# **ПРОТОКОЛ ТЕСТИРОВАНИЯ СОВМЕСТИМОСТИ С KESL 12**

# **Deckhouse** на ОС «**Astra**»

**1. Общая информация**

Тестировал: Сабиров Артур

E-mail: artur.sabirov@flant.ru

**2. Описание дистрибутива и прочего ПО**

1) ОС Astra Linux Special Edition 1.7 «Воронеж» из репозитория Яндекс.Клауд

Image\_id: fd8i7rqnc7782665c4eq

Family\_id: astralinux-alse

uname -a

Linux kesl12-artur-astra-master-0 6.1.50-1-generic #astra2+ci6 SMP PREEMPT\_DYNAMIC Fri Oct 6 14:38:42 UTC 2023 x86\_64 GNU/Linux

cat /etc/astra\_version

1.7.5



2) Наименование тестируемого ПО: Deckhouse v1.57.4 EE

3) Наименование тестируемого ПО: Kaspersky Endpoint Security 12.0 для Linux

Версия: 12.0.0.6064

**3. Предварительные условия**

1) Astra Linux ставится с профилем по умолчанию, диск разбивается автоматически

2) Создание пользователя с аутентификации на базе ключей SSH

3) Наличия лицензионного ключа

4) Установка выполняется согласно [Быстрому старту](https://deckhouse.ru/gs/bm/step2.html) и [Документации](https://deckhouse.ru/documentation/v1/deckhouse-overview.html)

5) Установка KESL согласно [инструкции](https://support.kaspersky.ru/kes-for-linux/12.0/248501)

**4. До начала проведения испытаний**

а) Установка любой программы выполняется на чистой системе только после обновления системы и ядра до актуального состояния:

****$ su-

# apt-get update

# apt-get dist-upgrade

# reboot

****

**5. Настройки KESL**

kesl-control --app-info

Название: Kaspersky Endpoint Security 12.0 для Linux

Версия: 12.0.0.6064

Политика: Не применена

Информация о лицензии: Ключ действителен

Статус файла MDR BLOB: Не загружен

Состояние Хранилища: Самый старый объект будет удален 2024-05-19 08:28:17

Использование Хранилища: Размер Хранилища не ограничен

Дата последнего запуска задачи Scan\_My\_Computer: Никогда не выполнялась

Дата последнего выпуска баз приложения: 2024-02-19 07:51:00

Базы приложения загружены: Да

Использование Kaspersky Security Network: Расширенный режим KSN

Инфраструктура Kaspersky Security Network: Kaspersky Security Network

Состояние Managed Detection and Response: Неактивный

Защита от файловых угроз: Задача доступна и выполняется

Мониторинг контейнеров: Задача доступна и не выполняется

Контроль целостности системы: Задача доступна и не выполняется

Управление сетевым экраном: Задача доступна и не выполняется

Защита от шифрования: Задача доступна и не выполняется

Защита от веб-угроз: Задача доступна и не выполняется

Контроль устройств: Задача доступна и выполняется

Проверка съемных дисков: Задача доступна и не выполняется

Защита от сетевых угроз: Задача доступна и не выполняется

Анализ поведения: Задача доступна и выполняется

Контроль приложений: Задача доступна и не выполняется

Интеграция с Kaspersky Endpoint Detection and Response (KATA): Задача доступна и не выполняется

Состояние обновления: Нет доступных обновлений для приложения



**Открытие порта для работы kubeapi**

kesl-control --add-rule --name kube\_api\_in --action allow --direction in --protocol TCP --local any:6443 && kesl-control --add-rule --name kube\_api\_out --action allow --direction out --protocol TCP --local any:6443



**6. Установочные файлы Deckhouse**

**config.yml**

https://deckhouse.ru/documentation/v1/installing/configuration.html#clusterconfiguration

apiVersion: deckhouse.io/v1

kind: ClusterConfiguration

clusterType: Static

podSubnetCIDR: 10.111.0.0/16

serviceSubnetCIDR: 10.222.0.0/16

kubernetesVersion: "Automatic"

clusterDomain: "cluster.local"

---

apiVersion: deckhouse.io/v1

kind: InitConfiguration

deckhouse:

imagesRepo: registry.deckhouse.ru/deckhouse/ee

registryDockerCfg: \*скрыто\*

---

https://deckhouse.ru/documentation/v1/modules/002-deckhouse/configuration.html

apiVersion: deckhouse.io/v1alpha1

kind: ModuleConfig

metadata:

name: deckhouse

spec:

version: 1

enabled: true

settings:

bundle: Default

releaseChannel: Stable

logLevel: Info

---

apiVersion: deckhouse.io/v1alpha1

kind: ModuleConfig

metadata:

name: global

spec:

version: 1

settings:

modules:

publicDomainTemplate: "%s.test.kesl"

---

apiVersion: deckhouse.io/v1alpha1

kind: ModuleConfig

metadata:

name: user-authn

spec:

version: 1

enabled: true

settings:

controlPlaneConfigurator:

dexCAMode: DoNotNeed

publishAPI:

enable: true

https:

mode: Global

global:

kubeconfigGeneratorMasterCA: ""

---

https://deckhouse.ru/documentation/v1/modules/021-cni-cilium/configuration.html

apiVersion: deckhouse.io/v1alpha1

kind: ModuleConfig

metadata:

name: cni-cilium

spec:

version: 1

enabled: true

settings:

tunnelMode: VXLAN

---

apiVersion: deckhouse.io/v1

kind: StaticClusterConfiguration

internalNetworkCIDRs:

- 172.16.16.0/24



**Resources.yml**

apiVersion: deckhouse.io/v1

kind: NodeGroup

metadata:

name: system

spec:

nodeTemplate:

labels:

node-role.deckhouse.io/system: ""

taints:

- effect: NoExecute

key: dedicated.deckhouse.io

value: system

nodeType: Static

---

apiVersion: deckhouse.io/v1

kind: NodeGroup

metadata:

name: worker

spec:

nodeType: Static

---

apiVersion: deckhouse.io/v1alpha1

kind: LocalPathProvisioner

metadata:

name: localpath-deckhouse

spec:

nodeGroups:

- system

path: "/opt/local-path-provisioner"

---

apiVersion: deckhouse.io/v1

kind: IngressNginxController

metadata:

name: nginx

spec:

ingressClass: nginx

inlet: HostPort

hostPort:

httpPort: 80

httpsPort: 443

nodeSelector:

node-role.kubernetes.io/system: ""

tolerations:

- operator: Exists

---

apiVersion: deckhouse.io/v1

kind: ClusterAuthorizationRule

metadata:

name: admin

spec:

subjects:

- kind: User

name: admin@test.kesl

accessLevel: SuperAdmin

portForwarding: true

---

apiVersion: deckhouse.io/v1

kind: User

metadata:

name: admin

spec:

email: admin@test.kesl

password: \*скрыто\*



**7. Результаты проведённых испытаний**

[**test\_01**](https://docs.google.com/document/d/1JsjSKknAbBMZkVbj_5x0wHlBF_1rdBea_AQaAul3l1I/edit#bookmark=id.gjdgxs)**: *Проверка запуска всех нод***

Цель данного теста: убедиться в работоспособности основных компонентов

| **#** | **Выполняемое действие** | **Ожидаемый результат** | **Шаг** |
| --- | --- | --- | --- |
| 1. | Выполнение команды с правами root:  kubectl get pods -A   | Все поды в находятся в статусе Running | **ПРОЙДЕН** |

**test\_02: *Проверка очереди задач***

Цель данного теста: убедиться в отсутствии задач застрявших в очереди

| **#** | **Выполняемое действие** | **Ожидаемый результат** | **Шаг** |
| --- | --- | --- | --- |
| 1. | Выполнение команды с правами root:  kubectl -n d8-system exec -it deploy/deckhouse -- deckhouse-controller queue list | grep "'main' queue"   | Очередь задач пустая:  Defaulted container "deckhouse" out of: deckhouse, kube-rbac-proxy, init-external-modules (init)  - 'main' queue: empty.   | **ПРОЙДЕН** |
|  | **ИТОГОВЫЙ ВЫВОД:** | **Все шаги пройдены без ошибок.** | **ПРОЙДЕН** |

**test\_03: *Проверка отсутствия ошибок***

Цель данного теста: убедиться в отсутствии задач застрявших в очереди

| **#** | **Выполняемое действие** | **Ожидаемый результат** | **Шаг** |
| --- | --- | --- | --- |
| 1. | Выполнение команды с правами root:  kubectl -n d8-system logs deploy/deckhouse | jq -r 'select(.module != null) | select(.level == "error")'   | Ошибок нет. | **ПРОЙДЕН** |
|  | **ИТОГОВЫЙ ВЫВОД:** | **Все шаги пройдены без ошибок.** | **ПРОЙДЕН** |

**test\_04: *Проверка отсутствия ошибок в ивентах***

Цель данного теста: убедиться в отсутствии задач застрявших в очереди

| **#** | **Выполняемое действие** | **Ожидаемый результат** | **Шаг** |
| --- | --- | --- | --- |
| 1. | Выполнение команды с правами root:  kubectl get event -A   | Ошибок нет. | **ПРОЙДЕН** |
|  | **ИТОГОВЫЙ ВЫВОД:** | **Все шаги пройдены без ошибок.** | **ПРОЙДЕН** |

**test\_05: *Проверка алертов***

Цель данного теста: убедиться в отсутствии задач застрявших в очереди

| **#** | **Выполняемое действие** | **Ожидаемый результат** | **Шаг** |
| --- | --- | --- | --- |
| 1. | Выполнение команды с правами root:  kubectl get clusteralerts   | Присутствует только алерт “DeadMansSwitch”.  NAME ALERT SEVERITY AGE LAST RECEIVED STATUS  ab17837fffa5e440 DeadMansSwitch 4 5m55s 54s firing   | **ПРОЙДЕН** |
|  | **ИТОГОВЫЙ ВЫВОД:** | **Все шаги пройдены без ошибок.** | **ПРОЙДЕН** |

**test\_06: *Проверка работы веб-интерфейсов***

Цель данного теста: убедиться в работоспособности веб-интерфейсов

| **#** | **Выполняемое действие** | **Ожидаемый результат** | **Шаг** |
| --- | --- | --- | --- |
| 1. | Переход в браузере по ссылке на веб-сайт:  https://grafana.domain.my/  (domain.my - необходимо заменить на URL кластера) | Ошибок нет. | **ПРОЙДЕН** |
| 2. | Переход в браузере по ссылке на веб-сайт:  https://dashboard.domain.my/  (domain.my - необходимо заменить на URL кластера) | Ошибок нет. | **ПРОЙДЕН** |
| 3. | Переход в браузере по ссылке на веб-сайт:  https://documentation.domain.my/  (domain.my - необходимо заменить на URL кластера) | Ошибок нет. | **ПРОЙДЕН** |
| 4. | Переход в браузере по ссылке на веб-сайт:  https://grafana.domain.my/prometheus  (domain.my - необходимо заменить на URL кластера) | Ошибок нет. | **ПРОЙДЕН** |
| 5. | Переход в браузере по ссылке на веб-сайт:  https://status.domain.my/  (domain.my - необходимо заменить на URL кластера) | Ошибок нет - все пункты в статусе “Operational”. | **ПРОЙДЕН** |
| 6. | Переход в браузере по ссылке на веб-сайт:  https://upmeter.domain.my/  (domain.my - необходимо заменить на URL кластера) | Ошибок нет. | **ПРОЙДЕН** |
|  | **ИТОГОВЫЙ ВЫВОД:** | **Все шаги пройдены без ошибок.** | **ПРОЙДЕН** |

**8. Выводы на основании проведенных испытаний**

**Основные функции работают**, установка дополнительных пакетов из репозитория не потребовалась, удаление выполняется без ошибок. Проблем в работе и взаимодействии компонентов продукта не обнаружено. Обновление компонентов ПО и ОС проходит без сбоев, целостность ОС не нарушается. ***Deckhouse*** **полностью совместим** с KESL 12.