**Azure IoT Edge Security Manager**

The key principals of the Security Manager is that it protects all of the Edge components and devices by abstracting the secure silicon hardware.

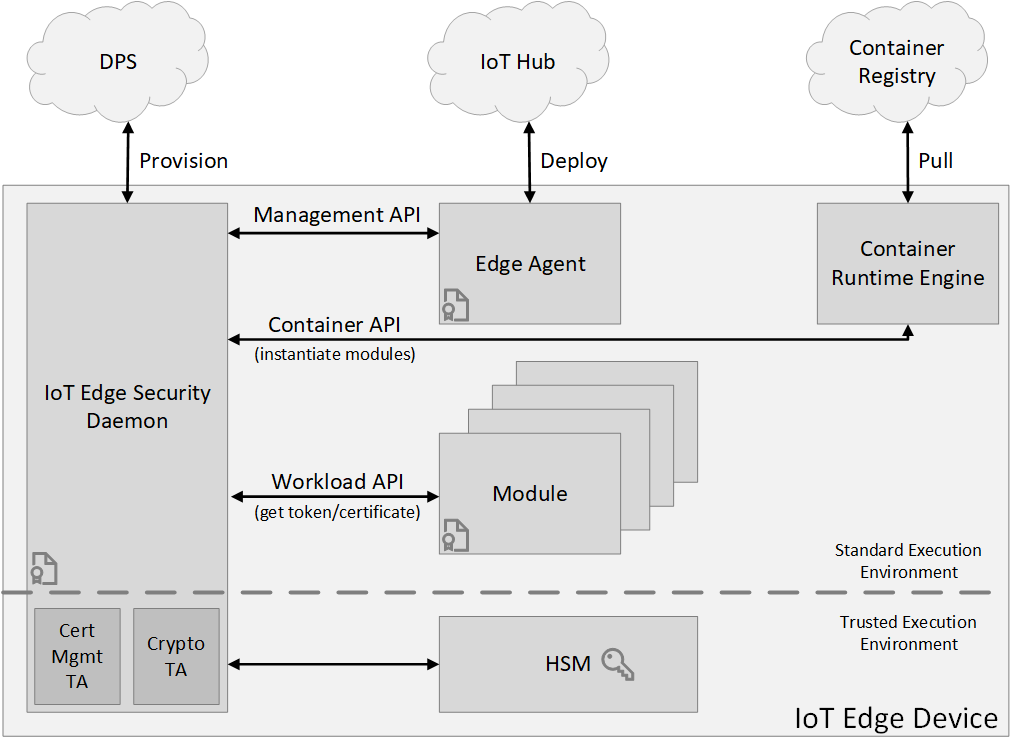
**The IoT Edge security manager is comprised of three components:**

--IoT Edge security daemon.

--Hardware security module platform abstraction Layer (HSM PAL).

--Optional but highly recommended hardware silicon root of trust or HSM.

**Architecture of IoT Edge security daemon**



**Management API**

IoT Edge security daemon offers a management API, which is called by the IoT Edge agent when creating/starting/stopping/removing an IoT Edge module. The security daemon stores “registrations” for all active modules. These registrations map a module’s identity to some properties of the module. For examples, these module properties include the process identifier (pid) of the process running in the container and the hash of the docker container’s contents.

These properties are used by the workload API (described below) to verify that the caller is authorized for an action.

The management API is a privileged API, callable only from the IoT Edge agent. Since the IoT Edge security daemon bootstraps and starts the IoT Edge agent, it verifies that the IoT Edge agent hasn't been tampered with, then it can create an implicit registration for the IoT Edge agent. The same attestation process that the workload API uses also restricts access to the management API to only the IoT Edge agent.

**Container API**

The container API interacts with the container system in use for module management, like Moby or Docker.

**Workload API**

The workload API is accessible to all modules. It provides proof of identity, either as an HSM rooted signed token or an X509 certificate, and the corresponding trust bundle to a module. The trust bundle contains CA certificates for all the other servers that the modules should trust.

The IoT Edge security daemon uses an attestation process to guard this API. When a module calls this API, the security daemon attempts to find a registration for the identity. If successful, it uses the properties of the registration to measure the module. If the result of the measurement process matches the registration, a new proof of identity is generated. The corresponding CA certificates (trust bundle) are returned to the module. The module uses this certificate to connect to IoT Hub, other modules, or start a server. When the signed token or certificate nears expiration, it's the responsibility of the module to request a new certificate.

**Hardware security module platform abstraction layer (HSM PAL)**

--The HSM PAL abstracts all root of trust hardware to isolate the developer or user of IoT Edge from their complexities. It includes a combination of application programming interface (API) and trans-domain communication procedures, for example communication between a standard execution environment and a secure enclave. The actual implementation of the HSM PAL depends on the specific secure hardware in use. Its existence enables the use of virtually any secure silicon hardware.

**Secure silicon root of trust hardware**

--Secure silicon is necessary to anchor trust inside the IoT Edge device hardware. Secure silicon come in variety to include Trusted Platform Module (TPM), embedded Secure Element (eSE), ARM TrustZone, Intel SGX, and custom secure silicon technologies. The use of secure silicon root of trust in devices is recommended given the threats associated with physical accessibility of IoT devices.

**Integration and Maintenance**

--All provisioning and services are deployed and managed via the IoT Edge Security Daemon which is accessed via the Edge Agent container image (module) that is deployed to each edge device as part of the Deployment process.

**Installation and updates**

--Installation and updates of the IoT Edge security daemon are managed through the operating system's package management system. IoT Edge devices with hardware root of trust should provide additional hardening to the integrity of the daemon by managing its lifecycle through the secure boot and updates management systems.