

Informa Parcial 3

Consigne los comandos de linux necesarios para el aprovisionamiento de los servicios solicitados. En este punto no debe incluir archivos tipo Dockerfile solo se requiere que usted identifique los comandos o acciones que debe automatizar (15%)

```
docker run -d --name=consul -p 8500:8500 consul

docker run -d --name=registrator --net=host --
volume=/var/run/docker.sock:/tmp/docker.sock
gliderlabs/registrator:latest -internal consul://localhost:8500

curl localhost:8500/v1/catalog/services
```

Escriba los archivos Dockerfile para cada uno de los servicios solicitados junto con los archivos fuente necesarios. Tenga en cuenta consultar buenas prácticas para la elaboración de archivos Dockerfile. (20%)

```
FROM alpine:3.5

RUN apk add --no-cache python3 && \
    python3 -m ensurepip && \
    rm -r /usr/lib/python*/ensurepip && \
    pip3 install --upgrade pip setuptools && \
    pip install flask==0.10.1 && \
    pip uninstall -y pip && \
    rm -rf /root/.cache/

COPY . /app

WORKDIR /app

EXPOSE 5000

CMD ["python3", "./app.py"]
```

Escriba el archivo docker-compose.yml necesario para el despliegue de la infraestructura (10%). No emplee configuraciones deprecated. Incluya un diagrama general de los componentes empleados.

```
version: '2'

services:
  registrator:
    image: gliderlabs/registrator:latest
    container_name: registrator
```

```

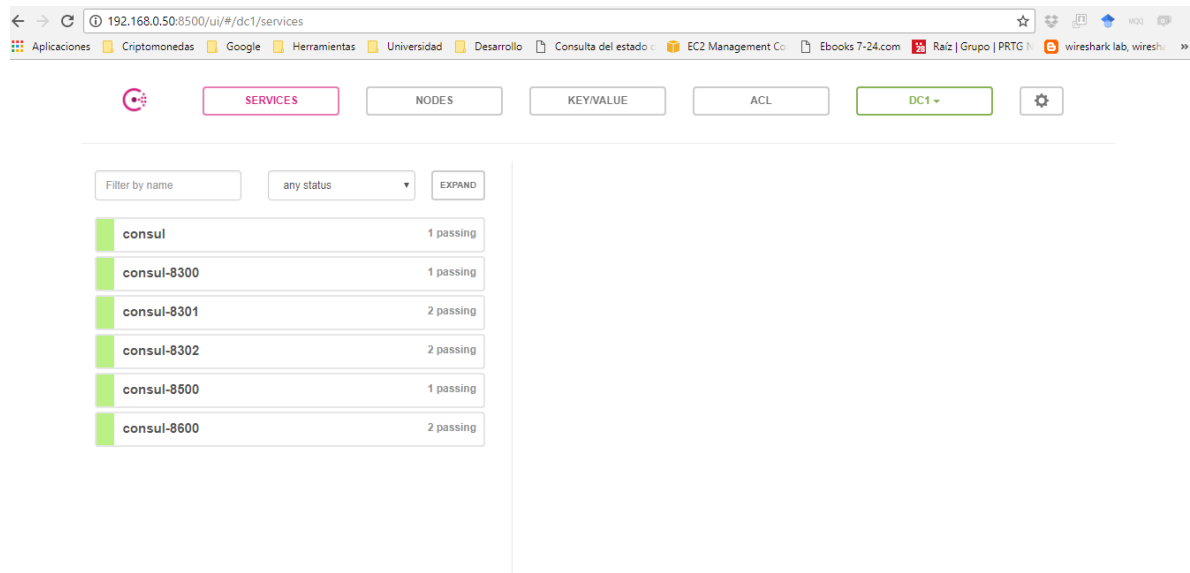
links:
  - consul:consul
volumes:
  - /var/run/docker.sock:/tmp/docker.sock
command: -internal consul://consul:8500
consul:
  image: consul
  container_name: consul
  ports:
    - 8500:8500
balanceador:
  build: balanceador/.
  container_name: balanceador
  ports:
    - "80:80"
  links:
    - consul:consul
  command: consul-template -consul-addr=consul:8500 -
template="/etc/haproxy/haproxy.ctmpl:/etc/haproxy/haproxy.cfg:/usr/sbin/h
aproxy -f /etc/haproxy/haproxy.cfg -D -p /var/run/haproxy.pid"
web_service:
  build: web_service/.
  ports:
    - "5000"
redis:
  image: "redis:alpine"
  container_name: redis
  ports:
    - "6379"

```

El informe debe publicarse en un repositorio de github el cual debe ser un fork de <https://github.com/ICESI-Training/sd-exam3> y para la entrega deberá hacer un Pull Request (PR) respetando la estructura definida. El código fuente y la url de github deben incluirse en el informe (15%). Tenga en cuenta publicar los archivos para el aprovisionamiento

<https://github.com/rocco522/sd-exam3>

Incluya evidencias que muestran el funcionamiento de lo solicitado (15%)



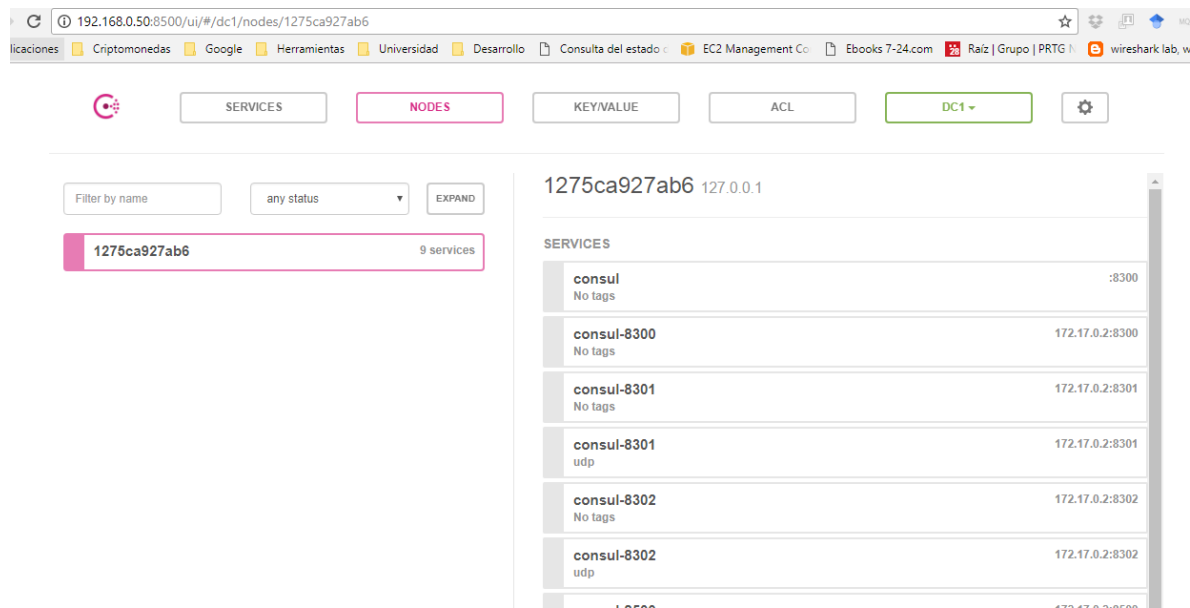
192.168.0.50:8500/ui/#/dc1/services

Aplicaciones Criptomonedas Google Herramientas Universidad Desarrollo Consulta del estado EC2 Management Co Ebooks 7-24.com Raiz | Grupo | PRTG wireshark lab, wireshark

SERVICES NODES KEY/VALUE ACL DC1

Filter by name any status EXPAND

consul	1 passing
consul-8300	1 passing
consul-8301	2 passing
consul-8302	2 passing
consul-8500	1 passing
consul-8600	2 passing



192.168.0.50:8500/ui/#/dc1/nodes/1275ca927ab6

Aplicaciones Criptomonedas Google Herramientas Universidad Desarrollo Consulta del estado EC2 Management Co Ebooks 7-24.com Raiz | Grupo | PRTG wireshark lab, wireshark

SERVICES NODES KEY/VALUE ACL DC1

Filter by name any status EXPAND

1275ca927ab6 9 services

1275ca927ab6 127.0.0.1

SERVICES

consul	No tags	:8300
consul-8300	No tags	172.17.0.2:8300
consul-8301	No tags	172.17.0.2:8301
consul-8301	udp	172.17.0.2:8301
consul-8302	No tags	172.17.0.2:8302
consul-8302	udp	172.17.0.2:8302

```
root@ubuntu:/home/ricardo/sd-exam3/A00268540# curl localhost:8500/v1/catalog/services
{
  "consul": [],
  "consul-8300": [],
  "consul-8301": [
    "udp"
  ],
  "consul-8302": [
    "udp"
  ],
  "consul-8500": [],
  "consul-8600": [
    "udp"
  ]
}
```

Documente algunos de los problemas encontrados y las acciones efectuadas para su solución al aprovisionar la infraestructura y aplicaciones (10%)

Luego de haber realizado el examen, tuve un problema muy habitual: no corría correctamente. En este caso me dediqué a buscar en internet información sobre los errores hasta que los fui corrigiendo uno por uno.

Trabajando con la máquina Ubuntu server, no tuve problemas en general con la ejecución, pero una vez estuvo finalizado el proyecto, intenté ejecutarlo en Windows con Docker, pero genera otra cantidad de errores.