



Fuente: IA Stable Diffusion

LABORATORIO 6. SEGURIDAD DE SO AUTOMATIZABLE Y POLÍTICAS DE SEGURIDAD

DEPARTAMENTO DE INFORMÁTICA. UNIVERSIDAD DE OVIEDO

Seguridad de Sistemas Informáticos | 2022 – 2023 (v3.1 "S-81 Isaac Peral")





CONTENIDO

La Infraestructura de este laboratorio.....	3
BLOQUE 1: Administración de directivas OSCAP	5
Actualizando las políticas de seguridad de <i>OpenSCAP</i>	6
Pasar un perfil <i>OpenSCAP</i> al navegador <i>Firefox</i>	6
Pasar perfiles de <i>OpenSCAP</i> al SO.....	7
Interpretación de informes OSCAP	8
Uso de OSCAP desde la línea de comandos	8
Bloque 2. Remediación con OSCAP	10
Remediación con <i>SCAP Workbench</i>	11
Remediación con <i>scripts</i> de <i>bash</i>	11
Remediación con <i>Ansible</i>	11
Bloque 3: Examinando STIGs.....	13
Uso de <i>OpenSCAP</i> con STIGs	14
Remediación STIG con <i>Ansible</i>	14
Bloque 4: Aplicar una política de seguridad de 3ºs a un <i>Ubuntu server</i>	15
Hardening de Ubuntu automatizado con <i>scripts</i> de terceros	16
<i>Script Ansible</i> de <i>Florian Utz</i>	16
El Proyecto <i>Egida</i> de hardening CIS automatizado con <i>Ansible</i>	18
Insignias y Autoevaluación	19



AVISO

Este documento forma parte de la asignatura “Seguridad de Sistemas Informáticos”, impartida en la *Escuela de Ingeniería Informática* de la *Universidad de Oviedo*. Es fruto del trabajo continuado de elaboración, soporte, mejora, actualización y revisión del siguiente equipo de profesores desde el año 2019

- Enrique Juan de Andrés Galiana
- Fernando Cano Espinosa
- Miguel Riesco Albizu
- José Manuel Redondo López
- Luís Vinuesa Martínez

Te pedimos por favor que **NO lo compartas públicamente en Internet**. No obstante, entendemos que puedas considerar este material interesante para otras personas. Por ese motivo, hemos creado una versión de este adaptada para que pueda cursarse de forma online, disponible gratuitamente para todo el mundo y que puedes encontrar en esta dirección: <https://ocw.uniovi.es/course/view.php?id=109>

A diferencia de esta versión, **la versión libre puedes promocionarla todo lo que quieras**, que para eso está 😊

GRACIAS POR TU COLABORACIÓN



LA INFRAESTRUCTURA DE ESTE LABORATORIO





Este laboratorio **no se puede ejecutar en contenedores**, ya que las herramientas que vamos a utilizar no están diseñadas para ejecutarse en ellos. De hecho, existe una versión especial de **oscap** diseñada para ser ejecutada dentro de contenedores *Docker* (**oscap-docker**), pero su uso está fuera del alcance de la asignatura. Por lo tanto, **te recomendamos encarecidamente que hagas una instantánea de tu máquina virtual actual** para ejecutar los programas de este laboratorio. **Consulta la documentación de laboratorios adicional** para saber **cómo crear instantáneas**.

La carpeta de infraestructura del Lab 6 contiene algunos archivos necesarios para realizar las actividades de este laboratorio, para ahorrar tiempo.





BLOQUE 1: ADMINISTRACIÓN DE DIRECTIVAS OSCAP



Actualizando las políticas de seguridad de OpenSCAP

Aplicación práctica: Necesitas actualizar las reglas de hardening del paquete `scap-security-guide` a la última versión

La herramienta de *Compliance as Code* OpenSCAP y SCAP Workbench (una GUI gratuita de OpenSCAP) ya están instaladas en la máquina virtual de la asignatura. Si creaste tu propia máquina, para instalar la herramienta OpenSCAP tienes que seguir los pasos de instalación oficiales para Ubuntu que aparecen aquí: <https://www.open-scap.org/tools/openscap-base/#download>. Instala también SCAP Workbench, con: `sudo apt install scap-workbench`

Para garantizar el mejor resultado al ejecutar OpenSCAP es necesario actualizar sus políticas de seguridad. Estas contienen reglas de validación y corrección para diferentes sistemas operativos y software.

- Descarga la última versión de estas políticas desde su repositorio oficial de GitHub (<https://github.com/ComplianceAsCode/content>). Actualiza el número de versión de este enlace al último que esté disponible: `wget https://github.com/ComplianceAsCode/content/releases/download/v0.1.59/scap-security-guide-0.1.59-oval-510.zip`, o vete a la sección **Releases** del repositorio de GitHub (**NOTA:** En los ficheros de la infraestructura del Lab 6 hay una carpeta con una política antigua de este repositorio, puedes utilizar este fichero y evitar descargar uno más actualizado, porque para esta actividad no hace falta "estar a la última" necesariamente).
- Descomprime el archivo descargado y copia su contenido en una carpeta que crees. Esta carpeta será la que usarás más adelante para cargar los contenidos de OpenSCAP.

Una vez hecho esto, tendremos los últimos **perfiles de seguridad** disponibles. Cada perfil de seguridad se adapta a un uso específico (organización, entidad pública...) e incluye una lista diferente de controles de seguridad. Al seleccionar uno de ellos, puedes verificar **cuántas de sus reglas de seguridad cumple tu sistema** e incluso corregir automáticamente algunas de ellas. Para saber los perfiles disponibles para Ubuntu 18.04 haz: `sudo oscap info <carpeta descomprimida>/ssg-ubuntu1804-ds-1.2.xml`

Resultados esperados: Esta actividad finalizará cuando descargues e instales los perfiles de seguridad más actualizados de `oscap` y puedas enumerar los perfiles de seguridad disponibles para Ubuntu

Pasar un perfil OpenSCAP al navegador Firefox

Aplicación práctica: Necesitas una lista de controles de seguridad para hacer hardening de Firefox manualmente

Abre **SCAP Workbench** (**System – Scap workbench**) y usa la opción **File – Open Other Content** para cargar un perfil de seguridad de Firefox desde la carpeta en la que descomprimiste el fichero descargado en el ejercicio anterior (fichero `ssg-firefox-ds-1.2.xml`). Ejecuta **Scan** y haz clic en **Show report**, analizando el contenido del informe. ¿Qué ha pasado? ¿Cuál es la utilidad de este perfil SCAP para este producto? ¿Cuál es el origen de este perfil de seguridad (es decir, de dónde se han sacado sus controles técnicos de seguridad)? (STIG, CIS, otros...)

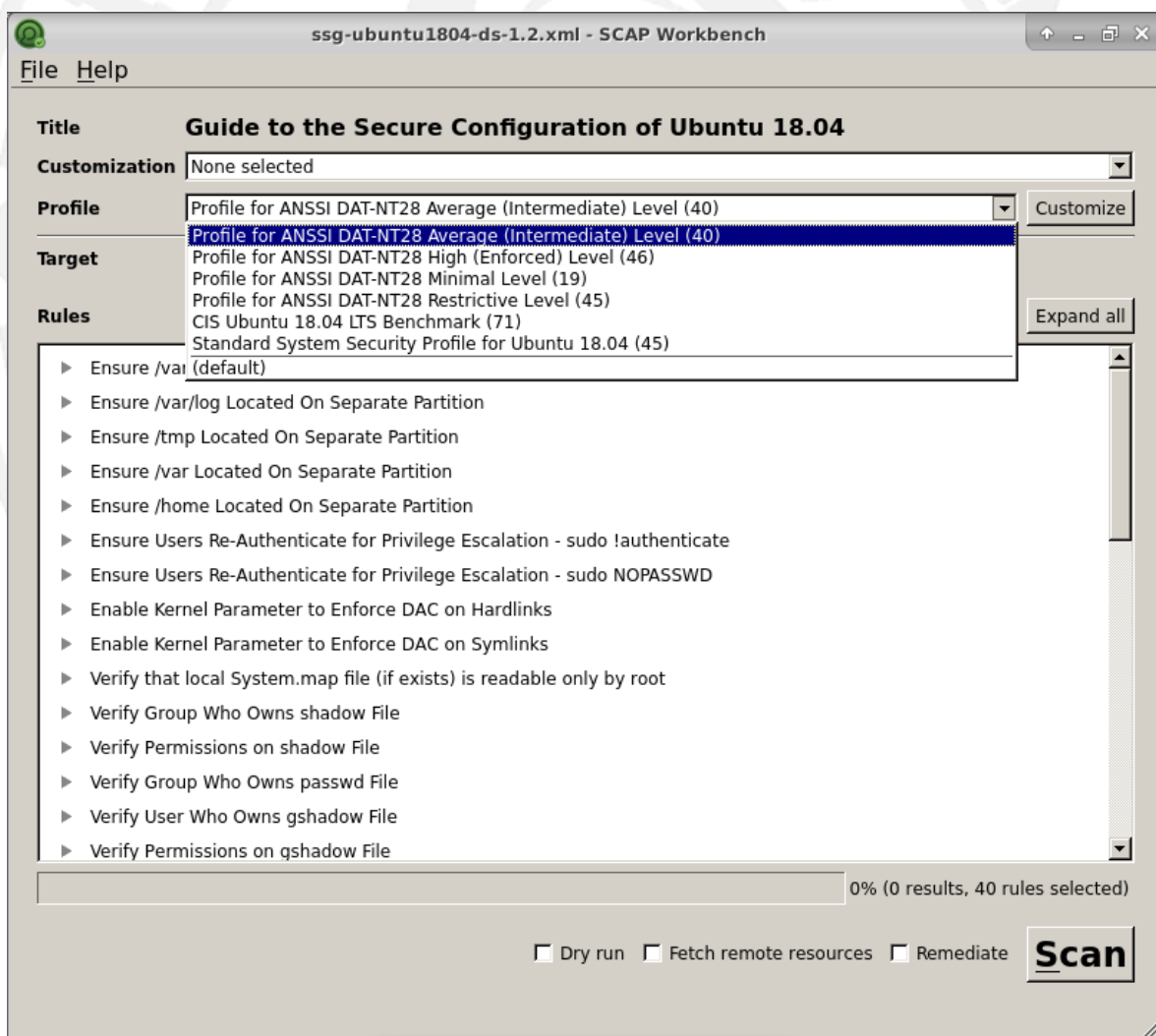
Resultados esperados: Esta actividad finalizará cuando ejecutes un perfil de seguridad de Firefox y seas capaz de interpretar el contenido del informe generado y sus resultados. ¿Se comprueba y/o aplica alguno de los controles de seguridad?

Pasar perfiles de OpenSCAP al SO

Aplicación práctica: Necesitas evaluar tu SO Linux de acuerdo con perfiles de seguridad internacionales

Repite el procedimiento anterior para cargar el perfil de seguridad para el sistema operativo de tu MV de la carpeta donde descomprimiste las reglas que descargarse de *GitHub* en el primer ejercicio (usa el archivo **ssg-ubuntu1804-ds-1.2.xml**). Analiza los perfiles de seguridad disponibles y la cantidad de controles de seguridad que contienen (entre paréntesis). Una vez cargados, haz las siguientes operaciones:

- Deja **"Dry run"** y **"Remediate"** sin marcar
 - Marca **"Fetch remote resources"**
 - Escanea el sistema operativo utilizando los siguientes tres perfiles, **guardando cada informe que obtengas después de cada escaneo en un fichero HTML diferente**. Si se notifican errores, ignóralos.
- Los perfiles son:
- *ANSSI DAT-NT28 High profile* (un estándar de seguridad francés)
 - El perfil de seguridad estándar de *Ubuntu*
 - El perfil que sigue los *CIS Benchmarks*.



Resultados esperados: Esta actividad finalizará cuando puedas usar *SCAP Workbench* para escanear el sistema operativo automáticamente en busca de vulnerabilidades de seguridad de acuerdo con un perfil de seguridad cargado y puedas también almacenar sus informes en diferentes archivos



Interpretación de informes OSCP

Aplicación práctica: Necesitas obtener un informe de seguridad OSCP y saber interpretarlo

Después de ejecutar el análisis de los perfiles mencionados en el ejercicio anterior, abre los informes HTML generados y asegúrate de comprender los siguientes elementos del informe.

- El **origen** del perfil de seguridad y la descripción inicial.
- Los **resultados**: número de reglas pasadas, reglas que el sistema no cumple y en otras situaciones
- La **distribución de gravedad** de las reglas que el sistema no cumple
- La **puntuación de seguridad** del sistema según la política de seguridad usada
- La **lista de controles de seguridad** clasificada y su estado actual de controles pasados y no pasados
- La **descripción detallada** de cada control de seguridad (Descripción, Justificación, Gravedad...)

Resultados esperados: Esta actividad finalizará cuando comprendas cada sección de un informe de **oscap** y localices todos los elementos enumerados dentro de él.

Uso de OSCP desde la línea de comandos

Aplicación práctica: Necesitas ejecutar **oscap** sin un GUI

Una cosa que debemos tener en cuenta es que en la mayoría de las ocasiones las máquinas servidores no instalarán un GUI para ahorrar recursos y disminuir vectores de ataque. Si no tenemos GUI, o la posibilidad de instalar *SCAP Workbench*, podemos realizar las operaciones de los ejercicios anteriores **usando la línea de comandos**. Antes de continuar con este laboratorio, debes aprender a ejecutar un escaneo como los anteriores utilizando solo la línea de comandos. Para ello, usa la siguiente información:

- La herramienta de línea de comandos se llama **oscap** y requiere privilegios de **root**.
- Para evaluar el estado de seguridad de la máquina se necesita añadir dos parámetros al comando anterior separados por un espacio en blanco: **xccdf eval**. **NOTA:** Si también queremos intentar remediar automáticamente los posibles controles de seguridad que fallan en el escaneo (**visto más adelante, ¡no lo hagas ahora!**), debemos usar **xccdf eval --remediate** en su lugar.
- Después de eso, requiere un parámetro **--profile**. Después de un espacio en blanco, hay que poner el nombre de id de perfil completo (con puntos) que puedes obtener de los resultados del comando **oscap info** de la primera actividad.
- Tras el perfil es necesario un parámetro **--results**. Tras un espacio en blanco, necesita un nombre de archivo **.xml** para guardar el informe en este formato.
- Después de los resultados es necesario un parámetro **--report**. Tras un espacio en blanco, lleva un nombre de archivo **.html** para guardar un informe "legible por humanos" en este formato (este es el que podemos ver en *SCAP Workbench*).
- Finalmente, después de un espacio en blanco, necesita la ruta al archivo **.xml** con la política de seguridad a usar con el sistema operativo actual. Estos archivos están en el archivo de políticas descargado de *GitHub* que descomprimos, y es el que define el nombre del perfil de seguridad especificado en la opción **--profile** (en nuestro caso es el archivo **ssg-ubuntu1804-ds-1.2.xml**)



Asegúrate de que el informe que obtengas tiene los mismos resultados que la ejecución equivalente con SCAP Workbench (a menos que hayas corregido algo antes, en este caso el informe probablemente tendrá más controles de seguridad solucionados). **Solo tienes que probar esto con uno de los tres perfiles que utilizaste en el ejercicio anterior.**

Resultados esperados: Esta actividad finalizará cuando puedas usar `oscap` desde línea de comandos





BLOQUE 2. REMEDIACIÓN CON OSCAP



Remediación con SCAP Workbench

Aplicación práctica: Necesitas hacer hardening automático a tu SO siguiendo las políticas del paquete **scap-security-guide** usando remediación OSCAP

Primero, ejecuta una auditoría con **Lynis** para tener una puntuación de seguridad base y almacena el resultado en un archivo que puedas consultar más adelante. Vuelve a seleccionar el perfil CIS e intenta corregir automáticamente sus controles de seguridad marcando "Remediate" en el GUI. Compara y analiza los resultados con la ejecución inicial del perfil CIS de la sección anterior. **Ejecuta la auditoría Lynis de nuevo y almacena el informe en un archivo diferente.** Compara la puntuación **Lynis** y el porcentaje de reglas SCAP aplicadas correctamente antes y después de la corrección. ¿Han mejorado?

Resultados esperados: Esta actividad finalizará cuando utilices **oscap** o **SCAP Workbench** para intentar corregir los controles de seguridad que no pasan sus comprobaciones. Una vez que tengas los resultados, debes comparar la puntuación de **Lynis** y el % de las reglas SCAP aplicadas correctamente para determinar si la corrección tuvo un efecto positivo, y determinar a partir de tus resultados si crees que los perfiles de **Lynis** y **SCAP** miden cosas diferentes (en otras palabras, si usan diferentes controles de seguridad)

Remediación con scripts de bash

Aplicación práctica: Necesitas hacer hardening automático a tu SO siguiendo las políticas del paquete **scap-security-guide** usando remediación con scripts de bash

Vete a los perfiles de seguridad descargados y descomprimidos en una carpeta al principio del laboratorio y accede a la carpeta **bash** dentro de ella. Localiza el *script* de shell correspondiente al perfil de seguridad CIS de **Ubuntu**, hazlo **ejecutable** y ejecútalo con privilegios de **root** para ver cuántos controles de seguridad se han corregido ahora. Audita el perfil de seguridad de nuevo **sin corrección automática** (a través de **SCAP Workbench** o línea de comandos) y compara el nuevo informe con el anterior. Además, haz de nuevo una auditoría **Lynis** y compárala con la hiciste en el ejercicio anterior.

Resultados esperados: Esta actividad finalizará cuando uses los scripts de corrección **bash** incluidos en el fichero de directivas de seguridad descargado para intentar corregir los controles de seguridad que no pasen sus comprobaciones. También debes comparar de nuevo el % de reglas aplicadas correctamente y las puntuaciones de **Lynis** antes y después de la remediación, ¿crees ahora que los perfiles SCAP y **Lynis** miden cosas diferentes?

Remediación con Ansible

Aplicación práctica: Necesitas hacer hardening automático a tu SO siguiendo las políticas del paquete **scap-security-guide** usando remediación con Ansible

El procedimiento de remediación final incluido en los archivos **oscap** es a través del programa de automatización de configuraciones **Ansible**. Para ejecutar las políticas de corrección con **Ansible** (ya instalado en la máquina



virtual de la asignatura, pero instalable con: `sudo apt install ansible`), debes hacer el siguiente procedimiento:

- Vete a los perfiles de seguridad descargados, accede a la carpeta `ansible` y localiza el archivo `.yaml` correspondiente al perfil CIS
- Edita este `.yaml` y cambia `hosts: all` por `hosts: localhost`
- Ejecuta `sudo ansible-playbook ubuntu1804-playbook-cis.yaml`
- Ejecuta una verificación de la política nuevamente con `oscap` y una auditoría con *Lynis* para ver si los resultados mejoran

Resultados esperados: Esta actividad finalizará cuando uses los *scripts* de corrección de *Ansible* incluidos en el archivo de directivas de seguridad descargados para intentar corregir los controles de seguridad que no superen sus comprobaciones. También debes comparar nuevamente el % de reglas aplicadas correctamente y las puntuaciones de *Lynis* antes y después de la remediación para determinar qué procedimiento de remediación obtiene mejores resultados.



BLOQUE 3: EXAMINANDO STIGS





Uso de OpenSCAP con STIGs

Aplicación práctica: Necesitas una lista de controles de seguridad para hacer hardening manual de tu SO de acuerdo con las políticas STIG

Las últimas versiones de los STIGs se pueden descargar desde aquí: <https://public.cyber.mil/stigs/downloads/>. Busca la última versión de un STIG adecuado para tu sistema *Ubuntu* (algo como "**Canonical Ubuntu 18.04 LTS STIG - Version 2, Release 5**"), descárgalo y descomprímelo como hicimos con el contenido del repositorio de *GitHub de ComplianceAsCode*. Examina sus perfiles utilizando los comandos de información de **oscap** (o *SCAP Workbench*) y busca paralelismos entre los perfiles que encuentres y los vistos en temas de teoría. Una versión de este archivo está en los materiales de laboratorio si no quieres descargarlo.

Una vez examinados los contenidos, evalúa el sistema actual con cualquiera de los perfiles de seguridad disponibles. Analiza los resultados y extrae conclusiones sobre los usos potenciales de los STIG proporcionados para *Ubuntu 18.04* con **oscap**. ¿Se pueden auditar o remediar automáticamente los controles de seguridad con este fichero?

Resultados esperados: Esta actividad finalizará cuando puedas descargar un perfil STIG adecuado para el sistema operativo actual, evaluarlo, analizar los resultados que produce y sus usos potenciales.

Remediación STIG con Ansible

Aplicación práctica: Necesitas hacer hardening automático a tu SO *Ubuntu* siguiendo las políticas STIG

Desde la misma página del ejercicio anterior, descarga **la versión de Ansible del perfil STIG** (algo así como "**Canonical Ubuntu 18.04 LTS STIG for Ansible - Ver 2, Rel 5**") y usa el mismo procedimiento que en el ejercicio "*Remediación con Ansible*" para ejecutar las correcciones automáticas contenidas en este archivo. Una versión de este archivo está en los materiales de laboratorio si no quieres descargarlo. Evalúa posteriormente el sistema con **oscap** y *Lynis* y compara los resultados con los ejercicios anteriores, determinando qué procedimiento de remediación automática es mejor.

Resultados esperados: Esta actividad finalizará cuando puedas descargar un script de corrección STIG *Ansible* adecuado para el STIG anterior, ejecutarlo y evaluar la puntuación de seguridad final del sistema.



BLOQUE 4: APLICAR UNA POLÍTICA DE SEGURIDAD DE 3ºS A UN *UBUNTU* *SERVER*

Hardening de Ubuntu automatizado con scripts de terceros

Aplicación práctica: Necesitas hacer hardening automático de tu SO Ubuntu siguiendo las políticas CIS usando para ello un script de terceros creado por Florian Utz

Script Ansible de Florian Utz

(**NOTA:** Después de ejecutar este script de remediación ya no tendrás acceso a la GUI (¡tener una GUI es un problema de seguridad en un servidor y el script la desinstala! 😞), por lo **que es muy importante tener una instantánea creada para restaurarla rápidamente**)

Aunque existe la opción de hacer hardening automatizado CIS para Ubuntu con políticas oficiales gracias, por ejemplo, a la licencia de **Ubuntu Pro** (<https://ubuntu.com/pro>), esto conlleva un registro previo (gratuito hasta 5 máquinas) que no es adecuado usar en este entorno de pruebas. Existen alternativas gratuitas (y no oficiales) creadas por otros usuarios y de código abierto disponibles que podemos usar para practicar la automatización. Vamos a utilizar una de ellas para comprobar las capacidades de hardening automatizado de las políticas CIS: <https://github.com/florianutz/Ubuntu1804-CIS>.

- **Te recomendamos encarecidamente que crees una nueva instantánea o clon de la máquina virtual para realizar esta actividad**, ya que, como hemos dicho, los cambios realizados en el sistema generan problemas para los siguientes laboratorios. Consulta la documentación adicional de los laboratorios para saber cómo crear instantáneas o clones de máquinas virtuales.
- Ahora sigue las instrucciones del repositorio de *GitHub* de esta política CIS para ejecutarla. Primero, crea un archivo `requirements.yml` con este contenido (los archivos de *Ansible* se procesan en *Python*, así mucho cuidado con la alineación de las líneas y **no uses tabuladores**). Para facilitar el trabajo, este archivo se incluye en la carpeta de la infraestructura del **Lab 6**:

```
- src: https://github.com/florianutz/Ubuntu1804-CIS.git
```

- Guarda este archivo e instale la directiva CIS con él mediante este comando

```
sudo ansible-galaxy install -p roles -r requirements.yml
```

(**NOTA:** Si este paso falla debido a problemas de conexión o falta de disponibilidad del repositorio por algún motivo, el resultado de esta operación se proporciona como parte de los archivos de laboratorio. Puedes obtener el resultado descomprimiendo el archivo `CISflorianutz.zip` que proporcionamos en el directorio en el que estás trabajando. Esto es solo una precaución en caso de que esto falle, no tienes por qué hacerlo así).

- Una vez instalada la directiva de hardening, crea un **archivo de configuración** para especificar dónde ejecutarla. Para ello, primero crea el archivo `server.yml` con el siguiente contenido.
 - Especifica que la política se aplicará en la propia máquina (`localhost`)
 - Requiere que el usuario se convierta en `root` para aplicarla
 - El nombre de los roles (conjunto de operaciones que se realizarán, `Ubuntu1804-CIS` como se indica en las instrucciones de la política) que se ejecutarán
 - Un nombre descriptivo para toda la operación

Para que sea más sencillo, este archivo se proporciona como parte de la carpeta de la **infraestructura del Lab 6**.



```
- name: Harden Server
  hosts: localhost
  become: yes

  roles:
    - Ubuntu1804-CIS
```

- Finalmente, **ejecuta la directiva** (`sudo ansible-playbook server.yml`) y espera a que el sistema se configure de forma segura. Después de eso, ejecuta **reboot** e inicia sesión normalmente antes de continuar con el siguiente paso para saber si el sistema está funcionando correctamente. **El proceso de ejecución lleva bastante tiempo y se ve así:**

```
TASK [Ubuntu1804-CIS : SCORED | 1.1.14 | PATCH | Ensure nodev option set on /dev/shm partition]
SCORED | 1.1.15 | PATCH | Ensure nosuid option set on /dev/shm partition
SCORED | 1.1.16 | PATCH | Ensure noexec option set on /dev/shm partition] ***
changed: [localhost]

TASK [Ubuntu1804-CIS : NOTSCORED | 1.1.17 | PATCH | Ensure nodev option set on removable media partitions] ***
ok: [localhost]

TASK [Ubuntu1804-CIS : NOTSCORED | 1.1.18 | PATCH | Ensure nosuid option set on removable media partitions] ***
ok: [localhost]

TASK [Ubuntu1804-CIS : NOTSCORED | 1.1.19 | PATCH | Ensure noexec option set on removable media partitions] ***
ok: [localhost]

TASK [Ubuntu1804-CIS : SCORED | 1.1.20 | PATCH | Ensure sticky bit is set on all world-writable directories] ***
ok: [localhost]

TASK [Ubuntu1804-CIS : SCORED | 1.1.21 | PATCH | Disable Automounting] *****
skipping: [localhost]

TASK [Ubuntu1804-CIS : NOTSCORED | 1.2.1 | PATCH | Ensure package manager repositories are configured] ***
ok: [localhost]

TASK [Ubuntu1804-CIS : NOTSCORED | 1.2.2 | PATCH | Ensure GPG keys are configured] *****
ok: [localhost]

TASK [Ubuntu1804-CIS : SCORED | 1.3.1 | PATCH | Ensure AIDE is installed (install nullmailer instead of postfix)] ***
skipping: [localhost]

TASK [Ubuntu1804-CIS : SCORED | 1.3.1 | PATCH | Ensure AIDE is installed] *****
```

(NOTA: Como ya comentamos, este perfil **desinstala xfce4**, ya que lo considera una vulnerabilidad potencial. Si estabas dentro del GUI, debes hacer **reboot** a la máquina inmediatamente después de que termine, ya que probablemente se colgará. Puedes restaurar **xfce4** de nuevo de esta forma. No obstante se recomienda restaurar una instantánea / clon para evitar problemas futuros:

- Ejecutar `sudo apt install --reinstall xfce4`
- Reiniciar y reinstalar **firefox**)

Después de este procedimiento de *hardening* automatizado, debes auditar el perfil CIS con *OpenSCAP* nuevamente para ver si cambian sus resultados. Además, debes verificar la seguridad del sistema con *Lynis* nuevamente para ver si el índice es más alto.

Resultados esperados: Esta actividad finalizará cuando puedas mejorar la seguridad de un sistema *Ubuntu* utilizando este *script Ansible* para implementar una gran cantidad de controles de seguridad CIS automáticamente y evaluar los resultados del proceso.

El Proyecto *Egida* de hardening CIS automatizado con *Ansible*

Aplicación práctica: Necesitas hacer hardening automático de tu SO Ubuntu siguiendo las políticas CIS usando un script de tercero cuyo autor es el egresado de esta escuela Antonio Payá

Este proyecto de *hardening* automatizado es similar al anterior, ya que consiste en aplicar *CIS Benchmark* a través de *Ansible*. **Crea otra instantánea de la máquina antes de ejecutar este software.** La ventaja que tiene (aparte de estar hecho en nuestra escuela 😊) es que está diseñado para ser más granular en el futuro, y presenta un menú y una configuración simple para que se aplique a múltiples máquinas al mismo tiempo. Para el propósito de nuestro curso, simplemente sigue las instrucciones del repositorio oficial: <https://github.com/Egida-Kassandra/egida/blob/master/docs/index.md#installation> para su instalación y uso solo en la máquina local (**localhost**). Si sigues las instrucciones y ejecutas el menú (**sudo egida menu -c local**, con opciones por defecto) para que se ejecuten todos los *benchmarks* CIS, obtendremos un resultado como este:

```
RUNNING HANDLER [egida-role-cis : sysctl flush ipv4 route table] *****
changed: [localhost]

RUNNING HANDLER [egida-role-cis : sysctl flush ipv6 route table] *****
changed: [localhost]

RUNNING HANDLER [egida-role-cis : restart auditd] *****
changed: [localhost]

RUNNING HANDLER [egida-role-cis : load audit rules] *****
changed: [localhost]

RUNNING HANDLER [egida-role-cis : update grub] *****
changed: [localhost]

RUNNING HANDLER [egida-role-cis : restart sshd] *****
changed: [localhost]

PLAY RECAP *****
localhost      : ok=219  changed=126  unreachable=0    failed=0    s
kipped=20     rescued=0    ignored=1
```

Al igual que con el *benchmark* anterior, la máquina se queda sin acceso a la GUI, ya que se considera una vulnerabilidad (se puede restaurar con el mismo procedimiento que vimos).

Después de ejecutar *Egida*, es necesario reevaluar el sistema nuevamente con *OpenSCAP* y *Lynis* para ver si el índice de *hardening* final ha mejorado. También puedes perder la conectividad a Internet en la máquina. **No intentes solucionarlo si sucede, simplemente vuelve a la instantánea anterior.**
















Resultados esperados: Esta actividad finalizará cuando puedas mejorar la seguridad de un sistema *Ubuntu* utilizando este script de *Ansible* para implementar una gran cantidad de controles de seguridad CIS de forma automática y evaluar los resultados del proceso, analizando las ventajas y desventajas que tienen estos *scripts* automatizados.






INSIGNIAS Y AUTOEVALUACIÓN



NOTA: Tienes una versión de esta tabla de insignias en formato editable disponible en el *Campus Virtual*. Puedes usar este archivo para crear un documento en el formato que desees y tomar notas extendidas de tus actividades para crear un log de lo que has hecho. Recuerda que el material que elabores se puede llevar a los exámenes de laboratorio.

Nivel de Insignia	Desbloqueada cuando	¿Desbloqueada?
	Puedes utilizar los perfiles de seguridad disponibles de <i>OpenSCAP</i> para cualquier producto soportado	
	Puedes cargar un perfil de seguridad adecuado para un sistema operativo soportado	
	Puedes utilizar <i>OpenSCAP</i> para intentar solucionar automáticamente posibles problemas en la aplicación de los controles de seguridad. Puedes responder a esta pregunta. <i>Incluso si no mejora significativamente el índice de seguridad de una máquina recién instalada, ¿qué crees que sucedería con una máquina que no se ha mantenido o administrado correctamente?</i>	
	Puedes utilizar los archivos de perfiles de seguridad de <i>OpenSCAP</i>	
	Puede responder a las siguientes preguntas: <i>¿Qué sucede si un control de seguridad aparece en un informe como "no aplicable"? ¿Cuáles son las posibles razones? ¿Sigue siendo útil este control de seguridad para mejorar la seguridad del sistema?</i>	
	Puedes realizar un análisis automatizado de un sistema operativo para detectar vulnerabilidades de acuerdo con un determinado perfil de seguridad.	
	Puedes utilizar oscap desde la línea de comandos para analizar y corregir los controles de seguridad	
	Puedes responder a la siguiente pregunta: <i>¿El Ubuntu 18.04 STIG incluye perfiles correspondientes a todas las combinaciones posibles de MAC y niveles de confidencialidad que vimos en teoría?</i>	
	Puedes responder a las siguientes preguntas: <i>¿Es el Ubuntu 18.04 STIG utilizable con OpenSCAP para automatizar la verificación / corrección de controles de seguridad? Si su respuesta es no, ¿qué puedes hacer con la información que proporciona?</i>	
	Puedes utilizar los scripts de corrección bash de cualquier directiva de seguridad SCAP.	
	Puedes utilizar los scripts de corrección de <i>Ansible</i> de cualquier directiva de seguridad SCAP.	
	Puede ejecutar un procedimiento de corrección de los STIGs con la herramienta <i>Ansible</i>	
	Puedes analizar y comprender el contenido de un informe oscap	
	Comprendes las ventajas y desventajas de las herramientas de <i>Compliance as Code</i> , y cómo intentar alcanzar de esta forma altas puntuaciones de seguridad pueden tener efectos secundarios que puede impedir que las máquinas funcionen correctamente y requieran intervención manual	
	Puedes responder a esta pregunta: <i>¿qué método automatizado tiene mejores resultados de corrección en el perfil CIS de Ubuntu: corrección de <i>Oscap</i>, corrección de script <i>bash</i> o corrección de <i>Ansible</i>? ¿La mejora de la seguridad es significativa a partir de un sistema recién instalado?</i>	



	Puedes ejecutar y aplicar el script <i>Ansible</i> de <i>hardening</i> automatizado de <i>Florian Utz</i> basado en <i>CIS</i> , evaluando luego el aumento en el índice de <i>hardening</i> general del sistema.	
	Puedes ejecutar y aplicar el script <i>Ansible</i> de <i>hardening</i> automatizado <i>egida</i> basado en <i>CIS</i> , evaluando luego el aumento en el índice de <i>hardening</i> general del sistema.	
	This is the way: Tu dominio de los procedimientos de <i>hardening</i> automatizado te permite mejorar la seguridad de un sistema <i>Ubuntu</i> de acuerdo con varias políticas de seguridad que garantizan diferentes niveles de esta. Además, puedes evaluar automáticamente la seguridad de cualquier sistema <i>Ubuntu</i> . Con esto, estás preparado para usar los mismos procedimientos en cualquier otro sistema que admita la misma clase de automatización.	

