Test task for Devops-support position (Shifted work)

Hello and welcome to the Soramitsu DevOps take home challenge! We value your time and don't expect you to spend anymore than 2-3 hours on this task. The challenge is a chance to demonstrate basic competency. This isn't just simply a pass/fail on meeting the requirements though - there will be some expectation of following **best practices** with whatever method you use and making it easy for others to iterate on your work. If you pass this stage further technical interview questions may expand on this scenario.

The Scenario

The old team of developers just released an application and they are very excited about having everyone running their application. There is a problem: the developers are old school guys and don't know that currently everyone runs applications in containers.

Your job is to create a Docker file for application Exim. Exim it is popular mail server

The Task

1. Build application https://github.com/Exim/exim from source code. The master branch don't have Makefile, you need to download archive with source code (e.g: exim-4.95.tar.gz) from https://www.exim.org/mirrors.html, it will have README file which will help you to build this application.
   After you build successfully you should have folder `build-<platform>` (e.g. `build-Linux-x86_64)` with builded files, you need to execute `./exim` binary, and get the message:

```
Exim is a Mail Transfer Agent. It is normally called by Mail User Agents,
not directly from a shell command line. Options and/or arguments control
what it does when called. For a list of options, see the Exim documentation.
```

2. Create multi staging Dockerfile, which will create docker image with only exim binary and

configuration files inside (for the purpose of this task the one binary is enough but actual application contains of several binaries/tools)

When you run docker container you should see following message:

```
Exim is a Mail Transfer Agent. It is normally called by Mail User Agents,
not directly from a shell command line. Options and/or arguments control
what it does when called. For a list of options, see the Exim documentation.
```

Docker file should contain:

```
FROM <IMAGE>

RUN <INSTALL DEPENDENCY>

# Download, verify the sha256sum  and unarchive file
RUN < Download http://exim.mirror.globo.tech/exim/exim4/exim-4.95.tar.gz >
    <verify what sha256sum of archive  is
6fdd83edd7bf53bdc0a208187643aa0c41861aba9a09a026f78351ac8e768b9c>
    <unarchive downloaded file>

RUN <step which you need before build>

# build
RUN make


FROM <IMAGE>

<Copy binary and configuration file from first stage>
<you may need to have extra steps>

CMD ["exim"]
```