# *How to add an SSL certificate for the application before deploying it on AWS Cloud*

1. *Choose the SSL Certificate*

*You can either :*

- *Obtain a certificate from the AWS Certificate Manager (ACM)*

*(recommended for AWS-managed services like ELB, CloudFront, or API Gateway).*

- *Purchase a third-party SSL certificate(e.g., from GoDaddy, DigiCert, etc).*

*If you need it for specific use cases or external deployments.*

- *Generate a self-signed certificate for testing purposes (not recommended for production).*

2. *Steps for AWS Certificate Manager (ACM)*

a. *Request an SSL Certificate*

1. *Log in to the AWS Management Console.*
2. *Navigate to Certificate Manager.*
3. *Click Request a Certificate  and select :*

- *Public Certificate (recommended for public-facing applications).*
- *Private Certificate (For internal services )*

4. *Provide the domain name(s)(e.g., example.com and www.example.com).*

5. *Choose validation method :*

- *DNS Validation: Add a CNAME record to your DNS.*
- *Email Validation: Respond to an email sent to the domain administrator.*

6. *Validate the domain and confirm the certificate.*

*b.Attach the SSL Certificate*

- *Use the certificate in AWS services like :*
- *Elastic Load Balancer (ELB): Configure the HTTPS listener and attach the ACM certificate.*
- *ClodFrint: Add the certificate in the distribution settings.*
- *API Gateway: Link the ACM certificate to your custom domain.*

*3. Steps to Third-Party Certificates*

*a. Generate a Certificate Signing Request(CSR)*

1. On your server, generate a private key and CSR using OpenSSL:

```
Openssl req -new -newkey rsa:-48 -node -keyout
private.key-out certificate.csr
```

2. Provide the CSR to the certificate authority(CA) when requesting the SSL certificate.

B. Upload the Certificate

1. Download the signed certificate from the CA (typically .crt or .pem files).
2. In AWS:

- Go to Certificate Manager(for ACM-managed services) or IAM (for custom deployments).
- Upload the certificate along with the private key and any intermediate certificates.

C. Configure Your Application

- For Ec2 or ECS:
- Install the certificate on the web server(e.g., Apache, Nginx).
- Update your web server configuration for SSL:
- Nginx:

```
Server {
Listen 443 SSL;
Server_name example.com;

Ssl_certificate /path/to/certificate.crt;
Ssl_certificate_key /path/to/private.key;
}
```

Apache :

```
 <virtualHost *:443>
       ServerName example.com
      SSLEngine on
     SSLCertificateFile /path/to/certificate.crt
     SSLCertificateKeyFile /path/to/private.key

</VirtualHost>
```

Restart the web server

4. Automate SSL Management

- Use Let’s Encrypt with tools like Certbot for free certificates and automatic renewal.
- For ACM, AWS automatically renews certificates issued by ACM.

*5. Test SSL Deployment*

- *Verify that your website is accessible over HTTPS using a browser.*
- *Use tools like SSL Labs to check for misconfigurations or vulnerabilities.*

*6. Monitor and Maintain*

- *Regularly monitor the certificate's expiration date.*
- *Set Up alerts for renewal in third-party cases.*

*Free SSL Certificate*

1. *AWS Certificate Manager(ACM)*

- *Cost: Free (only available for AWS-managed services like Elastic Load Balancer, CloudFront, and API Gateway.*
- *Use Case: Perfect for AWS-hosted resources and fully managed, including automatic renewal.*
- *Limitation: You cannot export the certificate for use outside AWS.*

2. *Let's Encrypt*

- *Cost: Free (open certificate authority).*
- *Use Case: Ideal for custom servers(e.g., EC2) or non -AWS Environments.*
- *Pros: Free, widely supported, and supports automatic renewal with tools like Certbot.*

3. *Self-Signed Certificates*

- *Cost: Varies based on the CA and features, typically $10-$500 per year.*
- *Examples: GoDaddy, DigitCert, GlobalSign, etc.*
- *Use Case: Required for advanced validation (e.g., Extended Validation or WildCard certificates) or for use outside AWS services.*

2. *AWS Marketplace or Other Providers*

- *Some certificates purchased via third-party marketplace may be additional costs, especially for specific features or multi-domain support.*

*When to Choose Free vs Paid*

- *Free SSL: Use AWS ACM or Let's Encrypt for most scenarios unless specific requirements demand otherwise.*
- *Paid SSL: Consider if you:*

1. *An EV(External Validation) or (Organization Validation) certificate is needed to display higher trust.*
2. *Require a Wildcard SSL certificate for subdomains.*
3. *Need multi-domain or cross-platform compatbility beyond AWS services.*