I·C·S
The Institute for Cyber Security

C·SPECC
Center for Security and Privacy Enhanced Cloud Computing

# The HABAC Model for Smart Home IoT and Comparison to EGRBAC

Safwa Ameer

Ravi Sandhu

SaT-CPS 2021

**Institute for Cyber Security (ICS)**
**Center for Security and Privacy Enhanced Cloud Computing (C-SPECC)**
**Department of Computer Science**
**University of Texas at San Antonio**



*World-Leading Research with Real-World Impact!*

UTSA Computer Science

# Introduction & Motivation



- The Internet of Things (IoT) is a new technology paradigm envisioned as a global network of machines and devices capable of interacting with each other.

- One of the most popular domains for deploying smart connected devices is the smart home.

# Problem Statement



- In the literature, several AC models have been proposed for IoT in general. Most of them are built on RBAC or ABAC.
- Some researchers argue that RBAC is more suitable for IoT, since it is simpler in management and review, while ABAC is complex.
- Others argue that ABAC models are more scalable and dynamic, since they can capture different devices and environment contextual information.
- However, RBAC models can be extended, such as the recent EGRBAC model [4] for smart home IoT which can express environment and device characteristics.
- Hence, when it comes to smart homes, at this point it is not fully clear what is the benefit of ABAC over RBAC, and vice versa.

# Problem Statement



- Our intuitive insight is that a hybrid model will better capture smart home IoT AC requirements as this was already the case for traditional access control models.

- In order to further investigate this intuition our approach is to develop pure RBAC and pure ABAC based models explicitly defined to meet smart home challenges and compare their benefits and drawbacks.

- This comparison will provide insights to guide us in designing "optimal" hybrid models.

# Threat Model



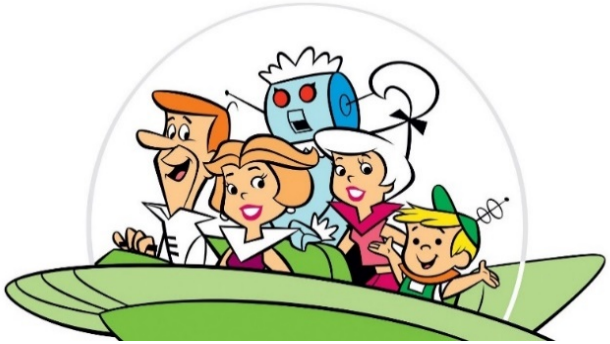
- Scope or threat model:

  In smart houses we have two types of adversaries:

  a- Outsider hacker who is trying to get digital or physical access to the house by exploiting system vulnerabilities.

  b- The household members themselves. The insiders who have legitimate digital and physical access to the house, such as family members, guests, and workers.

- **The central focus of our work** is making sure that those legitimate users get access only to what they are authorized to by the house owner.



# Our Contribution



1. Design, formalize ABAC based access control model for smart home IoT (HABAC).

2. Analyze HABAC model relative to the previously published EGRBAC model [4]. Compare the theoretical expressive power of these models by:

   a. Introduce HABAC configuration that translates EGRBAC policies in a manner that they can be implemented by HABAC.

   b. Provide an algorithm to convert an HABAC specification to EGRBAC.

   c. Discuss the insights for practical deployment of these models resulting from these constructions.

# Our Contribution



1. Design, formalize ABAC based access control model for smart home IoT (HABAC).

2. Analyze HABAC model relative to the previously published EGRBAC model [4]. Compare the theoretical expressive power of these models by:

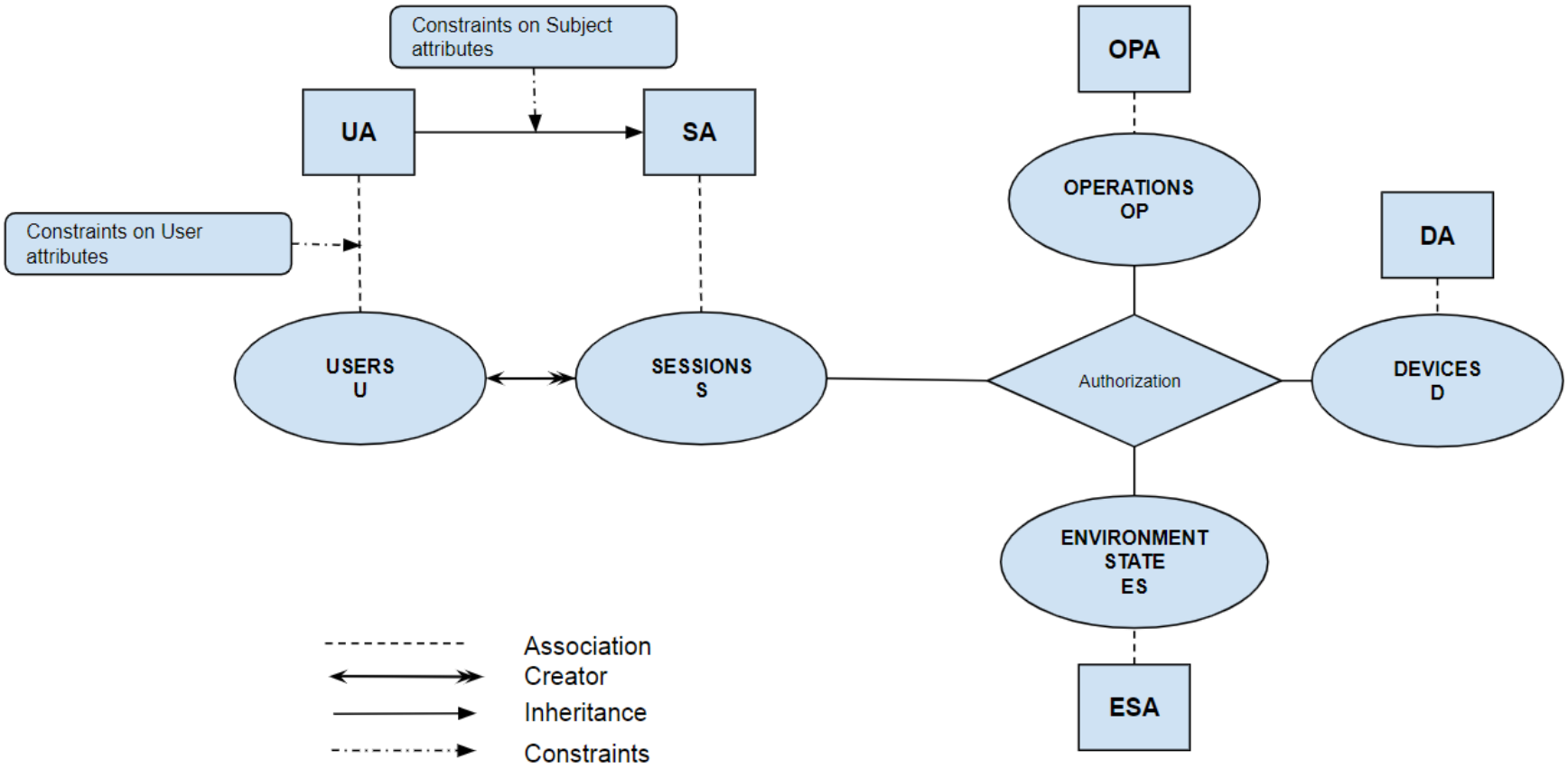
   a. Introduce HABAC configuration that translates EGRBAC policies in a manner that they can be implemented by HABAC.

   b. Provide an algorithm to convert an HABAC specification to EGRBAC.

   c. Discuss the insights for practical deployment of these models resulting from these constructions.

# The HABAC Model

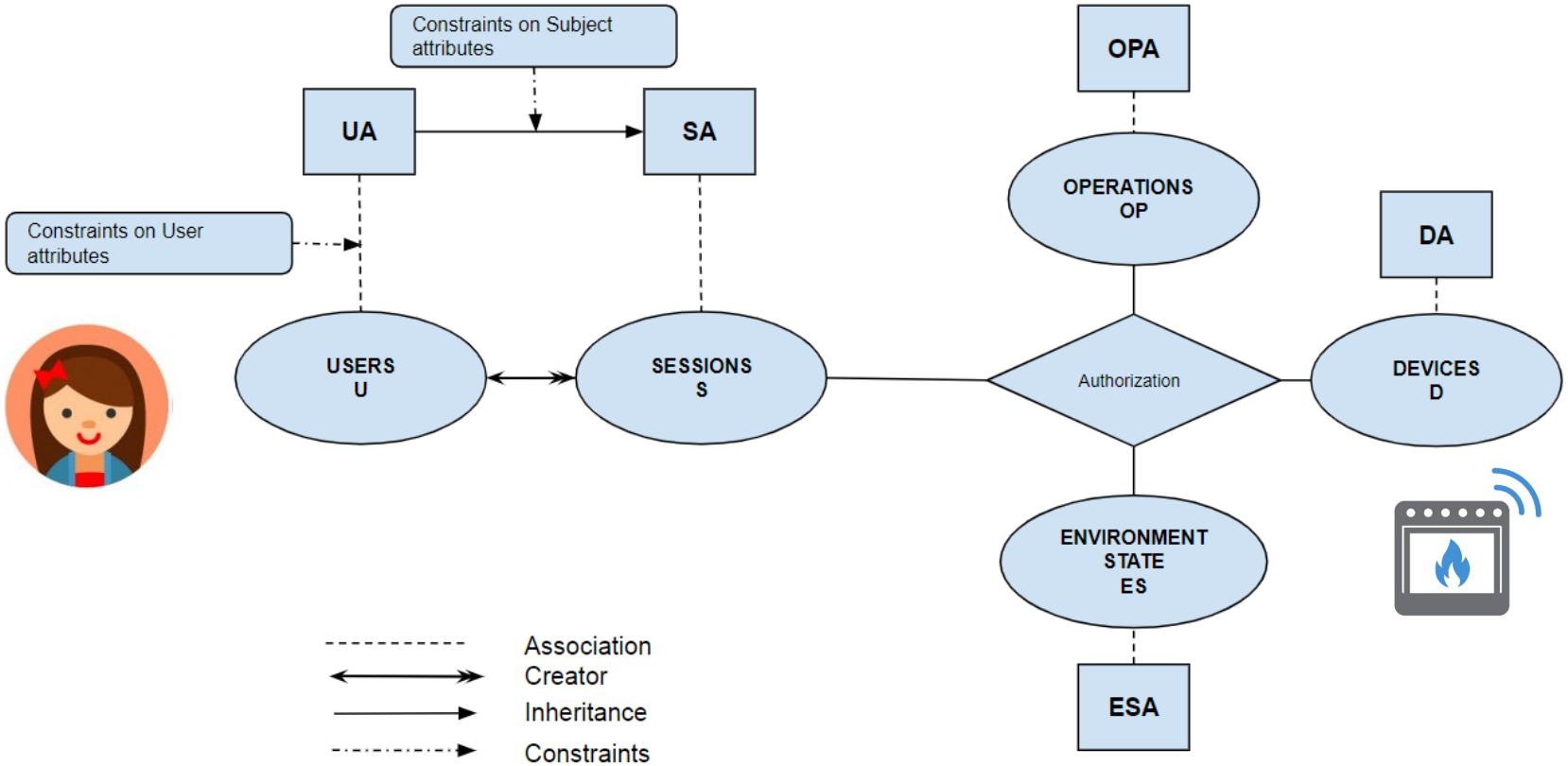




# Use Case

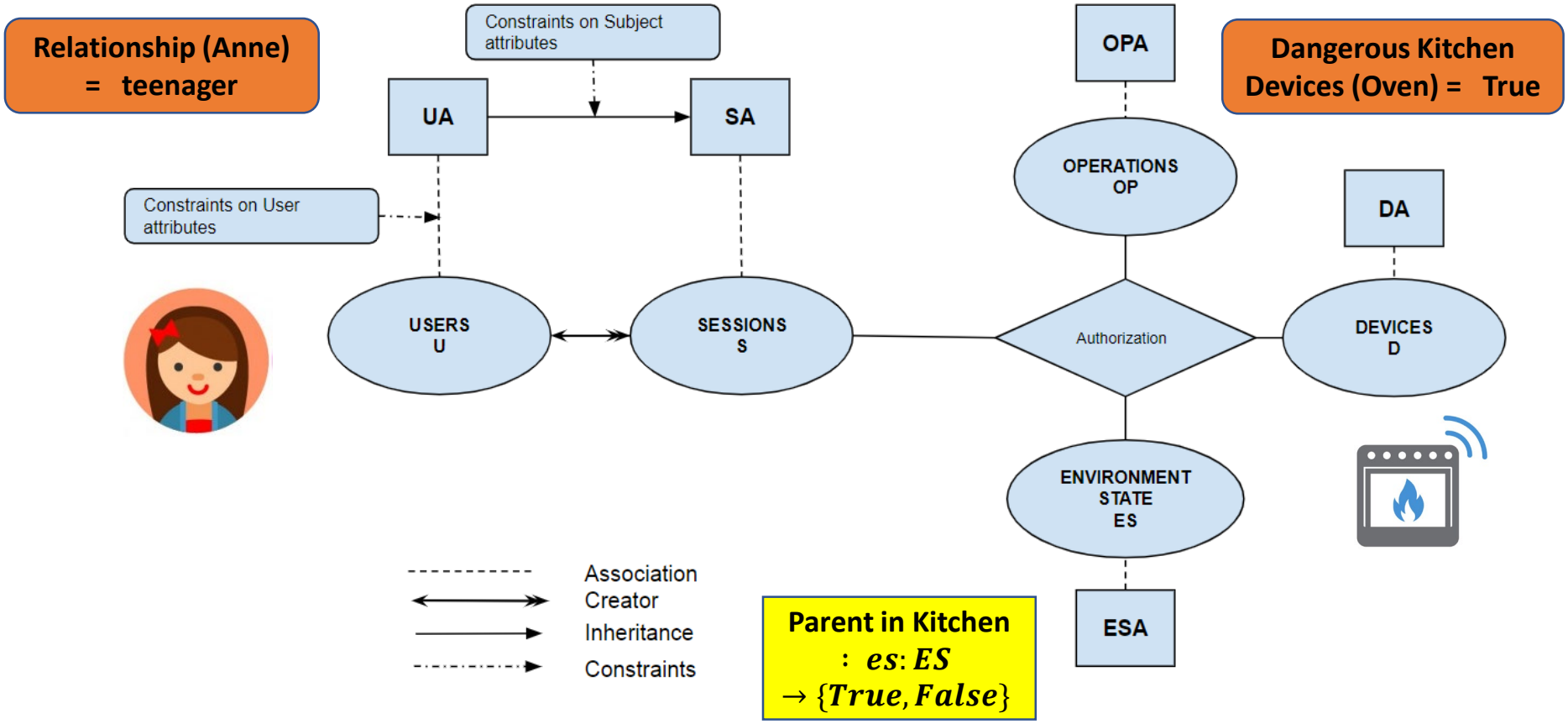


**The goal is to** : Authorize teenagers (Anne) to use dangerous kitchen devices (Oven) only when one of the parents is in the kitchen.



# Use Case



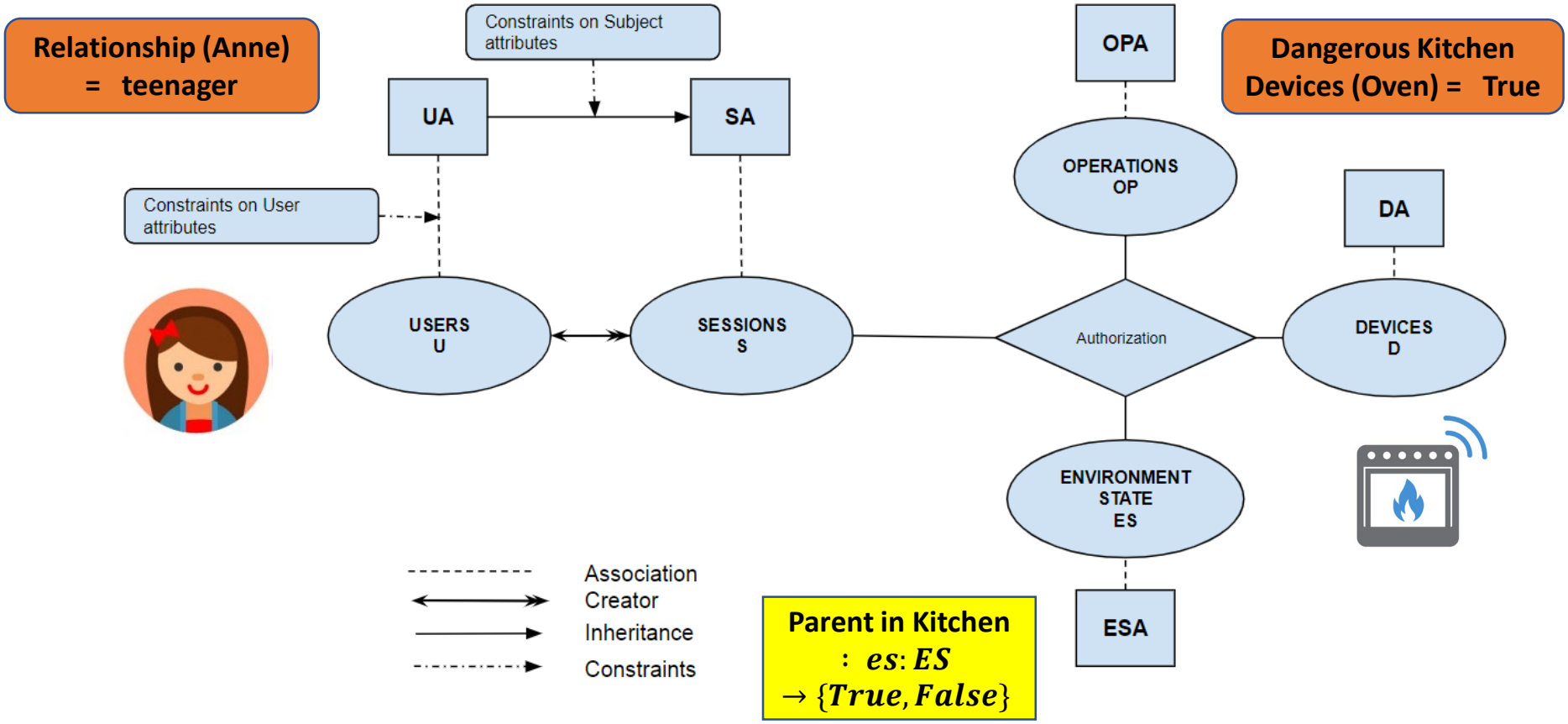


# Use Case



**Add the authorization rule:**

$$(Relationship(s) = teenager \wedge ParentInKitchen(current) = True \wedge DangerouseKitchenDevices(d) = True) \vee$$

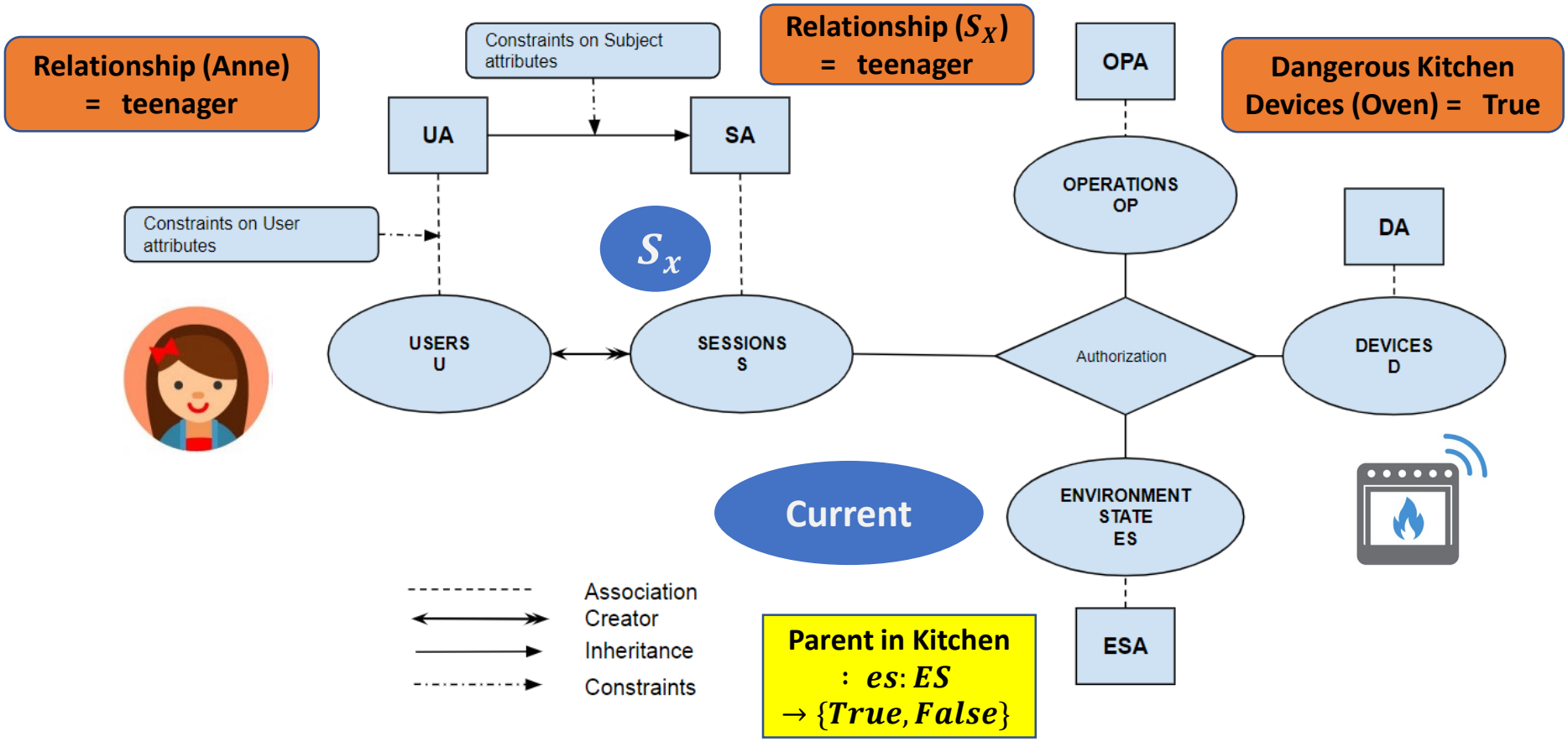


# Use Case



**Check the authorization policy:**

$$(Relationship(s) = teenager \wedge ParentInKitchen(current) = True \wedge DangerouseKitchenDevices(d) = True) \vee$$

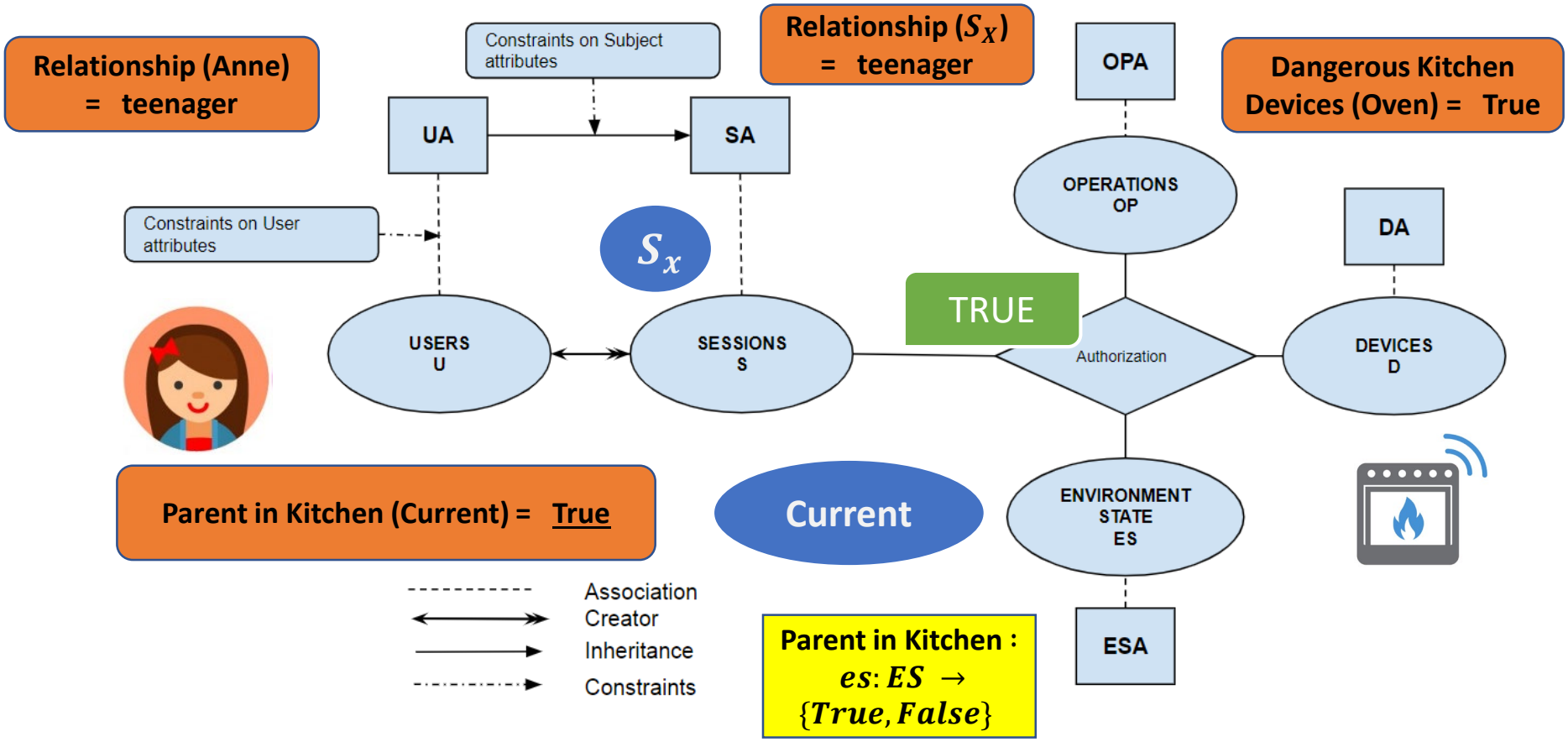


# Use Case



**Check the authorization policy:**

$$(Relationship(s) = teenager \wedge ParentInKitchen(current) = True \wedge DangerouseKitchenDevices(d) = True) \vee$$

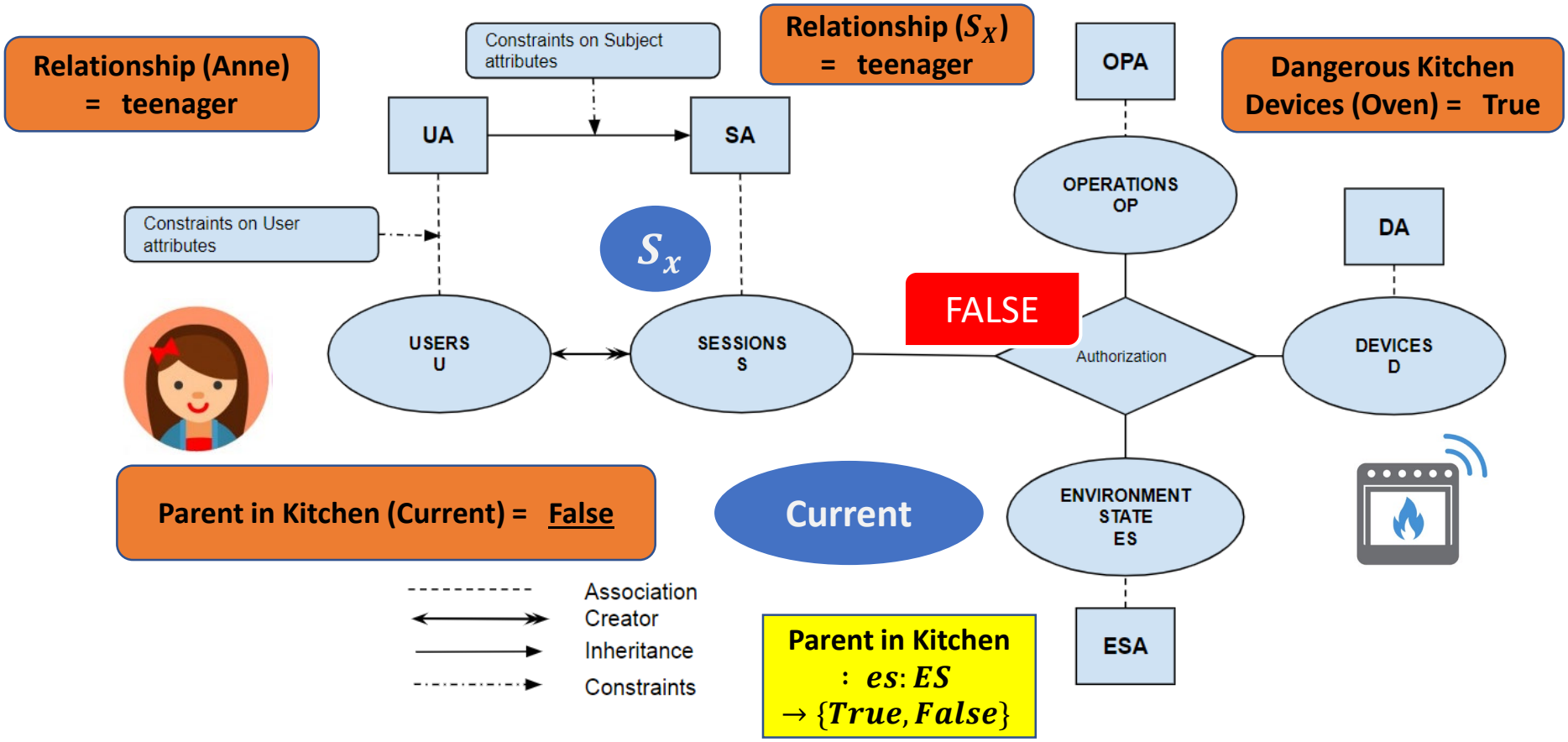


# Use Case



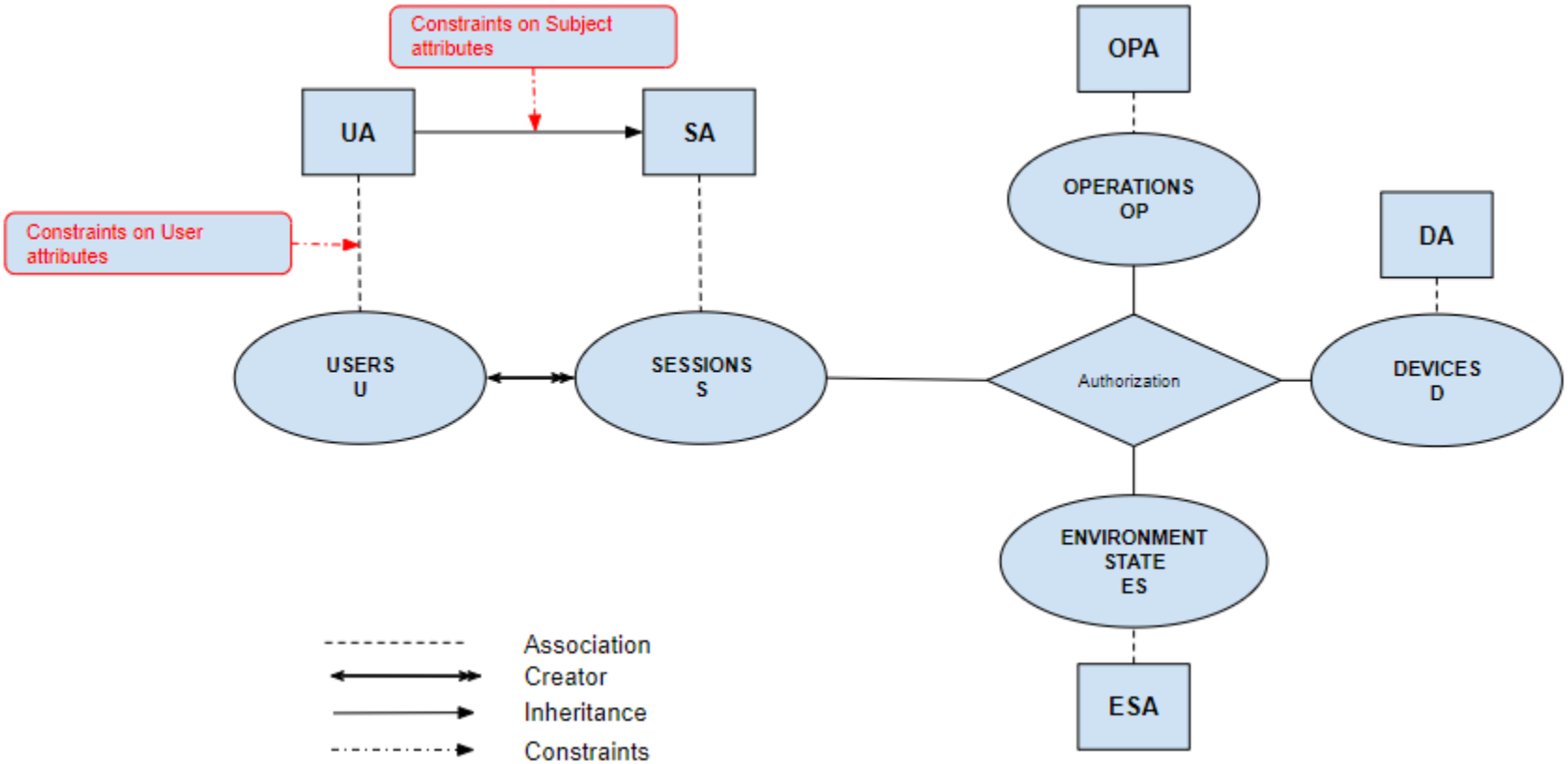
**Check the authorization policy:**

$$(Relationship(s) = teenager \wedge ParentInKitchen(current) = True \wedge DangerouseKitchenDevices(d) = True) \vee$$



# The HABAC Model
## 3- Constraints



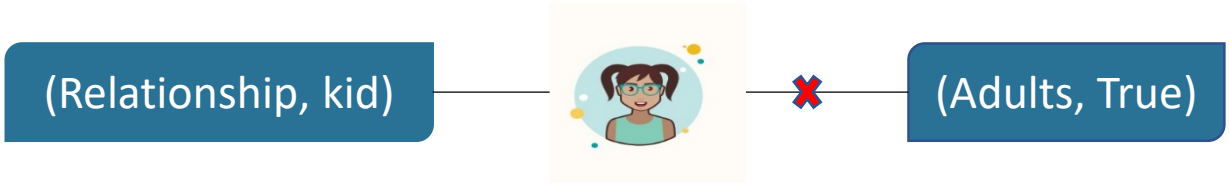


# The HABAC Model
## 3- Constraints



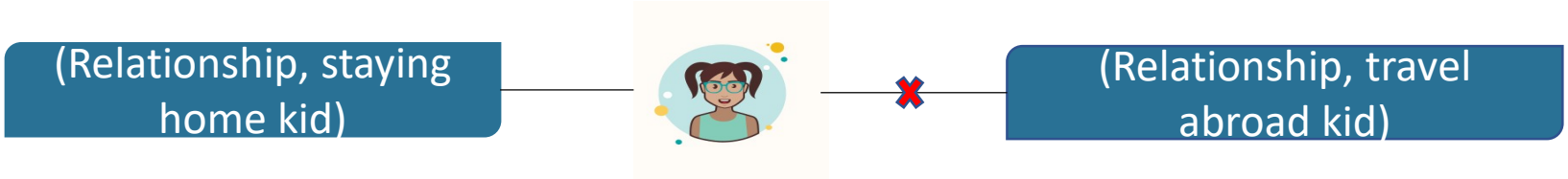
1- Constraints on user attributes: these constraints enforce restrictions on user attributes.

- $\boldsymbol{UAConstraints} \subseteq \boldsymbol{UAP} \times \boldsymbol{2^{UAP}}$ . Constitute a many to many user attribute pair to a subset of mutually exclusive user attribute pairs.
- Ex: $uac = \big((Relationship, kid)\ ,\ \{(Adults, True)\}\big)$




2- Constraints on session attributes: these constraints enforce restrictions on session attributes.

- $\boldsymbol{SAConstraints} \subseteq \boldsymbol{SAP} \times \boldsymbol{2^{SAP}}$ . Constitute a many to many session attribute pair to a subset of mutually exclusive session attribute pairs.
- Ex: $sac = \begin{pmatrix} (Relationship,\ staying\ home\ kid)\ , \\ \{(Relationship,\ \ travel\ abroad\ kid)\} \end{pmatrix}$



# Our Contribution



1. Design, formalize ABAC based access control model for smart home IoT (HABAC).

2. Analyze HABAC model relative to the previously published EGRBAC model [4]. Compare the theoretical expressive power of these models by:

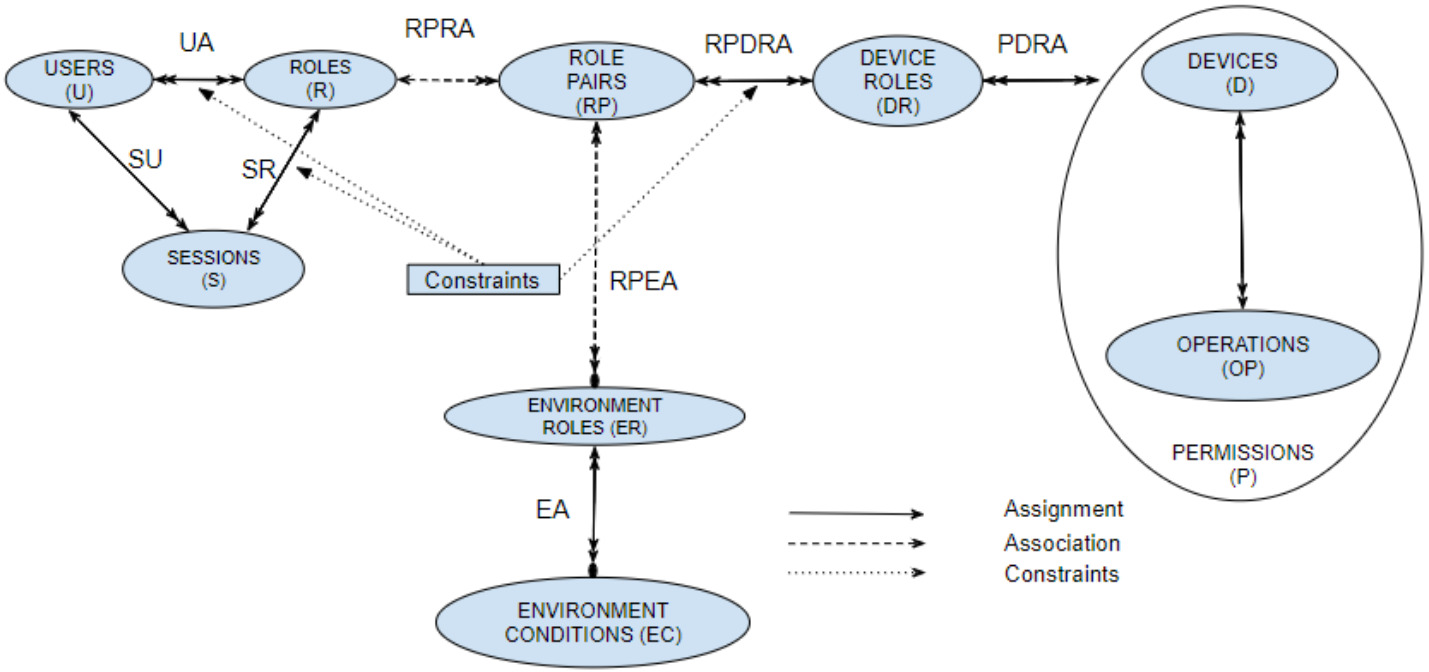
   a. Introduce HABAC configuration that translates EGRBAC policies in a manner that they can be implemented by HABAC.

   b. Provide an algorithm to convert an HABAC specification to EGRBAC.

   c. Discuss the insights for practical deployment of these models resulting from these constructions.

# The HABAC Model
## 3- Constraints






- The main idea in EGRBAC is that a user is assigned to a set of roles and according to the current active sessions, and current active environment roles some role pairs will be active, the user will get access to the permissions assigned to the device roles which are assigned to the current active role pairs.

# Our Contribution



1. Design, formalize ABAC based access control model for smart home IoT (HABAC).

2. Analyze HABAC model relative to the previously published EGRBAC model [4]. Compare the theoretical expressive power of these models by:

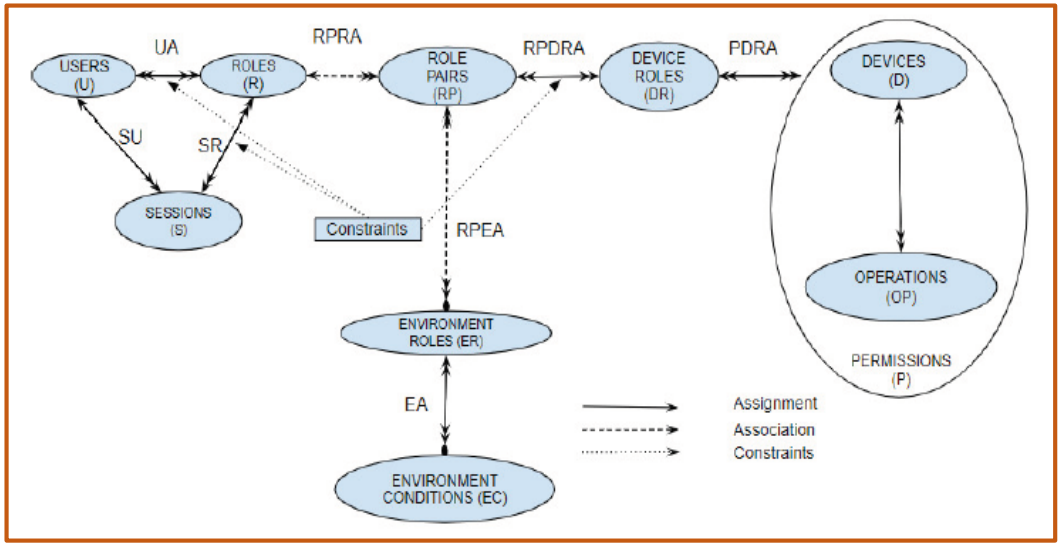
   a. Introduce HABAC configuration that translates EGRBAC policies in a manner that they can be implemented by HABAC.

   b. Provide an algorithm to convert an HABAC specification to EGRBAC.

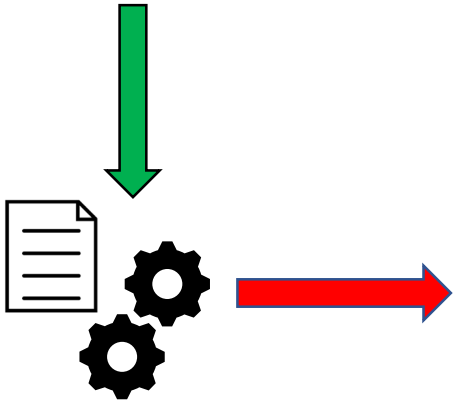
   c. Discuss the insights for practical deployment of these models resulting from these constructions.

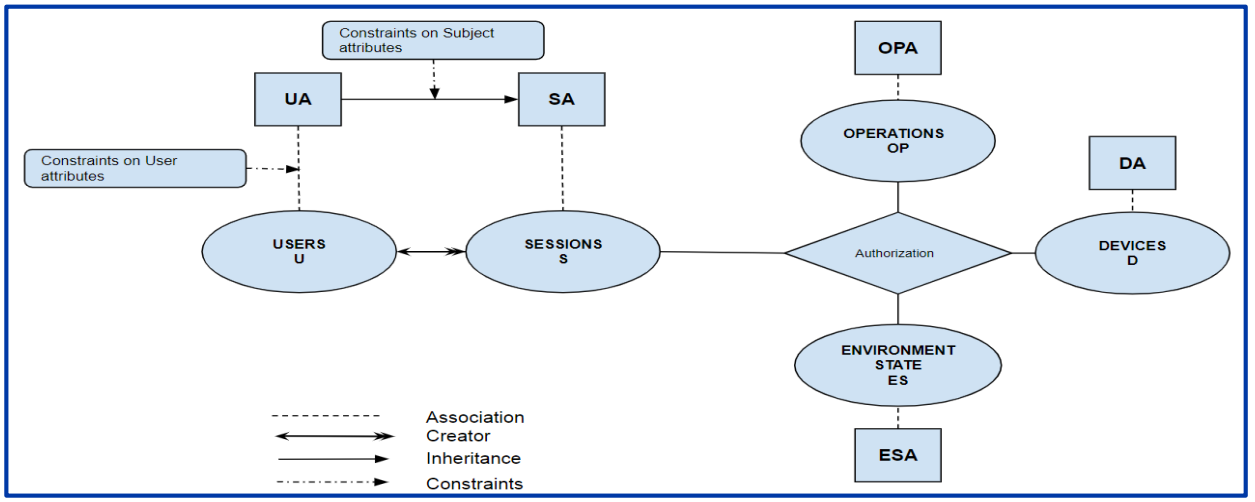
# Constructing HABAC From EGRBAC






The goal is to construct HABAC elements (U, UA, SA, ES, ESA, D, DA, OP, OPA) and the authorization policy function from EGRBAC policy in such a way that the authorizations are the same as those under EGRBAC.





# Constructing HABAC From EGRBAC



- $U_{HABAC} = U_{EGRBAC}$
- $UA = SA = \{Relationship\}$
- $Relationship : u \in U_{HABAC} \rightarrow 2^R$
- $Relationship : s \in S \rightarrow 2^R$
- $(\forall u_i \in U_{HABAC})[Relationship(u_i) = \{r_x | (u_i, r_x) \in UA\}]$

- $UAConstraint = \{uac_i\}$
- For all $ssdc_i = (r_i, R_j) \in SSDConstraints$:
$uac_i = ((Relationship, r_i), UAP_j)$, where $UAP_j = \{(Relationship, r_n) | r_n \in R_j\}$

- $SAConstraint = \{sac_i\}$
- For all $dsdc_i = (r_i, R_j) \in DSDConstraints$:
$sac_i = ((Relationship, r_i), SAP_j)$, where $SAP_j = \{(Relationship, r_n) | r_n \in R_j\}$

- The set of **users** are the same in both systems.
- Roles are expressed through the user attribute ***Relationship*** in HABAC.
- *Relationship* is a user and a session attribute that takes a user or a session as an input and returns the set of roles assigned to that user or that session

- Static separation of duty constraints *SSDConstraints* are translated into **user attributes constraints** in HABAC.
- Dynamic separation of duty constraints *DSDConstraints* are translated into **subject attributes constraints** in HABAC

# Constructing HABAC From EGRBAC



- $ES = \{Current\}$
- $ESA = ER$

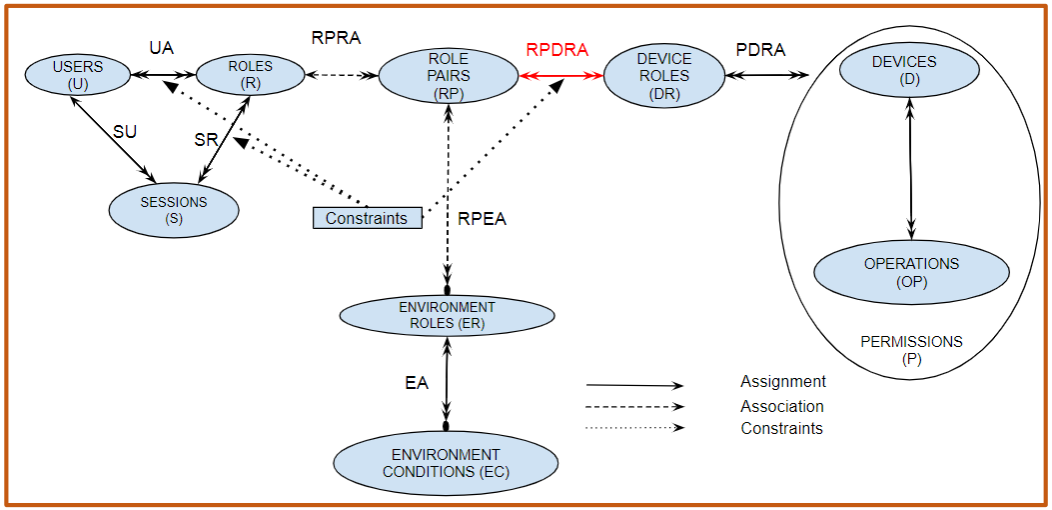
$(\forall esa_i \in ESA)[esa_i : es \in ES \rightarrow \{True, False\}]$

- $D_{HABAC} = D_{EGRBAC}, OP_{HABAC} = OP_{EGRBAC}$
- $DA = OPA = DR$
- $(\forall da_i \in DA)\ [da : d \in D_{HABAC} \rightarrow \{True, False\}]$
- $(\forall opa_i \in OPA)[opa : op \in OP_{HABAC} \rightarrow \{True, False\}]$
- $(\forall(dr_y \in DR, p_x \in \{p_i|(p_i, dr_y) \in PDRA\}))[dr_y(p_x.op) = True, dr_y(p_x.d) = True]$
- $(\forall(dr_y \in DR, p_y \in \{p_j|(p_j, dr_y) \notin PDRA\}))[dr_y(p_x.op) = False, dr_y(p_x.d) = False]$

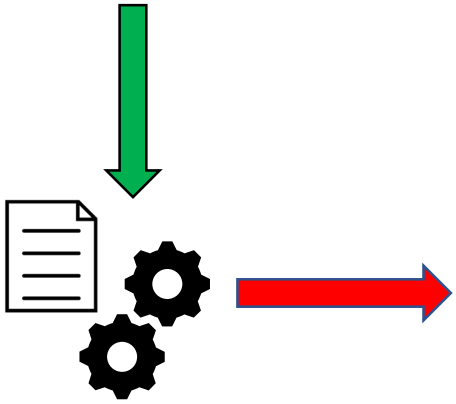
- Environment roles are translated into atomic environment state attributes.
- Device roles in EGRBAC are translated into atomic operation attributes and atomic device attributes with a range of values equal to $\{True, False\}$.

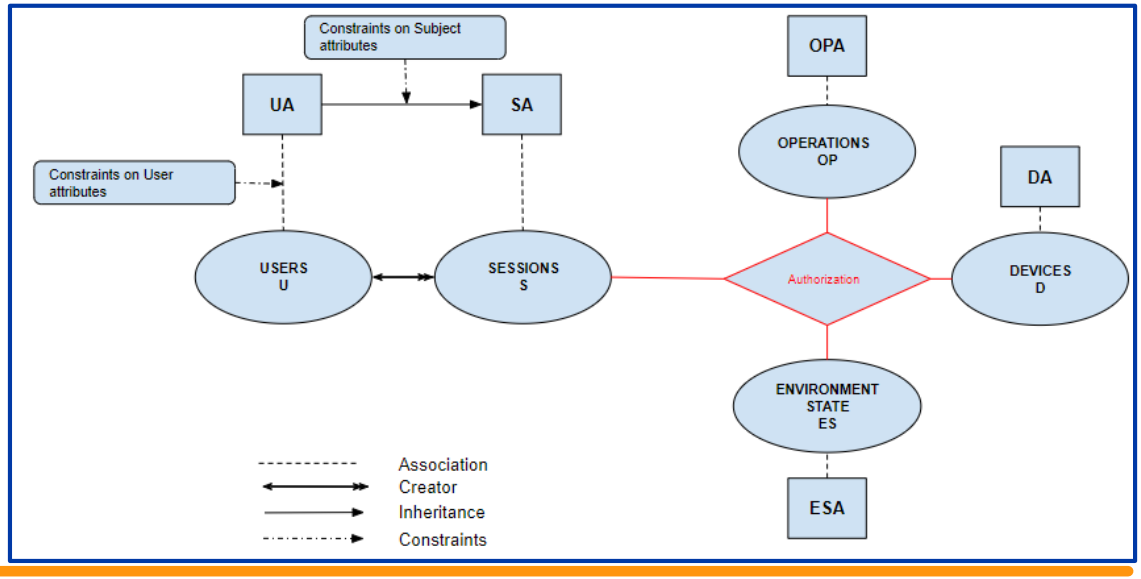
# Constructing HABAC From EGRBAC






- The final step is to construct the authorization policies.
- In EGRBAC it is the $RPDRA$ that gives specific role pairs and hence users access to specific device roles and hence permissions.





# Constructing HABAC From EGRBAC



- For each $rpdra_i = ((r_i, ER_i), dr_i) \in RPDRA$, we construct an authorization policy as following:
$[Authorization_{op}(s : S, es : ES, d : D) \equiv Relationship(s) = r_i \wedge dr_i(op) = True \wedge (\bigcap_{esa \in ER_i} esa(current) = True) \wedge dr_i(d) = True]$
- The final authorization policy is the disjunction of every created authorization policy.

- Therefore, we translate each $rpdra_i = ((r_i, ER_i), dr_i) \in RPDRA$ into an authorization policy.
- The final authorization policy is the disjunction of every created authorization policy.

# Our Contribution



1. Design, formalize ABAC based access control model for smart home IoT (HABAC).

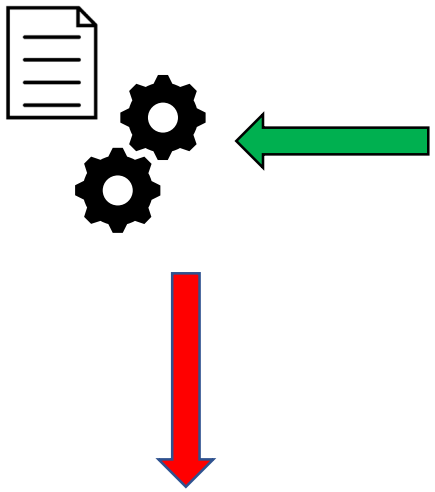
2. Analyze HABAC model relative to the previously published EGRBAC model [4]. Compare the theoretical expressive power of these models by:

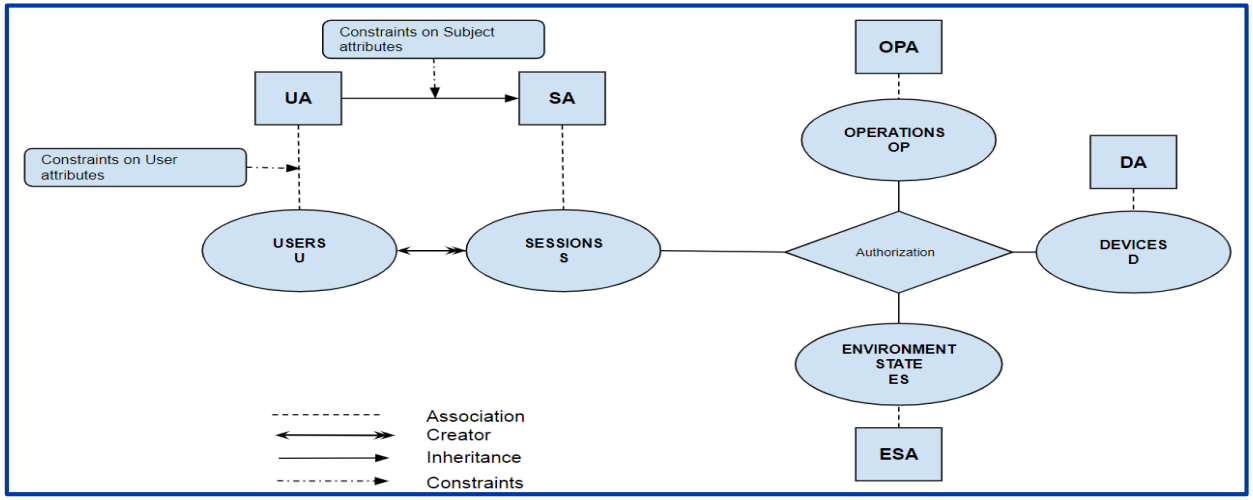
   a. Introduce HABAC configuration that translates EGRBAC policies in a manner that they can be implemented by HABAC.

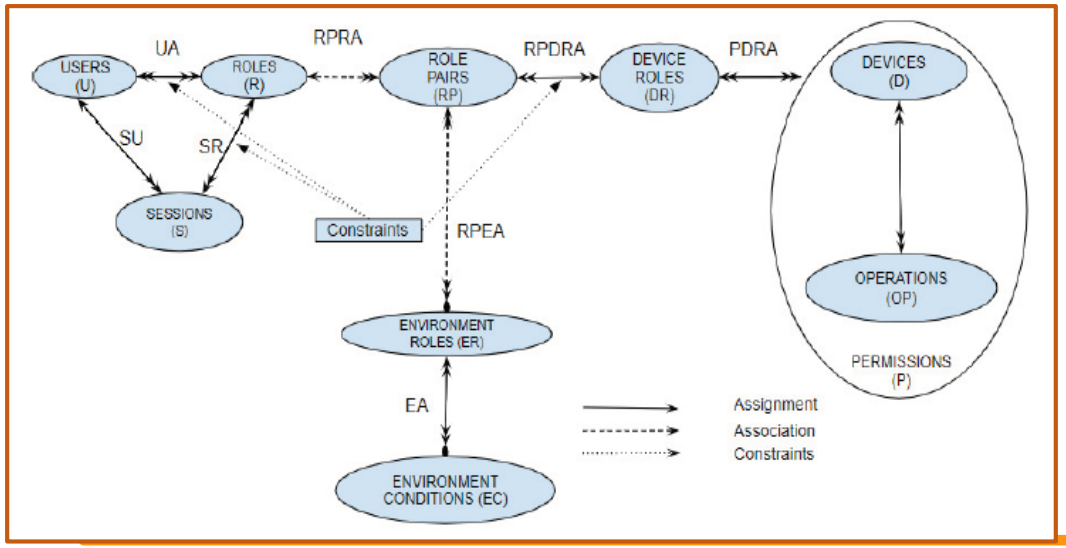
   b. Provide an algorithm to convert an HABAC specification to EGRBAC.

   c. Discuss the insights for practical deployment of these models resulting from these constructions.

# Constructing EGRBAC From HABAC






The goal is to construct EGRBAC elements (U, R, EC, ER, RP, D, OP, P, DR), assignments (UA, EA, PDRA, RPDRA), and associations (RPRA, RPEA) from HABAC policies in such a way that the authorizations are the same as those under HABAC.

# Constructing EGRBAC From HABAC



$U = \{alex, bob, anne\},$
$UA = \{Relationship, Location\}$
$Relationship : u : U \rightarrow \{parent, kid, teenager\}$
$Location : u : U \rightarrow \{Kitchen, MasterBedRoom, BedRoom_1, BedRoom_2, LivingRoom\}$
$Relationship(alex) = kid$
$Relationship(anne) = teenager$
$Relationship(bob) = parent$

$S = \{\ldots\}$
$SA = \{Relationship, Location\}$
$Relationship : s : S \rightarrow \{parent, kid, teenager\}$
$Location : s : S \rightarrow \{Kitchen, MasterBedRoom, BedRoom_1, BedRoom_2, LivingRoom\}$

$D = \{TV, PlayStation, Oven, Fridge, FrontDoor\}$
$DA = \{DangerouseKitchenDevices\}$
$DangerouseKitchenDevices : d : D \rightarrow \{True, False\}$
$DangerouseKitchenDevices(Oven) = True$
$DangerouseKitchenDevices(Fridge) = False$

$ES = \{Current\}$
$ESA = \{day, time, ParentInKitchen\}$
$day : es : ES \rightarrow \{S, M, T, W, Th, F, Sa\}$
$time : es : ES \rightarrow \{x | x \text{ is an hour of a day }\}$
$ParentInKitchen : es : ES \rightarrow \{True, False\}$

$OP_{TV} = \{G, PG, \ldots\}$
$OP_{PlayStation} = \{A3, A7, A12, BuyGames, \ldots\}$
$OP_{Oven} = \{ON, OFF\}$
$OP_{Fridge} = \{Open, Close\}$
$OP_{FrontDoor} = \{Lock, Unlock\}$
$OP = OP_{TV} \cup OP_{PlayStation} \cup OP_{Oven} \cup OP_{Fridge} \cup OP_{FrontDoor}$
$OPA = \{KidsFriendly\}$
$KidsFriendly : op : OP \rightarrow \{True, False\}$
$KidsFriendly(G) = KidsFriendly(A3) = KidsFriendly(A7) = True$
$KidsFriendly(PG) = KidsFriendly(A12) = KidsFriendly(BuyGames) = False$

# Constructing EGRBAC From HABAC



$$Authorization_{op}(s:S, es:ES, d:D) \equiv$$
$$(Relationship(s) = kid \wedge ((day(current) \in \{Sa, S\} \wedge 12:00 \leq time(current) \leq 19:00) \vee (day(current) \in \{M, T, W, Th, F\} \wedge 17:00 \leq time(current) \leq 19:00)) \wedge KidsFriendly(op) = True \vee$$
$$(Relationship(s) = teenager \wedge ParentInKitchen(current) = True \wedge DangerouseKitchenDevices(d) = True) \vee$$
$$(Relationship(s) = teenager \wedge DangerouseKitchenDevices(d) = False) \vee$$
$$(Relationship(s) = teenager \wedge (KidsFriendly(op) = True \vee KidsFriendly(op) = False)) \vee$$
$$(Relationship(s) = parent)$$

# Constructing EGRBAC From HABAC
## 1. From Authorization Policy to Authorization Array



- First, we convert the authorization policy into a disjunctive normal form (DNF)

$$Authorization_{op}(s:S, es:ES, d:D) \equiv$$
$$(Relationship(s) = kid \wedge ((day(current) \in \{Sa, S\} \wedge 12:00 \leq time(current) \leq 19:00) \vee (day(current) \in \{M, T, W, Th, F\} \wedge 17:00 \leq time(current) \leq 19:00)) \wedge KidsFriendly(op) = True \vee$$
$$(Relationship(s) = teenager \wedge ParentInKitchen(current) = True \wedge DangerouseKitchenDevices(d) = True) \vee$$
$$(Relationship(s) = teenager \wedge DangerouseKitchenDevices(d) = False) \vee$$
$$(Relationship(s) = teenager \wedge (KidsFriendly(op) = True \vee KidsFriendly(op) = False)) \vee$$
$$(Relationship(s) = parent)$$

$$Authorization_{op}(s:S, es:ES, d:D) \equiv$$
$$(Relationship(s) = kid \wedge day(current) \in \{Sa, S\} \wedge 12:00 \leq time(current) \leq 19:00 \wedge KidsFriendly(op) = True) \vee$$
$$(Relationship(s) = kid \wedge day(current) \in \{M, T, W, Th, F\} \wedge 17:00 \leq time(current) \leq 19:00 \wedge KidsFriendly(op) = True) \vee$$
$$(Relationship(s) = teenager \wedge ParentInKitchen(current) = True \wedge DangerouseKitchenDevices(d) = True) \vee$$
$$(Relationship(s) = teenager \wedge DangerouseKitchenDevices(d) = False) \vee$$
$$(Relationship(s) = teenager \wedge KidsFriendly(op) = False) \vee$$
$$(Relationship(s) = teenager \wedge KidsFriendly(op) = True) \vee$$
$$(Relationship(s) = parent).$$

# Constructing EGRBAC From HABAC
## 1. From Authorization Policy to Authorization Array



$$
\begin{aligned}
&Authorization_{op}(s:S, es:ES, d:D) \equiv \\
&(Relationship(s) = kid \wedge\ day(current) \in \{Sa, S\} \wedge 12:00 \leq time(current) \leq 19:00 \wedge KidsFriendly(op) = True) \vee \\
&(Relationship(s) = kid \wedge\ day(current) \in \{M, T, W, Th, F\} \wedge 17:00 \leq time(current) \leq 19:00 \wedge KidsFriendly(op) = True) \vee \\
&(Relationship(s) = teenager \wedge ParentInKitchen(current) = True \wedge DangerouseKitchenDevices(d) = True) \vee \\
&(Relationship(s) = teenager \wedge DangerouseKitchenDevices(d) = False) \vee \\
&(Relationship(s) = teenager \wedge KidsFriendly(op) = False) \vee \\
&(Relationship(s) = teenager \wedge KidsFriendly(op) = True) \vee \\
&(Relationship(s) = parent).
\end{aligned}
$$

- The Next step, is to construct what we call an authorization array AA, from this DNF statement.
- To construct the authorization array, we evaluate every $u_i \in U$, $d_j \in D$, and $op_k \in OP$ combination against each conjunctive clause.
- Whenever a combination satisfies every term (condition) in a conjunctive clause except those conditions which involve environment state attributes, we create a raw $(\boldsymbol{u_i}, \boldsymbol{d_j}, \boldsymbol{op_k}, \boldsymbol{current}, \boldsymbol{C})$ for that combination in the authorization array.
- Where, $C$ is the set of session and environment related conditions in the examined conjunctive clause.

# Constructing EGRBAC From HABAC
## 1. From Authorization Policy to Authorization Array



| User u | Device d | Operation op | Environment state es | Conditions C |
|---|---|---|---|---|
| alex | TV | *G* | current | X |
| alex | PS | *A3* | current | X |
| alex | PS | *A7* | current | X |
| alex | TV | *G* | current | Z |
| alex | PS | *A3* | current | Z |
| alex | PS | *A7* | current | Z |
| bob | TV | *G* | current | $\{Relationship(s) = parent\}$ |
| bob | TV | *PG* | current | $\{Relationship(s) = parent\}$ |
| bob | PS | *A3* | current | $\{Relationship(s) = parent\}$ |
| bob | PS | *A7* | current | $\{Relationship(s) = parent\}$ |
| bob | PS | *A12* | current | $\{Relationship(s) = parent\}$ |
| bob | PS | *BuyGames* | current | $\{Relationship(s) = parent\}$ |
| bob | Oven | *ON* | current | $\{Relationship(s) = parent\}$ |
| bob | Oven | *OFF* | current | $\{Relationship(s) = parent\}$ |
| bob | Fridge | *Open* | current | $\{Relationship(s) = parent\}$ |
| bob | Fridge | *Close* | current | $\{Relationship(s) = parent\}$ |
| bob | FrontDoor | *Lock* | current | $\{Relationship(s) = parent\}$ |
| bob | FrontDoor | *Unlock* | current | $\{Relationship(s) = parent\}$ |
| anne | TV | *G* | current | $\{Relationship(s) = teenager\}$ |
| anne | TV | *PG* | current | $\{Relationship(s) = teenager\}$ |
| anne | PS | *A3* | current | $\{Relationship(s) = teenager\}$ |
| anne | PS | *A7* | current | $\{Relationship(s) = teenager\}$ |
| anne | PS | *A12* | current | $\{Relationship(s) = teenager\}$ |
| anne | PS | *BuyGames* | current | $\{Relationship(s) = teenager\}$ |
| anne | Oven | *ON* | current | Y |
| anne | Oven | *OFF* | current | Y |
| anne | Fridge | *OPEN* | current | $\{Relationship(s) = teenager\}$ |
| anne | Fridge | *CLOSE* | current | $\{Relationship(s) = teenager\}$ |

$$X = \{Relationship(s) = kid\ ,\ day(current) \in \{Sa, S\}, 12:00 \leq time(current) \leq 19:00\}.$$

$$Y = \{Relationship(s) = teenager\ ,\ ParentInKitchen(current) = True\}.$$

$$Z = \{Relationship(s) = kid\ ,\ day(current) \in \{M, T, W, Th, F\}, 17:00 \leq time(current) \leq 19:00\}.$$

# Constructing EGRBAC From HABAC
## 2. From Authorization Array to EGRBAC Approach



- **Input:** HABAC set of users $U_{HABAC}$, set of devices $D_{HABAC}$ , set of operations $OP_{HABAC}, UA, SA, ESA, OPA, DA,$ and the **authorization array AA**.

- **Output:** EGRBAC components $U$, $R$, $UA$, $EC$, $ER$, $EA$, $RP$, $RPRA$, $RPEA$, $D$, $OP$, $P$, $DR$, $PDRA$, and $RPDRA$.

- **Steps:**

**Step 1:** Initialization.

- The set of users, devices, and operations are the same in both systems, hence $U = U_{HABAC}, D = D_{HABAC}, and\ OP = OP_{HABAC}$.
- For every operation $op_i$ , and device $d_j$ pair, where $op_i$ is assigned to $d_j$ by the device manufacturers, create a permission $(d_j , op_i)$.

Step 2: Create the set of **device roles**.

a. Create a device role $dr$ for each operation attribute instance, or device attribute instance.
b. Create one device role call it remaining permissions $RemPerm$ for all the permissions $p_l = (d_i, op_j)$, where $d_i$ is not assigned to any device attributes, and $op_j$ is not assigned to any operation attribute.

Step 3: Construct the permission device role assignment array $PDRA$.

- $PDRA \subseteq P \times DR$, a many-to-many mapping of permissions and DR.

# Constructing EGRBAC From HABAC
## 2. From Authorization Array to EGRBAC Approach



| | DangerouseKitchenDevices = True | DangerouseKitchenDevices = False | KidsFriendly = True | KidsFriendly = False | RemPerm |
|---|---|---|---|---|---|
| $(TV, G)$ | 0 | 0 | 1 | 0 | 0 |
| $(TV, PG)$ | 0 | 0 | 0 | 1 | 0 |
| $(PlayStation, A3)$ | 0 | 0 | 1 | 0 | 0 |
| $(PlayStation, A7)$ | 0 | 0 | 1 | 0 | 0 |
| $(PlayStation, A12)$ | 0 | 0 | 0 | 1 | 0 |
| $(PlayStation, BuyGames)$ | 0 | 0 | 0 | 1 | 0 |
| $(Oven, ON)$ | 1 | 0 | 0 | 0 | 0 |
| $(Oven, OFF)$ | 1 | 0 | 0 | 0 | 0 |
| $(Fridge, Open)$ | 0 | 1 | 0 | 0 | 0 |
| $(Fridge, Close)$ | 0 | 1 | 0 | 0 | 0 |
| $(FrontDoor, Lock)$ | 0 | 0 | 0 | 0 | 1 |
| $(FrontDoor, Unlock)$ | 0 | 0 | 0 | 0 | 1 |

- For every $PDRA[i, j] = 1$, add the pair $(p_i, dr_j)$ to the set $PDRA$ of EGRBAC.

Step 4: Construct the user device role authorization array $UDRAA$ from the **authorization array $AA$**, and the permission device role assignment array $PDRA$.

- $UDRAA \subseteq U \times DR$, a many-to-many mapping between **users** and **device roles**

# Constructing EGRBAC From HABAC
## 2. From Authorization Array to EGRBAC Approach



| User u | Device d | Operation op | Environment state es | Conditions C |
|---|---|---|---|---|
| alex | TV | *G* | current | X |
| alex | PS | *A3* | current | X |
| alex | PS | *A7* | current | X |
| alex | TV | *G* | current | Z |
| alex | PS | *A3* | current | Z |
| alex | PS | *A7* | current | Z |
| bob | TV | *G* | current | $\{Relationship(s) = parent\}$ |
| bob | TV | *PG* | current | $\{Relationship(s) = parent\}$ |
| bob | PS | *A3* | current | $\{Relationship(s) = parent\}$ |
| bob | PS | *A7* | current | $\{Relationship(s) = parent\}$ |
| bob | PS | *A12* | current | $\{Relationship(s) = parent\}$ |
| bob | PS | *BuyGames* | current | $\{Relationship(s) = parent\}$ |
| bob | Oven | *ON* | current | $\{Relationship(s) = parent\}$ |
| bob | Oven | *OFF* | current | $\{Relationship(s) = parent\}$ |
| bob | Fridge | *Open* | current | $\{Relationship(s) = parent\}$ |
| bob | Fridge | *Close* | current | $\{Relationship(s) = parent\}$ |
| bob | FrontDoor | *Lock* | current | $\{Relationship(s) = parent\}$ |
| bob | FrontDoor | *Unlock* | current | $\{Relationship(s) = parent\}$ |
| anne | TV | *G* | current | $\{Relationship(s) = teenager\}$ |
| anne | TV | *PG* | current | $\{Relationship(s) = teenager\}$ |
| anne | PS | *A3* | current | $\{Relationship(s) = teenager\}$ |
| anne | PS | *A7* | current | $\{Relationship(s) = teenager\}$ |
| anne | PS | *A12* | current | $\{Relationship(s) = teenager\}$ |
| anne | PS | *BuyGames* | current | $\{Relationship(s) = teenager\}$ |
| anne | Oven | *ON* | current | Y |
| anne | Oven | *OFF* | current | Y |
| anne | Fridge | *OPEN* | current | $\{Relationship(s) = teenager\}$ |
| anne | Fridge | *CLOSE* | current | $\{Relationship(s) = teenager\}$ |

$$X = \{Relationship(s) = kid\ ,\ day(current) \in \{Sa, S\}, 12:00 \leq time(current) \leq 19:00\}.$$
$$Y = \{Relationship(s) = teenager\ ,\ ParentInKitchen(current) = True\}.$$
$$Z = \{Relationship(s) = kid\ ,\ day(current) \in \{M, T, W, Th, F\}, 17:00 \leq time(current) \leq 19:00\}.$$

$$PDRA \subseteq P \times DR$$

| | DangerouseKitchenDevices = True | DangerouseKitchenDevices = False | KidsFriendly = True | KidsFriendly = False | RemPerm |
|---|---|---|---|---|---|
| alex | 0 | 0 | $\{X, Z\}$ | 0 | 0 |
| bob | $\{\{Relationship(s) = parent\}\}$ | $\{\{Relationship(s) = parent\}\}$ | $\{\{Relationship(s) = parent\}\}$ | $\{\{Relationship(s) = parent\}\}$ | $\{\{Relationship(s) = parent\}\}$ |
| anne | $\{Y\}$ | $\{\{Relationship(s) = teenager\}\}$ | $\{\{Relationship(s) = teenager\}\}$ | $\{\{Relationship(s) = teenager\}\}$ | 0 |

$$X = \{Relationship(s) = kid\ ,\ day(current) \in \{Sa, S\}, 12:00 \leq time(current) \leq 19:00\}.$$
$$Y = \{Relationship(s) = teenager\ ,\ ParentInKitchen(current) = True\}.$$
$$Z = \{Relationship(s) = kid\ ,\ day(current) \in \{M, T, W, Th, F\}, 17:00 \leq time(current) \leq 19:00\}.$$

# Constructing EGRBAC From HABAC
## 2. From Authorization Array to EGRBAC Approach



Step 5: Construct the rest of EGRBAC elements ($R$, $EC$, $ER$, $RP$), assignments ($UA$, $EA$, $RPDRA$), and associations ($RPRA$, $RPEA$) by following our proposed EGRBAC users and environment roles constructing Algorithm.

| | DangerouseKitchenDevices = True | DangerouseKitchenDevices = False | KidsFriendly = True | KidsFriendly = False | RemPerm |
|---|---|---|---|---|---|
| alex | 0 | 0 | {X, Z} | 0 | 0 |
| bob | {{Relationship(s) = parent}} | {{Relationship(s) = parent}} | {{Relationship(s) = parent}} | {{Relationship(s) = parent}} | {{Relationship(s) = parent}} |
| anne | {Y} | {{Relationship(s) = teenager}} | {{Relationship(s) = teenager}} | {{Relationship(s) = teenager}} | 0 |

$$X = \{Relationship(s) = kid\ ,\ day(current) \in \{Sa, S\}, 12:00 \leq time(current) \leq 19:00\}.$$

$$Y = \{Relationship(s) = teenager\ ,\ ParentInKitchen(current) = True\}.$$

$$Z = \{Relationship(s) = kid\ ,\ day(current) \in \{M, T, W, Th, F\}, 17:00 \leq time(current) \leq 19:00\}.$$

- The main idea is to loop through the columns of UDRAA, each column is corresponding to different user's access rights to a specific device role.
- Inside each columns loop through the fields of different rows to extract and construct different EGRBAC components.

For example, in this case, the algorithm do the following:

a. Create a user role $r_m$ which corresponds to accessing this column device role when Y satisfied.
b. Create an environment role $er_x$, and an environment condition $ec_x$ and assign them to each other. These environment role, and environment condition correspond to the environment attribute ($ParentInKiitchen$) .
c. Define a role pair $rp_z$, where $rp_z.r = r_m$ , and $rp_z.ER = \{er_x\}$ .
d. Assign the role pair $rp_z$ to the device role corresponding to this column.
e. Finally, assign the role $r_m$ to the user corresponding to this raw.

| | $DangerouseKitchenDevices = True$ |
|---|---|
| anne | {Y} |

$$Y = \{Relationship(s) = teenager\ ,\ ParentInKitchen(current) = True\}\ .$$

# Constructing EGRBAC From HABAC
## 2. From Authorization Array to EGRBAC Approach



- The algorithm do the same for all the entries in the UDRAA.
- By the end of this algorithm, all the EGRBAC elements (U, R, EC, ER, RP, D, OP, P, DR), assignments (UA, EA, PDRA, RPDRA), and associations (RPRA, RPEA) will be constructed.

Step 6: Finally, we merge similar roles.

| | DangerouseKitchenDevices = True | DangerouseKitchenDevices = False | KidsFriendly = True | KidsFriendly = False | RemPerm |
|---|---|---|---|---|---|
| alex | 0 | 0 | $\{X, Z\}$ | 0 | 0 |
| bob | $\{\{Relationship(s) = parent\}\}$ | $\{\{Relationship(s) = parent\}\}$ | $\{\{Relationship(s) = parent\}\}$ | $\{\{Relationship(s) = parent\}\}$ | $\{\{Relationship(s) = parent\}\}$ |
| anne | $\{Y\}$ | $\{\{Relationship(s) = teenager\}\}$ | $\{\{Relationship(s) = teenager\}\}$ | $\{\{Relationship(s) = teenager\}\}$ | 0 |

$$X = \{Relationship(s) = kid\ ,\ day(current) \in \{Sa, S\}, 12:00 \leq time(current) \leq 19:00\}.$$

$$Y = \{Relationship(s) = teenager\ ,\ ParentInKitchen(current) = True\}.$$

$$Z = \{Relationship(s) = kid\ ,\ day(current) \in \{M, T, W, Th, F\}, 17:00 \leq time(current) \leq 19:00\}.$$

# Our Contribution



1. Design, formalize ABAC based access control model for smart home IoT (HABAC).

2. Analyze HABAC model relative to the previously published EGRBAC model [4]. Compare the theoretical expressive power of these models by:

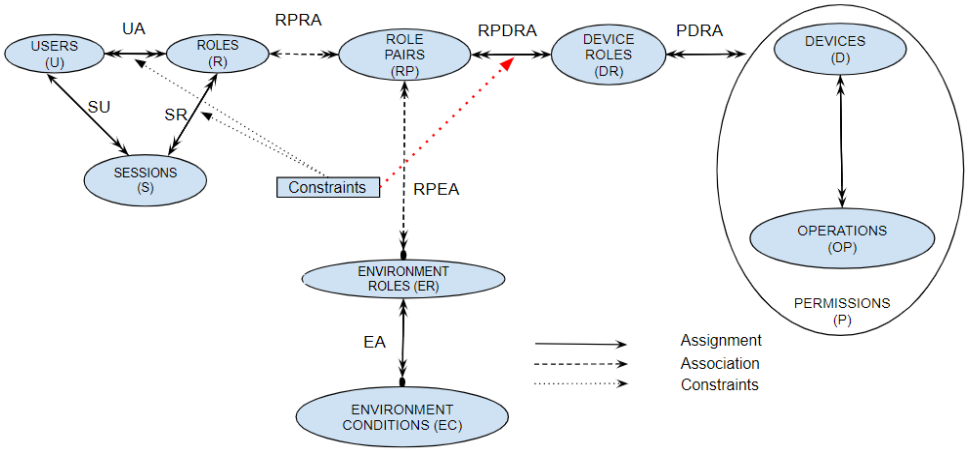
   a. Introduce HABAC configuration that translates EGRBAC policies in a manner that they can be implemented by HABAC.

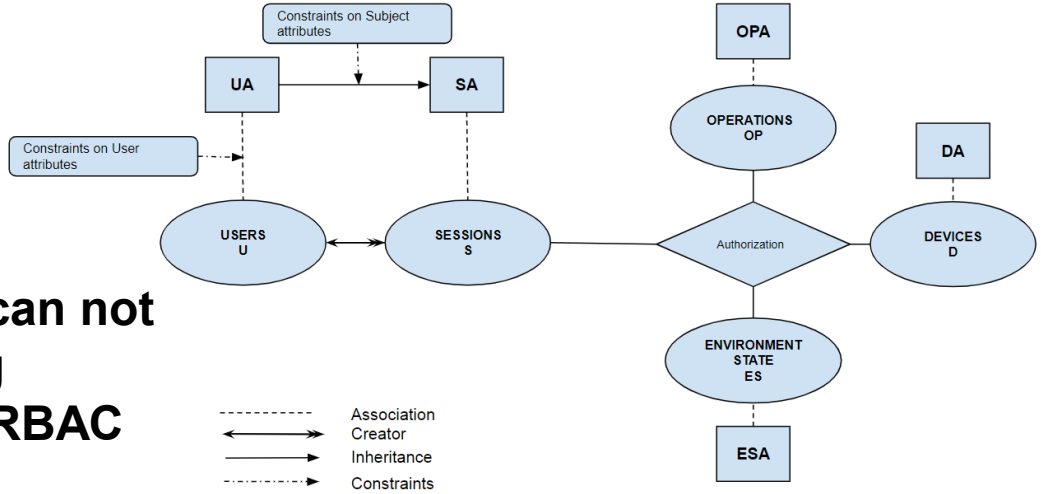
   b. Provide an algorithm to convert an HABAC specification to EGRBAC.

   c. Discuss the insights for practical deployment of these models resulting from these constructions.

# HABAC Vs EGRBAC





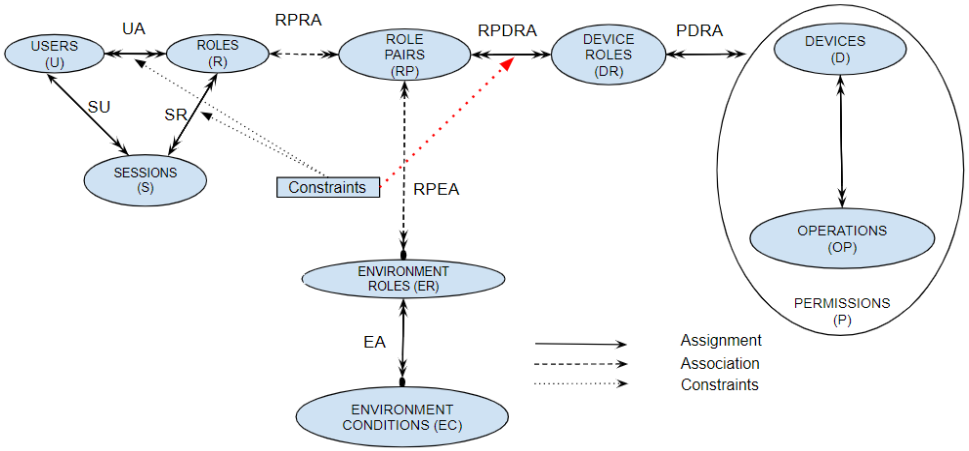




**1- In HABAC we can not create something equivalent to EGRBAC PRConstraints.**

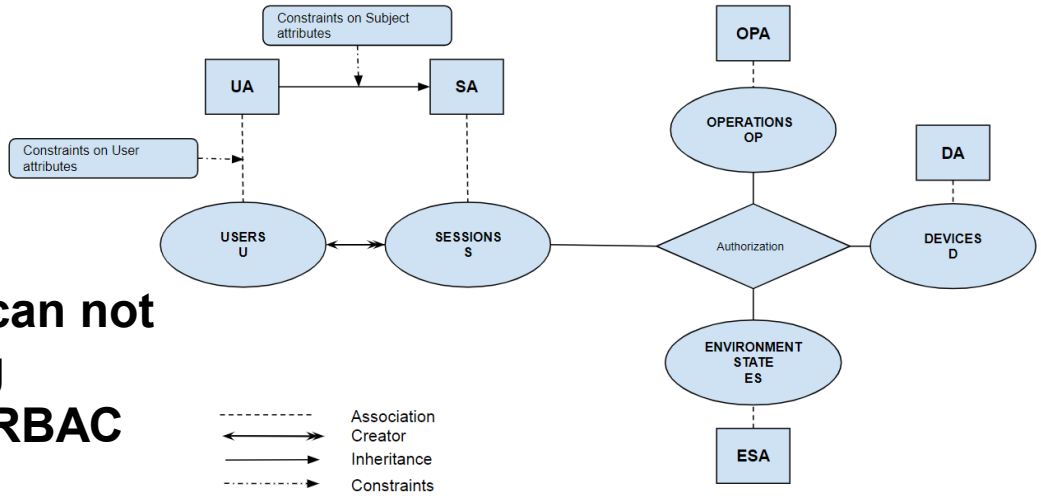
# HABAC Vs EGRBAC



**2- In EGRBAC it is easy to define who has what permissions, and who is not allowed to have a future access to specific permissions.**



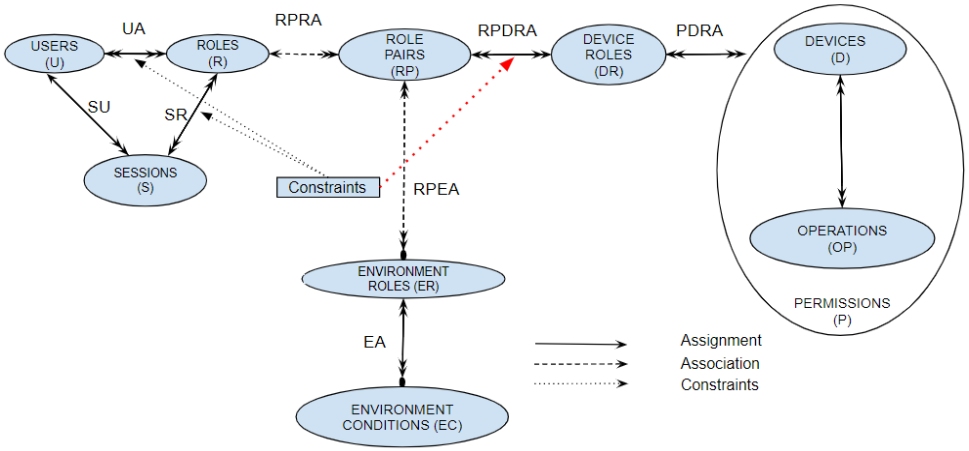

**1- In HABAC we can not create something equivalent to EGRBAC PRConstraints.**



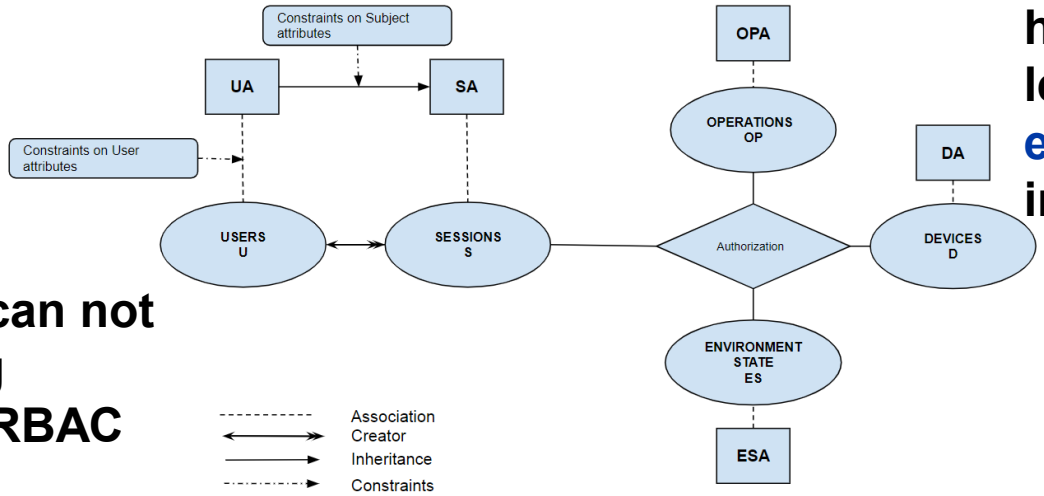
# HABAC Vs EGRBAC



**2- In EGRBAC it is easy to define who has what permissions, and who is not allowed to have a future access to specific permissions.**




**3- in EGRBAC, we can't handle HABAC policies that involve users, devices and operations dynamic attributes. Such handling may lead to role explosion in EGRBAC.**




**1- In HABAC we can not create something equivalent to EGRBAC PRConstraints.**

# Conclusion



- We introduce HABAC access control model for smart home IoT.
- It is a dynamic, fine grained ABAC based model that captures different attributes of users, environment, operations, and devices.
- We provide a use case scenario demonstration. Moreover, we compare the theoretical expressive power of our model to EGRBAC which is a dynamic contextual aware RBAC based access control model. We do that by providing approaches for converting an HABAC specification to EGRBAC and vice versa.
- In conclusion, we believe that a hybrid model retaining HABAC and EGRBAC features may be the most suitable for smart home IoT, and likely more generally.


Q&A

# Thank You