Chap. II : Politiques et modèles de sécurité

Laurent Poinsot

UMR 7030 - Université Paris 13 - Institut Galilée

Cours " Sécrypt "

Dans un système informatique, l'autorisation a pour but de ne permettre que les actions légitimes, c'est-à-dire à empêcher qu'un utilisateur puisse exécuter des opérations qui ne devraient pas lui être permises. Pour définir quelles sont les opérations autorisées et celles qui sont interdites, il faut établir une politique de sécurité. Le standard européen des ITSEC (Information Technology Security Evaluation Criteria) définissent une politique de sécurité comme étant " l'ensemble des lois, règles et pratiques qui régissent la façon dont l'information sensible et les autres ressources sont gérées, protégées et distribuées à l'intérieur d'un système spécifique ".

Dans un système informatique, l'autorisation a pour but de ne permettre que les actions légitimes, c'est-à-dire à empêcher qu'un utilisateur puisse exécuter des opérations qui ne devraient pas lui être permises. Pour définir quelles sont les opérations autorisées et celles qui sont interdites, il faut établir une politique de sécurité. Le standard européen des ITSEC (Information Technology Security Evaluation Criteria) définissent une politique de sécurité comme étant " l'ensemble des lois, règles et pratiques qui régissent la façon dont l'information sensible et les autres ressources sont gérées, protégées et distribuées à l'intérieur d'un système spécifique ".

Pour construire une politique de sécurité il faut :

- d'une part, définir un ensemble de propriétés de sécurité qui doivent être satisfaites par le système. Par exemple "une information classifiée ne doit pas être transmise à un utilisateur non habilité à la connaître" ;
- d'autre part, établir un schéma d'autorisation, qui présente les règles permettant de modifier l'état de protection du système. Par exemple "le propriétaire d'une information peut accorder un droit d'accès pour cette information à n'importe quel utilisateur".

Pour construire une politique de sécurité il faut :

- d'une part, définir un ensemble de propriétés de sécurité qui doivent être satisfaites par le système. Par exemple "une information classifiée ne doit pas être transmise à un utilisateur non habilité à la connaître" ;
- d'autre part, établir un schéma d'autorisation, qui présente les règles permettant de modifier l'état de protection du système. Par exemple "le propriétaire d'une information peut accorder un droit d'accès pour cette information à n'importe quel utilisateur".

Pour construire une politique de sécurité il faut :

- d'une part, définir un ensemble de propriétés de sécurité qui doivent être satisfaites par le système. Par exemple "une information classifiée ne doit pas être transmise à un utilisateur non habilité à la connaître" ;
- d'autre part, établir un schéma d'autorisation, qui présente les règles permettant de modifier l'état de protection du système. Par exemple "le propriétaire d'une information peut accorder un droit d'accès pour cette information à n'importe quel utilisateur".

Pour construire une politique de sécurité il faut :

- d'une part, définir un ensemble de propriétés de sécurité qui doivent être satisfaites par le système. Par exemple "une information classifiée ne doit pas être transmise à un utilisateur non habilité à la connaître" ;
- d'autre part, établir un schéma d'autorisation, qui présente les règles permettant de modifier l'état de protection du système. Par exemple "le propriétaire d'une information peut accorder un droit d'accès pour cette information à n'importe quel utilisateur".

Pour construire une politique de sécurité il faut :

- d'une part, définir un ensemble de propriétés de sécurité qui doivent être satisfaites par le système. Par exemple "une information classifiée ne doit pas être transmise à un utilisateur non habilité à la connaître" ;
- d'autre part, établir un schéma d'autorisation, qui présente les règles permettant de modifier l'état de protection du système. Par exemple "le propriétaire d'une information peut accorder un droit d'accès pour cette information à n'importe quel utilisateur".

Si la politique d'autorisation est cohérente, alors il ne doit pas être possible, partant d'un état initial sûr (c'est-à-dire satisfaisant les propriétés de sécurité), d'atteindre un état d'insécurité (c'est-à-dire un état où les propriétés de sécurité ne sont pas satisfaites) en appliquant le schéma d'autorisation. Les propriétés de sécurité peuvent être définies en fonction de la confidentialité, l'intégrité ou encore de la disponibilité d'informations.

Si la politique d'autorisation est cohérente, alors il ne doit pas être possible, partant d'un état initial sûr (c'est-à-dire satisfaisant les propriétés de sécurité), d'atteindre un état d'insécurité (c'est-à-dire un état où les propriétés de sécurité ne sont pas satisfaites) en appliquant le schéma d'autorisation. Les propriétés de sécurité peuvent être définies en fonction de la confidentialité, l'intégrité ou encore de la disponibilité d'informations.

Si la politique d'autorisation est cohérente, alors il ne doit pas être possible, partant d'un état initial sûr (c'est-à-dire satisfaisant les propriétés de sécurité), d'atteindre un état d'insécurité (c'est-à-dire un état où les propriétés de sécurité ne sont pas satisfaites) en appliquant le schéma d'autorisation. Les propriétés de sécurité peuvent être définies en fonction de la confidentialité, l'intégrité ou encore de la disponibilité d'informations.

Si la politique d'autorisation est cohérente, alors il ne doit pas être possible, partant d'un état initial sûr (c'est-à-dire satisfaisant les propriétés de sécurité), d'atteindre un état d'insécurité (c'est-à-dire un état où les propriétés de sécurité ne sont pas satisfaites) en appliquant le schéma d'autorisation. Les propriétés de sécurité peuvent être définies en fonction de la confidentialité, l'intégrité ou encore de la disponibilité d'informations.

Si la politique d'autorisation est cohérente, alors il ne doit pas être possible, partant d'un état initial sûr (c'est-à-dire satisfaisant les propriétés de sécurité), d'atteindre un état d'insécurité (c'est-à-dire un état où les propriétés de sécurité ne sont pas satisfaites) en appliquant le schéma d'autorisation. Les propriétés de sécurité peuvent être définies en fonction de la confidentialité, l'intégrité ou encore de la disponibilité d'informations.

Si la politique d'autorisation est cohérente, alors il ne doit pas être possible, partant d'un état initial sûr (c'est-à-dire satisfaisant les propriétés de sécurité), d'atteindre un état d'insécurité (c'est-à-dire un état où les propriétés de sécurité ne sont pas satisfaites) en appliquant le schéma d'autorisation. Les propriétés de sécurité peuvent être définies en fonction de la confidentialité, l'intégrité ou encore de la disponibilité d'informations.

Si la politique d'autorisation est cohérente, alors il ne doit pas être possible, partant d'un état initial sûr (c'est-à-dire satisfaisant les propriétés de sécurité), d'atteindre un état d'insécurité (c'est-à-dire un état où les propriétés de sécurité ne sont pas satisfaites) en appliquant le schéma d'autorisation. Les propriétés de sécurité peuvent être définies en fonction de la confidentialité, l'intégrité ou encore de la disponibilité d'informations.

Une politique de sécurité peut se développer dans trois directions distinctes : les politiques de sécurité physique, administrative et logique.

La politique de sécurité physique précise un ensemble de procédures et de moyens qui protègent les locaux et les biens contre des risques majeurs (incendie, inondation, etc.) et contrôlent les accès physiques aux matériels informatiques et de communication (gardiens, codes, badges, ...).

Une politique de sécurité peut se développer dans trois directions distinctes : les politiques de sécurité physique, administrative et logique.

La politique de sécurité physique précise un ensemble de procédures et de moyens qui protègent les locaux et les biens contre des risques majeurs (incendie, inondation, etc.) et contrôlent les accès physiques aux matériels informatiques et de communication (gardiens, codes, badges, ...).

Une politique de sécurité peut se développer dans trois directions distinctes : les politiques de sécurité physique, administrative et logique.

La politique de sécurité physique précise un ensemble de procédures et de moyens qui protègent les locaux et les biens contre des risques majeurs (incendie, inondation, etc.) et contrôlent les accès physiques aux matériels informatiques et de communication (gardiens, codes, badges, ...).

La politique de sécurité administrative définit un ensemble de procédures et moyens qui traite de tout ce qui ressort de la sécurité d'un point de vue organisationnel au sein de l'entreprise. La structure de l'organigramme ainsi que la répartition des tâches (séparation des environnements de développement, d'industrialisation et de production des applicatifs) en font partie. Les propriétés de sécurité recherchées visent, par exemple, à limiter les cumuls ou les délégations abusives de pouvoir, ou à garantir une séparation des pouvoirs.

La politique de sécurité administrative définit un ensemble de procédures et moyens qui traite de tout ce qui ressort de la sécurité d'un point de vue organisationnel au sein de l'entreprise. La structure de l'organigramme ainsi que la répartition des tâches (séparation des environnements de développement, d'industrialisation et de production des applicatifs) en font partie. Les propriétés de sécurité recherchées visent, par exemple, à limiter les cumuls ou les délégations abusives de pouvoir, ou à garantir une séparation des pouvoirs.

La politique de sécurité administrative définit un ensemble de procédures et moyens qui traite de tout ce qui ressort de la sécurité d'un point de vue organisationnel au sein de l'entreprise. La structure de l'organigramme ainsi que la répartition des tâches (séparation des environnements de développement, d'industrialisation et de production des applicatifs) en font partie. Les propriétés de sécurité recherchées visent, par exemple, à limiter les cumuls ou les délégations abusives de pouvoir, ou à garantir une séparation des pouvoirs.

La politique de sécurité administrative définit un ensemble de procédures et moyens qui traite de tout ce qui ressort de la sécurité d'un point de vue organisationnel au sein de l'entreprise. La structure de l'organigramme ainsi que la répartition des tâches (séparation des environnements de développement, d'industrialisation et de production des applicatifs) en font partie. Les propriétés de sécurité recherchées visent, par exemple, à limiter les cumuls ou les délégations abusives de pouvoir, ou à garantir une séparation des pouvoirs.

La politique de sécurité logique fait référence à la gestion du contrôle d'accès logique, lequel repose sur un triple service d'identification, d'authentification et d'autorisation. Elle spécifie qui à le droit d'accéder à quoi, et dans quelles circonstances. Ainsi, tout utilisateur, avant de se servir du système, devra décliner son identité (identification) et prouver qu'il est bien la personne qu'il prétend être (authentification). Une fois la relation établie, les actions légitimes que peut faire cet utilisateur sont déterminées par la politique d'autorisation.

La politique de sécurité logique fait référence à la gestion du contrôle d'accès logique, lequel repose sur un triple service d'identification, d'authentification et d'autorisation. Elle spécifie qui à le droit d'accéder à quoi, et dans quelles circonstances. Ainsi, tout utilisateur, avant de se servir du système, devra décliner son identité (identification) et prouver qu'il est bien la personne qu'il prétend être (authentification). Une fois la relation établie, les actions légitimes que peut faire cet utilisateur sont déterminées par la politique d'autorisation.

La politique de sécurité logique fait référence à la gestion du contrôle d'accès logique, lequel repose sur un triple service d'identification, d'authentification et d'autorisation. Elle spécifie qui à le droit d'accéder à quoi, et dans quelles circonstances. Ainsi, tout utilisateur, avant de se servir du système, devra décliner son identité (identification) et prouver qu'il est bien la personne qu'il prétend être (authentification). Une fois la relation établie, les actions légitimes que peut faire cet utilisateur sont déterminées par la politique d'autorisation.

La politique de sécurité logique fait référence à la gestion du contrôle d'accès logique, lequel repose sur un triple service d'identification, d'authentification et d'autorisation. Elle spécifie qui à le droit d'accéder à quoi, et dans quelles circonstances. Ainsi, tout utilisateur, avant de se servir du système, devra décliner son identité (identification) et prouver qu'il est bien la personne qu'il prétend être (authentification). Une fois la relation établie, les actions légitimes que peut faire cet utilisateur sont déterminées par la politique d'autorisation.

La politique de sécurité logique fait référence à la gestion du contrôle d'accès logique, lequel repose sur un triple service d'identification, d'authentification et d'autorisation. Elle spécifie qui à le droit d'accéder à quoi, et dans quelles circonstances. Ainsi, tout utilisateur, avant de se servir du système, devra décliner son identité (identification) et prouver qu'il est bien la personne qu'il prétend être (authentification). Une fois la relation établie, les actions légitimes que peut faire cet utilisateur sont déterminées par la politique d'autorisation.

La politique de sécurité logique fait référence à la gestion du contrôle d'accès logique, lequel repose sur un triple service d'identification, d'authentification et d'autorisation. Elle spécifie qui à le droit d'accéder à quoi, et dans quelles circonstances. Ainsi, tout utilisateur, avant de se servir du système, devra décliner son identité (identification) et prouver qu'il est bien la personne qu'il prétend être (authentification). Une fois la relation établie, les actions légitimes que peut faire cet utilisateur sont déterminées par la politique d'autorisation.

L'autorisation consiste à gérer et à vérifier les droits d'accès, en fonction des règles spécifiées dans la politique de sécurité. On dit qu'un sujet (entité qui demande l'accès, dite aussi entité active) possède un droit d'accès sur un objet (entité à laquelle le sujet souhaite accéder, dite aussi entité passive) si et seulement s'il est autorisé à effectuer la fonction d'accès correspondante sur cet objet. Les droits d'accès peuvent être symboliquement représentés dans une matrice de droits d'accès dont les lignes représentent les sujets et les colonnes représentent les objets. Une cellule de la matrice contient donc les droits d'accès d'un sujet sur un objet. La matrice est gérée conformément aux règles définies dans la politique de sécurité.

L'autorisation consiste à gérer et à vérifier les droits d'accès, en fonction des règles spécifiées dans la politique de sécurité. On dit qu'un sujet (entité qui demande l'accès, dite aussi entité active) possède un droit d'accès sur un objet (entité à laquelle le sujet souhaite accéder, dite aussi entité passive) si et seulement s'il est autorisé à effectuer la fonction d'accès correspondante sur cet objet. Les droits d'accès peuvent être symboliquement représentés dans une matrice de droits d'accès dont les lignes représentent les sujets et les colonnes représentent les objets. Une cellule de la matrice contient donc les droits d'accès d'un sujet sur un objet. La matrice est gérée conformément aux règles définies dans la politique de sécurité.

L'autorisation consiste à gérer et à vérifier les droits d'accès, en fonction des règles spécifiées dans la politique de sécurité. On dit qu'un sujet (entité qui demande l'accès, dite aussi entité active) possède un droit d'accès sur un objet (entité à laquelle le sujet souhaite accéder, dite aussi entité passive) si et seulement s'il est autorisé à effectuer la fonction d'accès correspondante sur cet objet. Les droits d'accès peuvent être symboliquement représentés dans une matrice de droits d'accès dont les lignes représentent les sujets et les colonnes représentent les objets. Une cellule de la matrice contient donc les droits d'accès d'un sujet sur un objet. La matrice est gérée conformément aux règles définies dans la politique de sécurité.

L'autorisation consiste à gérer et à vérifier les droits d'accès, en fonction des règles spécifiées dans la politique de sécurité. On dit qu'un sujet (entité qui demande l'accès, dite aussi entité active) possède un droit d'accès sur un objet (entité à laquelle le sujet souhaite accéder, dite aussi entité passive) si et seulement s'il est autorisé à effectuer la fonction d'accès correspondante sur cet objet. Les droits d'accès peuvent être symboliquement représentés dans une matrice de droits d'accès dont les lignes représentent les sujets et les colonnes représentent les objets. Une cellule de la matrice contient donc les droits d'accès d'un sujet sur un objet. La matrice est gérée conformément aux règles définies dans la politique de sécurité.

L'autorisation consiste à gérer et à vérifier les droits d'accès, en fonction des règles spécifiées dans la politique de sécurité. On dit qu'un sujet (entité qui demande l'accès, dite aussi entité active) possède un droit d'accès sur un objet (entité à laquelle le sujet souhaite accéder, dite aussi entité passive) si et seulement s'il est autorisé à effectuer la fonction d'accès correspondante sur cet objet.

Les droits d'accès peuvent être symboliquement représentés dans une matrice de droits d'accès dont les lignes représentent les sujets et les colonnes représentent les objets. Une cellule de la matrice contient donc les droits d'accès d'un sujet sur un objet. La matrice est gérée conformément aux règles définies dans la politique de sécurité.

L'autorisation consiste à gérer et à vérifier les droits d'accès, en fonction des règles spécifiées dans la politique de sécurité. On dit qu'un sujet (entité qui demande l'accès, dite aussi entité active) possède un droit d'accès sur un objet (entité à laquelle le sujet souhaite accéder, dite aussi entité passive) si et seulement s'il est autorisé à effectuer la fonction d'accès correspondante sur cet objet. Les droits d'accès peuvent être symboliquement représentés dans une matrice de droits d'accès dont les lignes représentent les sujets et les colonnes représentent les objets. Une cellule de la matrice contient donc les droits d'accès d'un sujet sur un objet. La matrice est gérée conformément aux règles définies dans la politique de sécurité.

L'autorisation consiste à gérer et à vérifier les droits d'accès, en fonction des règles spécifiées dans la politique de sécurité. On dit qu'un sujet (entité qui demande l'accès, dite aussi entité active) possède un droit d'accès sur un objet (entité à laquelle le sujet souhaite accéder, dite aussi entité passive) si et seulement s'il est autorisé à effectuer la fonction d'accès correspondante sur cet objet. Les droits d'accès peuvent être symboliquement représentés dans une matrice de droits d'accès dont les lignes représentent les sujets et les colonnes représentent les objets. Une cellule de la matrice contient donc les droits d'accès d'un sujet sur un objet. La matrice est gérée conformément aux règles définies dans la politique de sécurité.

L'autorisation consiste à gérer et à vérifier les droits d'accès, en fonction des règles spécifiées dans la politique de sécurité. On dit qu'un sujet (entité qui demande l'accès, dite aussi entité active) possède un droit d'accès sur un objet (entité à laquelle le sujet souhaite accéder, dite aussi entité passive) si et seulement s'il est autorisé à effectuer la fonction d'accès correspondante sur cet objet. Les droits d'accès peuvent être symboliquement représentés dans une matrice de droits d'accès dont les lignes représentent les sujets et les colonnes représentent les objets. Une cellule de la matrice contient donc les droits d'accès d'un sujet sur un objet. La matrice est gérée conformément aux règles définies dans la politique de sécurité.

D'une manière générale, les règles de la politique de sécurité sont spécifiées en terme de permissions (par exemple tout médecin a le droit d'accéder aux dossiers médicaux de ses patients) et d'interdictions (par exemple, les médecins n'ont pas le droit d'effacer des diagnostics déjà établis), mais aussi en terme d'obligations (les médecins sont obligés de conserver les dossiers médicaux pendant la durée fixée par la loi).

D'une manière générale, les règles de la politique de sécurité sont spécifiées en terme de permissions (par exemple tout médecin a le droit d'accéder aux dossiers médicaux de ses patients) et d'interdictions (par exemple, les médecins n'ont pas le droit d'effacer des diagnostics déjà établis), mais aussi en terme d'obligations (les médecins sont obligés de conserver les dossiers médicaux pendant la durée fixée par la loi).

D'une manière générale, les règles de la politique de sécurité sont spécifiées en terme de permissions (par exemple tout médecin a le droit d'accéder aux dossiers médicaux de ses patients) et d'interdictions (par exemple, les médecins n'ont pas le droit d'effacer des diagnostics déjà établis), mais aussi en terme d'obligations (les médecins sont obligés de conserver les dossiers médicaux pendant la durée fixée par la loi).

D'une manière générale, les règles de la politique de sécurité sont spécifiées en terme de permissions (par exemple tout médecin a le droit d'accéder aux dossiers médicaux de ses patients) et d'interdictions (par exemple, les médecins n'ont pas le droit d'effacer des diagnostics déjà établis), mais aussi en terme d'obligations (les médecins sont obligés de conserver les dossiers médicaux pendant la durée fixée par la loi).

D'une manière générale, les règles de la politique de sécurité sont spécifiées en terme de permissions (par exemple tout médecin a le droit d'accéder aux dossiers médicaux de ses patients) et d'interdictions (par exemple, les médecins n'ont pas le droit d'effacer des diagnostics déjà établis), mais aussi en terme d'obligations (les médecins sont obligés de conserver les dossiers médicaux pendant la durée fixée par la loi).

D'une manière générale, les règles de la politique de sécurité sont spécifiées en terme de permissions (par exemple tout médecin a le droit d'accéder aux dossiers médicaux de ses patients) et d'interdictions (par exemple, les médecins n'ont pas le droit d'effacer des diagnostics déjà établis), mais aussi en terme d'obligations (les médecins sont obligés de conserver les dossiers médicaux pendant la durée fixée par la loi).

Les premiers critères d'évaluation de la sécurité ont été définis aux États-Unis dans ce qui est couramment appelé le Livre Orange ou TCSEC (Trusted Computer System Evaluation Criteria). Ces critères, fondés à la fois sur des listes de fonctions de sécurité à remplir et sur les techniques employées pour la vérification, conduisent à classer les systèmes en sept catégories ou niveaux (D, C1, C2, B1, B2, B3, A1).

Les premiers critères d'évaluation de la sécurité ont été définis aux États-Unis dans ce qui est couramment appelé le Livre Orange ou TCSEC (Trusted Computer System Evaluation Criteria). Ces critères, fondés à la fois sur des listes de fonctions de sécurité à remplir et sur les techniques employées pour la vérification, conduisent à classer les systèmes en sept catégories ou niveaux (D, C1, C2, B1, B2, B3, A1).

Les premiers critères d'évaluation de la sécurité ont été définis aux États-Unis dans ce qui est couramment appelé le Livre Orange ou TCSEC (Trusted Computer System Evaluation Criteria). Ces critères, fondés à la fois sur des listes de fonctions de sécurité à remplir et sur les techniques employées pour la vérification, conduisent à classer les systèmes en sept catégories ou niveaux (D, C1, C2, B1, B2, B3, A1).

Pour chaque niveau, quatre familles de critères sont définies, traitant de la politique d'autorisation, de l'audit, de l'assurance et de la documentation :

- La politique d'autorisation stipule une politique précise à suivre en fonction des différents niveaux de certifications visés.
- Les critères d'audit précisent les fonctions requises en matière d'identification, d'authentification et d'audit des actions.
- Les critères d'assurance fixent des recommandations concernant des méthodes de conception et de vérification utilisées afin d'augmenter la confiance de l'évaluateur. Il s'agit de garantir que le système implémente bien la fonctionnalité qu'il prétend avoir.
- Les critères de documentation spécifient les documents qui doivent être fournis avec le produit lors de l'évaluation.

Pour chaque niveau, quatre familles de critères sont définies, traitant de la politique d'autorisation, de l'audit, de l'assurance et de la documentation :

- La politique d'autorisation stipule une politique précise à suivre en fonction des différents niveaux de certifications visés.
- Les critères d'audit précisent les fonctions requises en matière d'identification, d'authentification et d'audit des actions.
- Les critères d'assurance fixent des recommandations concernant des méthodes de conception et de vérification utilisées afin d'augmenter la confiance de l'évaluateur. Il s'agit de garantir que le système implémente bien la fonctionnalité qu'il prétend avoir.
- Les critères de documentation spécifient les documents qui doivent être fournis avec le produit lors de l'évaluation.

Pour chaque niveau, quatre familles de critères sont définies, traitant de la politique d'autorisation, de l'audit, de l'assurance et de la documentation :

- La politique d'autorisation stipule une politique précise à suivre en fonction des différents niveaux de certifications visés.
- Les critères d'audit précisent les fonctions requises en matière d'identification, d'authentification et d'audit des actions.
- Les critères d'assurance fixent des recommandations concernant des méthodes de conception et de vérification utilisées afin d'augmenter la confiance de l'évaluateur. Il s'agit de garantir que le système implémente bien la fonctionnalité qu'il prétend avoir.
- Les critères de documentation spécifient les documents qui doivent être fournis avec le produit lors de l'évaluation.

Pour chaque niveau, quatre familles de critères sont définies, traitant de la politique d'autorisation, de l'audit, de l'assurance et de la documentation :

- La politique d'autorisation stipule une politique précise à suivre en fonction des différents niveaux de certifications visés.
- Les critères d'audit précisent les fonctions requises en matière d'identification, d'authentification et d'audit des actions.
- Les critères d'assurance fixent des recommandations concernant des méthodes de conception et de vérification utilisées afin d'augmenter la confiance de l'évaluateur. Il s'agit de garantir que le système implémente bien la fonctionnalité qu'il prétend avoir.
- Les critères de documentation spécifient les documents qui doivent être fournis avec le produit lors de l'évaluation.

Pour chaque niveau, quatre familles de critères sont définies, traitant de la politique d'autorisation, de l'audit, de l'assurance et de la documentation :

- La politique d'autorisation stipule une politique précise à suivre en fonction des différents niveaux de certifications visés.
- Les critères d'audit précisent les fonctions requises en matière d'identification, d'authentification et d'audit des actions.
- Les critères d'assurance fixent des recommandations concernant des méthodes de conception et de vérification utilisées afin d'augmenter la confiance de l'évaluateur. Il s'agit de garantir que le système implémente bien la fonctionnalité qu'il prétend avoir.
- Les critères de documentation spécifient les documents qui doivent être fournis avec le produit lors de l'évaluation.

Pour chaque niveau, quatre familles de critères sont définies, traitant de la politique d'autorisation, de l'audit, de l'assurance et de la documentation :

- La politique d'autorisation stipule une politique précise à suivre en fonction des différents niveaux de certifications visés.
- Les critères d'audit précisent les fonctions requises en matière d'identification, d'authentification et d'audit des actions.
- Les critères d'assurance fixent des recommandations concernant des méthodes de conception et de vérification utilisées afin d'augmenter la confiance de l'évaluateur. Il s'agit de garantir que le système implémente bien la fonctionnalité qu'il prétend avoir.
- Les critères de documentation spécifient les documents qui doivent être fournis avec le produit lors de l'évaluation.

Les caractéristiques principales des différents niveaux définis par le livre orange sont ainsi :

- un système classé au niveau D est un système qui n'a pas été évalué ;
- jusqu'aux niveaux C1 et C2, le système peut utiliser une politique discrétionnaire (voir dans la suite du cours) ;
- pour les niveaux B1, B2, et B3 le système utilise une politique obligatoire (voir dans la suite du cours) ;
- un système classé A1 est fonctionnellement équivalant à un système classé B3, sauf qu'il est caractérisé par l'utilisation de méthodes formelles de vérification pour prouver que les contrôles utilisés permettent bien d'assurer la protection des informations sensibles.

Les caractéristiques principales des différents niveaux définis par le livre orange sont ainsi :

- un système classé au niveau D est un système qui n'a pas été évalué ;
- jusqu'aux niveaux C1 et C2, le système peut utiliser une politique discrétionnaire (voir dans la suite du cours) ;
- pour les niveaux B1, B2, et B3 le système utilise une politique obligatoire (voir dans la suite du cours) ;
- un système classé A1 est fonctionnellement équivalant à un système classé B3, sauf qu'il est caractérisé par l'utilisation de méthodes formelles de vérification pour prouver que les contrôles utilisés permettent bien d'assurer la protection des informations sensibles.

Les caractéristiques principales des différents niveaux définis par le livre orange sont ainsi :

- un système classé au niveau D est un système qui n'a pas été évalué ;
- jusqu'aux niveaux C1 et C2, le système peut utiliser une politique discrétionnaire (voir dans la suite du cours) ;
- pour les niveaux B1, B2, et B3 le système utilise une politique obligatoire (voir dans la suite du cours) ;
- un système classé A1 est fonctionnellement équivalant à un système classé B3, sauf qu'il est caractérisé par l'utilisation de méthodes formelles de vérification pour prouver que les contrôles utilisés permettent bien d'assurer la protection des informations sensibles.

Les caractéristiques principales des différents niveaux définis par le livre orange sont ainsi :

- un système classé au niveau D est un système qui n'a pas été évalué ;
- jusqu'aux niveaux C1 et C2, le système peut utiliser une politique discrétionnaire (voir dans la suite du cours) ;
- pour les niveaux B1, B2, et B3 le système utilise une politique obligatoire (voir dans la suite du cours) ;
- un système classé A1 est fonctionnellement équivalant à un système classé B3, sauf qu'il est caractérisé par l'utilisation de méthodes formelles de vérification pour prouver que les contrôles utilisés permettent bien d'assurer la protection des informations sensibles.

Les caractéristiques principales des différents niveaux définis par le livre orange sont ainsi :

- un système classé au niveau D est un système qui n'a pas été évalué ;
- jusqu'aux niveaux C1 et C2, le système peut utiliser une politique discrétionnaire (voir dans la suite du cours) ;
- pour les niveaux B1, B2, et B3 le système utilise une politique obligatoire (voir dans la suite du cours) ;
- un système classé A1 est fonctionnellement équivalant à un système classé B3, sauf qu'il est caractérisé par l'utilisation de méthodes formelles de vérification pour prouver que les contrôles utilisés permettent bien d'assurer la protection des informations sensibles.

Les TCSEC visent d'abord à satisfaire les besoins du DoD (Department of Defense) des États-Unis, privilégiant ainsi la confidentialité des données militaires. Par ailleurs, le manque de souplesse et la difficulté de leur mise en oeuvre, ont conduit au développement de nouvelles générations de critères. À titre d'exemple abordons les critères adoptés par l'ex-Communauté Européenne (ITSEC, 1991), mais d'autres pays tels que le Canada (CTCPEC, 1993) et le Japon (JCSEC, 1992) ont également élaboré leurs propres critères d'évaluation.

Les TCSEC visent d'abord à satisfaire les besoins du DoD (Department of Defense) des États-Unis, privilégiant ainsi la confidentialité des données militaires. Par ailleurs, le manque de souplesse et la difficulté de leur mise en oeuvre, ont conduit au développement de nouvelles générations de critères. À titre d'exemple abordons les critères adoptés par l'ex-Communauté Européenne (ITSEC, 1991), mais d'autres pays tels que le Canada (CTCPEC, 1993) et le Japon (JCSEC, 1992) ont également élaboré leurs propres critères d'évaluation.

Les TCSEC visent d'abord à satisfaire les besoins du DoD (Department of Defense) des États-Unis, privilégiant ainsi la confidentialité des données militaires. Par ailleurs, le manque de souplesse et la difficulté de leur mise en oeuvre, ont conduit au développement de nouvelles générations de critères. À titre d'exemple abordons les critères adoptés par l'ex-Communauté Européenne (ITSEC, 1991), mais d'autres pays tels que le Canada (CTCPEC, 1993) et le Japon (JCSEC, 1992) ont également élaboré leurs propres critères d'évaluation.

Les TCSEC visent d'abord à satisfaire les besoins du DoD (Department of Defense) des États-Unis, privilégiant ainsi la confidentialité des données militaires. Par ailleurs, le manque de souplesse et la difficulté de leur mise en oeuvre, ont conduit au développement de nouvelles générations de critères. À titre d'exemple abordons les critères adoptés par l'ex-Communauté Européenne (ITSEC, 1991), mais d'autres pays tels que le Canada (CTCPEC, 1993) et le Japon (JCSEC, 1992) ont également élaboré leurs propres critères d'évaluation.

Les ITSEC (Information Technology Security Evaluation Criteria) sont le résultat d'harmonisation de travaux réalisés au sein de quatre pays européens : l'Allemagne, la France, les Pays-Bas et le Royaume-Uni. La différence essentielle entre les TCSEC et les ITSEC réside dans la distinction entre fonctionnalité et assurance. Une classe de fonctionnalité décrit les fonctions que doit mettre en œuvre un système tandis qu'une classe d'assurance décrit l'ensemble des preuves qu'un système doit apporter pour montrer qu'il implémente les fonctions qu'il prétend fournir.

Les ITSEC (Information Technology Security Evaluation Criteria) sont le résultat d'harmonisation de travaux réalisés au sein de quatre pays européens : l'Allemagne, la France, les Pays-Bas et le Royaume-Uni. La différence essentielle entre les TCSEC et les ITSEC réside dans la distinction entre fonctionnalité et assurance.

Une classe de fonctionnalité décrit les fonctions que doit mettre en œuvre un système tandis qu'une classe d'assurance décrit l'ensemble des preuves qu'un système doit apporter pour montrer qu'il implémente les fonctions qu'il prétend fournir.

Les ITSEC (Information Technology Security Evaluation Criteria) sont le résultat d'harmonisation de travaux réalisés au sein de quatre pays européens : l'Allemagne, la France, les Pays-Bas et le Royaume-Uni. La différence essentielle entre les TCSEC et les ITSEC réside dans la distinction entre fonctionnalité et assurance. Une classe de fonctionnalité décrit les fonctions que doit mettre en œuvre un système tandis qu'une classe d'assurance décrit l'ensemble des preuves qu'un système doit apporter pour montrer qu'il implémente les fonctions qu'il prétend fournir.

Les ITSEC (Information Technology Security Evaluation Criteria) sont le résultat d'harmonisation de travaux réalisés au sein de quatre pays européens : l'Allemagne, la France, les Pays-Bas et le Royaume-Uni. La différence essentielle entre les TCSEC et les ITSEC réside dans la distinction entre fonctionnalité et assurance. Une classe de fonctionnalité décrit les fonctions que doit mettre en œuvre un système tandis qu'une classe d'assurance décrit l'ensemble des preuves qu'un système doit apporter pour montrer qu'il implémente les fonctions qu'il prétend fournir.

Les ITSEC introduisent également la notion de cible d'évaluation (TOE pour Target Of Evaluation). Une TOE rassemble les différents éléments du contexte d'évaluation, dont une politique de sécurité, une spécification des fonctions requises dédiées à la sécurité, une définition des mécanismes de sécurité (optionnelle), la cotation annoncée de la résistance minimum des mécanismes, ainsi que le niveau d'évaluation visé.

Les ITSEC introduisent également la notion de cible d'évaluation (TOE pour Target Of Evaluation). Une TOE rassemble les différents éléments du contexte d'évaluation, dont une politique de sécurité, une spécification des fonctions requises dédiées à la sécurité, une définition des mécanismes de sécurité (optionnelle), la cotation annoncée de la résistance minimum des mécanismes, ainsi que le niveau d'évaluation visé.

Les ITSEC proposent plusieurs classes de fonctionnalités de base :

- les classes de fonctionnalité F-C1, F-C2, F-B1, F-B2, F-B3 sont des classes de confidentialité qui correspondent aux exigences de fonctionnalité des classes C1 à A1 dans les TCSEC ;
- la classe de fonctionnalité F-IN concerne les TOE pour lesquelles il y a des exigences d'intégrité (par exemple, pour les bases de données) ;
- la classe de fonctionnalité F-AV impose des exigences de disponibilité ;
- la classe de fonctionnalité F-DI impose des exigences élevées pour l'intégrité des données au cours de leur transmission ;
- la classe de fonctionnalité F-DX est destinée aux réseaux exigeants en matière de confidentialité et d'intégrité de l'information.

Les ITSEC proposent plusieurs classes de fonctionnalités de base :

- les classes de fonctionnalité F-C1, F-C2, F-B1, F-B2, F-B3 sont des classes de confidentialité qui correspondent aux exigences de fonctionnalité des classes C1 à A1 dans les TCSEC ;
- la classe de fonctionnalité F-IN concerne les TOE pour lesquelles il y a des exigences d'intégrité (par exemple, pour les bases de données) ;
- la classe de fonctionnalité F-AV impose des exigences de disponibilité ;
- la classe de fonctionnalité F-DI impose des exigences élevées pour l'intégrité des données au cours de leur transmission ;
- la classe de fonctionnalité F-DX est destinée aux réseaux exigeants en matière de confidentialité et d'intégrité de l'information.

Les ITSEC proposent plusieurs classes de fonctionnalités de base :

- les classes de fonctionnalité F-C1, F-C2, F-B1, F-B2, F-B3 sont des classes de confidentialité qui correspondent aux exigences de fonctionnalité des classes C1 à A1 dans les TCSEC ;
- la classe de fonctionnalité F-IN concerne les TOE pour lesquelles il y a des exigences d'intégrité (par exemple, pour les bases de données) ;
- la classe de fonctionnalité F-AV impose des exigences de disponibilité ;
- la classe de fonctionnalité F-DI impose des exigences élevées pour l'intégrité des données au cours de leur transmission ;
- la classe de fonctionnalité F-DX est destinée aux réseaux exigeants en matière de confidentialité et d'intégrité de l'information.

Les ITSEC proposent plusieurs classes de fonctionnalités de base :

- les classes de fonctionnalité F-C1, F-C2, F-B1, F-B2, F-B3 sont des classes de confidentialité qui correspondent aux exigences de fonctionnalité des classes C1 à A1 dans les TCSEC ;
- la classe de fonctionnalité F-IN concerne les TOE pour lesquelles il y a des exigences d'intégrité (par exemple, pour les bases de données) ;
- la classe de fonctionnalité F-AV impose des exigences de disponibilité ;
- la classe de fonctionnalité F-DI impose des exigences élevées pour l'intégrité des données au cours de leur transmission ;
- la classe de fonctionnalité F-DX est destinée aux réseaux exigeants en matière de confidentialité et d'intégrité de l'information.

Les ITSEC proposent plusieurs classes de fonctionnalités de base :

- les classes de fonctionnalité F-C1, F-C2, F-B1, F-B2, F-B3 sont des classes de confidentialité qui correspondent aux exigences de fonctionnalité des classes C1 à A1 dans les TCSEC ;
- la classe de fonctionnalité F-IN concerne les TOE pour lesquelles il y a des exigences d'intégrité (par exemple, pour les bases de données) ;
- la classe de fonctionnalité F-AV impose des exigences de disponibilité ;
- la classe de fonctionnalité F-DI impose des exigences élevées pour l'intégrité des données au cours de leur transmission ;
- la classe de fonctionnalité F-DX est destinée aux réseaux exigeants en matière de confidentialité et d'intégrité de l'information.

Les ITSEC proposent plusieurs classes de fonctionnalités de base :

- les classes de fonctionnalité F-C1, F-C2, F-B1, F-B2, F-B3 sont des classes de confidentialité qui correspondent aux exigences de fonctionnalité des classes C1 à A1 dans les TCSEC ;
- la classe de fonctionnalité F-IN concerne les TOE pour lesquelles il y a des exigences d'intégrité (par exemple, pour les bases de données) ;
- la classe de fonctionnalité F-AV impose des exigences de disponibilité ;
- la classe de fonctionnalité F-DI impose des exigences élevées pour l'intégrité des données au cours de leur transmission ;
- la classe de fonctionnalité F-DX est destinée aux réseaux exigeants en matière de confidentialité et d'intégrité de l'information.

La tentative d'harmonisation des critères canadiens, européens et américains, a donné naissance aux critères communs (en anglais Common Criteria for Information Security Evaluation) qui sont maintenant une norme internationale (ISO 15408). Ces critères contiennent deux parties bien distinctes comme dans les ITSEC : fonctionnalité et assurance. Les critères communs définissent également une cible d'évaluation (TOE) ainsi que les profils de protection, déjà existants dans les critères fédéraux américains (Federal Criteria, 1992).

La tentative d'harmonisation des critères canadiens, européens et américains, a donné naissance aux critères communs (en anglais Common Criteria for Information Security Evaluation) qui sont maintenant une norme internationale (ISO 15408). Ces critères contiennent deux parties bien distinctes comme dans les ITSEC : fonctionnalité et assurance. Les critères communs définissent également une cible d'évaluation (TOE) ainsi que les profils de protection, déjà existants dans les critères fédéraux américains (Federal Criteria, 1992).

La tentative d'harmonisation des critères canadiens, européens et américains, a donné naissance aux critères communs (en anglais Common Criteria for Information Security Evaluation) qui sont maintenant une norme internationale (ISO 15408). Ces critères contiennent deux parties bien distinctes comme dans les ITSEC : fonctionnalité et assurance. Les critères communs définissent également une cible d'évaluation (TOE) ainsi que les profils de protection, déjà existants dans les critères fédéraux américains (Federal Criteria, 1992).

La tentative d'harmonisation des critères canadiens, européens et américains, a donné naissance aux critères communs (en anglais Common Criteria for Information Security Evaluation) qui sont maintenant une norme internationale (ISO 15408). Ces critères contiennent deux parties bien distinctes comme dans les ITSEC : fonctionnalité et assurance. Les critères communs définissent également une cible d'évaluation (TOE) ainsi que les profils de protection, déjà existants dans les critères fédéraux américains (Federal Criteria, 1992).

La tentative d'harmonisation des critères canadiens, européens et américains, a donné naissance aux critères communs (en anglais Common Criteria for Information Security Evaluation) qui sont maintenant une norme internationale (ISO 15408). Ces critères contiennent deux parties bien distinctes comme dans les ITSEC : fonctionnalité et assurance. Les critères communs définissent également une cible d'évaluation (TOE) ainsi que les profils de protection, déjà existants dans les critères fédéraux américains (Federal Criteria, 1992).

La plupart des politiques de sécurité reposent sur les notions de sujets, d'objets et de droits d'accès. Un sujet est une entité active, correspondant à un processus qui s'exécute pour le compte d'un utilisateur. Dans ce contexte, un utilisateur est soit une personne physique connue du système informatique et enregistrée comme utilisateur, soit un serveur, sorte de personne morale représentant des fonctions de service automatiques, tel que le serveur d'impression, serveur de base de données, serveur de messagerie, etc. Un objet est une entité considérée comme "passive" qui contient ou reçoit des informations. À un instant donné, un sujet a un droit d'accès sur un objet si et seulement si le processus correspondant au sujet est autorisé à exécuter l'opération correspondant à ce type d'accès sur cet objet.

La plupart des politiques de sécurité reposent sur les notions de sujets, d'objets et de droits d'accès. Un sujet est une entité active, correspondant à un processus qui s'exécute pour le compte d'un utilisateur. Dans ce contexte, un utilisateur est soit une personne physique connue du système informatique et enregistrée comme utilisateur, soit un serveur, sorte de personne morale représentant des fonctions de service automatiques, tel que le serveur d'impression, serveur de base de données, serveur de messagerie, etc. Un objet est une entité considérée comme "passive" qui contient ou reçoit des informations. À un instant donné, un sujet a un droit d'accès sur un objet si et seulement si le processus correspondant au sujet est autorisé à exécuter l'opération correspondant à ce type d'accès sur cet objet.

La plupart des politiques de sécurité reposent sur les notions de sujets, d'objets et de droits d'accès. Un sujet est une entité active, correspondant à un processus qui s'exécute pour le compte d'un utilisateur. Dans ce contexte, un utilisateur est soit une personne physique connue du système informatique et enregistrée comme utilisateur, soit un serveur, sorte de personne morale représentant des fonctions de service automatiques, tel que le serveur d'impression, serveur de base de données, serveur de messagerie, etc. Un objet est une entité considérée comme "passive" qui contient ou reçoit des informations. À un instant donné, un sujet a un droit d'accès sur un objet si et seulement si le processus correspondant au sujet est autorisé à exécuter l'opération correspondant à ce type d'accès sur cet objet.

La plupart des politiques de sécurité reposent sur les notions de sujets, d'objets et de droits d'accès. Un sujet est une entité active, correspondant à un processus qui s'exécute pour le compte d'un utilisateur. Dans ce contexte, un utilisateur est soit une personne physique connue du système informatique et enregistrée comme utilisateur, soit un serveur, sorte de personne morale représentant des fonctions de service automatiques, tel que le serveur d'impression, serveur de base de données, serveur de messagerie, etc. Un objet est une entité considérée comme "passive" qui contient ou reçoit des informations. À un instant donné, un sujet a un droit d'accès sur un objet si et seulement si le processus correspondant au sujet est autorisé à exécuter l'opération correspondant à ce type d'accès sur cet objet.

La plupart des politiques de sécurité reposent sur les notions de sujets, d'objets et de droits d'accès. Un sujet est une entité active, correspondant à un processus qui s'exécute pour le compte d'un utilisateur. Dans ce contexte, un utilisateur est soit une personne physique connue du système informatique et enregistrée comme utilisateur, soit un serveur, sorte de personne morale représentant des fonctions de service automatiques, tel que le serveur d'impression, serveur de base de données, serveur de messagerie, etc. Un objet est une entité considérée comme "passive" qui contient ou reçoit des informations. À un instant donné, un sujet a un droit d'accès sur un objet si et seulement si le processus correspondant au sujet est autorisé à exécuter l'opération correspondant à ce type d'accès sur cet objet.

La plupart des politiques de sécurité reposent sur les notions de sujets, d'objets et de droits d'accès. Un sujet est une entité active, correspondant à un processus qui s'exécute pour le compte d'un utilisateur. Dans ce contexte, un utilisateur est soit une personne physique connue du système informatique et enregistrée comme utilisateur, soit un serveur, sorte de personne morale représentant des fonctions de service automatiques, tel que le serveur d'impression, serveur de base de données, serveur de messagerie, etc. Un objet est une entité considérée comme "passive" qui contient ou reçoit des informations. À un instant donné, un sujet a un droit d'accès sur un objet si et seulement si le processus correspondant au sujet est autorisé à exécuter l'opération correspondant à ce type d'accès sur cet objet.

Les politiques de sécurité, ou plus précisément leurs schémas d'autorisation, se classent en deux grandes catégories : les politiques discrétionnaires (ou DAC pour Discretionary Access Control) et les politiques obligatoires (ou MAC pour Mandatory Access Control). Il existe également des variantes de ces politiques qui peuvent mieux s'adapter à des organisations particulières, comme les politiques basées sur la notion de rôles (ou RBAC pour Role-Based Access Control) ou encore sur la notion d'équipes (ou TMAC pour TeaM-based Access Control). Idéalement, il faut construire une politique de sécurité de telle sorte qu'aucune séquence valide d'applications des règles (du schéma d'autorisation) ne puisse amener le système dans un état tel qu'un objectif de sécurité soit violé, en partant d'un état inital sûr (on parle de politique de sécurité cohérente). Ceci suppose l'utilisation d'une méthode formelle de construction des règles du schéma d'autorisation, à partir d'une spécification formelle des objectifs de sécurité.

Les politiques de sécurité, ou plus précisément leurs schémas d'autorisation, se classent en deux grandes catégories : les politiques discrétionnaires (ou DAC pour Discretionary Access Control) et les politiques obligatoires (ou MAC pour Mandatory Access Control). Il existe également des variantes de ces politiques qui peuvent mieux s'adapter à des organisations particulières, comme les politiques basées sur la notion de rôles (ou RBAC pour Role-Based Access Control) ou encore sur la notion d'équipes (ou TMAC pour TeaM-based Access Control). Idéalement, il faut construire une politique de sécurité de telle sorte qu'aucune séquence valide d'applications des règles (du schéma d'autorisation) ne puisse amener le système dans un état tel qu'un objectif de sécurité soit violé, en partant d'un état inital sûr (on parle de politique de sécurité cohérente). Ceci suppose l'utilisation d'une méthode formelle de construction des règles du schéma d'autorisation, à partir d'une spécification formelle des objectifs de sécurité.

Les politiques de sécurité, ou plus précisément leurs schémas d'autorisation, se classent en deux grandes catégories : les politiques discrétionnaires (ou DAC pour Discretionary Access Control) et les politiques obligatoires (ou MAC pour Mandatory Access Control). Il existe également des variantes de ces politiques qui peuvent mieux s'adapter à des organisations particulières, comme les politiques basées sur la notion de rôles (ou RBAC pour Role-Based Access Control) ou encore sur la notion d'équipes (ou TMAC pour TeaM-based Access Control). Idéalement, il faut construire une politique de sécurité de telle sorte qu'aucune séquence valide d'applications des règles (du schéma d'autorisation) ne puisse amener le système dans un état tel qu'un objectif de sécurité soit violé, en partant d'un état inital sûr (on parle de politique de sécurité cohérente). Ceci suppose l'utilisation d'une méthode formelle de construction des règles du schéma d'autorisation, à partir d'une spécification formelle des objectifs de sécurité.

Les politiques de sécurité, ou plus précisément leurs schémas d'autorisation, se classent en deux grandes catégories : les politiques discrétionnaires (ou DAC pour Discretionary Access Control) et les politiques obligatoires (ou MAC pour Mandatory Access Control). Il existe également des variantes de ces politiques qui peuvent mieux s'adapter à des organisations particulières, comme les politiques basées sur la notion de rôles (ou RBAC pour Role-Based Access Control) ou encore sur la notion d'équipes (ou TMAC pour TeaM-based Access Control). Idéalement, il faut construire une politique de sécurité de telle sorte qu'aucune séquence valide d'applications des règles (du schéma d'autorisation) ne puisse amener le système dans un état tel qu'un objectif de sécurité soit violé, en partant d'un état inital sûr (on parle de politique de sécurité cohérente). Ceci suppose l'utilisation d'une méthode formelle de construction des règles du schéma d'autorisation, à partir d'une spécification formelle des objectifs de sécurité.

Les politiques de sécurité, ou plus précisément leurs schémas d'autorisation, se classent en deux grandes catégories : les politiques discrétionnaires (ou DAC pour Discretionary Access Control) et les politiques obligatoires (ou MAC pour Mandatory Access Control). Il existe également des variantes de ces politiques qui peuvent mieux s'adapter à des organisations particulières, comme les politiques basées sur la notion de rôles (ou RBAC pour Role-Based Access Control) ou encore sur la notion d'équipes (ou TMAC pour TeaM-based Access Control). Idéalement, il faut construire une politique de sécurité de telle sorte qu'aucune séquence valide d'applications des règles (du schéma d'autorisation) ne puisse amener le système dans un état tel qu'un objectif de sécurité soit violé, en partant d'un état inital sûr (on parle de politique de sécurité cohérente). Ceci suppose l'utilisation d'une méthode formelle de construction des règles du schéma d'autorisation, à partir d'une spécification formelle des objectifs de sécurité.

Les politiques de sécurité, ou plus précisément leurs schémas d'autorisation, se classent en deux grandes catégories : les politiques discrétionnaires (ou DAC pour Discretionary Access Control) et les politiques obligatoires (ou MAC pour Mandatory Access Control). Il existe également des variantes de ces politiques qui peuvent mieux s'adapter à des organisations particulières, comme les politiques basées sur la notion de rôles (ou RBAC pour Role-Based Access Control) ou encore sur la notion d'équipes (ou TMAC pour TeaM-based Access Control). Idéalement, il faut construire une politique de sécurité de telle sorte qu'aucune séquence valide d'applications des règles (du schéma d'autorisation) ne puisse amener le système dans un état tel qu'un objectif de sécurité soit violé, en partant d'un état inital sûr (on parle de politique de sécurité cohérente). Ceci suppose l'utilisation d'une méthode formelle de construction des règles du schéma d'autorisation, à partir d'une spécification formelle des objectifs de sécurité.

Les politiques d'autorisation les plus citées dans la littérature sont généralement associées à un modèle de sécurité. D'une manière générale, un modèle peut être défini comme un formalisme (souvent mathématique) qui offre une vue subjective mais pertinente de la réalité. On modélise pour mieux comprendre le système qu'on développe, c'est-à-dire pour visualiser ses propriétés, spécifier sa structure ou son comportement, documenter et guider sa construction, etc. À partir de là, un modèle de sécurité peut être défini comme un formalisme permettant de représenter, de façon claire et non-ambiguë, la politique de sécurité. Il aide à l'abstraire (afin de réduire sa complexité) et à faciliter sa compréhension, comme il peut servir à vérifier que cette politique est complète (tout est protégé) et cohérente, et que la mise en œuvre par le système de protection est conforme aux propriétés attendues du système.

Les politiques d'autorisation les plus citées dans la littérature sont généralement associées à un modèle de sécurité. D'une manière générale, un modèle peut être défini comme un formalisme (souvent mathématique) qui offre une vue subjective mais pertinente de la réalité. On modélise pour mieux comprendre le système qu'on développe, c'est-à-dire pour visualiser ses propriétés, spécifier sa structure ou son comportement, documenter et guider sa construction, etc. À partir de là, un modèle de sécurité peut être défini comme un formalisme permettant de représenter, de façon claire et non-ambiguë, la politique de sécurité. Il aide à l'abstraire (afin de réduire sa complexité) et à faciliter sa compréhension, comme il peut servir à vérifier que cette politique est complète (tout est protégé) et cohérente, et que la mise en œuvre par le système de protection est conforme aux propriétés attendues du système.

Les politiques d'autorisation les plus citées dans la littérature sont généralement associées à un modèle de sécurité. D'une manière générale, un modèle peut être défini comme un formalisme (souvent mathématique) qui offre une vue subjective mais pertinente de la réalité. On modélise pour mieux comprendre le système qu'on développe, c'est-à-dire pour visualiser ses propriétés, spécifier sa structure ou son comportement, documenter et guider sa construction, etc. À partir de là, un modèle de sécurité peut être défini comme un formalisme permettant de représenter, de façon claire et non-ambiguë, la politique de sécurité. Il aide à l'abstraire (afin de réduire sa complexité) et à faciliter sa compréhension, comme il peut servir à vérifier que cette politique est complète (tout est protégé) et cohérente, et que la mise en œuvre par le système de protection est conforme aux propriétés attendues du système.

Les politiques d'autorisation les plus citées dans la littérature sont généralement associées à un modèle de sécurité. D'une manière générale, un modèle peut être défini comme un formalisme (souvent mathématique) qui offre une vue subjective mais pertinente de la réalité. On modélise pour mieux comprendre le système qu'on développe, c'est-à-dire pour visualiser ses propriétés, spécifier sa structure ou son comportement, documenter et guider sa construction, etc. À partir de là, un modèle de sécurité peut être défini comme un formalisme permettant de représenter, de façon claire et non-ambiguë, la politique de sécurité. Il aide à l'abstraire (afin de réduire sa complexité) et à faciliter sa compréhension, comme il peut servir à vérifier que cette politique est complète (tout est protégé) et cohérente, et que la mise en œuvre par le système de protection est conforme aux propriétés attendues du système.

Les politiques d'autorisation les plus citées dans la littérature sont généralement associées à un modèle de sécurité. D'une manière générale, un modèle peut être défini comme un formalisme (souvent mathématique) qui offre une vue subjective mais pertinente de la réalité. On modélise pour mieux comprendre le système qu'on développe, c'est-à-dire pour visualiser ses propriétés, spécifier sa structure ou son comportement, documenter et guider sa construction, etc. À partir de là, un modèle de sécurité peut être défini comme un formalisme permettant de représenter, de façon claire et non-ambiguë, la politique de sécurité. Il aide à l'abstraire (afin de réduire sa complexité) et à faciliter sa compréhension, comme il peut servir à vérifier que cette politique est complète (tout est protégé) et cohérente, et que la mise en œuvre par le système de protection est conforme aux propriétés attendues du système.

Les modèles de sécurité peuvent être classés en deux grandes familles :

- des modèles généraux, qui sont plutôt des méthodes de description formelle, pouvant s'appliquer à toute sorte de politiques. C'est par exemple le cas de modèles de machines à états, représentant le système comme un ensemble d'états et de transitions qui, à partir d'un état courant et une valeur d'entrée, détermine le nouvel état du système. Il y a également les modèles basés sur les matrices d'accès, manipulant les trois concepts fondamentaux que sont les sujets, les objets et les actions. Les éléments qui se situent au croisement de la ligne L et de la colonne C correspondent aux droits possédés par le sujet correspondant à L sur l'objet de C ;

Les modèles de sécurité peuvent être classés en deux grandes familles :

- des modèles généraux, qui sont plutôt des méthodes de description formelle, pouvant s'appliquer à toute sorte de politiques. C'est par exemple le cas de modèles de machines à états, représentant le système comme un ensemble d'états et de transitions qui, à partir d'un état courant et une valeur d'entrée, détermine le nouvel état du système. Il y a également les modèles basés sur les matrices d'accès, manipulant les trois concepts fondamentaux que sont les sujets, les objets et les actions. Les éléments qui se situent au croisement de la ligne L et de la colonne C correspondent aux droits possédés par le sujet correspondant à L sur l'objet de C ;

Les modèles de sécurité peuvent être classés en deux grandes familles :

- des modèles généraux, qui sont plutôt des méthodes de description formelle, pouvant s'appliquer à toute sorte de politiques. C'est par exemple le cas de modèles de machines à états, représentant le système comme un ensemble d'états et de transitions qui, à partir d'un état courant et une valeur d'entrée, détermine le nouvel état du système. Il y a également les modèles basés sur les matrices d'accès, manipulant les trois concepts fondamentaux que sont les sujets, les objets et les actions. Les éléments qui se situent au croisement de la ligne L et de la colonne C correspondent aux droits possédés par le sujet correspondant à L sur l'objet de C ;

Les modèles de sécurité peuvent être classés en deux grandes familles :

- des modèles généraux, qui sont plutôt des méthodes de description formelle, pouvant s'appliquer à toute sorte de politiques. C'est par exemple le cas de modèles de machines à états, représentant le système comme un ensemble d'états et de transitions qui, à partir d'un état courant et une valeur d'entrée, détermine le nouvel état du système. Il y a également les modèles basés sur les matrices d'accès, manipulant les trois concepts fondamentaux que sont les sujets, les objets et les actions. Les éléments qui se situent au croisement de la ligne L et de la colonne C correspondent aux droits possédés par le sujet correspondant à L sur l'objet de C ;

Les modèles de sécurité peuvent être classés en deux grandes familles :

- des modèles généraux, qui sont plutôt des méthodes de description formelle, pouvant s'appliquer à toute sorte de politiques. C'est par exemple le cas de modèles de machines à états, représentant le système comme un ensemble d'états et de transitions qui, à partir d'un état courant et une valeur d'entrée, détermine le nouvel état du système. Il y a également les modèles basés sur les matrices d'accès, manipulant les trois concepts fondamentaux que sont les sujets, les objets et les actions. Les éléments qui se situent au croisement de la ligne L et de la colonne C correspondent aux droits possédés par le sujet correspondant à L sur l'objet de C ;

- des modèles spécifiques, développés pour représenter une politique d'autorisation particulière. Citons à titre d'exemple les modèles fondés sur les treillis, qui affectent à chaque utilisateur et à chaque objet un niveau de sécurité précis.

- des modèles spécifiques, développés pour représenter une politique d'autorisation particulière. Citons à titre d'exemple les modèles fondés sur les treillis, qui affectent à chaque utilisateur et à chaque objet un niveau de sécurité précis.

Dans le cas d'une politique discrétionnaire, les droits d'accès à chaque information sont manipulés librement par le responsable de l'information (généralement le propriétaire), à sa discrétion. Les droits peuvent être accordés par ce responsable à chaque utilisateur, à des groupes d'utilisateurs, ou bien aux deux. Ceci peut parfois amener le système dans un état d'insécurité (c'est-à-dire contraire aux objectifs de sécurité qui ont été choisis).

Dans le cas d'une politique discrétionnaire, les droits d'accès à chaque information sont manipulés librement par le responsable de l'information (généralement le propriétaire), à sa discrétion. Les droits peuvent être accordés par ce responsable à chaque utilisateur, à des groupes d'utilisateurs, ou bien aux deux. Ceci peut parfois amener le système dans un état d'insécurité (c'est-à-dire contraire aux objectifs de sécurité qui ont été choisis).

Dans le cas d'une politique discrétionnaire, les droits d'accès à chaque information sont manipulés librement par le responsable de l'information (généralement le propriétaire), à sa discrétion. Les droits peuvent être accordés par ce responsable à chaque utilisateur, à des groupes d'utilisateurs, ou bien aux deux. Ceci peut parfois amener le système dans un état d'insécurité (c'est-à-dire contraire aux objectifs de sécurité qui ont été choisis).

Prenons un exemple simple, reposant sur les mécanismes de protection d'Unix. Dans un tel système, les droits d'accès à un fichier sont définis et modifiables librement par l'utilisateur propriétaire du fichier (et donc par les processus qui s'exécutent pour son compte). Supposons que le schéma d'autorisation se définisse (informellement) de la manière suivante : un utilisateur peut créer des fichiers dont il devient alors propriétaire ; le propriétaire d'un fichier peut décider quels utilisateurs sont autorisés à lire ses fichiers. D'autre part, supposons que la politique exige de respecter l'objectif suivant : les utilisateurs qui n'ont pas le droit de lire un fichier ne doivent pas pouvoir en connaître le contenu.

Prenons un exemple simple, reposant sur les mécanismes de protection d'Unix. Dans un tel système, les droits d'accès à un fichier sont définis et modifiables librement par l'utilisateur propriétaire du fichier (et donc par les processus qui s'exécutent pour son compte).

Supposons que le schéma d'autorisation se définisse (informellement) de la manière suivante : un utilisateur peut créer des fichiers dont il devient alors propriétaire ; le propriétaire d'un fichier peut décider quels utilisateurs sont autorisés à lire ses fichiers. D'autre part, supposons que la politique exige de respecter l'objectif suivant : les utilisateurs qui n'ont pas le droit de lire un fichier ne doivent pas pouvoir en connaître le contenu.

Prenons un exemple simple, reposant sur les mécanismes de protection d'Unix. Dans un tel système, les droits d'accès à un fichier sont définis et modifiables librement par l'utilisateur propriétaire du fichier (et donc par les processus qui s'exécutent pour son compte). Supposons que le schéma d'autorisation se définisse (informellement) de la manière suivante : un utilisateur peut créer des fichiers dont il devient alors propriétaire ; le propriétaire d'un fichier peut décider quels utilisateurs sont autorisés à lire ses fichiers. D'autre part, supposons que la politique exige de respecter l'objectif suivant : les utilisateurs qui n'ont pas le droit de lire un fichier ne doivent pas pouvoir en connaître le contenu.

Prenons un exemple simple, reposant sur les mécanismes de protection d'Unix. Dans un tel système, les droits d'accès à un fichier sont définis et modifiables librement par l'utilisateur propriétaire du fichier (et donc par les processus qui s'exécutent pour son compte). Supposons que le schéma d'autorisation se définisse (informellement) de la manière suivante : un utilisateur peut créer des fichiers dont il devient alors propriétaire ; le propriétaire d'un fichier peut décider quels utilisateurs sont autorisés à lire ses fichiers. D'autre part, supposons que la politique exige de respecter l'objectif suivant : les utilisateurs qui n'ont pas le droit de lire un fichier ne doivent pas pouvoir en connaître le contenu.

Prenons un exemple simple, reposant sur les mécanismes de protection d'Unix. Dans un tel système, les droits d'accès à un fichier sont définis et modifiables librement par l'utilisateur propriétaire du fichier (et donc par les processus qui s'exécutent pour son compte). Supposons que le schéma d'autorisation se définisse (informellement) de la manière suivante : un utilisateur peut créer des fichiers dont il devient alors propriétaire ; le propriétaire d'un fichier peut décider quels utilisateurs sont autorisés à lire ses fichiers. D'autre part, supposons que la politique exige de respecter l'objectif suivant : les utilisateurs qui n'ont pas le droit de lire un fichier ne doivent pas pouvoir en connaître le contenu.

Prenons un exemple simple, reposant sur les mécanismes de protection d'Unix. Dans un tel système, les droits d'accès à un fichier sont définis et modifiables librement par l'utilisateur propriétaire du fichier (et donc par les processus qui s'exécutent pour son compte). Supposons que le schéma d'autorisation se définisse (informellement) de la manière suivante : un utilisateur peut créer des fichiers dont il devient alors propriétaire ; le propriétaire d'un fichier peut décider quels utilisateurs sont autorisés à lire ses fichiers. D'autre part, supposons que la politique exige de respecter l'objectif suivant : les utilisateurs qui n'ont pas le droit de lire un fichier ne doivent pas pouvoir en connaître le contenu.

Prenons un exemple simple, reposant sur les mécanismes de protection d'Unix. Dans un tel système, les droits d'accès à un fichier sont définis et modifiables librement par l'utilisateur propriétaire du fichier (et donc par les processus qui s'exécutent pour son compte). Supposons que le schéma d'autorisation se définisse (informellement) de la manière suivante : un utilisateur peut créer des fichiers dont il devient alors propriétaire ; le propriétaire d'un fichier peut décider quels utilisateurs sont autorisés à lire ses fichiers. D'autre part, supposons que la politique exige de respecter l'objectif suivant : les utilisateurs qui n'ont pas le droit de lire un fichier ne doivent pas pouvoir en connaître le contenu.

Une telle politique n'est pas réalisable par des mécanismes d'autorisation discrétionnaire, parce que :

- si s_1 est un sujet s'exécutant pour le compte de l'utilisateur u_1 propriétaire du fichier f_1, il peut donner au sujet s_2 (s'exécutant pour le compte d'u_2) le droit de lecture sur f_1 : ce que l'on note par $(s_1, f_1, \text{propriétaire}) \rightarrow (s_2, f_1, \text{lire})$;
- s_2 peut créer un fichier f_2 (dans lequel il peut donc écrire) sur lequel il peut donner le droit de lecture à s_3 (s'exécutant pour le compte d'u_3) : $(s_2, f_2, \text{créer}) \rightarrow (s_2, f_2, \text{écrire})$ et (s_3, f_2, lire) ;
- s_2 peut alors recopier f_1 dans f_2 pour transmettre les informations de f_1 à s_3 à l'insu du propriétaire s_1 :
  (s_2, f_1, lire) et $(s_2, f_2, \text{écrire})$ et $(s_3, f_2, \text{lire}) \rightarrow (s_3, c(f_1), \text{lire})$ où $(c(f_1)$ désigne une copie de $f_1)$

Une telle politique n'est pas réalisable par des mécanismes d'autorisation discrétionnaire, parce que :

- si s_1 est un sujet s'exécutant pour le compte de l'utilisateur u_1 propriétaire du fichier f_1, il peut donner au sujet s_2 (s'exécutant pour le compte d'u_2) le droit de lecture sur f_1 : ce que l'on note par $(s_1, f_1, \text{propriétaire}) \rightarrow (s_2, f_1, \text{lire})$;
- s_2 peut créer un fichier f_2 (dans lequel il peut donc écrire) sur lequel il peut donner le droit de lecture à s_3 (s'exécutant pour le compte d'u_3) : $(s_2, f_2, \text{créer}) \rightarrow (s_2, f_2, \text{écrire})$ et (s_3, f_2, lire) ;
- s_2 peut alors recopier f_1 dans f_2 pour transmettre les informations de f_1 à s_3 à l'insu du propriétaire s_1 :
  (s_2, f_1, lire) et $(s_2, f_2, \text{écrire})$ et $(s_3, f_2, \text{lire}) \rightarrow (s_3, c(f_1), \text{lire})$ où $(c(f_1)$ désigne une copie de $f_1)$

Une telle politique n'est pas réalisable par des mécanismes d'autorisation discrétionnaire, parce que :

- si s_1 est un sujet s'exécutant pour le compte de l'utilisateur u_1 propriétaire du fichier f_1, il peut donner au sujet s_2 (s'exécutant pour le compte d'u_2) le droit de lecture sur f_1 : ce que l'on note par $(s_1, f_1, \text{propriétaire}) \rightarrow (s_2, f_1, \text{lire})$;
- s_2 peut créer un fichier f_2 (dans lequel il peut donc écrire) sur lequel il peut donner le droit de lecture à s_3 (s'exécutant pour le compte d'u_3) : $(s_2, f_2, \text{créer}) \rightarrow (s_2, f_2, \text{écrire})$ et (s_3, f_2, lire) ;
- s_2 peut alors recopier f_1 dans f_2 pour transmettre les informations de f_1 à s_3 à l'insu du propriétaire s_1 :
  (s_2, f_1, lire) et $(s_2, f_2, \text{écrire})$ et $(s_3, f_2, \text{lire}) \rightarrow (s_3, c(f_1), \text{lire})$ où $(c(f_1)$ désigne une copie de $f_1)$

Une telle politique n'est pas réalisable par des mécanismes d'autorisation discrétionnaire, parce que :

- si s_1 est un sujet s'exécutant pour le compte de l'utilisateur u_1 propriétaire du fichier f_1, il peut donner au sujet s_2 (s'exécutant pour le compte d'u_2) le droit de lecture sur f_1 : ce que l'on note par $(s_1, f_1, \text{propriétaire}) \rightarrow (s_2, f_1, \text{lire})$;
- s_2 peut créer un fichier f_2 (dans lequel il peut donc écrire) sur lequel il peut donner le droit de lecture à s_3 (s'exécutant pour le compte d'u_3) : $(s_2, f_2, \text{créer}) \rightarrow (s_2, f_2, \text{écrire})$ et (s_3, f_2, lire) ;
- s_2 peut alors recopier f_1 dans f_2 pour transmettre les informations de f_1 à s_3 à l'insu du propriétaire s_1 :
  (s_2, f_1, lire) et $(s_2, f_2, \text{écrire})$ et $(s_3, f_2, \text{lire}) \rightarrow (s_3, c(f_1), \text{lire})$ où $(c(f_1)$ désigne une copie de $f_1)$

Une telle politique n'est pas réalisable par des mécanismes d'autorisation discrétionnaire, parce que :

- si s_1 est un sujet s'exécutant pour le compte de l'utilisateur u_1 propriétaire du fichier f_1, il peut donner au sujet s_2 (s'exécutant pour le compte d'u_2) le droit de lecture sur f_1 : ce que l'on note par $(s_1, f_1, \text{propriétaire}) \rightarrow (s_2, f_1, \text{lire})$;
- s_2 peut créer un fichier f_2 (dans lequel il peut donc écrire) sur lequel il peut donner le droit de lecture à s_3 (s'exécutant pour le compte d'u_3) : $(s_2, f_2, \text{créer}) \rightarrow (s_2, f_2, \text{écrire})$ et (s_3, f_2, lire) ;
- s_2 peut alors recopier f_1 dans f_2 pour transmettre les informations de f_1 à s_3 à l'insu du propriétaire s_1 :
  (s_2, f_1, lire) et $(s_2, f_2, \text{écrire})$ et $(s_3, f_2, \text{lire}) \rightarrow (s_3, c(f_1), \text{lire})$ où $(c(f_1)$ désigne une copie de $f_1)$

Une telle politique n'est pas réalisable par des mécanismes d'autorisation discrétionnaire, parce que :

- si s_1 est un sujet s'exécutant pour le compte de l'utilisateur u_1 propriétaire du fichier f_1, il peut donner au sujet s_2 (s'exécutant pour le compte d'u_2) le droit de lecture sur f_1 : ce que l'on note par $(s_1, f_1, \text{propriétaire}) \rightarrow (s_2, f_1, \text{lire})$;
- s_2 peut créer un fichier f_2 (dans lequel il peut donc écrire) sur lequel il peut donner le droit de lecture à s_3 (s'exécutant pour le compte d'u_3) : $(s_2, f_2, \text{créer}) \rightarrow (s_2, f_2, \text{écrire})$ et (s_3, f_2, lire) ;
- s_2 peut alors recopier f_1 dans f_2 pour transmettre les informations de f_1 à s_3 à l'insu du propriétaire s_1 :
  (s_2, f_1, lire) et $(s_2, f_2, \text{écrire})$ et $(s_3, f_2, \text{lire}) \rightarrow (s_3, c(f_1), \text{lire})$ où $(c(f_1)$ désigne une copie de $f_1)$

Une telle politique n'est pas réalisable par des mécanismes d'autorisation discrétionnaire, parce que :

- si s_1 est un sujet s'exécutant pour le compte de l'utilisateur u_1 propriétaire du fichier f_1, il peut donner au sujet s_2 (s'exécutant pour le compte d'u_2) le droit de lecture sur f_1 : ce que l'on note par $(s_1, f_1, \text{propriétaire}) \rightarrow (s_2, f_1, \text{lire})$;
- s_2 peut créer un fichier f_2 (dans lequel il peut donc écrire) sur lequel il peut donner le droit de lecture à s_3 (s'exécutant pour le compte d'u_3) : $(s_2, f_2, \text{créer}) \rightarrow (s_2, f_2, \text{écrire})$ et (s_3, f_2, lire) ;
- s_2 peut alors recopier f_1 dans f_2 pour transmettre les informations de f_1 à s_3 à l'insu du propriétaire s_1 :
  (s_2, f_1, lire) et $(s_2, f_2, \text{écrire})$ et $(s_3, f_2, \text{lire}) \rightarrow (s_3, c(f_1), \text{lire})$ où $(c(f_1)$ désigne une copie de $f_1)$

Une politique discrétionnaire n'est donc applicable que dans la mesure où il est possible de faire totalement confiance aux utilisateurs et aux sujets qui s'exécutent pour leur compte. Une telle politique est par là même vulnérable aux abus de pouvoir provoqués par maladresse ou par malveillance. Ainsi, s'il est possible à un utilisateur d'accéder à certains objets ou d'en modifier les droits d'accès, il est possible qu'un cheval de Troie s'exécutant pour le compte de cet utilisateur (à son insu) en fasse de même. De plus, si un utilisateur a le droit de lire une information, il a (en général) le droit de la transmettre à n'importe qui.

Une politique discrétionnaire n'est donc applicable que dans la mesure où il est possible de faire totalement confiance aux utilisateurs et aux sujets qui s'exécutent pour leur compte. Une telle politique est par là même vulnérable aux abus de pouvoir provoqués par maladresse ou par malveillance. Ainsi, s'il est possible à un utilisateur d'accéder à certains objets ou d'en modifier les droits d'accès, il est possible qu'un cheval de Troie s'exécutant pour le compte de cet utilisateur (à son insu) en fasse de même. De plus, si un utilisateur a le droit de lire une information, il a (en général) le droit de la transmettre à n'importe qui.

Une politique discrétionnaire n'est donc applicable que dans la mesure où il est possible de faire totalement confiance aux utilisateurs et aux sujets qui s'exécutent pour leur compte. Une telle politique est par là même vulnérable aux abus de pouvoir provoqués par maladresse ou par malveillance. Ainsi, s'il est possible à un utilisateur d'accéder à certains objets ou d'en modifier les droits d'accès, il est possible qu'un cheval de Troie s'exécutant pour le compte de cet utilisateur (à son insu) en fasse de même. De plus, si un utilisateur a le droit de lire une information, il a (en général) le droit de la transmettre à n'importe qui.

Une politique discrétionnaire n'est donc applicable que dans la mesure où il est possible de faire totalement confiance aux utilisateurs et aux sujets qui s'exécutent pour leur compte. Une telle politique est par là même vulnérable aux abus de pouvoir provoqués par maladresse ou par malveillance. Ainsi, s'il est possible à un utilisateur d'accéder à certains objets ou d'en modifier les droits d'accès, il est possible qu'un cheval de Troie s'exécutant pour le compte de cet utilisateur (à son insu) en fasse de même. De plus, si un utilisateur a le droit de lire une information, il a (en général) le droit de la transmettre à n'importe qui.

Une politique discrétionnaire n'est donc applicable que dans la mesure où il est possible de faire totalement confiance aux utilisateurs et aux sujets qui s'exécutent pour leur compte. Une telle politique est par là même vulnérable aux abus de pouvoir provoqués par maladresse ou par malveillance. Ainsi, s'il est possible à un utilisateur d'accéder à certains objets ou d'en modifier les droits d'accès, il est possible qu'un cheval de Troie s'exécutant pour le compte de cet utilisateur (à son insu) en fasse de même. De plus, si un utilisateur a le droit de lire une information, il a (en général) le droit de la transmettre à n'importe qui.

Une politique discrétionnaire n'est donc applicable que dans la mesure où il est possible de faire totalement confiance aux utilisateurs et aux sujets qui s'exécutent pour leur compte. Une telle politique est par là même vulnérable aux abus de pouvoir provoqués par maladresse ou par malveillance. Ainsi, s'il est possible à un utilisateur d'accéder à certains objets ou d'en modifier les droits d'accès, il est possible qu'un cheval de Troie s'exécutant pour le compte de cet utilisateur (à son insu) en fasse de même. De plus, si un utilisateur a le droit de lire une information, il a (en général) le droit de la transmettre à n'importe qui.

Modèle de Lampson (1/2)

La notion de matrice de contrôle d'accès, dédiée à la représentation des droits d'accès (autorisations), a été introduite par Lampson dès 1971 (B. Lampson, " Protection ", 5th Princeton Symposium on Information Sciences and Systems, 1971.). La structure de ce modèle est celle d'une machine à états où chaque état est un triplet (S,O,M) avec : S désignant un ensemble de sujets, O un ensemble d'objets et M une matrice de contrôle d'accès. Chaque cellule M(s,o) de cette matrice contient les droits d'accès que le sujet s possède sur l'objet o. Les droits correspondent généralement à des actions élémentaires telles que " lire " ou " écrire ".

Modèle de Lampson (1/2)

La notion de matrice de contrôle d'accès, dédiée à la représentation des droits d'accès (autorisations), a été introduite par Lampson dès 1971 (B. Lampson, " Protection ", 5th Princeton Symposium on Information Sciences and Systems, 1971.). La structure de ce modèle est celle d'une machine à états où chaque état est un triplet (S,O,M) avec : S désignant un ensemble de sujets, O un ensemble d'objets et M une matrice de contrôle d'accès. Chaque cellule M(s,o) de cette matrice contient les droits d'accès que le sujet s possède sur l'objet o. Les droits correspondent généralement à des actions élémentaires telles que " lire " ou " écrire ".

Modèle de Lampson (1/2)

La notion de matrice de contrôle d'accès, dédiée à la représentation des droits d'accès (autorisations), a été introduite par Lampson dès 1971 (B. Lampson, " Protection ", 5th Princeton Symposium on Information Sciences and Systems, 1971.). La structure de ce modèle est celle d'une machine à états où chaque état est un triplet (S,O,M) avec : S désignant un ensemble de sujets, O un ensemble d'objets et M une matrice de contrôle d'accès. Chaque cellule M(s,o) de cette matrice contient les droits d'accès que le sujet s possède sur l'objet o. Les droits correspondent généralement à des actions élémentaires telles que " lire " ou " écrire ".

Modèle de Lampson (1/2)

La notion de matrice de contrôle d'accès, dédiée à la représentation des droits d'accès (autorisations), a été introduite par Lampson dès 1971 (B. Lampson, " Protection ", 5th Princeton Symposium on Information Sciences and Systems, 1971.). La structure de ce modèle est celle d'une machine à états où chaque état est un triplet (S,O,M) avec : S désignant un ensemble de sujets, O un ensemble d'objets et M une matrice de contrôle d'accès. Chaque cellule M(s,o) de cette matrice contient les droits d'accès que le sujet s possède sur l'objet o. Les droits correspondent généralement à des actions élémentaires telles que " lire " ou " écrire ".

Modèle de Lampson (1/2)

La notion de matrice de contrôle d'accès, dédiée à la représentation des droits d'accès (autorisations), a été introduite par Lampson dès 1971 (B. Lampson, " Protection ", 5th Princeton Symposium on Information Sciences and Systems, 1971.). La structure de ce modèle est celle d'une machine à états où chaque état est un triplet (S,O,M) avec : S désignant un ensemble de sujets, O un ensemble d'objets et M une matrice de contrôle d'accès. Chaque cellule M(s,o) de cette matrice contient les droits d'accès que le sujet s possède sur l'objet o. Les droits correspondent généralement à des actions élémentaires telles que " lire " ou " écrire ".

Modèle de Lampson (1/2)

La notion de matrice de contrôle d'accès, dédiée à la représentation des droits d'accès (autorisations), a été introduite par Lampson dès 1971 (B. Lampson, " Protection ", 5th Princeton Symposium on Information Sciences and Systems, 1971.). La structure de ce modèle est celle d'une machine à états où chaque état est un triplet (S,O,M) avec : S désignant un ensemble de sujets, O un ensemble d'objets et M une matrice de contrôle d'accès. Chaque cellule M(s,o) de cette matrice contient les droits d'accès que le sujet s possède sur l'objet o. Les droits correspondent généralement à des actions élémentaires telles que " lire " ou " écrire ".

Modèle de Lampson (1/2)

La notion de matrice de contrôle d'accès, dédiée à la représentation des droits d'accès (autorisations), a été introduite par Lampson dès 1971 (B. Lampson, " Protection ", 5th Princeton Symposium on Information Sciences and Systems, 1971.). La structure de ce modèle est celle d'une machine à états où chaque état est un triplet (S,O,M) avec : S désignant un ensemble de sujets, O un ensemble d'objets et M une matrice de contrôle d'accès. Chaque cellule M(s,o) de cette matrice contient les droits d'accès que le sujet s possède sur l'objet o. Les droits correspondent généralement à des actions élémentaires telles que " lire " ou " écrire ".

Modèle de Lampson (1/2)

La notion de matrice de contrôle d'accès, dédiée à la représentation des droits d'accès (autorisations), a été introduite par Lampson dès 1971 (B. Lampson, " Protection ", 5th Princeton Symposium on Information Sciences and Systems, 1971.). La structure de ce modèle est celle d'une machine à états où chaque état est un triplet (S,O,M) avec : S désignant un ensemble de sujets, O un ensemble d'objets et M une matrice de contrôle d'accès. Chaque cellule M(s,o) de cette matrice contient les droits d'accès que le sujet s possède sur l'objet o. Les droits correspondent généralement à des actions élémentaires telles que " lire " ou " écrire ".

Modèle de Lampson (1/2)

La notion de matrice de contrôle d'accès, dédiée à la représentation des droits d'accès (autorisations), a été introduite par Lampson dès 1971 (B. Lampson, " Protection ", 5th Princeton Symposium on Information Sciences and Systems, 1971.). La structure de ce modèle est celle d'une machine à états où chaque état est un triplet (S,O,M) avec : S désignant un ensemble de sujets, O un ensemble d'objets et M une matrice de contrôle d'accès. Chaque cellule M(s,o) de cette matrice contient les droits d'accès que le sujet s possède sur l'objet o. Les droits correspondent généralement à des actions élémentaires telles que " lire " ou " écrire ".

Modèle de Lampson (1/2)

La notion de matrice de contrôle d'accès, dédiée à la représentation des droits d'accès (autorisations), a été introduite par Lampson dès 1971 (B. Lampson, " Protection ", 5th Princeton Symposium on Information Sciences and Systems, 1971.). La structure de ce modèle est celle d'une machine à états où chaque état est un triplet (S,O,M) avec : S désignant un ensemble de sujets, O un ensemble d'objets et M une matrice de contrôle d'accès. Chaque cellule M(s,o) de cette matrice contient les droits d'accès que le sujet s possède sur l'objet o. Les droits correspondent généralement à des actions élémentaires telles que " lire " ou " écrire ".

Modèle de Lampson (2/2)

La matrice des droits d'accès n'est pas figée. En effet, si de nouveaux objets, de nouveaux sujets ou de nouvelles actions sont ajoutées dans le système, il devient alors nécessaire d'enregistrer toutes les permissions accordées pour ces nouvelles entités. Par conséquent, la mise à jour d'une politique de sécurité exprimée par ce modèle est quelque peu fastidieuse. Enfin, ce modèle ne permet pas d'exprimer directement des interdictions ou des obligations.

Le modèle de Lampson a été progressivement amélioré pour donner naissance à d'autres modèles tels que HRU (1976) et Take-Grant (1976).

Modèle de Lampson (2/2)

La matrice des droits d'accès n'est pas figée. En effet, si de nouveaux objets, de nouveaux sujets ou de nouvelles actions sont ajoutées dans le système, il devient alors nécessaire d'enregistrer toutes les permissions accordées pour ces nouvelles entités. Par conséquent, la mise à jour d'une politique de sécurité exprimée par ce modèle est quelque peu fastidieuse. Enfin, ce modèle ne permet pas d'exprimer directement des interdictions ou des obligations.

Le modèle de Lampson a été progressivement amélioré pour donner naissance à d'autres modèles tels que HRU (1976) et Take-Grant (1976).

Modèle de Lampson (2/2)

La matrice des droits d'accès n'est pas figée. En effet, si de nouveaux objets, de nouveaux sujets ou de nouvelles actions sont ajoutées dans le système, il devient alors nécessaire d'enregistrer toutes les permissions accordées pour ces nouvelles entités. Par conséquent, la mise à jour d'une politique de sécurité exprimée par ce modèle est quelque peu fastidieuse. Enfin, ce modèle ne permet pas d'exprimer directement des interdictions ou des obligations.

Le modèle de Lampson a été progressivement amélioré pour donner naissance à d'autres modèles tels que HRU (1976) et Take-Grant (1976).

Modèle de Lampson (2/2)

La matrice des droits d'accès n'est pas figée. En effet, si de nouveaux objets, de nouveaux sujets ou de nouvelles actions sont ajoutées dans le système, il devient alors nécessaire d'enregistrer toutes les permissions accordées pour ces nouvelles entités. Par conséquent, la mise à jour d'une politique de sécurité exprimée par ce modèle est quelque peu fastidieuse. Enfin, ce modèle ne permet pas d'exprimer directement des interdictions ou des obligations.

Le modèle de Lampson a été progressivement amélioré pour donner naissance à d'autres modèles tels que HRU (1976) et Take-Grant (1976).

Modèle de Lampson (2/2)

La matrice des droits d'accès n'est pas figée. En effet, si de nouveaux objets, de nouveaux sujets ou de nouvelles actions sont ajoutées dans le système, il devient alors nécessaire d'enregistrer toutes les permissions accordées pour ces nouvelles entités. Par conséquent, la mise à jour d'une politique de sécurité exprimée par ce modèle est quelque peu fastidieuse. Enfin, ce modèle ne permet pas d'exprimer directement des interdictions ou des obligations.

Le modèle de Lampson a été progressivement amélioré pour donner naissance à d'autres modèles tels que HRU (1976) et Take-Grant (1976).

Modèle de Lampson (2/2)

La matrice des droits d'accès n'est pas figée. En effet, si de nouveaux objets, de nouveaux sujets ou de nouvelles actions sont ajoutées dans le système, il devient alors nécessaire d'enregistrer toutes les permissions accordées pour ces nouvelles entités. Par conséquent, la mise à jour d'une politique de sécurité exprimée par ce modèle est quelque peu fastidieuse. Enfin, ce modèle ne permet pas d'exprimer directement des interdictions ou des obligations.

Le modèle de Lampson a été progressivement amélioré pour donner naissance à d'autres modèles tels que HRU (1976) et Take-Grant (1976).

Modèle HRU (1/2)

Le Modèle HRU a été introduit par M.A. Harrison, W.L. Ruzzo et J.D. Ullman dans leur article M.A. Harrison, W.L. Ruzzo et J.D. Ullman, " Protection in Operating Systems ", Communication of the ACM, 19(8), pp. 461-471, daté d'août 1976. Comme dans le modèle de Lampson, HRU utilise une matrice d'accès classique, la différence réside en ce que HRU précise les commandes qui peuvent lui être appliquées. Les seules opérations possibles sont données dans le tableau suivant :

```
Enter a into M(s,o)     delete a from M(s,o)
Create subject s        destroy subject s          (1)
Create object o         destroy object o
```

où a est un droit.

Modèle HRU (1/2)

Le Modèle HRU a été introduit par M.A. Harrison, W.L. Ruzzo et J.D. Ullman dans leur article M.A. Harrison, W.L. Ruzzo et J.D. Ullman, " Protection in Operating Systems ", Communication of the ACM, 19(8), pp. 461-471, daté d'août 1976. Comme dans le modèle de Lampson, HRU utilise une matrice d'accès classique, la différence réside en ce que HRU précise les commandes qui peuvent lui être appliquées. Les seules opérations possibles sont données dans le tableau suivant :

Enter a into M(s,o)	delete a from M(s,o)
Create subject s	destroy subject s
Create object o	destroy object o

(1)

où a est un droit.

Modèle HRU (1/2)

Le Modèle HRU a été introduit par M.A. Harrison, W.L. Ruzzo et J.D. Ullman dans leur article M.A. Harrison, W.L. Ruzzo et J.D. Ullman, " Protection in Operating Systems ", Communication of the ACM, 19(8), pp. 461-471, daté d'août 1976. Comme dans le modèle de Lampson, HRU utilise une matrice d'accès classique, la différence réside en ce que HRU précise les commandes qui peuvent lui être appliquées. Les seules opérations possibles sont données dans le tableau suivant :

`Enter a into M(s,o)`	`delete a from M(s,o)`
`Create subject s`	`destroy subject s`
`Create object o`	`destroy object o`

(1)

où `a` est un droit.

Modèle HRU (1/2)

Le Modèle HRU a été introduit par M.A. Harrison, W.L. Ruzzo et J.D. Ullman dans leur article M.A. Harrison, W.L. Ruzzo et J.D. Ullman, " Protection in Operating Systems ", Communication of the ACM, 19(8), pp. 461-471, daté d'août 1976. Comme dans le modèle de Lampson, HRU utilise une matrice d'accès classique, la différence réside en ce que HRU précise les commandes qui peuvent lui être appliquées. Les seules opérations possibles sont données dans le tableau suivant :

`Enter a into M(s,o)`	`delete a from M(s,o)`
`Create subject s`	`destroy subject s`
`Create object o`	`destroy object o`

(1)

où `a` est un droit.

Modèle HRU (1/2)

Le Modèle HRU a été introduit par M.A. Harrison, W.L. Ruzzo et J.D. Ullman dans leur article M.A. Harrison, W.L. Ruzzo et J.D. Ullman, " Protection in Operating Systems ", Communication of the ACM, 19(8), pp. 461-471, daté d'août 1976. Comme dans le modèle de Lampson, HRU utilise une matrice d'accès classique, la différence réside en ce que HRU précise les commandes qui peuvent lui être appliquées. Les seules opérations possibles sont données dans le tableau suivant :

`Enter a into M(s,o)`	`delete a from M(s,o)`
`Create subject s`	`destroy subject s`
`Create object o`	`destroy object o`

(1)

où `a` est un droit.

Modèle HRU (2/2)

Dans le modèle HRU, une commande permet d'effectuer, sous certaines conditions, les opérations élémentaires pour la mise à jour de la matrice d'accès de façon automatisée.

Modèle Take-Grant (1/2)

Avant tout rappelons qu'un graphe est une figure graphique définie par ensemble de points (appelés nœuds ou sommets) reliés entre eux par des arcs (c'est-à-dire des flêches qui relient une source, l'un des nœuds, à une destination, un autre nœud).

Le modèle Take-Grant (A.K. Jones, R.J. Lipton, L. Snyder, " A Linear Time Algorithm for Deciding Security ", 17th Annual Symposium on Fundations of Computer Science, Houston, USA, 1976.) est constitué d'un graphe dont les nœuds sont des sujets ou des objets, et des règles de modification de ce graphe. Le graphe est une autre représentation de la matrice d'accès. On a un arc entre un sujet s et un objet o lorsqu'existe un droit d'accès α de s sur o. L'arc est alors étiqueté par le droit α.

Modèle Take-Grant (1/2)

Avant tout rappelons qu'un graphe est une figure graphique définie par ensemble de points (appelés nœuds ou sommets) reliés entre eux par des arcs (c'est-à-dire des flêches qui relient une source, l'un des nœuds, à une destination, un autre nœud).

Le modèle Take-Grant (A.K. Jones, R.J. Lipton, L. Snyder, " A Linear Time Algorithm for Deciding Security ", 17th Annual Symposium on Fundations of Computer Science, Houston, USA, 1976.) est constitué d'un graphe dont les nœuds sont des sujets ou des objets, et des règles de modification de ce graphe. Le graphe est une autre représentation de la matrice d'accès. On a un arc entre un sujet s et un objet o lorsqu'existe un droit d'accès α de s sur o. L'arc est alors étiqueté par le droit α.

Modèle Take-Grant (1/2)

Avant tout rappelons qu'un graphe est une figure graphique définie par ensemble de points (appelés nœuds ou sommets) reliés entre eux par des arcs (c'est-à-dire des flêches qui relient une source, l'un des nœuds, à une destination, un autre nœud).

Le modèle Take-Grant (A.K. Jones, R.J. Lipton, L. Snyder, " A Linear Time Algorithm for Deciding Security ", 17th Annual Symposium on Fundations of Computer Science, Houston, USA, 1976.) est constitué d'un graphe dont les nœuds sont des sujets ou des objets, et des règles de modification de ce graphe. Le graphe est une autre représentation de la matrice d'accès. On a un arc entre un sujet s et un objet o lorsqu'existe un droit d'accès α de s sur o. L'arc est alors étiqueté par le droit α.

Modèle Take-Grant (1/2)

Avant tout rappelons qu'un graphe est une figure graphique définie par ensemble de points (appelés nœuds ou sommets) reliés entre eux par des arcs (c'est-à-dire des flêches qui relient une source, l'un des nœuds, à une destination, un autre nœud).

Le modèle Take-Grant (A.K. Jones, R.J. Lipton, L. Snyder, " A Linear Time Algorithm for Deciding Security ", 17th Annual Symposium on Fundations of Computer Science, Houston, USA, 1976.) est constitué d'un graphe dont les nœuds sont des sujets ou des objets, et des règles de modification de ce graphe. Le graphe est une autre représentation de la matrice d'accès. On a un arc entre un sujet s et un objet o lorsqu'existe un droit d'accès α de s sur o. L'arc est alors étiqueté par le droit α.

Modèle Take-Grant (1/2)

Avant tout rappelons qu'un graphe est une figure graphique définie par ensemble de points (appelés nœuds ou sommets) reliés entre eux par des arcs (c'est-à-dire des flêches qui relient une source, l'un des nœuds, à une destination, un autre nœud).

Le modèle Take-Grant (A.K. Jones, R.J. Lipton, L. Snyder, " A Linear Time Algorithm for Deciding Security ", 17th Annual Symposium on Fundations of Computer Science, Houston, USA, 1976.) est constitué d'un graphe dont les nœuds sont des sujets ou des objets, et des règles de modification de ce graphe. Le graphe est une autre représentation de la matrice d'accès. On a un arc entre un sujet s et un objet o lorsqu'existe un droit d'accès α de s sur o. L'arc est alors étiqueté par le droit α.

Modèle Take-Grant (1/2)

Avant tout rappelons qu'un graphe est une figure graphique définie par ensemble de points (appelés nœuds ou sommets) reliés entre eux par des arcs (c'est-à-dire des flêches qui relient une source, l'un des nœuds, à une destination, un autre nœud).

Le modèle Take-Grant (A.K. Jones, R.J. Lipton, L. Snyder, " A Linear Time Algorithm for Deciding Security ", 17th Annual Symposium on Fundations of Computer Science, Houston, USA, 1976.) est constitué d'un graphe dont les nœuds sont des sujets ou des objets, et des règles de modification de ce graphe. Le graphe est une autre représentation de la matrice d'accès. On a un arc entre un sujet s et un objet o lorsqu'existe un droit d'accès α de s sur o. L'arc est alors étiqueté par le droit α.

Modèle Take-Grant (2/2)

La mise à jour des droits d'accès, c'est-à-dire du graphe le représentant, s'effectue à l'aide des quatres commandes suivantes :

- la commande `create` qui permet de créer un objet et d'attribuer initialement un droit d'accès à un sujet sur cet objet ;
- la commande `remove` qui permet de retirer un droit d'accès d'un sujet sur un objet ;
- la commande " take ", représentée par un arc étiqueté par la lettre `t` entre un sujet s et un sujet (ou objet) r, indique que s peut prendre tous les droits que r possède ;
- la commande `grant` qui permet à un sujet s possédant un droit d'accès a sur o ainsi que le droit `g` (pour " grant ") sur un autre sujet r, de céder à r le droit a sur o (que s possède sur o).

On peut voir ces opérations sur un schéma (ImageTake-Grant1).

Modèle Take-Grant (2/2)

La mise à jour des droits d'accès, c'est-à-dire du graphe le représentant, s'effectue à l'aide des quatres commandes suivantes :

- la commande `create` qui permet de créer un objet et d'attribuer initialement un droit d'accès à un sujet sur cet objet ;
- la commande `remove` qui permet de retirer un droit d'accès d'un sujet sur un objet ;
- la commande " take ", représentée par un arc étiqueté par la lettre `t` entre un sujet s et un sujet (ou objet) r, indique que s peut prendre tous les droits que r possède ;
- la commande `grant` qui permet à un sujet s possédant un droit d'accès a sur o ainsi que le droit `g` (pour " grant ") sur un autre sujet r, de céder à r le droit a sur o (que s possède sur o).

On peut voir ces opérations sur un schéma (ImageTake-Grant1).

Modèle Take-Grant (2/2)

La mise à jour des droits d'accès, c'est-à-dire du graphe le représentant, s'effectue à l'aide des quatres commandes suivantes :

- la commande `create` qui permet de créer un objet et d'attribuer initialement un droit d'accès à un sujet sur cet objet ;
- la commande `remove` qui permet de retirer un droit d'accès d'un sujet sur un objet ;
- la commande " take ", représentée par un arc étiqueté par la lettre `t` entre un sujet s et un sujet (ou objet) r, indique que s peut prendre tous les droits que r possède ;
- la commande `grant` qui permet à un sujet s possédant un droit d'accès a sur o ainsi que le droit `g` (pour " grant ") sur un autre sujet r, de céder à r le droit a sur o (que s possède sur o).

On peut voir ces opérations sur un schéma (ImageTake-Grant1).

Modèle Take-Grant (2/2)

La mise à jour des droits d'accès, c'est-à-dire du graphe le représentant, s'effectue à l'aide des quatres commandes suivantes :

- la commande `create` qui permet de créer un objet et d'attribuer initialement un droit d'accès à un sujet sur cet objet ;
- la commande `remove` qui permet de retirer un droit d'accès d'un sujet sur un objet ;
- la commande " take ", représentée par un arc étiqueté par la lettre `t` entre un sujet *s* et un sujet (ou objet) *r*, indique que *s* peut prendre tous les droits que *r* possède ;
- la commande `grant` qui permet à un sujet *s* possédant un droit d'accès *a* sur *o* ainsi que le droit `g` (pour " grant ") sur un autre sujet *r*, de céder à *r* le droit *a* sur *o* (que *s* possède sur *o*).

On peut voir ces opérations sur un schéma (ImageTake-Grant1).

Modèle Take-Grant (2/2)

La mise à jour des droits d'accès, c'est-à-dire du graphe le représentant, s'effectue à l'aide des quatres commandes suivantes :

- la commande `create` qui permet de créer un objet et d'attribuer initialement un droit d'accès à un sujet sur cet objet ;
- la commande `remove` qui permet de retirer un droit d'accès d'un sujet sur un objet ;
- la commande " take ", représentée par un arc étiqueté par la lettre `t` entre un sujet *s* et un sujet (ou objet) *r*, indique que *s* peut prendre tous les droits que *r* possède ;
- la commande `grant` qui permet à un sujet *s* possédant un droit d'accès *a* sur *o* ainsi que le droit `g` (pour " grant ") sur un autre sujet *r*, de céder à *r* le droit *a* sur *o* (que *s* possède sur *o*).

On peut voir ces opérations sur un schéma (ImageTake-Grant1).

Modèle Take-Grant (2/2)

La mise à jour des droits d'accès, c'est-à-dire du graphe le représentant, s'effectue à l'aide des quatres commandes suivantes :

- la commande `create` qui permet de créer un objet et d'attribuer initialement un droit d'accès à un sujet sur cet objet ;
- la commande `remove` qui permet de retirer un droit d'accès d'un sujet sur un objet ;
- la commande " take ", représentée par un arc étiqueté par la lettre `t` entre un sujet *s* et un sujet (ou objet) *r*, indique que *s* peut prendre tous les droits que *r* possède ;
- la commande `grant` qui permet à un sujet *s* possédant un droit d'accès *a* sur *o* ainsi que le droit `g` (pour " grant ") sur un autre sujet *r*, de céder à *r* le droit *a* sur *o* (que *s* possède sur *o*).

On peut voir ces opérations sur un schéma (ImageTake-Grant1).

Une politique de sécurité d'autorisation obligatoire (ou MAC de l'anglais " Mandatory Access Control ") impose des règles d'autorisation incontournables qui s'ajoutent aux règles discrétionnaires. Une politique obligatoire suppose que les utilisateurs et objets aient été étiquetés. Classiquement, les objets se voient attribuer une classification, tandis que les utilisateurs possèdent une habilitation. Les règles qui régissent les autorisations d'accès sont basées sur une comparaison de l'habilitation de l'utilisateur et de la classification de l'objet. Ces règles incontournables assurent que le système vérifie des propriétés générales de confidentialité ou d'intégrité par exemple. Ces règles sont souvent utilisées conjointement aux règles d'une politique discrétionnaire car leur pouvoir d'expression est en général relativement faible.

Une politique de sécurité d'autorisation obligatoire (ou MAC de l'anglais " Mandatory Access Control ") impose des règles d'autorisation incontournables qui s'ajoutent aux règles discrétionnaires. Une politique obligatoire suppose que les utilisateurs et objets aient été étiquetés. Classiquement, les objets se voient attribuer une classification, tandis que les utilisateurs possèdent une habilitation. Les règles qui régissent les autorisations d'accès sont basées sur une comparaison de l'habilitation de l'utilisateur et de la classification de l'objet. Ces règles incontournables assurent que le système vérifie des propriétés générales de confidentialité ou d'intégrité par exemple. Ces règles sont souvent utilisées conjointement aux règles d'une politique discrétionnaire car leur pouvoir d'expression est en général relativement faible.

Une politique de sécurité d'autorisation obligatoire (ou MAC de l'anglais " Mandatory Access Control ") impose des règles d'autorisation incontournables qui s'ajoutent aux règles discrétionnaires. Une politique obligatoire suppose que les utilisateurs et objets aient été étiquetés. Classiquement, les objets se voient attribuer une classification, tandis que les utilisateurs possèdent une habilitation. Les règles qui régissent les autorisations d'accès sont basées sur une comparaison de l'habilitation de l'utilisateur et de la classification de l'objet. Ces règles incontournables assurent que le système vérifie des propriétés générales de confidentialité ou d'intégrité par exemple. Ces règles sont souvent utilisées conjointement aux règles d'une politique discrétionnaire car leur pouvoir d'expression est en général relativement faible.

Une politique de sécurité d'autorisation obligatoire (ou MAC de l'anglais " Mandatory Access Control ") impose des règles d'autorisation incontournables qui s'ajoutent aux règles discrétionnaires. Une politique obligatoire suppose que les utilisateurs et objets aient été étiquetés. Classiquement, les objets se voient attribuer une classification, tandis que les utilisateurs possèdent une habilitation. Les règles qui régissent les autorisations d'accès sont basées sur une comparaison de l'habilitation de l'utilisateur et de la classification de l'objet. Ces règles incontournables assurent que le système vérifie des propriétés générales de confidentialité ou d'intégrité par exemple. Ces règles sont souvent utilisées conjointement aux règles d'une politique discrétionnaire car leur pouvoir d'expression est en général relativement faible.

Une politique de sécurité d'autorisation obligatoire (ou MAC de l'anglais " Mandatory Access Control ") impose des règles d'autorisation incontournables qui s'ajoutent aux règles discrétionnaires. Une politique obligatoire suppose que les utilisateurs et objets aient été étiquetés. Classiquement, les objets se voient attribuer une classification, tandis que les utilisateurs possèdent une habilitation. Les règles qui régissent les autorisations d'accès sont basées sur une comparaison de l'habilitation de l'utilisateur et de la classification de l'objet. Ces règles incontournables assurent que le système vérifie des propriétés générales de confidentialité ou d'intégrité par exemple. Ces règles sont souvent utilisées conjointement aux règles d'une politique discrétionnaire car leur pouvoir d'expression est en général relativement faible.

Une politique de sécurité d'autorisation obligatoire (ou MAC de l'anglais " Mandatory Access Control ") impose des règles d'autorisation incontournables qui s'ajoutent aux règles discrétionnaires. Une politique obligatoire suppose que les utilisateurs et objets aient été étiquetés. Classiquement, les objets se voient attribuer une classification, tandis que les utilisateurs possèdent une habilitation. Les règles qui régissent les autorisations d'accès sont basées sur une comparaison de l'habilitation de l'utilisateur et de la classification de l'objet. Ces règles incontournables assurent que le système vérifie des propriétés générales de confidentialité ou d'intégrité par exemple. Ces règles sont souvent utilisées conjointement aux règles d'une politique discrétionnaire car leur pouvoir d'expression est en général relativement faible.

Une politique de sécurité d'autorisation obligatoire (ou MAC de l'anglais " Mandatory Access Control ") impose des règles d'autorisation incontournables qui s'ajoutent aux règles discrétionnaires. Une politique obligatoire suppose que les utilisateurs et objets aient été étiquetés. Classiquement, les objets se voient attribuer une classification, tandis que les utilisateurs possèdent une habilitation. Les règles qui régissent les autorisations d'accès sont basées sur une comparaison de l'habilitation de l'utilisateur et de la classification de l'objet. Ces règles incontournables assurent que le système vérifie des propriétés générales de confidentialité ou d'intégrité par exemple. Ces règles sont souvent utilisées conjointement aux règles d'une politique discrétionnaire car leur pouvoir d'expression est en général relativement faible.

Par exemple, un utilisateur sera autorisé à manipuler une information dans le système si l'utilisateur en question possède le droit de lecture sur l'information (contrôle discrétionnaire) et s'il est habilité à manipuler cette information (contrôle obligatoire).

Par exemple, un utilisateur sera autorisé à manipuler une information dans le système si l'utilisateur en question possède le droit de lecture sur l'information (contrôle discrétionnaire) et s'il est habilité à manipuler cette information (contrôle obligatoire).

Par exemple, un utilisateur sera autorisé à manipuler une information dans le système si l'utilisateur en question possède le droit de lecture sur l'information (contrôle discrétionnaire) et s'il est habilité à manipuler cette information (contrôle obligatoire).

Dans cette partie, nous nous intéressons à un cas particulier de politique MAC, celle du DoD (*Department of Defense* des États-Unis) qui suit le modèle de sécurité dit de Bell-La Padula (introduit dans l'article de Bell et La Padula, *Secure Computer System : Unified Exposition and Multics Interpretation*, 1976). Elle a été mise au point au moment où les efforts se concentraient pour concevoir un système d'exploitation multi-utilisateurs. Les permissions d'accès sont définies à travers une matrice de contrôle d'accès et un ensemble de niveaux de sécurité. Cette politique considère seulement les flux d'informations qui se produisent quand un sujet observe ou modifie un objet. Destinée au domaine militaire, la propriété à assurer est la confidentialité.

Dans cette partie, nous nous intéressons à un cas particulier de politique MAC, celle du DoD (*Department of Defense* des États-Unis) qui suit le modèle de sécurité dit de Bell-La Padula (introduit dans l'article de Bell et La Padula, *Secure Computer System : Unified Exposition and Multics Interpretation*, 1976). Elle a été mise au point au moment où les efforts se concentraient pour concevoir un système d'exploitation multi-utilisateurs. Les permissions d'accès sont définies à travers une matrice de contrôle d'accès et un ensemble de niveaux de sécurité. Cette politique considère seulement les flux d'informations qui se produisent quand un sujet observe ou modifie un objet. Destinée au domaine militaire, la propriété à assurer est la confidentialité.

Dans cette partie, nous nous intéressons à un cas particulier de politique MAC, celle du DoD (*Department of Defense* des États-Unis) qui suit le modèle de sécurité dit de Bell-La Padula (introduit dans l'article de Bell et La Padula, *Secure Computer System : Unified Exposition and Multics Interpretation*, 1976). Elle a été mise au point au moment où les efforts se concentraient pour concevoir un système d'exploitation multi-utilisateurs. Les permissions d'accès sont définies à travers une matrice de contrôle d'accès et un ensemble de niveaux de sécurité. Cette politique considère seulement les flux d'informations qui se produisent quand un sujet observe ou modifie un objet. Destinée au domaine militaire, la propriété à assurer est la confidentialité.

Dans cette partie, nous nous intéressons à un cas particulier de politique MAC, celle du DoD (*Department of Defense* des États-Unis) qui suit le modèle de sécurité dit de Bell-La Padula (introduit dans l'article de Bell et La Padula, *Secure Computer System : Unified Exposition and Multics Interpretation*, 1976). Elle a été mise au point au moment où les efforts se concentraient pour concevoir un système d'exploitation multi-utilisateurs. Les permissions d'accès sont définies à travers une matrice de contrôle d'accès et un ensemble de niveaux de sécurité. Cette politique considère seulement les flux d'informations qui se produisent quand un sujet observe ou modifie un objet. Destinée au domaine militaire, la propriété à assurer est la confidentialité.

Dans cette partie, nous nous intéressons à un cas particulier de politique MAC, celle du DoD (*Department of Defense* des États-Unis) qui suit le modèle de sécurité dit de Bell-La Padula (introduit dans l'article de Bell et La Padula, *Secure Computer System : Unified Exposition and Multics Interpretation*, 1976). Elle a été mise au point au moment où les efforts se concentraient pour concevoir un système d'exploitation multi-utilisateurs. Les permissions d'accès sont définies à travers une matrice de contrôle d'accès et un ensemble de niveaux de sécurité. Cette politique considère seulement les flux d'informations qui se produisent quand un sujet observe ou modifie un objet. Destinée au domaine militaire, la propriété à assurer est la confidentialité.

Dans cette partie, nous nous intéressons à un cas particulier de politique MAC, celle du DoD (*Department of Defense* des États-Unis) qui suit le modèle de sécurité dit de Bell-La Padula (introduit dans l'article de Bell et La Padula, *Secure Computer System : Unified Exposition and Multics Interpretation*, 1976). Elle a été mise au point au moment où les efforts se concentraient pour concevoir un système d'exploitation multi-utilisateurs. Les permissions d'accès sont définies à travers une matrice de contrôle d'accès et un ensemble de niveaux de sécurité. Cette politique considère seulement les flux d'informations qui se produisent quand un sujet observe ou modifie un objet. Destinée au domaine militaire, la propriété à assurer est la confidentialité.

Le modèle associé à la politique de Bell-La Padula est fondé sur la notion mathématique de treillis. Commençons tout d'abord par rappeler ce qu'est un treillis. Soit E un ensemble. Une relation binaire R sur E est une partie de $E \times E$, *i.e.*, $R \subseteq E \times E$. Pour signifier que $(a, b) \in R$, on écrit parfois $a\ R\ b$. Une relation binaire R sur E est une relation d'ordre (partiel) si, et seulement si,

- R est réflexive, ce qui signfie que quel que soit $x \in E$, $(x, x) \in R$ (ou encore $x\ R\ x$) ;
- R est transitive, c'est-à-dire que quels que soient $x, y, z \in R$, si $x\ R\ y$ et si $y\ R\ z$, alors $x\ R\ z$;
- R est antisymétrique, *i.e.*, quels que soient $x, y \in E$, si $x\ R\ y$ et $y\ R\ x$, alors $x = y$.

On dit que la relation d'ordre est totale (on dit aussi que l'on a un ordre total) lorsque quels que soient $x, y \in E$, l'une des deux propriétés est vraie : $x\ R\ y$ ou $y\ R\ x$.

Le modèle associé à la politique de Bell-La Padula est fondé sur la notion mathématique de treillis. Commençons tout d'abord par rappeler ce qu'est un treillis. Soit E un ensemble. Une relation binaire R sur E est une partie de $E \times E$, *i.e.*, $R \subseteq E \times E$. Pour signifier que $(a, b) \in R$, on écrit parfois $a\ R\ b$. Une relation binaire R sur E est une relation d'ordre (partiel) si, et seulement si,

- R est réflexive, ce qui signfie que quel que soit $x \in E$, $(x, x) \in R$ (ou encore $x\ R\ x$) ;
- R est transitive, c'est-à-dire que quels que soient $x, y, z \in R$, si $x\ R\ y$ et si $y\ R\ z$, alors $x\ R\ z$;
- R est antisymétrique, *i.e.*, quels que soient $x, y \in E$, si $x\ R\ y$ et $y\ R\ x$, alors $x = y$.

On dit que la relation d'ordre est totale (on dit aussi que l'on a un ordre total) lorsque quels que soient $x, y \in E$, l'une des deux propriétés est vraie : $x\ R\ y$ ou $y\ R\ x$.

Le modèle associé à la politique de Bell-La Padula est fondé sur la notion mathématique de treillis. Commençons tout d'abord par rappeler ce qu'est un treillis. Soit E un ensemble. Une relation binaire R sur E est une partie de $E \times E$, *i.e.*, $R \subseteq E \times E$. Pour signifier que $(a, b) \in R$, on écrit parfois $a\ R\ b$. Une relation binaire R sur E est une relation d'ordre (partiel) si, et seulement si,

- R est réflexive, ce qui signfie que quel que soit $x \in E$, $(x, x) \in R$ (ou encore $x\ R\ x$) ;
- R est transitive, c'est-à-dire que quels que soient $x, y, z \in R$, si $x\ R\ y$ et si $y\ R\ z$, alors $x\ R\ z$;
- R est antisymétrique, *i.e.*, quels que soient $x, y \in E$, si $x\ R\ y$ et $y\ R\ x$, alors $x = y$.

On dit que la relation d'ordre est totale (on dit aussi que l'on a un ordre total) lorsque quels que soient $x, y \in E$, l'une des deux propriétés est vraie : $x\ R\ y$ ou $y\ R\ x$.

Le modèle associé à la politique de Bell-La Padula est fondé sur la notion mathématique de treillis. Commençons tout d'abord par rappeler ce qu'est un treillis. Soit E un ensemble. Une relation binaire R sur E est une partie de $E \times E$, *i.e.*, $R \subseteq E \times E$. Pour signifier que $(a, b) \in R$, on écrit parfois $a\ R\ b$. Une relation binaire R sur E est une relation d'ordre (partiel) si, et seulement si,

- R est réflexive, ce qui signfie que quel que soit $x \in E$, $(x, x) \in R$ (ou encore $x\ R\ x$) ;
- R est transitive, c'est-à-dire que quels que soient $x, y, z \in R$, si $x\ R\ y$ et si $y\ R\ z$, alors $x\ R\ z$;
- R est antisymétrique, *i.e.*, quels que soient $x, y \in E$, si $x\ R\ y$ et $y\ R\ x$, alors $x = y$.

On dit que la relation d'ordre est totale (on dit aussi que l'on a un ordre total) lorsque quels que soient $x, y \in E$, l'une des deux propriétés est vraie : $x\ R\ y$ ou $y\ R\ x$.

Le modèle associé à la politique de Bell-La Padula est fondé sur la notion mathématique de treillis. Commençons tout d'abord par rappeler ce qu'est un treillis. Soit E un ensemble. Une relation binaire R sur E est une partie de $E \times E$, *i.e.*, $R \subseteq E \times E$. Pour signifier que $(a, b) \in R$, on écrit parfois $a\ R\ b$. Une relation binaire R sur E est une relation d'ordre (partiel) si, et seulement si,

- R est réflexive, ce qui signfie que quel que soit $x \in E$, $(x, x) \in R$ (ou encore $x\ R\ x$) ;
- R est transitive, c'est-à-dire que quels que soient $x, y, z \in R$, si $x\ R\ y$ et si $y\ R\ z$, alors $x\ R\ z$;
- R est antisymétrique, *i.e.*, quels que soient $x, y \in E$, si $x\ R\ y$ et $y\ R\ x$, alors $x = y$.

On dit que la relation d'ordre est totale (on dit aussi que l'on a un ordre total) lorsque quels que soient $x, y \in E$, l'une des deux propriétés est vraie : $x\ R\ y$ ou $y\ R\ x$.

Le modèle associé à la politique de Bell-La Padula est fondé sur la notion mathématique de treillis. Commençons tout d'abord par rappeler ce qu'est un treillis. Soit E un ensemble. Une relation binaire R sur E est une partie de $E \times E$, *i.e.*, $R \subseteq E \times E$. Pour signifier que $(a, b) \in R$, on écrit parfois $a\ R\ b$. Une relation binaire R sur E est une relation d'ordre (partiel) si, et seulement si,

- R est réflexive, ce qui signfie que quel que soit $x \in E$, $(x, x) \in R$ (ou encore $x\ R\ x$) ;
- R est transitive, c'est-à-dire que quels que soient $x, y, z \in R$, si $x\ R\ y$ et si $y\ R\ z$, alors $x\ R\ z$;
- R est antisymétrique, *i.e.*, quels que soient $x, y \in E$, si $x\ R\ y$ et $y\ R\ x$, alors $x = y$.

On dit que la relation d'ordre est totale (on dit aussi que l'on a un ordre total) lorsque quels que soient $x, y \in E$, l'une des deux propriétés est vraie : $x\ R\ y$ ou $y\ R\ x$.

Le modèle associé à la politique de Bell-La Padula est fondé sur la notion mathématique de treillis. Commençons tout d'abord par rappeler ce qu'est un treillis. Soit E un ensemble. Une relation binaire R sur E est une partie de $E \times E$, *i.e.*, $R \subseteq E \times E$. Pour signifier que $(a, b) \in R$, on écrit parfois $a\ R\ b$. Une relation binaire R sur E est une relation d'ordre (partiel) si, et seulement si,

- R est réflexive, ce qui signfie que quel que soit $x \in E$, $(x, x) \in R$ (ou encore $x\ R\ x$) ;
- R est transitive, c'est-à-dire que quels que soient $x, y, z \in R$, si $x\ R\ y$ et si $y\ R\ z$, alors $x\ R\ z$;
- R est antisymétrique, *i.e.*, quels que soient $x, y \in E$, si $x\ R\ y$ et $y\ R\ x$, alors $x = y$.

On dit que la relation d'ordre est totale (on dit aussi que l'on a un ordre total) lorsque quels que soient $x, y \in E$, l'une des deux propriétés est vraie : $x\ R\ y$ ou $y\ R\ x$.

Le modèle associé à la politique de Bell-La Padula est fondé sur la notion mathématique de treillis. Commençons tout d'abord par rappeler ce qu'est un treillis. Soit E un ensemble. Une relation binaire R sur E est une partie de $E \times E$, *i.e.*, $R \subseteq E \times E$. Pour signifier que $(a, b) \in R$, on écrit parfois $a\ R\ b$. Une relation binaire R sur E est une relation d'ordre (partiel) si, et seulement si,

- R est réflexive, ce qui signfie que quel que soit $x \in E$, $(x, x) \in R$ (ou encore $x\ R\ x$) ;
- R est transitive, c'est-à-dire que quels que soient $x, y, z \in R$, si $x\ R\ y$ et si $y\ R\ z$, alors $x\ R\ z$;
- R est antisymétrique, *i.e.*, quels que soient $x, y \in E$, si $x\ R\ y$ et $y\ R\ x$, alors $x = y$.

On dit que la relation d'ordre est totale (on dit aussi que l'on a un ordre total) lorsque quels que soient $x, y \in E$, l'une des deux propriétés est vraie : $x\ R\ y$ ou $y\ R\ x$.

Le modèle associé à la politique de Bell-La Padula est fondé sur la notion mathématique de treillis. Commençons tout d'abord par rappeler ce qu'est un treillis. Soit E un ensemble. Une relation binaire R sur E est une partie de $E \times E$, *i.e.*, $R \subseteq E \times E$. Pour signifier que $(a, b) \in R$, on écrit parfois $a\ R\ b$. Une relation binaire R sur E est une relation d'ordre (partiel) si, et seulement si,

- R est réflexive, ce qui signfie que quel que soit $x \in E$, $(x, x) \in R$ (ou encore $x\ R\ x$) ;
- R est transitive, c'est-à-dire que quels que soient $x, y, z \in R$, si $x\ R\ y$ et si $y\ R\ z$, alors $x\ R\ z$;
- R est antisymétrique, *i.e.*, quels que soient $x, y \in E$, si $x\ R\ y$ et $y\ R\ x$, alors $x = y$.

On dit que la relation d'ordre est totale (on dit aussi que l'on a un ordre total) lorsque quels que soient $x, y \in E$, l'une des deux propriétés est vraie : $x\ R\ y$ ou $y\ R\ x$.

Le modèle associé à la politique de Bell-La Padula est fondé sur la notion mathématique de treillis. Commençons tout d'abord par rappeler ce qu'est un treillis. Soit E un ensemble. Une relation binaire R sur E est une partie de $E \times E$, *i.e.*, $R \subseteq E \times E$. Pour signifier que $(a, b) \in R$, on écrit parfois $a\ R\ b$. Une relation binaire R sur E est une relation d'ordre (partiel) si, et seulement si,

- R est réflexive, ce qui signfie que quel que soit $x \in E$, $(x, x) \in R$ (ou encore $x\ R\ x$) ;
- R est transitive, c'est-à-dire que quels que soient $x, y, z \in R$, si $x\ R\ y$ et si $y\ R\ z$, alors $x\ R\ z$;
- R est antisymétrique, *i.e.*, quels que soient $x, y \in E$, si $x\ R\ y$ et $y\ R\ x$, alors $x = y$.

On dit que la relation d'ordre est totale (on dit aussi que l'on a un ordre total) lorsque quels que soient $x, y \in E$, l'une des deux propriétés est vraie : $x\ R\ y$ ou $y\ R\ x$.

Le modèle associé à la politique de Bell-La Padula est fondé sur la notion mathématique de treillis. Commençons tout d'abord par rappeler ce qu'est un treillis. Soit E un ensemble. Une relation binaire R sur E est une partie de $E \times E$, *i.e.*, $R \subseteq E \times E$. Pour signifier que $(a, b) \in R$, on écrit parfois $a\ R\ b$. Une relation binaire R sur E est une relation d'ordre (partiel) si, et seulement si,

- R est réflexive, ce qui signfie que quel que soit $x \in E$, $(x, x) \in R$ (ou encore $x\ R\ x$) ;
- R est transitive, c'est-à-dire que quels que soient $x, y, z \in R$, si $x\ R\ y$ et si $y\ R\ z$, alors $x\ R\ z$;
- R est antisymétrique, *i.e.*, quels que soient $x, y \in E$, si $x\ R\ y$ et $y\ R\ x$, alors $x = y$.

On dit que la relation d'ordre est totale (on dit aussi que l'on a un ordre total) lorsque quels que soient $x, y \in E$, l'une des deux propriétés est vraie : $x\ R\ y$ ou $y\ R\ x$.

Le modèle associé à la politique de Bell-La Padula est fondé sur la notion mathématique de treillis. Commençons tout d'abord par rappeler ce qu'est un treillis. Soit E un ensemble. Une relation binaire R sur E est une partie de $E \times E$, *i.e.*, $R \subseteq E \times E$. Pour signifier que $(a, b) \in R$, on écrit parfois $a\ R\ b$. Une relation binaire R sur E est une relation d'ordre (partiel) si, et seulement si,

- R est réflexive, ce qui signfie que quel que soit $x \in E$, $(x, x) \in R$ (ou encore $x\ R\ x$) ;
- R est transitive, c'est-à-dire que quels que soient $x, y, z \in R$, si $x\ R\ y$ et si $y\ R\ z$, alors $x\ R\ z$;
- R est antisymétrique, *i.e.*, quels que soient $x, y \in E$, si $x\ R\ y$ et $y\ R\ x$, alors $x = y$.

On dit que la relation d'ordre est totale (on dit aussi que l'on a un ordre total) lorsque quels que soient $x, y \in E$, l'une des deux propriétés est vraie : $x\ R\ y$ ou $y\ R\ x$.

Le modèle associé à la politique de Bell-La Padula est fondé sur la notion mathématique de treillis. Commençons tout d'abord par rappeler ce qu'est un treillis. Soit E un ensemble. Une relation binaire R sur E est une partie de $E \times E$, *i.e.*, $R \subseteq E \times E$. Pour signifier que $(a, b) \in R$, on écrit parfois $a\ R\ b$. Une relation binaire R sur E est une relation d'ordre (partiel) si, et seulement si,

- R est réflexive, ce qui signfie que quel que soit $x \in E$, $(x, x) \in R$ (ou encore $x\ R\ x$) ;
- R est transitive, c'est-à-dire que quels que soient $x, y, z \in R$, si $x\ R\ y$ et si $y\ R\ z$, alors $x\ R\ z$;
- R est antisymétrique, *i.e.*, quels que soient $x, y \in E$, si $x\ R\ y$ et $y\ R\ x$, alors $x = y$.

On dit que la relation d'ordre est totale (on dit aussi que l'on a un ordre total) lorsque quels que soient $x, y \in E$, l'une des deux propriétés est vraie : $x\ R\ y$ ou $y\ R\ x$.

Le modèle associé à la politique de Bell-La Padula est fondé sur la notion mathématique de treillis. Commençons tout d'abord par rappeler ce qu'est un treillis. Soit E un ensemble. Une relation binaire R sur E est une partie de $E \times E$, *i.e.*, $R \subseteq E \times E$. Pour signifier que $(a, b) \in R$, on écrit parfois $a\ R\ b$. Une relation binaire R sur E est une relation d'ordre (partiel) si, et seulement si,

- R est réflexive, ce qui signfie que quel que soit $x \in E$, $(x, x) \in R$ (ou encore $x\ R\ x$) ;
- R est transitive, c'est-à-dire que quels que soient $x, y, z \in R$, si $x\ R\ y$ et si $y\ R\ z$, alors $x\ R\ z$;
- R est antisymétrique, *i.e.*, quels que soient $x, y \in E$, si $x\ R\ y$ et $y\ R\ x$, alors $x = y$.

On dit que la relation d'ordre est totale (on dit aussi que l'on a un ordre total) lorsque quels que soient $x, y \in E$, l'une des deux propriétés est vraie : $x\ R\ y$ ou $y\ R\ x$.

Le modèle associé à la politique de Bell-La Padula est fondé sur la notion mathématique de treillis. Commençons tout d'abord par rappeler ce qu'est un treillis. Soit E un ensemble. Une relation binaire R sur E est une partie de $E \times E$, *i.e.*, $R \subseteq E \times E$. Pour signifier que $(a, b) \in R$, on écrit parfois $a\ R\ b$. Une relation binaire R sur E est une relation d'ordre (partiel) si, et seulement si,

- R est réflexive, ce qui signfie que quel que soit $x \in E$, $(x, x) \in R$ (ou encore $x\ R\ x$) ;
- R est transitive, c'est-à-dire que quels que soient $x, y, z \in R$, si $x\ R\ y$ et si $y\ R\ z$, alors $x\ R\ z$;
- R est antisymétrique, *i.e.*, quels que soient $x, y \in E$, si $x\ R\ y$ et $y\ R\ x$, alors $x = y$.

On dit que la relation d'ordre est totale (on dit aussi que l'on a un ordre total) lorsque quels que soient $x, y \in E$, l'une des deux propriétés est vraie : $x\ R\ y$ ou $y\ R\ x$.

Par exemple, la relation d'inclusion (au sens large) $\subseteq$ des parties d'un ensemble donné est une relation d'ordre partielle (on peut trouver deux ensembles A et B pour lesquels aucune des deux relations n'est vérifiée : $A \subseteq B$ ou $B \subseteq A$). La relation d'ordre usuel sur les nombres (entiers ou réels) est une relation d'ordre totale. On appelle ensemble (partiellement) ordonné un ensemble E avec une relation d'ordre sur E. Si $(E, \leq)$ est un ensemble ordonné et $\leq$ est total, alors on dit que $(E, \leq)$ est ensemble totalement ordonné.

Par exemple, la relation d'inclusion (au sens large) $\subseteq$ des parties d'un ensemble donné est une relation d'ordre partielle (on peut trouver deux ensembles A et B pour lesquels aucune des deux relations n'est vérifiée : $A \subseteq B$ ou $B \subseteq A$). La relation d'ordre usuel sur les nombres (entiers ou réels) est une relation d'ordre totale. On appelle ensemble (partiellement) ordonné un ensemble E avec une relation d'ordre sur E. Si $(E, \leq)$ est un ensemble ordonné et $\leq$ est total, alors on dit que $(E, \leq)$ est ensemble totalement ordonné.

Par exemple, la relation d'inclusion (au sens large) $\subseteq$ des parties d'un ensemble donné est une relation d'ordre partielle (on peut trouver deux ensembles A et B pour lesquels aucune des deux relations n'est vérifiée : $A \subseteq B$ ou $B \subseteq A$). La relation d'ordre usuel sur les nombres (entiers ou réels) est une relation d'ordre totale. On appelle ensemble (partiellement) ordonné un ensemble E avec une relation d'ordre sur E. Si $(E, \leq)$ est un ensemble ordonné et $\leq$ est total, alors on dit que $(E, \leq)$ est ensemble totalement ordonné.

Par exemple, la relation d'inclusion (au sens large) $\subseteq$ des parties d'un ensemble donné est une relation d'ordre partielle (on peut trouver deux ensembles A et B pour lesquels aucune des deux relations n'est vérifiée : $A \subseteq B$ ou $B \subseteq A$). La relation d'ordre usuel sur les nombres (entiers ou réels) est une relation d'ordre totale. On appelle ensemble (partiellement) ordonné un ensemble E avec une relation d'ordre sur E. Si $(E, \leq)$ est un ensemble ordonné et $\leq$ est total, alors on dit que $(E, \leq)$ est ensemble totalement ordonné.

Par exemple, la relation d'inclusion (au sens large) $\subseteq$ des parties d'un ensemble donné est une relation d'ordre partielle (on peut trouver deux ensembles A et B pour lesquels aucune des deux relations n'est vérifiée : $A \subseteq B$ ou $B \subseteq A$). La relation d'ordre usuel sur les nombres (entiers ou réels) est une relation d'ordre totale. On appelle ensemble (partiellement) ordonné un ensemble E avec une relation d'ordre sur E. Si $(E, \leq)$ est un ensemble ordonné et $\leq$ est total, alors on dit que $(E, \leq)$ est ensemble totalement ordonné.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. Soient $x, y \in E$. On appelle infimum de x, y (ou borne inférieure) un élément $m \in E$ tel que

- $m \leq x$ et $m \leq y$;
- quel que soit $m' \in E$ tel que $m' \leq x$ et $m' \leq y$, on a $m' \leq m$.

S'il existe, alors l'infimum est unique et il est noté $\inf\{x, y\}$. Par exemple, $\inf\{12, 56\} = 12$. Un autre exemple : soit $A = \{0, 1, 2\}$ et $B = \{0, 2, 3\}$, alors $\inf\{A, B\} = \{0, 2\}$.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. Soient $x, y \in E$. On appelle infimum de x, y (ou borne inférieure) un élément $m \in E$ tel que

- $m \leq x$ et $m \leq y$;
- quel que soit $m' \in E$ tel que $m' \leq x$ et $m' \leq y$, on a $m' \leq m$.

S'il existe, alors l'infimum est unique et il est noté $\inf\{x, y\}$. Par exemple, $\inf\{12, 56\} = 12$. Un autre exemple : soit $A = \{0, 1, 2\}$ et $B = \{0, 2, 3\}$, alors $\inf\{A, B\} = \{0, 2\}$.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. Soient $x, y \in E$. On appelle infimum de x, y (ou borne inférieure) un élément $m \in E$ tel que

- $m \leq x$ et $m \leq y$;
- quel que soit $m' \in E$ tel que $m' \leq x$ et $m' \leq y$, on a $m' \leq m$.

S'il existe, alors l'infimum est unique et il est noté $\inf\{x, y\}$. Par exemple, $\inf\{12, 56\} = 12$. Un autre exemple : soit $A = \{0, 1, 2\}$ et $B = \{0, 2, 3\}$, alors $\inf\{A, B\} = \{0, 2\}$.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. Soient $x, y \in E$. On appelle infimum de x, y (ou borne inférieure) un élément $m \in E$ tel que

- $m \leq x$ et $m \leq y$;
- quel que soit $m' \in E$ tel que $m' \leq x$ et $m' \leq y$, on a $m' \leq m$.

S'il existe, alors l'infimum est unique et il est noté $\inf\{x, y\}$. Par exemple, $\inf\{12, 56\} = 12$. Un autre exemple : soit $A = \{0, 1, 2\}$ et $B = \{0, 2, 3\}$, alors $\inf\{A, B\} = \{0, 2\}$.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. Soient $x, y \in E$. On appelle infimum de x, y (ou borne inférieure) un élément $m \in E$ tel que

- $m \leq x$ et $m \leq y$;
- quel que soit $m' \in E$ tel que $m' \leq x$ et $m' \leq y$, on a $m' \leq m$.

S'il existe, alors l'infimum est unique et il est noté $\inf\{x, y\}$. Par exemple, $\inf\{12, 56\} = 12$. Un autre exemple : soit $A = \{0, 1, 2\}$ et $B = \{0, 2, 3\}$, alors $\inf\{A, B\} = \{0, 2\}$.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. Soient $x, y \in E$. On appelle infimum de x, y (ou borne inférieure) un élément $m \in E$ tel que

- $m \leq x$ et $m \leq y$;
- quel que soit $m' \in E$ tel que $m' \leq x$ et $m' \leq y$, on a $m' \leq m$.

S'il existe, alors l'infimum est unique et il est noté $\inf\{x, y\}$. Par exemple, $\inf\{12, 56\} = 12$. Un autre exemple : soit $A = \{0, 1, 2\}$ et $B = \{0, 2, 3\}$, alors $\inf\{A, B\} = \{0, 2\}$.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. Soient $x, y \in E$. On appelle infimum de x, y (ou borne inférieure) un élément $m \in E$ tel que

- $m \leq x$ et $m \leq y$;
- quel que soit $m' \in E$ tel que $m' \leq x$ et $m' \leq y$, on a $m' \leq m$.

S'il existe, alors l'infimum est unique et il est noté $\inf\{x, y\}$. Par exemple, $\inf\{12, 56\} = 12$. Un autre exemple : soit $A = \{0, 1, 2\}$ et $B = \{0, 2, 3\}$, alors $\inf\{A, B\} = \{0, 2\}$.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. Soient $x, y \in E$. On appelle infimum de x, y (ou borne inférieure) un élément $m \in E$ tel que

- $m \leq x$ et $m \leq y$;
- quel que soit $m' \in E$ tel que $m' \leq x$ et $m' \leq y$, on a $m' \leq m$.

S'il existe, alors l'infimum est unique et il est noté $\inf\{x, y\}$. Par exemple, $\inf\{12, 56\} = 12$. Un autre exemple : soit $A = \{0, 1, 2\}$ et $B = \{0, 2, 3\}$, alors $\inf\{A, B\} = \{0, 2\}$.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. Soient $x, y \in E$. On appelle infimum de x, y (ou borne inférieure) un élément $m \in E$ tel que

- $m \leq x$ et $m \leq y$;
- quel que soit $m' \in E$ tel que $m' \leq x$ et $m' \leq y$, on a $m' \leq m$.

S'il existe, alors l'infimum est unique et il est noté $\inf\{x, y\}$. Par exemple, $\inf\{12, 56\} = 12$. Un autre exemple : soit $A = \{0, 1, 2\}$ et $B = \{0, 2, 3\}$, alors $\inf\{A, B\} = \{0, 2\}$.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. Soient $x, y \in E$. On appelle supremum de x, y (ou borne supérieure) un élément $M \in E$ tel que

- $x \leq M$ et $y \leq M$;
- quel que soit $M' \in E$ tel que $x \leq M'$ et $y \leq M'$, on a $M \leq M'$.

S'il existe, alors le supremum est unique et il est noté $\sup\{x, y\}$. Par exemple, $\sup\{12, 56\} = 56$. Un autre exemple : soit $A = \{0, 1, 2\}$ et $B = \{0, 2, 3\}$, alors $\sup\{A, B\} = \{0, 1, 2, 3\}$.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. Soient $x, y \in E$. On appelle supremum de x, y (ou borne supérieure) un élément $M \in E$ tel que

- $x \leq M$ et $y \leq M$;
- quel que soit $M' \in E$ tel que $x \leq M'$ et $y \leq M'$, on a $M \leq M'$.

S'il existe, alors le supremum est unique et il est noté $\sup\{x, y\}$. Par exemple, $\sup\{12, 56\} = 56$. Un autre exemple : soit $A = \{0, 1, 2\}$ et $B = \{0, 2, 3\}$, alors $\sup\{A, B\} = \{0, 1, 2, 3\}$.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. Soient $x, y \in E$. On appelle supremum de x, y (ou borne supérieure) un élément $M \in E$ tel que

- $x \leq M$ et $y \leq M$;
- quel que soit $M' \in E$ tel que $x \leq M'$ et $y \leq M'$, on a $M \leq M'$.

S'il existe, alors le supremum est unique et il est noté $\sup\{x, y\}$. Par exemple, $\sup\{12, 56\} = 56$. Un autre exemple : soit $A = \{0, 1, 2\}$ et $B = \{0, 2, 3\}$, alors $\sup\{A, B\} = \{0, 1, 2, 3\}$.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. Soient $x, y \in E$. On appelle supremum de x, y (ou borne supérieure) un élément $M \in E$ tel que

- $x \leq M$ et $y \leq M$;
- quel que soit $M' \in E$ tel que $x \leq M'$ et $y \leq M'$, on a $M \leq M'$.

S'il existe, alors le supremum est unique et il est noté $\sup\{x, y\}$. Par exemple, $\sup\{12, 56\} = 56$. Un autre exemple : soit $A = \{0, 1, 2\}$ et $B = \{0, 2, 3\}$, alors $\sup\{A, B\} = \{0, 1, 2, 3\}$.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. Soient $x, y \in E$. On appelle supremum de x, y (ou borne supérieure) un élément $M \in E$ tel que

- $x \leq M$ et $y \leq M$;
- quel que soit $M' \in E$ tel que $x \leq M'$ et $y \leq M'$, on a $M \leq M'$.

S'il existe, alors le supremum est unique et il est noté $\sup\{x, y\}$. Par exemple, $\sup\{12, 56\} = 56$. Un autre exemple : soit $A = \{0, 1, 2\}$ et $B = \{0, 2, 3\}$, alors $\sup\{A, B\} = \{0, 1, 2, 3\}$.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. Soient $x, y \in E$. On appelle supremum de x, y (ou borne supérieure) un élément $M \in E$ tel que

- $x \leq M$ et $y \leq M$;
- quel que soit $M' \in E$ tel que $x \leq M'$ et $y \leq M'$, on a $M \leq M'$.

S'il existe, alors le supremum est unique et il est noté $\sup\{x, y\}$. Par exemple, $\sup\{12, 56\} = 56$. Un autre exemple : soit $A = \{0, 1, 2\}$ et $B = \{0, 2, 3\}$, alors $\sup\{A, B\} = \{0, 1, 2, 3\}$.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. Soient $x, y \in E$. On appelle supremum de x, y (ou borne supérieure) un élément $M \in E$ tel que

- $x \leq M$ et $y \leq M$;
- quel que soit $M' \in E$ tel que $x \leq M'$ et $y \leq M'$, on a $M \leq M'$.

S'il existe, alors le supremum est unique et il est noté $\sup\{x, y\}$. Par exemple, $\sup\{12, 56\} = 56$. Un autre exemple : soit $A = \{0, 1, 2\}$ et $B = \{0, 2, 3\}$, alors $\sup\{A, B\} = \{0, 1, 2, 3\}$.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. Soient $x, y \in E$. On appelle supremum de x, y (ou borne supérieure) un élément $M \in E$ tel que

- $x \leq M$ et $y \leq M$;
- quel que soit $M' \in E$ tel que $x \leq M'$ et $y \leq M'$, on a $M \leq M'$.

S'il existe, alors le supremum est unique et il est noté $\sup\{x, y\}$. Par exemple, $\sup\{12, 56\} = 56$. Un autre exemple : soit $A = \{0, 1, 2\}$ et $B = \{0, 2, 3\}$, alors $\sup\{A, B\} = \{0, 1, 2, 3\}$.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. Soient $x, y \in E$. On appelle supremum de x, y (ou borne supérieure) un élément $M \in E$ tel que

- $x \leq M$ et $y \leq M$;
- quel que soit $M' \in E$ tel que $x \leq M'$ et $y \leq M'$, on a $M \leq M'$.

S'il existe, alors le supremum est unique et il est noté $\sup\{x, y\}$. Par exemple, $\sup\{12, 56\} = 56$. Un autre exemple : soit $A = \{0, 1, 2\}$ et $B = \{0, 2, 3\}$, alors $\sup\{A, B\} = \{0, 1, 2, 3\}$.

Attention : l'infimum et/ou le supremum peuvent ne pas exister. Par exemple, soit $E = \{a, b, c\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$ et $c \leq c$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). Et $\sup\{b, c\}$ n'existe pas.
Un autre exemple, soit $E = \{a, b, c, d\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$, $c \leq c$ et $d \leq d$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). On a : $\sup\{b, c\}$ et $\inf\{a, d\}$ n'existent pas.

Néanmoins, si l'ordre $\leq$ est total, alors l'infimum et le supremum de deux éléments quelconques existent toujours. En effet, dans ce cas, on a soit $x \leq y$, soit $y \leq x$ (puisque l'ordre est total), dans le premier cas, on a $\inf\{x, y\} = x$ et $\sup\{x, y\} = y$, alors que dans le second cas on a $\inf\{x, y\} = y$ et $\sup\{x, y\} = x$.

Attention : l'infimum et/ou le supremum peuvent ne pas exister. Par exemple, soit $E = \{a, b, c\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$ et $c \leq c$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). Et $\sup\{b, c\}$ n'existe pas.

Un autre exemple, soit $E = \{a, b, c, d\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$, $c \leq c$ et $d \leq d$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). On a : $\sup\{b, c\}$ et $\inf\{a, d\}$ n'existent pas.

Néanmoins, si l'ordre $\leq$ est total, alors l'infimum et le supremum de deux éléments quelconques existent toujours. En effet, dans ce cas, on a soit $x \leq y$, soit $y \leq x$ (puisque l'ordre est total), dans le premier cas, on a $\inf\{x, y\} = x$ et $\sup\{x, y\} = y$, alors que dans le second cas on a $\inf\{x, y\} = y$ et $\sup\{x, y\} = x$.

Attention : l'infimum et/ou le supremum peuvent ne pas exister. Par exemple, soit $E = \{a, b, c\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$ et $c \leq c$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). Et $\sup\{b, c\}$ n'existe pas.

Un autre exemple, soit $E = \{a, b, c, d\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$, $c \leq c$ et $d \leq d$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). On a : $\sup\{b, c\}$ et $\inf\{a, d\}$ n'existent pas.

Néanmoins, si l'ordre $\leq$ est total, alors l'infimum et le supremum de deux éléments quelconques existent toujours. En effet, dans ce cas, on a soit $x \leq y$, soit $y \leq x$ (puisque l'ordre est total), dans le premier cas, on a $\inf\{x, y\} = x$ et $\sup\{x, y\} = y$, alors que dans le second cas on a $\inf\{x, y\} = y$ et $\sup\{x, y\} = x$.

Attention : l'infimum et/ou le supremum peuvent ne pas exister. Par exemple, soit $E = \{a, b, c\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$ et $c \leq c$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). Et $\sup\{b, c\}$ n'existe pas.

Un autre exemple, soit $E = \{a, b, c, d\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$, $c \leq c$ et $d \leq d$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). On a : $\sup\{b, c\}$ et $\inf\{a, d\}$ n'existent pas.

Néanmoins, si l'ordre $\leq$ est total, alors l'infimum et le supremum de deux éléments quelconques existent toujours. En effet, dans ce cas, on a soit $x \leq y$, soit $y \leq x$ (puisque l'ordre est total), dans le premier cas, on a $\inf\{x, y\} = x$ et $\sup\{x, y\} = y$, alors que dans le second cas on a $\inf\{x, y\} = y$ et $\sup\{x, y\} = x$.

Attention : l'infimum et/ou le supremum peuvent ne pas exister. Par exemple, soit $E = \{a, b, c\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$ et $c \leq c$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). Et $\sup\{b, c\}$ n'existe pas.
Un autre exemple, soit $E = \{a, b, c, d\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$, $c \leq c$ et $d \leq d$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). On a : $\sup\{b, c\}$ et $\inf\{a, d\}$ n'existent pas.

Néanmoins, si l'ordre $\leq$ est total, alors l'infimum et le supremum de deux éléments quelconques existent toujours. En effet, dans ce cas, on a soit $x \leq y$, soit $y \leq x$ (puisque l'ordre est total), dans le premier cas, on a $\inf\{x, y\} = x$ et $\sup\{x, y\} = y$, alors que dans le second cas on a $\inf\{x, y\} = y$ et $\sup\{x, y\} = x$.

Attention : l'infimum et/ou le supremum peuvent ne pas exister. Par exemple, soit $E = \{a, b, c\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$ et $c \leq c$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). Et $\sup\{b, c\}$ n'existe pas.
Un autre exemple, soit $E = \{a, b, c, d\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$, $c \leq c$ et $d \leq d$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). On a : $\sup\{b, c\}$ et $\inf\{a, d\}$ n'existent pas.

Néanmoins, si l'ordre $\leq$ est total, alors l'infimum et le supremum de deux éléments quelconques existent toujours. En effet, dans ce cas, on a soit $x \leq y$, soit $y \leq x$ (puisque l'ordre est total), dans le premier cas, on a $\inf\{x, y\} = x$ et $\sup\{x, y\} = y$, alors que dans le second cas on a $\inf\{x, y\} = y$ et $\sup\{x, y\} = x$.

Attention : l'infimum et/ou le supremum peuvent ne pas exister. Par exemple, soit $E = \{a, b, c\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$ et $c \leq c$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). Et $\sup\{b, c\}$ n'existe pas.
Un autre exemple, soit $E = \{a, b, c, d\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$, $c \leq c$ et $d \leq d$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). On a : $\sup\{b, c\}$ et $\inf\{a, d\}$ n'existent pas.

Néanmoins, si l'ordre $\leq$ est total, alors l'infimum et le supremum de deux éléments quelconques existent toujours. En effet, dans ce cas, on a soit $x \leq y$, soit $y \leq x$ (puisque l'ordre est total), dans le premier cas, on a $\inf\{x, y\} = x$ et $\sup\{x, y\} = y$, alors que dans le second cas on a $\inf\{x, y\} = y$ et $\sup\{x, y\} = x$.

Attention : l'infimum et/ou le supremum peuvent ne pas exister. Par exemple, soit $E = \{a, b, c\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$ et $c \leq c$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). Et $\sup\{b, c\}$ n'existe pas.
Un autre exemple, soit $E = \{a, b, c, d\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$, $c \leq c$ et $d \leq d$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). On a : $\sup\{b, c\}$ et $\inf\{a, d\}$ n'existent pas.

Néanmoins, si l'ordre $\leq$ est total, alors l'infimum et le supremum de deux éléments quelconques existent toujours. En effet, dans ce cas, on a soit $x \leq y$, soit $y \leq x$ (puisque l'ordre est total), dans le premier cas, on a $\inf\{x, y\} = x$ et $\sup\{x, y\} = y$, alors que dans le second cas on a $\inf\{x, y\} = y$ et $\sup\{x, y\} = x$.

Attention : l'infimum et/ou le supremum peuvent ne pas exister. Par exemple, soit $E = \{a, b, c\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$ et $c \leq c$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). Et $\sup\{b, c\}$ n'existe pas.
Un autre exemple, soit $E = \{a, b, c, d\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$, $c \leq c$ et $d \leq d$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). On a : $\sup\{b, c\}$ et $\inf\{a, d\}$ n'existent pas.

Néanmoins, si l'ordre $\leq$ est total, alors l'infimum et le supremum de deux éléments quelconques existent toujours. En effet, dans ce cas, on a soit $x \leq y$, soit $y \leq x$ (puisque l'ordre est total), dans le premier cas, on a $\inf\{x, y\} = x$ et $\sup\{x, y\} = y$, alors que dans le second cas on a $\inf\{x, y\} = y$ et $\sup\{x, y\} = x$.

Attention : l'infimum et/ou le supremum peuvent ne pas exister. Par exemple, soit $E = \{a, b, c\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$ et $c \leq c$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). Et $\sup\{b, c\}$ n'existe pas.
Un autre exemple, soit $E = \{a, b, c, d\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$, $c \leq c$ et $d \leq d$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). On a : $\sup\{b, c\}$ et $\inf\{a, d\}$ n'existent pas.

Néanmoins, si l'ordre $\leq$ est total, alors l'infimum et le supremum de deux éléments quelconques existent toujours. En effet, dans ce cas, on a soit $x \leq y$, soit $y \leq x$ (puisque l'ordre est total), dans le premier cas, on a $\inf\{x, y\} = x$ et $\sup\{x, y\} = y$, alors que dans le second cas on a $\inf\{x, y\} = y$ et $\sup\{x, y\} = x$.

Attention : l'infimum et/ou le supremum peuvent ne pas exister. Par exemple, soit $E = \{a, b, c\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$ et $c \leq c$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). Et $\sup\{b, c\}$ n'existe pas.
Un autre exemple, soit $E = \{a, b, c, d\}$ et $\leq$ la relation binaire sur E définie par : $a \leq a$, $a \leq b$, $a \leq c$, $b \leq b$, $c \leq c$ et $d \leq d$. On peut vérifier que $\leq$ est une relation d'ordre (partiel car on n'a ni $b \leq c$ ni $c \leq b$). On a : $\sup\{b, c\}$ et $\inf\{a, d\}$ n'existent pas.

Néanmoins, si l'ordre $\leq$ est total, alors l'infimum et le supremum de deux éléments quelconques existent toujours. En effet, dans ce cas, on a soit $x \leq y$, soit $y \leq x$ (puisque l'ordre est total), dans le premier cas, on a $\inf\{x, y\} = x$ et $\sup\{x, y\} = y$, alors que dans le second cas on a $\inf\{x, y\} = y$ et $\sup\{x, y\} = x$.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. On dit que $(E, \leq)$ est un treillis si, et seulement si, quels que soient $x, y \in E$, $\inf\{x, y\}$ et $\sup\{x, y\}$ existent.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. On dit que $(E, \leq)$ est un treillis si, et seulement si, quels que soient $x, y \in E$, $\inf\{x, y\}$ et $\sup\{x, y\}$ existent.

Soit $(E, \leq)$ un ensemble (partiellement) ordonné. On dit que $(E, \leq)$ est un treillis si, et seulement si, quels que soient $x, y \in E$, $\inf\{x, y\}$ et $\sup\{x, y\}$ existent.

Soit E un ensemble et R une relation binaire sur E. Pour montrer que (E, R) est une treillis, il faut démontrer que

1. (E, R) est un ensemble ordonné, soit encore que
   1. R est réflexive ;
   2. R est transitive ;
   3. R est antisymétrique.
2. Puis démontrer que quels que soient $x, y \in E$, $\inf\{x, y\}$ et $\sup\{x, y\}$ existent.

Soit E un ensemble et R une relation binaire sur E. Pour montrer que (E,R) est une treillis, il faut démontrer que

1. (E,R) est un ensemble ordonné, soit encore que
   1. R est réflexive ;
   2. R est transitive ;
   3. R est antisymétrique.
2. Puis démontrer que quels que soient $x, y \in E$, $\inf\{x,y\}$ et $\sup\{x,y\}$ existent.

Soit E un ensemble et R une relation binaire sur E. Pour montrer que (E,R) est une treillis, il faut démontrer que

1. (E,R) est un ensemble ordonné, soit encore que
   1. R est réflexive ;
   2. R est transitive ;
   3. R est antisymétrique.
2. Puis démontrer que quels que soient $x, y \in E$, $\inf\{x,y\}$ et $\sup\{x,y\}$ existent.

Soit E un ensemble et R une relation binaire sur E. Pour montrer que (E,R) est une treillis, il faut démontrer que

1. (E,R) est un ensemble ordonné, soit encore que
   1. R est réflexive ;
   2. R est transitive ;
   3. R est antisymétrique.
2. Puis démontrer que quels que soient $x, y \in E$, $\inf\{x,y\}$ et $\sup\{x,y\}$ existent.

Soit E un ensemble et R une relation binaire sur E. Pour montrer que (E,R) est une treillis, il faut démontrer que

1. (E,R) est un ensemble ordonné, soit encore que
   1. R est réflexive ;
   2. R est transitive ;
   3. R est antisymétrique.
2. Puis démontrer que quels que soient $x, y \in E$, $\inf\{x,y\}$ et $\sup\{x,y\}$ existent.

Soit E un ensemble et R une relation binaire sur E. Pour montrer que (E, R) est une treillis, il faut démontrer que

1. (E, R) est un ensemble ordonné, soit encore que
   1. R est réflexive ;
   2. R est transitive ;
   3. R est antisymétrique.
2. Puis démontrer que quels que soient $x, y \in E$, $\inf\{x, y\}$ et $\sup\{x, y\}$ existent.

Soit E un ensemble et R une relation binaire sur E. Pour montrer que (E,R) est une treillis, il faut démontrer que

1. (E,R) est un ensemble ordonné, soit encore que
   1. R est réflexive ;
   2. R est transitive ;
   3. R est antisymétrique.
2. Puis démontrer que quels que soient $x, y \in E$, $\inf\{x,y\}$ et $\sup\{x,y\}$ existent.

Soit E un ensemble et R une relation binaire sur E. Pour montrer que (E, R) est une treillis, il faut démontrer que

1. (E, R) est un ensemble ordonné, soit encore que
   1. R est réflexive ;
   2. R est transitive ;
   3. R est antisymétrique.
2. Puis démontrer que quels que soient $x, y \in E$, $\inf\{x, y\}$ et $\sup\{x, y\}$ existent.

Soit E un ensemble et R une relation binaire sur E. Pour montrer que (E,R) est une treillis, il faut démontrer que

1. (E,R) est un ensemble ordonné, soit encore que
   1. R est réflexive ;
   2. R est transitive ;
   3. R est antisymétrique.
2. Puis démontrer que quels que soient $x, y \in E$, $\inf\{x,y\}$ et $\sup\{x,y\}$ existent.

Le modèle associé à la politique de Bell-LaPadula est donc fondé sur la notion de treillis. Il s'appuie sur l'association de différents niveaux aux sujets (niveaux d'habilitation) et aux objets (niveaux de classification). Chaque niveau $n = (cl, C)$ est un couple caractérisé par ses deux attributs cl et C :

- C est un compartiment défini par un ensemble de catégories, par exemple $\{$nucléaire, défense$\}$;
- cl est une classification prise dans un ensemble **totalement ordonné**, par exemple : $\{$non classifié, confidentiel, secret, top secret$\}$.

Le modèle associé à la politique de Bell-LaPadula est donc fondé sur la notion de treillis. Il s'appuie sur l'association de différents niveaux aux sujets (niveaux d'habilitation) et aux objets (niveaux de classification). Chaque niveau $n = (cl, C)$ est un couple caractérisé par ses deux attributs cl et C :

- C est un compartiment défini par un ensemble de catégories, par exemple $\{$nucléaire, défense$\}$;
- cl est une classification prise dans un ensemble **totalement ordonné**, par exemple : $\{$non classifié, confidentiel, secret, top secret$\}$.

Le modèle associé à la politique de Bell-LaPadula est donc fondé sur la notion de treillis. Il s'appuie sur l'association de différents niveaux aux sujets (niveaux d'habilitation) et aux objets (niveaux de classification). Chaque niveau $n = (cl, C)$ est un couple caractérisé par ses deux attributs cl et C :

- C est un compartiment défini par un ensemble de catégories, par exemple $\{$nucléaire, défense$\}$;
- cl est une classification prise dans un ensemble **totalement ordonné**, par exemple : $\{$non classifié, confidentiel, secret, top secret$\}$.

Le modèle associé à la politique de Bell-LaPadula est donc fondé sur la notion de treillis. Il s'appuie sur l'association de différents niveaux aux sujets (niveaux d'habilitation) et aux objets (niveaux de classification). Chaque niveau $n = (cl, C)$ est un couple caractérisé par ses deux attributs cl et C :

- C est un compartiment défini par un ensemble de catégories, par exemple $\{$nucléaire, défense$\}$;
- cl est une classification prise dans un ensemble **totalement ordonné**, par exemple : $\{$non classifié, confidentiel, secret, top secret$\}$.

Pour un objet o, la classification $c(o)$ est un moyen de représenter le danger que peut constituer la divulgation de l'information contenue dans cet objet ; pour un sujet s, c'est une habilitation $h(s)$ qui désigne la confiance qui lui est accordée. Les niveaux constituent un treillis pour la relation d'ordre partielle dite relation de domination notée $\preceq$ et définie par :
si $n = (cl, C)$ et $n' = (cl', C')$, alors $n \preceq n'$ (" n' domine n " ou " n est dominé par n' ") si et seulement si $cl \leq cl'$ et $C \subseteq C'$ c'est-à-dire la classification cl du niveau n est plus petite que la classification cl' du niveau n', et le compartiment C du niveau n est contenu dans le compartiment C' du niveau n'.
La classification $c(o)$ d'un objet o ainsi que l'habilitation $h(s)$ d'un sujet s sont toutes deux des niveaux. Ils peuvent donc être comparer via la relation de domination.

Pour un objet o, la classification $c(o)$ est un moyen de représenter le danger que peut constituer la divulgation de l'information contenue dans cet objet ; pour un sujet s, c'est une habilitation $h(s)$ qui désigne la confiance qui lui est accordée. Les niveaux constituent un treillis pour la relation d'ordre partielle dite relation de domination notée $\preceq$ et définie par :
si $n = (cl, C)$ et $n' = (cl', C')$, alors $n \preceq n'$ (" n' domine n " ou " n est dominé par n' ") si et seulement si $cl \leq cl'$ et $C \subseteq C'$ c'est-à-dire la classification cl du niveau n est plus petite que la classification cl' du niveau n', et le compartiment C du niveau n est contenu dans le compartiment C' du niveau n'.
La classification $c(o)$ d'un objet o ainsi que l'habilitation $h(s)$ d'un sujet s sont toutes deux des niveaux. Ils peuvent donc être comparer via la relation de domination.

Pour un objet o, la classification $c(o)$ est un moyen de représenter le danger que peut constituer la divulgation de l'information contenue dans cet objet ; pour un sujet s, c'est une habilitation $h(s)$ qui désigne la confiance qui lui est accordée. Les niveaux constituent un treillis pour la relation d'ordre partielle dite relation de domination notée $\preceq$ et définie par :

si $n = (cl, C)$ et $n' = (cl', C')$, alors $n \preceq n'$ (" n' domine n " ou " n est dominé par n' ") si et seulement si $cl \leq cl'$ et $C \subseteq C'$ c'est-à-dire la classification cl du niveau n est plus petite que la classification cl' du niveau n', et le compartiment C du niveau n est contenu dans le compartiment C' du niveau n'.

La classification $c(o)$ d'un objet o ainsi que l'habilitation $h(s)$ d'un sujet s sont toutes deux des niveaux. Ils peuvent donc être comparer via la relation de domination.

Pour un objet o, la classification $c(o)$ est un moyen de représenter le danger que peut constituer la divulgation de l'information contenue dans cet objet ; pour un sujet s, c'est une habilitation $h(s)$ qui désigne la confiance qui lui est accordée. Les niveaux constituent un treillis pour la relation d'ordre partielle dite relation de domination notée $\preceq$ et définie par :

si $n = (cl, C)$ et $n' = (cl', C')$, alors $n \preceq n'$ (" n' domine n " ou " n est dominé par n' ") si et seulement si $cl \leq cl'$ et $C \subseteq C'$ c'est-à-dire la classification cl du niveau n est plus petite que la classification cl' du niveau n', et le compartiment C du niveau n est contenu dans le compartiment C' du niveau n'.

La classification $c(o)$ d'un objet o ainsi que l'habilitation $h(s)$ d'un sujet s sont toutes deux des niveaux. Ils peuvent donc être comparer via la relation de domination.

Pour un objet o, la classification $c(o)$ est un moyen de représenter le danger que peut constituer la divulgation de l'information contenue dans cet objet ; pour un sujet s, c'est une habilitation $h(s)$ qui désigne la confiance qui lui est accordée. Les niveaux constituent un treillis pour la relation d'ordre partielle dite relation de domination notée $\preceq$ et définie par :
si $n = (cl, C)$ et $n' = (cl', C')$, alors $n \preceq n'$ (" n' domine n " ou " n est dominé par n' ") si et seulement si $cl \leq cl'$ et $C \subseteq C'$ c'est-à-dire la classification cl du niveau n est plus petite que la classification cl' du niveau n', et le compartiment C du niveau n est contenu dans le compartiment C' du niveau n'.
La classification $c(o)$ d'un objet o ainsi que l'habilitation $h(s)$ d'un sujet s sont toutes deux des niveaux. Ils peuvent donc être comparer via la relation de domination.

Pour un objet o, la classification $c(o)$ est un moyen de représenter le danger que peut constituer la divulgation de l'information contenue dans cet objet ; pour un sujet s, c'est une habilitation $h(s)$ qui désigne la confiance qui lui est accordée. Les niveaux constituent un treillis pour la relation d'ordre partielle dite relation de domination notée $\preceq$ et définie par :
si $n = (cl, C)$ et $n' = (cl', C')$, alors $n \preceq n'$ (" n' domine n " ou " n est dominé par n' ") si et seulement si $cl \leq cl'$ et $C \subseteq C'$ c'est-à-dire la classification cl du niveau n est plus petite que la classification cl' du niveau n', et le compartiment C du niveau n est contenu dans le compartiment C' du niveau n'.
La classification $c(o)$ d'un objet o ainsi que l'habilitation $h(s)$ d'un sujet s sont toutes deux des niveaux. Ils peuvent donc être comparer via la relation de domination.

Pour un objet o, la classification $c(o)$ est un moyen de représenter le danger que peut constituer la divulgation de l'information contenue dans cet objet ; pour un sujet s, c'est une habilitation $h(s)$ qui désigne la confiance qui lui est accordée. Les niveaux constituent un treillis pour la relation d'ordre partielle dite relation de domination notée $\preceq$ et définie par :
si $n = (cl, C)$ et $n' = (cl', C')$, alors $n \preceq n'$ (" n' domine n " ou " n est dominé par n' ") si et seulement si $cl \leq cl'$ et $C \subseteq C'$ c'est-à-dire la classification cl du niveau n est plus petite que la classification cl' du niveau n', et le compartiment C du niveau n est contenu dans le compartiment C' du niveau n'.
La classification $c(o)$ d'un objet o ainsi que l'habilitation $h(s)$ d'un sujet s sont toutes deux des niveaux. Ils peuvent donc être comparer via la relation de domination.

Pour un objet o, la classification $c(o)$ est un moyen de représenter le danger que peut constituer la divulgation de l'information contenue dans cet objet ; pour un sujet s, c'est une habilitation $h(s)$ qui désigne la confiance qui lui est accordée. Les niveaux constituent un treillis pour la relation d'ordre partielle dite relation de domination notée $\preceq$ et définie par :
si $n = (cl, C)$ et $n' = (cl', C')$, alors $n \preceq n'$ (" n' domine n " ou " n est dominé par n' ") si et seulement si $cl \leq cl'$ et $C \subseteq C'$ c'est-à-dire la classification cl du niveau n est plus petite que la classification cl' du niveau n', et le compartiment C du niveau n est contenu dans le compartiment C' du niveau n'.
La classification $c(o)$ d'un objet o ainsi que l'habilitation $h(s)$ d'un sujet s sont toutes deux des niveaux. Ils peuvent donc être comparer via la relation de domination.

Pour un objet o, la classification $c(o)$ est un moyen de représenter le danger que peut constituer la divulgation de l'information contenue dans cet objet ; pour un sujet s, c'est une habilitation $h(s)$ qui désigne la confiance qui lui est accordée. Les niveaux constituent un treillis pour la relation d'ordre partielle dite relation de domination notée $\preceq$ et définie par :
si $n = (cl, C)$ et $n' = (cl', C')$, alors $n \preceq n'$ (" n' domine n " ou " n est dominé par n' ") si et seulement si $cl \leq cl'$ et $C \subseteq C'$ c'est-à-dire la classification cl du niveau n est plus petite que la classification cl' du niveau n', et le compartiment C du niveau n est contenu dans le compartiment C' du niveau n'.
La classification $c(o)$ d'un objet o ainsi que l'habilitation $h(s)$ d'un sujet s sont toutes deux des niveaux. Ils peuvent donc être comparer via la relation de domination.

Pour un objet o, la classification $c(o)$ est un moyen de représenter le danger que peut constituer la divulgation de l'information contenue dans cet objet ; pour un sujet s, c'est une habilitation $h(s)$ qui désigne la confiance qui lui est accordée. Les niveaux constituent un treillis pour la relation d'ordre partielle dite relation de domination notée $\preceq$ et définie par :
si $n = (cl, C)$ et $n' = (cl', C')$, alors $n \preceq n'$ (" n' domine n " ou " n est dominé par n' ") si et seulement si $cl \leq cl'$ et $C \subseteq C'$ c'est-à-dire la classification cl du niveau n est plus petite que la classification cl' du niveau n', et le compartiment C du niveau n est contenu dans le compartiment C' du niveau n'.
La classification $c(o)$ d'un objet o ainsi que l'habilitation $h(s)$ d'un sujet s sont toutes deux des niveaux. Ils peuvent donc être comparer via la relation de domination.

Les objectifs de sécurité de cette politique sont les suivants :

- interdire toute fuite d'information d'un objet possédant une certaine classification vers un objet possédant un niveau de classification inférieur ;
- interdire à tout sujet possédant une certaine habilitation d'obtenir des informations d'un objet d'un niveau de classification supérieur à cette habilitation.

Les objectifs de sécurité de cette politique sont les suivants :

- interdire toute fuite d'information d'un objet possédant une certaine classification vers un objet possédant un niveau de classification inférieur ;
- interdire à tout sujet possédant une certaine habilitation d'obtenir des informations d'un objet d'un niveau de classification supérieur à cette habilitation.

Les objectifs de sécurité de cette politique sont les suivants :

- interdire toute fuite d'information d'un objet possédant une certaine classification vers un objet possédant un niveau de classification inférieur ;
- interdire à tout sujet possédant une certaine habilitation d'obtenir des informations d'un objet d'un niveau de classification supérieur à cette habilitation.

Le schéma d'autorisation associé à ces objectifs de sécurité en découle directement. On considère que l'on peut distinguer vis-à-vis de la confidentialité, parmi les différentes opérations qu'un sujet peut effectuer sur un objet, les opérations de lecture et d'écriture, et on introduit les règles suivantes, incontournables même par les propriétaires des informations :

- Propriété simple : un sujet s ne peut lire un objet o que si son habilitation $h(s)$ domine la classification $c(o)$ de l'objet : on note alors $(s, o, \texttt{lire}) \Rightarrow c(o) \preceq h(s)$;
- Propriété étoile : un sujet s ne peut lire un objet o_j et écrire un autre o_k que si la classification de l'objet o_k domine celle de l'objet o_j : on note alors $(s, o_j, \texttt{lire})$ et $(s, o_k, \text{écrire}) \Rightarrow c(o_j) \preceq c(o_k)$.

Le schéma d'autorisation associé à ces objectifs de sécurité en découle directement. On considère que l'on peut distinguer vis-à-vis de la confidentialité, parmi les différentes opérations qu'un sujet peut effectuer sur un objet, les opérations de lecture et d'écriture, et on introduit les règles suivantes, incontournables même par les propriétaires des informations :

- Propriété simple : un sujet s ne peut lire un objet o que si son habilitation $h(s)$ domine la classification $c(o)$ de l'objet : on note alors $(s, o, \texttt{lire}) \Rightarrow c(o) \preceq h(s)$;
- Propriété étoile : un sujet s ne peut lire un objet o_j et écrire un autre o_k que si la classification de l'objet o_k domine celle de l'objet o_j : on note alors $(s, o_j, \texttt{lire})$ et $(s, o_k, \text{écrire}) \Rightarrow c(o_j) \preceq c(o_k)$.

Le schéma d'autorisation associé à ces objectifs de sécurité en découle directement. On considère que l'on peut distinguer vis-à-vis de la confidentialité, parmi les différentes opérations qu'un sujet peut effectuer sur un objet, les opérations de lecture et d'écriture, et on introduit les règles suivantes, incontournables même par les propriétaires des informations :

- Propriété simple : un sujet s ne peut lire un objet o que si son habilitation $h(s)$ domine la classification $c(o)$ de l'objet : on note alors $(s, o, \texttt{lire}) \Rightarrow c(o) \preceq h(s)$;
- Propriété étoile : un sujet s ne peut lire un objet o_j et écrire un autre o_k que si la classification de l'objet o_k domine celle de l'objet o_j : on note alors $(s, o_j, \texttt{lire})$ et $(s, o_k, \text{écrire}) \Rightarrow c(o_j) \preceq c(o_k)$.

Le schéma d'autorisation associé à ces objectifs de sécurité en découle directement. On considère que l'on peut distinguer vis-à-vis de la confidentialité, parmi les différentes opérations qu'un sujet peut effectuer sur un objet, les opérations de lecture et d'écriture, et on introduit les règles suivantes, incontournables même par les propriétaires des informations :

- Propriété simple : un sujet s ne peut lire un objet o que si son habilitation $h(s)$ domine la classification $c(o)$ de l'objet : on note alors $(s, o, \texttt{lire}) \Rightarrow c(o) \preceq h(s)$;
- Propriété étoile : un sujet s ne peut lire un objet o_j et écrire un autre o_k que si la classification de l'objet o_k domine celle de l'objet o_j : on note alors $(s, o_j, \texttt{lire})$ et $(s, o_k, \text{écrire}) \Rightarrow c(o_j) \preceq c(o_k)$.

Le schéma d'autorisation associé à ces objectifs de sécurité en découle directement. On considère que l'on peut distinguer vis-à-vis de la confidentialité, parmi les différentes opérations qu'un sujet peut effectuer sur un objet, les opérations de lecture et d'écriture, et on introduit les règles suivantes, incontournables même par les propriétaires des informations :

- Propriété simple : un sujet s ne peut lire un objet o que si son habilitation $h(s)$ domine la classification $c(o)$ de l'objet : on note alors $(s, o, \texttt{lire}) \Rightarrow c(o) \preceq h(s)$;
- Propriété étoile : un sujet s ne peut lire un objet o_j et écrire un autre o_k que si la classification de l'objet o_k domine celle de l'objet o_j : on note alors $(s, o_j, \texttt{lire})$ et $(s, o_k, \text{écrire}) \Rightarrow c(o_j) \preceq c(o_k)$.

Le schéma d'autorisation associé à ces objectifs de sécurité en découle directement. On considère que l'on peut distinguer vis-à-vis de la confidentialité, parmi les différentes opérations qu'un sujet peut effectuer sur un objet, les opérations de lecture et d'écriture, et on introduit les règles suivantes, incontournables même par les propriétaires des informations :

- Propriété simple : un sujet s ne peut lire un objet o que si son habilitation $h(s)$ domine la classification $c(o)$ de l'objet : on note alors $(s, o, \texttt{lire}) \Rightarrow c(o) \preceq h(s)$;
- Propriété étoile : un sujet s ne peut lire un objet o_j et écrire un autre o_k que si la classification de l'objet o_k domine celle de l'objet o_j : on note alors $(s, o_j, \texttt{lire})$ et $(s, o_k, \text{écrire}) \Rightarrow c(o_j) \preceq c(o_k)$.

Le schéma d'autorisation associé à ces objectifs de sécurité en découle directement. On considère que l'on peut distinguer vis-à-vis de la confidentialité, parmi les différentes opérations qu'un sujet peut effectuer sur un objet, les opérations de lecture et d'écriture, et on introduit les règles suivantes, incontournables même par les propriétaires des informations :

- Propriété simple : un sujet s ne peut lire un objet o que si son habilitation $h(s)$ domine la classification $c(o)$ de l'objet : on note alors $(s, o, \texttt{lire}) \Rightarrow c(o) \preceq h(s)$;
- Propriété étoile : un sujet s ne peut lire un objet o_j et écrire un autre o_k que si la classification de l'objet o_k domine celle de l'objet o_j : on note alors $(s, o_j, \texttt{lire})$ et $(s, o_k, \text{écrire}) \Rightarrow c(o_j) \preceq c(o_k)$.

Le schéma d'autorisation associé à ces objectifs de sécurité en découle directement. On considère que l'on peut distinguer vis-à-vis de la confidentialité, parmi les différentes opérations qu'un sujet peut effectuer sur un objet, les opérations de lecture et d'écriture, et on introduit les règles suivantes, incontournables même par les propriétaires des informations :

- Propriété simple : un sujet s ne peut lire un objet o que si son habilitation $h(s)$ domine la classification $c(o)$ de l'objet : on note alors $(s, o, \texttt{lire}) \Rightarrow c(o) \preceq h(s)$;
- Propriété étoile : un sujet s ne peut lire un objet o_j et écrire un autre o_k que si la classification de l'objet o_k domine celle de l'objet o_j : on note alors
$(s, o_j, \texttt{lire})$ et $(s, o_k, \text{écrire}) \Rightarrow c(o_j) \preceq c(o_k)$.

La propriété simple interdit de lire des informations d'une classification supérieure à l'habilitation, et la propriété étoile empêche les flux d'information d'une classification donnée vers une classification inférieure, ce qui constituerait une fuite d'information.

La propriété simple interdit de lire des informations d'une classification supérieure à l'habilitation, et la propriété étoile empêche les flux d'information d'une classification donnée vers une classification inférieure, ce qui constituerait une fuite d'information.