# Bloque 1: Administración de directivas OpenSCAP

## Ejercicio L6B1\_OSCAPUPD: Actualizando las políticas de seguridad de OpenSCAP

Descargar la última versión de las políticas

wget <https://github.com/ComplianceAsCode/content/releases/download/v0.1.66/scap-securityguide-0.1.66.zip>

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.

Para saber los perfiles disponibles para Ubuntu 18.04, por ejemplo, haz: sudo oscap info <carpeta descomprimida>/ssg-ubuntu1804-ds-1.2.xml

## Ejercicio L6B1\_OSCAPAUDITSW: Pasar un perfil OpenSCAP al navegador Firefox

Aplicaciones -> sistema -> scap workbench -> file -> open other content -> seleccionamos ssg-firefox-ds-1.2.xml... -> scan -> show report

## Ejercicio L6B1\_OSCAPAUDITSO: Pasar perfiles de OpenSCAP al SO

Aplicaciones -> sistema -> scap workbench -> file -> open other content -> seleccionamos la imagen de ubuntu18.04... -> scan -> show report

## Ejercicio L6B1\_OSCAPREPORT: Interpretación de informes OpenSCAP

Entender estos puntos:

• 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...)

## Ejercicio L6B1\_OSCAPCMD: Uso de OSCAP desde la línea de comandos

hacer dry run en la gui y sacar el comando de ahi, poniendo sudo delante

• 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 descomprimimos, 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)

# Bloque 2. Remediación con OpenSCAP

## Ejercicio L6B2\_OSCAPREMWB: Remediación con SCAP Workbench

sudo lynis audit system > lynisantes.txt

hacer dry run y remediate con los 3 perfiles anteriores

sudo lynis audit system > lynisdespues.txt

comparar el indice de hardening antes y después

## Ejercicio L6B2\_OSCAPREMBASH: Remediación con scripts de bash

sudo chmod +x archivo

sudo ./archivo

despues hacer lo mismo de antes, una auditoria con cada cosa y corregir con cis

## Ejercicio L6B2\_OSCAPREMANS: 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 .yml correspondiente al perfil CIS

• Edita este .yml y cambia hosts: all por hosts: localhost

• Ejecuta sudo ansible-playbook ubuntu1804-playbook-cis.yml

• Ejecuta una verificación de la política nuevamente con oscap y una auditoría con Lynis para ver si los resultados mejoran

# Bloque 3: Auditoría y remediación con STIGs

Instalar la herramienta

Localizar fichero .deb

sudo dpkg -i <ruta del fichero .deb>

sudo /opt/scc/scc

Sale la interfaz

Para probar la herramienta, marca solo el STIG de Ubuntu 20, aunque no sea la versión correcta de tu SO, y asegúrate de que la opción “Run content regardless of applicability” está marcada para que SCC escanee el SO igualmente.

Cuando lances el escaneo en el botón de la izquierda el sistema te advierte de que ese STIG tiene algunos controles que te hacen preguntas que debes responder manualmente, y como crear las respuestas. No lo vamos a hacer en esta prueba, por lo que podemos simplemente Continuar para escanear el sistema con todo lo que sea automatizable.

Espera a que termine la auditoría del sistema y se genere un informe. Para poder verlo desde la herramienta necesitas instalar otra herramienta aparte, pero no es necesario en nuestro caso. Simplemente busca en el directorio home del usuario actual, y abre el fichero HTML que encontrarás en SCC/Sessions/<fechayhorainforme>. Dentro del informe, accede a “All Settings”

## Ejercicio L6B3\_STIGSCCAUDIT: Auditoría automatizada con SCAP Compliance Checker (SCC) y STIGs

# Bloque 4: Auditoría y remediación con CIS Benchmarks

## Ejercicio L6B4\_CISUBUNTUPRO: La licencia Ubuntu Pro

## Ejercicio L6B4\_CISWAZUH: Wazuh como gestor de políticas CIS

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente