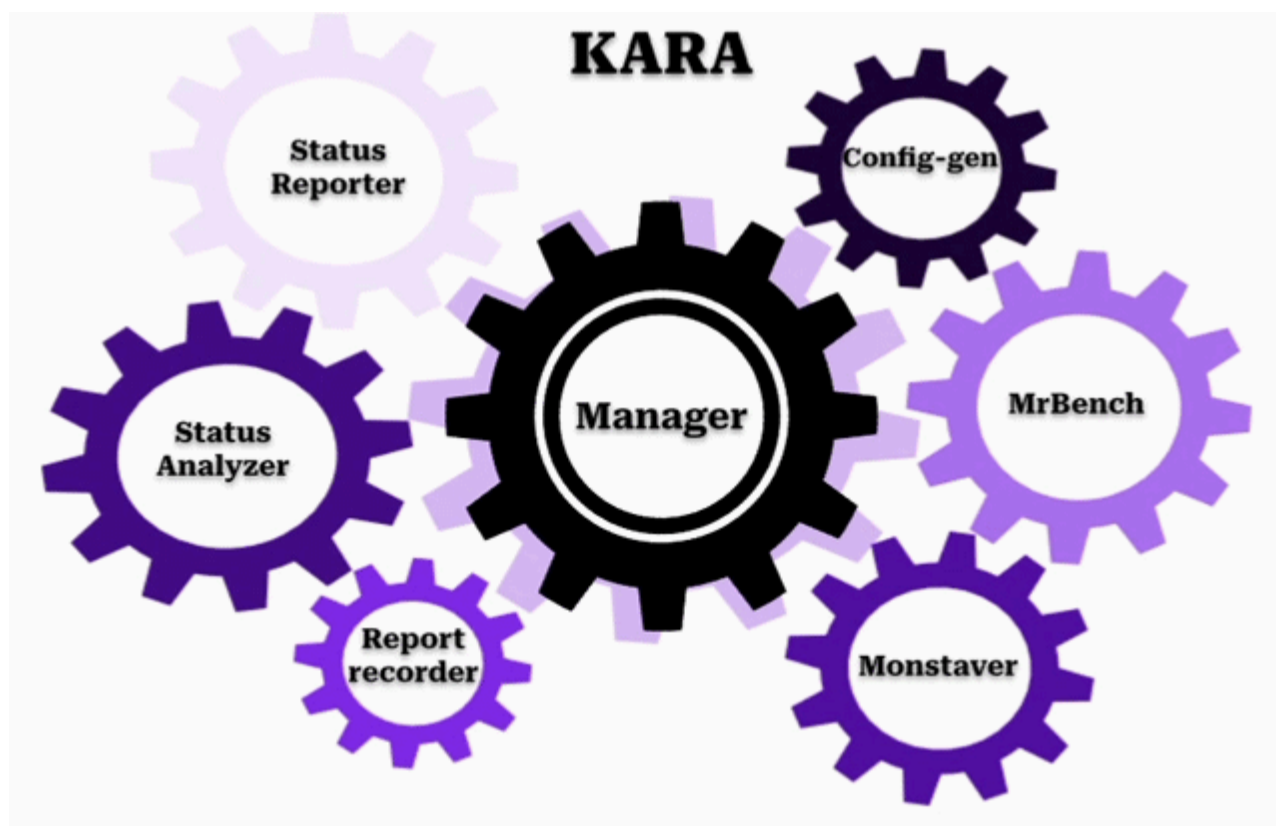


# راهنمای نصب Kara-v1



## ۱- روش های راه اندازی کارا

### ۱-۱- docker

ابتدا فایل فشرده kara-image.tar را از طریق لینک موجود در سند فایل های نصب دانلود کرده و image آن را به کمک دستور زیر load کنید:

```
docker load -q -i kara-image.tar
```

با دستور docker images مطمئن شوید که image بارگذاری شده با موفقیت در مخزن ایمیجها نصب شده است:

docker images	REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
kara-cosbench		1.0.1	30908f41dbaf	3 hours ago	1.28GB

مسیر kara را ایجاد کنید و وارد مسیر آن شوید:

```
mkdir kara
cd kara
```

سپس فایل docker-compose را ایجاد کنید:

```
nano kara-docker-compose.yaml
```

محتوای فایل kara-docker-compose :

```
# ~/kara/kara-docker-compose.yaml
version: "3.8"

services:
  kara:
    image: "${IMAGE_NAME}:${IMAGE_TAG}"
    container_name: ${CONTAINER_NAME}
    hostname: ${HOSTNAME}
    init: true
    env_file:
      - .env
    restart: always
    volumes:
      - /etc/localtime:/etc/localtime:ro
      - $PWD/backup:/tmp/influxdb-backup
      - $PWD/scenario_dir:/home/kara/manager/scenario_dir
      - $PWD/results:/home/kara/results
      - $PWD/db_shard.conf:/home/kara/manager/db_shard/db_shard.conf
      - $PWD/kara-configs:/etc/kara
      - $PWD/jsons:/home/kara/status_reporter/jsons
      - $PWD/metrics:/home/kara/status_reporter/metrics
    ports:
      - "19088:19088"
      - "18088:18088"
    extra_hosts:
      - MC: <server_ip>
    environment:
      - DOCKER_USR=kara
      - DOCKER_GRP=kara
    sysctl:
      - net.ipv4.ip_local_port_range=1024 65535
      - net.ipv4.tcp_tw_reuse=1
    healthcheck:
      test: /home/kara/healthcheck.sh | grep -w "OK" || exit 1
      interval: 10s
      timeout: 5s
      retries: 5

    networks:
      - karanet

networks:
  karanet:
    name: karanet
```

فایل db\_shard.conf را با دستور touch ایجاد کنید:

```
touch db_shard.conf
```

سپس فایل env. را ایجاد کنید:

```
nano .env
```

محتویات فایل env. : توجه داشته باشید که مقادیر CONTAINER\_NAME و HOSTNAME را برابر با مقدار دلخواه قرار دهید.

```
IMAGE_NAME="kara-cosbench"
IMAGE_TAG="1.0.1"
CONTAINER_NAME="kara"
HOSTNAME="kara"
```

قبل از راه اندازی کانتینر کارا برای اولین بار دستور زیر را اجرا کنید:

```
docker-compose -f kara-docker-compose.yaml run kara sudo /home/kara/run-once.sh
```

پس از اجرای این دستور فایل های پیکربندی در مسیر جاری ساخته می شوند:

```
$ ls
backup  configure.conf  jsons  kara-configs  kara-docker-compose.yaml  metrics  results  scenario_dir
```

برای بالا آمدن کانتینر از دستور زیر استفاده کنید:

```
docker-compose -f kara-docker-compose.yaml up -d
```

با استفاده از دستورات زیر می توانید اقدام به حذف، متوقف یا راه اندازی مجدد کانتینر کنید.

```
docker-compose -f kara-docker-compose.yaml down
docker-compose -f kara-docker-compose.yaml stop
docker-compose -f kara-docker-compose.yaml restart
```

برای سلامت سنجی کانتینر از دستور docker ps استفاده کنید:

```
$ docker ps
CONTAINER ID   IMAGE                                COMMAND                                     CREATED        STATUS
PORTS
NAMES
1653fd3ef043   kara-cosbench:1.0.1                "/bin/sh -c 'cd $COS..." 3 hours ago    Up 3 hours (healthy)
0.0.0.0:18088->18088/tcp, :::18088->18088/tcp, 18089/tcp, 0.0.0.0:19088->19088/tcp, :::19088->19088/tcp, 19089/tcp
kara
```

حال اقدامات پس از نصب را در قسمت فرآیند های پس از نصب در ادامه ی این سند دنبال کنید.

## ۲-۱- git

### ۱-۲-۱- نصب نرم افزار بنچمارکینگ Cosbench:

در گام اول نیاز است که ابزار CosBench نصب شود. برای آشنایی و راهنمای نصب می توانید به سند ( [Cloud\\_Object\\_Storage\\_Benchmark\\_\(COSBench\)](#) ) مراجعه کنید.

پس از نصب به دایرکتوری اصلی می رویم و با استفاده از دستورات زیر، اسکریپت cli.sh را به حالت قابل اجرا در می آوریم و یک soft link در مسیر /usr/bin/ ایجاد می کنیم.

```
sudo chmod +x /home/user/cosbench/0.4.2.c4/cli.sh
sudo ln -s /home/user/cosbench/0.4.2.c4/cli.sh /usr/bin/cosbench
```

### ۲-۲-۱- کلون کردن آخرین ورژن از برنامه از opengit:

```
git clone https://opengit.ir/smartlab/kara
```

۳-۲-۱- تنظیمات سیستم عامل: ساخت user مخصوص کارا در mc و تمام سرورهای هیولا و یا در موارد خاص استفاده از user های موجود با دسترسی sudo:

```
adduser kara
```

تغییر فایل sudoers و دادن دسترسی به کاربری که کارا اجرا می‌کند در سرور میزبان کارا و سرور های هیولا برای اجرای دستورات sudo بدون نیاز به password:

```
# visudo

%sudo ALL=(ALL:ALL) ALL
kara ALL=(ALL) NOPASSWD: ALL
```

۴-۲-۱- اجرای ابزار configure پس از انجام مراحل قبلی به دایرکتوری manager رفته و برنامه configure را اجرا کنید تا یکسری از فرآیندهای پیشنهادی اجرا و نصب کارا را انجام دهد.

```
bash configure.sh
```

۱-۴-۲-۱- نصب کتابخانه های پیشنهادی ابزار:

نکته: فقط در صورتی که ابزار configure موفق به نصب آنها نشد اقدام به نصب آنها در سرور اجرا کننده کارا کنید.

```
apt install -y python pip pip install pytz datetime matplotlib pandas alive_progress BeautifulSoup4
wikitextparser mwpaserfromhell sshpass
```

## ۲- فرآیند های پس از نصب:

### ۱-۲- تغییرات در فایل MC-docker-compose

در بعضی از گزارشات کارا از سرور های هیولا نیاز است تصاویری از گرافانا دریافت شوند برای این کار نصب این پلاگین ضروری است. برای نصب پلاگین image renderer گرافانا نیاز است این موارد ذکر شده در پایین به فایل اصلی docker-compose سرور mc افزوده شوند.

نکته: برای نصب ورژن پیشنهادی 3.5.0 نیاز به داشتن گرافانا با ورژن بالاتر از 7.0.0 است. گرافانا در ورژن های جدید خود ممکن است با ورژن های قدیمی influxdb به خوبی سازگار نباشد. برای رفع این مشکل بالاترین ورژن پیشنهادی گرافانا برای حداکثر سازگاری با پلاگین image renderer و influxdb و استفاده در کارا ورژن 9.3.6 آن است.

```
image_renderer:
  image: grafana/grafana-image-renderer:3.5.0
  container_name: grafana-image-renderer
  hostname: renderer
  restart: always
  environment:
    - ENABLE_METRICS=true
    - HTTP_PORT=8081
    - RENDERER_LOG_LEVEL=debug
  ports:
    - 8081:8081
```

```
networks:  
- mcnet
```

## ۲-۲- نصب نرم افزار در سرور هیولا

این پکیج نیز باید در سرور های هیولا نصب شود تا اطلاعات فایل سیستم xfs نیز دریافت و در مستندات خروجی ذخیره شود.

```
apt install xfsprogs
```

## ۲-۳- تغییر shard دیتابیس

پس از نصب و راه اندازی اولیه کارا با یکی از دو روش ذکر شده به دایرکتوری `manager/db_shard/` مراجعه کنید و ابتدا فایل کانفیگ `db_shard.conf` را با توجه به سرور های mc یا دارنده influxdb کانفیگ کرده تا shard به بازه های زمانی ذخیره سازی داده به یک ساعت تغییر کند این کار باعث افزایش راندمان و دقت در زمان بکاپ گرفتن و گزارش گیری کارا می شود و فقط یک بار نیاز به انجام این کار است. با اجرا شدن این بخش توسط ابزار `db_shard_script.py` در تمام دیتابیس های ذکر شده shard به یک ساعت تغییر پیدا می کند.

نکته: پس از تغییرات shard در دیتابیس برای ذخیره سازی درست اطلاعات حداقل ۲ ساعت باید صبر کنید و پس از آن دیگر ابزار های کارا که مرتبط با دیتابیس هستند را اجرا کنید.

```
database_info: # this part of the config is for change shard hour in databases  
  
mc:  
  ip: 0.0.0.0  
  ssh_port: 22  
  ssh_user: user  
  container_name: influxdb  
  databases: # list of databases  
    - opentsdb
```

اجرای ابزار:

```
python3 db_shard_script.py
```

برگرفته از «[https://kateb.burna.ir/w/index.php?title=راهنمای\\_نصب\\_Kara-v1&oldid=84421](https://kateb.burna.ir/w/index.php?title=راهنمای_نصب_Kara-v1&oldid=84421)»

مشارکت کنندگان: محمد تقوا, رضوان رضایی

■