Qué podés auditar hoy en tu sistema GNU/Linux

con software libre



flisol
CABA 2026

Sultanovich

$ whoami

```
@sultanovich  # Pablo
```

- Sysadmin renombrado a SecOps/SRE
- Uso GNU/Linux desde hace años (Mandrake 7 y Debian 3)
- Participo en Debian Argentina y FLISoL CABA
- Me interesan el software libre y la seguridad informatica
- Actualmente trabajo en Wazuh (plataforma open source de seguridad)

Antes de empezar

Esta charla `NO es`

- Un hardening completo
- Una guía CIS paso a paso
- Una receta mágica
- Una charla de una herramientas específicas

Esta charla `SÍ es`

- Una forma simple de empezar
- Un checklist inicial
- Una base para tomar decisiones

La Pregunta

> *¿Mi sistema GNU/Linux está razonablemente seguro para lo que hace?*

Para eso necesito saber:

- ¿Quién puede entrar?
- ¿Qué estoy ejecutando?
- ¿Qué está expuesto a internet?
- Si algo cambia, ¿me entero?
- ¿Quedan registros de esos cambios? (`logs`)

Porque es importante?

No puedo proteger `lo que desconozco`

> *No se trata de `sentir` que estoy seguro.*

Tengo que poder responder:

- **Usuarios / Permisos**
  - ¿Son todos necesarios?
- **Servicios / Puertos**
  - ¿Necesito todos los activos?
- **Registros (Logs)**
  - ¿Guardo registro de los cambios?
- **Archivos de configuración**
  - ¿Están protegidos de cambios?

Checklist general

Para cualquier sistema GNU/Linux

- Usuarios y grupos
- Privilegios con `sudo`
- Actualizaciones
- Servicios activos
- Puertos abiertos
- Registros del sistema (`Logs`)
- Software instalado

Revisión en Capas

1. Controles generales

- Aplican a cualquier sistema GNU/Linux

2. Notebook / Desktop

- **Controles principal:** Cifrado de disco, backups, pérdida o robo

3. Servidores

- **Controles principal:** Acceso remoto, Servicios expuestos, Firewall y monitoreo

Usuarios: ¿quién puede entrar?

```
$ getent passwd
$ getent group sudo
$ lastlog
$ last
```

Qué buscar

- Usuarios que no reconocés
- Cuentas sin uso
- Usuarios con privilegios
- Inicios de sesión inesperados

```
$ getent passwd | head
root:x:0:0:root:/root:/usr/bin/zsh
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
$ getent group sudo
sudo:x:27:pgonza
$ lastlog | grep -v "Never logged in"
Username         Port     From                                       Latest
root             tty1                                                Fri Jan 10 15:59:41 -0300 2025
pgonza           pts/8    192.168.100.26                             Fri May 29 10:27:31 -0300 2026
$
$ last -n 10
pgonza   pts/8        192.168.100.26   Fri May 29 10:27 - 13:24  (02:56)
pgonza   pts/7        192.168.100.26   Fri May 29 10:14   still logged in
pgonza   pts/6        192.168.100.26   Fri May 29 10:11   still logged in
pgonza   pts/5        192.168.100.26   Thu May 28 17:59   still logged in
pgonza   pts/1        192.168.100.26   Thu May 28 09:26   still logged in
pgonza   pts/4        192.168.100.26   Tue May 26 22:47   still logged in
pgonza   pts/2        192.168.100.26   Tue May 26 14:46   still logged in
pgonza   pts/1        192.168.100.26   Tue May 26 09:33 - 09:25 (1+23:52)
pgonza   pts/2        192.168.100.26   Sat May 23 01:11 - 14:45 (3+13:33)
pgonza   pts/0        192.168.100.26   Sat May 23 01:09 - 12:07 (6+10:58)

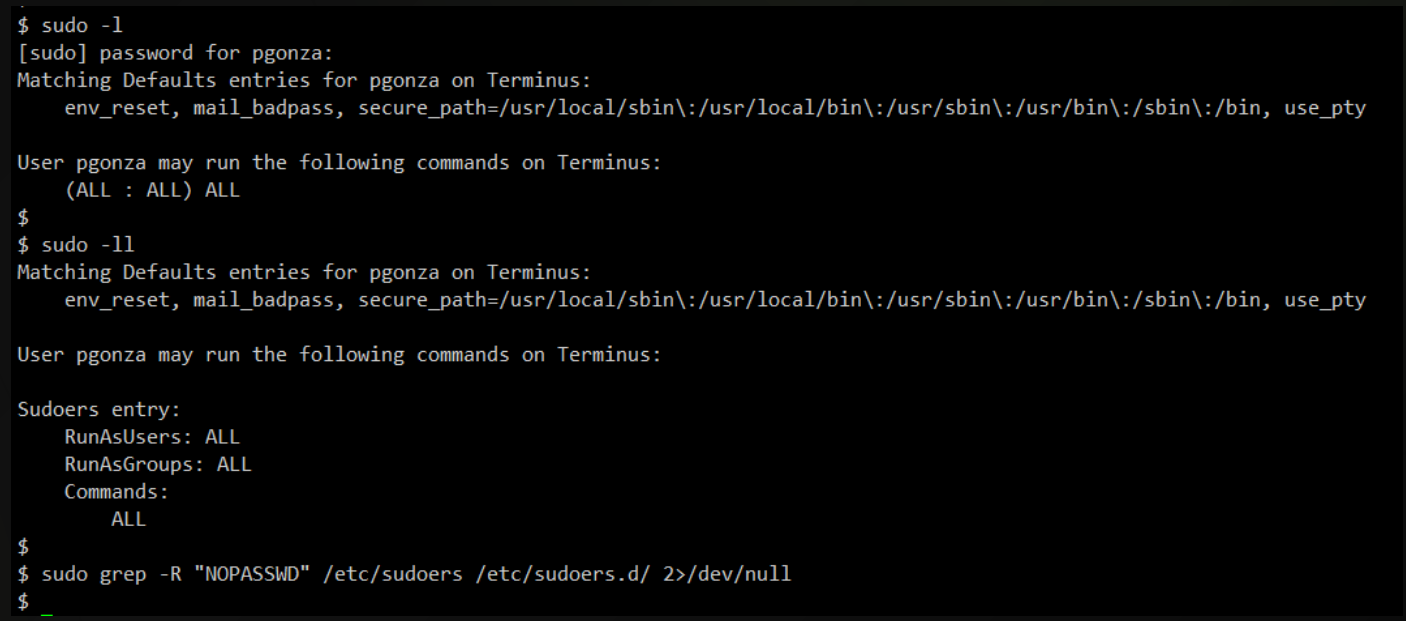
wtmp begins Mon Jan  6 12:15:02 2025
$
```

Privilegios: ¿quién puede hacer qué?

```
$ sudo -l
$ sudo -ll
$ sudo grep -R "NOPASSWD" \
$ /etc/sudoers /etc/sudoers.d/ 2>/dev/null
```

Señales de alerta

- `NOPASSWD` sin justificación en SUDO
- Permitir usuarios sin contraseñas
- Usuarios administrativos de más
- Reglas demasiado amplias
- Archivos en `/etc/sudoers.d/` olvidados



Actualizaciones: ¿estoy atrasado?

Debian / Ubuntu

```
$ apt update
$ apt list --upgradable
```

```
$ apt update
Hit:1 http://deb.debian.org/debian bookworm InRelease
Hit:2 http://deb.debian.org/debian bookworm-updates InRelease
Hit:3 http://deb.debian.org/debian-security bookworm-security InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
$
$ apt list --upgradable
Listing... Done
$
```

RHEL / Fedora / derivados

```
$ dnf check-update
$ dnf updateinfo list security
```

```
[root@AL2023 ~]# dnf check-update
Last metadata expiration check: 0:02:07 ago on Fri May 29 21:48:58 2026.
[root@AL2023 ~]#
[root@AL2023 ~]# dnf updateinfo list security
Last metadata expiration check: 0:02:22 ago on Fri May 29 21:48:58 2026.
[root@AL2023 ~]#
```

Software: ¿necesito todo lo instalado?

Debian / Ubuntu

```
$ dpkg -l | less
$ apt-mark showmanual
```

RHEL / Fedora / derivados

```
$ rpm -qa | less
```

```
$ snap list 2>/dev/null
$ flatpak list 2>/dev/null
```

Qué buscar

- Paquetes que no recordás haber instalado
- Herramientas duplicadas
- Repositorios extraños

Servicios activos

```
$ systemctl --type=service --state=running
$ systemctl --failed
```

Preguntas útiles

- ¿Reconozco estos servicios?
- ¿Necesitan estar activos?
- ¿Fallan servicios críticos?
- ¿Hay servicios heredados?

```
$ systemctl --type=service --state=running
  UNIT                      LOAD   ACTIVE SUB     DESCRIPTION
  containerd.service        loaded active running containerd container runtime
  cron.service              loaded active running Regular background program processing daemon
  dbus.service              loaded active running D-Bus System Message Bus
  docker.service            loaded active running Docker Application Container Engine
  exim4.service             loaded active running LSB: exim Mail Transport Agent
  getty@tty1.service        loaded active running Getty on tty1
  ssh.service               loaded active running OpenBSD Secure Shell server
  systemd-journald.service  loaded active running Journal Service
  systemd-logind.service    loaded active running User Login Management
  systemd-udevd.service     loaded active running Rule-based Manager for Device Events and Files
  user@1000.service         loaded active running User Manager for UID 1000

LOAD   = Reflects whether the unit definition was properly loaded.
ACTIVE = The high-level unit activation state, i.e. generalization of SUB.
SUB    = The low-level unit activation state, values depend on unit type.
11 loaded units listed.
$
$ systemctl --failed
  UNIT                    LOAD   ACTIVE SUB    DESCRIPTION
● dailyaidecheck.service  loaded failed failed daily AIDE check
● vboxadd.service         loaded failed failed vboxadd.service

LOAD   = Reflects whether the unit definition was properly loaded.
ACTIVE = The high-level unit activation state, i.e. generalization of SUB.
SUB    = The low-level unit activation state, values depend on unit type.
2 loaded units listed.
```

Puertos abiertos

```
$ ss -tulpen
```

Qué mirar

- Dirección local: `127.0.0.1` vs `0.0.0.0`
- Protocolo: TCP / UDP
- Puerto
- Proceso asociado
- Usuario del proceso

```
$ ss -tulpen
Netid   State    Recv-Q   Send-Q     Local Address:Port      Peer Address:Port  Process
udp     UNCONN   0        0             0.0.0.0:68             0.0.0.0:*      users:(("dhclient",pid=405,fd=7)) ino:1424
tcp     LISTEN   0        4096      127.0.0.1:64437            0.0.0.0:*      users:(("session-manager",pid=9812,fd=13))
tcp     LISTEN   0        4096        0.0.0.0:3030             0.0.0.0:*      users:(("docker-proxy",pid=1316,fd=7)) inc
tcp     LISTEN   0        128       127.0.0.1:6016             0.0.0.0:*      users:(("sshd",pid=203578,fd=9)) uid:1000
tcp     LISTEN   0        128       127.0.0.1:6011             0.0.0.0:*      users:(("sshd",pid=14674,fd=9)) uid:1000 i
tcp     LISTEN   0        128       127.0.0.1:6012             0.0.0.0:*      users:(("sshd",pid=16267,fd=9)) uid:1000 i
tcp     LISTEN   0        128       127.0.0.1:6013             0.0.0.0:*      users:(("sshd",pid=15239,fd=9)) uid:1000 i
tcp     LISTEN   0        128       127.0.0.1:6014             0.0.0.0:*      users:(("sshd",pid=133543,fd=9)) uid:1000
tcp     LISTEN   0        128       127.0.0.1:6015             0.0.0.0:*      users:(("sshd",pid=203528,fd=9)) uid:1000
tcp     LISTEN   0        4096      127.0.0.1:64356            0.0.0.0:*      users:(("session-manager",pid=203649,fd=13
tcp     LISTEN   0        128       127.0.0.1:10058            0.0.0.0:*      users:(("ssh",pid=203657,fd=5)) uid:1000 i
tcp     LISTEN   0        128         0.0.0.0:22               0.0.0.0:*      users:(("sshd",pid=495,fd=3)) ino:14444 sk
tcp     LISTEN   0        20        127.0.0.1:25               0.0.0.0:*      users:(("exim4",pid=95639,fd=4)) ino:35227
tcp     LISTEN   0        20            [::1]:25                  [::]:*      users:(("exim4",pid=95639,fd=5)) ino:35227
tcp     LISTEN   0        128           [::1]:10058               [::]:*      users:(("ssh",pid=203657,fd=4)) uid:1000 i
tcp     LISTEN   0        128           [::1]:6012                [::]:*      users:(("sshd",pid=16267,fd=7)) uid:1000 i
tcp     LISTEN   0        128           [::1]:6013                [::]:*      users:(("sshd",pid=15239,fd=7)) uid:1000 i
tcp     LISTEN   0        128           [::1]:6014                [::]:*      users:(("sshd",pid=133543,fd=7)) uid:1000
tcp     LISTEN   0        128           [::1]:6015                [::]:*      users:(("sshd",pid=203528,fd=7)) uid:1000
tcp     LISTEN   0        128           [::1]:6011                [::]:*      users:(("sshd",pid=14674,fd=7)) uid:1000 i
tcp     LISTEN   0        128           [::1]:6016                [::]:*      users:(("sshd",pid=203578,fd=7)) uid:1000
tcp     LISTEN   0        128            [::]:22                  [::]:*      users:(("sshd",pid=495,fd=4)) ino:14455 sk
$
```

Logs: el sistema habla

```
$ journalctl -p warning..alert -b
$ journalctl -u ssh --since "today"
$ journalctl --since "1 hour ago"
```

Qué buscar

- Errores repetidos
- Fallos de autenticación
- Servicios reiniciándose
- Mensajes que se repiten

```
$ sudo journalctl -p warning..alert -b | head
May 18 09:57:23 Terminus kernel: RETBleed: WARNING: Spectre v2 mitigation leaves CPU vulnerable to RETBleed attacks,
May 18 09:57:23 Terminus kernel: APIC calibration not consistent with PM-Timer: 103ms instead of 100ms
May 18 09:57:23 Terminus kernel: acpi PNP0A03:00: fail to add MMCONFIG information, can't access extended PCI config
May 18 09:57:23 Terminus kernel: device-mapper: core: CONFIG_IMA_DISABLE_HTABLE is disabled. Duplicate IMA measureme
May 18 09:57:24 Terminus systemd-journald[234]: File /var/log/journal/1f5541b2ec98487983894534b8863984/system.journa
May 18 09:57:26 Terminus (udev-worker)[262]: vboxguest: /etc/udev/rules.d/60-vboxadd.rules:1 Only network interfaces
May 18 09:57:26 Terminus (udev-worker)[262]: vboxuser: /etc/udev/rules.d/60-vboxadd.rules:2 Only network interfaces
May 18 09:57:26 Terminus lvm[289]: PV /dev/sda5 online, VG Terminus-vg is complete.
May 18 09:57:26 Terminus lvm[289]: VG Terminus-vg finished
May 18 09:57:28 Terminus kernel: [drm:vmw_host_printf [vmwgfx]] *ERROR* Failed to send host log message.
$
$ sudo journalctl -u ssh --since "today"
-- No entries --
$
$ sudo journalctl --since "1 hour ago" | head
May 30 00:00:08 Terminus systemd[1]: Starting dpkg-db-backup.service - Daily dpkg database backup service...
May 30 00:00:08 Terminus systemd[1]: Starting exim4-base.service - exim4-base housekeeping...
May 30 00:00:08 Terminus systemd[1]: exim4-base.service: Deactivated successfully.
May 30 00:00:08 Terminus systemd[1]: Finished exim4-base.service - exim4-base housekeeping.
May 30 00:00:08 Terminus systemd[1]: Starting logrotate.service - Rotate log files...
May 30 00:00:08 Terminus systemd[1]: dpkg-db-backup.service: Deactivated successfully.
May 30 00:00:08 Terminus systemd[1]: Finished dpkg-db-backup.service - Daily dpkg database backup service.
May 30 00:00:08 Terminus systemd[1]: logrotate.service: Deactivated successfully.
May 30 00:00:08 Terminus systemd[1]: Finished logrotate.service - Rotate log files.
May 30 00:17:01 Terminus CRON[207245]: pam_unix(cron:session): session opened for user root(uid=0) by (uid=0)
$
```

Archivos sensibles

Archivos que conviene conocer

Del Sistema

```
/etc/passwd
/etc/shadow
/etc/group
/etc/sudoers
/etc/pam.d/
/etc/security/
```

De servicios criticos

```
/etc/ssh/sshd_config
```

Notebook / desktop: controles rápidos

Riesgos típicos

- Robo o pérdida física
- Sesiones abiertas
- Software instalado sin control
- Backups inexistentes

Controles mínimos

- Cifrado de disco
- Bloqueo de pantalla
- Actualizaciones frecuentes
- Backups probados
- Usuario sin privilegios para uso diario

Servidores: controles críticos

Riesgos típicos

- SSH expuesto
- Servicios escuchando en red
- Firewall inexistente
- Logs que nadie revisa
- Cambios no detectados

Prioridad

- Primero reducir exposición.
- Después mejorar detección.

SSH: revisar antes de endurecer

```
$ ss -tulpen | grep ':22'
$ cat /etc/ssh/sshd_config | \
$ grep -E '^(PermitRootLogin|PasswordAuthentication|PubkeyAuthentication)'
$ sudo journalctl -u ssh --since "today"
```

Preguntas

- ¿SSH está expuesto?
- ¿Se permite root?
- ¿Se permite password?
- ¿Hay intentos fallidos?

```
$
$ ss -tulpen | grep ':22'
tcp   LISTEN 0      128        127.0.0.1:6013       0.0.0.0:*    uid:1000 ino:224640 sk:7 cgroup:/user.sl
tcp   LISTEN 0      128          0.0.0.0:22         0.0.0.0:*    ino:14444 sk:8 cgroup:/system.slice/ssh.
tcp   LISTEN 0      128            [::1]:6013          [::]:*    uid:1000 ino:224639 sk:a cgroup:/user.sl
tcp   LISTEN 0      128             [::]:22            [::]:*    ino:14455 sk:d cgroup:/system.slice/ssh.
$ sudo ss -tulpen | grep ':22'
tcp   LISTEN 0      128        127.0.0.1:6013       0.0.0.0:*    users:(("sshd",pid=15239,fd=9)) uid:1000
tcp   LISTEN 0      128          0.0.0.0:22         0.0.0.0:*    users:(("sshd",pid=495,fd=3)) ino:14444
tcp   LISTEN 0      128            [::1]:6013          [::]:*    users:(("sshd",pid=15239,fd=7)) uid:1000
tcp   LISTEN 0      128             [::]:22            [::]:*    users:(("sshd",pid=495,fd=4)) ino:14455
$
$
$
$ ss -tulpen | grep ':22'
tcp   LISTEN 0      128        127.0.0.1:6013       0.0.0.0:*    uid:1000 ino:224640 sk:7 cgroup:/user.sl
tcp   LISTEN 0      128          0.0.0.0:22         0.0.0.0:*    ino:14444 sk:8 cgroup:/system.slice/ssh.
tcp   LISTEN 0      128            [::1]:6013          [::]:*    uid:1000 ino:224639 sk:a cgroup:/user.sl
tcp   LISTEN 0      128             [::]:22            [::]:*    ino:14455 sk:d cgroup:/system.slice/ssh.
$
$ cat /etc/ssh/sshd_config | grep -E '^(PermitRootLogin|PasswordAuthentication|PubkeyAuthentication)'
PubkeyAuthentication yes
PasswordAuthentication yes
$
$ sudo journalctl -u ssh --since "today"
-- No entries --
$
```

Firewall

> *no alcanza con “tenerlo instalado”*

Ver estado

```
$ sudo ufw status verbose
$ sudo nft list ruleset
```

Preguntas

- ¿Hay política por defecto?
- ¿Qué puertos están permitidos?
- ¿La regla sigue siendo necesaria?
- ¿Coincide con los servicios reales?

Lynis: ordenar la auditoría

```
$ sudo lynis audit system --quick
```

Para qué sirve

- Revisión rápida
- Sugerencias de hardening
- Punto de partida repetible

NO es

- Una garantía de seguridad
- Un reemplazo del criterio técnico

```
================================================================================

  Lynis security scan details:

  Hardening index : 64 [#############       ]
  Tests performed : 278
  Plugins enabled : 2

  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: cómo leer resultados

Clasificá hallazgos (sin entrar en pánico)

1. **Crítico y expuesto a internet**
2. **Importante pero interno**
3. **Mejora recomendada**
4. **No aplica a mi caso**

> *No todo warning exige acción inmediata*

systemd-analyze: check de servicios

```
$ systemd-analyze security
```

Para qué `sirve`

- Evalúar aislamiento de los servicios
- Revisa privilegios y capacidades
- Ayuda a pensar en menor privilegio

Para qué `NO sirve`

- Un detector de vulnerabilidades
- Asegurar que el servicio es inseguro



UNIT	EXPOSURE	PREDICATE	HAPPY
clamav-freshclam.service	9.6	UNSAFE	😨
containerd.service	9.6	UNSAFE	😨
cron.service	9.6	UNSAFE	😨
dbus.service	9.6	UNSAFE	😨
dm-event.service	9.5	UNSAFE	😨
docker.service	9.6	UNSAFE	😨
emergency.service	9.5	UNSAFE	😨
exim4.service	9.6	UNSAFE	😨
getty@tty1.service	9.6	UNSAFE	😨
ifup@enp0s3.service	9.5	UNSAFE	😨
lvm2-lvmpolld.service	9.5	UNSAFE	😨
rc-local.service	9.6	UNSAFE	😨
rescue.service	9.5	UNSAFE	😨
ssh.service	9.6	UNSAFE	😨
systemd-ask-password-console.service	9.4	UNSAFE	😨
systemd-ask-password-wall.service	9.4	UNSAFE	😨
systemd-fsckd.service	9.5	UNSAFE	😨
systemd-initctl.service	9.4	UNSAFE	😨
systemd-journald.service	4.3	OK	🙂
systemd-logind.service	2.8	OK	🙂
systemd-networkd.service	2.6	OK	🙂
systemd-timesyncd.service	2.1	OK	🙂
systemd-udevd.service	7.1	MEDIUM	😐
user@1000.service	9.4	UNSAFE	😨
vboxadd-service.service	9.6	UNSAFE	😨

Mini checklist

Hoy mismo podés revisar

- Usuarios con acceso
- Privilegios administrativos
- Política de contraseñas
- Servicios activos
- Puertos abiertos
- Logs recientes
- Estado de actualizaciones
- Configuración SSH
- Firewall
- Hallazgos de Lynis

El límite de la auditoría manual

> *Auditar una vez sirve para saber cómo estás hoy, pero no te dice qué cambia mañana.*

Problemas

- No escala a muchos sistemas
- Depende de acordarse o de programar ejecución
- Llega tarde ante incidentes
- Es difícil comparar cambios en el tiempo

Auditoría manual vs Monitoreo continuo

Cambio de enfoque

Antes

- yo `busco problemas`
- revisión `puntual`
- `comandos` sueltos

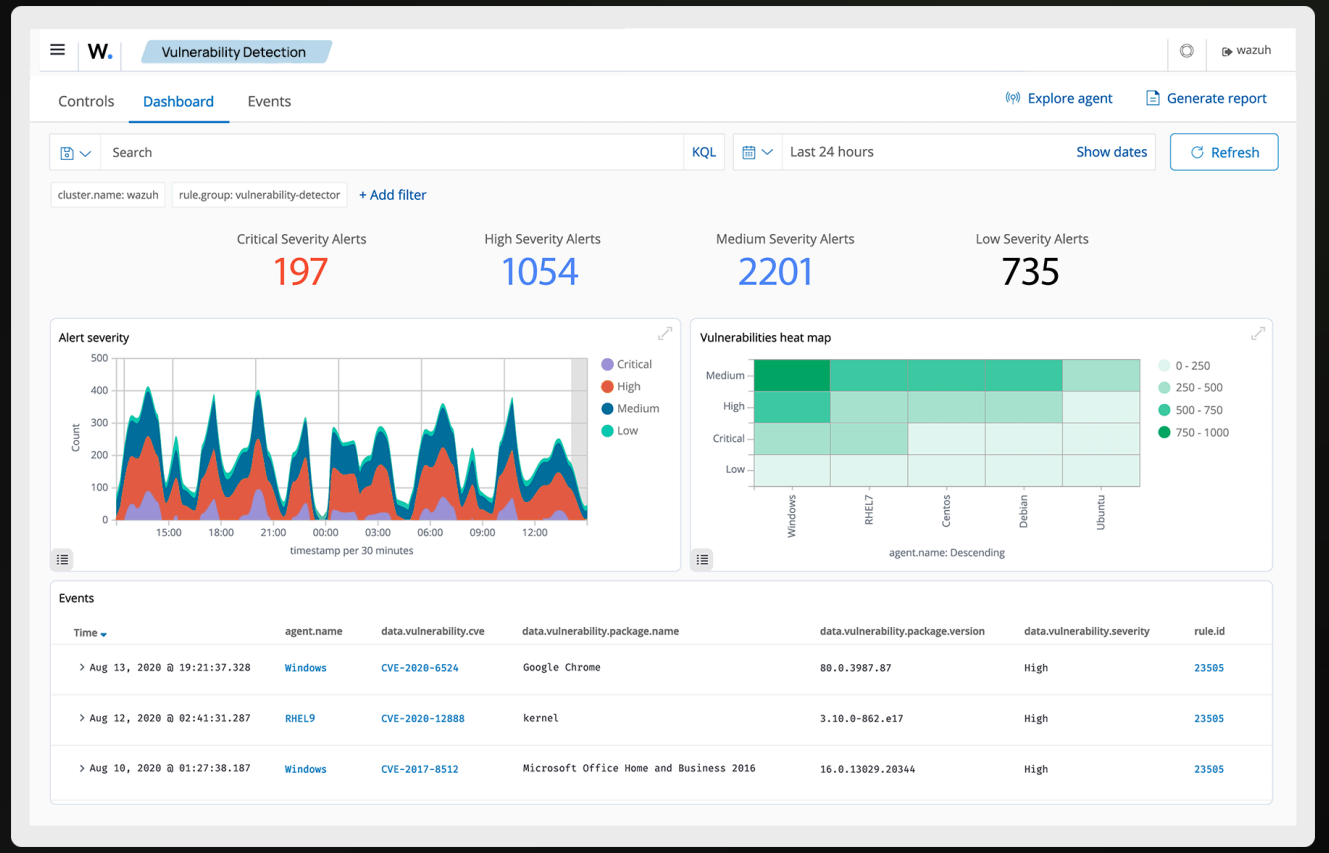
Despues

- el sistema `me avisa`
- visibilidad `continua`
- `eventos` centralizados

Monitoreo continuo (Wazuh)

Qué puede aportar

- `Inventario` de sistemas
- `Detección` de vulnerabilidades
- `Integridad` de archivos
- `Análisis` de logs
- Alertas `centralizadas`



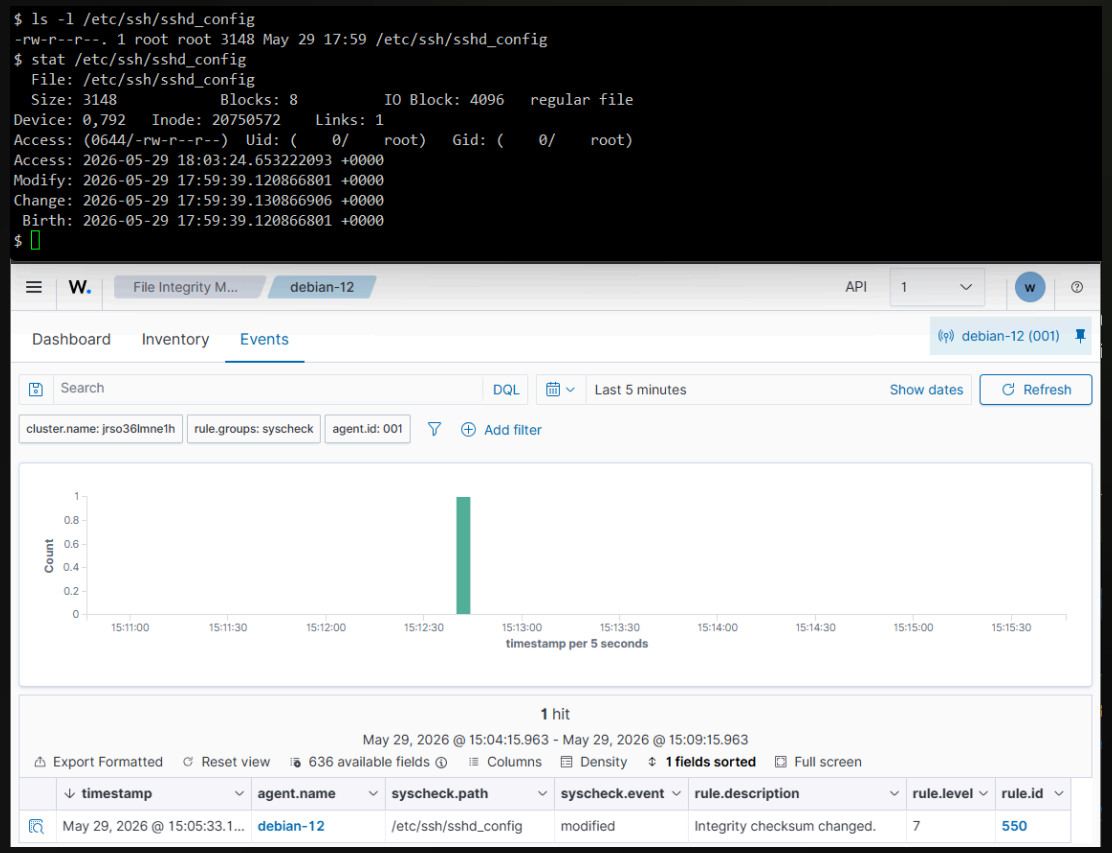
Ejemplo: Cambio en archivo crítico

Auditoría manual

```
$ ls -l /etc/ssh/sshd_config
$ stat /etc/ssh/sshd_config
```

Monitoreo continuo

- Se modifica el archivo
- Se genera un evento
- Se dispara una alerta
- Alguien puede investigarlo



Recursos para seguir

Checklists y guías

- How-To-Secure-A-Linux-Server
- CIS Benchmarks
- awesome-security-hardening
- Distros:
  - Debian Security Manual
  - Red Hat Security hardening
  - Arch Linux Security

Herramientas

- Lynis
- OpenSCAP / SCAP Security Guide
- auditd
- Wazuh

Entonces, ¿con que comienzo?

Seguridad práctica

- Empezá por lo visible
- Medí antes de endurecer
- Priorizá según exposición
- Automatizá lo repetible
- Revisá de nuevo

¿Preguntas, ideas o ganas de seguir charlando?



🌐 linktr.ee/sultanovich.sh | 🐙 github.com/sultanovich

```
$ echo "despedida.txt"

                    Gracias por quedarte hasta el final 🙏
```