TP 2

Exo 1 :   
Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Le contenu généré par l’IA peut être incorrect.

Questions :

• Combien de vulnérabilités sont détectées ?

* Total de 467 vulnérabilités ont été détectées, réparties comme suit :

UNKNOWN : 4

LOW : 137

MEDIUM : 186

HIGH : 112

CRITICAL : 28

• Quelle est la vulnérabilité la plus critique ?

* Parmi les vulnérabilités de niveau critique détectées :
* CVE-2022-1292 et CVE-2022-2068 dans libssl1.1 et openssl : Injection de commandes dans le script c\_rehash
* CVE-2022-37434 dans zlib1g : Débordement de tampon et lecture excessive dans la fonction inflate()
* CVE-2022-1586 et CVE-2022-1587 dans libpcre2-8-0 : Lecture hors limites dans les fonctions de compilation d'expressions régulières
* CVE-2022-29155 dans libldap-2.4-2 : Injection SQL dans OpenLDAP
* CVE-2019-8457 dans libdb5.3 : Lecture hors limites dans la fonction rtreenode()
* CVE-2024-37371 dans les bibliothèques krb5 : Problème de gestion de jetons de message GSS

• Quels packages sont affectés ?

* Les packages les plus critiques incluent :
* curl/libcurl4 : Multiples vulnérabilités critiques et de haute gravité
* openssl/libssl1.1 : Problèmes d'injection de commandes et diverses failles cryptographiques
* libtiff5 : Nombreuses vulnérabilités de dépassement de tampon et d'exécution de code
* zlib1g : Débordement de tampon dans la fonction inflate
* libexpat1 : Plusieurs problèmes de dépassement de tampon et d'utilisation après libération
* libpcre2-8-0 : Problèmes de lecture hors limites
* libxml2, libxslt1.1 : Problèmes de gestion de mémoire et déréférencement de pointeur NULL
* libfreetype6 : Problèmes de dépassement de tampon
* libc6/libc-bin : Vulnérabilités dans la bibliothèque C GNU
* libkrb5 : Plusieurs problèmes dans le système d'authentification Kerberos
* nginx : Vulnérabilités spécifiques au serveur web

Exo 2 :

Commande : trivy image --format json nginx:X > nginx.X.json

* Comparez les résultats
* jq '.Results[].Vulnerabilities[] | .VulnerabilityID' nginx.X.json > vulns .X.txt
* Générez des rapports JSON pour chaque image

Questions :

* Combien de vulnérabilités sont détectées ?
* 141 pour v.latest ; 467 pour v.1.20 et pour v.1.21 463

Une image contenant texte, capture d’écran, ligne, Police

Le contenu généré par l’IA peut être incorrect.

Une image contenant texte, capture d’écran, ligne, Police

Le contenu généré par l’IA peut être incorrect.

Pour V 1.20 cap du 1er exo

* Y a-t-il des vulnérabilités communes entre les versions ?
* Oui

Exo 3 :  
Une image contenant capture d’écran, texte, logiciel, Logiciel multimédia

Le contenu généré par l’IA peut être incorrect.

11 vulnérabilités ont été détectées dans le projet sur les packages :

* body-parser, cookie, express, lodash, path-to-regexp, qs, send, serve-static

Après : npm audit fix –force

Exo 4 :

1. trivy image --ignorefile .trivyignore nginx:1.20
2. trivy image --severity HIGH,CRITICAL nginx:1.20
3. trivy image --format json --output result.json nginx:1.20

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Le contenu généré par l’IA peut être incorrect.

Exo 5 :

Une image contenant texte, capture d’écran, Police

Le contenu généré par l’IA peut être incorrect.

Exo 6 :

trivy config ./docker-compose.yaml

Exo 7 :

Une image contenant texte, capture d’écran, logiciel, Police

Le contenu généré par l’IA peut être incorrect.

1. Mise en place d'un cron job
2. Métriques à surveiller
3. Nombre de vulnérabilités par niveau de sévérité :
   * Comptage des vulnérabilités CRITICAL, HIGH, MEDIUM, LOW
4. Taux de correction :
   * Évolution du nombre de vulnérabilités au fil du temps
5. Images affectées :
   * Pourcentage d'images avec des vulnérabilités critiques
6. Délai moyen de correction :
   * Temps entre la détection et la correction
7. Intégration avec d'autres systèmes
   1. Prometheus/Grafana :
      * Exposez les métriques au format Prometheus
      * Créez un dashboard Grafana pour visualiser l'évolution
   2. ElasticSearch/Kibana
      * Envoyez les logs vers ElasticSearch
      * Créez des dashboards dans Kibana
   3. Intégration CI/CD
      * Exécutez le scan automatiquement lors des builds
      * Bloquez les déploiements en cas de vulnérabilités critiques