Dynamic Policy Rules for Emergency Healthcare

Sondes Baccouri and Takoua Abdellatif

2024

1 Introduction

In emergency healthcare, real-time access to critical patient data is essential. The dynamic nature of such environments requires adaptable policy rules for varying contexts, such as healthcare professionals' roles and patient conditions. The BIG-ABAC framework addresses this need through dynamic policy generation and real-time updates, leveraging distributed data processing for low latency and high responsiveness. The complete implementation can be found in the GitHub repository: `https://github.com/BaccouriSondes/BIG-ABAC`.

2 Dynamic Policy Rules for Emergency Healthcare

BIG-ABAC adjusts access control policies in real-time, based on contextual attributes. This lightweight process minimizes the impact on processing time, even with numerous contextual variables. Policy updates respond to changes such as user location and medical emergencies, ensuring security and efficiency.

2.1 Policy Integration into Emergency Contexts

Table 1 presents dynamic policy rules integrated into emergency medical contexts.

Table 1: Integration into Emergency Medical Contexts

Rule	**Description**	**Formula**
Access Time	Request time must meet or exceed the context-defined time.	$t_{request} \geq P$
Location	Access is allowed only within approved geographic areas.	$user\_location \in approved\_locations$
Medical Specialty	User must have the required specialty.	$user\_specialty \in specialties$
Consent	Access requires patient or authorized representative consent.	$patient\_consent = true \vee family\_authorized = true$
Environmental Conditions	Access depends on the current environmental status.	$environmental\_status = P$
Medical Emergency	Special access is triggered by a medical emergency.	$emergency\_detected = true$
Ambulance Status	Access is based on the ambulance's status and location.	$ambulance\_status = en\_route \wedge user\_location = ambulance\_location$
Session Duration	Access is limited to a specific time after session start.	$t_{session\_end} \leq t_{session\_start} + P$

2.2 Rule Combinations for Actions

Table 2 outlines the rule combinations required for each action type.

Table 2: Rule Combinations for Each Action

Action	**Policy**
Read	$R1 \wedge R2 \wedge R3 \wedge R4 \wedge R5 \wedge R6$
Update	$R1 \wedge R2 \wedge R3 \wedge R4 \wedge R5 \wedge R7 \wedge R8$
Start ES	$R1 \wedge R2 \wedge R3 \wedge R9$
End ES	$R1 \wedge R2 \wedge R3 \wedge R4 \wedge R10$

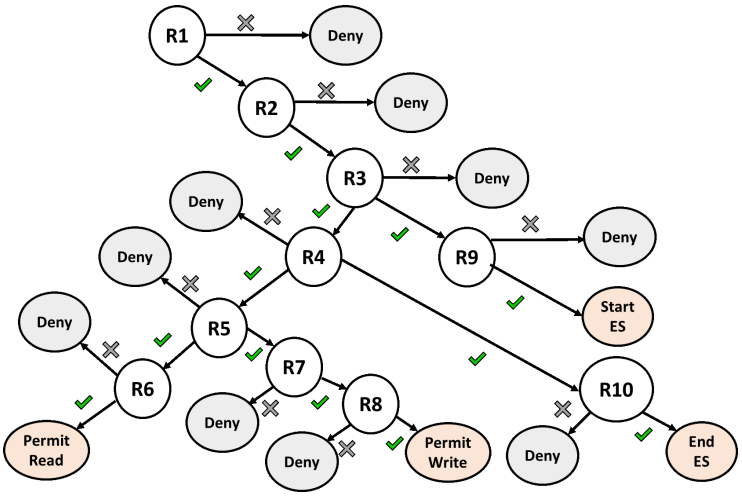

Figure 1: Policies Decision Tree

2.3 Core Policy Rules

In emergency settings, rapid and secure access is crucial. The BIG-ABAC framework implements a "PERMIT unless DENY" algorithm, granting access by default unless a specific condition triggers a denial. This approach balances accessibility and security in urgent scenarios.

Example parameters include user role, emergency session status, and time of access. Table 3 outlines rule parameters with real-time context-specific values.

Table 3: Rule Parameters with Real-Time Context-Specific Values

Entity	Rule	Description	Logical Representation
Subject	R1	The emergency is detected.	emergency_detected=true
	R2	The healthcare professional must have the required medical specialty.	user_specialty $\in$ {cardiologist, anesthetist, emergency_physician, ...}
	R3	The patient or authorized representative has given consent.	patient_consent=true $\vee$ family_authorized=true
Resource	R4	Only the EMR of the patient under emergency session (ES) must be available to the active emergency care team.	patient_id $\in$ ES $\wedge$ team_id $\in$ active_care_team
Action	R5	The request is made within the authorized time period.	current_time $\in$ [access_start_time, access_end_time]
	R6	The user is accessing from an approved location.	user_ip $\in$ {hospital_ip_range} $\vee$ user_location=ambulance_location
	R7	The treatment is required for patient care.	treatment_required=true
	R8	The request to update data is made by an authorized healthcare professional within the specified time.	update_authorized=true $\wedge$ within_time=true
	R9	The healthcare professional has the right to start the emergency session.	emergency_detected=true $\wedge$ valid_specialty=true $\wedge$ consent_or_authorization=true
	R10	The healthcare professional from the hospital acute care team has the right to end the emergency session unless it was started by another professional.	emergency_detected=true $\wedge$ valid_specialty=true $\wedge$ consent_or_authorization=true $\wedge$ within_time=true $\wedge$ not_starter=true

3 Policy Examples

3.1 Emergency Physician Access

This policy grants an emergency physician access to a patient's EMR during an active emergency session, within the authorized time window and location.

```
{
  "policyId": "policy123",
  "policyName": "Emergency Session Access Control",
  "subjects": [
    {
```

```
      "role": "emergency_physician",
      "status": "active"
    }
  ],
  "objects": [
    {
      "resourceType": "EMR",
      "resourcePath": "/healthcare/emr/patient12345"
    }
  ],
  "conditions": {
    "emergencySession": "active",
    "timeOfAccess": "09:00-17:00",
    "location": "hospital, ambulance"
  },
  "actions": [
    "read", "write"
  ],
  "decision": "PERMIT"
}
```

3.2 Nurse Access During Emergency

This policy permits a nurse to read a patient's EMR during an emergency session.

```
{
  "policyId": "policy456",
  "policyName": "Nurse Access Control",
  "subjects": [
    {
      "role": "nurse",
      "permissions": ["read"]
    }
  ],
  "objects": [
    {
     ```latex
      "resourceType": "EMR",
      "resourcePath": "/healthcare/emr/patient56789"
    }
  ],
  "conditions": {
    "emergencySession": "active",
    "timeOfAccess": "during_emergency"
  },
  "actions": [
    "read"
  ],
```
```

```
  "decision": "PERMIT"
}
```

4 Conclusion

The BIG-ABAC framework enables real-time adaptation of access policies in emergency healthcare, ensuring healthcare professionals have appropriate access during critical situations. These dynamic policy rules are crucial for managing access to sensitive patient data efficiently and securely.

For more detailed examples and the full implementation, refer to the repository: `https://github.com/BaccouriSondes/BIG-ABAC`.