Confidential Computing in Computing Aware Network
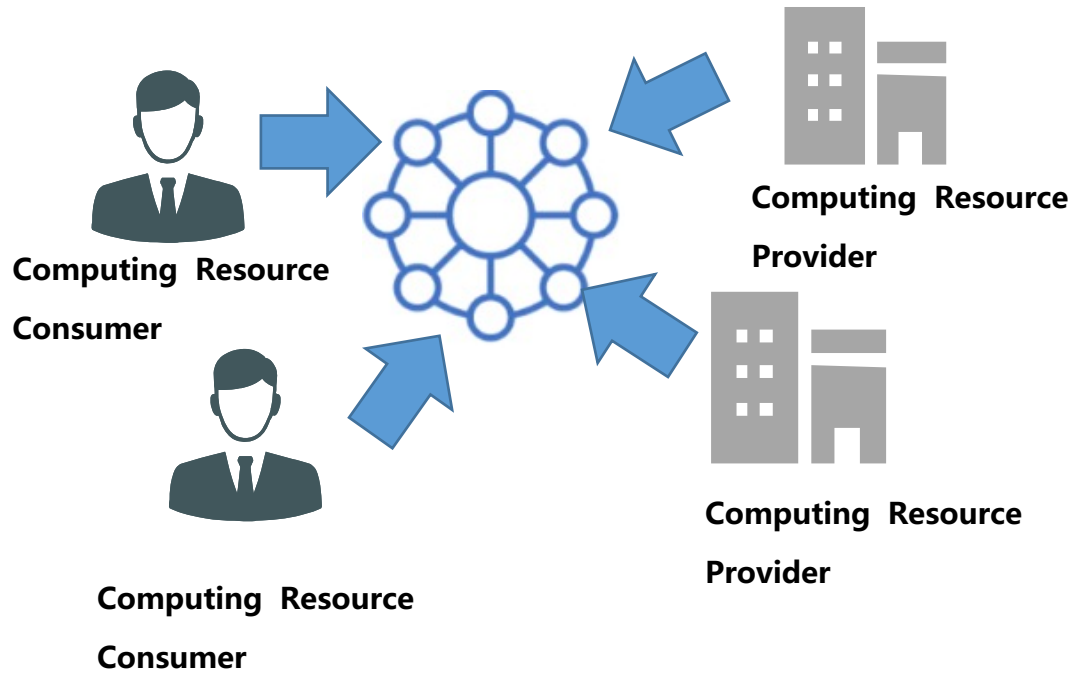
What is CAN

CFN (Computing Force Network)
- ITU-T-SG13
- Draft new Recommendation ITU-T Y.IMT2020-CNC-req: "Requirements of computing and network convergence for IMT-2020 and beyond "
- etc.

CPN (Computing Power Network)
- ITU-T-SG13
- Computing power network – Framework and architecture
- etc.

CAN (Computing Aware Network)
- IETF BOF, Mailing List: dyncast@ietf.org, https://www.ietf.org/mailman/listinfo/dyncast
- o draft-liu-dyncast-ps-usecases-01
- etc.

Computing-aware Networking (CAN), which is computing and network resource joint optimization based on the awareness, control and management over network and computing resources, to determine the appropriate service node，dispatch the service request and provide a better user experience.

What is CAN

Brief Architecture of CAN

Computing Resource Consumer

Computing Resource Provider

Computing Resource Consumer

Computing Resource Provider

Logic Layers of CAN

Operation Layer

| Computing Resource Transaction | Unified Operation | Open Capability | etc. |

Orchestration/Management Layer

| Computing Resource Orchestration | AI Engine | Base Mgmt | etc. |

CAN Infrastructure

Edge Computing Resource

Central Computing Resource

Edge Computing Resource

IP Network

Edge Computing Resource
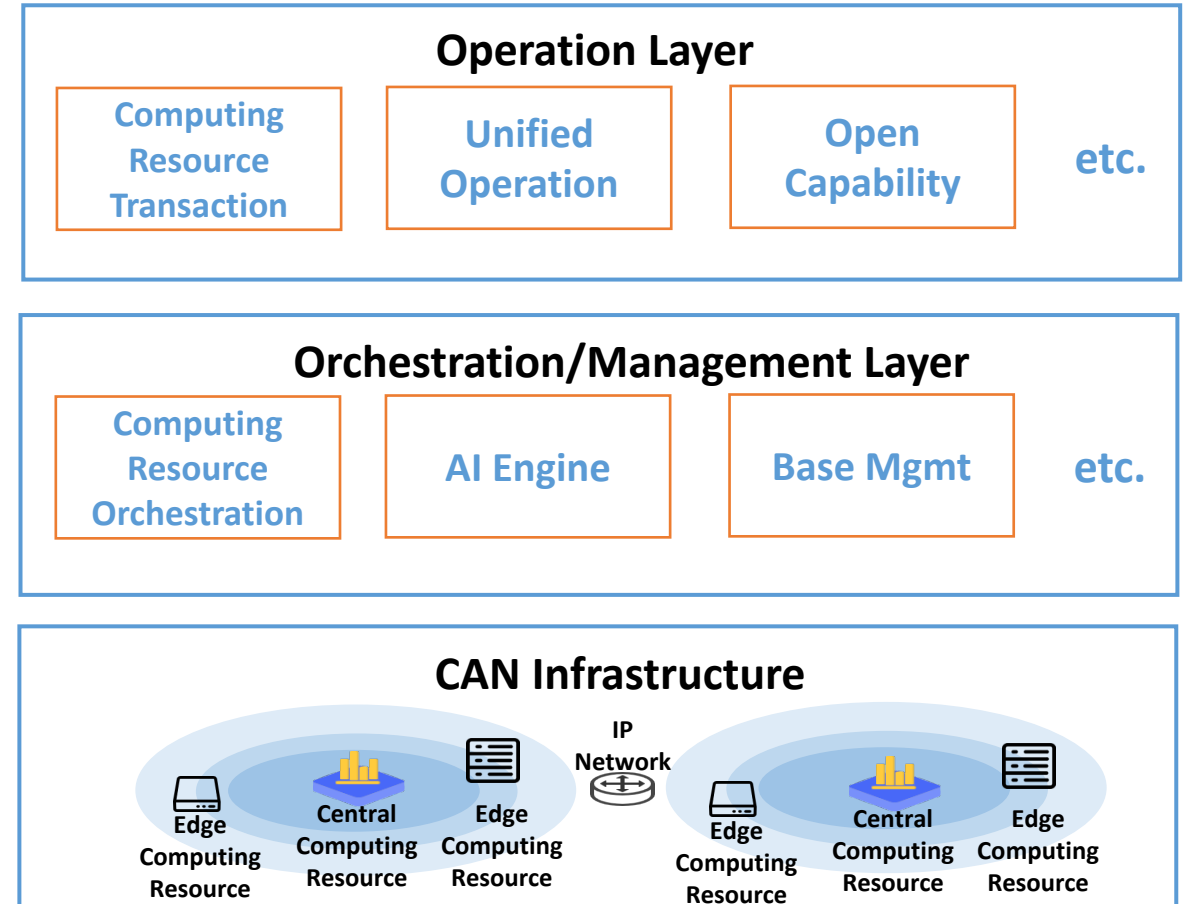
Central Computing Resource

Edge Computing Resource

Difference between CAN and Cloud Computing:

CAN is based on network layer, computing task requirement and scheduling are carried by protocols in IP network. From the perspective of network layer, cloud computing is just application layer data which cannot be aware by network.

Advantage: Network latency awareness; Network topology awareness; Convergence with network resource and computing resource
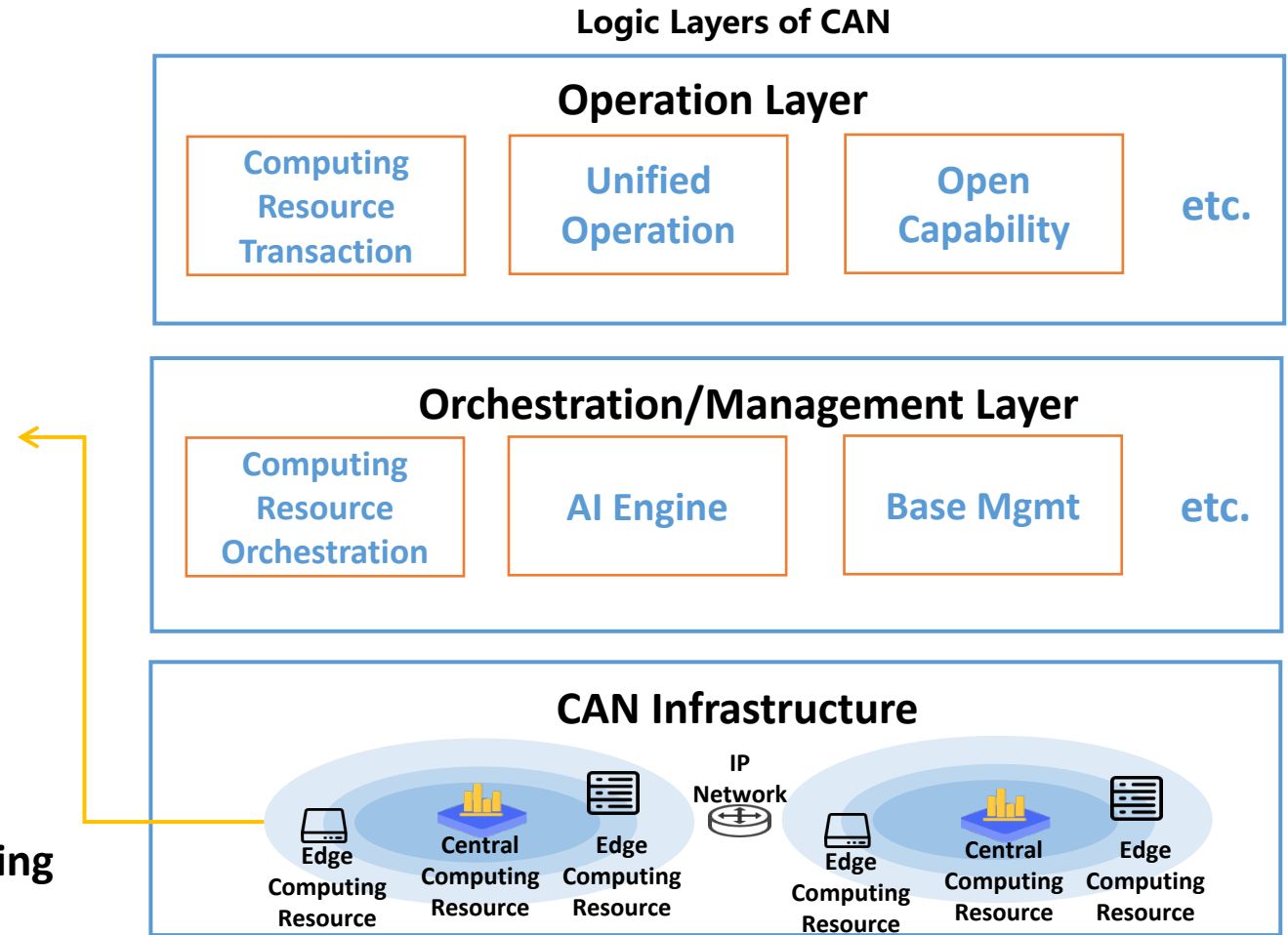
Confidential Computing in CAN

Requirements for Confidential Computing

➢ **If the computing resource is a third party source like the edge CR, which both the CAN and the network user cannot trust its security, then the Confidential Computing technique is needed.**

➢**If the network user cannot trust the computing resource controlled by CAN like Central CR, then the Confidential Computing technique is needed.**

Two components in CAN are mostly involved when using confidential computing :

✅ CPU of Computing Resource with confidential computing feature, like SGX, TDX, SEV, CCA, etc.

✅ O/M with confidential computing feature, like TEEP and RATs.

Logic Layers of CAN

Operation Layer

| Computing Resource Transaction | Unified Operation | Open Capability | etc. |

Orchestration/Management Layer

| Computing Resource Orchestration | AI Engine | Base Mgmt | etc. |

CAN Infrastructure

IP Network

Edge Computing Resource Central Computing Resource Edge Computing Resource Edge Computing Resource Central Computing Resource Edge Computing Resource
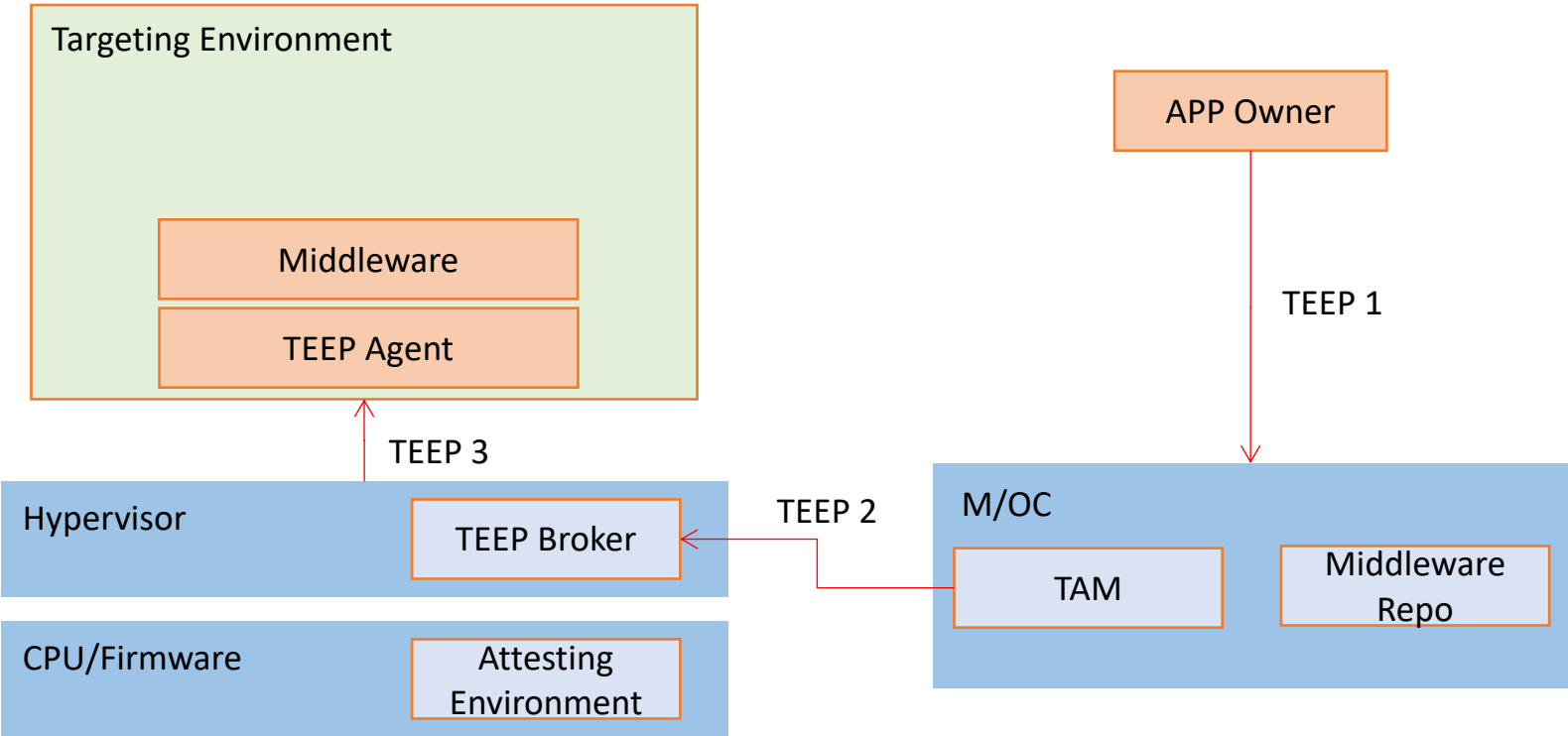
Confidential Computing in CAN
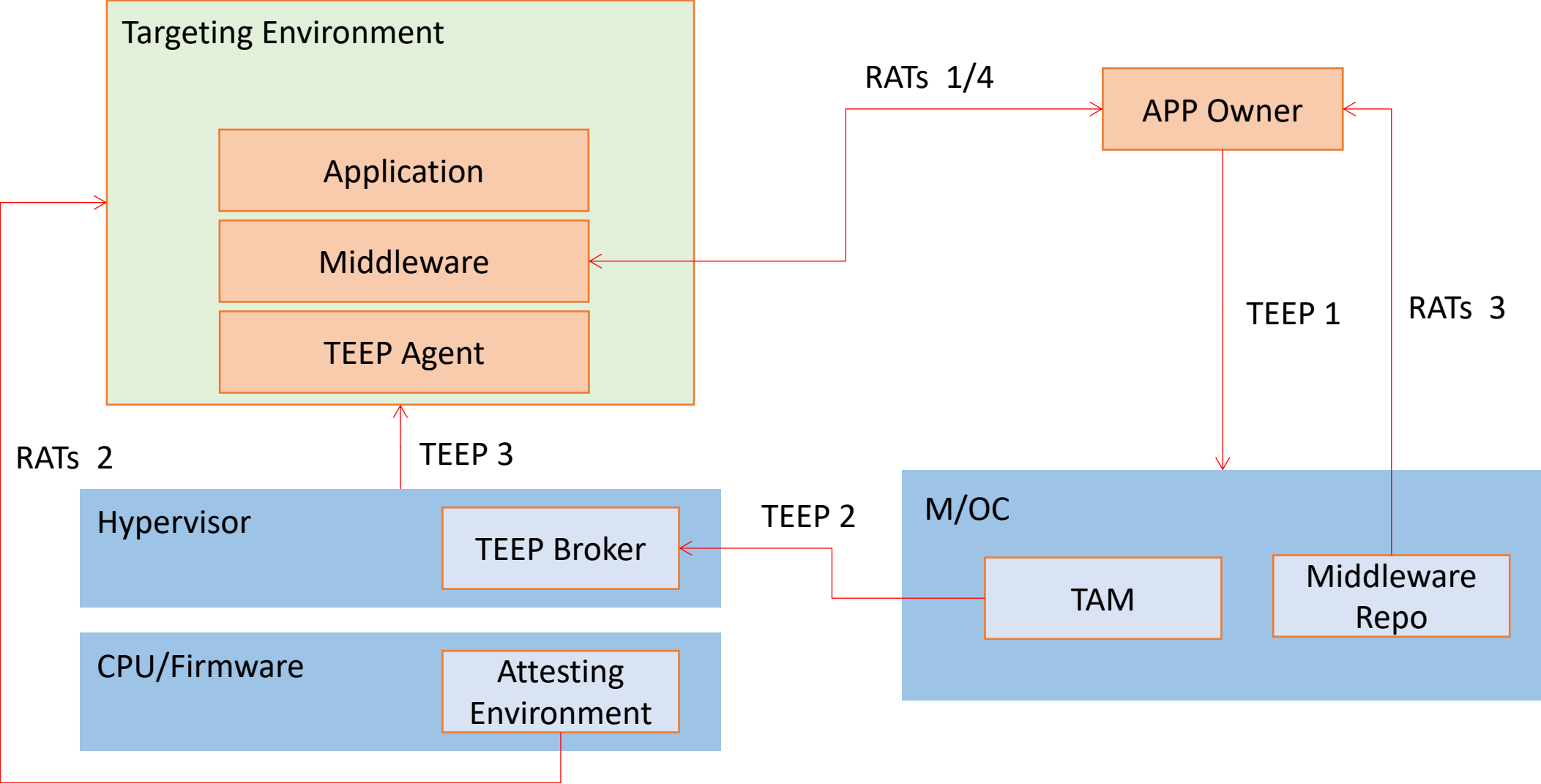
Tow components are added in this architecture: Middleware and Middleware Repository. Middleware has two functions: decouple heterogeneous CPUs in different computing resource, provide remote attestation.

TEEP in CAN

Targeting Environment
- Middleware
- TEEP Agent

Hypervisor
- TEEP Broker

CPU/Firmware
- Attesting Environment

APP Owner

TEEP 1

TEEP 2

TEEP 3

M/OC
- TAM
- Middleware Repo

➢Step 1: App Owner request for CC resource.

➢Step 2: TAM establishes connection to TEEP Broker and send the provisioning information like the Middleware to TEEP Broker.

➢Step3: TEEP Broker triggers the Hypervisor to establish the Targeting Environment with TEEP Agent and Middleware.

Mapping between TEEP Concept and Instantiation:

TEEP Concept	Instantiations
M/OC, TAM and Middleware Repo	Computing Resource Orchestration in CAN
TEEP Broker	Function in OpenStack
TEEP Agent	OpenStack initiates guest VM with CC access, which includes a standard TEEP Agent, and specific Middleware like Enarx, Gramine, Occlum, etc.
Middleware	

RATs in CAN
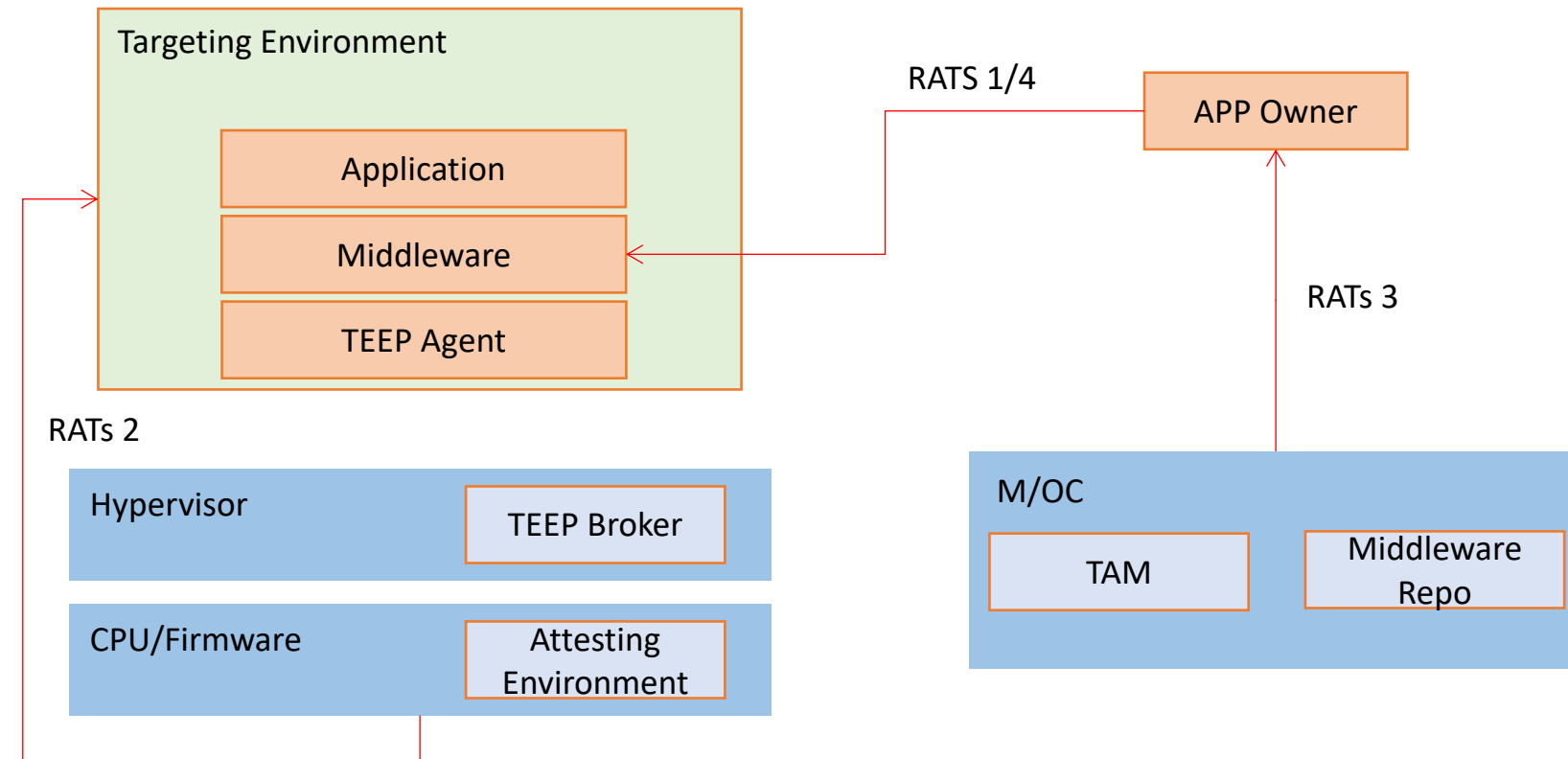
> **Step 1:** App Owner asks for remote attestation.
> **Step 2:** Middleware triggers Attesting Environment to generate remote attestation evidence for Targeting Environment.
> **Step3：** APP Owner gets the reference value, which the APP owner could generate with the source code and images, or could query from third party authority.
> **Step 4：** Middleware return the evidence to App Owner. The Owner could match it with the reference value.

Mapping between RATs Concept and Instantiation:

RATs Concept	Instantiations
Attesting Environment	TDX/SGX: Quote Enclave, SEV: SP, CCA: RMM, etc.
Targeting Environment	Guest VM, which includes TEEP, Middleware and Application

Thanks