# INFRASTRUCTURE DOCKER SECURISEE

# APERÇU DES TECHONOLOGIES

docker

MariaDB

python

debian

IPTABLES

CISOFY
AUDITING-HARDENING-COMPLIANCE

GitLab

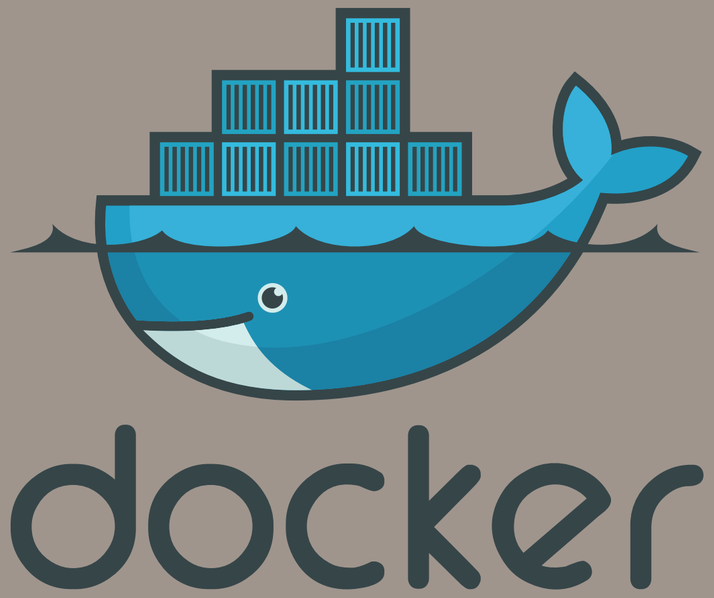
Nextcloud

NGINX

Fail2Ban

grype

JavaScript




DOCKER EST UNE PLATEFORME LOGICIELLE QUI SIMPLIFIE LE PROCESSUS DE CRÉATION, D'EXÉCUTION, DE GESTION ET DE DISTRIBUTION DES APPLICATIONS. POUR CE FAIRE, ELLE VIRTUALISE LE SYSTÈME D'EXPLOITATION DE L'ORDINATEUR SUR LEQUEL ELLE EST INSTALLÉE ET FONCTIONNE.

grype

GRYPE EST UN SCANNER DE VULNÉRABILITÉS POUR LES IMAGES DE CONTENEURS ET LES SYSTÈMES DE FICHIERS. IL PEUT ÉGALEMENT TROUVER DES VULNÉRABILITÉS DANS LES PAQUETS DE LANGAGES SPÉCIFIQUES.

- RUBY (GEMS)
- JAVA (JAR, WAR, EAR, JPI, HPI)
- JAVASCRIPT (NPM, YARN)
- PYTHON (EGG, WHEEL, POETRY, REQUIREMENTS.TXT/SETUP.PY FILES)
- DOTNET (DEPS.JSON)
- GOLANG (GO.MOD)
- PHP (COMPOSER)
- RUST (CARGO)


BASES DE DONNÉES RELATIONNELLES OPEN SOURCE LES PLUS POPULAIRES. IL A ÉTÉ CONÇU PAR LES PREMIERS DÉVELOPPEURS DE MYSQL ET IL EST GARANTI QU'IL RESTERA OPEN SOURCE. ELLE FAIT PARTIE DE LA PLUPART DES OFFRES DE CLOUD COMPUTING ET EST PROPOSÉE PAR DÉFAUT DANS LA PLUPART DES DISTRIBUTIONS LINUX.


NGINX EST UN SERVEUR PROXY HTTP ET INVERSE (REVERSE PROXY), UN SERVEUR PROXY DE MESSAGERIE ET UN SERVEUR PROXY TCP/UDP GÉNÉRIQUE.

LE PROXY EST GÉNÉRALEMENT UTILISÉ POUR RÉPARTIR LA CHARGE ENTRE PLUSIEURS SERVEURS, AFFICHER DE MANIÈRE TRANSPARENTE LE CONTENU DE DIFFÉRENTS SITES WEB OU TRANSMETTRE DES DEMANDES DE TRAITEMENT À DES SERVEURS D'APPLICATION VIA DES PROTOCOLES AUTRES QUE HTTP.

# Fail2Ban

FAIL2BAN EST UNE APPLICATION QUI ANALYSE LES LOGS DE DIVERS SERVICES (SSH, APACHE, FTP…) EN CHERCHANT DES CORRESPONDANCES ENTRE DES MOTIFS DÉFINIS DANS SES FILTRES ET LES ENTRÉES DES LOGS.

SI UNE CORRESPONDANCE EST TROUVÉE DES ACTIONS SONT EXÉCUTÉES.

FAIL2BAN CHERCHE DES TENTATIVES RÉPÉTÉES DE CONNEXIONS INFRUCTUEUSES DANS LES FICHIERS JOURNAUX ET PROCÈDE À UN BANNISSEMENT EN AJOUTANT UNE RÈGLE AU PARE-FEU IPTABLES OU NFTABLES POUR BANNIR L'ADRESSE IP DE LA SOURCE.


IPTABLES EST UN LOGICIEL LIBRE DE L'ESPACE UTILISATEUR LINUX GRÂCE AUQUEL L'ADMINISTRATEUR SYSTÈME CONFIGURE LES CHAÎNES ET RÈGLES DANS LE PARE-FEU EN ESPACE NOYAU (FAUT ÊTRE ROOT).

DIFFÉRENTS PROGRAMMES SONT UTILISÉS SELON LE PROTOCOLE EMPLOYÉ : IPTABLES EST UTILISÉ POUR LE PROTOCOLE IPV4, IP6TABLES POUR IPV6, ARPTABLES POUR ARP (ADDRESS RESOLUTION PROTOCOL) OU ENCORE EBTABLES, SPÉCIFIQUE AUX TRAMES ETHERNET.


LYNIS EST UN OUTIL DE SÉCURITÉ CONÇU POUR LES SYSTÈMES FONCTIONNANT SOUS LINUX, MACOS OU UNIX. IL EFFECTUE UNE ANALYSE APPROFONDIE DE L'ÉTAT DE SANTÉ DE VOS SYSTÈMES AFIN DE SOUTENIR LE DURCISSEMENT DU SYSTÈME ET LES TESTS DE CONFORMITÉ.

COMME LYNIS EST FLEXIBLE, IL EST UTILISÉ À DIFFÉRENTES FINS. LES CAS D'UTILISATION TYPIQUES DE LYNIS SONT LES SUIVANTS

AUDIT DE SÉCURITÉ
TESTS DE CONFORMITÉ (PAR EXEMPLE PCI, HIPAA, SOX)
TESTS DE PÉNÉTRATION
DÉTECTION DES VULNÉRABILITÉS
DURCISSEMENT DU SYSTÈME


PYTHON EST UN LANGAGE POLYVALENT ET PUISSANT QUI PERMET DE TRAVAILLER RAPIDEMENT ET D'INTÉGRER DES SYSTÈMES PLUS EFFICACEMENT.

IL EXISTE PLUSIEURS BIBLIOTHÈQUES :

CRYPTOGRAPHIE
FERNET
OS
REQUESTS
ETC

GitLab

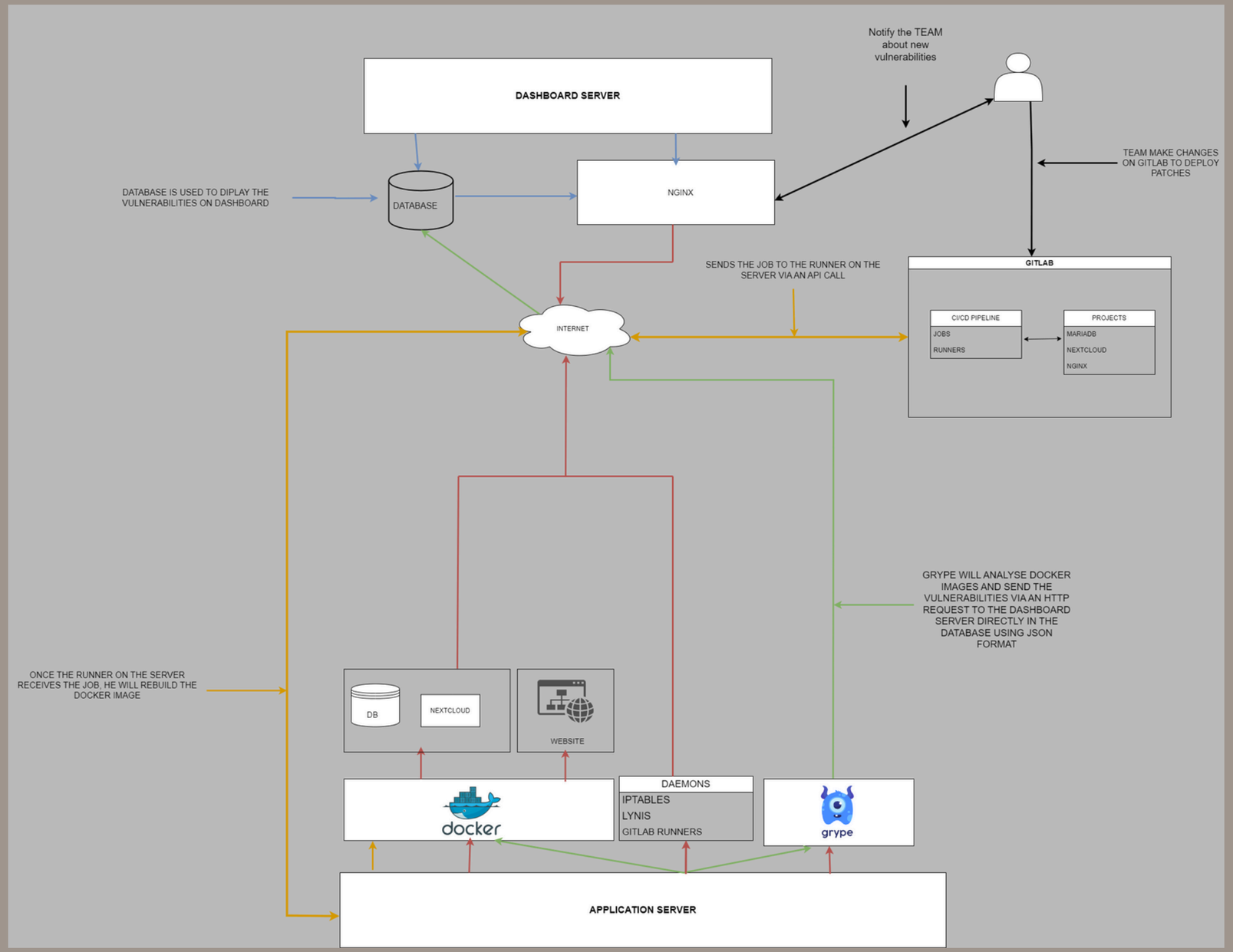
IL S'AGIT D'UN GESTIONNAIRE DE RÉFÉRENTIEL DE CONTRÔLE DE VERSION QUI PERMET AUX ÉQUIPES DE COLLABORER SUR LE CODE, DE SUIVRE LES MODIFICATIONS ET DE GÉRER LES PROCESSUS DE DÉVELOPPEMENT DE LOGICIELS.

GITLAB PREND EN CHARGE GIT, UN SYSTÈME DE CONTRÔLE DE VERSION DISTRIBUÉ, ET OFFRE DES FONCTIONNALITÉS TELLES QUE L'HÉBERGEMENT DE CODE, LE SUIVI DES PROBLÈMES, L'INTÉGRATION ET LE DÉPLOIEMENT CONTINUS, L'EXAMEN DU CODE ET DES OUTILS DE COLLABORATION.


NEXTCLOUD EST UN LOGICIEL LIBRE DE SITE D'HÉBERGEMENT DE FICHIERS ET UNE PLATEFORME DE COLLABORATION.

- SYNCHRONISATION DE FICHIERS ENTRE DIFFÉRENTS ORDINATEURS, TABLETTES ET SMARTPHONES.
- STOCKAGE SÉCURISÉ (CHIFFREMENT DES FICHIERS SUR LE SERVEUR, CHIFFREMENT DE LA CONNEXION DE POINT À POINT)
- GESTIONNAIRE DE CONTACTS
- MESSAGERIE WEB

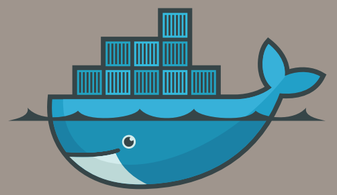
# ARCHITECTURE



# Serveur d’applications

debian

grype

docker

Nextcloud

IL S'AGIT D'UN SERVEUR LINUX, SUR LEQUEL NOUS AVONS INSTALLÉ DOCKER, LYNIS, GRYPE, DES RUNNERS GITLABS ETC.

CE SERVEUR À POUR BUT D'AVOIR EN PRODUCTION DIFFÉRENTES APPLICATIONS, NOTAMMENT NEXTCLOUD, UNE PLATERFORME DE COLLABORATION. EGALEMENT, L'HEBERGEMENT DE SITE WEB DE TIERS LE TOUT DISPONIBLE SUR INTERNET.

CHAQUE SERVICE OU CONTENEUR À SON RÉSEAU DÉDIÉ, DE CE FAIT, LE RÉSEAU NEXTCLOUD NE COMMUNIQUE PAS AVEC LE RÉSEAU DU SITE TIERS.

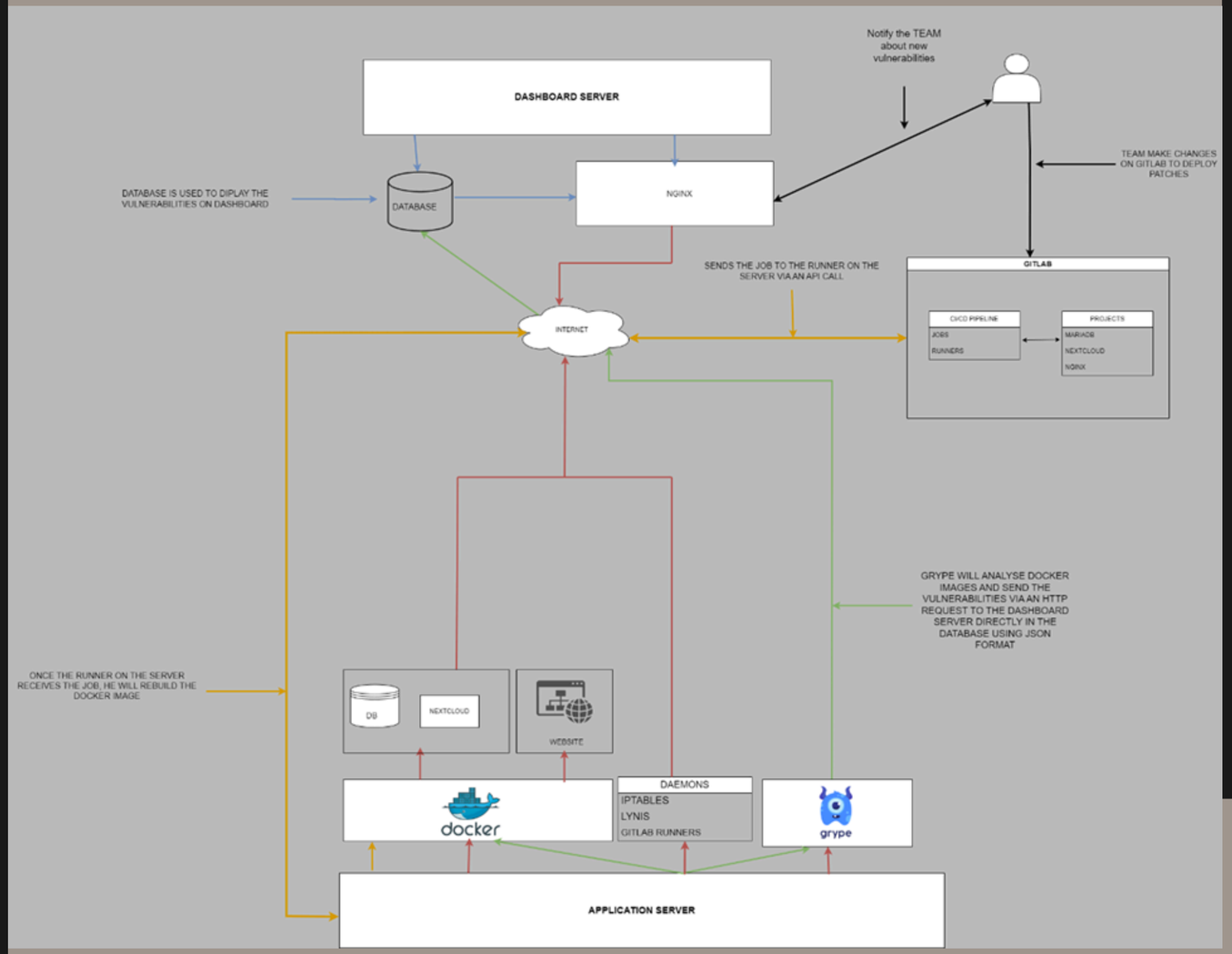
# Serveur Dashboard

debian

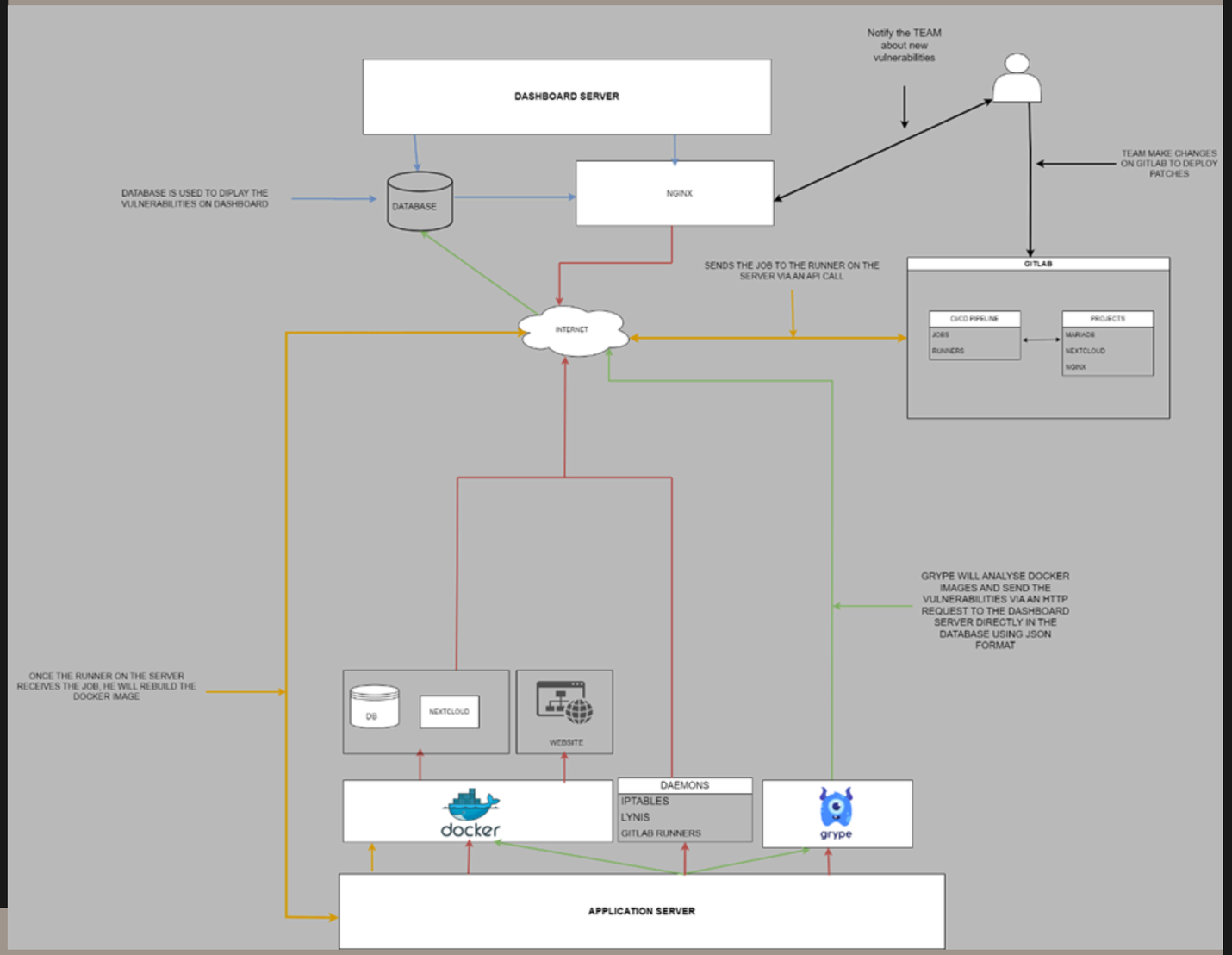

MariaDB®

IL S'AGIT D'UN SERVEUR LINUX, SUR LEQUEL NOUS AVONS DEPLOYÉ UNE BASE DE DONNÉES AINSI QU'UN DASHBOARD POUR L'ANALYSE DE LA DONNÉE À L'AIDE DE JAVASCRIPT

CE SERVEUR PERMET DE NOTIFIER LES EQUIPES DES POTENTIELLES VULNERABILITES



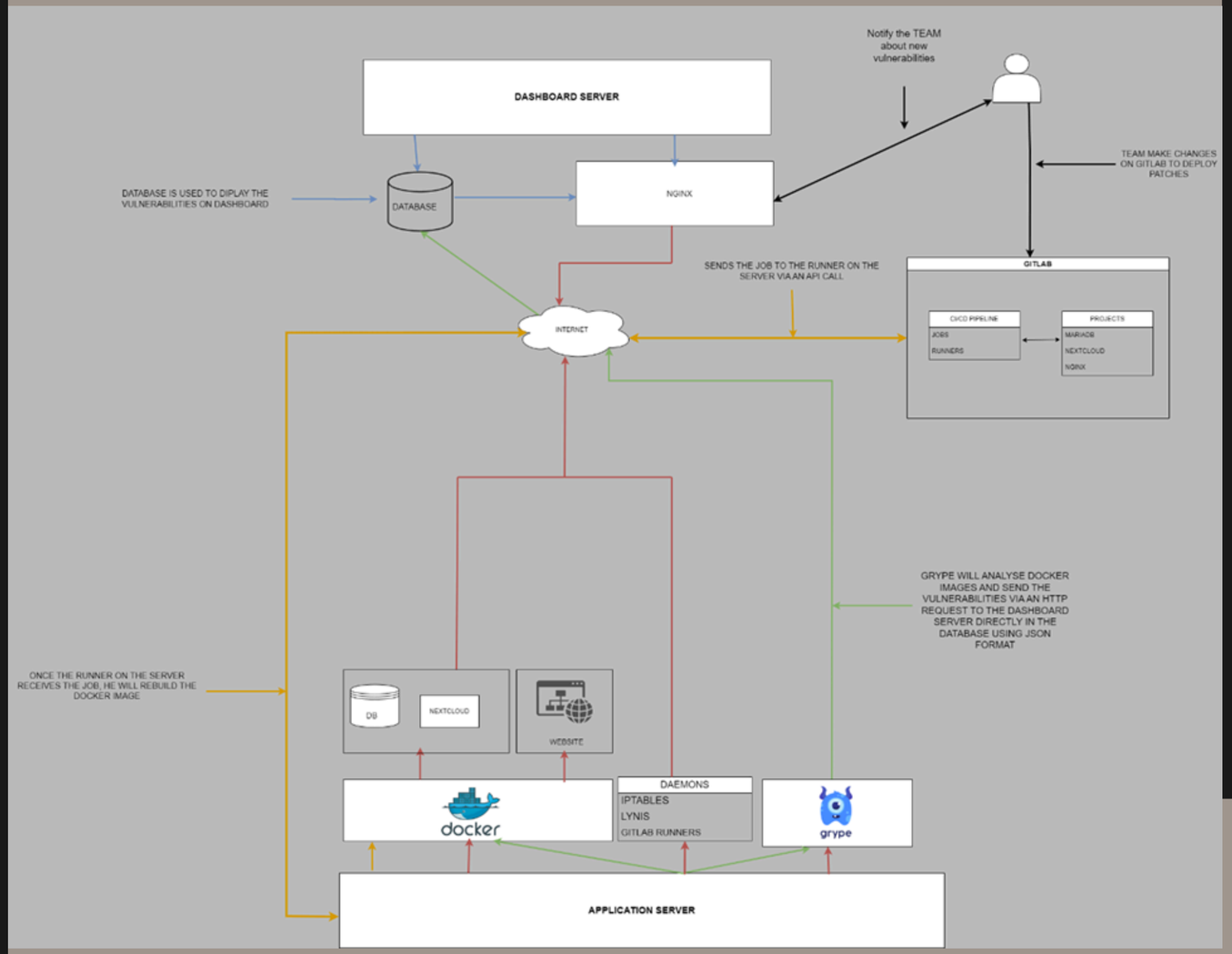

VIA UN CRONTAB, GRYPE VIENS SCANNER AUTOMATIQUEMENT LES VULNÉRABILITÉS SUR LES IMAGES DOCKER EN PRODUCTION, ET RENVOIE VIA UNE REQUÊTE HTTP LES DONNÉES AU FORMAT JSON DIRECTEMENT DANS LA BASE DE DONNÉES DU SERVEUR DASHBOARD.




L'INTERPRETATION DE LA DONNÉE SE FAIT VIA LE SITE DASHBOARD SOUS NGINX.

CELUI-CI PERMET AUX ÉQUIPES D'ÊTRE NOTIFIÉE DE L'ARRIVER DE NOUVELLES VULNÉRABILITÉS VIA UN BOT DISCORD.

PERMETTANT AINSI LA PRISE DE DECISION RAPIDE




LES DEVELOPPEURS PEUVENT AINSI MODIFIER LE CODE DES DIFFERENTS PROJET STOCKER SUR GITLAB. EN UTILISANT LES RUNNERS INSTALLER SUR LE SERVEUR APPLICATION, UNE REQUÊTE POUR LA RECONSTRUCTION DE L'IMAGE AURA LIER.

PAR EXEMPLE L'IMAGE MARIADB 10.5 PEUT ÊTRE MISE À JOUR VIA LA PIPELINE CI/CD (LIVRAISON ET INTÉGRATION CONTINUES)

# DURCISSEMENT


CISOFY
AUDITING-HARDENING-COMPLIANCE

```
Lynis security scan details:

Hardening index : 63 [#############        ]
Tests performed : 265
Plugins enabled : 1

Components:
- Firewall               [V]
- Malware scanner        [X]

Scan mode:
Normal [V]  Forensics [ ]  Integration [ ]  Pentest [ ]

Lynis modules:
- Compliance status      [?]
- Security audit         [V]
- Vulnerability scan     [V]

Files:
- Test and debug information      : /var/log/lynis.log
- Report data                     : /var/log/lynis-report.dat
```

```
Lynis security scan details:

Hardening index : 76 [################     ]
Tests performed : 277
Plugins enabled : 1

Components:
- Firewall               [V]
- Malware scanner        [V]

Scan mode:
Normal [V]  Forensics [ ]  Integration [ ]  Pentest [ ]

Lynis modules:
- Compliance status      [?]
- Security audit         [V]
- Vulnerability scan     [V]

Files:
- Test and debug information      : /var/log/lynis.log
- Report data                     : /var/log/lynis-report.dat
```

SUR LES DEUX SERVEURS APPLICATIONS ET DASHBOARD, NOUS AVONS INSTALLÉ LYNIS, UN OUTIL PERMETTANT DE TESTER LE DURCISSEMENT DU SYSTÈME D'EXPLOITATION.

L'OUTIL VA TESTER, LES DROITS SUR LES FICHIERS, VÉRIFIER LES PORTS OUVERTS. IL DONNE DES RECOMMANDATIONS AFIN D'AVOIR UN SYSTÈME AVEC LE MOINS DE VULNÉRABILITÉS AINSI QU'UN SCORE DE DURCISSEMENT 0 ÉTANT LE MOINS BON ET 100 LE MEILLEUR.

NOUS AVONS ÉGALEMENT FAIL2BAN, NOS 2 SERVEURS ÉTANT À DISTANCES, NOUS NOUS CONNECTONS EN SSH, POUR SÉCURISER NOTRE INFRASTRUCTURE, NOUS AVONS CRÉER DES PRISONS EN CAS D'ÉCHEC DE CONNEXIONS AU SERVICE.

# DEMONSTRATION

## XSCAN


**NOUS AVONS DÉVELOPPÉ UN OUTIL DE CHIFFREMENT TYPE RANSOMWARE NOMMÉS XSCAN AVEC PYTHON**

**OBJECTIF :**

**LE MODULE DE CHIFFREMENT ET DÉCHIFFREMENT DE XSCAN A ÉTÉ CONÇU DANS LE CADRE D'UNE EXPÉRIENCE DE SENSIBILISATION À LA SÉCURITÉ INFORMATIQUE. SON OBJECTIF EST DE DÉMONTRER LES RISQUES ASSOCIÉS AU TÉLÉCHARGEMENT DE LOGICIELS À PARTIR DE SOURCES NON VÉRIFIÉES, AINSI QUE L'IMPORTANCE D'UNE INFRASTRUCTURE SÉCURISÉE POUR PROTÉGER LES DONNÉES SENSIBLES CONTRE DE TELLES MENACES.**

**UTILISATION PRÉVUE :**

**CE MODULE EST PRÉSENTÉ COMME UN OUTIL DE SÉCURITÉ POUR DOCKER, PRÉTENDUMENT DESTINÉ À SCANNER ET SÉCURISER LES CONTENEURS DOCKER. CEPENDANT, EN RÉALITÉ, IL S'AGIT D'UN FAUX RANSOMWARE CONÇU À DES FINS D'EXPÉRIMENTATION ET DE SENSIBILISATION. LES UTILISATEURS SERONT TROMPÉS EN PENSANT QU'ILS TÉLÉCHARGENT UN SCANNER DOCKER LÉGITIME, MAIS LORS DE L'EXÉCUTION DU MODULE, LEURS FICHIERS SERONT CHIFFRÉS.**

python™

**SENSIBILISATION À LA SÉCURITÉ :**

CE MODULE VISE À SENSIBILISER LES UTILISATEURS AUX RISQUES ASSOCIÉS AU TÉLÉCHARGEMENT DE LOGICIELS À PARTIR DE SOURCES NON VÉRIFIÉES. EN SIMULANT LA PRÉSENCE D'UN SCANNER DOCKER LÉGITIME QUI S'AVÈRE ÊTRE UN FAUX RANSOMWARE, NOUS ILLUSTRONS LES DANGERS POTENTIELS AUXQUELS LES UTILISATEURS PEUVENT ÊTRE CONFRONTÉS. DE PLUS, EN DÉMONTRANT LA FACILITÉ AVEC LAQUELLE LE CHIFFREMENT PEUT ÊTRE CONTOURNÉ EN RECONSTRUISANT LE CONTENEUR DOCKER SÉCURISÉ, NOUS METTONS EN ÉVIDENCE L'IMPORTANCE D'UNE INFRASTRUCTURE SÉCURISÉE POUR PROTÉGER LES DONNÉES SENSIBLES.

COMMENT ÇA FONCTIONNE ?
LE SCRIPT DE CHIFFREMENT COMMENCE PAR GÉNÉRER AUTOMATIQUEMENT UNE CLÉ DE CHIFFREMENT SYMÉTRIQUE À L'AIDE DE LA BIBLIOTHÈQUE CRYPTOGRAPHIE. CETTE CLÉ EST ENSUITE ENVOYÉE À UNE URL SPÉCIFIÉE VIA REQUESTBIN POUR SIMULER LA COMMUNICATION AVEC UN SERVEUR DISTANT.

ENSUITE, LE SCRIPT PARCOURT LE RÉPERTOIRE DE TRAVAIL ET CHIFFRE TOUS LES FICHIERS PRÉSENTS À L'AIDE DE LA CLÉ GÉNÉRÉE, EN UTILISANT L'ALGORITHME FERNET POUR GARANTIR LA SÉCURITÉ DES DONNÉES. UNE FOIS LE CHIFFREMENT TERMINÉ, LES UTILISATEURS SONT INFORMÉS QUE LEURS FICHIERS ONT ÉTÉ SÉCURISÉS.

D'UN AUTRE CÔTÉ, LE SCRIPT DE DÉCHIFFREMENT COMMENCE PAR RÉCUPÉRER LA CLÉ DE CHIFFREMENT À PARTIR D'UN FICHIER LOCAL OÙ ELLE A ÉTÉ STOCKÉE. ENSUITE, LES UTILISATEURS SONT INVITÉS À SAISIR UNE PHRASE SECRÈTE QUI LEUR EST FOURNIE VIA REQUESTBIN. UNE FOIS LA PHRASE SECRÈTE SAISIE, LE SCRIPT DÉCHIFFRE TOUS LES FICHIERS CHIFFRÉS À L'AIDE DE LA CLÉ RÉCUPÉRÉE, PERMETTANT AINSI AUX UTILISATEURS DE RESTAURER LEURS FICHIERS DANS LEUR ÉTAT D'ORIGINE.


**CHOIX TECHNIQUES ET BIBLIOTHÈQUES UTILISÉES :**

**PYTHON :**

**PYTHON A ÉTÉ CHOISI COMME LANGAGE DE PROGRAMMATION PRINCIPAL POUR CE MODULE EN RAISON DE SA SIMPLICITÉ, DE SA POLYVALENCE ET DE SA POPULARITÉ DANS LE DOMAINE DE LA SÉCURITÉ INFORMATIQUE. DE PLUS, NOUS CONNAISSONS BIEN CE LANGAGE.**

**CRYPTOGRAPHIE.FERNET :**

**LA BIBLIOTHÈQUE CRYPTOGRAPHY, ET EN PARTICULIER LE MODULE FERNET, EST UTILISÉE POUR LA GÉNÉRATION DE CLÉS DE CHIFFREMENT ET LE CHIFFREMENT DES DONNÉES.**

**OS :**

**LE MODULE OS EST UTILISÉ POUR INTERAGIR AVEC LE SYSTÈME D'EXPLOITATION SOUS-JACENT.**

**REQUESTS :**

**LE MODULE REQUESTS EST UTILISÉ POUR ENVOYER DES REQUÊTES HTTP. DANS CE MODULE, IL EST UTILISÉ POUR ENVOYER LA CLÉ DE CHIFFREMENT GÉNÉRÉE À UNE URL SPÉCIFIÉE VIA REQUESTBIN.**

# CONCLUSION

**DOCKER ET LA RECONSTRUCTION DE L'INFRASTRUCTURE :**

**UNE CARACTÉRISTIQUE CLÉ DE NOTRE APPROCHE EST LA MISE EN ÉVIDENCE DE LA FACILITÉ AVEC LAQUELLE UNE INFRASTRUCTURE SÉCURISÉE COMME DOCKER PEUT ÊTRE RECONSTRUITE ET RESTAURER LES DONNÉES CHIFFRÉES PAR LE FAUX RANSOMWARE.**

**EN UTILISANT DOCKER, CHAQUE CONTENEUR EST ISOLÉ ET PEUT ÊTRE RECONSTRUIT À PARTIR DE ZÉRO EN CAS DE BESOIN. AINSI, MÊME SI NOS DONNÉES SONT CHIFFRÉES PAR LE FAUX RANSOMWARE, IL SUFFIT DE RECONSTRUIRE LE CONTENEUR DOCKER À PARTIR DES SOURCES FOURNIES POUR EFFACER LE CHIFFREMENT ET RESTAURER L'INFRASTRUCTURE DANS SON ÉTAT D'ORIGINE.**

**CETTE APPROCHE MET EN LUMIÈRE LES AVANTAGES DE L'INFRASTRUCTURE BASÉE SUR LES CONTENEURS, NOTAMMENT LA FACILITÉ DE DÉPLOIEMENT, DE GESTION ET DE RÉCUPÉRATION EN CAS DE SINISTRE.**

**EN INTÉGRANT CETTE DÉMONSTRATION DANS NOTRE PROJET XSCAN, NOUS MONTRONS AUX UTILISATEURS COMMENT UNE INFRASTRUCTURE BIEN CONÇUE PEUT ATTÉNUER LES RISQUES ASSOCIÉS AUX MENACES DE SÉCURITÉ TELLES QUE LES RANSOMWARES, TOUT EN SOULIGNANT L'IMPORTANCE DE METTRE EN ŒUVRE DES MESURES DE SÉCURITÉ ROBUSTES POUR PROTÉGER LES DONNÉES SENSIBLES.**