RA-TLS and its Variants

Muhammad Usama Sardar[1], Arto Niemi[2], Hannes Tschofenig[3] and Thomas Fossati[4]

[1]TU Dresden, Germany

[2]Huawei Technologies, Helsinki, Finland

[3]University of Applied Sciences Bonn-Rhein-Sieg and Siemens, Germany

[4]Linaro, Lausanne, Switzerland

October 22, 2024

CPEC CENTER FOR PERSPICUOUS COMPUTING

RA-TLS and its Variants

1. Intel's RA-TLS[1]
2. CCC Attestation SIG's Interoperable RA-TLS[2]
3. Intel's new design[3] (RA-TLS + Attested CSR)

Discussion

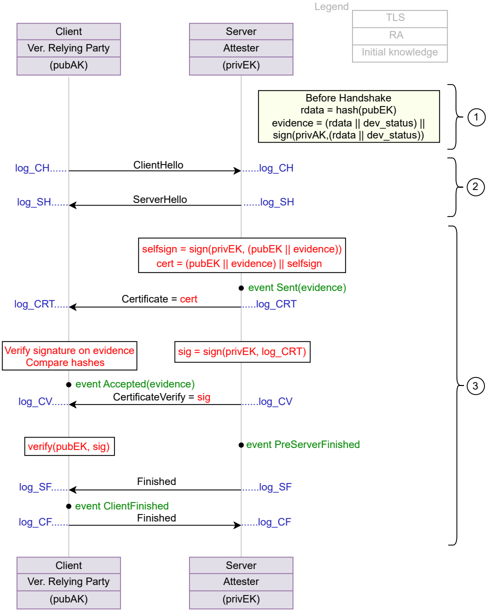
i. Does Intel have specs of any of those internally?
ii. "per-session evidence freshness" was one of the objectives of #2. Any update?
iii. Technical difference between #2 and #1?
iv. Identity of server (PKIX/CA-signed certs) alongside attestation?

[1] https://arxiv.org/pdf/1801.05863

[2] https://github.com/CCC-Attestation/interoperable-ra-tls/

[3] https://github.com/CCC-Attestation/interoperable-ra-tls/issues/14#issuecomment-2407903545

Intel's RA-TLS



Legend

TLS

RA

Initial knowledge

Client

Ver. Relying Party

(pubAK)

Server

Attester

(privEK)

Before Handshake
rdata = hash(pubEK)
evidence = (rdata || dev_status) ||
sign(privAK,(rdata || dev_status))

1

log_CH...... ClientHello ......log_CH

log_SH...... ServerHello ......log_SH

2

selfsign = sign(privEK, (pubEK || evidence))
cert = (pubEK || evidence) || selfsign

● event Sent(evidence)

log_CRT..... Certificate = cert .....log_CRT

Verify signature on evidence
Compare hashes

sig = sign(privEK, log_CRT)

● event Accepted(evidence)

log_CV...... CertificateVerify = sig ......log_CV

verify(pubEK, sig)

● event PreServerFinished

log_SF...... Finished ......log_SF

● event ClientFinished

log_CF...... Finished ......log_CF

3

Client

Ver. Relying Party

(pubAK)

Server

Attester

(privEK)