ORACLE

Remote Attestation Procedure Daemon(RATSd)

Evidence Collector

Ian Chin (Tom) Wang, Jag Raman

Oracle

Sep 9th, 2025

Safe harbor statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.

Agenda

- Motivation
- System Design
- RATS conceptual message collection Daemon (RATSd)
- Type of evidence and leaf attesters
- Configfs-TSM
- Evidence Formats
- Current Status
- Challenges

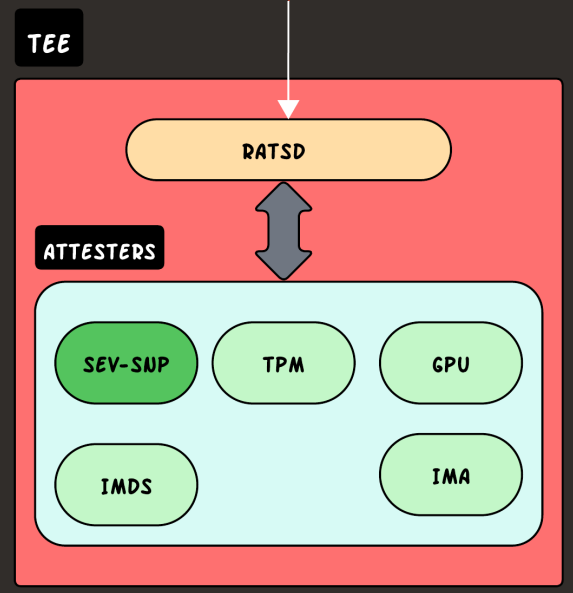
Motivation

- **System Composition:** Systems often include multiple attesters
- **Hardware Root of Trust (RoT):**
  - Remote Attestation requires it
  - Manufacturers provide HW modules acting as RoTs
- **Multiple manufacturers:**
  - Systems use compute hardware from various manufacturers, each with its own RoT
  - Each RoT addresses different parts of the Trusted Computing Base (TCB)
- **Evidence Collection:** Need a mechanism to collect evidence from all attesters
- **Compatibility:** Tool must adhere to data formats in the RATS architecture for integration with existing tools
- **Gap in existing tools:** No existing tools met these requirements, leading to the development of RATSd
- **Multi-Verifier:** https://datatracker.ietf.org/doc/draft-deshpande-rats-multi-verifier/

System Design

In: Nonce
Out: Attestation Token (EAT)

Relying Party



In: Attestation Token (EAT)
Out: Attestation Result (EAR)

Verifier (Veraison Services)

RATS conceptual message collection Daemon (RATSd)

- Provides the combination of evidence from each leaf attester for a system
- Implements leaf attesters as plugins
- Provides uniformed APIs to retrieve an attestation token
- Plugin Verification
  - For now, ratsd core (lead attester) and composite attester are assumed to be distributed by a trusted package registry
- Complex queries
  - Relying Party can pick what goes into the token
  - Plugin-specific query parameters

Leaf Attesters

- TSM (Trusted Secure Module) Report
  - Should work for both SEV-SNP and TDX
  - Attestation Report from the Secure Processor (OutBlob)
  - X509 Certificates (Auxblob)
- TPM
  - Requires SVSM
  - A TPM quote from TPM2 Tools returns PCR values with a signature
- GPU / TPU
- Integrity Measurement Architecture (IMA)

ConfigFS TSM

- Vendors have proprietary attestation report formats
  - SEV-SNP introduced the chardev, accessible via ioctl()
- Configfs is a filesystem-based manager of kernel objects, or config_items
  - Adopted as the cross-vendor mechanism to retrieve CoCo attestation report start v6.7
- Usage

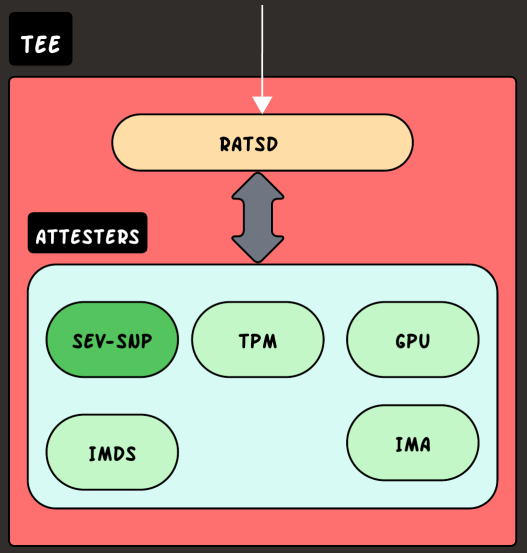
Tsm-report

- OutBlob
  - Binary Attestation report
  - Generated based on inblob
- Auxblob
  - Optional auxiliary data
  - Cert-table in SEV-SNP
- Service Report
  - Provider info (sev-guest, tdx-guest)
  - Manifest Blob

```
tsm-report = {
  ? auxblob: binary-string
  outblob: binary-string
  provider: tstr
  ? service-report
}

service-report = ((
  manifestblob: binary-string
  service_provider: tstr
) // service_provider: tstr)

binary-string = base64url-string .feature "json" /
bstr .feature "cbor"

base64url-string = tstr .b64u bstr
```

Evidence Formats

- Conceptual Message Wrapper (CMW) collection
- EAT envelope
- Registered TSM report media-type with IANA: "application/vnd.veraison.tsm-report+cbor"
- RATSd: https://github.com/veraison/ratsd



CMW collection

```
{
    "__cmwc_t": "application/
vnd.oracle.VMStandardE5Flex",
    "sevsnp": [
        {
            "type": "application/vnd.veraison.tsm-
report+cbor",
            "value": "<< tsm-report >>"
        }
    ],
    "tpm": [
        {
            "type": "application/vnd.tcg.tpm",
            "value": "<< TPMS_ATTEST >>"
        }
    ]
}
```

IANA recognizes "application/vnd.veraison.configfs-tsm+json".

Yet to register "application/vnd.oracle.VMStandardE5Flex" & "application/vnd.tcg.tpm

RATSD Response

```
[root@nsh-x10m-1 ~]# curl -X POST http://localhost:8895/ratsd/chares -H
"Content-type: application/vnd.veraison.chares+json" -d '{"nonce":
"TUlEQk5IMjhpaW9pc2pQeXh4eHh4eHh4eHh4eHhNSURCTkgyOGlpb2lzalB5eHh4eHh4eHh4
eHh4eHh4eA"}'
{"cmw":"eyJfX2Ntd2NfdCI6InRhZzpnaXRodWIuY29tLDIwMjU6dmVyYWlzb24vcmF0c2QvY213I
iwibW9jay10c20iOlsiYXBwbGljYXRpb24vdm5kLnZlcmFpc29uLmNvbmZpZ2ZzLXRzbStqc29uIi
wiZXlKaGRYaGliRzlpSWpvaVdWaFdORmx0ZUhaWlp5SXNJbTkxZEdKc2IySWlPaUpqU0Vwd1pHMTR
iR1J0Vm5OUGFVRjNRMjFzZFZsdGVIWlphbTluVGtkUk1FOVVVVEJPUkVrd1dsUlJORTE2U1hwUFJG
azFUbXByTWxwcVdUVk9lazB5V1ZSVmQwNTZhek5QUkdNMFRucG5NMDlFWXpST2VtY3pUMFJqTkU1N
lp6TlBSR00wVG5wbk0wOUVZelJPZW1jelQwUlNhMDVFYXpCT1JGRjVUa2RWTUU5RVRYbE5lbWN5VD
FSWk5VNXRXVEpQVkdONlRtMUZNVTFFWXpWT2VtY3pUMFJqTkU1Nlp6TlBSR00wVG5wbk0wOUVZelJ
PZW1jelQwUmpORTU2WnpOUFJHTTBUbnBuSWl3aWNISnZkbWxrWlhJaU9pSm1ZV3RsWEc0aWZRIl19
","eat_nonce":"TUlEQk5IMjhpaW9pc2pQeXh4eHh4eHh4eHh4eHhNSURCTkgyOGlpb2lzal
B5eHh4eHh4eHh4eHh4eA","eat_profile":"tag:github.com,2024:veraison/ratsd"}
```

Sample CMW Collection


```
{"__cmwc_t":"tag:github.com,2025:veraison/ratsd/cmw","tsm":
["application/vnd.veraison.configfs-
tsm+json","eyJhdXhibG9iIjoiWVhWNFlteHZZZyIsIm91dGJsb2IiOiJjSEpwZG14bGR
tVnNPaUF3Q21sdVlteHZZam9nTkdRME9UUTBOREkwWlRRNE16SXpPRFk1TmprMlpqWTVOe
k0yWVRVd056azNPRGM0TnpnM09EYzROemczT0RjNE56ZzNPRGM0TnpnM09EYzROemczT0R
Sa05EazBORFF5TkdVME9ETXlNemcyT1RZNU5tWTJPVGN6Tm1FMU1EYzVOemczT0RjNE56Z
zNPRGM0TnpnM09EYzROemczT0RjNE56ZzNPRGM0TnpnIiwicHJvdmlkZXIiOiJmYWtlXG4
ifQ"]}
```

Current upstream status of Ratsd

- ✓Lead attester / Ratsd Core
- ✓Configfs TSM leaf-attester
- ✓Leaf-attester query options
- ✓Leaf-attester selections
- ✓Cryptographically-verified leaf-attester
- ❑More leaf attesters (TPM quotes, IMA, etc.)
- ❑Signed CMW collection

Sample CMW Collection


```
{"__cmwc_t":"tag:github.com,2025:veraison/ratsd/cmw","tsm":
["application/vnd.veraison.configfs-
tsm+json","eyJhdXhibG9iIjoiWVhWNFlteHZZZyIsIm91dGJsb2IiOiJjSEpwZG14bGR
tVnNPaUF3Q21sdVlteHZZam9nTkdRME9UUTBOREkwWlRRNE16SXpPRFk1TmprMlpqWTVOe
k0yWVRVd056azNPRGM0TnpnM09EYzROemczT0RjNE56ZzNPRGM0TnpnM09EYzROemczT0R
Sa05EazBORFF5TkdVME9ETXlNemcyT1RZNU5tWTJPVGN6Tm1FMU1EYzVOemczT0RjNE56Z
zNPRGM0TnpnM09EYzROemczT0RjNE56ZzNPRGM0TnpnIiwicHJvdmlkZXIiOiJmYWtlXG4
ifQ"]}
```

Challenges

- Establishing trust in the lead attester
- Interaction between Veraison service and Veraison RATSd
  - The upstream Veraison service does not handle composite evidence
- Workload attestation
  - Current implementation of RATSd supports only single-VM
- CSR to obtain signing key

References

- Veraison main repo
  - https://github.com/veraison
- RATSd Repo
  - https://github.com/veraison/ratsd
- Veraison-service (Verifier)
  - https://github.com/veraison/services
  - SEV-SNP: https://github.com/veraison/corim/pull/167
- Driving upstream discussion
  - https://docs.google.com/document/d/1YfAatMWj6D1xxYncw4Kh64Mc3wbDNG5W0J8yjZ_QtjA/edit?tab=t.0

Thank you