# Code Runner (Optimal)

| OS Vendor Server | Developer Server | Normal world | Secure world (SW) |
|---|---|---|---|

**OS Vendor Server:**
CertPrivVendor(PubVerifySW)
PrivVendor, PupVendor
PrivSignVendor, PubVerifyVendor
CertPrivCA(PubVerifyDev)

**Developer Server:**
secret,
PubVerifyVendor,
PrivSignDev,
a2 = app signature

**Normal world:**
PubDev
PubVerifyDev

**Secure world (SW):**
CertPrivVendor(PubVerifySW),
PrivSignSW,
PupVendor, PubVerifyVendor

① ← GetNonce() —

generate $N_R \in_R \{0,1\}^{256}$

**2**

— $N_R$ → — requestAttestation($N_R$) →

**1**

② generate PubA, PrivA
check system integrity
a1 = read app signature
statement = IMEI+ system integrity + $N_R$ +pubA +a1
σVendor = $Sign_{PrivSignSW}$(statement)
blob = $Enc_{PubVendor}$(statement+σVendor)

← blob — ← blob — ← blob —

**4** **3**

③ Statement , σVendor = $Dec_{PrivVendor}$(blob)
check IMEI, system integrity, statement
verdict = (checks)? valid : invalid
statementDev = statement + verdict
σDev = $Sign_{PrivSignVendor}$(statement)
~~blob = $Enc_{PubDev}$(statement+o)~~

— statmentDev + σDev →

**5**

④ Verify σDev, statementDev
check verdict, $N_R$, (a1 == a2)
c = $Enc_{PupA}$(secret + a2)
o = $Sign_{PrivSignDevServer}$ (c)

**5**

— c, o → ⑤ Verfiy o

— c →

⑥ check system integrity
If integrity is valid ,
    Secret, a2 = $Dec_{privA}$(c)
    a1 = read app signature
    Check (a1 == a2)

a1 as auth ← Secret — Error

Delete pupA, privA