

Modélisation de la disponibilité

Frédéric Cuppens – Nora Cuppens-Boulahia
ENST Bretagne
{ frederic,nora }.cuppens@enst-bretagne.fr

Définition de la disponibilité

On peut définir la propriété de disponibilité comme *l'aptitude d'un système à répondre à une demande d'un service, d'une ressource en garantissant des contraintes d'horaires, de délai et de performances.*

Politique de disponibilité

La spécification des contraintes qui doivent être garanties pour assurer la disponibilité constitue une politique de disponibilité. Ces contraintes peuvent dépendre :

- Du sujet ayant fait la demande.
- Du service demandé.
- Des ressources demandées.
- Du contexte de la demande.
- De l'organisation à laquelle le sujet demandeur de ressources appartient ou celle dont il sollicite les ressources.

Par exemple, dans un contexte médical, les contraintes à respecter peuvent être différentes suivant que le sujet demandeur est un médecin urgentiste ou bien une secrétaire médicale. De même, les contraintes peuvent dépendre du contexte dans lequel s'effectue la demande, par exemple un contexte de consultation d'un patient ou un contexte d'une opération chirurgicale urgente. Les contraintes peuvent également dépendre de l'organisation dans laquelle la demande a été effectuée, par exemple une clinique ou un hôpital.

Au besoin, on pourra structurer la politique de disponibilité en raisonnant sur le rôle que joue le sujet plutôt que sur le sujet lui-même. On pourra également structurer les concepts de service et de ressource.

Différence entre permissions et droits

Habituellement, une politique de contrôle d'accès exprime un ensemble de permissions. Ces permissions spécifient les services que les sujets peuvent réaliser sur les ressources et dans quels contextes ces permissions sont accordées. Si, dans un certain contexte, un sujet demande à réaliser un service nécessitant certaines ressources, alors le contrôleur d'accès accepte cette demande si la politique de contrôle d'accès spécifie que ce sujet a effectivement la permission correspondante. Sinon, la demande est rejetée.

Cependant, si la demande est acceptée, rien ne garantit que le sujet pourra effectivement réaliser le service en temps borné voir même en temps fini.

Nous proposons donc de distinguer la notion de permission de celle de **droit**. Un droit implique une permission avec en plus une obligation pour le système de satisfaire la demande

en respectant des contraintes temporelles. On dira que le droit est *simple* si l'obligation est de réaliser le service en temps fini. Le droit sera dit *contraint* si l'obligation est de réaliser le service en temps borné.

Une politique de disponibilité inclura donc la spécification d'un ensemble de droits (simple ou borné).

Environnement d'indisponibilité

L'objectif du contrôle d'accès est de démontrer que seules les demandes permises sont effectivement réalisées dans le système.

Cela ne suffit pas pour assurer la disponibilité. Il faut également démontrer que le système remplit ses obligations lorsqu'un sujet active un droit simple ou contraint.

Nous pensons qu'une telle démonstration nécessite un modèle d'indisponibilité.

Dans le cas de fautes accidentelles, ce modèle doit par exemple spécifier les probabilités de défaillance des composants des systèmes ainsi que d'éventuelles hypothèses d'indépendance de ces défaillances. Ce type de modèle est déjà bien étudié dans le domaine de la sûreté de fonctionnement.

Dans le cas de fautes malveillantes (attaques ou intrusions), un tel modèle d'indisponibilité doit préciser les possibilités d'attaques en déni de services contre les composants du système et les conséquences de ces attaques sur ces composants. L'objectif n'est pas de modéliser des attaques particulières mais les conséquences de ces attaques telles que :

- Défaillance définitive d'un composant. Le composant n'est plus utilisable et doit être changé ou réparé,
- Défaillance durable d'un composant. Le composant reste inactif tant qu'une intervention extérieure n'a pas lieu,
- Défaillance temporaire d'un composant. Le composant retrouve un fonctionnement normal dès que l'attaque cesse.

Le modèle d'indisponibilité pourra également préciser les conséquences de certaines attaques sur les communications, par exemple communication impossible entre certains composants ou bien communication seulement retardée ou ralentie.

Modélisation des propriétés de disponibilité

Les propriétés de disponibilité à prouver sont dérivées de la politique de disponibilité. Elles permettent de prouver que le système garantit *la continuité de service*, c'est-à-dire il a la capacité de fournir des services aux utilisateurs en respectant leurs droits simples ou contraints.

La disponibilité correspond donc à une *liveness property*, c'est-à-dire une propriété qui garantit que le service sera rendu dans le respect des conditions temporelles spécifiées dans les droits.

Preuve des propriétés de disponibilité

La preuve consiste à montrer qu'il est possible de dériver les propriétés de disponibilité de la spécification du système compte tenu d'un *environnement d'indisponibilité*. Cet environnement doit spécifier quels sont les types d'attaques élémentaires envisageables ainsi que les combinaisons possibles de ces attaques élémentaires. L'environnement d'indisponibilité joue le rôle d'hypothèses. Ce sont sous ces hypothèses que l'on essaye de prouver les propriétés de disponibilité. Par exemple, le « cas pire » est un exemple d'hypothèse qu'il s'agira de spécifier sous forme d'un certain environnement d'indisponibilité.

Cohérence d'une politique de disponibilité

Comme toute politique de sécurité, une politique de disponibilité doit être cohérente. Intuitivement, il s'agit de montrer que l'ensemble des règles constituant la politique de disponibilité ne contient pas d'exigence conflictuelle, c'est-à-dire qu'il serait impossible de satisfaire simultanément.