

## Phase 1

Dockerfile:

```
FROM alpine:3.17
RUN apk update && apk add curl
ENTRYPOINT [ "curl" ]
```

برنامه curl را بر ایمج alpine نصب می کنیم و به عنوان entrypoint کاننتینر قرار می دهیم.

```
sina@nej:~/Documents/Courses/Cloud/Hw2/phase1$ docker build . -t ccurl:v1
Sending build context to Docker daemon 2.048kB
Step 1/3 : FROM alpine:3.17
---> 49176f190c7e
Step 2/3 : RUN apk update && apk add curl
---> Using cache
---> c2281a6076b5
Step 3/3 : ENTRYPOINT [ "curl" ]
---> Using cache
---> 6ea0b6096bc2
Successfully built 6ea0b6096bc2
Successfully tagged ccurl:v1
```

ایمچ های موجود روی سیستم

```
Successfully tagged ccurl:v1
sina@nej:~/Documents/Courses/Cloud/Hw2/phase1$ docker image ls | grep ccurl
ccurl          latest                6ea0b6096bc2   23 hours ago    11.9MB
ccurl          v1                   6ea0b6096bc2   23 hours ago    11.9MB
sina@nej:~/Documents/Courses/Cloud/Hw2/phase1$
```

نتیجه تلاش برای پوش کردن رو گیت هاب:

```
sina@nej:~/Documents/Courses/Cloud/Hw2/phase1$ proxychains docker push sinamna/ccurl:v1
ProxyChains-3.1 (http://proxychains.sf.net)
The push refers to repository [docker.io/sinamna/ccurl]
3c43619f259f: Retrying in 6 seconds
ded7a220bb05: Retrying in 1 second
^C
sina@nej:~/Documents/Courses/Cloud/Hw2/phase1$ proxychains docker push sinamna/ccurl:v1
ProxyChains-3.1 (http://proxychains.sf.net)
The push refers to repository [docker.io/sinamna/ccurl]
Get "https://registry-1.docker.io/v2/": net/http: TLS handshake timeout
sina@nej:~/Documents/Courses/Cloud/Hw2/phase1$ proxychains docker push sinamna/ccurl:v1
ProxyChains-3.1 (http://proxychains.sf.net)
The push refers to repository [docker.io/sinamna/ccurl]
Get "https://registry-1.docker.io/v2/": net/http: TLS handshake timeout
sina@nej:~/Documents/Courses/Cloud/Hw2/phase1$ proxychains docker push sinamna/ccurl:v1
ProxyChains-3.1 (http://proxychains.sf.net)
The push refers to repository [docker.io/sinamna/ccurl]
```

ران کردن کاننتینر:

```
sina@nej:~/Documents/Courses/Cloud/Hw2/phase1$ docker run ccurl:v1 google.com
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total     Spent    Left  Speed
100   219   100   219    0     0   184      0  0:00:01  0:00:01 --:--:--  184
<HTML><HEAD><meta http-equiv="content-type" content="text/html; charset=utf-8">
<TITLE>301 Moved</TITLE></HEAD><BODY>
<H1>301 Moved</H1>
The document has moved
<A HREF="http://www.google.com/">here</A>
</BODY></HTML>
sina@nej:~/Documents/Courses/Cloud/Hw2/phase1$
```

Phase 2:

داگر فایل پروژه

```
## Build
FROM m.docker-registry.ir/golang:1.18.5-alpine3.15

ENV GO111MODULE=auto

WORKDIR /app

COPY go.mod ./
COPY go.sum ./
RUN go mod download

COPY . ./

RUN CGO_ENABLED=0 GOOS=linux GOARCH=amd64 go build -o /retriever

EXPOSE 8080

ENTRYPOINT ["/retriever"]
```

Docker-compose of project:

```
version: '3'
services:
  redis:
    image: redis:7-alpine
    container_name: redis
    environment:
      - ALLOW_EMPTY_PASSWORD=yes
      - REDIS_DISABLE_COMMANDS=FLUSHDB,FLUSHALL
    ports:
      - '6379:6379'
```

```

volumes:
  - './data/redis:/bitnami/redis/data'
retriever:
  build: .
  container_name: retriever
  depends_on:
    - redis
  ports:
    - "8080:8080"

```

در ساخت سرویس ردیس از ایمج موجود در داکرهاب و volume مشخص شده برای persist کردن آن استفاده کردیم.  
بیلد کردن داکر کامپوز:

```

sina@nej:~/Documents/Courses/Cloud/inf2/phase2$ docker-compose build
redis uses an image, skipping
Building retriever
Sending build context to Docker daemon  88.06kB
Step 1/10 : FROM m.docker-registry.ir/golang:1.18.5-alpine3.15
--> 397a291acfd6
Step 2/10 : ENV GO111MODULE=auto
--> Using cache
--> dc7080df8d92
Step 3/10 : WORKDIR /app
--> Using cache
--> a8598f45d652
Step 4/10 : COPY go.mod ./
--> Using cache
--> c92953269e5b
Step 5/10 : COPY go.sum ./
--> Using cache
--> ca9d5e9ffea3
Step 6/10 : RUN go mod download
--> Using cache
--> 4de2e56cbffc
Step 7/10 : COPY . ./
--> 1db8c6b602d8
Step 8/10 : RUN CGO_ENABLED=0 GOOS=linux GOARCH=amd64 go build -o /retriever
--> Running in 48e88f7af143
Removing intermediate container 48e88f7af143
--> 06199095cf34
Step 9/10 : EXPOSE 8080
--> Running in 666144ed8c43
Removing intermediate container 666144ed8c43
--> b7aaf5718294

```

ران کردن برنامه و وصل شدن آن به ردیس:

```
sina@nej:~/Documents/Courses/Cloud/HW2/phase2$ docker-compose up
Starting redis ... done
Recreating retriever ... done
Attaching to redis, retriever
redis      | 1:C 20 Dec 2022 19:00:51.162 # o000o000o000o Redis is starting o000o000o000o
redis      | 1:C 20 Dec 2022 19:00:51.162 # Redis version=7.0.7, bits=64, commit=00000000, modified=0, pid=1, just started
redis      | 1:C 20 Dec 2022 19:00:51.162 # Warning: no config file specified, using the default config. In order to specify a config file use redis-server /path/to/redis.conf
redis      | 1:M 20 Dec 2022 19:00:51.162 * monotonic clock: POSIX clock_gettime
redis      | 1:M 20 Dec 2022 19:00:51.163 * Running mode=standalone, port=6379.
redis      | 1:M 20 Dec 2022 19:00:51.163 # Server initialized
redis      | 1:M 20 Dec 2022 19:00:51.163 # WARNING Memory overcommit must be enabled! Without it, a background save or replication may fail under low memory condition. Being di
sabled, it can can also cause failures without low memory condition, see https://github.com/jemalloc/jemalloc/issues/1328. To fix this issue add 'vm.overcommit_memory = 1' to /etc
/sysctl.conf and then reboot or run the command 'sysctl vm.overcommit_memory=1' for this to take effect.
retriever  | loading config...
redis      | 1:M 20 Dec 2022 19:00:51.163 * Loading RDB produced by version 7.0.7
redis      | 1:M 20 Dec 2022 19:00:51.163 * RDB age 287 seconds
redis      | 1:M 20 Dec 2022 19:00:51.163 * RDB memory usage when created 0.89 Mb
redis      | 1:M 20 Dec 2022 19:00:51.163 * Done loading RDB, keys loaded: 0, keys expired: 0.
retriever  | &{Redis:{ExpireTime:100 Host:redis Port:6379 Db:0 Password:} Server:{Port:8080 ApiKey:6543CBA2-5094-4D4D-AA90-F3142F48C821}}
retriever  | connecting to redis...
redis      | 1:M 20 Dec 2022 19:00:51.163 * DB loaded from disk: 0.000 seconds
retriever  | PONG
retriever  | redis connected!!!
retriever  | [GIN-debug] [WARNING] Creating an Engine instance with the Logger and Recovery middleware already attached.
retriever  |
retriever  | [GIN-debug] [WARNING] Running in "debug" mode. Switch to "release" mode in production.
```

Docker ps:

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
81c413c6fb11	phase2_retriever	"/retriever"	37 seconds ago	Up 37 seconds	0.0.0.0:8080->8080/tcp, :::8080->8080/tcp	retrie
840ba6c28a00	redis:7-alpine	"docker-entrypoint.s..."	46 minutes ago	Up 37 seconds	0.0.0.0:6379->6379/tcp, :::6379->6379/tcp	redis

Docker stats:

CONTAINER ID	NAME	CPU %	MEM USAGE / LIMIT	MEM %	NET I/O	BLOCK I/O	PIDS
81c413c6fb11	retriever	0.03%	4.457MiB / 15.31GiB	0.03%	5.57kB / 370B	4.1kB / 0B	5
840ba6c28a00	redis	0.12%	4.102MiB / 15.31GiB	0.03%	5.85kB / 297B	2.02MB / 0B	5

Docker inspect:

```
sina@nej:~$ docker inspect retriever
[
  {
    "Id": "81c413c6fb11e0022f75e6f84d2015162263d3ec01cf0783a16f38068164b120",
    "Created": "2022-12-20T19:00:51.171134979Z",
    "Path": "/retriever",
    "Args": [],
    "State": {
      "Status": "running",
      "Running": true,
      "Paused": false,
      "Restarting": false,
      "OOMKilled": false,
      "Dead": false,
      "Pid": 73060,
      "ExitCode": 0,
      "Error": "",
      "StartedAt": "2022-12-20T19:00:51.502964025Z",
      "FinishedAt": "0001-01-01T00:00:00Z"
    },
    "Image": "sha256:4beeb42d82a7f98e0101d30c5b36f802105c7db61ad1adb58420b001d747bdd4",
    "ResolvConfPath": "/var/lib/docker/containers/81c413c6fb11e0022f75e6f84d2015162263d3ec01cf0783a16f38068164b120/resolv.conf",
    "HostnamePath": "/var/lib/docker/containers/81c413c6fb11e0022f75e6f84d2015162263d3ec01cf0783a16f38068164b120/hostname",
    "HostsPath": "/var/lib/docker/containers/81c413c6fb11e0022f75e6f84d2015162263d3ec01cf0783a16f38068164b120/hosts",
    "LogPath": "/var/lib/docker/containers/81c413c6fb11e0022f75e6f84d2015162263d3ec01cf0783a16f38068164b120/81c413c6fb11e0022f75e6f84d2015162263d3ec01cf0783a16f38068164b120-json.log",
    "NetworkSettings": {
      "Bridge": "br-81c413c6fb11",
      "SandboxID": "81c413c6fb11e0022f75e6f84d2015162263d3ec01cf0783a16f38068164b120",
      "HairpinMode": false,
      "LinkLocalIPv6Address": "",
      "LinkLocalIPv6Prefix": "",
      "MacAddress": "02:42:9d:1d:1a:1b",
      "NetworkInterface": "eth0",
      "PortMapping": {
        "HostPort": 8080,
        "ContainerPort": 8080,
        "Protocol": "tcp"
      },
      "Ports": {
        "8080/tcp": [
          {
            "HostPort": 8080,
            "ContainerPort": 8080,
            "Protocol": "tcp"
          }
        ]
      }
    }
  }
]
```

همانطور که مشاهده می کنیم داکر کامپوز برای برقرار ارتباط بین سرویس های موجود در فایل یک شبکه از نوع bridge درست می کند:

```

sina@nej:~$ docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
5425e5f6b9a3        bridge              bridge              local
bb803c511173        host                host                local
259dab343fdf        none                null                local
5fab2dd0d687        phase2_default      bridge              local
sina@nej:~$

```

همچنین فایل کانفیگ سرور به صورت زیر می باشد

```

redis:
  expire_time: 100
  host: redis
  port: 6379
  db: 0
  password: ""

server:
  port: 8080
  api_key: 6543CBA2-5094-4D4D-AA90-F3142F48C821

```

Phase 3:

To be continued