

## [MDAS] - Principios y herramientas de desarrollo

### Entregable 3 – Kubernetes (Parte II)

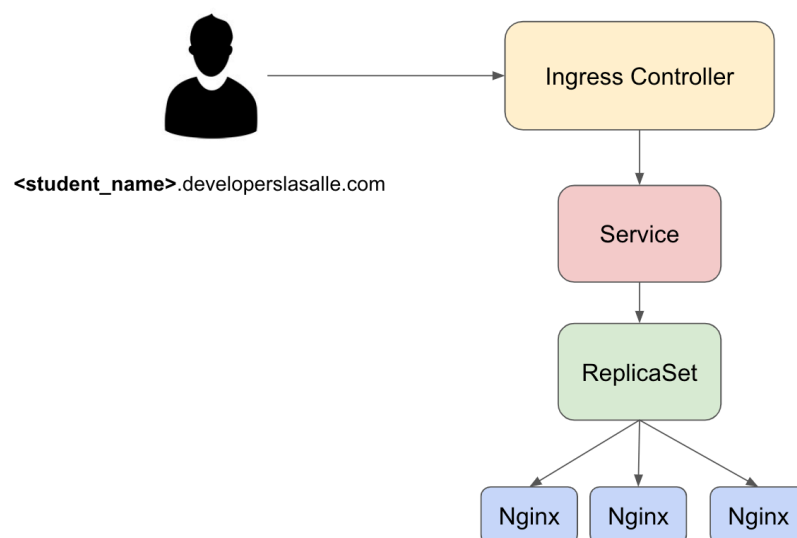
1. [Ingress Controller / Secrets] Crea los siguientes objetos de forma declarativa con las siguientes especificaciones:

- Imagen: nginx
- Version: 1.19.4
- 3 replicas
- Label: app: nginx-server
- Exponer el puerto 80 de los pods
- Limits:
  - CPU: 20 milicores
  - Memoria: 128Mi
- Requests:
  - CPU: 20 milicores
  - Memoria: 128Mi

a. A continuación, tras haber expuesto el servicio en el puerto 80, se deberá acceder a la página principal de Nginx a través de la siguiente URL:

[http://<student\\_name>.student.lasalle.com](http://<student_name>.student.lasalle.com)

Diagrama:



Validación:

- Navegador web



## Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to [nginx.org](http://nginx.org).  
Commercial support is available at [nginx.com](http://nginx.com).

*Thank you for using nginx.*

- Otra forma, mediante la terminal a partir de la ejecución de la siguiente instrucción:

```
curl -k http://<student name>.student.lasalle.com
```

```
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
  body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif;
  }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
```

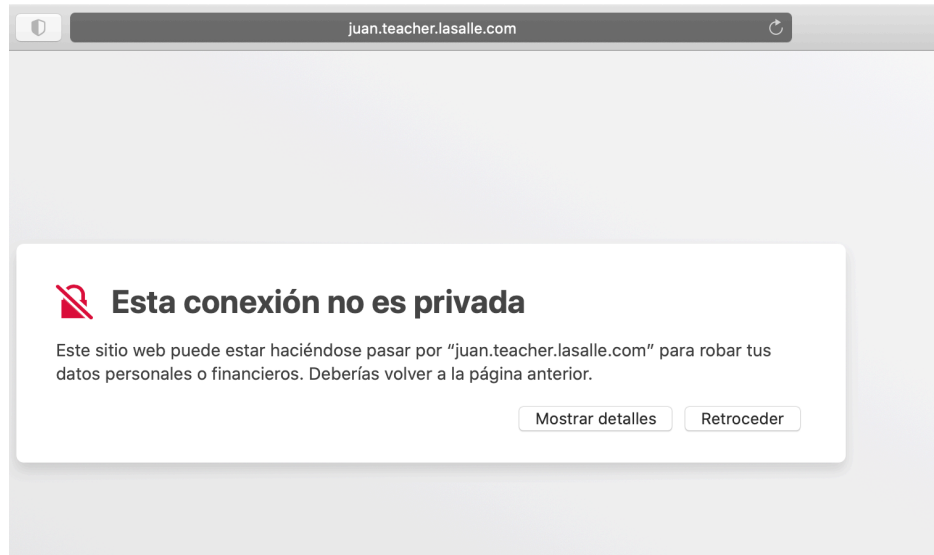
- b. Una vez realizadas las pruebas con el protocolo HTTP, se pide acceder al servicio mediante la utilización del protocolo HTTPS, para ello:

- Crear un certificado mediante la herramienta OpenSSL u otra similar
- Crear un secret que contenga el certificado

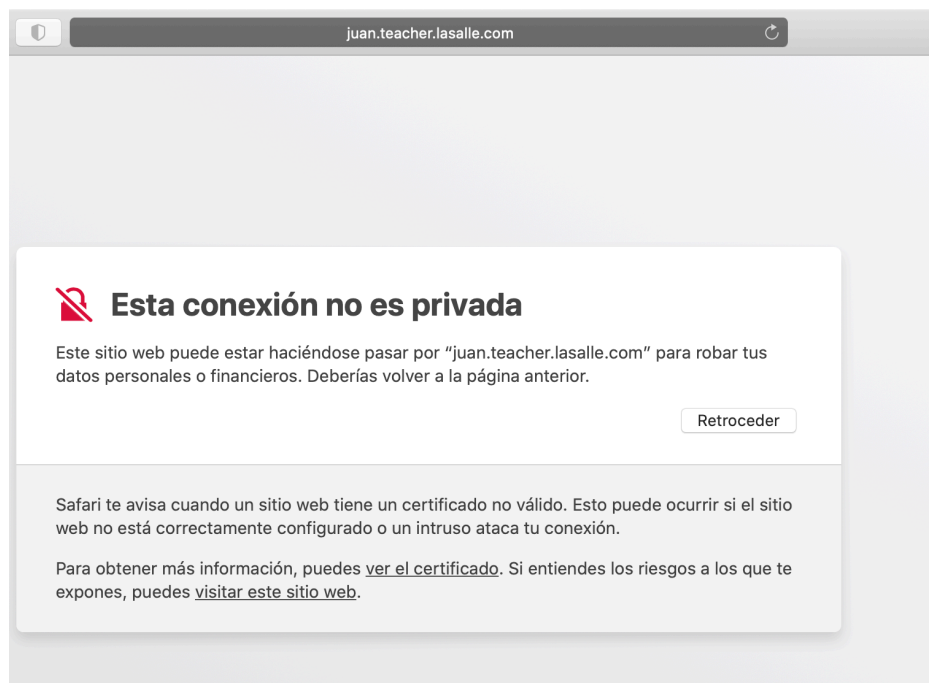
Validación:

- Navegador web

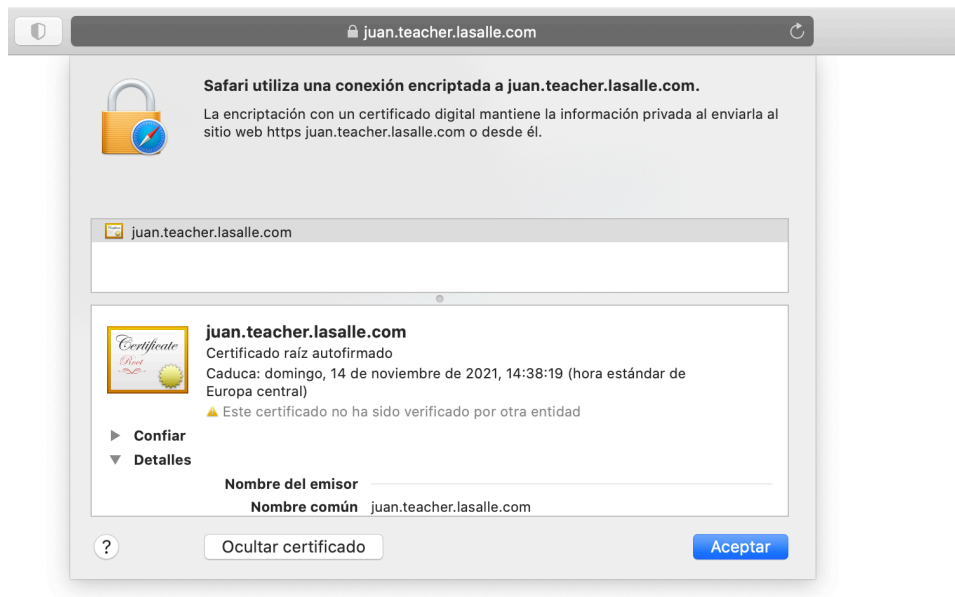
Como el certificado no está expedido por una entidad de confianza (Comodo, DigiCert, Let's Encrypt, ... ), os aparecerá el siguiente mensaje en el navegador:



Deberéis hacer click en *Mostrar detalles* > *Visitar este sitio web*:



Si lo habéis configurado correctamente, accederéis a la página principal de Nginx. Si os fijáis en la barra superior aparecerá un candado, eso significa que estáis accediendo mediante HTTPS. Si hacéis click en el certificado y lo mostráis, deberéis ver el certificado creado:



- Otra forma, mediante terminal a partir del siguiente comando curl:

curl --cacert <certificate> https://<student\_name>.student.lasalle.com

```
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
  body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif;
  }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
```

#### Aclaraciones:

- En la URL, se deberá sustituir <student\_name> por el nombre del estudiante
- No es necesario cambiar los ficheros de configuración de Nginx
- No es necesario exponer el servicio hacia el exterior (ClusterIP)

2. [StatefulSet] Crear un StatefulSet con 3 instancias de MongoDB (ejemplo visto en clase)

Se pide:

- Habilitar el clúster de MongoDB
  - Realizar una operación en una de las instancias a nivel de configuración y verificar que el cambio se ha aplicado al resto de instancias
  - Diferencias que existiría si el montaje se hubiera realizado con el objeto de ReplicaSet
- 
3. [Horizontal Pod Autoscaler] Crea un objeto de kubernetes HPA, que escale a partir de las métricas CPU o memoria (a vuestra elección). Establece el umbral al 50% de CPU/memoria utilizada, cuando pase el umbral, automáticamente se deberá escalar al doble de replicas.

Podéis realizar una prueba de estrés realizando un número de peticiones masivas mediante la siguiente instrucción:

```
kubectrl run -i --tty load-generator --rm --image=busybox --restart=Never --  
/bin/sh -c "while sleep 0.01; do wget -q -O- http://<svc_name>; done"
```

**IMPORTANTE:** Adjunta todos los ficheros que has utilizado y añade una breve descripción con el soporte de capturas que evidencien que has conseguido el objetivo propuesto en la descripción del ejercicio.

**ENTREGA:** Crear un proyecto en vuestro repositorio de código (Github, Gilab, ...) llamado **kube-exercises**. Dentro de ese proyecto, crear una carpeta para la segunda entrega, por ejemplo **hw-03**.

Dentro de esa carpeta, crear un fichero por pregunta **answer\_exercise\_1.md**. Si necesitáis subir ficheros, crear una carpeta para ese ejercicio con todo el contenido. Si necesitáis hacer algún tipo de aclaración, hacerlo en el fichero **README.md**.

En el enlace de entrega del campus, entregar una nota (notes\_hw\_3.txt) con el enlace del repositorio y el código del commit.