

Общество с ограниченной ответственностью «Гиперион»

ОГРН 1237700921083 ИНН 9731127087

**Программное обеспечение
«Гиперион.КУБ»**

**Документация, содержащая информацию,
необходимую для эксплуатации ПО**

1 Общие положения	2
1.1 Полное наименование системы и ее обозначение	2
1.2 Назначение системы	2
1.3 Цель развития	2
2 Ограничение программной среды функционирования	3
3 Регламентные работы.....	4
3.1 Отслеживание доступности ПО	4
3.1.1 Компоненты системы отслеживания доступности.....	4
3.1.2 Объекты отслеживания доступности	5
3.2 Восстановление доступности ПО	5
3.2.1 Отключение кластера.....	5
3.2.2 Восстановление кластера	6
3.2.3 Перезапуск кластера после восстановления	6

1 Общие положения

1.1 Полное наименование системы и ее обозначение

Полное наименование: программное обеспечение «Гиперион.КУБ».

Сокращенное наименование в тексте: ПО «Гиперион.КУБ».

1.2 Назначение системы

ПО «Гиперион.КУБ» предназначено для оркестровки контейнерных приложений как в окружении bare-metal, так и в средах виртуализации, без предварительно установленной операционной системы.

1.3 Цель развития

Целью развития ПО «Гиперион.КУБ» является обеспечение непрерывного функционирования. Основными целями развития ПО «Гиперион.КУБ» являются:

- развитие внутренних функциональных возможностей ПО «Гиперион.КУБ»;
- улучшение технических характеристик работы ПО «Гиперион.КУБ»;
- повышение удобства пользователей при работе с ПО «Гиперион.КУБ».

2 Ограничение программной среды функционирования

ПО «Гиперион.КУБ» может функционировать в следующих режимах:

Штатный режим, в котором система выполняет свои основные функции, является основным режимом функционирования ПО «Гиперион.КУБ», при котором:

- ПТК серверов обеспечивает возможность круглосуточного функционирования с перерывами на обслуживание;
- исправно работает оборудование, составляющее комплекс технических средств.

Сервисный режим для проведения планового обслуживания, реконфигурации, замены и пополнения новыми компонентами, установки обновленных версий программного обеспечения:

- установка, настройка и мониторинг работоспособности системного программного обеспечения;
- инсталляция и настройка прикладного программного обеспечения;
- оптимизация функционирования прикладных баз данных по времени отклика, скорости доступа к данным;
- аварийное восстановление данных;
- конфигурирование и настройка программно-технических средств;
- модернизация, настройка и мониторинг работоспособности комплекса технических средств;
- настройка локальной компьютерной сети и сетевого окружения.

3 Регламентные работы

3.1 Отслеживание доступности ПО

Для отслеживания доступности используется стек компонентов ПО «Гиперион.КУБ», который позволяет своевременно выявить проблемы функционирования как самого ПО, так и развернутых в кластере ПО контейнерных приложений.

ПО «Гиперион.КУБ» содержит следующие средства отслеживания доступности и принятия решения при нарушении доступности:

- метрики;
- уведомления;
- панель мониторинга.

3.1.1 Компоненты системы отслеживания доступности

В конфигурации по умолчанию ПО «Гиперион.КУБ» задействует для отслеживания доступности следующие компоненты:

- Оператор мониторинга кластера (Cluster Monitoring Operator) – основной компонент стека отслеживания доступности, который разворачивает, поддерживает функционирование и обеспечивает обновление экземпляров приложений, входящих в стек: сервера сбора метрик, Prometheus, Alertmanager и Thanos Querier.
- Сервер сбора метрик – инструмент, который собирает ресурсные метрики, организует их и предоставляет компонентам стека.
- Prometheus – система мониторинга, которая содержит базу данных метрик, а также набор проверок для оценки поступающих метрик и принятия решения. При нарушении доступности (нештатном результате проверки метрик) Prometheus задействует компонент Alertmanager для отправки уведомлений.
- Alertmanager – инструмент отправки уведомлений как внутри кластера (компонентам ПО «Гиперион.КУБ»), так и за пределами кластера (внешним системам уведомлений).
- Thanos Querier – инструмент агрегации и резервирования метрик.

3.1.2 Объекты отслеживания доступности

В конфигурации по умолчанию ПО «Гиперион.КУБ» отслеживает доступность следующих объектов:

- CoreDNS;
- etcd;
- HAProxy;
- Image registry;
- Kubelets;
- Hyperkube API server;
- Hyperkube controller manager;
- Hyperkube scheduler;
- Lifecycle Manager.

3.2 Восстановление доступности ПО

Для восстановления кластера или проведения технических работ необходимо отключить его, выполнить действия по обслуживанию, после чего перезапустить кластер.

3.2.1 Отключение кластера

Отключение кластера позволяет выполнить действия по его обслуживанию и перезапустить. Если между установкой и перезапуском ПО прошло менее года, дополнительных действий не требуется. При отключении кластера на длительный период необходимо установить дату истечения срока действия сертификатов.

Перед тем, как отключить кластер, необходимо убедиться, что соблюдены следующие требования:

- получен доступ к кластеру с полномочиями администратора;
- создана резервная копия etcd.

Чтобы отключить кластер, выполните следующие шаги:

1. Установите дату истечения срока действия сертификатов, если это требуется:

```
$ oc -n openshift-kube-apiserver-operator get secret kube-apiserver-to-kubelet-signer -o jsonpath='{.metadata.annotations.auth\.openshift\.io/certificate-not-after}'
```
2. Последовательно отключите все узлы кластера:

```
$ for node in $(oc get nodes -o jsonpath='{.items[*].metadata.name}'); do  
oc debug node/${node} -- chroot /host shutdown -h 1; done
```

3. Отключите элементы окружения, которые более не задействуются, такие как внешнее хранилище или сервер LDAP.

3.2.2 Восстановление кластера

Для создания резервных копий и последующего восстановления используется инструмент `hkdp`, который сохраняет ресурсы и образы кластера на уровне пространства имен (namespace) в виде persistent volumes (PV) – объектов, совместимых с объектными хранилищами.

3.2.3 Перезапуск кластера после восстановления

Чтобы перезапустить кластер, выполните следующие шаги:

1. Подключите элементы окружения, которые задействуются в работе кластера, такие как внешнее хранилище или сервер LDAP.
2. Запустите узлы кластера. Подождите 10 минут, чтобы у всех узлов было достаточно времени для полного запуска и загрузки.
3. Проверьте готовность узлов управления:
`oc get nodes -l node-role.kubernetes.io/master`

Примечание: если узлы управления запущены штатно и готовы к работе, рядом с каждым узлом отобразится статус «Ready».

4. Если узлы управления не готовы к работе, проверьте текущее состояние сертификатов:
 - 4.1. Запросите список сертификатов:
`$ oc get csr`
 - 4.2. Проверьте действительность каждого сертификата:
`$ oc describe csr <csr_name>`
 - 4.3. Подтвердите использование действительных сертификатов:
`$ oc adm certificate approve <csr_name>`

5. Проверьте готовность worker-узлов:
`$ oc get nodes -l node-role.kubernetes.io/worker`
6. Проверьте, что запуск кластера прошел штатно:
`$ oc get clusteroperators`

Примечание: при штатном запуске у операторов кластера не должно быть признаков «Degraded» со значением «True».

7. Проверьте готовность (статус «Ready») всех узлов кластера:
`$ oc get nodes`