master/deploy/federatorai-launcher.sh | bash
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
Dload  Upload   Total   Spent    Left  Speed
100 16783  100 16783    0     0  25155      0 --:--:-- --:--:-- --:--:-- 25161
Please input Federator.ai version tag (e.g., v4.4.1): v4.5.1-ga
Please input Federator.ai files save path [default: /opt]:

Downloading v4.5.1-ga tgz file ...
Done
Do you want to use a private repository URL? [default: n]:
Do you want to launch the Federator.ai installation script? [default: y]:

Executing install.sh ...
Checking environment version...
...Passed
Previous build with tag v4.2.761 detected in namespace federatorai

----------------------------------------
Upgrade:
tag_number = v4.5.1-ga
install_namespace = federatorai
----------------------------------------
Is the above information correct? [default: y]:
Do you want to backup your configuration before upgrading Federator.ai? [default: y]: y
Please input path for storing backup configuration: [default: /opt/federatorai/configuration_backup]
Backup configuration...
backup yamls saved to folder /opt/federatorai/configuration_backup/federatorai-backup-1616551135
Done.
```

Downloading v4.5.1-ga tgz file ...
Done
Updating InfluxDB owner...
Done

Applying Federator.ai operator yaml files...
deployment.apps "federatorai-operator" deleted
Applying 00-namespace.yaml...
namespace/federatorai unchanged
Applying 01-serviceaccount.yaml...
serviceaccount/federatorai-operator unchanged
Applying 02-alamedaservice.crd.yaml...
customresourcedefinition.apiextensions.k8s.io/alamedaservices.federatorai.containers.ai configured
Delay applying 03-federatorai-operator.deployment.yaml
Applying 04-clusterrole.yaml...
clusterrole.rbac.authorization.k8s.io/federatorai-operator configured
clusterrole.rbac.authorization.k8s.io/alameda-gc configured
Applying 05-clusterrolebinding.yaml...
clusterrolebinding.rbac.authorization.k8s.io/federatorai-operator unchanged
Applying 06-role.yaml...
role.rbac.authorization.k8s.io/federatorai-operator configured
Applying 07-rolebinding.yaml...
rolebinding.rbac.authorization.k8s.io/federatorai-operator unchanged
Applying 08-service.yaml...
service/federatorai-operator-service created
Applying 09-secret.yaml...
secret/federatorai-operator-service-cert created
Applying 10-mutatingwebhook.yaml...
mutatingwebhookconfiguration.admissionregistration.k8s.io/federatorai-operator-servicemutation created
Applying 11-validatingwebhook.yaml...
validatingwebhookconfiguration.admissionregistration.k8s.io/federatorai-operator-servicestandardation created
Applying 03-federatorai-operator.deployment.yaml...
deployment.apps/federatorai-operator created
federatorai-operator pod is present.
Waiting for pod federatorai-operator-75bd65496-tz6fx in namespace federatorai to be ready ... phase: [Pending]
Waiting for pod federatorai-operator-75bd65496-tz6fx in namespace federatorai to be ready ... phase: [Pending]
Waiting for pod federatorai-operator-75bd65496-tz6fx in namespace federatorai to be ready ... phase: [Pending]
Waiting for pod federatorai-operator-75bd65496-tz6fx in namespace federatorai to be ready ... phase: [Running]

federatorai-operator pod is ready.

Downloading Federator.ai alamedaservice sample file ...
Done

Downloading Federator.ai alamedascaler sample files ...
Done
========================================
Update alamedaservice...
alamedaservice.federatorai.containers.ai/my-alamedaservice patched
Done.
alamedaservice.federatorai.containers.ai/my-alamedaservice patched
Processing...
Waiting for datahub(v4.5.1-ga) pod to appear ...

datahub pod is present.

Checking pods...
Waiting for pod alameda-ai-667df48565-99qft in namespace federatorai to be ready. phase: [Running]
Waiting for pods in namespace federatorai to be ready...
Waiting for pod alameda-ai-667df48565-99qft in namespace federatorai to be ready. phase: [Running]
Waiting for pods in namespace federatorai to be ready...
Waiting for pod alameda-ai-dispatcher-6845456b68-8kjfr in namespace federatorai to be ready. phase: [Running]
Waiting for pods in namespace federatorai to be ready...

All pods under namespace(federatorai) are ready.
The default alamedaorganization under namespace federatorai is ready.

You can now access GUI through https://<YOUR IP>:31012
The default login credential is admin/admin

Also, you can start to apply alamedascaler CR for the target you would like to monitor.
Review the administration guide for further details.

You can now access Federator.ai REST API through https://<YOUR IP>:31011
The default login credential is admin/admin
The REST API online document can be found in https://<YOUR IP>:31011/apis/v1/swagger/index.html

Install Federator.ai v4.5.1-ga successfully

Downloaded YAML files are located under /opt/federatorai/installation
Downloaded files are located under /opt/federatorai/repo/v4.5.1-ga

3. Verify Federator.ai pods are running properly

```
~# kubectl get pod -n federatorai
NAME                  READY STATUS    RESTARTS AGE
alameda-ai-6b56d6db77-x2r9x  1/1   Running   0           19m
alameda-ai-dispatcher-7d46f46849-nd4z2  1/1   Running   3           15m
alameda-analyzer-56bd4d4f8d-jww6f  1/1   Running   0           15m
alameda-datahub-597fb6f964-pqmh6  1/1   Running   3           20m
alameda-executor-6b4bff9b47-d9fdw  1/1   Running   4           19m
alameda-influxdb-0  1/1   Running   0           19m
alameda-notifier-87cf6b94c-xn5pz  1/1   Running   1           19m
alameda-operator-cf64f66c9-c877m  1/1   Running   0           20m
alameda-rabbitmq-addrcc8dd7-99jv2  1/1   Running   0           19m
```
Installing Datadog Watermark Pod Autoscaler (WPA)

If you wish to enable HPA autoscaling via Datadog WPA for your application, please follow the instructions below to install Datadog WPA.

- **Download Datadog WPA package**

```
~# wget https://github.com/DataDog/watermarkpodautoscaler/archive/master.zip
~# unzip master.zip
```

- **Install Watermark Pod Autoscaler controller**

  *WPA Helm Chart package requires using ‘helm’ to install. If you don’t have ‘helm’ installed, use the following command to install.*

```
```

- **Set up environment variables and then use ‘helm’ command to install WPA**

```
$ DD_NAMESPACE="default"
$ DD_NAMEWPA="wpacontroller"
$ helm install $DD_NAMEWPA -n $DD_NAMESPACE ./chart/watermarkpodautoscaler
```

```
~# pwd
/root/datadog_wpa/watermarkpodautoscaler
~# DD_NAMESPACE="default"
~# DD_NAMEWPA="wpacontroller"
~# helm install $DD_NAMEWPA -n $DD_NAMESPACE ./chart/watermarkpodautoscaler
~# kubectl get pods -n default
<table>
<thead>
<tr>
<th>NAME</th>
<th>READY</th>
<th>STATUS</th>
<th>RESTARTS</th>
<th>AGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>datadog-monitoring-6lckr</td>
<td>2/2</td>
<td>Running</td>
<td>0</td>
<td>2d19h</td>
</tr>
<tr>
<td>datadog-monitoring-cluster-agent-7d79559979-cnjhj</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>2d19h</td>
</tr>
<tr>
<td>datadog-monitoring-dwq7f</td>
<td>2/2</td>
<td>Running</td>
<td>0</td>
<td>2d19h</td>
</tr>
<tr>
<td>datadog-monitoring-hlm8x</td>
<td>2/2</td>
<td>Running</td>
<td>0</td>
<td>2d19h</td>
</tr>
<tr>
<td>datadog-monitoring-kube-state-metrics-765978777d-b5dnq</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>6d3h</td>
</tr>
<tr>
<td>nfs-client-provisioner-7cd5f68cf7-cfqkb</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>6d3h</td>
</tr>
<tr>
<td>wpacontroller-watermarkpodautoscaler-68484f8dd4-zxm22</td>
<td>1/1</td>
<td>Running</td>
<td>18</td>
<td>6d3h</td>
</tr>
</tbody>
</table>
```

- **Download WPA pod autoscaler CR yaml file**
~# wget https://github.com/DataDog/watermarkpodautoscaler/blob/master/deploy/crds/datadoghq.com_watermarkpodautoscalers_cr.yaml

- Edit datadoghq.com_watermarkpodautoscalers_cr.yaml
  Configure WPA to auto-scale Kafka consumer group and generic application (NGINX)

```yaml
~# mv datadoghq.com_watermarkpodautoscalers_cr.yaml wpa.yaml
~# vi wpa.yaml
apiVersion: datadoghq.com/v1alpha1
kind: WatermarkPodAutoscaler
metadata:
  name: consumer
  namespace: myproject
spec:
  # Add fields here
  # algorithm must be average
  algorithm: average
  maxReplicas: 10
  minReplicas: 1
  tolerance: 0.01
downscaleForbiddenWindowSeconds: 300
upscaleForbiddenWindowSeconds: 15
scaleUplimitFactor: 90
scaleDownLimitFactor: 90
scaleTargetRef:
  kind: Deployment
  apiVersion: apps/v1
  name: consumer
readinessDelay: 10
metrics:
  # Resource or External type supported
  # Example usage of External type
  - type: External
    external:
      # do not edit highWatermark, and lowWatermark
      # highWatermark and lowWatermark must be 1
      highWatermark: "1"
      lowWatermark: "1"
      metricName: federatorai.recommendation
      metricSelector:
        matchLabels:
          resource: replicas
          kube_cluster: k8s-4-205  # see below #notes-1 for more details
          kube_deployment: consumer
          kube_namespace: myproject

  # Example usage of Resource type
  #  type: Resource
  #  resource:
  #    highWatermark: "50"
  #    lowWatermark: "10"
  #    name: cpu
  #    metricSelector:
  #      matchLabels:
  #        foo: bar
---
```

---
apiVersion: datadoghq.com/v1alpha1
class: WatermarkPodAutoscaler
metadata:
  name: nginx-sample
  namespace: nginx-sample
spec:
  # Add fields here
  # algorithm must be average
  algorithm: average
  maxReplicas: 5
  minReplicas: 1
  tolerance: 0.01
downscaledForbiddendWindowSeconds: 300
  upscaleForbiddenWindowSeconds: 15
  scaleUpLimitFactor: 90
  scaleDownLimitFactor: 90
  scaleTargetRef:
    kind: Deployment
    apiVersion: apps/v1
    name: nginx-sample
  readinessDelay: 10
  metrics:
  # Resource or External type supported
  # Example usage of External type
  - type: External
    external:
      # do not edit highWatermark, and lowWatermark
      # highWatermark and lowWatermark must be 1
      highWatermark: "1"
      lowWatermark: "1"
      metricName: federatorai.recommendation
      metricSelector:
        matchLabels:
          resource: replicas
          kube_cluster: k8s-4-205 \(^* see below \#notes-1 for more details*
          kube_deployment: nginx-sample
          kube_namespace: nginx-sample

\#notes-1: “kube_cluster” value must match with DD_TAGS (value=“kube_cluster:<cluster_name>“) configured in Datadog Agent (datadog-values.yaml)

- Deploy WPA and confirm the status

~# kubectl apply -f wpa.yaml
Appendix

Datadog Dashboards Overview

The following Custom Datadog Dashboards are available after Federator.ai is installed.
ProphetStor Federator.ai Kafka Overview

With integration of ProphetStor Federator.ai, users can easily track the Kafka message production and consumption rates, as well as the prediction of message production rate from Federator.ai dashboard. Based on the prediction or message production rate, Federator.ai automatically scales Kafka consumer replicas to handle the workload. This can be visualized from Federator.ai dashboard where the recommended consumer replicas and the current number of consumer replicas are shown. Additionally, overall consumer lags as well as the average latency in the queue before a message is received by a consumer are also shown on the dashboard for better performance monitoring.

ProphetStor Federator.ai Cost Analysis Overview

Multicloud Cost Analysis

Current Cluster Cost:

Recommended Cluster - AWS:

Recommended Cluster Configuration - AWS:

Recommended Cluster - Azure:

Recommended Cluster Configuration - Azure:

Recommended Cluster - GCP:

Recommended Cluster Configuration - GCP:
Sysdig Dashboard Overview

The following Custom Sysdig Dashboards are available after Federator.ai is installed.

Federator.ai Cluster Overview

Federator.ai Application Overview
Federator.ai installation/uninstallation using Ansible

Only support Federator.ai since v4.4.0 or later

Prerequisite

**Ansible Control Node**

<table>
<thead>
<tr>
<th>Software:</th>
<th>Version:</th>
<th>Query Command:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ansible</td>
<td>2.10.2 or later</td>
<td>ansible --version</td>
</tr>
<tr>
<td>Ansible Collection - community.kubernetes</td>
<td>1.1.1 or later</td>
<td>ansible-galaxy collection list or</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ansible-galaxy collection install</td>
</tr>
<tr>
<td></td>
<td></td>
<td>community.kubernetes -vvv</td>
</tr>
<tr>
<td>Python</td>
<td>3.7 or later</td>
<td>python3 --version</td>
</tr>
<tr>
<td>OpenShift python client (Required by community.kubernetes collection)</td>
<td>0.11.2 or later</td>
<td>pip3 list</td>
</tr>
<tr>
<td>kubeconfig file (Need to copy target cluster’s kubeconfig file to the Ansible Control Node)</td>
<td></td>
<td>e.g. file is put on /root/.kube/config.135</td>
</tr>
</tbody>
</table>
Preparation (Ansible Control Node):

1. Install Ansible
   https://docs.ansible.com/ansible/latest/installation_guide/intro_installation.html

2. Install collection “community.kubernetes”

   ```
   ~# ansible-galaxy collection install community.kubernetes
   ```

3. Install python & pip
   https://www.python.org/downloads/

4. Install OpenShift python client if you are using OpenShift clusters

   ```
   ~# pip3 install openshift
   ```

5. Download Ansible playbook for Federator.ai

6. Modify user_variable.yml file for customizing needed info.

Installing Federator.ai

Variables for in user_variable.yml

<table>
<thead>
<tr>
<th>Group</th>
<th>Variable Name</th>
<th>Sample value</th>
<th>Description</th>
<th>Mandatory</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Federator.ai env</strong></td>
<td>federatorai_version</td>
<td>v4.5.1-ga</td>
<td>Federator.ai version tag</td>
<td>Y</td>
</tr>
<tr>
<td>Storage for Federator.ai pods</td>
<td>storage_type</td>
<td>ephemeral or persistent</td>
<td>Using ephemeral persistent volume type</td>
<td>Y</td>
</tr>
<tr>
<td>Storage info (Only be used when storage_type is persistent)</td>
<td>log_storage_size</td>
<td>10</td>
<td>Log size reserved for every pod. 10 means 10GB</td>
<td>N</td>
</tr>
<tr>
<td>Private repo</td>
<td>enable_private_repo</td>
<td>y</td>
<td>Using private repo to pull the Federator.ai required docker images</td>
<td>N</td>
</tr>
<tr>
<td>Pod resource</td>
<td>enable_resource_requirement</td>
<td>y</td>
<td>Add pod resource requirement (limits &amp; requests) for every Federator.ai pod</td>
<td>N</td>
</tr>
<tr>
<td>Expose services (Only be used when openshift_env is “n”)</td>
<td>expose_dashboard_and_rest_api_services</td>
<td>y</td>
<td>Expose the dashboard and API services in the Kubernetes cluster.</td>
<td>Y</td>
</tr>
<tr>
<td>Cluster type</td>
<td>openshift_env</td>
<td>n</td>
<td>Input “y” if installed cluster is OpenShift cluster</td>
<td>Y</td>
</tr>
<tr>
<td>Variable Name</td>
<td>Sample value</td>
<td>Description</td>
<td>Mandatory</td>
<td></td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>-------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------</td>
<td></td>
</tr>
<tr>
<td>installed_namespace</td>
<td>federatorai</td>
<td>namespace where Federator.ai will be installed</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>image_url_prefix</td>
<td>&quot;&quot;</td>
<td>Input the private repo URL</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>ai_engine_size</td>
<td>10</td>
<td>Storage size reserved for Alameda AI engine.</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>influxdb_storage_size</td>
<td>100</td>
<td>Data size reserved for InfluxDB pod.</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>storage_class_name</td>
<td>scname</td>
<td>To specifying storage class name for provisioning persistent volumes</td>
<td>Y</td>
<td></td>
</tr>
</tbody>
</table>

### Steps:

1. **Go to Ansible playbook folder**
   ```
   ~# cd ansible_for_federatorai
   ```

2. **Modify user_variable.yaml (under uninstaller folder) file for customizing needed info.**

3. **Export K8S_AUTH_KUBECONFIG to specify kubeconfig file path for Ansible collection**
   ```
   ~# export K8S_AUTH_KUBECONFIG=/root/.kube/config.135
   ```

4. **Run Ansible playbook**
   ```
   ~# ansible-playbook federtorai_installation.yaml
   ```

### Uninstalling Federator.ai

For Uninstallation, please use the file under ansible_for_federatorai/uninstaller directory.

### Variables in user_variable.yml.

<table>
<thead>
<tr>
<th>Group</th>
<th>Variable Name</th>
<th>Sample value</th>
<th>Description</th>
<th>Mandatory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage for Federator.ai pods</td>
<td>storage_type</td>
<td>ephemeral or persistent</td>
<td>Specify current Federator.ai storage type (ephemeral or persistent)</td>
<td>Y</td>
</tr>
<tr>
<td>Preserve current persistent volume (Only be used when storage_type is persistent)</td>
<td>preserve_pv</td>
<td>Y</td>
<td>Specify whether to preserve Federator.ai PVs</td>
<td></td>
</tr>
</tbody>
</table>
Steps:

1. Get to Ansible playbook uninstallation folder
   ```bash
   ~# cd ansible_for_federatorai/uninstaller
   ```

2. Modify user_variable.yaml (under uninstaller folder) file for customizing needed info.

3. Export `K8S_AUTH_KUBECONFIG` to specify kubeconfig file path for Ansible collection (community.kubernetes)
   ```bash
   ~# export K8S_AUTH_KUBECONFIG=/root/.kube/config.135
   ```

4. Run Ansible playbook
   ```bash
   ~# ansible-playbook federatorai_uninstaller.yaml
   ```

Troubleshooting

Downgrade from v4.5.1

- **v4.5.1 -> v4.4.1**
  Following v4.4.1 installation step to rollback to previous version v4.4.1 directly. V4.4.1 installation script automatically keeps existing configuration, metrics, and prediction data, which is stored on persistent volume.

- **v4.5.1 -> v4.3.1**
  Following v4.3.1 installation step to rollback to previous version v4.3.1 directly. V4.3.1 installation script keeps existing configuration, metrics, and prediction data, which is stored on persistent volume.

- **v4.5.1 -> v4.2**
  If Federator.ai is upgraded from v4.2, rollback could be done from the 4.2 configuration backup saved during the upgrade process. Here is the general workflow for downgrading to the 4.2 version:
  1. Run v4.5.1/Uninstall.sh script.
  2. Re-install v4.2.
  3. Restore 4.2 backup configuration.

**Step 1:** Run uninstall.sh. The uninstall script is placed under `/opt/federatorai/repo/v4.5.1-ga/scripts/` directory.

```bash
~# cd /opt/federatorai/repo/v4.5.1-ga/scripts/
~# bash uninstall.sh
Do you want to preserve your Federator.ai persistent volumes? [default: y]:
Patching pv pvc-09324a63-01cc-44d1-9d67-313d2172b41e ...
persistentvolume/pvc-09324a63-01cc-44d1-9d67-313d2172b41e patched (no change)
Done.
```
Patching pv pvc-0a0b1fb2-b96b-4c74-abdf-5aa1ef930f4f ... 
persistentvolume/pvc-0a0b1fb2-b96b-4c74-abdf-5aa1ef930f4f patched (no change) 
Done.
Patching pv pvc-0da9f9c8-9e0-4ac1-b5dc-50e6311d5920 ... 
persistentvolume/pvc-0da9f9c8-9e0-4ac1-b5dc-50e6311d5920 patched (no change) 
Done.
Patching pv pvc-0f6554ab-d0d6-46f1-a295-b3cf133ceef6 ... 
persistentvolume/pvc-0f6554ab-d0d6-46f1-a295-b3cf133ceef6 patched (no change) 
Done.
Patching pv pvc-15eeef793-2012-4a47-9b6b-067fbb999e0 ... 
persistentvolume/pvc-15eeef793-2012-4a47-9b6b-067fbb999e0 patched (no change) 
Done.
Patching pv pvc-29e8d506-b659-4f78-b22e-b74a0b3eaa80e ... 
persistentvolume/pvc-29e8d506-b659-4f78-b22e-b74a0b3eaa80e patched (no change) 
Done.
Patching pv pvc-33caea9a-8b6d-4786-806d-34ac3ca2a3d5 ... 
persistentvolume/pvc-33caea9a-8b6d-4786-806d-34ac3ca2a3d5 patched (no change) 
Done.
Patching pv pvc-4531b2ae-6678-4342-b83f-03e757013523 ... 
persistentvolume/pvc-4531b2ae-6678-4342-b83f-03e757013523 patched (no change) 
Done.
Patching pv pvc-4ad88729-6c1b-4fb7-95b9-fbc30748c2b6 ... 
persistentvolume/pvc-4ad88729-6c1b-4fb7-95b9-fbc30748c2b6 patched (no change) 
Done.
Patching pv pvc-5452d9fd-4e71-42a5-a03c-2435c7539972 ... 
persistentvolume/pvc-5452d9fd-4e71-42a5-a03c-2435c7539972 patched (no change) 
Done.
Patching pv pvc-570ad717-a306-4800-b6a0-cbe02a1805e3 ... 
persistentvolume/pvc-570ad717-a306-4800-b6a0-cbe02a1805e3 patched (no change) 
Done.
Patching pv pvc-65bb40fe-0c98-4f5c-8af0-42558f0510f1 ... 
persistentvolume/pvc-65bb40fe-0c98-4f5c-8af0-42558f0510f1 patched (no change) 
Done.
Patching pv pvc-6a1257b4-7582-4ab9-be66-5f7d8e85badc ... 
persistentvolume/pvc-6a1257b4-7582-4ab9-be66-5f7d8e85badc patched (no change) 
Done.
Patching pv pvc-6f6b2a5a-7b53-421a-85e4-25491688057a ... 
persistentvolume/pvc-6f6b2a5a-7b53-421a-85e4-25491688057a patched (no change) 
Done.
Patching pv pvc-7412750a-fe39-4a79-a78b-b47fd6f18f68 ... 
persistentvolume/pvc-7412750a-fe39-4a79-a78b-b47fd6f18f68 patched (no change) 
Done.
Patching pv pvc-79dfbb73-cdc7-4ac0-a73e-94b1b973f60b ... 
persistentvolume/pvc-79dfbb73-cdc7-4ac0-a73e-94b1b973f60b patched (no change) 
Done.
Patching pv pvc-7fdb8acb-461a-4633-815a-2eea4b8d1148 ... 
persistentvolume/pvc-7fdb8acb-461a-4633-815a-2eea4b8d1148 patched (no change) 
Done.
Patching pv pvc-83f71f04-9516-44fa-a083-84732e9240ed ... 
persistentvolume/pvc-83f71f04-9516-44fa-a083-84732e9240ed patched (no change) 
Done.
Patching pv pvc-8de6d659-d003-4243-91c3-ca7526f33c2d ... 
persistentvolume/pvc-8de6d659-d003-4243-91c3-ca7526f33c2d patched (no change)
Patching pv PVC ... 
persistentvolume/pvc patched (no change) 
Done.
Patching pv PVC ... 
persistentvolume/pvc patched (no change) 
Done.
Patching pv PVC ... 
persistentvolume/pvc patched (no change) 
Done.
Patching pv PVC ... 
persistentvolume/pvc patched (no change) 
Done.
Patching pv PVC ... 
persistentvolume/pvc patched (no change) 
Done.
Patching pv PVC ... 
persistentvolume/pvc patched (no change) 
Done.
Patching pv PVC ... 
persistentvolume/pvc patched (no change) 
Done.
Patching pv PVC ... 
persistentvolume/pvc patched (no change) 
Done.
Patching pv PVC ... 
persistentvolume/pvc patched (no change) 
Done.
Patching pv PVC ... 
persistentvolume/pvc patched (no change) 
Done.
Patching pv PVC ... 
persistentvolume/pvc patched (no change) 
Done.
Patching pv PVC ... 
persistentvolume/pvc patched (no change) 
Done.
Patching pv PVC ... 
persistentvolume/pvc patched (no change) 
Done.
Patching pv PVC ... 
persistentvolume/pvc patched (no change) 
Done.
Patching pv PVC ... 
persistentvolume/pvc patched (no change) 
Done.
Patching pv PVC ... 
persistentvolume/pvc patched (no change) 
Done.
Patching pv PVC ... 
persistentvolume/pvc patched (no change) 
Done.
Patching pv PVC ... 
persistentvolume/pvc patched (no change) 
Done. 

Installation and Configuration Guide
persistentvolume/pvc-f4f884c6-066e-4ebf-90e4-426a132417cf patched (no change) 
Done.
Patching pv pvc-f8257bf4-abf9-4de2-b3da-1f2daa1451ad ...
persistentvolume/pvc-f8257bf4-abf9-4de2-b3da-1f2daa1451ad patched (no change) 
Done.
Patching pv pvc-fb35cb57-436a-4561-80f3-2a3e0b763c8f ...
persistentvolume/pvc-fb35cb57-436a-4561-80f3-2a3e0b763c8f patched (no change) 
Done.
Patching pv pvc-fe27c9cf-80d4-4acc-b50b-94bd09d575a4 ...
persistentvolume/pvc-fe27c9cf-80d4-4acc-b50b-94bd09d575a4 patched (no change) 
Done.

----------------------------------------
Starting to remove the Federator.ai product 
----------------------------------------

Please input your Federator.ai Operator tag: v4.5.1-ga

----------------------------------------
Your tag number = v4.5.1-ga 
----------------------------------------

Is the above information correct? [default: y]:
Downloading file 00-namespace.yaml ...
Downloading file 01-serviceaccount.yaml ...
Downloading file 02-alamedaservice.crd.yaml ...
Downloading file 03-federatorai-operator.deployment.yaml ...
Downloading file 04-clusterrole.yaml ...
Downloading file 05-clusterrolebinding.yaml ...
Downloading file 06-role.yaml ...
Downloading file 07-rolebinding.yaml ...
Downloading file 08-service.yaml ...
Downloading file 09-secret.yaml ...
Downloading file 10-mutatingwebhook.yaml ...
Downloading file 11-validatingwebhook.yaml ...

Deleting my-alamedaservice in federatorai namespace...
clusterrole.rbac.authorization.k8s.io "alameda-gc" deleted

Deleting 11-validatingwebhook.yaml ...
validatingwebhookconfiguration.admissionregistration.k8s.io "federatorai-operator-servicesvalidation" deleted

Deleting 10-mutatingwebhook.yaml ...
mutatingwebhookconfiguration.admissionregistration.k8s.io "federatorai-operator-servicesmutation" deleted

Deleting 09-secret.yaml ...
secret "federatorai-operator-admission" deleted

Deleting 08-service.yaml ...
service "federatorai-operator" deleted

Deleting 07-rolebinding.yaml ...
rolebinding.rbac.authorization.k8s.io "federatorai-operator" deleted
Deleting 06-role.yaml ...
role.rbac.authorization.k8s.io "federatorai-operator" deleted
Deleting 05-clusterrolebinding.yaml ...
clusterrolebinding.rbac.authorization.k8s.io "federatorai-operator" deleted
Deleting 04-clusterrole.yaml ...
clusterrole.rbac.authorization.k8s.io "federatorai-operator" deleted
Error from server (NotFound): error when deleting "04-clusterrole.yaml":
clusterroles.rbac.authorization.k8s.io "alameda-gc" not found
Error in removing 04-clusterrole.yaml
Deleting 03-federatorai-operator.deployment.yaml ...
deployment.apps "federatorai-operator" deleted
Deleting 02-alamedaservice.crd.yaml ...
customresourcedefinition.apimachinery.k8s.io "alamedaservices.federatorai.containers.ai" deleted
Deleting 01-serviceaccount.yaml ...
serviceaccount "federatorai-operator" deleted
Deleting 00-namespace.yaml ...
namespace "federatorai" deleted
Namespace federatorai is removed successfully.

**Step 2:** Reinstall Federator.ai 4.2.

**Step 3:** Restore 4.2 backup configuration.

Follow the steps below:

a. Go to /opt/federatorai/configuration_backup, which is the default federator.ai configuration backup directory.

b. Change to the directory where the 4.2 configuration backup is stored.

c. Run backup-restore.sh script.

```bash
~# cd /opt/federatorai/configuration_backup
~# cd federatorai-backup-1611212333
~# bash backup-restore.sh -r
```
Installing Federator.ai containers.ai

1. Downloading file 07-rolebinding.yaml ...

2. /opt/federatorai/configuration_backup/federatorai-backup-1611212333

3. namespace/federatorai created

4. serviceaccount/federatorai created

5. customresourcedefinition.apiextensions.k8s.io/alamedaservices.federatorai.created

6. deployment.apps/federatorai created

7. rolebinding.rbac.authorization.k8s.io/federatorai created

8. clusterrole.rbac.authorization.k8s.io/federatorai created

9. clusterrolebinding.rbac.authorization.k8s.io/federatorai created

10. role.rbac.authorization.k8s.io/federatorai created

11. rolebinding.rbac.authorization.k8s.io/federatorai created

12. Restore service

13. alamedaservice.federatoraicontainers.ai created

14. customresourcedefinition.apiextensions.k8s.io created

15. rolebinding.rbac.authorization.k8s.io created

16. role.rbac.authorization.k8s.io created

17. serviceaccount/federatorai created

18. namespace/federatorai created

19. Downloading file 07-serviceaccount/federatorai -v4.5.1

20. operator created

21. alamedaservice cr created

22. Patch pv if necessary

23. persistentvolume/pvc-09324a63-01cc-44d1-9d67-313d21272b41e patched

24. persistentvolume/pvc-0a0b1fb2-b96b-4c74-abdf-5aa1ef930f4af patched

25. persistentvolume/pvc-0da9f9c8-9ee0-4ac1-b5dc-50e6311d5920 patched

26. persistentvolume/pvc-0f6554ab-d0d6-4f61-a295-b3c1f33ceef6 patched

27. persistentvolume/pvc-15eeef79-2012-4aa7-9b6b-067fbb999e0 patched

28. persistentvolume/pvc-29e8d506-b659-4f78-b22e-b74a0baee80e patched

29. persistentvolume/pvc-33cae9a9-8b6d-4786-806d-34ac3ca2a3d5 patched

30. persistentvolume/pvc-4531b2ae-6678-4342-b83f-03e757013523 patched

31. persistentvolume/pvc-4ad88729-6c1b-4f7b-95b9-fbc30748c2b6 patched

32. persistentvolume/pvc-5452d9fd-e471-42a5-a03c-2435753972 patched

33. persistentvolume/pvc-570a717-a306-4800-b6a0-cbe02a1805e3 patched

34. persistentvolume/pvc-65bb40fe-0c98-4f5c-8af0-42558f0510f1 patched

35. persistentvolume/pvc-6a1257b4-7582-4ab9-be66-5f7d8e85badc patched

36. persistentvolume/pvc-6feb2a5a-7b53-421a-85e4-25491680857a patched

37. persistentvolume/pvc-7412750a-fe39-4a79-a7b8-247fd6f18f68 patched

38. persistentvolume/pvc-79dfbb73-cdc7-4ac0-a73e-94b1973f60b patched

39. persistentvolume/pvc-7fadb8ac-b461-a633-815a-2ea4a8bd1d4e patched

40. persistentvolume/pvc-837f1f04-9516-44fa-a083-8473e9240ed patched

41. persistentvolume/pvc-8de66d59-d003-4243-91c3-ca7526f33c2d patched

42. persistentvolume/pvc-8fe930f1-24cd-4a9e-b7e1-4e7b907d6f65 patched

43. persistentvolume/pvc-98b4d67b-5730-4300-86c6-1ee65c9a0588 patched

44. persistentvolume/pvc-9a4ecf7e-8579-45d0-92cd-655aa9053f9 patched

45. persistentvolume/pvc-9ac0c3d-c299-44b4-bc59-b5a9e8b56521 patched

46. persistentvolume/pvc-9b7ca77-0bde-4748-9eda-ca067dc6c710 patched

47. persistentvolume/pvc-9e7429c9-30df-4790-b706-61a1b88cbe35 patched

48. persistentvolume/pvc-b10c40d3-6485-4ddb-828e-dec8693ca31e patched

49. persistentvolume/pvc-b3b35cad-1a5b-4f6b-93de-2026e4502112 patched

50. persistentvolume/pvc-b517207b-54b6-4a42-81da-936acff9fd30 patched

51. persistentvolume/pvc-bc70b3d2-9e14-442a-93ob-3f817f3120b9 patched

52. persistentvolume/pvc-bd3cf813-ec79-4649-a685-1a8fac8f375c patched

53. persistentvolume/pvc-c4e1717a-bbff-4997-ab94-d3c6e13c05a3 patched

54. persistentvolume/pvc-ced6151e-962a-4bd9-854c-28083ca292e8 patched

55. persistentvolume/pvc-d09e0e1-5f54-4c01-8c28-c02a9f951b7d patched

56. persistentvolume/pvc-d88f0a6e-e645-4980-a477-b354f1182ae patched

57. persistentvolume/pvc-dccdbbf7-6f7f-46a9-93cc-8a5b97e7126d patched

58. persistentvolume/pvc-e7667f8f-8e7d-4a7b-9fae-4d59726a59d patched

59. persistentvolume/pvc-f4f884c0-066e-4ebf-90e4-426a132417cf patched

60. persistentvolume/pvc-f8257bf4-4bf9-4de2-b3da-1f2daa1451ad patched

61. alamedaservice.federatoraicontainers.ai created
Datadog Integration

1. Datadog WPA dumps errors during autoscaling

   • Error message in WPA Controller

```
~# kubectl get pod -n default
NAME                        AGE     READY STATUS  RESTARTS
---                         ----     ------ ------  ------
datadog-agent-2m6kk         2d       1/1    Running  2
/datadog-agent-8kd54        2d       1/1    Running  0
datadog-agent-94rl6         2d       1/1    Running  0
datadog-agent-mq4mv         2d       1/1    Running  0
datadog-cluster-agent-74f44fdd4d-82tjp 1d       1/1    Running  0
docker-registry-1-vw59s      324d     1/1    Running  4
/prometheus-adap799b7dfc4f-rs7zj 6d       1/1    Running  1
registry-console-2-jxfdl    6d       1/1    Running  2
router-1-sw78l              324d     1/1    Running  4
/wpacontroller-watermarkpodautoscaler-7ffbb97f9d-hcbsg 1d       1/1    Running  0

~# kubectl logs wpacontroller-watermarkpodautoscaler-7ffbb97f9d-hcbsg -n default
```
• Reason
  - WPA is incompatible with Kubernetes 1.11
  - Install WPA on Kubernetes 1.11 dumps errors

```
must only have "properties", "required" or "description" at the root if
the status subresource is enabled
```

• Workaround
  - Comment out ‘subresources’ key in WatermarkPodAutoscaler CRD

```
# cd
datadog_wpa/watermarkpodautoscaler_for_k8s_1.11/chart/watermarkpodautoscaler/templates
# vi datadoghq.com_watermarkpodautoscalers_crd.yaml
...
...
shortNames:
  - wpa
  singular: watermarkpodautoscaler
scope: Namespaced
#subresources: ← comment out
# status: {} ← comment out
validation:
  openAPIV3Schema:
    description: WatermarkPodAutoscaler is the Schema for the
    watermarkpodautoscalers
    API
    properties:
      apiVersion:
        description: 'APIVersion defines the versioned schema of this
        representation
```

Note: It can auto-scale monitored application, but dump some errors during update status
2. Data Adapter reports errors

- Error messages in Data Adapter logs
  
  ```bash
  # oc exec -it $oc get pods|grep federatorai-data-adapter|grep Running|awk '{print $1}' | -- cat /var/log/telegraf.log
  # cat telegraf.log | grep "E:\"
  2020-05-15T09:59:33Z E! [datadog][application_aware] Failed to get kafka consumer spec replicas
  ```

- Reason
  Datadog Agent does not work with ‘kube-state-metrics’ comes with OpenShift

- Solution
  Install another compatible ‘kube-state-metrics’
  
  If there is another kube-state-metrics running on openshift, rename all the clusterrole and clusterrolebinding name of kube-state-metrics to prevent kube-state-metrics clusterrole name collision
  
  restart datadog agent and make sure agent integrate with kube-state-metrics properly.
  check all the node agent status by following command
  
  ```bash
  # oc exec <datadog-agent-pod-name> agent status
  ```