Access Controls in Smart Cars: Needs and Solutions

Maanak Gupta
Postdoctoral Research Fellow
Institute for Cyber Security
University of Texas at San Antonio

Guest Lecture CS 6393/4593
April 05, 2019

gmaanakg@yahoo.com
http://sites.google.com/view/maanakgupta
Smart Cars Ecosystem

Safety and Assistance

Information and Entertainment

High Mobility, Location Centric
Time Sensitive, Dynamic Pairing
Multiple Fog/Cloud Infrastructures
No More Isolated!

100 million lines of code

Software Reliance, Broad Attack Surface, Untrusted Entities
The Perfect World.!

I TRUST my users.
Everything is Secure.!!

Confidentiality
Integrity
Availability

Thank YOU.!!
A user [U] is allowed to perform an operation [OP] on an object [OB] if security policy [P] is satisfied.
A user [U] is allowed to perform an operation [OP] on an object [OB] if security policy [P] is satisfied.
Three Dominant Models: DAC, MAC and RBAC.

**ABAC**: Decision based on the attributes of entities

Attributes are name value pair: `age (Alice) → 29`

Core entities in ABAC include:
- Users
- Objects
- Environment or Context
- Operations

**Authorization Policies**: determine rights just in time
  - retrieve attributes of relevant entities in request

Enhance flexibility and fine grained access control

NIST Guidelines to ABAC
Access Control Needs in Smart Cars

❖ On-Board Data, Applications and Sensors
❖ Third Party devices
❖ V2X fake messages
❖ User Privacy Preferences
❖ Over the Air updates
❖ Loss of Information in Cloud
❖ Location and time sensitivity of the services.
❖ In-vehicle communication
Scope of Contribution

➢ Contribution
   ❖ Access Control Oriented Architecture for Smart Cars.
   ❖ Propose formalized ABAC model for cloud assisted applications.
   ❖ Dynamic groups and user preferences.
   ❖ Implementation of the model in AWS.

➢ Scope
   ❖ Single Central Cloud
   ❖ No direct access and physical tampering
   ❖ Communication Channel is encrypted.
   ❖ Data in Cloud is secure
   ❖ In-vehicle security not considered
Extended Access Control Oriented Architecture

E-ACO architecture

Vehicular IoT components in architecture
An Authorization Framework:

- Helps understand access control needs.
- Helps understand models suitability for each.
Authorization Framework

**AWS-IoTAC Model**
- Policy Based
- Limited Attributes

**AWS-IoT-ACMVO Model**
- For Virtual Objects
- ACLs, RBAC, ABAC

Cloud Services Level
- AP - AP
- FG - CL
- CSR - CSR
- AP - CSR
- FG - FG
- U - AP

Object Level
- AP - OAP
- AP - OB
- OAP - OAP
- CO - OB
- U - OB
- OB - OB
- CO - CO
- U - OAP

Virtual Object Level
- AP - CL-VOB - OB
- AP - CL-VCO - CO
- CSR - VOB
- AP - VOB
- VCO - VOB
- AP - VCO
- OB - VOB
- OAP - VOB
- VCO - VCO
- VOB - VOB

U: User
CO: Clustered Objects
OB: Objects
OAP: Object Layer Applications
CL: Cloud
FG: Fog
CSR: Cloud Services
VCO: Virtual Clustered Objects
VOB: Virtual Objects
AP: User Applications
Authorization Framework

AWS-IoTAC Model
• Policy Based
• Limited Attributes

Our Solution
• Pure ABAC
• User Privacy

Cloud Supported

AWS-IoT-ACMVO Model
• For Virtual Objects
• ACLs, RBAC, ABAC

Cloud Services Level

Virtual Object Level

Object Level

CSR: Cloud Services  VCO: Virtual Clustered Objects  VOB: Virtual Objects  AP: User Applications
Location Groups

- Categorizing wide locations into smaller groups.
- Vehicles dynamically become member based on current GPS, vehicle-type or individual user preferences.
- Ensure relevance of alerts and notifications.
Speed Limit: 50 mph
Deer Threat: ON
Ice on Road: NO

Speed Limit: 30 mph
Flood Warning: ON
Road Work: ON

Speed Limit: 20 mph
School Zone: ON
Amber Alert: ABC123

Vehicle moves and are assigned to different groups and inherits their attributes/alerts.
Using Location Groups

Administrative Questions:
- How the attributes or alerts of groups are updated?
- How are moving entities assigned to groups?
- How groups hierarchy is created?

Operational Questions:
- How attributes and groups are used to provide security?
- How user privacy preferences are considered?

Reported MQTT message:
{"state": {"reported": {"Latitude": "29.4769353", "Longitude": "-98.5018237"}}}

Speed Limit: 50 mph
Deer Threat: ON
Ice on Road: NO

Speed Limit: 30 mph
Flood Warning: ON
Road Work: ON
CV-ABAC$_G$ Model

- **Activity Decision**: S performs

- **System Level**: POL, ATT

- **Activities (A)**: CO, O, OP, G

- **Attribute / Policy Association**: POL to ATT

- **Many to Many**: POL to CO, CO to O, O to OP, OP to G

- **Many to Many Dynamic Group Association**: G to OP

- **Zero or More**: One to Many Association

© Maanak Gupta

World Leading Research with Real World Impact!
Model Components

user, sensor, car, mechanic, restaurant

{ location, size, IP, direction, speed, VIN, cuisine-type}
Model Components

{ read, write, control, notify, administrative actions }
Model Components

- Cars, traffic lights, smart-devices
- Sensor, ECU, on-board apps
- Location groups, service-specific, vehicle-type
Model Components

Operational and Administrative Activities
{notification, alerts, group hierarchy updates}

System Wide Policies
Individualized Privacy Policies
Basic Sets and Functions
- S, CO, O, G, OP are finite sets of sources, clustered objects, objects, groups and operations respectively [blue circles in Figure 4].
- A is a finite set of activities which can be performed in system.
- ATT is a finite set of attributes associated with S, CO, O, G and system-wide.
- For each attribute \( \text{att} \) in ATT, Range(\( \text{att} \)) is a finite set of atomic values.
- \( \text{attType}: \) ATT = \{set, atomic\} defines attributes to be set or atomic valued.
- Each attribute \( \text{att} \) in ATT maps entities in S, CO, O, G to attribute values. Formally,
  \[
  \text{att} : S \cup CO \cup O \cup G \cup \{\text{system-wide}\} \rightarrow \begin{cases} 
  \text{Range}(\text{att}) \cup \{\bot\} & \text{if attType(\text{att}) = atomic} \\
  2^{\text{Range}(\text{att})} & \text{if attType(\text{att}) = set}
  \end{cases}
  \]
- POL is a finite set of authorization policies associated with individual S, CO, O, G.
- directG : CO \( \rightarrow \) G, mapping each clustered object to a system group, equivalently CGA \( \subseteq \) CO \times G.
- parentCO : O \( \rightarrow \) CO, mapping each object to a clustered object, equivalently OCA \( \subseteq \) O \times CO.
- \( GH \subseteq G \times G \), a partial order relation \( \geq_{g} \) on G. Equivalently, parentG : G \( \rightarrow \) \( 2^{G} \), mapping group to a set of parent groups in hierarchy.
Effective Attributes of Groups, Clustered Objects and Objects (Derived Functions)

- For each attribute att in ATT such that attType(att) = set:
  - effG\_att : G \rightarrow 2^{\text{Range}(att)}, defined as effG\_att(g_i) = att(g_i) \cup \bigcup_{g \in \{g_i | g_i \geq g\}} \text{effG}\_att(g).
  - effCO\_att : CO \rightarrow 2^{\text{Range}(att)}, defined as effCO\_att(co) = att(co) \cup \text{effG}\_att(directG(co)).
  - effO\_att : O \rightarrow 2^{\text{Range}(att)}, defined as effO\_att(o) = att(o) \cup \text{effCO}\_att(parentCO(o)).

- For each attribute att in ATT such that attType(att) = atomic:
  - effG\_att : G \rightarrow \text{Range}(att) \cup \{\bot\}, defined as effG\_att(g_i) = \begin{cases} att(g_i) & \text{if } \forall g' \in \text{parentG}(g_i), \text{effG}\_att(g') = \bot \\ \text{effG}\_att(g') & \text{if } \exists \text{parentG}(g_i), \text{effG}\_att(\text{parentG}(g_i)) \neq \bot \text{ then select parent } g' \text{ with } \text{effG}\_att(g') \neq \bot \text{ updated most recently.} \end{cases}
  - effCO\_att : CO \rightarrow \text{Range}(att) \cup \{\bot\}, defined as effCO\_att(co) = \begin{cases} att(co) & \text{if } \text{effG}\_att(directG(co)) = \bot \\ \text{effG}\_att(directG(co)) & \text{otherwise} \end{cases}
  - effO\_att : O \rightarrow \text{Range}(att) \cup \{\bot\}, defined as effO\_att(o) = \begin{cases} att(o) & \text{if } \text{effCO}\_att(parentCO(o)) = \bot \\ \text{effCO}\_att(parentCO(o)) & \text{otherwise} \end{cases}

Attributes more Dynamic
Attributes Inheritance
Authorization Functions (Policies)

Authorization Function: For each op ∈ OP, Auth_{op}(s : S, ob : CO ∪ O ∪ G) is a propositional logic formula returning true or false, which is defined using the following policy language:

- α ::= α ∧ α | α ∨ α | (α) | ¬α | ∃x ∈ set.α | ∀x ∈ set.α | set ∆ set | atomic ∈ set | atomic ∉ set
- ∆ ::= ⊆ | ⊊ | ⊄ | ∩ | ∪
- set ::= eff_{att}(i) | att(i)
- atomic ::= eff_{att}(i) | att(i) | value

for att ∈ ATT, i ∈ S ∪ CO ∪ O ∪ G ∪ {system-wide}, attType(att) = set

for att ∈ ATT, i ∈ S ∪ CO ∪ O ∪ G ∪ {system-wide}, attType(att) = atomic

❖ Administrators in the police department can send alert to location-groups in city limits.

\[ \text{Auth}_{\text{alert}}(u:U, g:G) :: \text{dept}(u) \land \text{Police} \land \text{parent-city}(g) = \text{Austin} \land \text{Austin} \in \text{jursidiction}(u). \]

❖ Only mechanic in the technician department from Toyota-X dealership must be able to read sensor in Camry LE. Further, this operation must be done between time 9 am to 6 pm.

\[ \text{Auth}_{\text{read}}(u:U, co:CO) :: \text{role}(u) \land \text{Technician} \land \text{employer}(u) = \text{Toyota-X} \land \text{make}(co) = \text{Toyota} \land \text{model}(co) = \text{Camry LE} \land \text{operation_time}(u) \in \{9am,10,11...6pm\} \]
Authorization Decision

A source $s \in S$ is allowed to perform an activity $a \in A$, stated as $\text{Authorization}(a : A, s : S)$, if the required policies needed to allow the activity are included and evaluated to make a final decision. These multi-layer policies must be evaluated for individual operations ($\text{op}_j \in \text{OP}$) to be performed by source $s \in S$ on relevant objects ($x_i \in CO \cup LO \cup G$).

Formally, $\text{Authorization}(a : A, s : S) \Rightarrow \text{Auth}_\text{op}_1(s : S, x_1), \text{Auth}_\text{op}_2(s : S, x_2), \ldots, \text{Auth}_\text{op}_n(s : S, x_n)$

Evaluate all relevant policies to make a decision

A restaurant in group A must be allowed to send notifications to all vehicles in location group A and group B.

I only want notifications from Cheesecake factory.

System defined

User Preference
Implementation in Amazon Web Services (AWS)
Vehicles and Groups

4 Location Groups (static demarcation)

Vehicles movement (coordinates generated using Google API)

('Received new coordinates from:', 'Vehicle-1')
Sun May 27 02:56:30 2018
Location A
  Car-A: ['Vehicle-1', 'Vehicle-2']
  Bus-A: []
Location B
  Car-B: []
  Bus-B: ['Vehicle-6']
Location C
  Car-C: ['Vehicle-3', 'Vehicle-4']
  Bus-C: []
Location D
  Car-D: []
  Bus-D: ['Vehicle-5']

Snapshot (table keeps changing)
Implementated Policies

➢ Administrative Policy
  ❖ Road side motion sensor with \([id = 1]\) and current GPS in group \([\text{Location-A}]\) can only [modify] attribute \([\text{Deer Threat}]\) to value \([\text{ON, OFF}]\) for group \([\text{Location-A}]\).

➢ Operational Policy
  
  **Restaurant Notification Use Case**

  **System Defined Policy**
  ❖ A restaurant located within group \([\text{Location-A}]\) can only [send notifications] to members of groups \([\text{Location-A, Location-B}]\).

  **User Preferences**
  ❖ Send notifications only between \([7 \text{ pm to 9 pm}]\) only on \([\text{Wednesdays}]\).
### Performance Metrics

#### Policy Enforcement Time

<table>
<thead>
<tr>
<th>Number of Requests</th>
<th>Policy Enforcer Execution Time (in ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0.0501</td>
</tr>
<tr>
<td>20</td>
<td>0.1011</td>
</tr>
<tr>
<td>30</td>
<td>0.1264</td>
</tr>
<tr>
<td>40</td>
<td>0.1630</td>
</tr>
<tr>
<td>50</td>
<td>0.1999</td>
</tr>
</tbody>
</table>

#### Relevance of Alerts and Notifications

<table>
<thead>
<tr>
<th>n&lt;sup&gt;th&lt;/sup&gt; Request</th>
<th>With ABAC Policy</th>
<th>Without Policy</th>
</tr>
</thead>
<tbody>
<tr>
<td>41&lt;sup&gt;st&lt;/sup&gt;</td>
<td>20</td>
<td>50</td>
</tr>
<tr>
<td>42&lt;sup&gt;nd&lt;/sup&gt;</td>
<td>30</td>
<td>50</td>
</tr>
<tr>
<td>43&lt;sup&gt;rd&lt;/sup&gt;</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>44&lt;sup&gt;th&lt;/sup&gt;</td>
<td>30</td>
<td>50</td>
</tr>
<tr>
<td>45&lt;sup&gt;th&lt;/sup&gt;</td>
<td>20</td>
<td>50</td>
</tr>
<tr>
<td>46&lt;sup&gt;th&lt;/sup&gt;</td>
<td>30</td>
<td>50</td>
</tr>
</tbody>
</table>
Performance Metrics

Comparing Policy vs No Policy Execution Time
Lets Talk ..!!

Questions, Comments or Concerns

gmaanakg@yahoo.com

https://sites.google.com/view/maanakgupta