

# OpenSSL in 21CS VSE<sup>n</sup>

Shahin R Krishna  
VSE<sup>n</sup> Software Engineer

# Introduction to SSL

# What is Secure Sockets Layer (SSL)



Encryption based network security protocol



Developed & released by Netscape in 1995.



Ensures privacy, authentication and data integrity.



SSL evolved into Transport Layer Security (TLS)

# SSL & TLS

## Secure Sockets Layer (SSL)

- Originally Designed and Released by Netscape
- SSL 1.0 never publicly released
- SSL 2.0 released in 1995
  - Major vulnerabilities found and lead to development of SSL 3.0
- SSL 3.0 released in 1996
  - Was widely used till 2014 – Google Security team finds a major security vulnerability

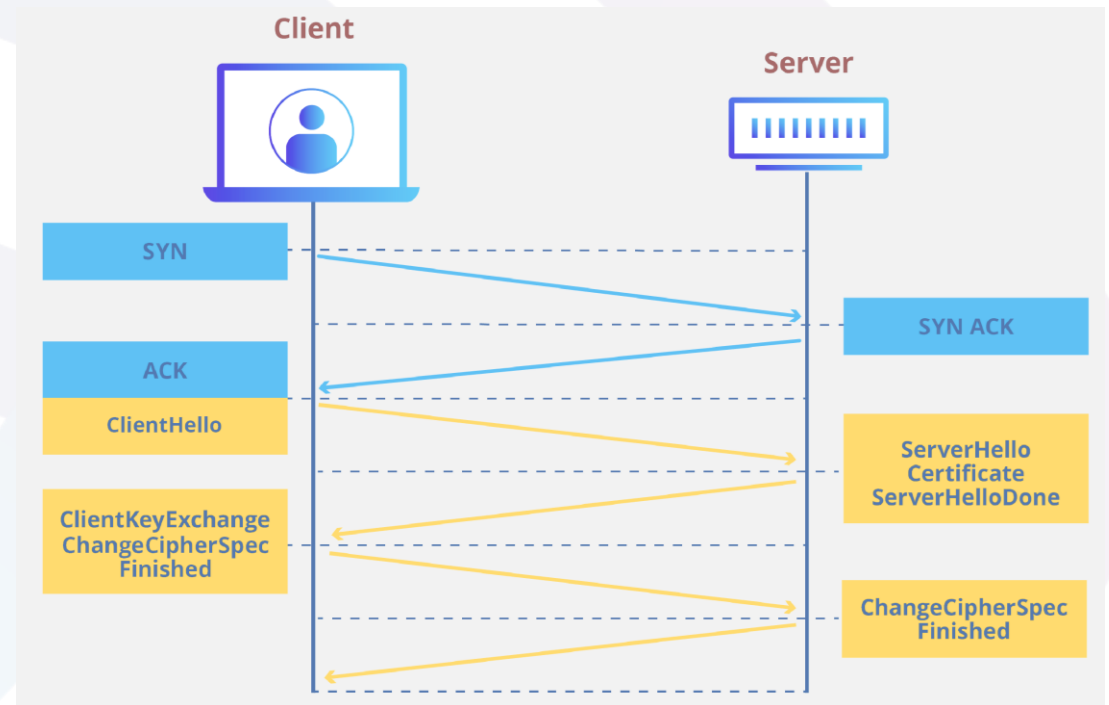
## Transport Layer Security (TLS)

- Designed as an upgrade of SSL 3.0 in 1999
- SSL 3.0 & TLS 1.0 can't interoperate
- Current Version is TLS V1.3 [TLS V1.0, TLS V1.1, TLS V1.2]

# Secure Communication

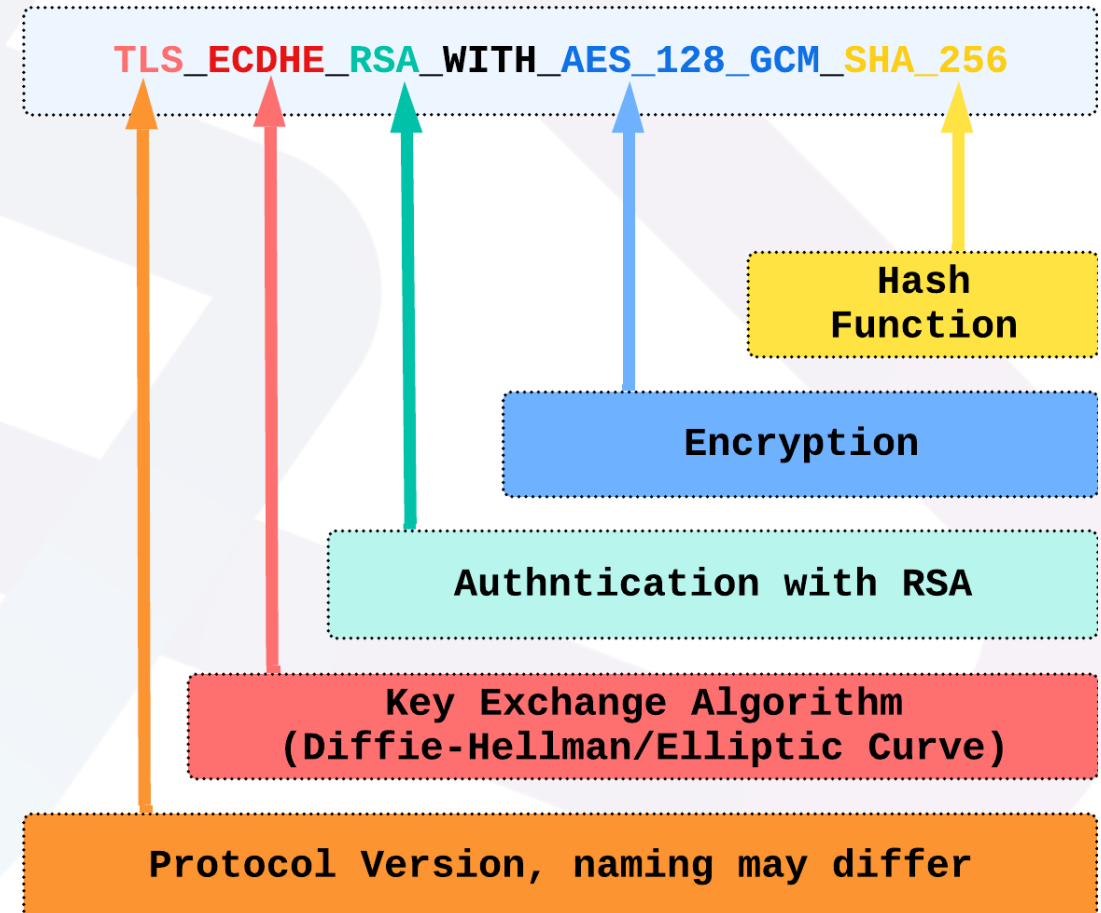
## TLS Handshake

- Version of TLS to Use
- Cipher suite selection
- Authenticate the identity of the server using server's certificate
- Key Exchange
- Generate session keys for encrypting messages



# Cipher Suite

- A collection of cryptographic algorithms.
  - Hash Functions
  - Asymmetric encryption algorithm for handshake
  - Symmetric encryption algorithm to maintain session security



# SSL Certificate

- A small data file that contains
  - Domain Name
  - Owner (Company, Device or person)
  - Subdomain names
  - Issuing Certificate Authority (CA or Intermediate CA)
  - CA's or Intermediate CA's digital signature
  - Issuance Date
  - Expiration Date
  - Public Key

```
C:\Users\openst\Documents\certs\openst\X509 - in path\sexx.k1024\cert.pem - text
Certificate:
Data:
  Version: 3 (0x2)
  Serial Number: 4096 (0x1000)
  Signature Algorithm: sha256WithRSAEncryption
  Issuer: C = DE, ST = Baden Wuerttemberg, O = 21st Century Software, OU = VSEn R&D, CN = Intermediate CA
  Validity
    Not Before: Jun  4 07:57:27 2024 GMT
    Not After : Jun 14 07:57:27 2025 GMT
  Subject: C = DE, ST = Baden Wuerttemberg, L = Stuttgart, O = 21st Century Software, OU = VSEn R&D, CN = PTHVSEXX-1024
  Subject Public Key Info:
    Public Key Algorithm: rsaEncryption
    RSA Public-Key: (1024 bit)
    Modulus:
      00:b2:6b:49:64:bf:46:f7:67:e2:0e:5d:81:5e:cc:
      84:29:60:fa:c5:6d:0b:98:79:31:e9:08:e0:8e:10:
      82:7d:fc:97:05:e1:e9:07:6e:a3:f9:ff:23:60:6a:
      ce:80:fc:85:11:f1:43:76:2d:aa:3e:1e:e7:e2:c7:
      3e:cd:59:db:e5:48:c5:15:c9:15:a8:7e:53:1e:86:
      9c:a9:e9:a3:59:b1:42:cf:55:24:46:5d:41:80:e8:
      ee:9a:b5:ec:9e:a6:f7:dd:9d:7c:8e:e9:8b:cb:bf:
      e2:0b:b0:19:55:b6:0b:6f:74:c4:41:d2:b4:2a:01:
      5a:11:55:8a:d5:49:a4:e1:4d
    Exponent: 65537 (0x10001)
  X509v3 extensions:
    X509v3 Basic Constraints:
      CA:FALSE
    Netscape Cert Type:
      SSL Server
    X509v3 Key Usage: critical
      Digital Signature, Key Encipherment
    X509v3 Extended Key Usage:
      TLS Web Server Authentication
    X509v3 Authority Key Identifier:
      keyid:A8:0F:19:5E:F3:4F:96:00:A1:B2:EB:9E:38:C3:6E:F6:58:6F:53:AB

  Signature Algorithm: sha256WithRSAEncryption
  69:22:89:a9:90:3f:ee:93:4d:51:0f:ce:12:0b:60:e6:52:e0:
  23:9d:13:46:3f:88:d2:e8:e0:6a:2e:df:6a:61:b0:58:37:61:
  34:51:44:92:48:69:e1:06:25:f9:cf:0f:58:6e:cb:85:6f:cd:
  32:07:ed:4f:ad:52:49:53:b1:77:8e:cd:a8:f9:c3:ff:6a:2e:
  f0:e0:d3:3e:a9:b5:5b:47:56:11:fa:6b:2c:52:bc:8d:4d:36:
  34:ed:0c:0a:97:db:4d:36:7b:2b:de:17:19:24:30:7c:e8:7e:
  7f:e9:76:62:70:37:ab:42:ce:d1:6a:0c:28:be:a7:6e:ea:fb:
  23:ed
```

# OpenSSL in VSE<sup>n</sup>



# OpenSSL in VSE<sup>n</sup>

- OpenSSL 1.1.1t (Current)
  - VP00081 (VA00077)
    - CVE-2023-0286
    - CVE-2023-0215
    - CVE-2022-4450
    - CVE-2022-4304
  - VP00107 (VA00113)
    - CVE-2023-0464
    - CVE-2023-0465
    - CVE-2023-2650
    - CVE-2023-3446
    - CVE-2023-3817
- OpenSSL 3.0 (Underway)

# Preparing for SSL enablement



Setup TCP/IP Socket API Multiplexer – EDCTCPMC



Generate Certificates



Upload Key and Certificate as PEM to VSE<sup>n</sup>



Configure applications to use SSL

# TCP/IP Socket API Multiplexer - EDCTCPMC

Sample Job: ICCF.62(EDCTCPMC)

```
EDCTCPME SYSID='nn',PHASE='xxxxxxxx',SSLPHASE='IJBSSLLE',  
EZAPHASE='xxxxxxxx',EZASSL='IJBZAOS'
```

\*\* For OLTP Web Support you may have to add the following in your OLTP startup job

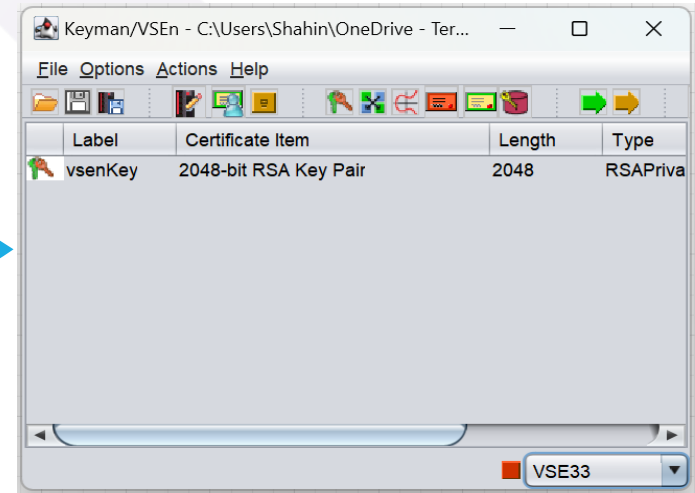
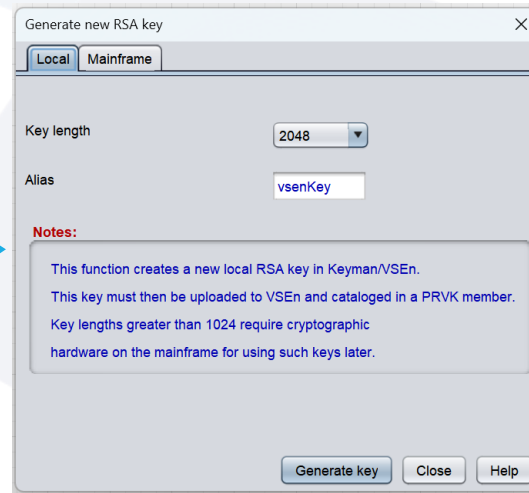
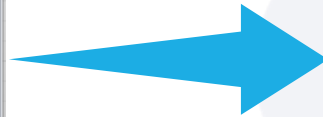
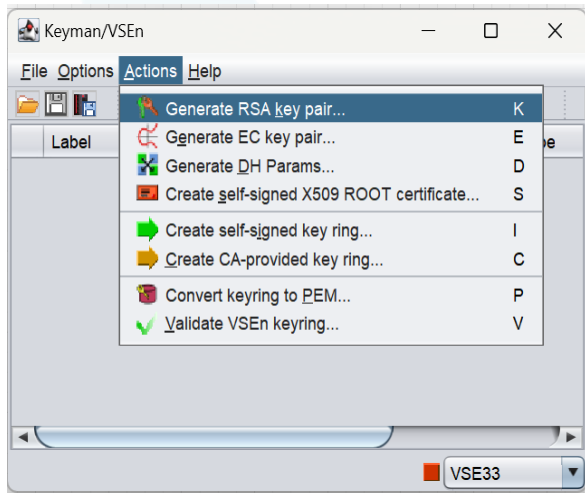
```
// SETPARM [SYSTEM,]BPX$GSK='IJBGSKOS'
```

# Generating Certificates

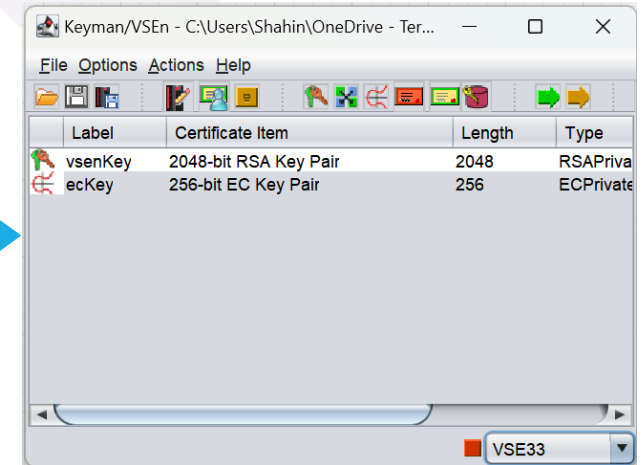
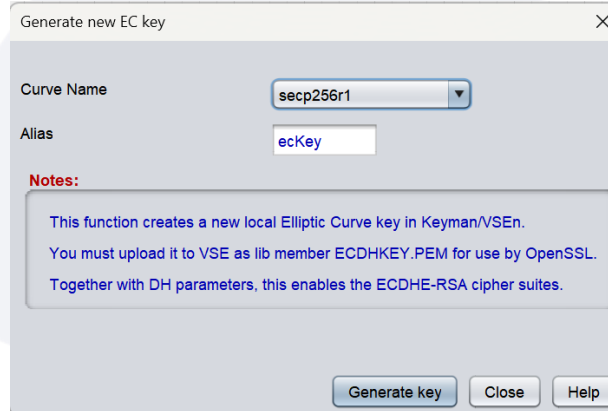
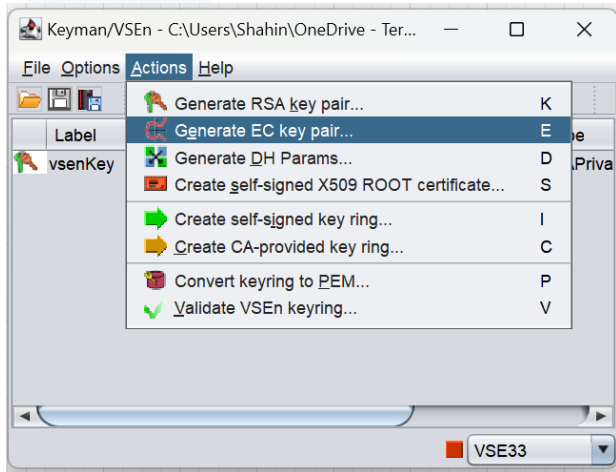
- To generate certificates, You can use
  - VSE<sup>n</sup> Keyman
  - OpenSSL CLI Tool
  - Java Keytool

# Generating Certificate using VSE<sup>n</sup> Keyman

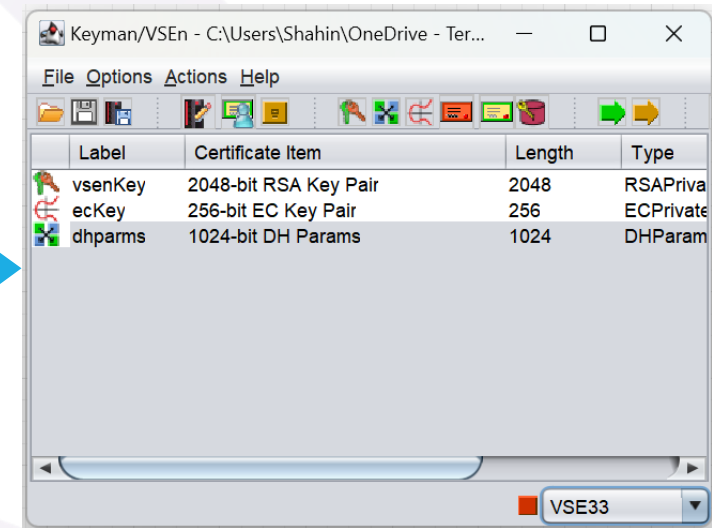
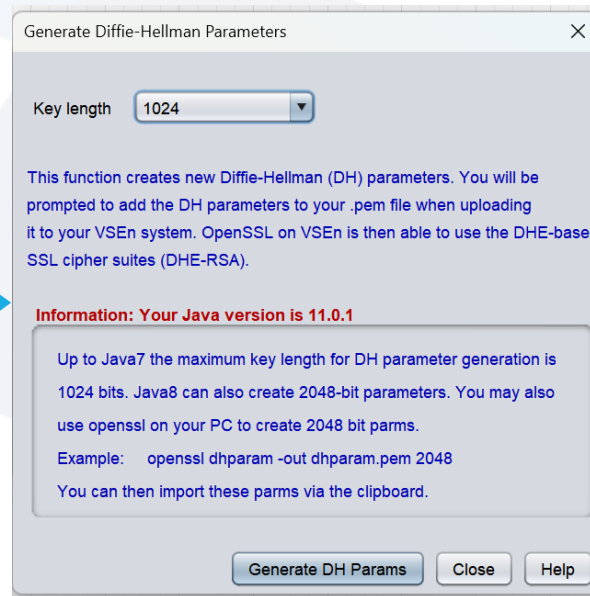
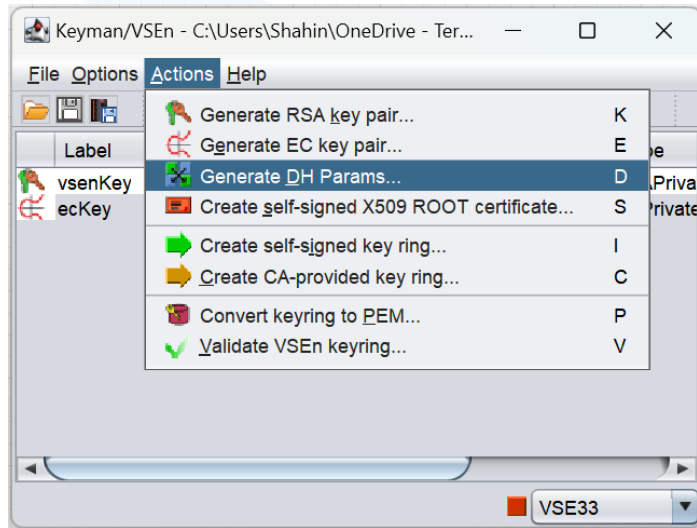
# Generate Key Pair



# Generate EC Key Pair

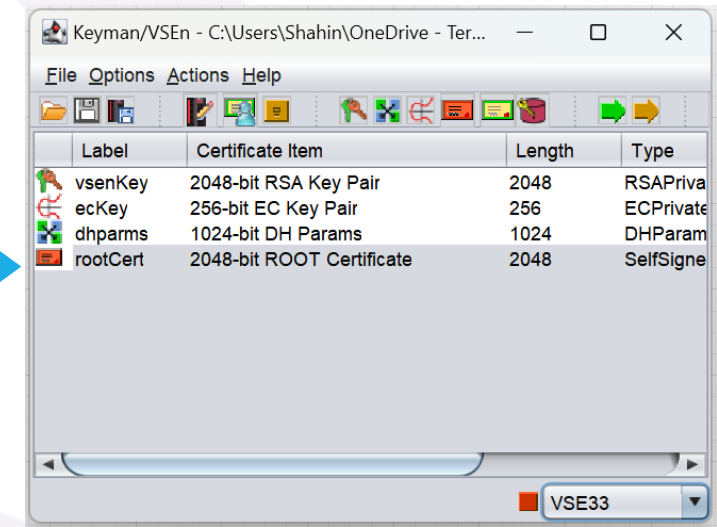
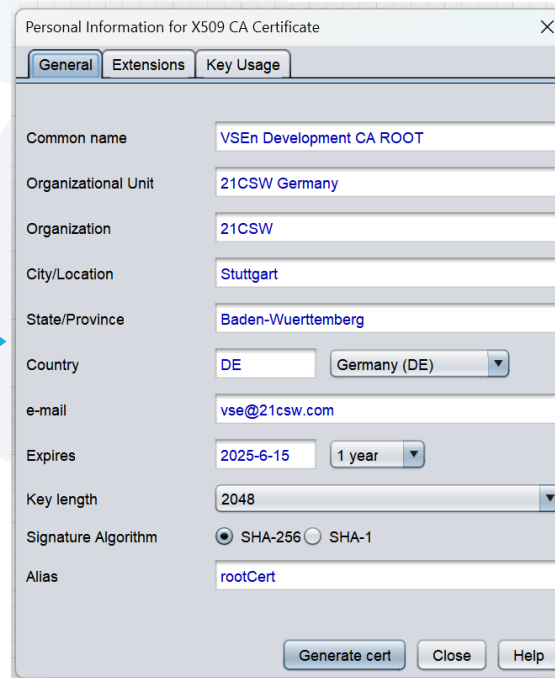
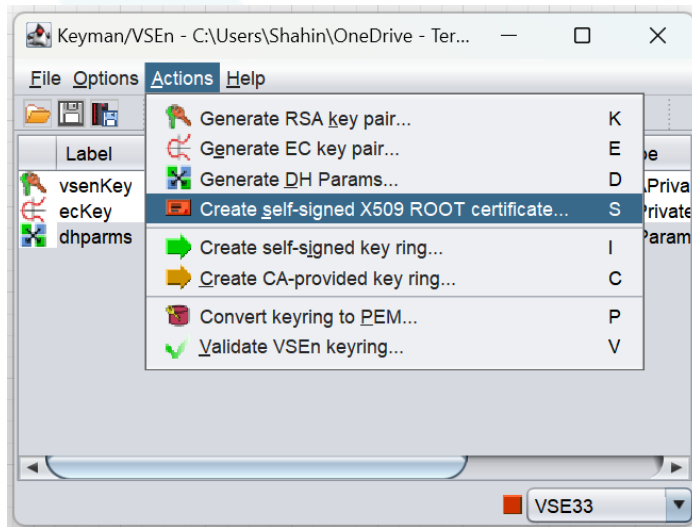


# Generate DH Parameters

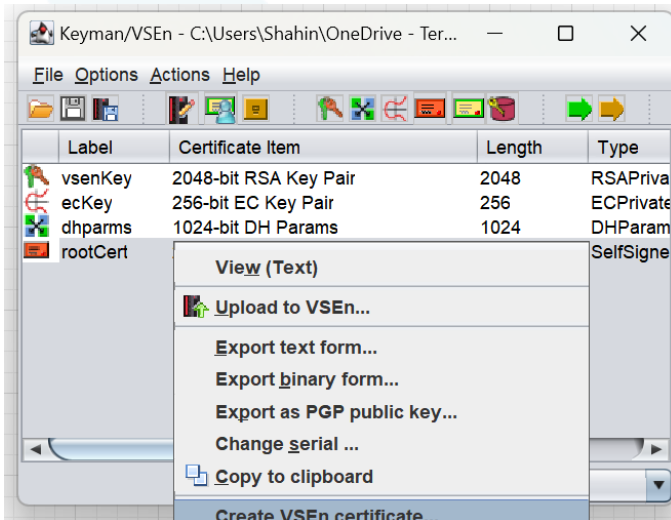




# Generate Self Signed Root Certificate



# Generate Server Certificate



Label	Certificate Item	Length	Type
vsenKey	2048-bit RSA Key Pair	2048	RSAPriva
ecKey	256-bit EC Key Pair	256	ECPrivate
dhparms	1024-bit DH Params	1024	DHParam
rootCert			SelfSigne

- View (Text)
- Upload to VSEn...
- Export text form...
- Export binary form...
- Export as PGP public key...
- Change serial ...
- Copy to clipboard
- Create VSEn certificate...



Personal information for X509 certificate

General

Common name: 192.168.22.133

Organizational Unit: Development

Organization: 21CSW

City/Location: Stuttgart

State/Province: Baden-Wuerttemberg

Country: DE (Germany (DE))

e-mail: vse@21csw.com

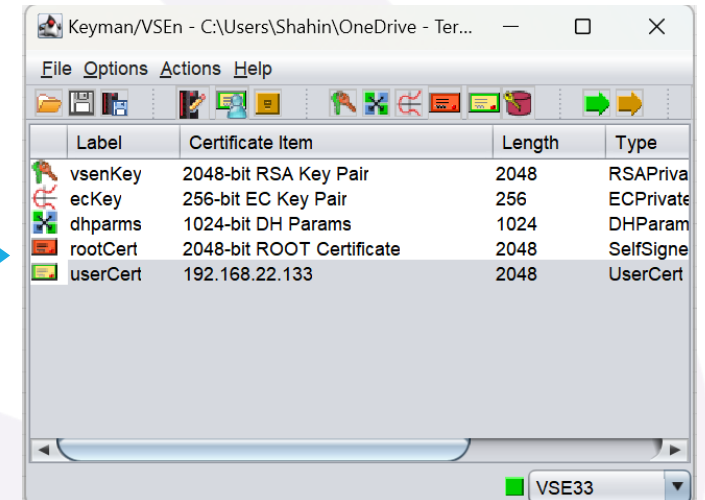
Key length: Taken from local RSA key pair

Signature Algorithm:  SHA-256  SHA-1

Notes:

In some cases (e.g. secure Telnet with the Attachmate emulator) it is required that the IP address of your VSEn system is specified as the Common Name (currently: 192.168.22.133).

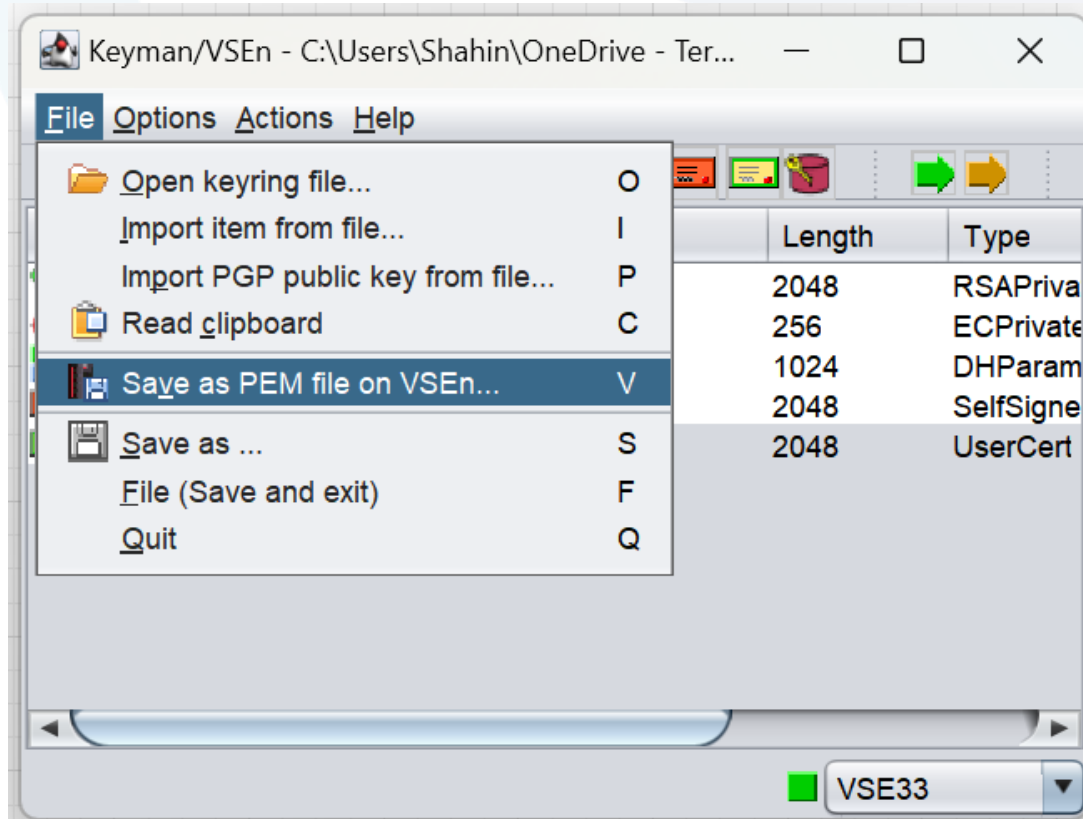
Buttons: Generate cert request, Close, View output, Help



Label	Certificate Item	Length	Type
vsenKey	2048-bit RSA Key Pair	2048	RSAPriva
ecKey	256-bit EC Key Pair	256	ECPrivate
dhparms	1024-bit DH Params	1024	DHParam
rootCert	2048-bit ROOT Certificate	2048	SelfSigne
userCert	192.168.22.133	2048	UserCert

Buttons: VSE33

# Uploading Certificate to VSE<sup>n</sup>



- Use File->Save As to save the generated certificates in PEM format
- OpenSSL supports only Privacy Enhanced Mail (PEM) format certificates

# OpenSSL & VSE<sup>n</sup> TCP/IP Stacks

IPV6 for  
VSE<sup>n</sup>

- Uses OpenSSL as SSL Engine

TCP/IP for  
VSE<sup>n</sup>

- By default, uses builtin SSL Engine (**CSINTSSL**); supports up to TLSV1.2
- To switch to OpenSSL
  - Configure EDTCPMC to use IJBSSLE as SSLPHASE
  - Include **SET OPENSSL ON** in IPINIT
  - Alternatively:  
use **// SETPARM CRYSRV='OPENSSL'**

# SETPARMs for SSL control

```
// SETPARAM SSL$MIN=' [TLSV1 | TLSV1.2 | TLSV1.3]'
```

```
// SETPARAM SSL$MAX=' [TLSV1 | TLSv1.2 | TLSV1.3]'
```

```
// SETPARAM SSL$CPH=' C027C014C013C012'
```

```
// SETPARAM SSL$DBG=' [YES|NO]
```

```
// SETPARAM BPX$GSK=IJBGSKOS & ENCRYPTION=STRONG in DFHSIT
```

```
// SETPARAM SSL$TRC='DD:lib.sublib(membername.membertype)'
```

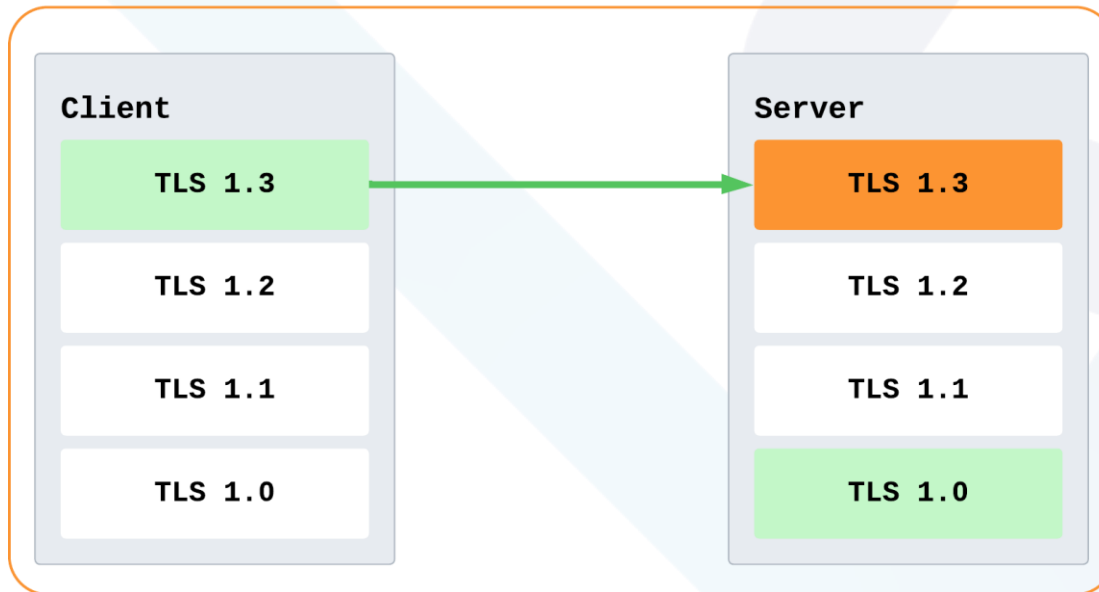
```
// SETPARAM SSL$ICA=' [YES|NO]'
```

```
// SETPARAM SSL$CSI=' [YES|NO]'
```

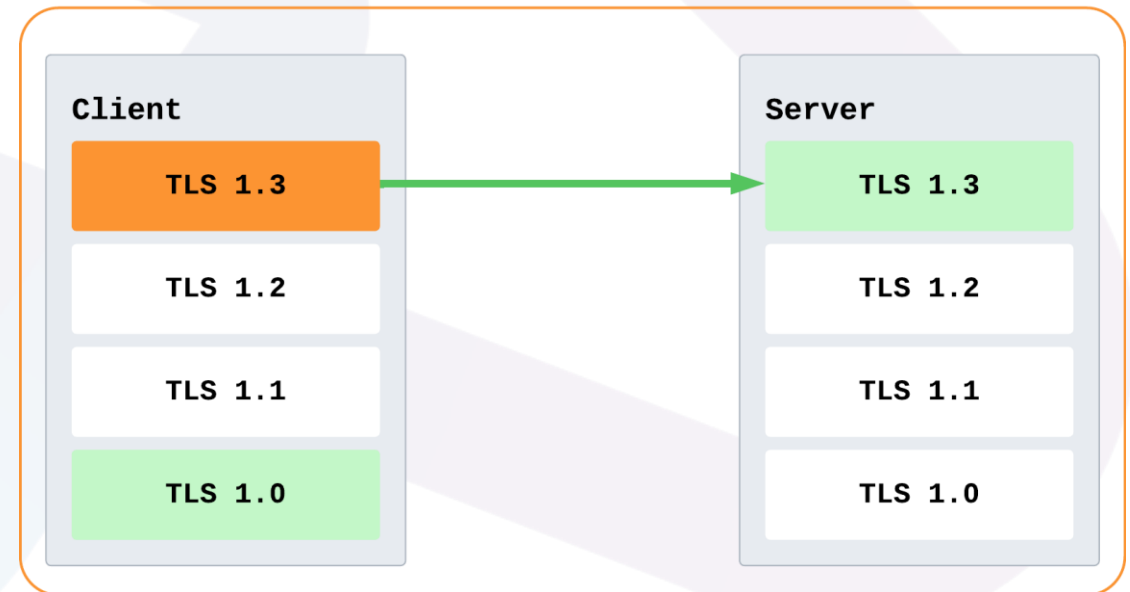
# TLS Version Control

- Specifying a TLS version means – “This version or Higher”

TLS Version Selection



TLS Version Selection



■ Version specified in configuration
 ■ Version selected for communication

# TLS Version Control

- Use SETPARAM SSL\$MIN & SSL\$MAX to control the minimum and maximum version supported dynamically.
  - E.g.,  
You can have VCS with **SSLVERSION = TLSV1.1.** and in VCS startup job

```
// SETPARAM SSL$MIN='TLSV1.3'
```

```
// SETPARAM SSL$MAX='TLSV1.3'
```

This will force the VCS to use TLSV1.3

# TLS Version Control

```
09-06-2024 09:44:56 -----  
09-06-2024 09:44:56 Summary:  
09-06-2024 09:44:56 OpenSSL version = OpenSSL 1.1.1t 7 Feb 2023  
09-06-2024 09:44:56 Compiler = IBM z/OS XL C/C++ METAL  
09-06-2024 09:44:56 sec_types = [TLSV1.2]  
09-06-2024 09:44:56 override min = [TLSV1.3]  
09-06-2024 09:44:56 override max = [TLSV1.3]  
09-06-2024 09:44:56 keyring = [CRYPTO.KEYRING]  
09-06-2024 09:44:56 keyring_pw = []  
09-06-2024 09:44:56 keyring_stash = []  
09-06-2024 09:44:56 V2_session_timeout = 0  
09-06-2024 09:44:56 V3_session_timeout = 86400  
09-06-2024 09:44:56 LDAP_server = []  
09-06-2024 09:44:56 LDAP_port = 0  
09-06-2024 09:44:56 LDAP_user = []  
09-06-2024 09:44:56 LDAP_password = []  
09-06-2024 09:44:56 LDAP_CA_roots = 0  
09-06-2024 09:44:56 auth_type = 0  
09-06-2024 09:44:56 &gsk_init_data = 0x88406378  
09-06-2024 09:44:56 -----
```

- If you have set `SSL$DBG=YES`, then you get to see the version override information in the Handshake summary.



# Recommendations

## Symmetric Encryption

- Advanced Encryption Standard(AES) - 128/256 bits

## Asymmetric Encryption

- RSA(Rivest, Shamir, Adleman) - 2048 bits
- ECC (Elliptic Curve Cryptography) - with RSA, 256-bit curve

## Hash Algorithms

- SHA-2 - 256 bits

## SSL/TLS Protocol Versions

- TLS 1.3 (If available)
- TLS 1.2
- TLS 1.0/1.1

# Thank You



