A08:2021 – Software and Data Integrity Failures

Groupe 5:
Younouss Athie
Mohamed Bensouda
Salimata Diallo
Ndeye Fatou Gueye
Pape Makhtar Gueye

Professeur: Pr Mendy

Plan:

I- Description:

Il s'agit d'une nouvelle catégorie pour 2021 qui se concentre sur la formulation d'hypothèses relatives aux mises à jour logicielles, aux données critiques et aux pipelines CI/CD sans vérification de l'intégrité.
En effet, les défaillances de l'intégrité des logiciels et des données sont liées au **code** et à l**'infrastructure** qui ne sont pas protégés contre les violations de l'intégrité, une fonctionnalité de mise à jour automatique et à l'exécution sur toutes les installations de mises à jour distribués par des attaquants qui peuvent de plus voir et modifier des objets ou des données qui sont codés ou sérialisés dans une structure et qui sont vulnérables à une dé-sérialisation non sécurisée.

II- Matching avec le cours:

En fonction des notions abordées dans la catégorie A08-2021, on note 4 chapitres qui les abordent :

- Chapitre 1
- Chapitre 4
- Chapitre 5
- Chapitre 6

Cette catégorie aborde la notion d'intégrité des données et des logiciels. Or dès les premiers chapitres , Chapitre 1, du cours de cryptographie la notion d'intégrité ainsi que les différents types d'attaque existant ont été abordés.

Intégrité:

C'est le service de sécurité qui s'occupe d'identifier toute altération des données. Le destinataire d'un message doit pouvoir vérifier que celui-ci n'a pas été modifié en chemin. Et ces modifications découlent du fait qu'un attaquant soit capable de faire passer un faux message pour un légitime.

Attaque:

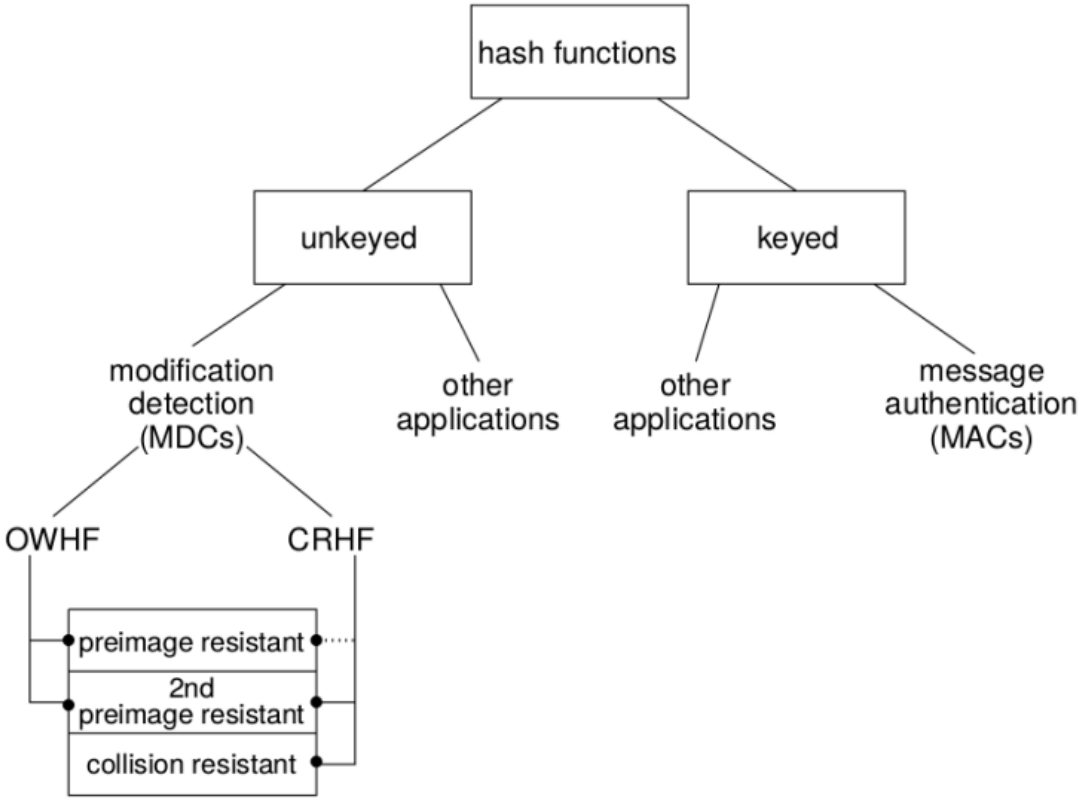
Il s'agit d'une tentative de cryptanalyse. Les attaques sont généralement catégorisées en fonction de l'action effectuée par l'attaquant. Une attaque peut donc être passive ou active.

Ici on s'intéresse aux attaques de type active qui entraîne soit une modification des informations d'une manière non autorisée, d'initier une transmission non intentionnelle ou non autorisée d'informations, d'altérer des données d'authentification, de supprimer de façon non autorisée des données, provoquer le refus d'accès à l'information pour les utilisateurs légitimes.

Cette vulnérabilité peut se reposer sur :

- Les fonction de hachage:

Notion abordée au chapitre 6 du cours où on aborde le principe : comment l'utiliser afin de garantir l'intégrité et l'authenticité des données. On note deux type de fonction : les fonctions de hachage avec clé qui nécessite en plus du message une clé privé à l'entrée et les fonction de hachage sans clé qui sont des fonctions itératifs comportant des entrées de longueur arbitraire en traitant des blocs successifs de taille fixe de l'entrée



- Signature numérique:

La signature numérique dépend du signataire et du document.
La signature numérique appartient à un seul document et donc il est impossible de découper une signature sur un message et la recoller sur un autre. Le document signé ne peut être modifié. Une signature ne peut être falsifiée ni reniée.

- Les chiffrement à clé asymétriques

Avec les signatures numériques ont peut utiliser les schémas de ElGamal qui tire sa sécurité de la difficulté de calculer des logarithmes discrets.

III- Mise en relief avec les CWE/CVE

CWE: Définition :

Common Weakness Enumeration ou CWE est une liste des vulnérabilités que l'on peut rencontrer dans les logiciels.

CVE: Définition :

Common Vulnerabilities and Exposures ou CVE est un dictionnaire des informations publiques relatives aux vulnérabilités de sécurité.

- CWE-345 Insufficient Verification of Data Authenticity

Cette notion d'authenticité des données est mise en relief dans la partie du cours relatant du chiffrement par blocs (chapitre 4) dont l'un des objectifs principaux est l'authenticité des données communiquées. En effet les données qui sont transmises doivent être chiffrées par une clé privée (celle de l'émetteur) de sorte à ce que tous les destinataires puissent s'assurer de sa provenance. De ce fait, une personne qui reçoit l'information pourra attester de son authenticité. Lorsque ce critère n'est pas respecté, on est face à ce problème d'insuffisance de la vérification de l'authenticité des données dont traite cette CWE.

- CWE-353 Missing Support for Integrity Check

Le logiciel utilise un protocole de transmission qui ne comprend pas de mécanisme de vérification de l'intégrité des données lors de la transmission, tel qu'une somme de contrôle (Chapitre 6).

- CWE-494 Download of Code Without Integrity Check & CWE-502 Deserialization of Untrusted Data

Le produit télécharge le code source ou un exécutable depuis un emplacement distant et exécute le code sans vérifier suffisamment l'origine et l'intégrité du code. Cette vérification se fait souvent par le biais des signatures numériques (chapitre 5).

- CWE-565 et CWE-784 : Reliance on Cookies without Validation and Integrity Checking and Reliance on Cookies without Validation and Integrity Checking in a Security Decision

Dans ce type de vulnérabilité, les attaquants peuvent facilement modifier les cookies, dans le navigateur ou en implémentant le code côté client en dehors du navigateur. Ils peuvent contourner les mécanismes de protection tels que l'autorisation et l'authentification en modifiant le cookie pour qu'il contienne une valeur attendue. On peut voir ces notions de sécurité au niveau du chapitre 1, qui donne leur définition et concept général de sécurité.

- CWE-829 Inclusion of Functionality from Untrusted Control Sphere

Lors de l'inclusion de fonctionnalités tierces, telles qu'un widget Web, une bibliothèque ou une autre source de fonctionnalités, le logiciel doit effectivement faire confiance à cette fonctionnalité. Sans mécanismes de protection suffisants, la fonctionnalité pourrait être de nature malveillante (provenant d'une source non fiable, usurpée ou modifiée en transit à partir d'une source fiable). La fonctionnalité peut également contenir ses propres faiblesses ou accorder l'accès à des fonctionnalités supplémentaires et à des informations d'état qui doivent rester privées pour le système de base, telles que des informations sur l'état du système, des données d'application sensibles ou le DOM d'une application Web.

II- Implémentation des scénarios d'attaques :

- **Scénario n°1 mise à jour sans signature :**

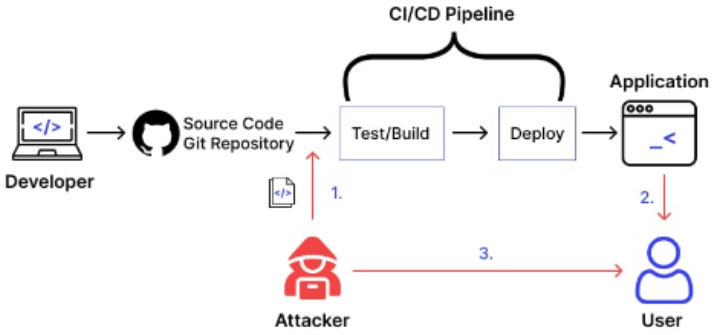
Les microprogrammes non signés constituent une cible de plus en plus importante pour les attaquants et leur nombre ne devrait cesser d'augmenter, En effet, plusieurs équipements ne vérifient pas les mises à jour via un firmware signé. Cependant il n'y a pas de mécanisme pour y remédier, si ce n'est de corriger dans une version future et d'attendre que les versions précédentes soient périmées.

- **Scénario n°2 mise à jour malveillante de SolarWinds :**

l'attaque SolarWinds Orion: la société a distribué une mise à jour malveillante très ciblée à plus de 18 000 organisations, dont une centaine ont été touchées suite à des altérations de processus. Il s'agit de l'une des violations de cette nature les plus étendues et les plus importantes de l'histoire

Scénario n°3 désérialisation non sécurisée

Une application React appelle un ensemble de microservices Spring Boot. Étant des programmeurs fonctionnels, ils ont essayé de s'assurer que leur code est immuable. La solution qu'ils ont trouvée consiste à sérialiser l'état de l'utilisateur et à le transmettre dans les deux sens à chaque requête. Un attaquant remarque la signature d'objet Java "rO0" (en base64) et utilise l'outil Java Serial Killer pour obtenir l'exécution de code à distance sur le serveur d'application.


CI/CD Pipeline
Application
Source Code
Git Repository
Test/Build
Deploy
Developer
1.
2.
3.
Attacker
User