Гайд по развертыванию Cross-DC Kaspersky EDR Expert (on-premise)

> ?????????? ?????????? ?? ?????? ??????????? ???????????? KEDR 8.3 ? ????? ????????????.
> ???? ???????????: 12.10.2026

1. Архитектура и требования к ресурсам

1.1. Выбор модели

- **[3DC] 3 полноценных ЦОДа:** ???????????? ???????????. Storage: 1 ???? × 3 ???????. K8s Workers: 3+3+3. HAProxy: 3 ????.
- **[2.5DC] Модель «2,5 ЦОДа»:** ?????????????. ??? 3 - ?????? ????????. Storage: 1 ???? × 4 ??????? (2+2+0). K8s Workers: 3+3+1. HAProxy: 2 ????.

1.2. Сетевые требования

- **Ingress IP** (???? ????????) ?????? ???? ? **той же подсети**, ??? ? ???? Primary/Worker ? ?????? ????.
- ??? ???? Kubernetes ?????? ?????? ???? ?????? ?????????? ? **одном широковещательном домене (L2)**.
- ???????? ????? ??????: **≤ 20 мс**.

1.3. Настройки гипервизора

??? ?????????? ?????? ???????? Kubernetes ? ???????????? ??????? (MetalLB) ?? ?????? ??????????? ??? ???????? ??????????? ??? ???? ??????? ?????? **необходимо отключить защиту от ARP/MAC-спуфинга**:

- **VMware ESXi:** ? ?????????? vSwitch ??? Port Group ???????? **Forged Transmits** ? **MAC Address Changes** (Accept).
- **Hyper-V:** ???????? **MAC Address Spoofing** ??? ??????????? ??????? ?????????.
- **Физические коммутаторы:** ????????? Dynamic ARP Inspection (DAI) ?? ?????? ??? ????????? ?? ??? ?????????? (trusted).

---

2. Подготовка операционной системы (Все узлы)

????????? ??? ????????????? **root** ??? **<USER_NAME>** ? ??????? **sudo NOPASSWD**.

```
# 1. ?????????? ? ??????? ??????
apt update && apt upgrade -y
apt install -y python3 python3-apt curl systemd-timesyncd net-tools jq nfs-common tar wireguard wireguard-tools
libnfs12 open-iscsi cryptsetup cryptsetup-bin netcat-traditional containerd python3-cryptography python3-cffi-
backend python3-netaddr

# 2. ???????? cgroup v2 (???????????)
mount | grep cgroup # ????????? ?????: cgroup2 on /sys/fs/cgroup type cgroup2

# 3. ?????????? swap
swapoff -a
sed -i '/ swap / s/^\(.*\)$/#\1/g' /etc/fstab

# 4. ?????????? SELinux (???? ??????????)
setenforce 0
sed -i 's/^SELINUX=enforcing/SELINUX=disabled/' /etc/selinux/config

# 5. ????????????? ??????? (??????? ? 5 ??????)
apt install -y chrony
systemctl enable chrony && systemctl start chrony
timedatectl set-ntp true
timedatectl set-timezone Europe/Moscow
timedatectl set-local-rtc 0

# 6. ???????? ???????????? ??? KDT
useradd -m -s /bin/bash <USER_NAME>
echo "<USER_NAME> ALL=(ALL) NOPASSWD: ALL" | sudo tee /etc/sudoers.d/<USER_NAME>
chmod 440 /etc/sudoers.d/<USER_NAME>

# 7. ????????? UFW (???? ????????????)
ufw allow 22/tcp
ufw allow 6443/tcp       # Kubernetes API
ufw allow 2379:2380/tcp  # etcd
ufw allow 10250/tcp      # Kubelet
ufw allow 5432/tcp       # PostgreSQL
ufw allow 7210/tcp       # KUMA Core API
ufw allow 7220:7240/tcp  # KUMA ???????
ufw allow 8123/tcp       # ClickHouse
ufw allow 9443/tcp
ufw allow 13000/tcp      # ????????? ??????
ufw allow 80/tcp         # HTTP redirect
ufw allow 514/tcp        # Syslog (???????????, ?? ?????????????)
sed -i 's/#net.ipv4.ip_forward=1/net.ipv4.ip_forward=1/' /etc/sysctl.conf
sysctl -p
# ??? Primary ????? K8s ????????? IP-?????????????:
sed -i 's/DEFAULT_FORWARD_POLICY="DROP"/DEFAULT_FORWARD_POLICY="ACCEPT"/' /etc/default/ufw
ufw reload
```

---

3. Кластер PostgreSQL

3.1. Установка (на всех 3 узлах БД)

```
apt update
apt install -y postgresql-16 patroni etcd etcd-client
systemctl stop postgresql
systemctl disable postgresql
```

3.2. Конфигурация Patroni (/etc/patroni/patroni.yml)

```
scope: xdr_cluster
namespace: /service/
name: <???_????> # db-dc1 / db-dc2 / db-dc3
```

```
restapi:
  listen: <IP_?????_????>:8008
  connect_address: <IP_?????_????>:8008

etcd:
  hosts: <DB_DC1_IP>:2379,<DB_DC2_IP>:2379,<DB_DC3_IP>:2379

bootstrap:
  dcs:
    ttl: 30
    loop_wait: 10
    retry_timeout: 10
    maximum_lag_on_failover: 1048576
    synchronous_mode: true
    synchronous_mode_strict: true
    postgresql:
      use_pg_rewind: true
      parameters:
        listen_addresses: '*'
        port: 5432
        max_connections: 512
        shared_buffers: 8192MB
        temp_buffers: 24MB
        work_mem: 16MB
        maintenance_work_mem: 128MB
        max_stack_depth: 7MB
        max_parallel_workers_per_gather: 0
        standard_conforming_strings: on
        log_hostname: 1
        synchronous_commit: "on"

postgresql:
  listen: <IP_?????_????>:5432
  connect_address: <IP_?????_????>:5432
  data_dir: /var/lib/postgresql/16/main
  bin_dir: /usr/lib/postgresql/16/bin
  pgpass: /tmp/pgpass
  authentication:
    replication:
      username: replicator
      password: <REPLICATOR_PASSWORD>
    superuser:
      username: postgres
      password: <POSTGRES_PASSWORD>
```

Специфика для [2.5DC] (Узел в ЦОД 3 — Witness): ? ?????? **tags** ????????:

```
tags:
  nofailover: true
  noloadbalance: true
  nosync: true
```

(??? [3DC] ???????? tags ??????? ??? ?? ?????????? false).

3.3. Запуск и настройка прав

```
systemctl enable patroni && systemctl start patroni
patronictl -c /etc/patroni/patroni.yml list

# ???????? ???????????? (????????? ?? Leader-???? ????? psql)
sudo -u postgres psql -h <DB_VIP_IP> -p 5432 <<EOF
CREATE USER <kaspersky_admin> WITH PASSWORD '<StrongPassword123!>' SUPERUSER;
CREATE DATABASE <kaspersky_admin> OWNER <kaspersky_admin>;
```

```
GRANT pg_read_all_stats TO "<kaspersky_admin>";
EOF
```

3.4. pg_hba.conf (на всех узлах БД)

```
echo "host all all 0.0.0.0/0 scram-sha-256" | tee -a /etc/postgresql/16/main/pg_hba.conf
echo "host replication replicator 0.0.0.0/0 scram-sha-256" | tee -a /etc/postgresql/16/main/pg_hba.conf
sudo -u postgres psql -c "SELECT pg_reload_conf();"
```

---

4. Балансировщик (HAProxy + Keepalived)

4.1. Установка и Keepalived

```
apt update && apt install -y haproxy keepalived
```

4.2. Keepalived (/etc/keepalived/keepalived.conf)

[3DC] ??? ???? ? ???????????? 101 / 100 / 99.

[2.5DC] ??? ???? ? ???????????? 101 (MASTER, ??? 1) ? 100 (BACKUP, ??? 2).

```
vrrp_script chk_haproxy {
    script "killall -0 haproxy"
    interval 2
    weight 2
}

vrrp_instance VI_1 {
    state MASTER        # BACKUP ?? ????????? ?????
    interface <???_??????????>
    virtual_router_id 51
    priority 101        # 100 / 99 ?? ?????????
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass <KEEPALIVED_AUTH_PASSWORD>
    }
    virtual_ipaddress {
        <HAPROXY_VIP_IP>
    }
    track_script {
        chk_haproxy
    }
}


systemctl enable keepalived && systemctl start keepalived
# ????????: ip a | grep <HAPROXY_VIP_IP>
```

4.3. HAProxy (/etc/haproxy/haproxy.cfg)

```
global
    log /dev/log local0
    log /dev/log local1 notice
    chroot /var/lib/haproxy
    stats socket /run/haproxy/admin.sock mode 660 level admin
    stats timeout 30s
    user haproxy
    group haproxy
    daemon

defaults
    log global
    mode tcp
```

```
    option tcplog
    option dontlognull
    timeout connect 5s
    timeout client 50s
    timeout server 50s

frontend http_front
    bind *:80
    default_backend http_back
frontend https_front
    bind *:443
    default_backend https_back
frontend https9_front
    bind *:9443
    default_backend https9_back
frontend https4_front
    bind *:4443
    default_backend https4_back
frontend agent_front
    bind *:13000
    default_backend agent_back

backend http_back
    option tcp-check
    timeout check 2s
    server dc1_ingress <INGRESS_IP_DC1>:80 check inter 5s
    server dc2_ingress <INGRESS_IP_DC2>:80 check inter 5s
    server dc3_ingress <INGRESS_IP_DC3>:80 check inter 5s

backend https_back
    option tcp-check
    timeout check 2s
    server dc1_ingress <INGRESS_IP_DC1>:443 check inter 5s
    server dc2_ingress <INGRESS_IP_DC2>:443 check inter 5s
    server dc3_ingress <INGRESS_IP_DC3>:443 check inter 5s

backend https9_back
    option tcp-check
    timeout check 2s
    server dc1_ingress <INGRESS_IP_DC1>:9443 check inter 5s
    server dc2_ingress <INGRESS_IP_DC2>:9443 check inter 5s
    server dc3_ingress <INGRESS_IP_DC3>:9443 check inter 5s

backend https4_back
    option tcp-check
    timeout check 2s
    server dc1_ingress <INGRESS_IP_DC1>:4443 check inter 5s
    server dc2_ingress <INGRESS_IP_DC2>:4443 check inter 5s
    server dc3_ingress <INGRESS_IP_DC3>:4443 check inter 5s

backend agent_back
    option tcp-check
    timeout check 2s
    server dc1_ingress <INGRESS_IP_DC1>:13000 check inter 5s
    server dc2_ingress <INGRESS_IP_DC2>:13000 check inter 5s
    server dc3_ingress <INGRESS_IP_DC3>:13000 check inter 5s


systemctl enable haproxy && systemctl restart haproxy
```

4.4. DNS

?? ????????????? DNS-??????? ????????? A-??????, ??????????? ?? **VIP балансировщика** (**<HAPROXY_VIP_IP>**):

```
console.<smp_domain>
admsrv.<smp_domain>
api.<smp_domain>
monitoring.<smp_domain>
updater.<smp_domain>
agentserver.<smp_domain>
kuma.<smp_domain>
*.kuma.<smp_domain>   ? wildcard ??????????
```

---

5. Подготовка устройства администратора

```
# 1. ????????? Docker
apt update && apt install -y ca-certificates curl gnupg lsb-release
mkdir -p /etc/apt/keyrings
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
echo "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg]
https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" | sudo tee
/etc/apt/sources.list.d/docker.list > /dev/null
apt update && apt install -y docker-ce docker-ce-cli containerd.io docker-compose-plugin
usermod -aG docker <USER_NAME>
newgrp docker

# 2. ????????? SSH-??????
su - <USER_NAME>
ssh-keygen -t rsa -b 4096 -f ~/.ssh/id_rsa -N ""
# ??????????? ?? ??? ???? (K8s, DB, KUMA, HAProxy):
ssh-copy-id <USER_NAME>@<TARGET_IP>
```

---

6. Конфигурационные файлы для KDT

6.1. Файл инвентаря KUMA (kuma_distributed_inventory.yml)

??????? ?? Admin Node.

[3DC]

```
all:
  vars:
    generate_etc_hosts: false
    deploy_example_services: false
    ansible_connection: local
    ansible_user: nonroot

kuma:
  vars:
    ansible_connection: ssh
    ansible_user: "<USER_NAME>"
    ansible_become: true
    ansible_become_method: sudo
  children:
    kuma_utils:
      hosts:
        kuma-utils.local:
          ansible_host: <ADMIN_NODE_IP>

    kuma_collector:
      hosts:
        kuma-coll-dc1.local:
          ansible_host: <KUMA_COLL_DC1_IP>
        kuma-coll-dc2.local:
          ansible_host: <KUMA_COLL_DC2_IP>
```

```
        kuma-coll-dc3.local:
          ansible_host: <KUMA_COLL_DC3_IP>

    kuma_correlator:
      hosts:
        kuma-corr-dc1.local:
          ansible_host: <KUMA_CORR_DC1_IP>
        kuma-corr-dc2.local:
          ansible_host: <KUMA_CORR_DC2_IP>
        kuma-corr-dc3.local:
          ansible_host: <KUMA_CORR_DC3_IP>

    kuma_storage:
      hosts:
        kuma-stor-dc1.local:
          ansible_host: <KUMA_ST_DC1_IP>
          shard: 1
          replica: 1
          keeper: 1
        kuma-stor-dc2.local:
          ansible_host: <KUMA_ST_DC2_IP>
          shard: 1
          replica: 2
          keeper: 2
        kuma-stor-dc3.local:
          ansible_host: <KUMA_ST_DC3_IP>
          shard: 1
          replica: 3
          keeper: 3

    kuma_keeper:
      hosts:
        kuma-keep-dc1.local:
          ansible_host: <KUMA_KP_DC1_IP>
          keeper_id: 1
        kuma-keep-dc2.local:
          ansible_host: <KUMA_KP_DC2_IP>
          keeper_id: 2
        kuma-keep-dc3.local:
          ansible_host: <KUMA_KP_DC3_IP>
          keeper_id: 3
```

(??? [2.5DC] ???????????? ???????????? 1 ???? × 4 ???????: 2 ???? Storage ? ??? 1, 2 ???? ? ??? 2, 0 ? ??? 3. Collector/Correlator ?????? ? ??? 1 ? 2).

6.2. Основной файл параметров KDT (params.yaml)

```
schemaType: ParameterSet
schemaVersion: 1.0.1
namespace: ""
name: bootstrap
project: xdr

regions:
  dc1_active:
    description: "Data Center 1"
  dc2_standby:
    description: "Data Center 2"
  dc3_arbiter:
    description: "Data Center 3"

zones:
  zone1:
    region: dc1_active
    ingress_ip: <INGRESS_IP_DC1>
  zone2:
```

```
    region: dc2_standby
    ingress_ip: <INGRESS_IP_DC2>
  zone3:
    region: dc3_arbiter
    ingress_ip: <INGRESS_IP_DC3>

nodes:
  # --- ??? 1 ---
  - desc: k8s-pri-dc1
    type: primary
    zone: zone1
    host: <K8S_PRI_DC1_IP>
    access:
      ssh:
        user: "<USER_NAME>"
        key: "/home/<USER_NAME>/.ssh/id_rsa"
  - desc: k8s-w-dc1-1
    type: worker
    zone: zone1
    host: <K8S_W_DC1_1_IP>
    access:
      ssh:
        user: "<USER_NAME>"
        key: "/home/<USER_NAME>/.ssh/id_rsa"
  - desc: k8s-w-dc1-2
    type: worker
    zone: zone1
    host: <K8S_W_DC1_2_IP>
    access:
      ssh:
        user: "<USER_NAME>"
        key: "/home/<USER_NAME>/.ssh/id_rsa"
  - desc: k8s-w-dc1-3
    type: worker
    zone: zone1
    host: <K8S_W_DC1_3_IP>
    access:
      ssh:
        user: "<USER_NAME>"
        key: "/home/<USER_NAME>/.ssh/id_rsa"

  # --- ??? 2 ---
  - desc: k8s-pri-dc2
    type: primary
    zone: zone2
    host: <K8S_PRI_DC2_IP>
    access:
      ssh:
        user: "<USER_NAME>"
        key: "/home/<USER_NAME>/.ssh/id_rsa"
  - desc: k8s-w-dc2-1
    type: worker
    zone: zone2
    host: <K8S_W_DC2_1_IP>
    access:
      ssh:
        user: "<USER_NAME>"
        key: "/home/<USER_NAME>/.ssh/id_rsa"
  - desc: k8s-w-dc2-2
    type: worker
    zone: zone2
    host: <K8S_W_DC2_2_IP>
    access:
      ssh:
        user: "<USER_NAME>"
        key: "/home/<USER_NAME>/.ssh/id_rsa"
  - desc: k8s-w-dc2-3
    type: worker
    zone: zone2
```

```
    host: <K8S_W_DC2_3_IP>
    access:
      ssh:
        user: "<USER_NAME>"
        key: "/home/<USER_NAME>/.ssh/id_rsa"

  # --- ??? 3 ---
  # [3DC]: ???????? ???? ??? 3 worker ???? (k8s-w-dc3-2, k8s-w-dc3-3) ?? ????????.
  # [2.5DC]: ???????? ?????? 1 primary ? 1 worker ????, ??? ??????? ????.
  - desc: k8s-pri-dc3
    type: primary
    zone: zone3
    host: <K8S_PRI_DC3_IP>
    access:
      ssh:
        user: "<USER_NAME>"
        key: "/home/<USER_NAME>/.ssh/id_rsa"
  - desc: k8s-w-dc3-1
    type: worker
    zone: zone3
    host: <K8S_W_DC3_1_IP>
    access:
      ssh:
        user: "<USER_NAME>"
        key: "/home/<USER_NAME>/.ssh/id_rsa"

parameters:
  - name: psql_dsn
    source:
      value:
"postgres://<kaspersky_admin>:<StrongPassword123!>@<DB_DC1_IP>:5432,<DB_DC2_IP>:5432,<DB_DC3_IP>:5432"
  - name: ssh_pk
    source:
      path: "/home/<USER_NAME>/.ssh/id_rsa"
  - name: admin_password
    source:
      value: "<STRONG_ADMIN_PASSWORD>"
  - name: inventory
    source:
      value: "/home/<USER_NAME>/kuma_distributed_inventory.yml"
  - name: license
    source:
      value: "/home/<USER_NAME>/license.key"
  - name: smp_domain
    source:
      value: "<smp_domain>"
  - name: pki_host_list
    source:
      value: "admsrv api console kuma monitoring agentserver updater"
  - name: grafana_admin_user
    source:
      value: "xdr_grafana"
  - name: grafana_admin_password
    source:
      value: "<STRONG_GRAFANA_PASSWORD>"

# ?????: ????? ????????? ?????? ??????????? ?????? ???? ???? ?????? ??????? ?????? (??????? Enter)!
```

---

7. Развертывание ядра системы (KDT)

?? **Admin Node**:

```
cd ~
mkdir kdt && tar -xzf bin.tar.gz -C kdt
chmod +x ./kdt/kdt
```

```
./kdt/kdt apply -k /home/<USER_NAME>/xdr-<version>.tar.gz -i /home/<USER_NAME>/params.yaml --accept-eula
```

---

8. Установка внешних сервисов KUMA

8.1. Подготовка хостов KUMA через KDT

???? ????? ?? ???? ???????????? ?? ????? ????????? SMP, ????????? ?? Admin Node:

```
./kdt/kdt invoke kuma --action addHosts --param
host_inventory='/home/<USER_NAME>/kuma_distributed_inventory.yml'
```

8.2. Установка сервисов через CLI

1. **Storage:** ? ??????? OSMP: *?????????? ? ??????? ? ??????? ? ????????? ? ???????*.

   ??????? FQDN, shard, replica, keeper_id. ??????????? **<service_id>**. ?? ????? ?????????:

   ```
   sudo /opt/kaspersky/kuma/kuma storage --core https://kuma.<smp_domain>:7210 --id <service_id> --
   api.port 7230 --install
   ```

2. **Точка назначения Storage:** *?????????? ? ??????? ? ??????? ? ????? ?????????? ? ???????* (???: storage).
3. **Correlator:** ? ???????: *??????????? ? ???????*. ????????? ???????, ? ????????????? ??????? ????? ?????????? Storage. ??????????? ???????. ?? ????? ?????????:

   ```
   sudo /opt/kaspersky/kuma/kuma correlator --core https://kuma.<smp_domain>:7210 --id <service_id> --
   api.port 7220 --install
   ```

4. **Точка назначения Correlator:** ??????? ?????????? (???: correlator).
5. **Collector:** ? *???????*: *?????????? ? ???????*. ??????? ?????????, *???????*. ?

   ????????????? ???????? **обе** ????? ?????????? (Correlator ? Storage). ?? ????? ?????????:

   ```
   sudo /opt/kaspersky/kuma/kuma collector --core https://kuma.<smp_domain>:7210 --id <service_id> --
   api.port 7221 --install
   ```

   (??????????: ???? ????????? ???????? ?? ????? ?????, ??????????? ?????????? ?????, ???????? 7222, 7223).

---

9. Ограничения при Failover

??? ?????????? ?????? ?????? ???? (?????? ?? ????????):

1. **Потеря детектов Sandbox:** ?????, ???????????? ? ??????? ??? ? ?? ???????? ????????????, ????????. ???? ??????: **15 минут** (??? ??????? ???????? - ?? 1 ????).

2. **Потеря задач IRP:** ?????? "Get File and Scan" ? "IoA Scan", ?? ??????????? ? ??????? ????, ???????? ????????????.
3. **Задержки KSB:** ?????????????? ?????? ???????? Kaspersky Sandbox ?? ?????????????? ????????????? ? ??????? **24 часов**.
4. **Перегрузка очередей:** ??? ??????????? ???? ???????? ?? ???? ??? ???????? ?????????? Sandbox ? «????» ????? ???????.
5. **Неконсистентность:** ???????? ????????? ???????????????? ?????? ? ???? **10 минут** ?? ? ????? ?????? ????.

Revision #9
Created 12 October 2026 07:59:39 by ??????
Updated 20 October 2026 07:59:01 by ??????