5.8 . Framework Architecture and Enforcement in IoT Systems

The enforcement architecture in IoT systems is crucial for managing security and controlling access to resources. The SecKit framework demonstrates a structured approach to policy enforcement in IoT environments, as outlined below.

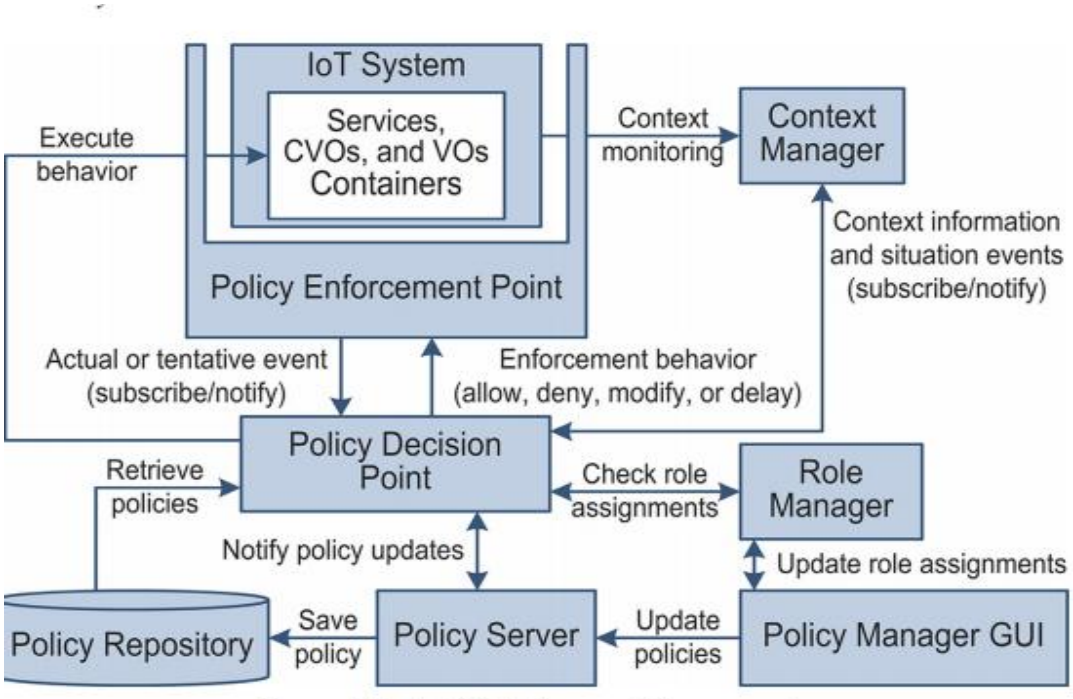

Figure 5.7 SecKit Enforcement Components

1. Enforcement Components Overview

The SecKit architecture uses specific components for enforcing policies in an IoT system:

- **Policy Enforcement Point (PEP):** Monitors and intercepts service requests, invoking events according to defined policies. The PEP sends event signals to the Policy Decision Point (PDP) and waits for enforcement actions.
- **Policy Decision Point (PDP):** Processes event signals from the PEP. It can retrieve information on Virtual Objects (VOs) and Composite Virtual Objects (CVOs) if needed, and subscribe to context information via the Context Manager.
- **Context Manager:** Manages context information and events that may influence decisions made by the PDP.

These components collaborate to ensure that actions in the IoT framework align with specified policies.

2. Technology-Specific Extensions for SecKit

SecKit's architecture is adaptable for various IoT platforms. For practical applications, it can be extended with specific runtime monitoring components:

- **MQTT Broker Extension:** In the iCore project, an extension supports policy monitoring and enforcement for MQTT (Message Queuing Telemetry Transport) brokers. MQTT is widely used for communication in IoT environments, where devices like VOs and CVOs (e.g., staff or medical equipment in a hospital) interact via message exchanges.
- **Hospital Scenario Example:** In a hospital setting, policies can be applied to control access to sensitive data such as staff location and medical device usage. The MQTT broker facilitates this by managing secure communication between VOs and CVOs according to specified access policies.

3. Rule Engine and Enforcement Actions

The SecKit rule engine provides a runtime interface to implement and enforce policies:

- **Event Interception and Notification:** An extended MQTT broker intercepts messages using a publish-subscribe mechanism, notifying the rule engine about relevant events.
- **Policy Enforcement Actions:** Based on the events, the rule engine can trigger specific actions such as "Allow," "Deny," or "Modify" to control access. These actions ensure that only authorized entities can access certain data or devices.

The rule engine thus enables dynamic enforcement of policies, enhancing the security and functionality of IoT systems in real-time contexts.

This framework provides a robust, flexible system to monitor, manage, and enforce security policies in IoT environments, adapting to the specific requirements of each technology platform and application context.