TUTOS-INFO
L'informatique par l'exemple

TUTORIEL

AUDITER DEBIAN 12 avec © Lynis

SOMMAIRE

EXPERT
DIFFICULTE

1 – LYNIS C'EST QUOI ?

Lynis est un outil de sécurité éprouvé pour les systèmes exécutant Linux, macOS ou un système d'exploitation basé sur Unix. Il effectue une analyse approfondie de l'état de santé de vos systèmes pour prendre en charge le renforcement du système et les tests de conformité. Le projet est un logiciel open source sous licence GPL et disponible depuis 2007.

Lynis effectue des audits approfondis, examinant chaque aspect du système pour y déceler des faiblesses. Il vérifie les configurations, les permissions de fichiers, les paramètres de sécurité du réseau et bien d'autres éléments cruciaux. Cette analyse complète permet de **déceler les failles** qui pourraient être exploitées par des attaquants.

Contrairement à d'autres outils d'audit, **Lynis** ne se contente pas de signaler les problèmes : il fournit également des **recommandations concrètes** pour les résoudre. Chaque avertissement est accompagné de conseils pratiques et de directives pour améliorer la sécurité du système. Cette approche pédagogique est particulièrement bénéfique pour les utilisateurs moins expérimentés.

Étant donné que Lynis est flexible, il est utilisé à plusieurs fins différentes. Les cas d'utilisation typiques de Lynis sont les suivants :

- Audit de sécurité
- Tests de conformité (p. ex. PCI, HIPAA, SOx)
- Tests d'intrusion
- Détection des vulnérabilités
- Durcissement du système

Voici ce qui se passe lors d'un scan typique avec Lynis :

1. Initialisation
2. Effectuer des vérifications de base, telles que la propriété des fichiers
3. Déterminer le système d'exploitation et les outils
4. Rechercher les composants logiciels disponibles
5. Vérifier la dernière version de Lynis
6. Exécuter des plug-ins activés
7. Exécuter des tests de sécurité par catégorie
8. Effectuer l'exécution de vos tests personnalisés (facultatif)
9. Signaler l'état de l'analyse de sécurité

Outre le rapport et les informations affichés à l'écran, tous les détails techniques de l'analyse sont stockés dans un fichier journal (lynis.log). Les résultats, tels que les avertissements et les suggestions, sont stockés dans un fichier de rapport séparé (lynis-report.dat).

Lynis effectue des centaines de tests individuels. Chaque test aidera à déterminer l'état de sécurité du système. La plupart des tests sont écrits en script shell et ont un identifiant unique (par exemple KRNL-6000).

2 – INSTALLER LYNIS SUR DEBIAN 12

1ère étape : installation de Lynis

L'installation de Lynis peut être réalisée soit :

- Via un gestionnaire de paquet (par exemple avec « apt » sur Debian)
- Via Git, en clonant le projet
- Par téléchargement direct (depuis le site officiel)

Dans ce tutoriel, nous allons utiliser Git pour l'installation (cette solution permet de disposer de la dernière version de lynis à ce jour, c'est-à-dire la **version 3.1.2 – avril 2024**).

- Connectez-vous à votre machine Debian 12 (avec un utilisateur « sudoer » ou en « root »)

- Créez un répertoire Lynis sur votre machine (emplacement de votre choix, Git créera un sous-répertoire 'lynis' contenant le programme complet) : **« mkdir lynis »**

- Depuis votre répertoire Lynis, saisissez la commande suivante et patientez :

git clone https://github.com/CISOfy/lynis

```
gilles@debian-gilles:~/lynis$ git clone https://github.com/CISOfy/lynis
Clonage dans 'lynis'...
remote: Enumerating objects: 15107, done.
remote: Counting objects: 100% (513/513), done.
remote: Compressing objects: 100% (205/205), done.
remote: Total 15107 (delta 345), reused 431 (delta 306), pack-reused 14594
Réception d'objets: 100% (15107/15107), 8.05 Mio | 27.19 Mio/s, fait.
Résolution des deltas: 100% (11111/11111), fait.
```

Une fois le dépôt cloné, il est nécessaire de lancer Lynis. Pour cela, effectuez les manipulations suivantes :

- Placez-vous dans le dossier Lynis contenant le dépôt cloné
- Ouvrez le sous-répertoire « lynis » créé avec la commande **« cd lynis »**
- Vous pouvez vérifier la présence des fichiers avec la commande **« ls »**

Les répertoires et fichiers principaux de Lynis apparaissent :

```
gilles@debian-gilles:~/lynis/lynis$ ls
CHANGELOG.md        CONTRIBUTORS.md  developer.prf  HAPPY_USERS.md  LICENSE  plugins     SECURITY.md
CODE_OF_CONDUCT.md  db               extras         include         lynis    README      TODO.md
CONTRIBUTING.md     default.prf      FAQ            INSTALL         lynis.8  README.md
```

2ème étape : contrôle de la version installée

Nous allons vérifier, ici, la version Lynis installée en effectuant les commandes suivantes :

- Placez-vous dans le sous-dossier Lynis
- Saisissez la commande suivante :

./lynis show version

La version de Lynis s'affiche (3.1.2 – avril 2024) :

```
gilles@debian-gilles:~/lynis/lynis$ ./lynis show version
3.1.2
```

3ème étape : afficher les commandes Lynis utiles

- Depuis le sous-répertoire Lynis, saisissez la commande suivante :

./lynis

```
Usage: lynis command [options]


Command:

  audit
      audit system                    : Perform local security scan
      audit system remote <host>      : Remote security scan
      audit dockerfile <file>         : Analyze Dockerfile

  show
      show                            : Show all commands
      show version                    : Show Lynis version
      show help                       : Show help

  update
      update info                     : Show update details
```

```
Options:

  Alternative system audit modes
  --forensics                       : Perform forensics on a running or mounted system
  --pentest                         : Non-privileged, show points of interest for pentesting

  Layout options
  --no-colors                       : Don't use colors in output
  --quiet (-q)                      : No output
  --reverse-colors                  : Optimize color display for light backgrounds
  --reverse-colours                 : Optimize colour display for light backgrounds

  Misc options
  --debug                           : Debug logging to screen
  --no-log                          : Don't create a log file
  --profile <profile>               : Scan the system with the given profile file
  --view-manpage (--man)            : View man page
  --verbose                         : Show more details on screen
  --version (-V)                    : Display version number and quit
  --wait                            : Wait between a set of tests
  --slow-warning <seconds>  : Threshold for slow test warning in seconds (default 10)
```

4ème étape : vérification des mises à jour Lynis

La commande ci-dessous permet de vérifier que nous possédons la dernière version en date de Lynis :

./lynis update info

```
gilles@debian-gilles:~/lynis/lynis$ gilles@debian-gilles:~/lynis/lynis$ ./lynis update info

 == Lynis ==

  Version            : 3.1.2
  Status             : Up-to-date
  Release date       : 2024-03-18
  Project page       : https://cisofy.com/lynis/
  Source code        : https://github.com/CISOfy/lynis
  Latest package     : https://packages.cisofy.com/
```

3 – LANCEMENT UN 1er AUDIT DE VOTRE SYSTEME

Nous lançons le premier audit du système, depuis le sous-répertoire Lynis. Attention, si la commande est exécutée par un utilisateur « sudoer », certains tests ne pourront être effectués. L'audit est lancé avec la commande suivante :

./lynis audit system

Une fois l'audit terminé, un descriptif complet est affiché et on trouve ceci à la fin du rapport :

```
Lynis security scan details:

Hardening index : 68 [##############        ]
Tests performed : 240
Plugins enabled : 2

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

Scan mode:
Normal [ ]  Forensics [ ]  Integration [ ]  Pentest [V] (running non-privileged)

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

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

Il convient d'analyser chaque partie du rapport affiché afin d'y apporter des améliorations. L'intérêt de Lynis réside dans le fait que des suggestions d'améliorations sont proposées en fin de rapport (à lire impérativement !).

Exemple de rapport détaillé :

La rubrique « **Démarrage et services** » indique que la protection des services via « *systemd* » pourrait être améliorée. Grub2 est présent et il n'y a pas de problème avec les permissions sur les fichiers :

```
[+] Démarrage et services
------------------------------------
  - Service Manager                                           [ systemd ]
  - Checking presence GRUB2                                   [ TROUVÉ ]
    - Checking for password protection                        [ AUCUN ]
  - Check running services (systemctl)                        [ FAIT ]
        Result: found 13 running services
  - Check enabled services at boot (systemctl)                [ FAIT ]
        Result: found 13 enabled services
  - Check startup files (permissions)                         [ OK ]
  - Running 'systemd-analyze security'
        - chrony.service:                                     [ PROTÉGÉ ]
        - containerd.service:                                 [ RISQUÉ ]
        - cron.service:                                       [ RISQUÉ ]
        - dbus.service:                                       [ RISQUÉ ]
        - docker.service:                                     [ RISQUÉ ]
        - emergency.service:                                  [ RISQUÉ ]
        - fail2ban.service:                                   [ RISQUÉ ]
        - getty@tty1.service:                                 [ RISQUÉ ]
        - ifup@ens18.service:                                 [ RISQUÉ ]
        - rc-local.service:                                   [ RISQUÉ ]
        - rescue.service:                                     [ RISQUÉ ]
        - ssh.service:                                        [ RISQUÉ ]
        - systemd-ask-password-console.service:               [ RISQUÉ ]
        - systemd-ask-password-wall.service:                  [ RISQUÉ ]
        - systemd-fsckd.service:                              [ RISQUÉ ]
        - systemd-initctl.service:                            [ RISQUÉ ]
```

La partie « Systèmes de fichier » nous informe que des suggestions d'améliorations sont disponibles et conseillées (voir à la fin du rapport) :

```
[+] Systèmes de fichier
------------------------------------
  - Checking mount points
    - Checking /home mount point                              [ SUGGESTION ]
    - Checking /tmp mount point                               [ SUGGESTION ]
    - Checking /var mount point                               [ SUGGESTION ]
  - Query swap partitions (fstab)                             [ OK ]
  - Testing swap partitions                                   [ OK ]
  - Checking for old files in /tmp                            [ OK ]
  - Checking /tmp sticky bit                                  [ OK ]
  - Checking /var/tmp sticky bit                              [ OK ]
  - Mount options of /                                        [ PAS PAR DÉFAUT ]
  - Mount options of /dev                                     [ PARTIELLEMENT RENFORCÉ ]
  - Mount options of /dev/shm                                 [ PARTIELLEMENT RENFORCÉ ]
  - Mount options of /run                                     [ RENFORCÉ ]
  - Total without nodev:9 noexec:11 nosuid:7 ro or noexec (W^X): 11 of total 30
  - Disable kernel support of some filesystems
```

Fail2ban a été détecté sur notre système dans la rubrique « Logiciel : System tooling » :

```
[+] Logiciel : System tooling
------------------------------------
  - Checking automation tooling
  - Automation tooling                                    [ NON TROUVÉ ]
  - Checking presence of Fail2ban                         [ TROUVÉ ]
    - Checking Fail2ban jails                             [ ACTIVÉ ]
  - Checking for IDS/IPS tooling                          [ TROUVÉ ]
```

La rubrique « Prise en charge SSH » est souvent sujette à améliorations (nous en parlerons plus loin) :

```
[+] Prise en charge SSH
------------------------------------
  - Checking running SSH daemon                           [ TROUVÉ ]
    - Searching SSH configuration                         [ TROUVÉ ]
    - OpenSSH option: AllowTcpForwarding                  [ SUGGESTION ]
    - OpenSSH option: ClientAliveCountMax                 [ SUGGESTION ]
    - OpenSSH option: ClientAliveInterval                 [ OK ]
    - OpenSSH option: FingerprintHash                     [ OK ]
    - OpenSSH option: GatewayPorts                        [ OK ]
    - OpenSSH option: IgnoreRhosts                        [ OK ]
    - OpenSSH option: LoginGraceTime                      [ OK ]
    - OpenSSH option: LogLevel                            [ SUGGESTION ]
    - OpenSSH option: MaxAuthTries                        [ SUGGESTION ]
    - OpenSSH option: MaxSessions                         [ SUGGESTION ]
    - OpenSSH option: PermitRootLogin                     [ OK ]
    - OpenSSH option: PermitUserEnvironment               [ OK ]
    - OpenSSH option: PermitTunnel                        [ OK ]
    - OpenSSH option: Port                                [ SUGGESTION ]
    - OpenSSH option: PrintLastLog                        [ OK ]
    - OpenSSH option: StrictModes                         [ OK ]
    - OpenSSH option: TCPKeepAlive                        [ SUGGESTION ]
    - OpenSSH option: UseDNS                              [ OK ]
    - OpenSSH option: X11Forwarding                       [ SUGGESTION ]
    - OpenSSH option: AllowAgentForwarding                [ SUGGESTION ]
    - OpenSSH option: AllowUsers                          [ NON TROUVÉ ]
    - OpenSSH option: AllowGroups                         [ NON TROUVÉ ]
```

Le score de l'audit **Lynis** est une mesure quantitative qui reflète l'état de sécurité de votre système. À la fin de chaque audit, **Lynis** attribue un score, exprimé en pourcentage, qui évalue la robustesse de votre configuration de sécurité.

Ce score est calculé en fonction de divers facteurs, tels que les vulnérabilités détectées, les configurations de sécurité non optimales et les bonnes pratiques en matière de sécurité qui sont déjà en place.

```
Lynis security scan details:

Hardening index : 68 [#############       ]
Tests performed : 240
Plugins enabled : 2

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

Scan mode:
Normal [ ]  Forensics [ ]  Integration [ ]  Pentest [V] (running non-privileged)

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

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

Un score élevé indique que votre système est bien configuré et qu'il respecte de nombreuses bonnes pratiques de sécurité. Cela signifie que **Lynis** a trouvé moins de problèmes et de vulnérabilités et que les mesures de sécurité importantes sont déjà en place. Inversement, un score bas suggère que des améliorations significatives sont nécessaires pour renforcer la sécurité de votre système. Il indique généralement la présence de nombreuses vulnérabilités ou de configurations de sécurité inadéquates.

Il est important d'utiliser le score de l'audit comme un indicateur de progression dans vos efforts de sécurisation. Après avoir apporté des changements recommandés par **Lynis**, vous devriez ré-exécuter l'outil pour voir si vos actions ont conduit à une amélioration du score. L'augmentation du score d'un audit à l'autre est un signe positif que vous avez réussi à améliorer la sécurité de votre système.

4 – AMELIORATION DE LA CONFIGURATION DU SERVICE SSHD

Comme souvent, le service « SSHD » nécessite des améliorations afin de protéger efficacement le système. Editez le fichier « sshd_config » situé dans « /etc/ssh » afin de l'améliorer :

Les suggestions d'améliorations proposées sont les suivantes :

```
* Consider hardening SSH configuration [SSH-7408]
  - Details  : AllowTcpForwarding (set YES to NO)
    https://cisofy.com/lynis/controls/SSH-7408/

* Consider hardening SSH configuration [SSH-7408]
  - Details  : ClientAliveCountMax (set 3 to 2)
    https://cisofy.com/lynis/controls/SSH-7408/

* Consider hardening SSH configuration [SSH-7408]
  - Details  : LogLevel (set INFO to VERBOSE)
    https://cisofy.com/lynis/controls/SSH-7408/

* Consider hardening SSH configuration [SSH-7408]
  - Details  : MaxAuthTries (set 6 to 3)
    https://cisofy.com/lynis/controls/SSH-7408/

* Consider hardening SSH configuration [SSH-7408]
  - Details  : MaxSessions (set 10 to 2)
    https://cisofy.com/lynis/controls/SSH-7408/

* Consider hardening SSH configuration [SSH-7408]
  - Details  : Port (set 22 to )
    https://cisofy.com/lynis/controls/SSH-7408/

* Consider hardening SSH configuration [SSH-7408]
  - Details  : TCPKeepAlive (set YES to NO)
    https://cisofy.com/lynis/controls/SSH-7408/
```

```
* Consider hardening SSH configuration [SSH-7408]
  - Details  : X11Forwarding (set YES to NO)
    https://cisofy.com/lynis/controls/SSH-7408/

* Consider hardening SSH configuration [SSH-7408]
  - Details  : AllowAgentForwarding (set YES to NO)
    https://cisofy.com/lynis/controls/SSH-7408/
```

On commence par éditer le fichier « sshd_config » :

nano /etc/ssh/sshd_config

On applique les suggestions recommandées par Lynis dans le fichier « sshd_config ». Une fois les modifications effectuées, on quitte et on enregistre le fichier « sshd_config » modifié et on redémarre le service avec la commande « **systemctl restart ssh** ».

On relance l'audit afin de voir si les modifications ont apporté des améliorations :

./lynis audit system

Le score final s'est nettement amélioré, ce qui prouve l'importance d'une bonne configuration du service SSHD !

```
Lynis security scan details:

Hardening index : 73 [###############      ]
```

La rubrique « Prise en charge SSH » est maintenant parfaitement sécurisée :

```
[+] Prise en charge SSH
------------------------------------
  - Checking running SSH daemon                           [ TROUV
    - Searching SSH configuration                         [ TROUV
    - OpenSSH option: AllowTcpForwarding                  [ OK ]
    - OpenSSH option: ClientAliveCountMax                 [ OK ]
    - OpenSSH option: ClientAliveInterval                 [ OK ]
    - OpenSSH option: FingerprintHash                     [ OK ]
    - OpenSSH option: GatewayPorts                        [ OK ]
    - OpenSSH option: IgnoreRhosts                        [ OK ]
    - OpenSSH option: LoginGraceTime                      [ OK ]
    - OpenSSH option: LogLevel                            [ OK ]
    - OpenSSH option: MaxAuthTries                        [ OK ]
    - OpenSSH option: MaxSessions                         [ OK ]
    - OpenSSH option: PermitRootLogin                     [ OK ]
    - OpenSSH option: PermitUserEnvironment               [ OK ]
    - OpenSSH option: PermitTunnel                        [ OK ]
    - OpenSSH option: Port                                [ OK ]
    - OpenSSH option: PrintLastLog                        [ OK ]
    - OpenSSH option: StrictModes                         [ OK ]
    - OpenSSH option: TCPKeepAlive                        [ OK ]
    - OpenSSH option: UseDNS                              [ OK ]
    - OpenSSH option: X11Forwarding                       [ OK ]
    - OpenSSH option: AllowAgentForwarding                [ OK ]
```

Les fichiers « log » de Lynis sont stockés dans le sous-répertoire « Lynis » ainsi que le rapport d'audit :

```
gilles@debian-gilles:~$ ls
lynis  lynis.log  lynis-report.dat
```

5 – AUTOMATISER L'AUDIT DU SYSTEME AVEC « CRON »

CRON est un planificateur de tâches sous Unix qui peut être utilisé pour automatiser l'exécution de **Lynis**. Pour planifier un audit régulier, nous allons créer une tâche « cron » qui lancera **Lynis** à une fréquence spécifique (quotidienne, hebdomadaire, mensuelle, etc.).

La crontab (table cron) est utilisée pour **automatiser tous les types de tâches sur les systèmes Linux**.

Crontab est un processus démon qui tourne en arrière-plan sur presque toutes les machines Linux et qui est utilisé pour planifier et exécuter des tâches automatisées à des intervalles de temps déterminés. Il est également souvent utilisé par les développeurs de logiciels pour exécuter des tâches fastidieuses en arrière-plan.

Attention, il convient de distinguer les 3 éléments « **daemon** », « **crontab** » et « **cronjob** » :

Elément	Nom	Signification
Daemon	crond	Prononcé "demon" ou "day-mon". Il s'agit de processus d'arrière-plan du système Linux.
Table	crontab	Vous écrivez des lignes dans ce tableau lorsque vous entrez une commande crontab. Chaque astérisque "*" représente un segment de temps et une colonne correspondante dans chaque ligne.
Job	Cron Job	La tâche spécifique à effectuer est décrite dans une ligne, associée à l'heure à laquelle elle doit être effectuée.

Chaque tâche « **crontab** » se compose d'une ligne et est formatée comme suit (exemple) :

On planifie, par exemple, un audit Lynis **chaque vendredi à 6 h** ; cela donne la tâche suivante :

0 6 * * 6 /home/gilles/lynis/lynis ./audit system

SYNTAXE A RESPECTER :

```
Example of job definition:
.---------------- minute (0 - 59)
|  .------------- hour (0 - 23)
|  |  .---------- day of month (1 - 31)
|  |  |  .------- month (1 - 12) OR jan,feb,mar,apr ...
|  |  |  |  .---- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
|  |  |  |  |
*  *  *  *  *  user command to be executed
```

PLANIFICATION D'UNE TACHE PAR UTILISATEUR :

Chaque utilisateur peut avoir son « cron » ; pour cela, il vous suffit de vous loguer avec votre utilisateur et de saisir la commande suivante pour modifier l'entrée « crontab » de l'utilisateur connecté :

crontab -e

Comme nous n'avons pas encore créé de « crontab », le système demande quel éditeur nous souhaitons utiliser. Nano est suggéré comme le programme le plus facile à utiliser : **choix « 1 »**.

Saisissez « **1** » pour valider le choix de l'éditeur « nano » et pressez la touche « **Entrée** ».

On saisit la commande « **crontab -e** » et on valide le choix « **1** » :

```
gilles@debian-gilles:~$ crontab -e
no crontab for gilles - using an empty one

Select an editor.  To change later, run 'select-editor'.
  1. /bin/nano        <---- easiest
  2. /usr/bin/vim.tiny

Choose 1-2 [1]: 1
```

On ajoute la planification de la tâche en fin de fichier et on sauvegarde le fichier :

```
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command
0 6 * * 6 /home/gilles/lynis/lynis ./audit system
```

Désormais, l'audit système Lynis sera effectué chaque vendredi à 6 h (pour l'utilisateur « gilles » ici).

COMMANDES CRON A CONNAITRE

crontab -l (permet de lister les tâches planifiées de l'utilisateur connecté)

crontab -r (permet de supprimer toutes les tâches de la crontab de l'utilisateur connecté)

crontab -u nom_user -l (permet au « root » de voir les tâches planifiées de l'utilisateur nommé)

crontab -u nom_user -r (permet au « root » de supprimer la crontab de l'utilisateur nommé)

Il est possible d'aller plus loin avec « cron » en paramétrant des tâches planifiées faisant appel à des scripts par exemple.