Le contrôle d'accès a posteriori : une alliance entre politique de sécurité et analyse des logs.

Auteur : Farah DERNAIKA

WG-5 EFCSE

EUROPEAN CYBER SECURITY MONTH

[log] vient de l'anglais log file, journal en français : fichier informatique en format texte classique, contenant tous les enregistrements des événements qui concernent un système informatique (programme, application, serveur, activité d'un réseau informatique…) et toutes les actions qui résultent de ces événements.

Be secure

Be critical

A propos de l'auteur

Farah DERNAIKA

Titulaire d'un diplôme d'ingénieur en Télécommunications et Réseaux et d'un Master II en Cybersécurité de l'Ecole Supérieure d'Ingénieurs de Beyrouth. Farah réalise sa thèse de doctorat en Cybersécurité à IMT Atlantique, Rennes, sous la direction de Nora et Frédéric Cuppens, travaillant sur l'analyse des logs pour le contrôle d'accès a posteriori.

La thèse se déroule en collaboration avec le groupe be-ys, Farah y travaille avec l'entité dédiée recherche et innovation be-studys, ainsi qu'avec la filiale be-almerys, expert en technologies appliquées à l'assurance et à la santé.

Le contrôle d'accès a posteriori, qu'est-ce que c'est ?

Les modes de contrôles d'accès traditionnels consistent à vérifier les permissions des utilisateurs avant de leur donner accès aux ressources d'informations. Ce sont des modes préventifs pour lesquels toute tentative d'accès au système d'information qui violerait la politique de sécurité sera systématiquement bloquée. Or dans certaines situations sensibles, comme par exemple le cas des urgences dans le domaine médical, les professionnels ont besoin d'avoir un accès immédiat, et sans risque de rejet. Il nous faut donc avoir recours à un mode de contrôle d'accès plus flexible.

Le contrôle d'accès a posteriori répond à ce besoin. Il consiste à surveiller les actions des utilisateurs, une fois qu'ils ont accès au système d'information. Les principes de ce contrôle d'accès reposent sur des mécanismes de monitoring efficaces, permettant de détecter toute violation de la politique de sécurité déployée. Une politique de sanction et de réparation dissuasive est aussi associée pour que les utilisateurs ne soient pas tentés de violer la politique de sécurité.

Les fichiers journaux, communément appelés logs dans la littérature scientifique, enregistrent tout ce qui se passe dans le système d'information. Par conséquent, ces fichiers sont les premières sources de données que les spécialistes de sécurité consultent quand ils soupçonnent une déviance, ou un problème quelconque.

Dans la littérature, le contrôle d'accès a posteriori se constitue de trois étapes : le traitement des logs, l'analyse des logs, et l'assignation de responsabilité. Le but de la première étape est d'extraire des informations pertinentes des logs, tandis que la deuxième consiste à faire une analyse pour déterminer s'il y a eu violation ou non. Dans cette dernière, l'investigateur peut obtenir des preuves, démontrant que l'utilisateur n'a pas violé la politique de sécurité et que le problème provenait d'une malveillance externe ou d'une erreur du système, ou qu'il y a bien eu une violation. Dans ce cas, il peut se pencher sur les circonstances dans lesquelles l'utilisateur a fait telle ou telle action, et montrer s'il y a danger ou non. Enfin, la dernière étape, l'assignation de responsabilité, consiste à appliquer les sanctions et les réparations, une fois que la décision est prise.

Malheureusement, ce mode de contrôle d'accès peut prêter à confusion. Il faut noter qu'en déployant ce type de contrôle d'accès, on est dans un mode « Détective ». Un point important qui rend ces enquêtes douteuses, est que l'analyse des logs est basée essentiellement sur l'expertise de la personne qui l'effectue. Par exemple, l'administrateur système peut utiliser une liste générique de contrôles de sécurité qui n'est pas nécessairement adaptée au système cible.

Ainsi, plusieurs travaux ont été menés pour détecter les anomalies à partir des journaux de bord tels que le process mining et les techniques de machine learning, qui ont également été intégrés dans certains outils d'analyse des logs, comme Splunk. Pourtant, ces méthodes nécessitent toujours une intervention humaine, pour mener une analyse plus approfondie afin de décider ce qui est vraiment normal ou anormal.

En matière de contrôle d'accès, et pour ne pas avoir une analyse biaisée, le référent par rapport auquel l'analyse des logs sera effectuée, est la politique de sécurité. Une comparaison entre ce qui est logué et ce qui est défini dans la politique de sécurité est donc nécessaire.

Comme nous venons de le voir, le contrôle d'accès à posteriori repose sur deux piliers majeurs : la politique de sécurité et l'analyse des fichiers logs.

La politique de sécurité : un élément à ne pas négliger

Dans un système d'information, une politique de sécurité correspond à un ensemble de règles définissant des exigences de contrôle d'accès (permissions, interdictions), et de contrôle d'usage (obligations) relatives aux actions d'un utilisateur sur ce système d'information. Elle peut être représentée selon différents modèles, parmi lesquels, DAC [1], MAC [2], RBAC [3], ABAC [4], et OrBAC [5]. Toutefois, contrôler la bonne application des règles de la politique de sécurité d'un système d'information reste un besoin incontournable pour assurer les propriétés de confidentialité, d'intégrité et de disponibilité. Ainsi, sans contrôle de sécurité, nous nous exposons à plusieurs fraudes, notamment dans les domaines bancaire et financier.

Normalement, les organisations définissent les politiques de sécurité en exprimant les demandes d'accès qui doivent être autorisées. Afin de les mettre en œuvre, l'administrateur des politiques les renseigne pour qu'elles soient interprétables par la machine. C'est ce qu'on appelle l'implémentation. Cependant, pendant la mise en œuvre, l'organisation peut subir plusieurs changements, et des erreurs humaines peuvent se produire. Par exemple, un manque de compétences chez le personnel peut être en cause, ce qui introduit plusieurs erreurs dans la politique implémentée. Ainsi, on peut classer ces erreurs selon deux types : les autorisations incorrectes qui présentent les demandes d'accès autorisées par la politique implémentée et qui ne doivent pas l'être, et les refus incorrects représentés par les demandes d'accès non autorisées par la politique implémentée et qui doivent l'être.

Intuitivement, les autorisations incorrectes sont les plus difficiles à détecter et les plus problématiques car elles peuvent être utilisées pour lire des données confidentielles et attaquer le système. Par contre, les refus incorrects sont généralement moins problématiques. Lorsque l'utilisateur découvre qu'une demande d'accès valide n'est pas

autorisée, il la signale à l'administrateur pour modifier ce qui est implémenté. Il faut toutefois faire attention à ce que ces modifications, qui peuvent nécessiter du temps de mise en œuvre, puissent rendre la politique déployée alambiquée.

De plus, il n'est pas anormal de ne pas trouver de documentation qui présente les habilitations d'une application spécifique. Les concepteurs de l'application sont plus centrés sur l'implémentation. Ils ne documentent pas forcement toutes les tâches réalisées. Il est donc primordial d'avoir un document qui présente la politique de sécurité, gouvernant l'application, et la comparer avec la politique implémentée. Des techniques de « Role Engineering » [6], ont été proposées pour concevoir une politique à partir des habilitations mise en œuvre et vice versa. N'oublions pas qu'il faut penser aussi à mettre à jour la documentation lors des modifications des implémentations.

Par ailleurs, le contrôle a posteriori sera plus pertinent si on utilise des politiques de sécurité plus expressive comme ABAC ou OrBAC. Ces modèles-là peuvent être plus complexes qu'un simple modèle RBAC, où les permissions sont assignées aux rôles d'utilisateurs, mais expriment mieux le contexte des actions réalisées, au vu des contraintes qui peuvent être ajoutées relativement aux attributs et aux conditions environnementales définies.

L'importance de l'analyse des logs

La journalisation est un aspect essentiel de l'archivage électronique, car elle contribue à donner à l'archive, sa valeur probante. Le système enregistre, de manière chronologique, tous les événements associés au système d'exploitation, aux applications en cours d'exécution et au réseau auquel il est connecté. Ainsi, plusieurs types de logs peuvent être distingués : les logs applicatifs qui enregistrent ce qui se passe dans le code de l'application, les logs de trafic http qui tracent les requêtes envoyées et reçues par l'utilisateur, les logs de trafic de base de données, représentant par exemple, les appels au serveur SQL par exemple, et les logs de données maintenant, les actions des utilisateurs.

Pour généraliser, un fichier journal enregistre « Ce qui s'est passé ? Quand ? Et par Qui ? » dans le système. Il peut être référencé à des fins de diagnostic, de pistes d'audit et d'investigation en cas d'activités malveillantes, d'attaques système, ou de violations de la sécurité. Il peut également être utilisé à des fins comptables. C'est pourquoi, un fichier journal s'appelle le « juge de la paix ».

L'intérêt que porte l'analyse des logs peut différer selon le contexte dans lequel on fait l'investigation. Dans le cas d'un contrôle a posteriori, le but sera de détecter toute potentielle violation de la politique de sécurité, afin d'appliquer les sanctions et/ou réparation si besoin.

Souvent oubliée, l'analyse des logs est pourtant la première étape d'un bon référencement, dont il faut « déchiffrer » le contenu. C'est évidemment une source de confiance, offrant des données intègres, si l'on considère que *tous les accès sont consignés dans les logs*.

D'ici, on attire l'attention sur une problématique importante à laquelle nous allons nous intéresser dans ce qui suit.

Malheureusement, la majorité des entreprises sous-estiment l'importance des logs lors de la conception d'un produit. Pour eux, la génération des logs est faite juste « histoire de ».

Il faut se pencher sur l'aspect critique de manque de preuve (logs), surtout dans le cas d'activités frauduleuse perpétrées à cause d'un problème de droits d'accès et l'incapacité de pénaliser le malfaiteur.

Pour cela, il convient de mentionner que pour bien gérer les accès, il faut avoir un niveau de traçabilité suffisant pour que l'analyse soit pertinente. Des logs corrompus impactent la prise de décisions.

Il est également important de savoir quoi loguer. Il ne faut surtout pas regrouper tous les types de logs dans un seul fichier. Les fichiers logs sont caractérisés par le bruit qui peut s'y appliquer. Dans le cas contraire, pas la peine de se noyer dans une longue investigation, on vivrait dans le paradis ! Mais il est important aussi d'avoir des informations plus ou moins complètes, qui permettent de reconstituer, au moins la session de l'utilisateur. Ceci a été confirmé par la CNIL [7] : « la journalisation doit concerner, au minimum, les accès des utilisateurs en incluant leur identifiant, la date et l'heure de connexion, la date et l'heure de leur déconnexion ». En plus, il vaut mieux loguer si l'action demandée a été autorisée par le système, ou non.

Cependant, il ne faut pas tout loguer ! Il peut être utile d'avoir des logs simplifiés, qui présentent ce qui est utile, surtout au niveau du contrôle d'accès. Comme bonne pratique, l'ANSSI a publié des recommandations de sécurité pour la mise en œuvre d'un système de journalisation [8].

Même si nous logons les informations utiles dans les fichiers journaux, l'analyse des logs reste une tâche difficile à cause de son grand volume, d'où le besoin d'un traitement. De plus, celle-ci est encore compliquée car ces journaux sont généralement faiblement structurés, utilisant une variété de formats et de terminologies incohérents, et sont répartis sur différents fichiers et systèmes. Pour ces raisons, plusieurs travaux s'intéressent à cette problématique d'analyse sémantique, en utilisant des ontologies, au niveau des traitements et d'extractions d'information des logs [9][10]. L'ajout de cette couche sémantique facilite l'interprétation des logs, surtout lorsqu'une politique expressive gouverne le processus.

Enfin, même si plusieurs mécanismes sont déployés pour faciliter la tâche de traitement et d'extraction d'information des logs, leur analyse reste un défi dans le cas d'un contrôle d'accès a posteriori. Comme déjà mentionné, les difficultés résident dans la multiplicité des sources et des formats de logs, l'expression du format de référence de la politique de sécurité, et l'établissement des liens entre les logs et la politique. A ceux-ci, vient s'ajouter l'idée d'avoir une information qui manque dans les logs. Comment pourra-t-on alors décider s'il y a eu violation ou non ? En outre, avec la nouvelle réglementation RGPD, des données présentent dans les logs peuvent être rendues anonymes ou chiffrées, ce qui rend l'investigation inévitable. Comment pourrait-on, dans ce cas, comparer les résultats de fouille de logs avec la politique de sécurité ?

--------- Références ---------

[1] Sabrina De Capitani di Vimercati. 2011. Discretionary Access Control Policies (DAC). Springer US, Boston, MA, 356–358. https://doi.org/10.1007/978-1-4419-5906-5_817

[2] D Bell, Leonard J LaPadula, MBen-Ari, G Benson, G Benson, B Appelbe, I Akyildiz, C Date, D Denning, P Denning, et al. 1988. Secure computer system unified exposition and multics interpretation. Commun. ACM 1 (1988), 271–280.

[3] David Ferraiolo, Janet Cugini, and D Richard Kuhn. 1995. Role-based access control (RBAC): Features and motivations. In Proceedings of 11th annual computer security application conference. 241–48.

[4] Vincent C Hu, David Ferraiolo, Rick Kuhn, Arthur R Friedman, Alan J Lang, Margaret M Cogdell, Adam Schnitzer, Kenneth Sandlin, Robert Miller, Karen Scarfone, et al. 2013. Guide to attribute based access control (abac) definition and considerations (draft). NIST special publication 800, 162 (2013).

[5] Anas Abou El Kalam, Rania El Baida, Philippe Balbiani, Salem Benferhat, Frédéric Cuppens, Yves Deswarte, Alexandre Miege, Claire Saurel, and Gilles Trouessin. 2003. Or-BAC: un modèle de contrôle d'accès basé sur les organisations. Cahiers francophones de la recherche en sécurité de l'information 1 (2003), 30–43.

[6] Role Engineering for Enterprise Security Management (Book)

[7] https://www.cnil.fr/fr/securite-tracer-les-acces-et-gerer-les-incidents

[8] https://www.ssi.gouv.fr/uploads/IMG/pdf/NP_Journalisation_NoteTech.pdf

[9] Farah Dernaika, Nora Cuppens-Boulahia, Frédéric Cuppens, and Olivier Raynaud. 2019. Semantic Mediation for A Posteriori Log Analysis. In Proceedings of the 14th International Conference on Availability, Reliability and Security. ACM, 88.

[10] Hanieh Azkia, Nora Cuppens-Boulahia, Frédéric Cuppens, Gouenou Coatrieux : Ontology Based Log Content Extraction Engine for a posteriori Security Control, 24th European Medial Informatics Conference – MIE2012, August 26th-29th, 2012 Pisa, Italy.