Les failles de sécurité

Veille technologique

Compétences validé :

5. Organiser son développement professionnel
   - Mettre en œuvre des outils et stratégies de veille informationnelle

1. Qu’est-ce qu’une faille de sécurité ?

Une faille de sécurité ou vulnérabilité informatique concerne toute faiblesse d’un système (ex : un logiciel), qui permettrait à une personne malveillante de l’exploiter et d'altérer le fonctionnement du système ou encore d'accéder à des données sensibles.

Une faille est généralement involontaire et peut résider dans la conception du logiciel ou un problème plus profond au niveau matériel, certaines failles ne peuvent être corrigées compte tenu de la spécificité du matériel, du logiciel, du protocole...

2. Qu’est-ce qu’une vulnérabilité CVE ?

Le système CVE (Common Vulnerabilities and Exposures) permet de recenser toutes les failles et les menaces liées à la sécurité des systèmes d’information. Pour ce faire, un identifiant unique est attribué à chaque faille.

Ainsi, les failles de sécurité informatique sont présentées par un identifiant unique, composé de l’année de la découverte et du numéro d'identification de l’organisme.

Lorsqu’une faille est détectée mais qu’elle n’a pas encore fait l’objet d’une publication, celle-ci est nommée “0 day”.

Ensuite, l’entreprise qui a découvert cette faille, peut demander à l’entreprise concernée par celle-ci si elle accepte la faille. Si c’est le cas, l’entreprise la soumet à l’organisation MITRE. C’est elle qui décidera, en fonction de la notoriété et de l’utilisation du logiciel utilisé, de la publier ou non.

MITRE fournit alors un identifiant unique CVE à l’entreprise et celle-ci doit remplir un formulaire, dans lequel elle doit détailler la faille. Elle peut se tourner vers la personne ou l’entreprise qui a découvert la faille.

3. Les Différents Types et Niveaux des Failles

Une faille de sécurité peut concerner tous les aspects de l’informatique, allant des protocoles réseaux aux logiciels, des CMS (Content Management System) aux langages de programmation.

De plus, la criticité des failles est généralement évaluée sur une échelle de 1 à 10 selon le système de notation standardisé CVSS (Common Vulnerability Scoring System) qui prend en compte des facteurs temporels tels que la divulgation, l'exploitation et les contre-mesures.

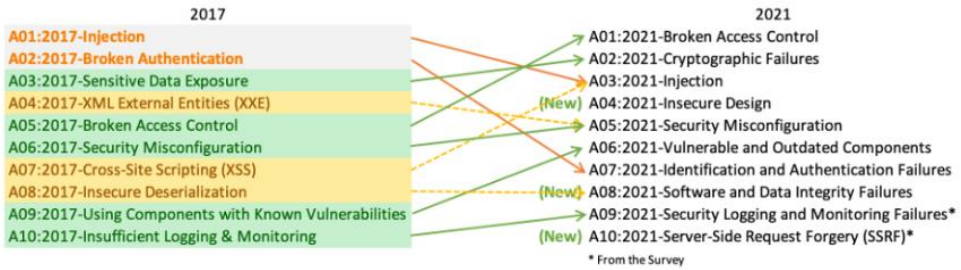
4. OWASP

Open Web Application Security Project est une communauté en ligne spécialisée dans la cybersécurité qui s'est donnée pour mission de recommander aux différents utilisateurs (internautes, administrateurs…) des méthodes et outils permettant de contrôler le niveau de sécurisation de ses applications WEB.

La communauté OWASP a été créée en 2001 avec pour projet de donner des éléments, informations et solutions, aux développeurs pour prendre des décisions en matière de sécurisation de leur applications Web comme le Top 10 OWASP.

Le top 10 OWASP informe sur les dix failles de sécurité les plus recensées sur l'année précédente. L'actuel top 10 porte donc sur les failles de 2021. Celui-ci est comparé avec le top 10 de l'année 2017.

Nous retrouvons certaines failles, celle-ci ne sont plus à la même place. Nous pouvons également voir que de nouvelles failles ont fait leur apparition.



5. Exemples de vulnérabilités

CVE-2021-4034 :

Nommée Pwnkit, il s'agit d'une énorme faille de sécurité située dans pkexec qui a été découverte en 2021 par Qualys. La faille existe depuis douze ans et touche potentiellement toutes les distributions Linux avec un niveau de criticité de 7.5/10.

Pkexec, fait parti du paquet Polkit, anciennement PolicyKit qui définit et gère les politiques qui permettent aux processus et applications s'exécutant avec des droits restreints de

communiquer avec des services privilégiés du système. Celui-ci a été créé pour permettre aux développeur d'exécuter des actions qui nécessitent des privilèges élevés.

Pkexec, quant à lui, est une alternative à sudo. En effet, il permet à un utilisateur autorisé d'exécuter une commande en tant qu'autre utilisateur.

La faille concerne une élévation de privilège, c'est-à-dire que cela permet à un utilisateur authentifié d'obtenir les pleins privilèges root, ce qui lui permet d'exécuter en tant qu'administrateur.

L'origine de la vulnérabilité se trouverait dans un problème de corruption de mémoire dans la façon dont pkexec gère ses arguments en ligne de commande. "Pkexec" est une commande qui exécute une fonction appelée main() qui contient un bout de code dans lequel, deux arguments sont définis : "argc" et "argv". Cette fonction est écrite en langage C. Dans celui-ci. Le premier argument, est le nom du fichier. Tandis que le second, est un argument appelé dans la fonction.

En C, on doit définir la taille du deuxième argument, car cela permet de réserver un espace mémoire sur le disque.

Si on lui définit un espace de 4 caractère, et qu'en fait, l'argument en fait 8, cela écrit sur les données qui sont après. Lors de l'utilisation de pkexec, l'attaquant fait en sorte de dépasser cette mémoire allouée. La faille réside dans le fait, que toute commande écrite en dehors de cet espace mémoire, est exécutée en tant que root.

Pour protéger son serveur Linux de toute attaque par l'intermédiaire de "pkexec", il est nécessaire de corriger cette vulnérabilités avec un patch correctif qui a été mis en place pour les distributions Linux touchées en mettant simplement à jour le paquet concerné ou le système complet.

Il existe toutefois une méthode de contournement pour les personnes qui ne serait pas en mesure de se procurer le patch. Celle-ci consiste à supprimer les droits de pkexec, c'est-à-dire aussi bien les droits en écriture qu'en lecture. Il suffit de taper la commande suivante dans son terminal : chmod 0755 /usr/bin/pkexec, cela permet de lui retirer les droits SUID représentés par le "s" dans -rwsr.

CVE-2021-44228 Log4Shell :

Qu'est-ce que Log4J et Log4Shell ?

Log4J est une librairie développée par Apache, pour permettre aux développeurs de gérer les logs de leurs applications, c'est-à-dire, de gérer les erreurs rencontrées lors du développement.

Log4Shell est une vulnérabilité détectée le 24 novembre 2021, dans la bibliothèque Log4J d'Apache. Ce dernier est le logiciel de serveur web le plus populaire et le plus utilisé par les entreprises ou particuliers développant des applications.

Son niveau de criticité s'élève à 9.3/10 sur le site de cvedetails.com et 10/10 sur le site officiel d'Apache -logging.apache.org. Au vu de la criticité de cette faille, l'ANSI - l'Agence nationale de la sécurité des systèmes d'informations - a également posté un avertissement sur son site la concernant.

En quoi consiste Log4Shell ?

Log4Shell permet à un attaquant de faire de l'exécution de code à distance. Celui-ci peut donc télécharger un ransomware, ou n'importe quel autre virus, mais également installer des logiciels malveillants voir même des bots. Ceci en prenant le contrôle du serveur à distance. En effet, Log4J se connecte sur des serveurs extérieurs pour aller récupérer des données et ce, grâce à une fonction de recherche appelée "jndi". Cette dernière utilise les protocoles réseaux pour aller chercher ces informations.

Or, aucun contrôle n'est opéré sur les données récupérées. Si le serveur distant est contrôlé par un attaquant, il peut injecter du code dans les données récupérées par Log4J. Une fois celles-ci réceptionnées, Log4J vient les exécuter.

L'attaquant peut donc faire exécuter un Reverse Shell par la librairie et prendre ainsi la main à distance sur le serveur touché.

Les protocoles exploités par cette vulnérabilités, sont les protocoles LDAP, DNS et RMI.

La mise à jour de Log4J vers la version 2.15.0 (Java 8) vient corriger cette faille.

D'autres vulnérabilités peuvent en cacher d'autres, en effet, les patchs de correction apportés à Log4J pour parer celle-ci ont apportés leurs propres vulnérabilité. On ne dénombre pas moins de quatre faille pour Log4J.

Un attaquant peut aussi combiner les failles de sécurité. Ainsi, Log4shell, qui est une faille permettant d'exécuter des commandes sur une machine ciblée avec des permissions limitée, peut être combinée à Pwnkit, qui permet une élévation de privilèges en passant root sur un système.

Ainsi, un attaquant pourrait exécuter des commandes sur n'importe quelle machine tout en ayant les privilèges de l'utilisateur Root.

6. Les conséquences des failles de sécurités.

Lorsqu'une vulnérabilité est découverte, elle est généralement patchée rapidement mais il y a souvent un long délai entre cette découverte par la victime, ce qui laisse le temps à un attaquant de l'exploiter.

Ainsi, la faille Pwnkit précédemment présentée a été découverte après 12 ans et a potentiellement été exploitée par plusieurs attaquants avant d'être découverte.

Une faille peut amener à de lourdes conséquences pour les personnes et entreprises (fuite d'informations confidentielles, indisponibilité…), qui peuvent avoir des coûts importants pour ces dernières.