

# Глоссарий

Nova Container Platform - это платформа, построенная на основе Kubernetes. Поэтому, многие технологии, термины и определения являются общими.

Далее приведен глоссарий основных терминов, устоявшихся выражений, примитивов и определений, которые вы можете встретить в данной документации и при работе с платформой.

## 1. Общие термины и определения

---

### ***Admission plugins***

Специальные контроллеры Kubernetes, предназначенные для перехвата запросов к серверу Kubernetes API с целью проверки запроса или его изменения. Применяются для контроля настроек безопасности, конфигурации развертывания или количества выделяемых ресурсов.

### ***Bootstrap***

Процесс первоначального развертывания платформы Nova Container Platform.

### ***Container workloads***

Приложения и пользовательские сервисы, упакованные в контейнеры.

### ***Containerd***

Среда исполнения контейнеров, совместимая с Kubernetes, предоставляющая интерфейс (CRI) взаимодействия с Kubelet.

### ***Control plane***

Control Plane Nodes, Master Nodes

Control plane (плоскость управления) - служебные компоненты, предоставляющие все основные API для обеспечения жизненного цикла контейнеров в Kubernetes (например, `kube-apiserver`, `kube-scheduler`, `kube-controller-manager`). Для компонентов Control Plane, как правило, выделяются отдельные вычислительные мастер-узлы.

### ***Custom Resource (CR)***

Объект CR является частью какого-либо расширения Kubernetes API.

### ***Deployment***

Ресурс в среде Kubernetes, с помощью которого можно управлять жизненным циклом пользовательского приложения.

## ***Dockerfile***

Текстовый файл, описывающий конфигурацию сборки контейнера.

## ***Config map***

Объект ConfigMap предоставляет возможность добавлять конфигурационные данные в Pod. Такие данные можно получить внутри Pod, если к нему примонтировать том с типом ConfigMap, после чего приложения в Pod смогут их использовать.

## ***Ingress***

Ресурс Kubernetes, предназначенный для публикации сервиса Kubernetes на балансировщике Ingress Controller. С помощью ресурса Ingress можно обеспечить доступ пользователей к приложениями, развернутым в Kubernetes.

## ***Installer-provisioned infrastructure (IPI)***

Автоматизированный метод развертывания в инфраструктуре, подготовленной узлом nova-ctl для управления платформой.

## ***Kubelet***

Компонент Kubernetes (серверный агент), запускаемый на каждом узле кластера и обеспечивающий работу контейнеров в составе Pod.

## ***Namespaces***

Ресурс Namespace (пространство имен) предназначен для изоляции групп ресурсов в Kubernetes.

## ***Node***

Узел кластера Kubernetes в Nova Container Platform. Узел может быть как виртуальной машиной, так и физическим сервером.

## ***Pod***

Один или более контейнеров с общими ресурсами, такими как тома и IP-адреса. Pod является минимальным определяемым ресурсом для запуска какого-либо приложения в Kubernetes.

## ***Service***

Ресурс сервиса Kubernetes обеспечивает единую точку доступа к Pod приложения.

## ***User-provisioned infrastructure (UPI)***

Автоматизированный метод развертывания в инфраструктуре, подготовленной пользователем.

## ***Аутентификация***

Authentication

Для контроля доступа к кластеру Nova Container Platform, администратор кластера может настроить параметры аутентификации пользователей. Это гарантирует доступ к

кластеру только подтвержденным пользователям. Для работы с Nova Container Platform необходимо пройти аутентификацию для сервера Kubernetes API с помощью OAuth-токенов или сертификатов TLS, предоставляемых в запросах к Kubernetes API.

### **Веб-консоль управления кластером**

web console

Пользовательский интерфейс графической консоли управления Nova Container Platform

### **Вычислительные узлы**

Compute nodes, Worker nodes, Nodes

Узлы кластера, не обслуживающие нагрузки Control Plane. Данные узлы отвечают за работу инфраструктурных сервисов платформы, балансировщиков нагрузки или конечных пользовательских сервисов.

### **Гибридные развертывания**

Развертывания кластера Nova Container Platform, в которых часть вычислительных узлов находится в отдельной среде виртуализации, частном или публичном облаке или на физическом оборудовании.

### **Зеркало хранилища образов контейнеров**

mirror registry

Локальное хранилище образов контейнеров, в котором размещаются контейнеры Nova Container Platform.

### **Контейнеры**

Containers

Легковесные запускаемые образы, в состав которых входит некоторое ПО и его зависимости. Поскольку в контейнерах виртуализируется операционная система, вы можете запускать контейнеры одинаково в любом совместимом окружении.

### **Контрольные группы Linux (cgroups)**

Control groups

Механизм ОС, с помощью которого для группы процессов на уровне ядра может быть установлена изоляция и ограничения потребляемых ресурсов.

### **Манифест Kubernetes**

kubernetes manifest

Спецификация объекта Kubernetes API в формате JSON или YAML. Один манифест может содержать конфигурацию множества ресурсов Kubernetes ( `deployments` , `configmaps` , `secrets` , `statefulsets` ).

### **Масштабирование**

scaling

Увеличение или уменьшение количества ресурсов узлам или приложениям.

Масштабирование может быть как горизонтальным (увеличение количества узлов или реплик Pod), так и вертикальным (увеличение количества ресурсов, выделяемых узлу или Pod).

### **Метаданные**

metadata

Любая дополнительная информация, которая может быть указана для ресурса Kubernetes. Как правило, в метаданных находятся метки ( labels ) и аннотации ( annotations ) для ресурсов Kubernetes.

### **Микросервисы**

microservices

Подход к разработке комплексных приложений, предполагающий разделение приложения на минимально возможные небольшие независимые компоненты (микросервисы), способные взаимодействовать по сети.

### **Монолитные приложения**

monolithic applications

Приложение в виде единого общего модуля, в состав которого входят все его компоненты.

### **Оператор Kubernetes**

Operator

Один из способов поставки, развертывания и управления приложениями в Kubernetes.

### **Отклонение конфигурации**

Configuration drift

Ситуация, когда состояние объекта в кластере отличается от состояния, описанного в файле его конфигурации.

### **Политики доступа**

access policies

Набор определенных действий, связанных с некоторыми объектами (пользователь, сущность, приложение) в кластере или за его пределами, определяющий границы их взаимодействия. Политики доступа повышают безопасность работы.

### **Провайдер инфраструктуры**

infrastructure provider

Платформа виртуализации или облачный сервис, предоставляющие API для автоматизированного развертывания объектов инфраструктуры кластера Nova Container Platform.

### ***Система управления контейнерами***

container orchestration engine

ПО, предназначенное для автоматизации задач развертывания, управления, масштабирования и обеспечения сетевой связанностью контейнеров.

### ***Управление доступом на основе ролей (RBAC)***

role-based access control (RBAC)

Основной механизм управления разграничениями доступа к ресурсам кластера на основе ролей.

### ***Хранилище***

storage

Nova Container Platform поддерживает различные типы хранилищ. По умолчанию, для служебных нужд доступно персистентное хранилище на базе локальных директорий на инфраструктурных узлах кластера.

# Резервное копирование и восстановление

Данный раздел содержит статьи полезные для настройки резервного копирования и восстановления в Nova Container Platform.

## 1. Резервное копирование и восстановление в Nova Container Platform

Резервное копирование и восстановление - это процессы создания резервной копии компонентов платформы Nova и последующего восстановления данных компонентов из этой копии, если что-то пойдет не так.

### 1.1. Поддерживаемые операции с мастер-узлами

Иногда может возникать необходимость остановить или перезапустить кластер Nova Container Platform для обслуживания оборудования или среды виртуализации. Вы можете воспользоваться соответствующими руководствами по корректному выключению и перезапуску кластера.

Перед выключением или перезапуском кластера вы должны сделать полную резервную копию мастер-узлов, которая включает в себя набор критически важных компонентов для работы платформы, таких как Etcd, StarVault, TLS-сертификаты, ключи шифрования, а также различные конфигурационные файлы.



Кластерные TLS-сертификаты имеют срок действия 1 год с момента их выпуска при установке платформы. В течение этого периода вы можете рассчитывать на корректное включение или перезапуск кластера. Однако, если срок действия сертификатов истечет к моменту включения узлов кластера, вам будет необходимо обратиться к **поддержке платформы** для принудительного перевыпуска сертификатов.

Полная резервная копия мастер-узлов Kubernetes может быть выполнена как с помощью регулярного задания CronJob в Kubernetes, при котором данные сохраняются на предоставляемое пользователем NFS-хранилище, так и с помощью модуля Data Protection, в случае которого данные сохраняются во внешнем объектном хранилище.

### 1.2. Сценарии восстановления данных

Возможны ситуации, когда компоненты Nova Container Platform или кластер Kubernetes частично или полностью перестает функционировать из-за непредвиденных обстоятельств, например:

- При выходе из строя узлов виртуализации, повреждении оборудования или его компонентов, программных сбоев или потери сетевой связанности.
- При удалении критически важных данных на мастер-узлах платформы или в кластере Kubernetes.
- При потере кворума на мастер-узлах в кластерах Etcd, StarVault.

В разделе [Восстановление данных на мастер-узлах](#) вы можете найти подробную информацию о том, как действовать в том или ином сценарии.

## 1.3. Резервное копирование и восстановление пользовательских данных

Для резервного копирования и восстановления приложений (пользовательских данных) в Nova Container Platform используется дополнительный модуль Nova Data Protection на базе открытого ПО [Velero](#).

Velero поддерживает резервное копирование и восстановление пользовательских ресурсов Kubernetes на уровне пространств имен, а с помощью решения [Kopia](#) позволяет также осуществлять резервное копирование файловой системы, используемой приложениями и сервисами.

### 1.3.1. Основные требования

Для настройки и использования модуля Data Protection вам необходимо иметь учетную запись в кластере Kubernetes с правами администратора, а также доступ к какому-либо S3-совместимому объектному хранилищу, например:

- Объектное хранилище [K2 Cloud](#)
- Объектное хранилище [Yandex Cloud](#)
- Частные решения на базе [MinIO](#) и [Ceph](#)

Для подробной информации ознакомьтесь с разделом [Защита пользовательских данных с помощью модуля Data Protection](#)

### 1.3.2. Системные требования

Общие требования по вычислительным ресурсам можно найти [в статье](#)

## 2. Содержание раздела

---

- Резервное копирование мастер-узлов
- Восстановление данных на мастер-узлах
- Защита пользовательских данных с помощью модуля Data Protection





# Веб-консоль

Веб-консоль Nova Container Platform это один из основных инструментов работы наравне с `kubectl`.

Она предоставляет возможность работы с кubernetes инженеру, который ранее не работал с данной технологией.

## 1. Содержание раздела

---

- [ConsoleLinks](#)
- [ConsoleYAMLSample](#)