

?????????????? ?? ?????????? KDT

Дата обновления: 14.07.2026

KDT (Kaspersky Deployment Tool) - это основной инструмент командной строки для установки, обновления, настройки и обслуживания компонентов Kaspersky XDR Expert в кластере Kubernetes.

1. ??????? ? ?????????????? ????????????

Прежде чем запускать команды, важно понимать, как работает KDT. Глобальные флаги указываются **перед** основной командой:

```
./kdt [ГЛОБАЛЬНЫЕ_ФЛАГИ] <команда> [ФЛАГИ_КОМАНДЫ]
```

Полезные глобальные флаги:

- `-v` (`--version`): Показать версию и завершить работу.
- `-p` (`--keep-log`): Сохранить временные файлы логов (полезно для отладки).
- `--no-progress`: Скрыть индикатор прогресса (удобно для CI/CD пайплайнов).

Важные особенности:

1. **Предварительные проверки:** Перед выполнением большинства команд KDT проверяет аппаратное обеспечение, ОС, сеть и наличие установленного Bootstrap.
2. **Коды возврата:**
 - `0` — успех.
 - `1` — общая ошибка или провал валидации.
 - `2` — прерывание по сигналу SIGINT (Ctrl+C). *Обратите внимание: в KDT используется код `2`, а не стандартный `130`.*
 - `12` — ошибка драйвера выполнения контейнеров.

2. ?????????? ?????????????????????? ????????????????????

Рекомендуемый путь установки состоит из двух этапов: генерация конфигурации и ее применение.

??? 1: ?????????????????? ??????? (`wizard`)

Вместо ручного написания YAML-файла используйте встроенный мастер. Он задаст все необходимые вопросы: от топологии узлов (primary/worker) и SSH-ключей до строки подключения к PostgreSQL и требований к паролю администратора.

```
./kdt wizard -k xdr-2.1.tar -o deployment.yaml
```

Требования к паролю администратора XDR: 10-256 символов, минимум 3 категории (верхний/нижний регистр, цифры, спецсимволы), без пробелов и комбинации `[@]`.

??? 2: ?????????? ?????????????? (`apply`)

После генерации файла запустите развертывание. Флаг `--accept-eula` позволяет выполнить установку в полностью автоматическом (неинтерактивном) режиме.

```
./kdt apply -k xdr-2.1.tar -i deployment.yaml --accept-eula
```

3. ?????????????? ?????????????????????

После успешной установки вам потребуются команды для управления состоянием системы.

????????? ?????????? ?

- **Список компонентов:** `./kdt state --list` (или короткий алиас `./kdt s --list`). Добавьте `--with-deps`, чтобы увидеть зависимости.
- **Детальный лог компонента:** `./kdt state --log backup-operator`
- **Получение логов контейнеров:**

```
./kdt logs get --app kuma --last 24h --destination ./kuma_logs.txt  
# или короткий алиас: ./kdt l --app kuma --last 24h
```

????????????? ?????????? ??? ?????????????? (`invoke`)

Команда `invoke` позволяет вызывать специфичные для компонентов действия (например, изменение размера дисков, настройка бэкапов, смена IP).

Примеры:

```
# Настроить NFS-хранилище для бэкапов  
./kdt invoke backup-operator -a setBackupStore --param  
backup_target_url="nfs://storage.local:/backups"  
  
# Создать ручной бэкап с хранением 5 копий  
./kdt invoke backup-operator -a createBackup --param backup_retain=5
```

```
# Изменить ingress-IP шлюза
./kdt invoke gateway -a changeIngressIP --param ingress_ip="192.168.1.100"

# Увеличить диск KUMA Core
./kdt invoke kuma -a setCorePvcSize --param core_disk_request="500Gi"
```

Чтобы узнать, какие действия доступны для компонента, используйте: `./kdt describe <имя_компонента>` (алиас `./kdt d`).

????????? ???????????? ? ?????????????????? ?????????? (state)

KDT позволяет делать снапшоты внутренней базы данных состояний:

```
# Создание бэкапа состояния
./kdt state --backup state_backup.tar

# Восстановление состояния (с опциональной очисткой старых данных)
./kdt state --restore --from-backup state_backup.tar --wipe

# Ротация ключа шифрования statedb
./kdt state --rotate-secret
```

4. ?????? ? ?????????????????? ?????????? (Air-Gap)

Для сред без доступа к интернету используется отдельная утилита `kdt-airgap-preparation`. Процесс состоит из трех четких этапов:

1. **Сканирование (scan)**: Выполняется на целевом изолированном узле. Создает файл инвентаризации с перечнем недостающих пакетов.

```
kdt-airgap-preparation scan -k ~/.ssh/id_rsa -c config.yaml --airgap -i nodes-airgap.yaml
```

2. **Загрузка (pull)**: Файл `nodes-airgap.yaml` переносится на машину с интернетом. Утилита скачивает все зависимости и упаковывает их в TAR-архив.

```
kdt-airgap-preparation pull -n nodes-airgap.yaml -a airgap-packages.tar
```


KDT_YES	(не задана)	Установите <code>1</code> для автоматического подтверждения всех запросов (идеально для Ansible/CI/CD).
KDT_HOME	<code>~/.kdt</code>	Позволяет изменить домашнюю директорию утилиты.
KDT_PARALLELS_LIMIT	<code>-1</code> (авто)	Ограничение на количество параллельных операций.

Рекомендация: Всегда используйте команду `./kdt describe` перед вызовом новых действий через `invoke`, чтобы убедиться в правильности передаваемых параметров, и регулярно экспортируйте конфигурацию (`export-config`) после внесения значимых изменений в кластер.