MODULE *AccessControlManagement*

EXTENDS *Naturals*, *Sequences*

CONSTANTS *Processes*, *Resources*

$NULL \triangleq$ "NULL"
$ALLOWED \triangleq$ "ALLOWED"
$REJECTED \triangleq$ "REJECTED"
$REQUESTED \triangleq$ "REQUESTED"
$IN\_USE \triangleq$ "IN_USE"

$Boolean \triangleq \{\text{TRUE}, \text{FALSE}\}$
$ResourceStatus \triangleq \{NULL, REQUESTED, ALLOWED, REJECTED, IN\_USE\}$

```
** --algorithm AccessControlManagement
{
    variables Acl = [a ∈ Processes ↦ [r ∈ Resources ↦ NULL]] ;
              Acl2 = [a ∈ Processes ↦ [r ∈ Resources ↦ NULL]] ;

    macro Request( p, r )
    {
     if ( Acl[p][r] = NULL )
      Acl[p][r] := REQUESTED ;
     }

    macro Decide( p, r )
    {
     if ( Acl[p][r] = REQUESTED )
      with ( b ∈ Boolean )
      {
       if ( b = TRUE )
        Acl[p][r] := ALLOWED ;
       else
        Acl[p][r] := REJECTED ;
      }
     }

    macro Revoke( p, r )
    {
     if ( Acl[p][r] = ALLOWED )
      Acl[p][r] := NULL ;
     }

    macro Use( p, r )
    {
     if ( Acl[p][r] = ALLOWED )
      Acl[p][r] := IN_USE ;
     }
```

```
  fair process ( AcmNext ∈ Processes )
  variable Resource = 1 ;
  {
      s0: while ( TRUE )
      {
       s1: Resource := 1 ;
       s2: Acl2 := Acl ;

       either { a: Request(self, Resource) ; }
       or { b: Decide(self, Resource)     ; }
       or { c: Revoke(self, Resource)     ; }
       or { d: Use(self, Resource) ; } ;

       N: Resource := Resource + 1 ;

       if ( Resource ∈ Resources )
        goto s1 ;
       }
   }
}
```

**

BEGIN TRANSLATION ($chksum(pcal) =$ "815326e1" $\land\ chksum(tla) =$ "5b152e55")

VARIABLES Acl, $Acl2$, pc, $Resource$

$vars \triangleq \langle Acl,\ Acl2,\ pc,\ Resource \rangle$

$ProcSet \triangleq (Processes)$

$Init \triangleq$ Global variables
$\land\ Acl = [a \in Processes \mapsto [r \in Resources \mapsto NULL]]$
$\land\ Acl2 = [a \in Processes \mapsto [r \in Resources \mapsto NULL]]$
Process $AcmNext$
$\land\ Resource = [self \in Processes \mapsto 1]$
$\land\ pc = [self \in ProcSet \mapsto$ "s0"$]$

$s0(self) \triangleq \land\ pc[self] =$ "s0"
$\land\ pc' = [pc$ EXCEPT $![self] =$ "s1"$]$
$\land$ UNCHANGED $\langle Acl,\ Acl2,\ Resource \rangle$

$s1(self) \triangleq \land\ pc[self] =$ "s1"
$\land\ Resource' = [Resource$ EXCEPT $![self] = 1]$
$\land\ pc' = [pc$ EXCEPT $![self] =$ "s2"$]$
$\land$ UNCHANGED $\langle Acl,\ Acl2 \rangle$

$s2(self) \triangleq \land\ pc[self] =$ "s2"
$\land\ Acl2' = Acl$
$\land\ \lor\ \land\ pc' = [pc$ EXCEPT $![self] =$ "a"$]$

$$
\begin{array}{l}
\quad \lor \land pc' = [pc \text{ EXCEPT } ![self] = \text{“b”}] \\
\quad \lor \land pc' = [pc \text{ EXCEPT } ![self] = \text{“c”}] \\
\quad \lor \land pc' = [pc \text{ EXCEPT } ![self] = \text{“d”}] \\
\land \text{UNCHANGED } \langle Acl,\ Resource \rangle
\end{array}
$$

$$
\begin{array}{lll}
a(self) & \triangleq & \land pc[self] = \text{“a”} \\
& & \land \text{IF } Acl[self][Resource[self]] = NULL \\
& & \quad \text{THEN } \land Acl' = [Acl \text{ EXCEPT } ![self][Resource[self]] = REQUESTED] \\
& & \quad \text{ELSE } \land \text{TRUE} \\
& & \qquad\ \ \land Acl' = Acl \\
& & \land pc' = [pc \text{ EXCEPT } ![self] = \text{“N”}] \\
& & \land \text{UNCHANGED } \langle Acl2,\ Resource \rangle
\end{array}
$$

$$
\begin{array}{lll}
b(self) & \triangleq & \land pc[self] = \text{“b”} \\
& & \land \text{IF } Acl[self][Resource[self]] = REQUESTED \\
& & \quad \text{THEN } \land \exists\, b \in Boolean : \\
& & \qquad\ \ \text{IF } b = \text{TRUE} \\
& & \qquad\quad \text{THEN } \land Acl' = [Acl \text{ EXCEPT } ![self][Resource[self]] = ALLOWED] \\
& & \qquad\quad \text{ELSE } \land Acl' = [Acl \text{ EXCEPT } ![self][Resource[self]] = REJECTED] \\
& & \quad \text{ELSE } \land \text{TRUE} \\
& & \qquad\ \ \land Acl' = Acl \\
& & \land pc' = [pc \text{ EXCEPT } ![self] = \text{“N”}] \\
& & \land \text{UNCHANGED } \langle Acl2,\ Resource \rangle
\end{array}
$$

$$
\begin{array}{lll}
c(self) & \triangleq & \land pc[self] = \text{“c”} \\
& & \land \text{IF } Acl[self][Resource[self]] = ALLOWED \\
& & \quad \text{THEN } \land Acl' = [Acl \text{ EXCEPT } ![self][Resource[self]] = NULL] \\
& & \quad \text{ELSE } \land \text{TRUE} \\
& & \qquad\ \ \land Acl' = Acl \\
& & \land pc' = [pc \text{ EXCEPT } ![self] = \text{“N”}] \\
& & \land \text{UNCHANGED } \langle Acl2,\ Resource \rangle
\end{array}
$$

$$
\begin{array}{lll}
d(self) & \triangleq & \land pc[self] = \text{“d”} \\
& & \land \text{IF } Acl[self][Resource[self]] = ALLOWED \\
& & \quad \text{THEN } \land Acl' = [Acl \text{ EXCEPT } ![self][Resource[self]] = IN\_USE] \\
& & \quad \text{ELSE } \land \text{TRUE} \\
& & \qquad\ \ \land Acl' = Acl \\
& & \land pc' = [pc \text{ EXCEPT } ![self] = \text{“N”}] \\
& & \land \text{UNCHANGED } \langle Acl2,\ Resource \rangle
\end{array}
$$

$$
\begin{array}{lll}
N(self) & \triangleq & \land pc[self] = \text{“N”} \\
& & \land Resource' = [Resource \text{ EXCEPT } ![self] = Resource[self] + 1] \\
& & \land \text{IF } Resource'[self] \in Resources \\
& & \quad \text{THEN } \land pc' = [pc \text{ EXCEPT } ![self] = \text{“s1”}] \\
& & \quad \text{ELSE } \land pc' = [pc \text{ EXCEPT } ![self] = \text{“s0”}] \\
& & \land \text{UNCHANGED } \langle Acl,\ Acl2 \rangle
\end{array}
$$

$$AcmNext(self) \triangleq s0(self) \lor s1(self) \lor s2(self) \lor a(self) \lor b(self) \lor c(self) \lor d(self) \lor N(self)$$

$$Next \triangleq (\exists\, self \in Processes : AcmNext(self))$$

$$Spec \triangleq \land Init \land \Box[Next]_{vars} \land \forall\, self \in Processes : \mathrm{WF}_{vars}(AcmNext(self))$$

END TRANSLATION

$$AcmTypeOK \triangleq \land Acl \in [Processes \to [Resources \to ResourceStatus]] \land Acl2 \in [Processes \to [Resources \to ResourceStatus]]$$

$$AcmConsistent \triangleq \neg(\exists\, p \in Processes : \exists\, r \in Resources : Acl[p][r] = IN\_USE \land Acl2[p][r] \neq ALLOWED \land Acl2[p][r] \neq IN\_USE)$$

$$AcmLiveness \triangleq \Diamond(\exists\, p \in Processes : \exists\, r \in Resources : Acl[p][r] = REQUESTED \rightsquigarrow Acl[p][r] = ALLOWED \lor Acl[p][r] = REJECTED)$$

* Modification History
* Last modified *Fri* May 26 14:23:17 *GMT* + 03:30 2023 by *Amirhosein*
* Created *Thu Mar* 23 07:45:26 *GMT* + 03:30 2023 by *Amirhosein*