

HTTPS & SELF SIGNED SSL CERTIFICATE

HTTP

Hypertext Transfer Protocol

This is the protocol used by a web browser to make requests to a web server for a webpage. If there is an error with the request, it will return a code (e.g. 404).



HTTPS

Hypertext Transfer Protocol Secure

SSL (Secure Sockets Layer) certificate is a digital file that authenticates a website and enables an encrypted connection between a web server and a web browser.



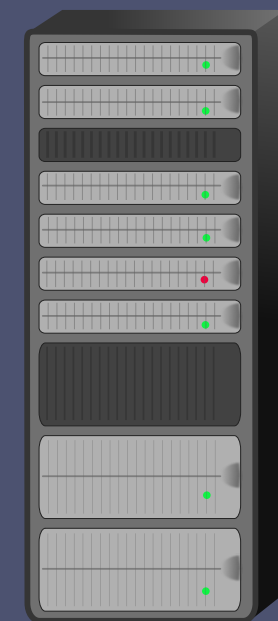
PREREQUISITES

You need to trust that public key cryptography & signature works:

1. Any message **encrypted** with **Oak's public key** can only be **decrypted** with **Oak's private key**.
2. Anyone with access to **Bow's public key** can verify that a message (signature) could only have been created by someone with access to **Bow's private key**.



Your Browser

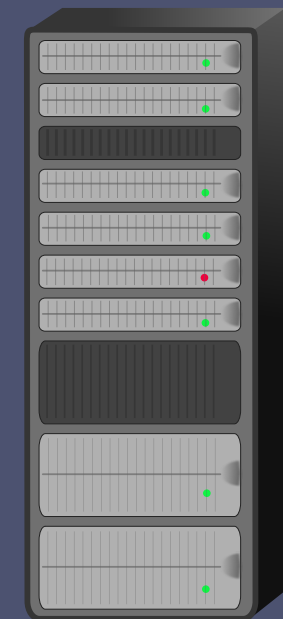


<https://google.com>



Your Browser

I want google.com



<https://google.com>



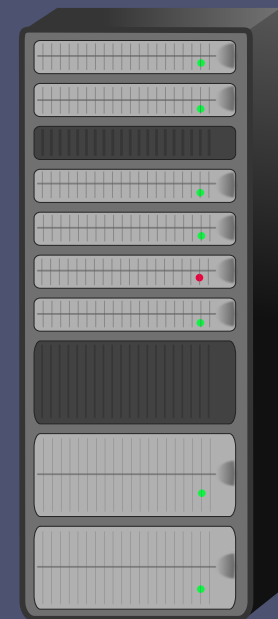
Your Browser



I want google.com



OK ! This is my certificate
containing my public key.
It was signed by Google CA



<https://google.com>



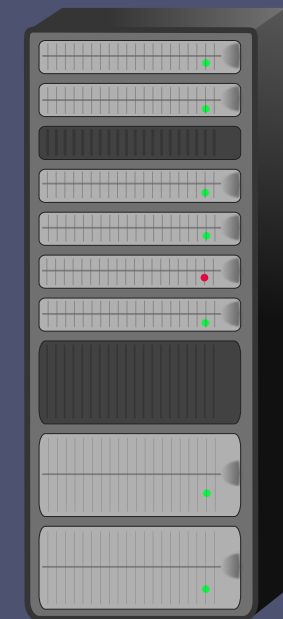
Your Browser



I trust Google CA
I will verify this public key

I want google.com

OK ! This is my certificate
containing my public key.
It was signed by Google CA



<https://google.com>



Your Browser



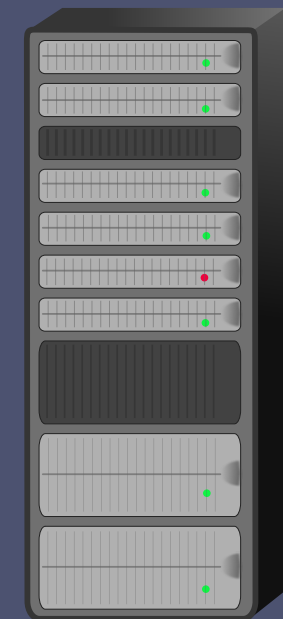
I trust Google CA
I will verify this public key

I want google.com

OK ! This is my certificate
containing my public key.
It was signed by Google CA

I trust Google CA

I have created a new secret key
and encrypted with your public key



<https://google.com>



Your Browser



I trust Google CA
I will verify this public key



I want google.com

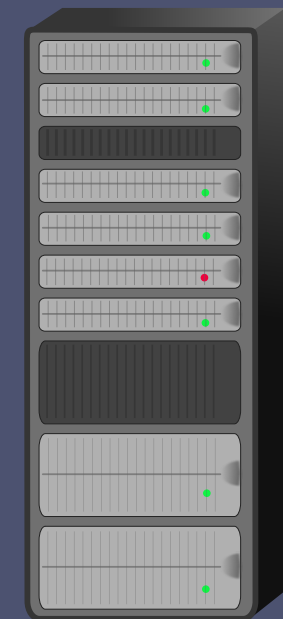


OK ! This is my certificate
containing my public key.
It was signed by Google CA



I trust Google CA

I have created a new secret key
and encrypted with your public key



<https://google.com>



I have my private key
and I can decrypt this
so now I have your new
secret key!



Your Browser



I trust Google CA
I will verify this public key



I want google.com



OK ! This is my certificate
containing my public key.
It was signed by Google CA



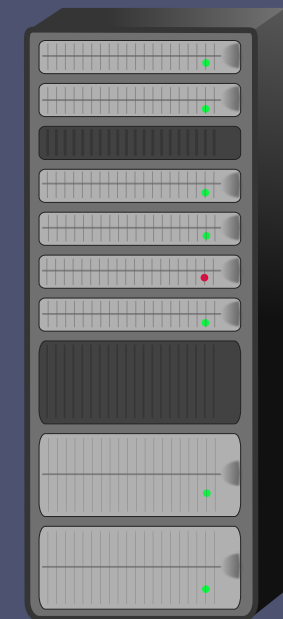
I trust Google CA

I have created a new secret key
and encrypted with your public key



We are only two machine
who know this new secret key

Now let's encrypt all of our communication
with this key



<https://google.com>

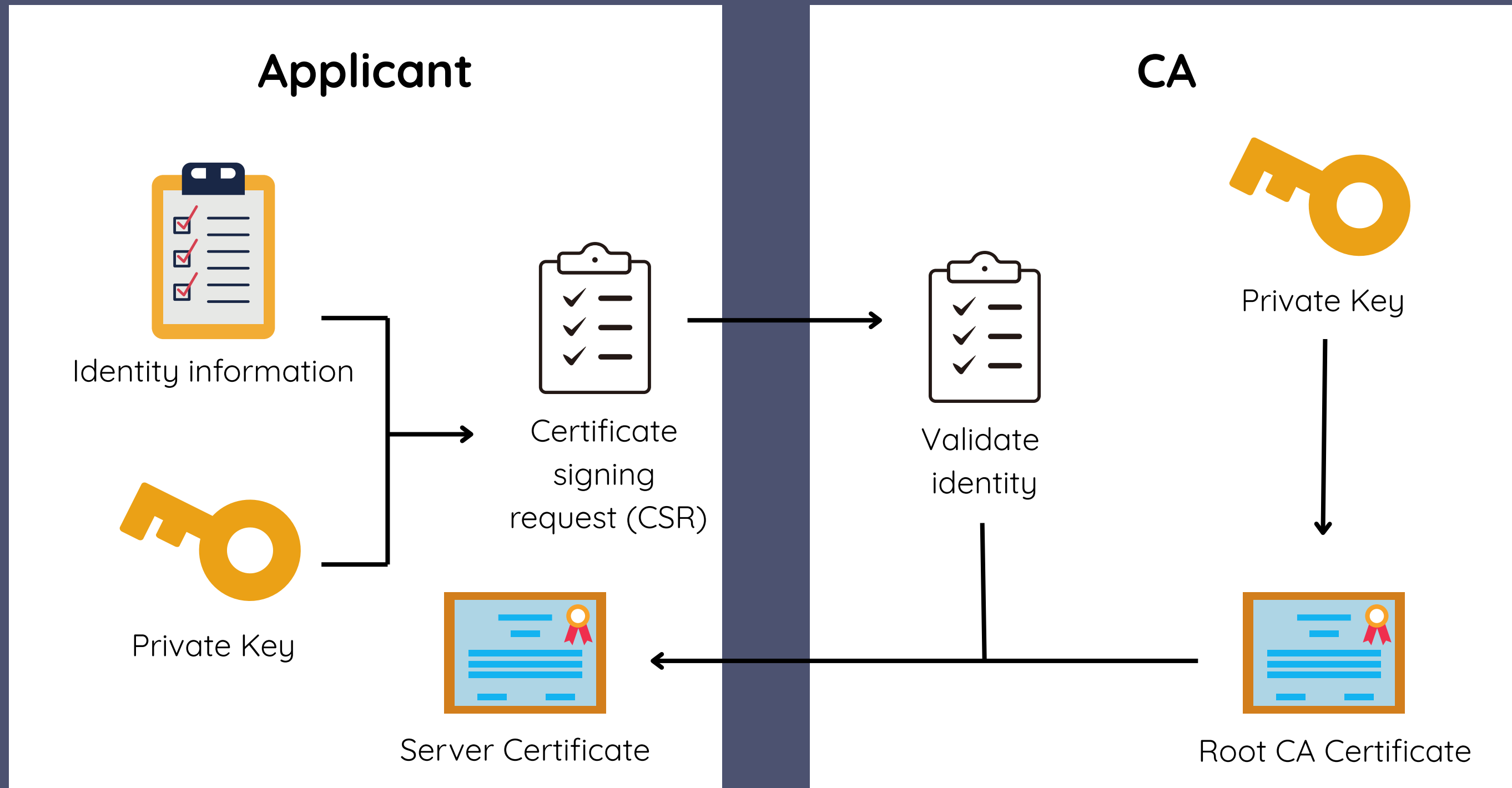


I have my private key
and I can decrypt this
so now I have your new
secret key!

WHAT'S CERTIFICATE AUTHORITY(CA) ?

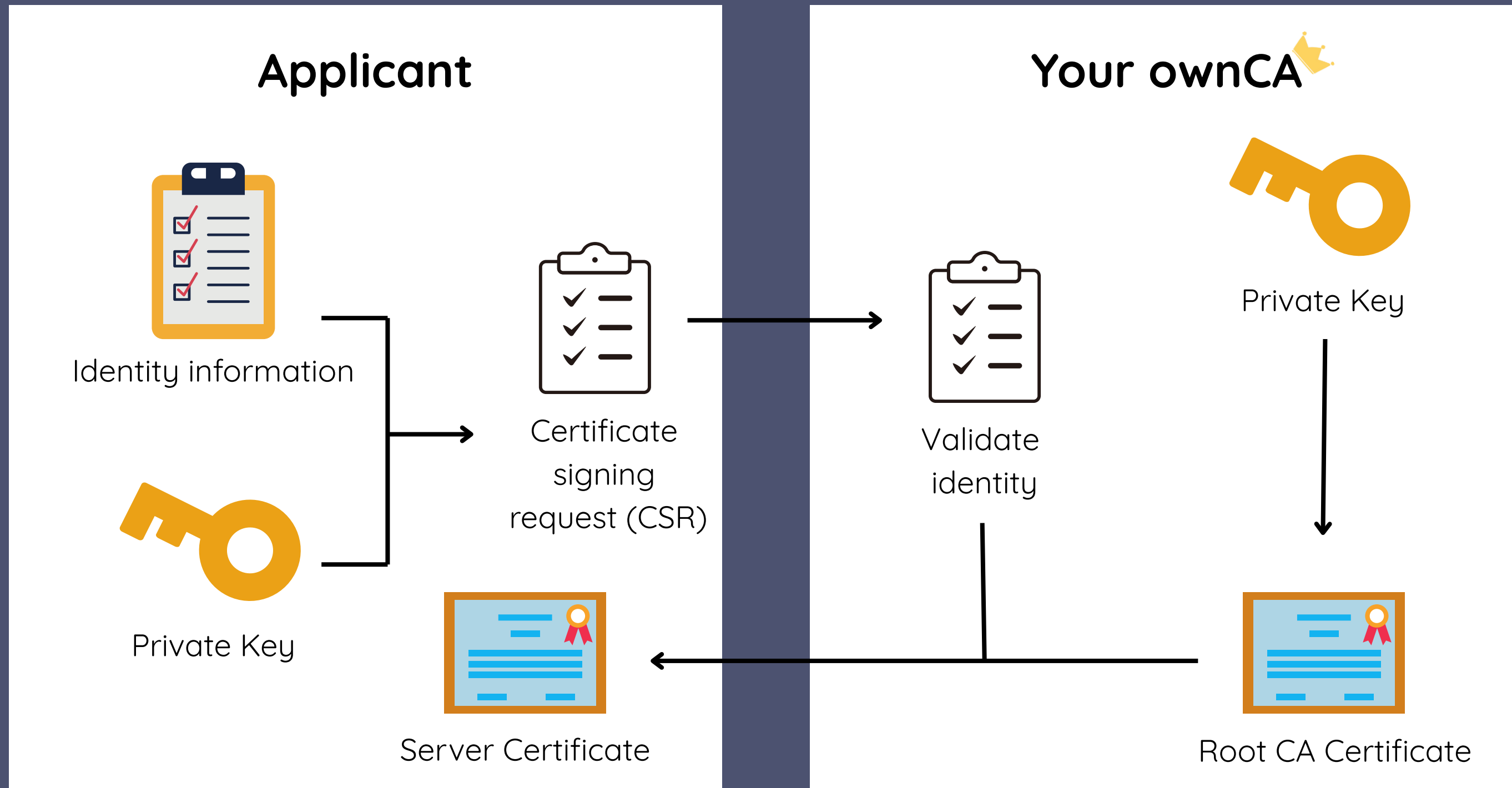
HOW CERTIFICATE SIGNED?

How certificate signed?



WHAT'S SELF-SIGNED CERTIFICATE?

How certificate signed?

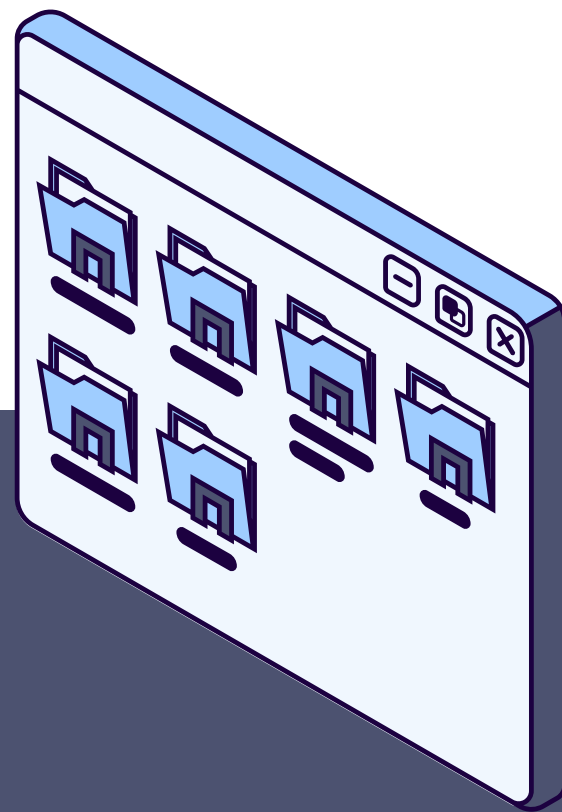


LET'S BEGIN THE SELF - SIGNED WORKSHOP

Make sure you already installed OpenSSL

```
openssl version -a
```

-a mean show all informations



OPENSSL LIBRARY

Create a private key for your CA

```
openssl genrsa -out myCA.key 2048
```

2048 = key size in bits

Output: myCA.key

OPENSSL LIBRARY

```
root@oak-lb-test-1: ~  
  
*** System restart required ***  
Last login: Sun Apr 27 18:13:41 2025  
root@oak-lb-test-1:~# ^[[200~  
: command not found  
root@oak-lb-test-1:~# ~  
^C  
root@oak-lb-test-1:~# openssl genrsa -out myCA.key 2048  
root@oak-lb-test-1:~# ls  
myCA.key  snap  
root@oak-lb-test-1:~# cat myCA.key  
-----BEGIN PRIVATE KEY-----  
MIIEvQIBADANBgkqhkiG9w0BAQEFAASCBCwggSjAgEAAoIBAQDnyClrM+nKD95q  
nMIyxG/kFNSjxR1KUPj0bAHcWdqM+GAknw+kSWN9HdGx0Eu/95aqxeCI+guqoIWQ  
llgXtG4e72rFvJQgMwjOo8C6KK+ARY5lwNiK25/2Qm9lx3F+IYxZdtFvEiZ5tKe6  
BijhJySuuePWIqsvDAyZ3MX0ocAVJP+tSzMeNavp53pvshTQv0bSudC5EZtkk2fj  
TCEF5BeDe4Y0BsdLFrNPcwK/ZodDNDXW0UglRsfhq2UskblLB8ceFzDtpblS2MeK  
NQRHJvoveEB6l4pKKNld09yUcc0nylV0RkECT1QuZrX90OWLH3oQQ90LIX70aQ2zT  
fbmdjSSTAgMBAAECggEAVEipoxWQHM2zc+mN0rcy9u4KOvlpQsZ6GSuTa6Ty/9Mr  
6vU+U1/qgStQPwg/G3HlhGkLUtQWXPgdSax3OLuAVdh666e8946HU9khMrCrzQm+  
GZzEQY5Gtl6GMq0ZXKPnbPrzgp/HhPdIDLHDuMggoyzuqEYYGLYZePPauI/9cNI/  
cRgKxyFHTDyjWMRrWwImE63su68x+GekJtn7bkrGx0rU0SoixUB0yG8T6i2db8+a  
PpaqNgFqT+xeuS3g/HlmB9hiXaaywogWbcVtOUaG9jb6uP9Ce0JVJ8HGMPun67V  
e6JRgRhdSyDASFbtWV5cNPheDzxR8AVB0NCaFtDvqQKBgQDr29uIeUYsLZemQVN8  
/E9vtPuIXxIoJri3kdKpc/CULORjhr4peaUwuTuugKkGwDIpKf6wLpZEkgKqUtGF  
Azjhzgv8GhOrrdliUnHOYlqsDExvGYWV6l9Gf5lj4yW+q1EbK80zhxKniHLbiULA  
WUotiG8p2nU+ryoJtduz6FYYeQKBgQD7ky2B6I/NyvQ4LMK9f2spRKPu6AqMggQ2  
L6rubEgo7/8lfbz+7xK5TeCqvKTRIB9l9/yVEYCzSgkja7mmDaYdhWVIK8rNRDtXu  
obTaB9dng8lxtNQvxGnc9dsyJDbf775WJkywmRkIoXkqYuLDli9CcowSVoYRz+17  
DBRqgdG6awKBgB6yxQYkUr/ZR026XvV7mZM+iSacyS5CNUSfVD39+yguF6Pkm8em  
L5wB4AS8dkOMrHqzHpUr2x8dqf81PGQFoVeyi9jKRBkqTYsAlBN6tbaC+0fSmJKE  
qNXyMjgYDoeCoLfoKuM0B3eKyPhOC0Qh2wfGJEHcTzpTbjNJWkk2Sk5xAoGBAOj4  
CIJu7Qy0T6jdupFlt8UFR2IGDA7GMERFrFMSrlhJt9rAtjKaFplxgIYJP5Mwcou  
R0KklZlzzSgA9HEffqE16BX/XxNHw9FqgroN3OodXqUPwhdiGm7mt1B/DCogh/Ca  
EdmQTWByuBQKctjMNogtBofShSgqtjx/M4M3j1lAoGAVSDPykSGKrdYJvptQMqa
```




OPENSSL LIBRARY

Create a self-signed root certificate for your CA

```
openssl req -x509 -new -nodes -key myCA.key -sha256 -days  
3650 -out myCA.pem
```

- **-x509** = create a self-signed certificate (not CSR)
- **-days 3650** = certificate validity (10 years)
- **Output:** myCA.pem (Root CA certificate)
-

● Now you have a CA ready: myCA.key and myCA.pem



```
root@oak-lb-test-1:~# openssl req -x509 -new -nodes -key myCA.key -sha256 -days 3650 -out myCA.pem
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:
State or Province Name (full name) [Some-State]:
Locality Name (eg, city) []:
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:
root@oak-lb-test-1:~#
```

```
root@oak-lb-test-1:~# ls
myCA.key  myCA.pem  snap
root@oak-lb-test-1:~# cat myCA.pem
-----BEGIN CERTIFICATE-----
MIIDazCCAIOgAwIBAgIUsvYvHr26k/WmnG3tWv/QMyIf1zgwDQYJKoZIhvcNAQEL
BQAwRTElMAkGA1UEBhMCQVUxEzARBgNVBAgMC1NvbWUtU3RhdGUxITAfBgNVBAoM
GE1udGVybWV0IFdpZGdpdHMgUHR5IEEx0ZDAeFw0yNTA0MjgxNDQlMDdaFw0zNTA0
MjYxNDQlMDdaMEUxCzAJBgNVBAYTAFVMRmEQYDVQIDApTb211LVN0YXR1MSEw
HwYDVQQKDBhJbnRlcm5ldCBXaWRnaXRzIFB0eSBMdGQwggEiMA0GCSqGSIb3DQEB
AQUAA4IBDwAwggEKAoIBAQNdyClrM+nKD95qhMIyxG/kFNSjxR1KUPj0bAHcWdqM
+GAknw+kSWN9HdGx0Eu/95aqxeCI+guqoIWQ1lgXtG4e72rFvJQgMwjOo8C6KK+A
RY5lwNiK25/2Qm9lx3F+IYxZdtFvEiZ5tKe6BijhJySuuePWIqsvDAYZ3MX0ocAV
JP+tSzMeNavp53pvshTQv0bSudC5EZtkk2fjTCEF5BeDe4Y0BsdLFrNpcwk/ZodD
NDXW0UglRsfhq2Uskb1LB8ceFzDtpb1S2MeKWQRHJvoeEB614pKKNld09yUcc0ny
lVORkECTlQuZrX90OWLH3oQQ90LIX70aQ2zTMbmdjSSTAgMBAAGjUzBRMB0GA1Ud
DgQWBBRjZzdjl++OiFKbN3GzfXP78ixBwjAfBgNVHSMEGDAWgBRjZzdjl++OiFKb
N3GzfXP78ixBwjAPBgNVHRMBAf8EBTADAQH/MA0GCSqGSIb3DQEBCwUAA4IBAQDD
b0Bs18yotNhgyYJkDsLulmp9eCshqCYtpvn+TBcQaK6rr4jrgMWNEbTRMthAy7Xe
4bkEQFDjDDD7AVG5c5kTgyiwONSE405D/qDIamPTbNK5ZeARmLEiv22zAuunNmX+
6rOUQ1XL0C+lnXMScf6S5HZS9/qsIksj4YnmayFF4vFRRBqfVXDERUDn/+17fmy8
iYj43CLdJQlIt8DxKcDj6g5KXm3f50XObUpw4E/SW8fiUu/BokWQn12vaWLxNtKu
LIvMsiQXWcwNVov18QNjhDF/K0J8jorDKQH4ST0fR/FGb3ZRL/lhg+qi915HvJwH
nFz0sYG2kJeZ6VDfIpei
-----END CERTIFICATE-----
root@oak-lb-test-1:~#
```

**USE YOUR CA TO SIGN A NEW
CERTIFICATE**

Create a private key for the server:

```
openssl genrsa -out myServer.key 2048
```

This will be the private key for your server.

OPENSSL LIBRARY

```
root@oak-lb-test-1:~# openssl genrsa -out myServer.key 2048
root@oak-lb-test-1:~# ls
myCA.key  myCA.pem  myServer.key  snap
root@oak-lb-test-1:~# cat myServer.key
-----BEGIN PRIVATE KEY-----
MIIEvgIBADANBgkqhkiG9w0BAQEFAASCBAgEAAoIBAQC7XCztYniH0nSS
DT4xn8hQ0N+kIi8qtUh8dzUnW8oqU8NhZZlpZAOY7PeVXWJ684ISymrl2ieWFWKl
KDe8yzuSrjzKytEBLJIEMpJFXlckV6ohQHsT6d6B7R0XZE6YsddyCGCO904NLUcb
T84wj8opwP2t9IXCofuKDlIqHHcQiTmXPClzyzsbJzJi78q8sD/ejp2WHLzddnhO
E8pE5WjdftpHtDEGAkcj0DhrJePwTYr5nfQVQnRKFPVdWwgvaruzIzxv9/w8caMxx
QDGBq/JDGZdF4Km3XU7qqvra5jjCkBGXhRdEC3SwP9pw6rYDkZCatNJt6xzzYl9V
h4ho/fovAgMBAAECggEAQy4S4e9b/n+E0tWyxxJfjpjKu7gBf3zVvT0nDImC0RacR
OHwgnPyCmbPmNoVsiV+E9U3lNJObfCIg48le5vAMZVTacXBlrrEXji+/6OX9rgdM
9mlTdFl4MigeNUP/NpsWAEpmrPyZ3PmuNGnSKfF08LHFhDlYOOix89f4gyb/vkqH
yhReR7W74wqoO3apM7Q4QfRUp90D0Y5KuHXk4Fgod2rUz8Nn7f5AKPCAwRaF+aGR
V/9VqPdASCFFJtp6E386AJbfKRvwPsq5m0lGuaYZZ4QGVwgGLAWaLD4ns3X2j2AJ
z9soRi8dJAyeGxvxoUfCQvTfHhVFE10ZaQZZBgWHeQKBgQDCEZ3Z0EsVNkNhnKjV
+geRD6TVEJLx+u7/fyJe/1QlBypBV8ikO0Ule17u+wJ+YPKofTmWE0MSB3rbsSVf
tX5jTi33jkEen4l87q8bE+a0OfkDL/7xESTm8Nl3jEq0prAHhlMQVHlP7Xw29Fe/
TO3Dr0dIjKq/si6hIS0liIiXcwKBgQD3JnlUrok/vMAMGjTSM0yS5U4W3qfJkkVz
ZY7S7S063PMgTg7bjwd0lSpBerH7nrl6s3KqAovZvKJizpxH8RoJCjFqVUD2YQA3
FsHAZw62IBOPxVOwQqV/UtSTI+2aS+PGIgfCn/xCYUlfZs6iKXsjSn/DejYDYgeS
AuV2YGdLVQKBgCLlA2cBsXSeEqMZkfwBLq29kKZkduOpnLVfFDtIbyuiH8VnZ5fV
qU3oHCCWDEvxcKRS8jWLQLCVslayYRYNXjfLFFKcSDSY7uaLI/WPryXVr3oYgiaw
POJwjeuPlMEXqWuhkYLghJeWiQLPfqlR2549C20REYy4LtgNXtRexKPrAoGBAIVl
q+4CCqhxS0L8/heagrEivURQx0VgAsZnQXCNeGSKjluXZiORjoMWtekDvX8vS7Nz
meczWqKaUxd7pxSZiNDqRIO9JcY6UvBpAwjDV0hsBcLq4eHYBxWy8nO8x6kpXvZ5
Mjz5SSHw+tBYyBlcwE7rDgzHMNIgni9sCsga/oXFAoGBALDQXgrVLO/EH5BUMfGw
ORiwOd2FgC5jErxfDB4lx4yRNN/qvPH40cgbXFOn+FlmRyPTHYL5H7I06OmaiCXz
aFDwLbY7lhqP+xPNkyZUOGP2P0oWSjKsUu/Thb9TJfQC8fJpcpODlkbLaKFtbi/h
```



OPENSSL LIBRARY

Create a Certificate Signing Request (CSR)

```
openssl req -new -key myServer.key -out myServer.csr
```

It will ask for Common Name (CN) → use your server's domain/IP

```
root@oak-lb-test-1:~# openssl req -new -key myServer.key -out myServer.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:
State or Province Name (full name) [Some-State]:
Locality Name (eg, city) []:
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
** Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
root@oak-lb-test-1:~#
```

```
root@oak-lb-test-1:~# ls
myCA.key  myCA.pem  myServer.csr  myServer.key  snap
root@oak-lb-test-1:~# cat myServer.csr
-----BEGIN CERTIFICATE REQUEST-----
MIICijCCAXICAQAwRTELMAkGA1UEBhMCQVUxEzARBgNVBAGMC1NvbWUtU3RhdGUx
ITAfBgNVBAoMGEludGVybmV0IFdpZGdpdHMgUHR5IEExOZDCCASIwDQYJKoZIhvcN
AQEBBQADggEPADCCAQoCggEBALtcLOlieIfSdJINPjGfyFDQ36QiLyq1SHx3NSdb
yipTw2FlmWlkA5js95VdYnrzghLKauXaJ5YVYqUoN7zLO5KuPMrK0QEsKgQykkVf
VyRXqiFAexPp3oHtHRdkTpixl3IIYI73Tg0tRxtPzjCPyinA/a30hcKh+4oPUioc
dxCJOZc8KXPL0xsnMmLvrywP96OnZYcvN12eE4TykTlaN1+ke0MQYCRyPQOGsl4
/BNivmd9BVCdEoU9V1bCC9qu7MjPG/3/DxxozGRAMZur8kMZl0XgqbddTuqq+trm
OMKQEZeFFOQLdLA/2nDqtgORkJq00m3rHPNiXlWHiGj9+i8CAwEAAaAAMA0GCSqG
SIb3DQEBCwUAA4IBAQA18xLsI3bVhd7df32WoiM64BfylIAODVjtNROIZhTng5YV
/0o2d/GPpfKXYlskoEzJS124uAT0tGlyu/Hwx+gpO/4W3+zifsamel3eLYiTe7X8
m/D/889nWRgiWIPw+AwwEpoRji//60ef7OaTo4A6+NcbM2vHLqPcN7w9X7MXv9+T
m/v4lnmjfJxaTaNuv8l2do0HBnchW55b3CHiC4+ZAXrjUkzKQvd964NGvZxjUy8b
HWFwfB9Q9ithrlsYWGK/ACJAODTxyl2xEH7sqPj+dal4HCowRMxE13eDs3Qt804E
RH55DOvPb+VKE4tE7DnJ5m3c0DULUJyZpq5zo2F2
-----END CERTIFICATE REQUEST-----
root@oak-lb-test-1:~#
```

NOTE: After entering the command, you will be prompted to fill in the identity information of the requester. You may choose to either fill it in or leave it blank, since you are creating a Self-Signed SSL Certificate.

OPENSSL LIBRARY

Sign the server CSR with your CA to issue the server certificate

```
openssl x509 -req -in myServer.csr -CA myCA.pem -CAkey  
myCA.key -CAcreateserial \  
-out myServer.crt -days 825 -sha256
```

- **x509** = Tells OpenSSL: "work with X.509 certificates" (standard certificate format).
- **-req** = Tells OpenSSL: "input is a Certificate Signing Request (CSR)", not a certificate.
- **-in myServer.csr** = Specifies the input file: the server's CSR you made earlier.
- **-CA myCA.pem** = Tells OpenSSL: "use myCA.pem as the Certificate Authority (CA) certificate."
- **-CAkey myCA.key** = Use myCA.key as the CA's private key to sign the new certificate.
- **-CAcreateserial** = If a serial number file (.srl) does not exist yet, automatically create one (ex: myCA.srl). Each cert must have a unique serial number.
- **-out myServer.crt** = Output the new signed certificate into a file called myServer.crt.
- **-days 825** = The certificate is valid for 825 days.

```
root@oak-lb-test-1:~# openssl x509 -req -in myServer.csr -CA myCA.pem -CAkey myCA.key -CAcreateserial \  
out myServer.crt -days 825 -sha256  
Certificate request self-signature ok  
subject=C = AU, ST = Some-State, O = Internet Widgits Pty Ltd  
root@oak-lb-test-1:~# ls  
myCA.key myCA.pem myServer.crt myServer.csr myServer.key snap  
root@oak-lb-test-1:~# cat myServer.crt  
-----BEGIN CERTIFICATE-----  
MIIDETCCAfkCFCEtUAH2rCJYsOQLqXTck67whl43MA0GCSqGSIb3DQEBCwUAMEUx  
CzAJBgNVBAYTAkFVMRMwEQYDVOQIDApTb211LVN0YXR1MSEwHwYDVQQKDBhJbnRl  
cm5ldCBXaWRnaXRzIFB0eSBMdGQwHhcNMjUwNDI4MTUxMzI4WhcNMjcwODAxMTUx  
MzI4WjBFMQswCQYDVQQGEwJBVTETMBEGA1UECAwKU29tZS1TdGF0ZTEhMB8GA1UE  
CgwYSW50ZXJuZXQgV2lka2Z1cyBQdHkgTHRkMIIBIjANBgkqhkiG9w0BAQEFAAOCA  
AQ8AMIIBCBgKCAQEAulws7WJ4h9J0kg0+MZ/IUNDfPCiVkrVifHclJlvKK1PDYWWZ  
aWQDmOz3lVlievOCespg5donlhVipSg3vMs7kq48ysrRASySBDKSRV9XJFqIUB7  
E+nege0dF2R0mLHXcghgjvdODS1HG0/OMI/KKcD9rfSFwqH7ig9SKhx3EIk5lzw  
p8s7GycyYu/KvLA/3o6dlhy83XZ4ThPKROVo3X6R7QxBgJHI9A4ayXj8E2K+Z30F  
UJ0ShTlXVsIL2q7syM8b/f8PHGjMZEAXm6vyQxmXREcpt11O6qr62uY4wpAR14UX  
RAAt0sD/acOq2A5GQmrTSbesc82JfVYeIaP36LwIDAQABMA0GCSqGSIb3DQEBCwUA  
A4IBAQCsrKzND1r5oyzOZOFeSrwn8w2QNk2seYbkt4wwfRSnnMzo+J0SCS1PMB5C  
Q14VMfRhLF3TYKMlnROu6Jh1hDAUXj5XLdWPK641Up18C7+tlAS10TVamRSduKYs  
0kbIEzY5PU7ukPMoft8yDl++k/blY5shNu5ytRQXnQ19Ykd6QVizR7J5cDS9Hm7b  
t1VB15DyDztR+t+/OD6y64uRMiv9V7qjPWA01Szq4lco7P513riw0kx9n/r4Wvcc  
+CH4U0gWdNAPkSj4CO/PCjHL4tnr59Zm6oypIDphVQY3Q4iZqVAVcn7jlowPQOAT  
CuJlYrGBzURsVN5P/ikJL0ir21LD  
-----END CERTIFICATE-----
```

**ADD CERTIFICATE ON
GDCC OPENSTACK CLOUD SERVICE**

ADD CERTIFICATE

III

GDCC

GOVERNMENT DATA CLOUD CONNECT

Home

Resources

Application

0

AP-GDCCTHAILAND

ap-southeast-20...

Enter keywords

Favorites: No data

Click [I](#) to add a service to favorites. 10 services can be added.

Computing

Image Management Service

Elastic Cloud Server

Auto Scaling

Cloud Container Engine

Big Data

MapReduce

Storage

Elastic Volume Service

Cloud Backup and Recovery

Data Analysis

Server Migration Service

Offline Services

Offline Services

Network

Virtual Private Cloud

Virtual Private Network

Elastic Load Balance

Direct Connect

Domain Name Service

Network ACLs

NAT Gateway

Elastic IP

VPC Endpoint

Security

Web Application Firewall

Application

Simple Message Notification

SoftWare Repository for Cont...

Application Orchestration Ser...

Application Operation Manag...

Mgmt & Deployment

Cloud Eye

Tag Management Service

Recycle Bin

Tag Management

Cloud Trace Service

Log Tank Service

Guide Help

ADD CERTIFICATE

[illegible]

ADD CERTIFICATE

IP

Network Console

Dashboard

Virtual Private Cloud

Subnets

Route Tables

Access Control

VPC Flow Logs

Elastic IP and Bandwidth

NAT Gateway

Elastic Load Balance

Load Balancers

Certificates

VPC Peering

VPC Endpoint

Virtual Private Network

Direct Connect

Elastic Cloud Server

Certificates ?

Create Certificate

Name

Type

Domain Name

Listener (Frontend Protocol/Port)

Description

Last Updated

Operation

No data available.

ADD CERTIFICATE

☰

Network Console

Dashboard

Virtual Private Cloud

Subnets

Route Tables

Access Control

VPC Flow Logs

Elastic IP and Bandwidth

NAT Gateway

Elastic Load Balance

Load Balancers

Certificates

VPC Peering

VPC Endpoint

Virtual Private Network

Direct Connect

Elastic Cloud Server

Certificates ?

Name	Updated	Operation
------	---------	-----------

Create Certificate

Create Certificate

* Certificate Name

cert-73e9

* Certificate Type

Server certificate

CA certificate ?

* Certificate

PEM encoded ?

Upload

View Example

* Private Key

PEM encoded ?

Upload

View Example

Domain Name

The domain name must be specified if the certificate will be used for SNI. Only one domain name can be specified for each certificate.

Description

OK

Cancel

ADD CERTIFICATE

Network Console

Dashboard

Virtual Private Cloud

Subnets

Route Tables

Access Control ▾

VPC Flow Logs

Elastic IP and Bandwidth ▾

NAT Gateway

Elastic Load Balance ▲

Load Balancers

Certificates

VPC Peering

VPC Endpoint ▾

Virtual Private Network 🔗

Direct Connect 🔗

Elastic Cloud Server 🔗

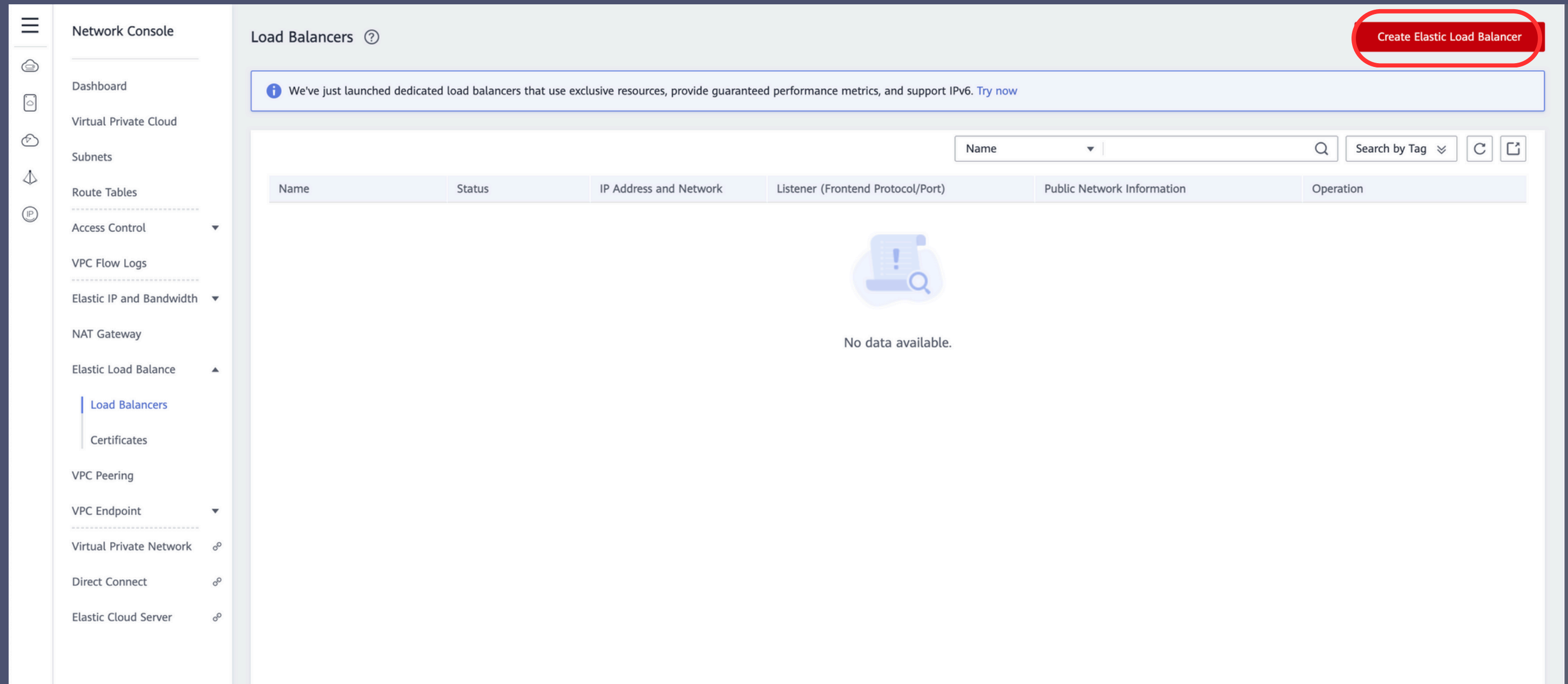
Certificates ?

Create Certificate

Name ▼ | Q C

Name	Type	Domain Name	Listener (Frontend Protocol/Port)	Description	Last Updated	Operation
cert-oak	Server certificate	--	--	--	Apr 27, 2025 19:42:18 GMT+07:00	Modify Delete

ADD CERTIFICATE



ADD CERTIFICATE

<

Create Elastic Load Balancer ?

*

Region

AP-GDCCTHAILAND (nt... ▾)

Regions are geographic areas isolated from each other. Resources are region-specific and cannot be used across regions through internal network connections. For low network latency and quick resource access, select the nearest region.

*

AZ

AZ-NONTHABURI ✕ ▾

You can choose to deploy the load balancer in multiple AZs for higher availability.

Network Type

☒ Public IPv4 network (Public network traffic) ☐ Private IPv4 network (Private network traffic) ☐ IPv6 network (Public and private network traffic) ?

*

VPC

vpc-default ▾

↻ View VPC

*

EIP

☒ New EIP ☐ Use existing ?

*

Bandwidth

5

10

20

50

100

Custom

−

 50

+

The value ranges from 1 to 1,000 Mbit/s.

*

Name

OAKHTTPS-LB

Advanced Settings ▾

Description | Tag

Load Balancing Diagram

Network Type

Public IPv4 network

Private IPv4 network

IPv6

Public network

Private network

Load Balancer Type

ELB

AP-GDCCTHAILAND(nt-planning-dev-3)

Server

Server

Server

Create Now

ADD CERTIFICATE

☰

Network Console

Dashboard

Virtual Private Cloud

Subnets

Route Tables

Access Control

VPC Flow Logs

Elastic IP and Bandwidth

NAT Gateway

Elastic Load Balance

Load Balancers

Certificates

VPC Peering

VPC Endpoint

Virtual Private Network

Direct Connect

Elastic Cloud Server

Load Balancers

Create Elastic Load Balancer

We've just launched dedicated load balancers that use exclusive resources, provide guaranteed performance metrics, and support IPv6. [Try now](#)

Name

Q

Search by Tag

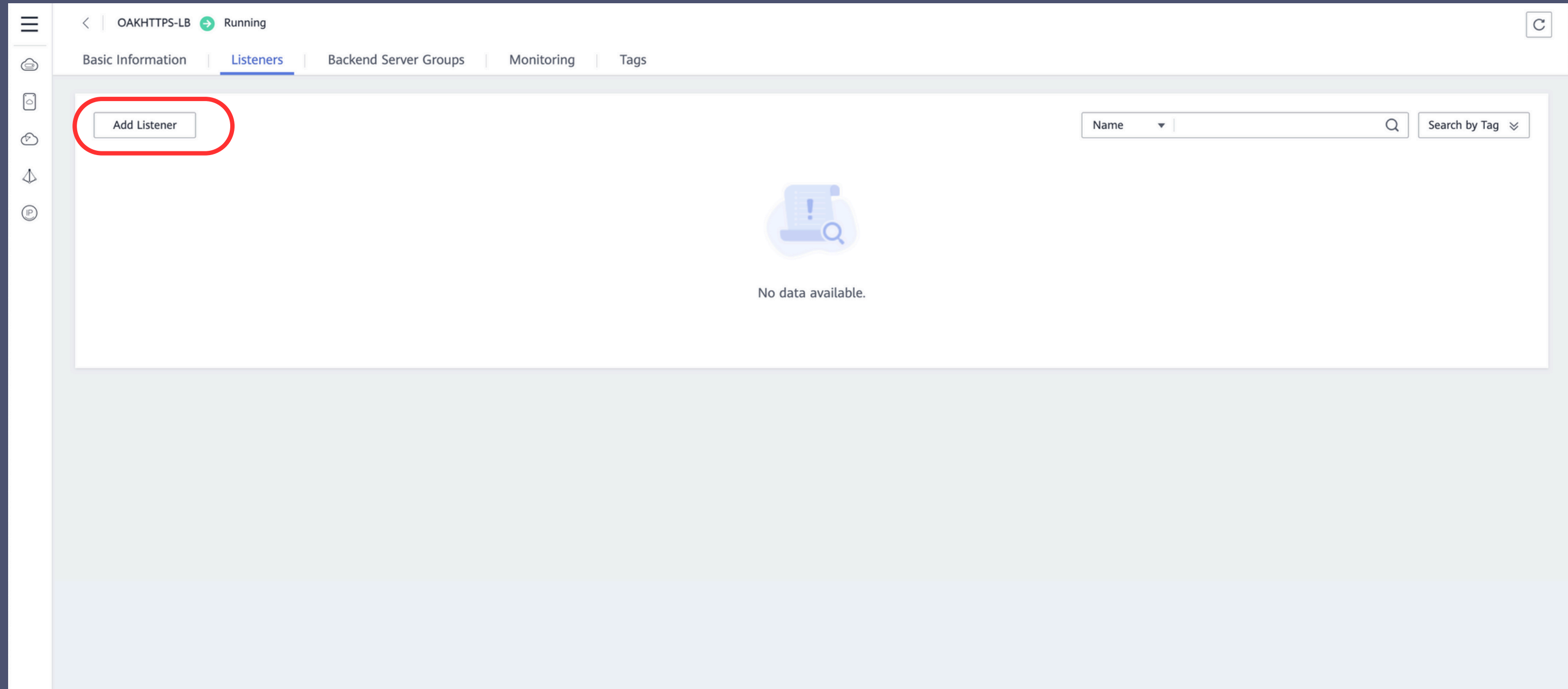
↕

↺

↻

Name	Status	IP Address and Network	Listener (Frontend Protocol/Port)	Public Network Information	Operation
OAKHTTPS-LB	<div><div></div>Running</div>	209.15.116.40 (IPv4 EIP) vpc-default (VPC)	<div>Add listener</div>	IPv4 50 Mbit/s	<div>Modify IPv4 Bandwidth</div> <div>More</div>

ADD CERTIFICATE



ADD CERTIFICATE

OAKHTTPS-LB Running

Basic Information

Listeners

Backend Server Groups

Monitoring

Tags

Add Listener

Add Listener

1 Configure Listener

2 Configure Backend Server Group

3 Finish

* Name

listener-264b

* Frontend Protocol/Port

HTTPS

443

Value range: 1 to 65535

Select TCP or UDP for load balancing at Layer 4. Select HTTP or HTTPS for load balancing at Layer 7.

When HTTPS is selected, the backend protocol can only be HTTP.

* Server Certificate

cert-oak

View Certificate

Enable SNI

Advanced Settings

Cancel

Next

ADD CERTIFICATE

Basic Information

Listeners

Backend Server Groups

Monitoring

Tags

Add Listener

Add Listener

×

1 Configure Listener

2 Configure Backend Server Group

3 Finish

Backend Server Group

Create newUse existing

★ Name

server_group-2660

★ Backend Protocol

HTTP

★ Load Balancing Algorithm

Weighted round robin

?

Sticky Session

☐

?

Description

0/255

Health Check Configuration

Enable Health Check ?☒

Previous

Cancel

Finish

Health Check Configuration

Enable Health Check ?☒

★ Protocol

HTTP

Domain Name

Port ?

80

Value range

Health Check Configuration

Enable Health Check ?

☒

* Protocol

HTTP ▼

Domain Name

Port ?

80

Value range: 1 to 65535

If you do not specify a port number, the port used by the backend server to receive traffic will be used.

Advanced Settings ▼

ADD CERTIFICATE

OAKHTTPS-LB

Running

Basic Information

Listeners

Backend Server Groups

Monitoring

Tags

Add Listener

HTTPS-TEST

HTTPS/443

Forwarding Policies Add

Basic Information

Backend Server Groups

Tags

Name

server_group-2660

ID

0d611143-656f-421f-95a5-ea4c3c8e8bdf

Load Balancing Algorithm

Weighted round robin

Backend Protocol

HTTP

Sticky Session

Disabled

Health Check

Enabled | Configure

IP Address Type

IPv4

Add

Modify Weight

Remove

All

Name

Name

Status

Private IP Address

Health Check Result

Weight

Backend Port

No data available.

ADD CERTIFICATE

OAKHTTPS-LB Running

Basic Information

Listeners

Backend Server Groups

Monitoring

Tags

Add Listener

HTTPS-TEST
HTTPS/443

Forwarding Policies Add

Add Backend Server

Ensure that the security group that contains the backend servers has rules allowing access from the backend subnet of the load balancer. If access is not allowed, health checks will fail. Learn more

Batch Add Ports

OK

Private IP Address	Server	Backend Port ?	Weight ?	Operation
192.168.0.49	OAK-LB-TEST-2 2 vCPUs 4 GB s.c2m4	80	1	<a>Copy <a>Remove
192.168.0.210	OAK-LB-TEST-1 2 vCPUs 4 GB s.c2m4	80	1	<a>Copy <a>Remove

Previous

Cancel

Finish

No data available.

ADD CERTIFICATE

Basic Information

Listeners

Backend Server Groups

Monitoring

Tags

server_group-2660 | HTTP

Basic Information

Name

server_group-2660

Listener

HTTPS-TEST

Load Balancing Algorithm

Weighted round robin

Sticky Session

Disabled

IP Address Type

IPv4

ID

0d611143-656f-421f-95a5-ea4c3c8e8bdf

Backend Protocol

HTTP

Health Check

Enabled | Configure

Description

--

Add

Modify Weight

Remove

Available servers: 2

All

Name

<input type="checkbox"/>	Name	Status	Private IP Address	Health Check Result	Weight	Backend Port
<input type="checkbox"/>	OAK-LB-TEST-1	Running	192.168.0.210	Healthy	1	80
<input type="checkbox"/>	OAK-LB-TEST-2	Running	192.168.0.49	Healthy	1	80



Your connection is not private

Attackers might be trying to steal your information from **209.15.116.212** (for example, passwords, messages, or credit cards). [Learn more about this warning](#)

NET::ERR_CERT_AUTHORITY_INVALID



[Turn on enhanced protection](#) to get Chrome's highest level of security

Hide advanced

Back to safety

This server could not prove that it is **209.15.116.212**; its security certificate is not trusted by your computer's operating system. This may be caused by a misconfiguration or an attacker intercepting your connection.

[Proceed to 209.15.116.212 \(unsafe\)](#)