Установка Neuvector в конфигурации по умолчанию

Для установки модуля Neuvector в Nova Container Platform используйте один из представленных далее манифестов:

▸ Для кластера в конфигурации по умолчанию

▸ Для высокодоступного кластера в рекомендуемой конфигурации

1. Установка модуля Neuvector

1.1. Предварительные условия

- ✓ Вы ознакомились с архитектурой и концепциями Neuvector в Nova Container Platform.
- ✓ Вы ознакомились с документацией по планированию установки и системным требованиям Neuvector.
- ✓ У вас есть доступ к кластеру с учетной записью, имеющей роль `cluster-admin` в Kubernetes.
- ✓ Вы установили утилиту `kubectl` для работы с Kubernetes.

1.2. Установка с помощью kubectl

Процедура

1. Сохраните представленный выше манифест в файл, например, `neuvector.yaml`.
2. Установите манифест в кластер Kubernetes, выполнив команду:

```
kubectl apply -f neuvector.yaml
```
BASH

3. Проверьте состояние запущенных компонентов Neuvector, выполнив команду:

```
kubectl get pods -n nova-neuvector
```
BASH

4. Проверьте состояние Cluster Kustomizations, выполнив команду:

```bash
kubectl get ks -l nova-application-group=cluster-security -n nova-gitops
```

1.3. Установка с помощью Nova Console

Процедура

1. Скопируйте представленный выше манифест в буфер обмена.
2. Выполните вход в Nova Console.
3. Используйте опцию импорта нового объекта и вставьте в форму ранее скопированный манифест.

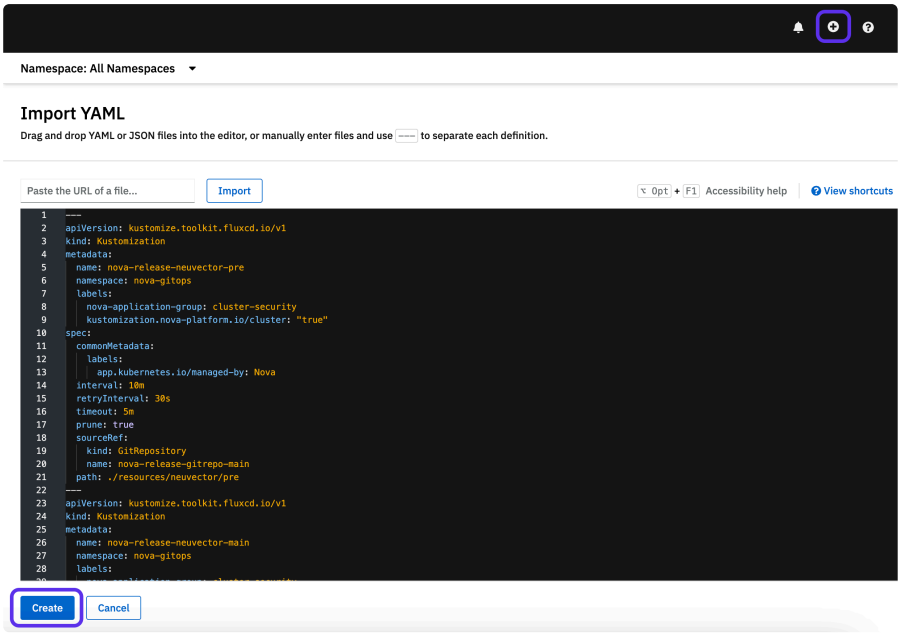


Рисунок 1. Установка модуля Neuvector

4. Перейдите в раздел *Workloads*, далее *Pods* и выберете пространство имен `nova-neuvector`. Проверьте состояние запущенных компонентов Neuvector.
5. Перейдите в раздел *Administration*, далее *Cluster Settings*, во вкладке *Configuration* выберете *Cluster Kustomizations*. Проверьте состояние *Cluster Kustomizations* с именем `nova-neuvector`.

Планирование установки и системные требования

1. Общие требования к установке

Для установки модуля Neuvector в Nova Container Platform должны быть выполнены следующие условия:

- Nova Container Platform должна иметь версию v2.0.0 и выше.
- У вас есть доступ к кластеру с учетной записью, имеющей роль `cluster-admin` в Kubernetes.
- На инфраструктурных узлах кластера достаточно ресурсов для запуска компонентов Neuvector либо для размещения Neuvector подготовлены отдельные узлы кластера Kubernetes.

2. Системные требования

Общие требования по вычислительным ресурсам можно найти в статье

3. Требования к вычислительным ресурсам

В таблице ниже представлены минимальные требования к ресурсам компонентов Neuvector.

Компонент	vCPU, шт.	RAM, GB
Controller	1	2
Enforcer	1	2
Scanner	1	2
Manager	1	2

Во избежание некорректной или непредсказуемой работы компонентов Neuvector рекомендуется не устанавливать лимиты на ресурсы компонентов в Kubernetes. Необходимо в первую очередь разместить в кластере Kubernetes все планируемые нагрузки, правила сетевой фильтрации и политики безопасности. Далее - выполнить оценку потребления ресурсов компонентами Neuvector с течением времени, и только по результатам данной оценки установить лимиты на ресурсы.

4. Требования к количеству компонентов

В таблице ниже представлены требования к количеству компонентов Neuvector.

Компонент	**Рекомендации по количеству реплик**
Controller	Минимальное количество - 1.Количество реплик для высокой доступности компонента - 3.
Enforcer	Всегда 1 реплика на каждом узле кластера Kubernetes.
Scanner	Минимальное количество - 1.Количество реплик для высокой доступности компонента - 2 и более.
Manager	Минимальное количество - 1.Количество реплик для высокой доступности компонента - 2 и более.

5. Рекомендации по планированию ресурсов

Далее представлены отдельные рекомендации, на которые следует обратить внимание при планировании объема ресурсов, потребляемых компонентами Neuvector в зависимости от типа, характера нагрузки и объемов обрабатываемых данных.

> **Информация**
>
> Всем компонентам Neuvector для корректной работы требуется достаточное количество ресурсов CPU и RAM. Убедитесь, что узлы кластера Kubernetes позволяют разместить все компоненты с учетом их минимальных требований.

5.1. Общие рекомендации

Следует учесть факторы, влияющие на потребление ресурсов компонентами Neuvector:

- **Сканирование больших образов:** чем больше сканируемый в системе образ, тем больше ресурсов RAM требуется компоненту Scanner.
- **Большое количество сетевых соединений в режиме *Protect*** влияет на потребление ресурсов CPU и RAM компонентами *Enforcer* на каждом узле кластера Kubernetes.
- **Количество узлов в кластере:** чем больше узлов в кластере Kubernetes, тем больше соединений обслуживается компонентами *Enforcer*, а также растет количество событий, координируемых компонентами *Controller*.
- **Количество Pod на каждом узле кластера:** чем больше Pod на узлах кластера Kubernetes, тем больше возрастает объем трафика, в том числе в пределах одного узла.

Это, в свою очередь, создает дополнительную нагрузку как на *Enforcer*, так и *Controller*.

- **Количество образов в сканируемом хранилище**: чем больше образов в хранилище, тем дольше времени занимает процесс сканирования. Если в вашем хранилище более 1000 образов, или время сканирования не удовлетворительное, рекомендуется использовать большее количество реплик компонентам *Scanner*. Также вы можете использовать опцию автоматического масштабирования *Scanner* в параметрах Neuvector.

5.2. Рекомендации по компоненту Enforcer

В режиме *Monitor* (наблюдение за потоками трафика) компонент *Enforcer* не испытывает высокой нагрузки и обрабатывает трафик без задержек.

В режиме *Protect* (блокирование соединений “на лету”) компоненту *Enforcer* требуются значительные ресурсы CPU и RAM для фильтрации и глубокой инспекции сетевых пакетов (DPI). Как правило, рекомендуемых минимальных требований для *Enforcer* достаточно, однако, вы всегда можете увеличить их, если ваш кластер Kubernetes обслуживает большое количество нагрузок.

5.3. Рекомендации по компоненту Scanner

Управление процессом сканирования образа контейнера выполняется компонентом *Controller*, а само сканирование образа выполняется компонентом *Scanner*. Образ контейнера загружается в память *Scanner*, распаковывается и анализируется. Минимальные требования к ресурсам *Scanner* предполагают, что размер одного сканируемого образа составляет не более 512 MB.

Если вы планируете сканировать образы размером более 512 MB, то размер RAM для *Scanner* следует устанавливать как размер самого большого образа и дополнительно 512 MB для работы сканера.

Пример

Максимальный размер вашего образа составляет 2.4 GB. Размер RAM для *Scanner* будет составлять 2.4 GB + 0.5 GB = 2.9 GB.

6. Конфигурация по умолчанию

В Nova Container Platform по умолчанию не установлены лимиты на потребление ресурсов (Resource Limits) компонентами Neuvector. В таблице ниже представлены объемы запрашиваемых ресурсов по умолчанию, необходимых для стабильной работы компонентов Neuvector.

Компонент	CPU Requests	Memory Requests
Controller	100m	2280Mi
Enforcer	100m	2280Mi
Scanner	100m	2280Mi
Manager	100m	2280Mi
Registry Adapter	100m	1Gi
Updater	1m	10Mi
API docs	100m	100Mi
Provisioner	250m	256Mi

В таблице ниже представлены дополнительные компоненты Neuvector с установленными лимитами на потребление ресурсов.

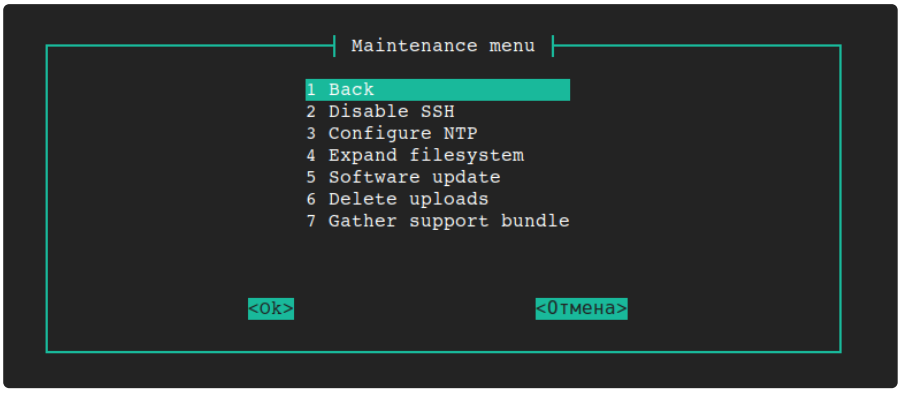
Компонент	CPU Requests	CPU Limits	Memory Requests	Memory Limits
Prometheus Exporter	100m	100m	128Mi	128Mi

Восстановление сервисов после сбоя вызванного переполнением системного диска

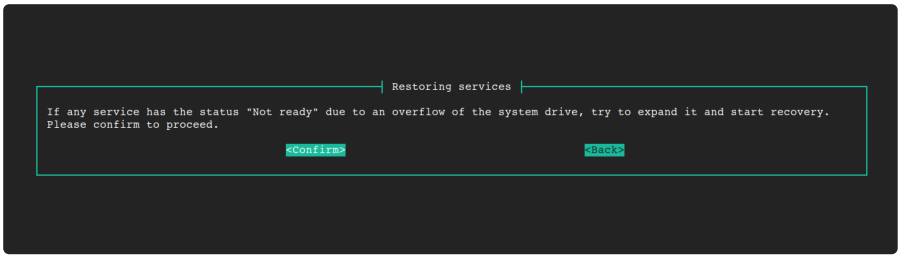
Статья описывает восстановление сервисов после сбоя вызванного переполнением системного диска.

1. Как скачать архив логов

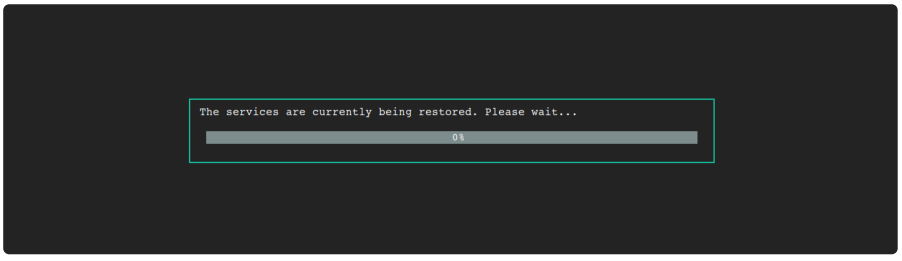
1. Зайдите в консоль Universe
2. Нажмите **Configure > System settings > Maintenance menu > Restore services**



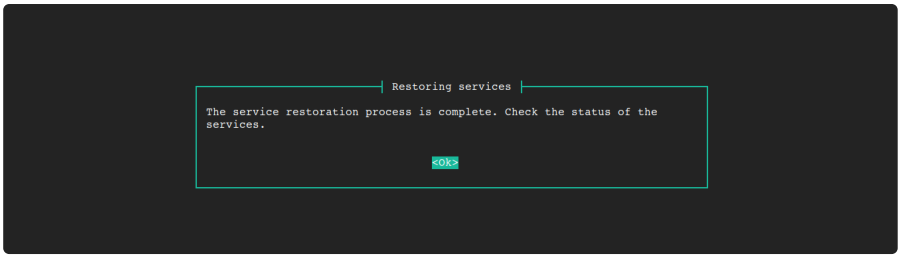
3. Если Nova Universe инициализирован, то появится окно подтвержения. Нажмите **[Confirm]**.



4. Подождите пока восстановятся сервисы.



5. После завершения процесса восстановления проверте статус сервисов.



Восстановление сервисов возможно, если Nova Universe инициализирован.

Пользователь получит сообщение с предупреждением, если Nova Universe не инициализирован.