Edge Security

Keys, Certificates and Keystores and TLS

Agenda

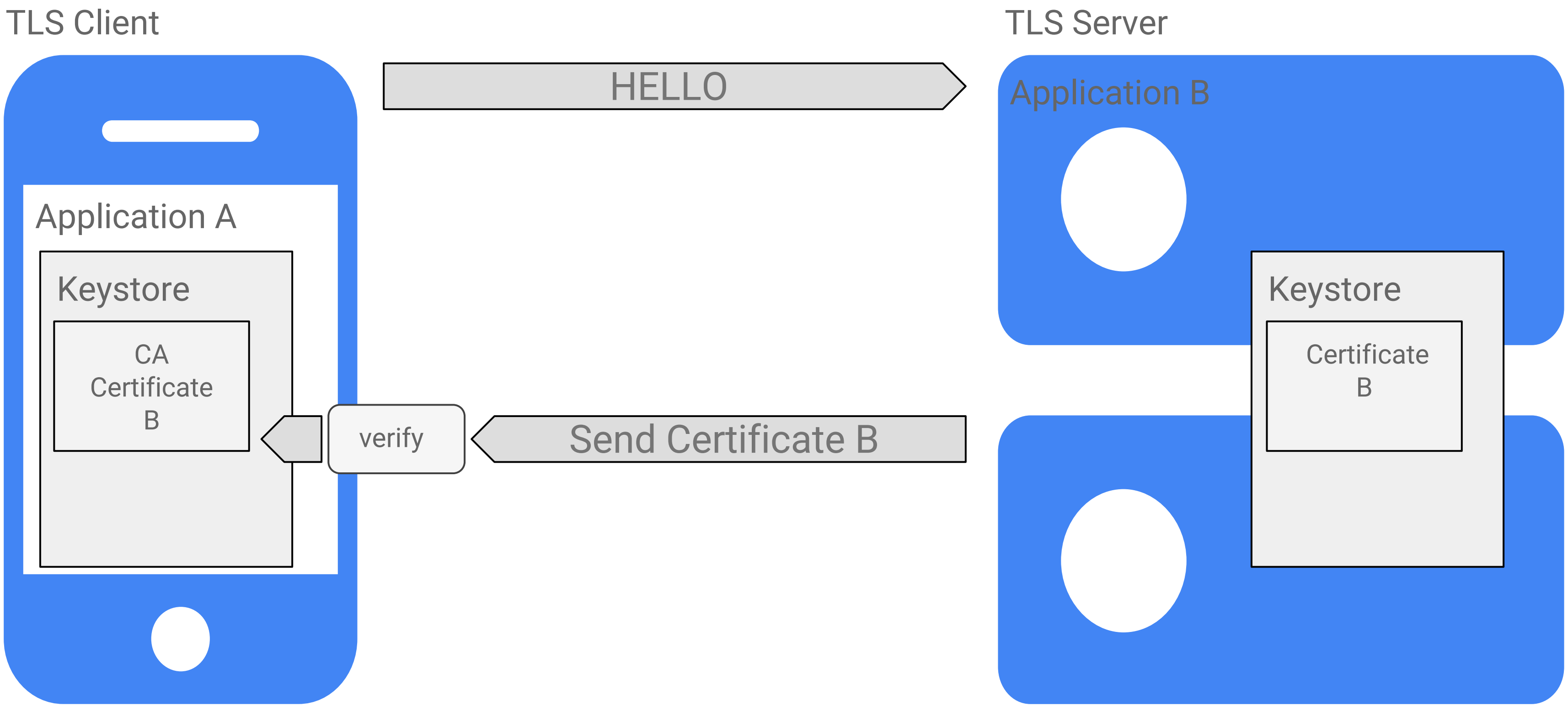
- TLS and SSL
- Private Keys
- Certificates and signing requests
- Java Keystores

TLS and SSL

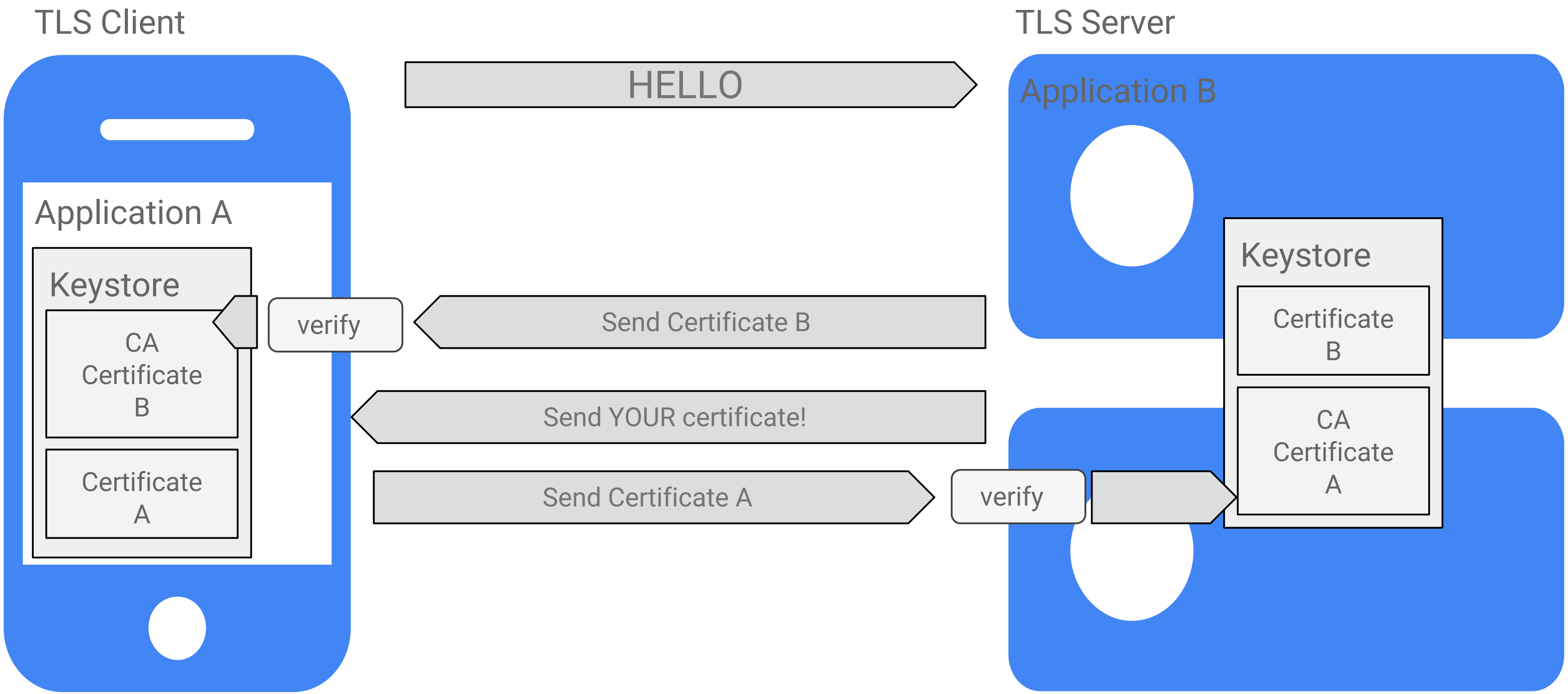
- TLS (Transport Layer Security, whose predecessor is SSL) is the standard security technology for ensuring secure, encrypted messaging across your API environment, from apps to Apigee Edge to your back-end services.

- Edge supports one-way TLS and two-way TLS in both our cloud and on-premises deployments.

TLS and SSL - 1 Way TLS



TLS and SSL - 2 Way TLS



TLS Client

TLS Server

Application A

Keystore

CA Certificate B

Certificate A

Application B

Keystore

Certificate B

CA Certificate A

HELLO

verify

Send Certificate B

Send YOUR certificate!

Send Certificate A

verify

Private Keys

- Private keys are used to identify the server
- They are used when creating certificate requests as well as when creating a keystore/truststore

Creating the key

Create somewhere to store the private key:

```
mkdir -p ~/certs/servers/tlsdemo.apigee-training.google.com
chmod 700 -R ~/certs
```

Create and secure the key:

```
openssl genrsa -out
"certs/servers/tlsdemo.apigee-training.google.com/private.pem" 2048
chmod 600 -R certs/servers/tlsdemo.apigee-training.google.com/private.pem
```

Certificates and signing requests (CSRs)

For a machine serving request for `https://tlsdemo.apigee-training.google.com` a signed, trusted certificate is required

To request a signed certificate, create a request using the openssl tool and the private key for the server, i.e.

```
openssl req -new -key certs/servers/certdemo.yourdomain.xyz/private.pem
-out certs/tmp/tlsdemo.apigee-training.google.com.csr.pem -subj
"/C=AU/ST=NSW/L=Sydney/O=Cert Demo/CN=tlsdemo.apigee-training.google.com"
```

This creates a CSR (Certificate Signing Request) that is used to request your public certificate.

If you have a certificate and a private key, you can check they match by viewing the modulus - this should be the same for both the key and the certificate:

```
openssl rsa -noout -modulus -in private.key | openssl md5
openssl x509 -noout -modulus -in server.cert | openssl md5
```

apigee

Creating a Java Keystore

Before securing Edge, you need to create a Java keystore.

When creating a keystore, it is good practice to create it with an alias to make it easy to identify:

```
openssl pkcs12 -export -clcerts -in
certs/signed/tlsdemo.apigee-training.google.com.cert.pem -inkey
certs/servers/tlsdemo.apigee-training.google.com/private.pem -out
/tmp/prodapi_keystore.pkcs12 -name prodapi
```

Next, use the Java keytool utility to convert to JKS format:

```
keytool -importkeystore -srckeystore /tmp/prodapi_keystore.pkcs12
-srcstoretype pkcs12 -destkeystore /tmp/prodapi_keystore.jks
-deststoretype jks -alias prodapi
```

Thank You