

Solution for certificate problem:

make menuconfig

1. Enter Cryptographic API.

```
.config - Linux/x86 5.15.10 Kernel Configuration
Linux/x86 5.15.10 Kernel Configuration
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----).
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes
features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in
[ ] excluded <M> module < > module capable

    General setup --->
    [*] 64-bit kernel
        Processor type and features --->
            Power management and ACPI options --->
                Bus options (PCI etc.) --->
                    Binary Emulations --->
    [*] Virtualization --->
        General architecture-dependent options --->
    [*] Enable loadable module support --->
    [*] Enable the block layer --->
        IO Schedulers --->
        Executable file formats --->
        Memory Management options --->
    [*] Networking support --->
        Device Drivers --->
        File systems --->
        Security options --->
    -*- Cryptographic API --->
        Library routines --->
        Kernel hacking --->

<Select>  < Exit >  < Help >  < Save >  < Load >
```

2. Scroll to the bottom of the page and enter Certificates for signature checking.

```
.config - Linux/x86 5.15.10 Kernel Configuration
+ Cryptographic API
    Cryptographic API
    Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----).
    Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes
    features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in
    [ ] excluded <M> module < > module capable
    *(-)
        <M> L24 compression algorithm
        <M> L24HC compression algorithm
        {M} Zstd compression algorithm
            *** Random Number Generation ***
        <M> Pseudo Random Number Generation for Cryptographic modules
    -*-
        NIST SP800-90A DRBG --->
        Jitterentropy Non-Deterministic Random Number Generator
        User-space interface for hash algorithms
        User-space interface for symmetric key cipher algorithms
        User-space interface for random number generator algorithms
        [ ]   Enable CAVP testing of DRBG
        <M> User-space interface for AEAD cipher algorithms
    [*]   Enable obsolete cryptographic algorithms for userspace
    [*]   Crypto usage statistics for User-space
            *** Crypto library routines ***
        < > BLAKE2s hash function library
        < > ChaCha library interface
        < > Curve25519 scalar multiplication library
        < > Poly1305 library interface
        < > ChaCha20-Poly1305 AEAD support (8-byte nonce library version)
    [*]   Hardware crypto devices --->
    -*-
        Asymmetric (public-key cryptographic) key type --->
            Certificates for signature checking --->

<Select>  < Exit >  < Help >  < Save >  < Load >
```

3. Enter these two options and empty their contents.

```
.config - Linux/x86 5.15.10 Kernel Configuration
+ Cryptographic API + Certificates for signature checking
    Certificates for signature checking
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----).
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes
features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in
[ ] excluded <M> module < > module capable

(certs/signing_key.pem) File name or PKCS#11 URI of module signing key
    Type of module signing key to be generated (RSA) --->
-*= Provide system-wide ring of trusted keys
    () Additional X.509 keys for default system keyring
    [*] Reserve area for inserting a certificate without recompiling
    (4096) Number of bytes to reserve for the extra certificate
    [*] Provide a keyring to which extra trustable keys may be added
    [*] Provide system-wide ring of blacklisted keys
    () Hashes to be preloaded into the system blacklist keyring
    [*] Provide system-wide ring of revocation certificates
    () X.509 certificates to be preloaded into the system blacklist keyring

<Select> < Exit > < Help > < Save > < Load >
```

```
.config - Linux/x86 5.15.10 Kernel Configuration
+ Cryptographic API + Certificates for signature checking
    Certificates for signature checking
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----).
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes
features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in
[ ] excluded <M> module < > module capable

(certs/signing_key.pem) File name or PKCS#11 URI of module signing key
    Type of module signing key to be generated (RSA) --->
-*= Provide system-wide ring of trusted keys
    () Additional X.509 keys for default system keyring
    [*] Reserve area for inserting a certificate without recompiling
    (4096) Number of bytes to reserve for the extra certificate
    [*] Provide a keyring to which extra trustable keys may be added
    [*] Provide system-wide ring of blacklisted keys
    () Hashes to be preloaded into the system blacklist keyring
    [*] Provide system-wide ring of revocation certificates
    () X.509 certificates to be preloaded into the system blacklist keyring

<Select> < Exit > < Help > < Save > < Load >
```

4. Continue on compile-kernel-instructions