Auditoria de Seguridad en Windows
Práctica 11

1. Objetivo

En esta práctica el alumno debe aprender a utilizar las herramientas que permiten auditar la seguridad de un computador. Para ello, el alumno debe realizar 3 tareas:
1.-Activar y configurar los controles de seguridad.
2.-Activar y configurar la auditoría de seguridad, para controlar los eventos que se generan.
3.-Analizar los registros de eventos de seguridad, para evaluar problemas con la seguridad.

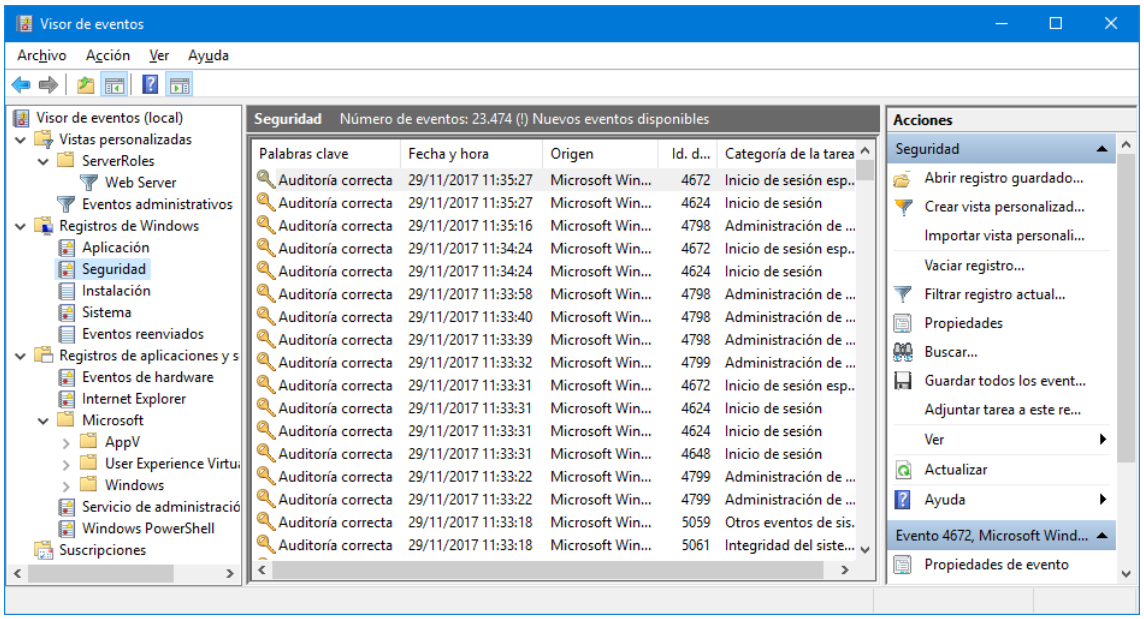
2. El visor de eventos de Windows

Permite analizar registros de eventos. Para arrancar el visor de eventos hacer:

Inicio > Panel de control > Herramientas administrativas > Visor de eventos

O teclear **eventvwr** (*event viewer*) en la consola.

Desplegar el árbol del visor en el panel izquierdo y al seleccionar "Seguridad" aparece:



El uso del visor es totalmente intuitivo. La parte interesante para la asignatura se centra en los Registros de Windows, y en particular en el Registro de Seguridad de Windows.

También pueden ser de interés algunos "Registros de aplicaciones y servicios". Por ejemplo en:

Reg app y serv > Microsoft > Windows > Windows Firewall With Advanced Security > Firewall

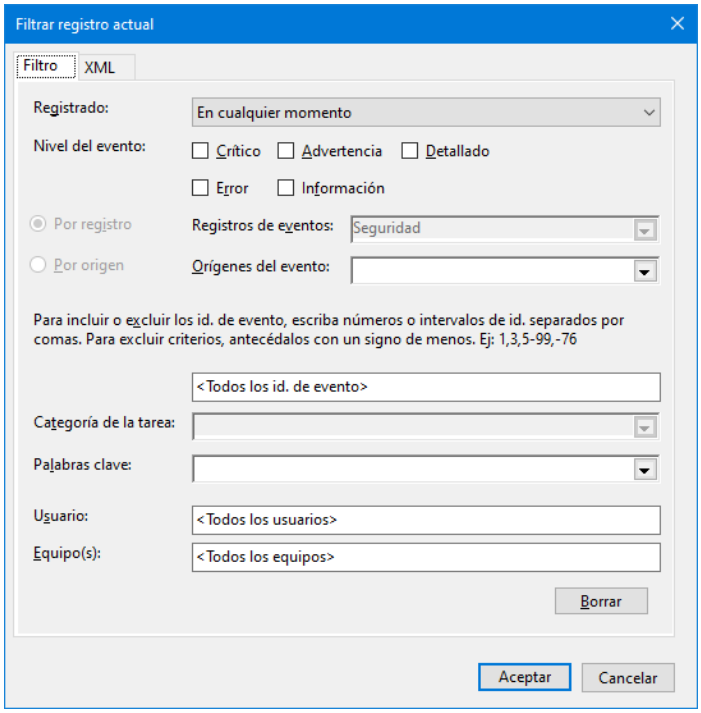
Observar los eventos que hay en el registro con las directivas de auditoría que tiene activadas el sistema por defecto. Buscar información en Internet sobre los códigos numéricos que aparecen.

Luego, realizar tareas como Buscar, Filtrar, Guardar, etc.

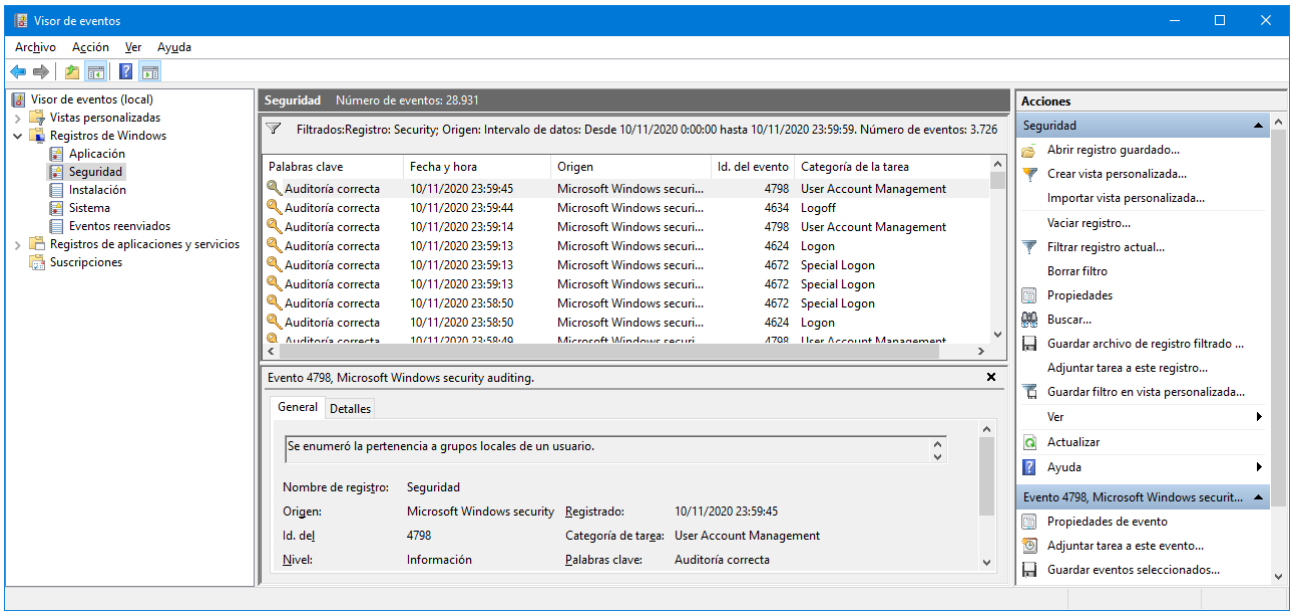
Ejemplo: guardar todos los eventos de seguridad de todo un día, por ejemplo de antes de ayer.

En el panel izquierdo seleccionar el Registro de Windows “Seguridad”. En el panel derecho de Acciones selecciona “Filtrar registro actual…”.

En la ventana emergente “Filtrar registro actual” usa el campo Registrado y cambia la opción “En cualquier momento” por la opción “Intervalo personalizado…”.



Tras el filtrado, el visor muestra una ventana como la siguiente. Debes ensanchar el panel central para poder ver el número de eventos en la esquina superior derecha del panel.

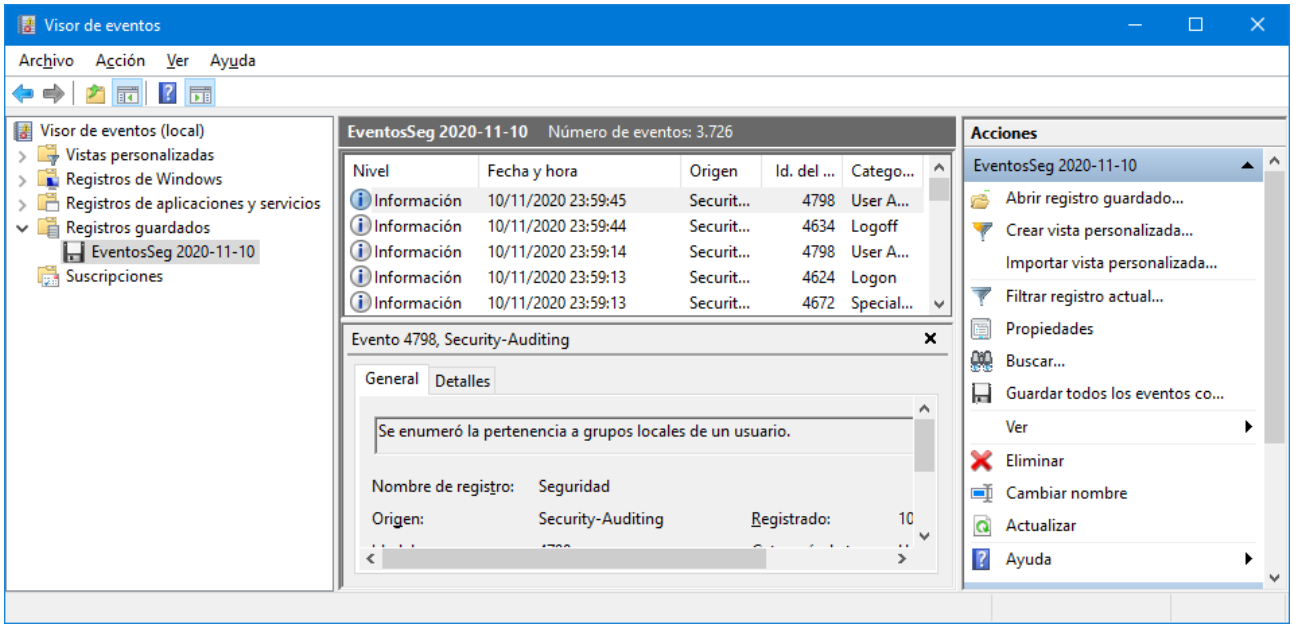


Selecciona la Acción “Guardar archivo de registro filtrado…” y guarda los eventos.

Ahora debes cerrar el Visor de eventos.

Vete al directorio donde esté el fichero con los eventos guardados (.evtx) y haz doble clic sobre él.

Se abre el Visor de eventos mostrando en el panel izquierdo un nuevo elemento: Registros guardados, que contiene el fichero seleccionado. Además, los eventos del fichero se muestran en el panel central tal como se muestra en la figura siguiente:



Borra “EventosSeg 2020-11-10” en el panel izquierdo del Visor de eventos. Comprueba que el Visor ya no dispone de los eventos, pero que el fichero continua en el directorio en el que estaba almacenado.

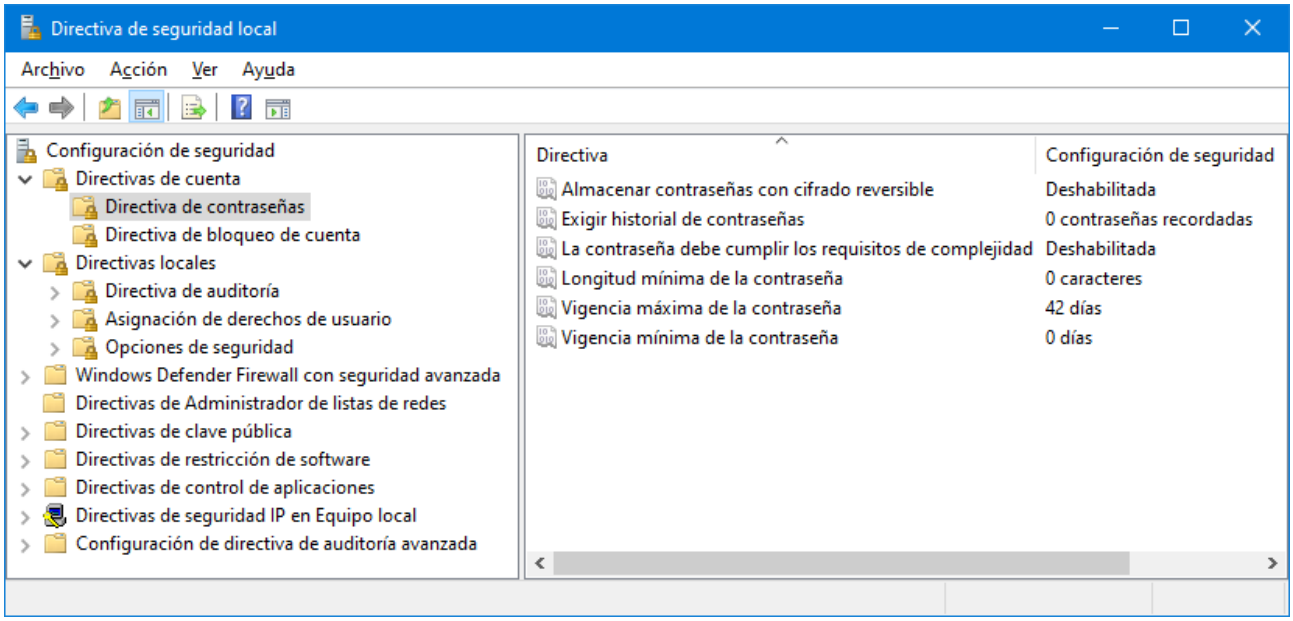
El elemento Registros guardados del Visor de eventos funciona como una tabla de acceso rápido a los ficheros con los que ha trabajado el Visor.

3. Configurar controles de seguridad

En un sistema operativo, como Windows, hay muchos controles de seguridad que se pueden activar/desactivar y configurar. Una de las herramientas que permite configurar varios controles de seguridad es la "Directiva de seguridad local" (secpol). Para usar esta herramienta hacer:

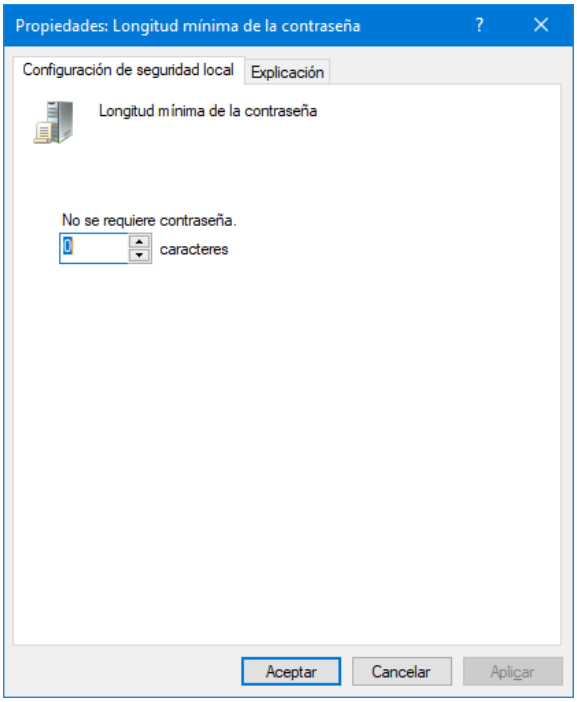
Panel de control > Herramientas administrativas > Directiva de seguridad local

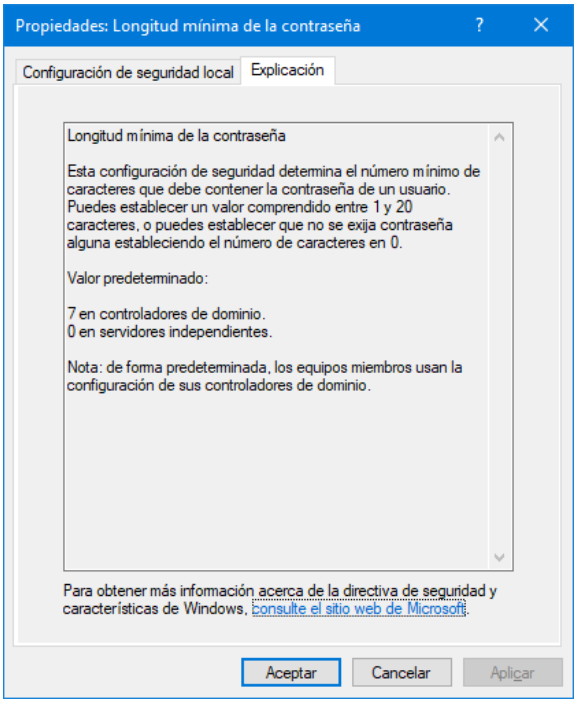
O teclear **secpol** (*security policy*) en la consola. Se despliega la siguiente ventana:



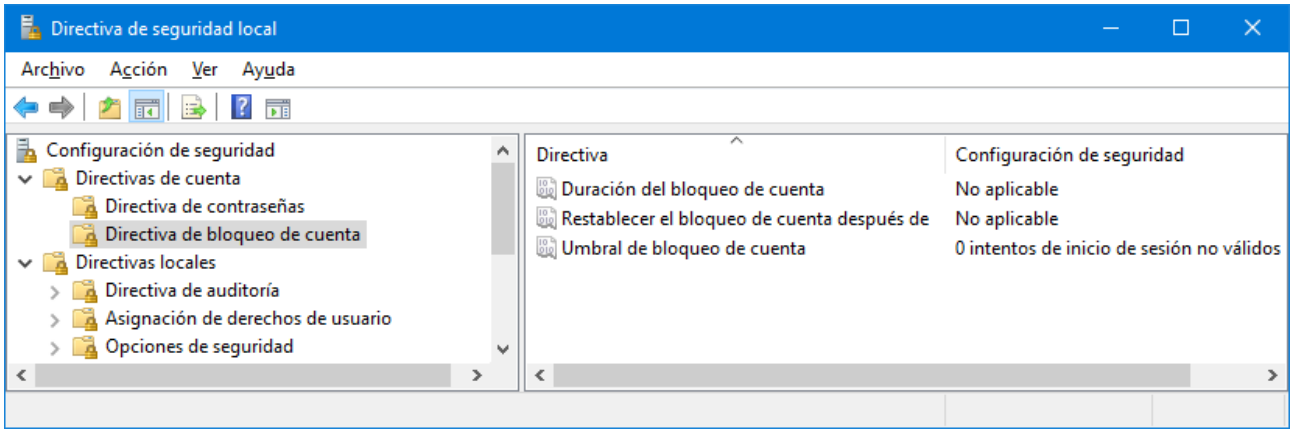
El panel de la izquierda contiene múltiples opciones de "Configuración de seguridad". En la ventana anterior se ha seleccionado la opción "Directivas de cuenta > Directiva de contraseñas". En el panel derecho se puede observar las directivas y su configuración.

Por ejemplo la directiva "Longitud mínima de la contraseña" está configurada a 0 caracteres. Haz doble clic en la directiva para cambiar su configuración. Aparecen estas ventanas:





Observar la “Directiva de bloqueo de cuenta”:



Observar las opciones que ofrece y aprender a configurar el bloqueo. Comprobar que las opciones configuradas funcionan realmente.

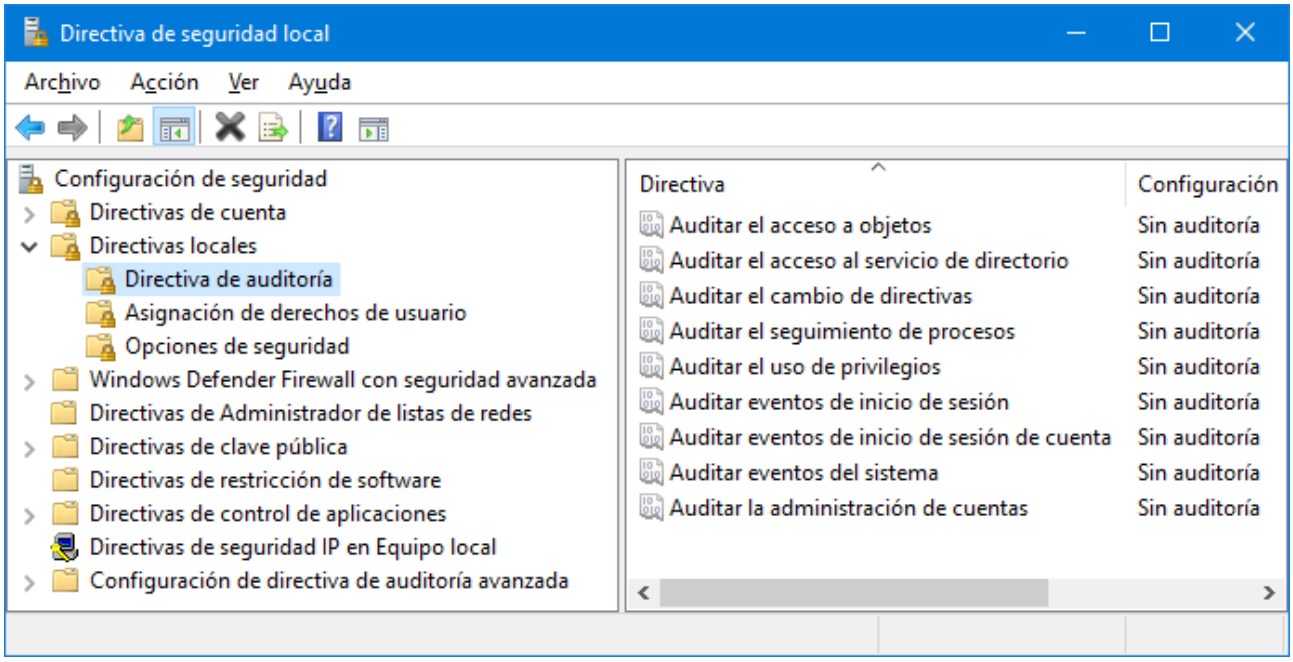
En la opción del panel izquierdo “Directivas locales” observar las directivas que se pueden configurar en “Asignación de derechos de usuario” y en “Opciones de seguridad”.

Comprueba que en la opción “Windows Defender Firewall con seguridad avanzada” da acceso a la configuración de Firewall de Windows, pero la configuración se puede hacer en el propio Firewall.

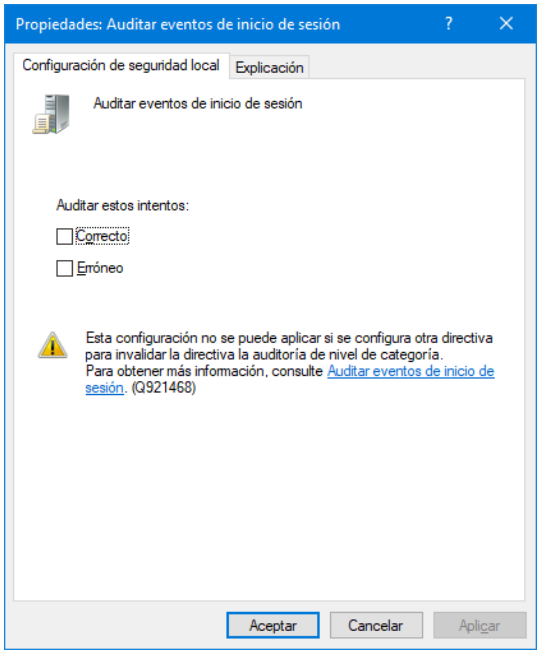
Observa el resto de tipos de Directivas que permite configurar la herramienta, para tener una idea de lo que permite configurar la herramienta secpol. Para usarlas correctamente, primero hay que conocer la tecnología con la que trabajan.

4. Configurar los eventos de auditoría que se deben registrar

Una forma de establecer los eventos de auditoría a registrar es mediante la herramienta secpol. Para la auditoría es interesante, dentro de **Directivas locales**, la opción "**Directiva de auditoría**". También es interesante el último grupo "**Configuración de directiva de auditoría avanzada**".



Observar en la figura anterior que la "Directiva de auditoría" permite auditar las categorías que se muestran en el panel derecho. Si se selecciona una directiva cualquiera del panel derecho aparece una ventana como la siguiente:



Observar que tenemos 4 opciones:

1) No auditar los inicios de sesión, dejando sin seleccionar ambas casillas.

2) Auditar solo los inicios de sesión correctos, seleccionando solo la casilla "Correcto".

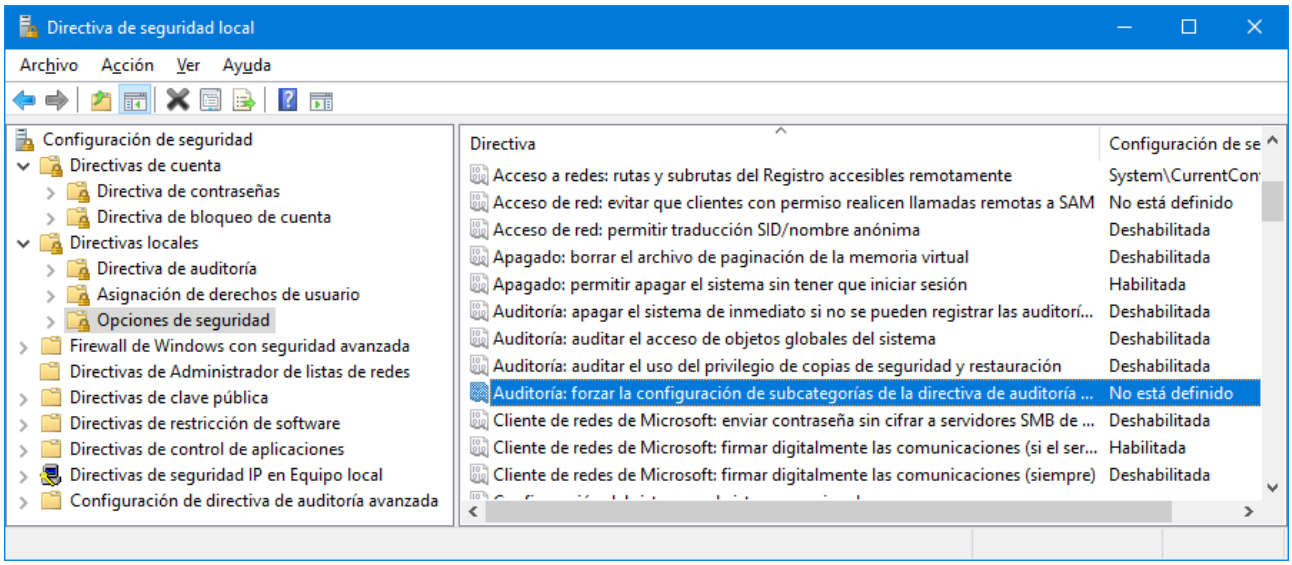
3) Auditar solo los inicios de sesión erróneos, seleccionando solo la casilla "Erróneo".

4) Auditar los inicios de sesión correctos y los erróneos, seleccionando ambas casillas.

Para las nueve directivas de auditoría se puede seleccionar las cuatro opciones anteriores. Para disponer de mayor detalle en la selección de los eventos a auditar es necesario utilizar la "Configuración de directiva de auditoría avanzada".

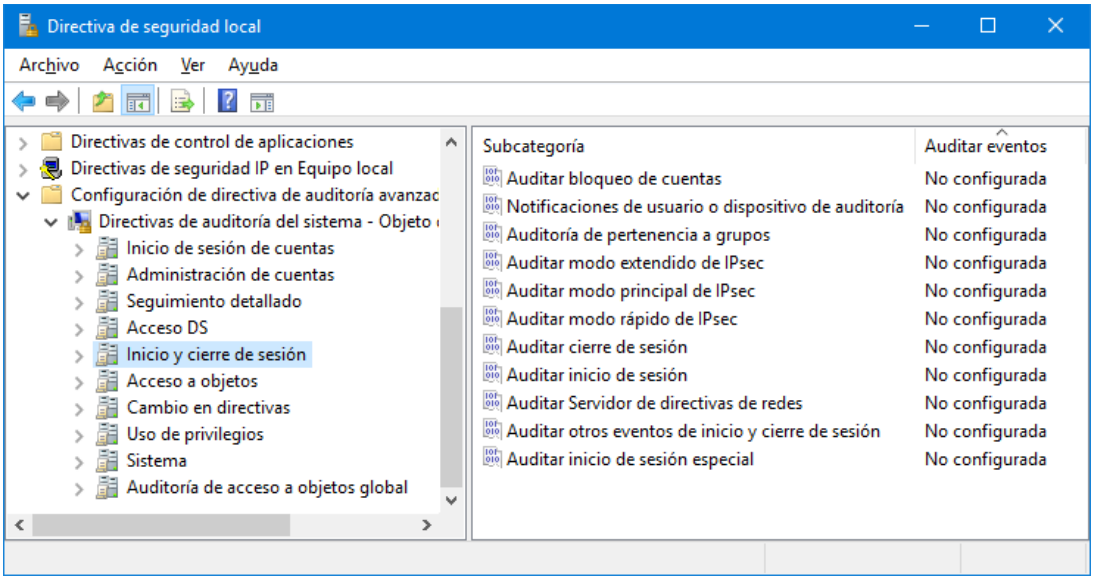
En las prácticas y trabajos de la asignatura de seguridad usar la Auditoría Avanzada.

Para activarla hay que ir primero a "Opciones de seguridad" en el panel izquierdo y luego en el panel derecho seleccionar la directiva "Auditoría: forzar la configuración de subcategorías ..." como muestra la figura siguiente:

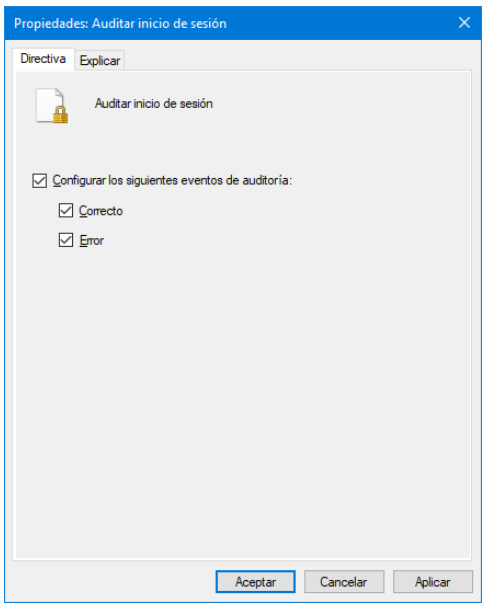


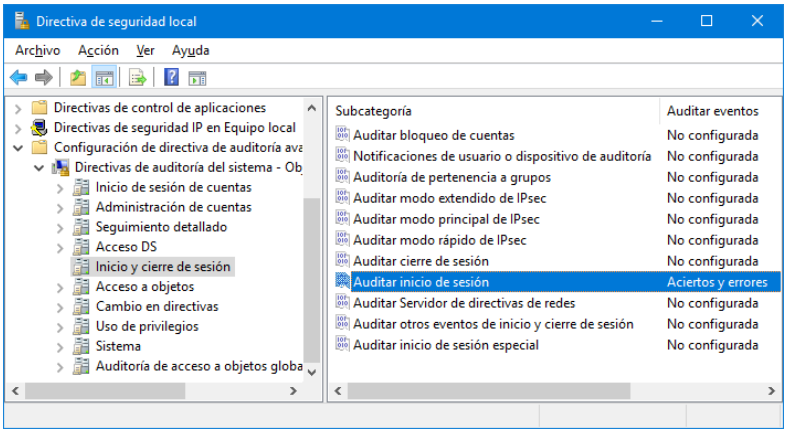
Hacer doble clic sobre la directiva y en la ventana que se muestra, habilitarla.

Después desplegamos la última opción del panel izquierdo. Observar que en el panel izquierdo tenemos las mismas directivas (o categorías) de auditoría que aparecían antes en el panel derecho. Pero ahora al seleccionar una en el panel izquierdo aparecen en el panel derecho sus subcategorías.



Ahora, el Inicio y cierre de sesión permite auditar nueve eventos independientes. Por ejemplo, seleccionar en el panel derecho "Auditar inicio de sesión" y auditar tanto los inicios correctos como los intentos de inicio que generan un error.



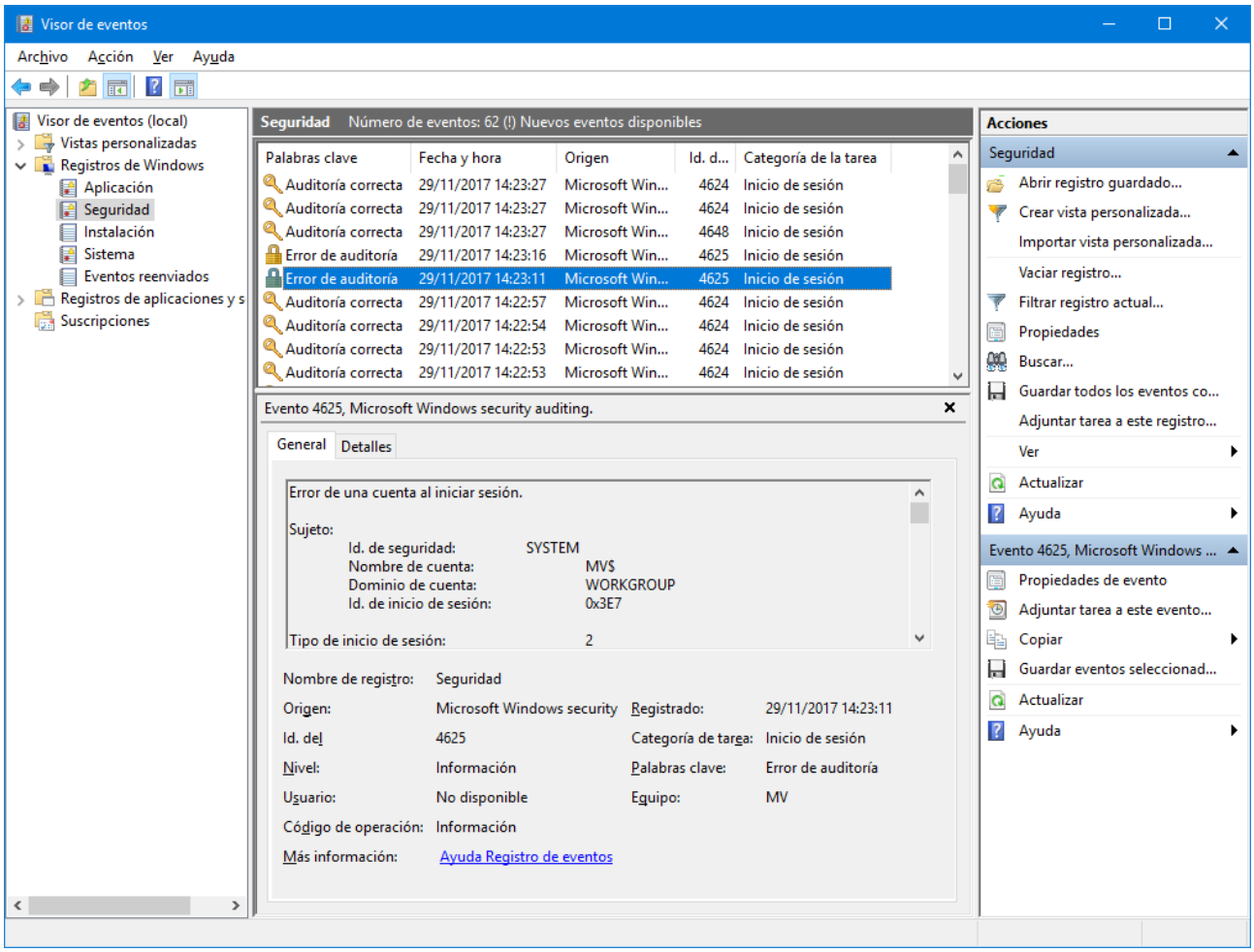


Comprobar que el SO está realizando la captura de eventos de auditoría del inicio de sesión.

En el Visor de eventos Vaciar el registro de eventos de seguridad.

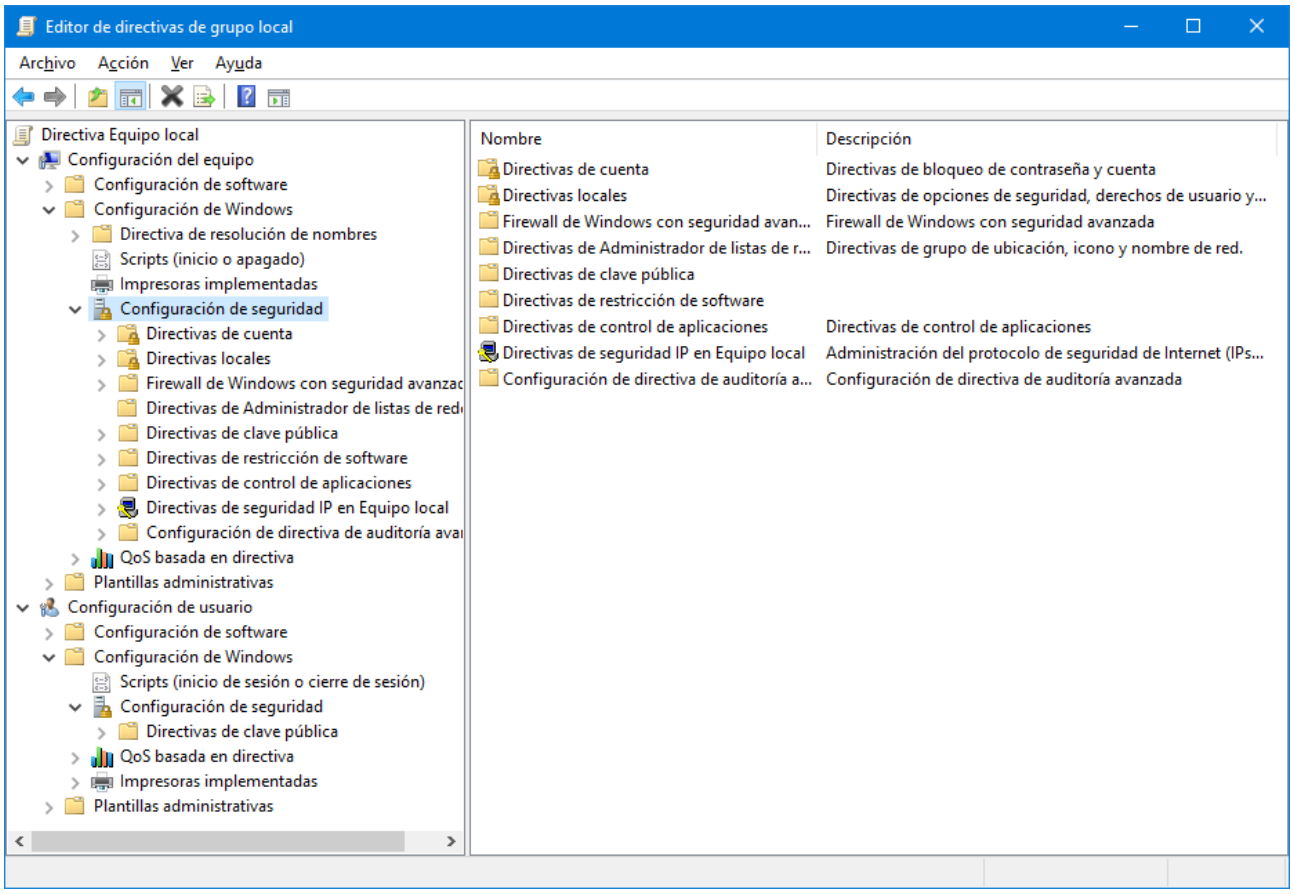
Cerrar la sesión en Windows, y hacer dos intentos de logon con contraseña incorrecta. Al tercer intento usar la contraseña correcta para a acceder nuevamente al sistema.

Usar el Visor de eventos para ver lo que ha ocurrido:



Otra forma de acceder a esta funcionalidad es usando el Editor de directivas de grupo local.

Ejecutar gpedit.msc en una consola y aparece la ventana siguiente:



Como se puede comprobar dentro de toda la "Configuración del equipo" tenemos una sección dedicada a la "Configuración de seguridad".

Uso de la herramienta Auditpol.exe

Auditpol /? → Muestra los comandos disponibles

Auditpol /list /? → Muestra ayuda sobre las opciones del comando /list

Auditpol /list /subcategory:* → Muestra toda las categorías y subcategorías disponibles, que es necesario conocer con exactitud para luego usarlas en los comandos.

Auditpol /get /subcategory:"Cambio de la directiva del nivel de reglas de MPSSVC"

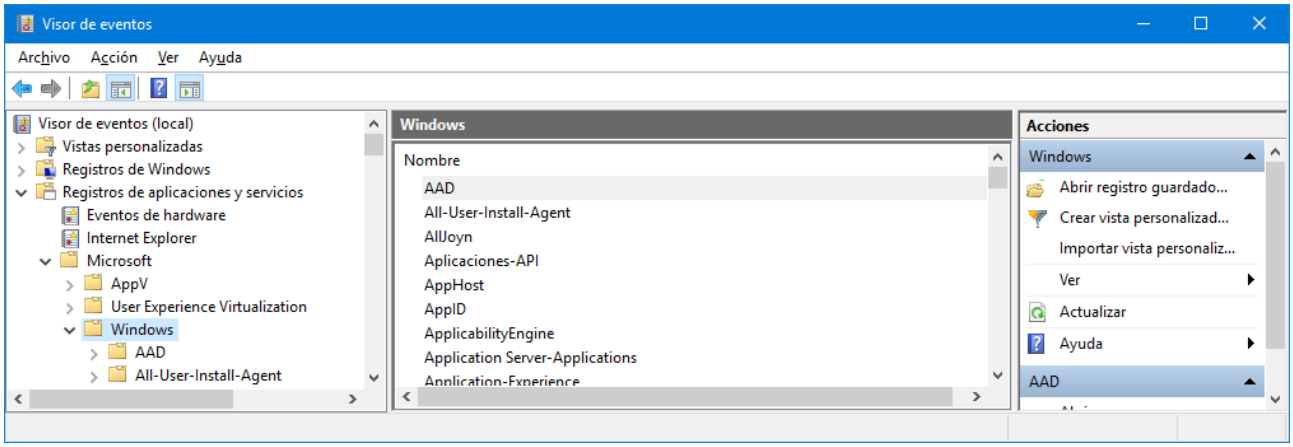
```
Directiva de auditoría del sistema
Categoría o subcategoría            Configuración
Cambio de plan
  Cambio de la directiva del nivel de reglas de MPSSVCSin auditoría
```

Observar la respuesta Sin auditoría pegada al texto descriptivo previo.

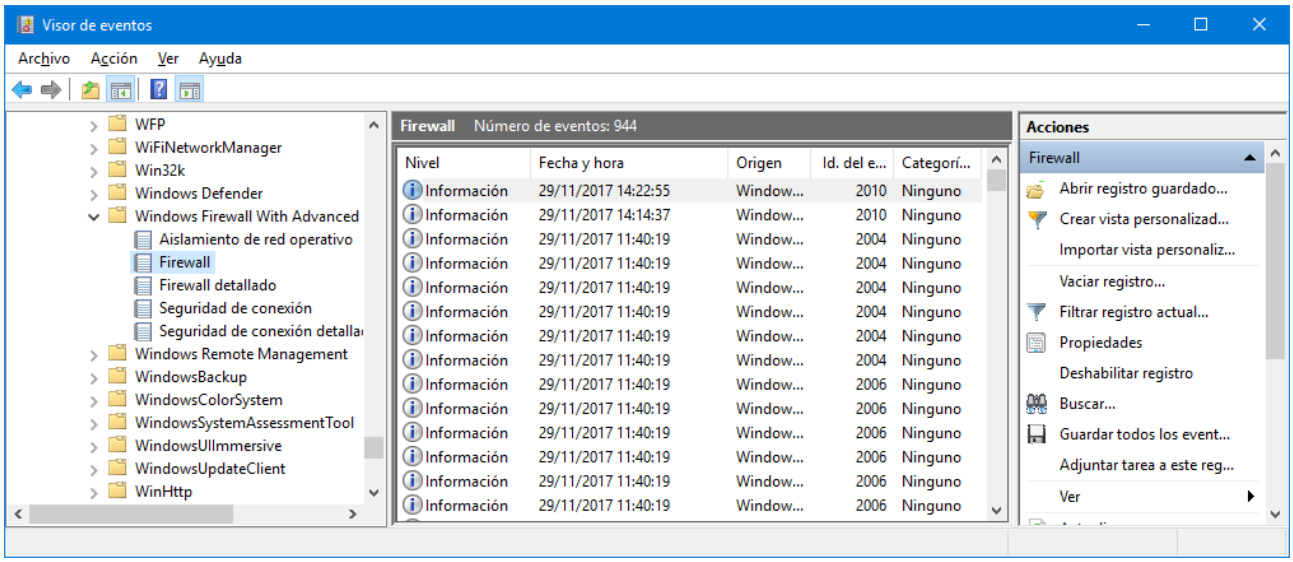
CONCLUSIÓN: Es mejor usar las herramientas secpol.msc o gpedit.msc

5. Auditar el cortafuegos

Para ver los eventos que genera el Firewall en el Visor de Eventos hay que navegar en el panel izquierdo del visor: Registros de aplicaciones y servicios > Microsoft > Windows >



Y seguimos ... > Windows Firewall With Advanced Security

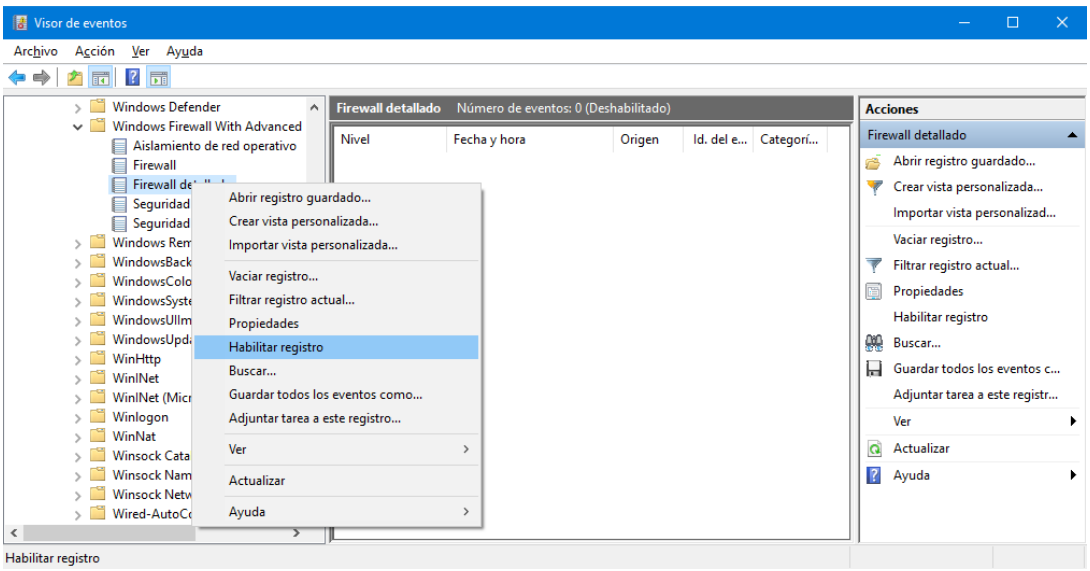


Se puede ver que hay 5 registros de eventos disponibles.

El registro "Firewall" contiene los eventos relacionados con la configuración del Firewall. Se añade un evento cada vez que se añade, quita o modifica una regla, o cuando se cambia el perfil de una interfaz de red. Compruébalo analizando unos cuantos eventos que aparecen en el panel central del visor de eventos.

El registro "Firewall detallado" contiene los eventos relacionados con el estado operativo del Firewall. Por defecto este registro esta deshabilitado. Para activarlo hacer clic en el botón derecho del ratón y seleccionar "Habilitar registro" en el menú contextual que aparece. También se puede habilitar en el panel derecho de acciones. Observar que el menú contextual que aparece contiene exactamente las mismas opciones que el panel derecho de acciones.

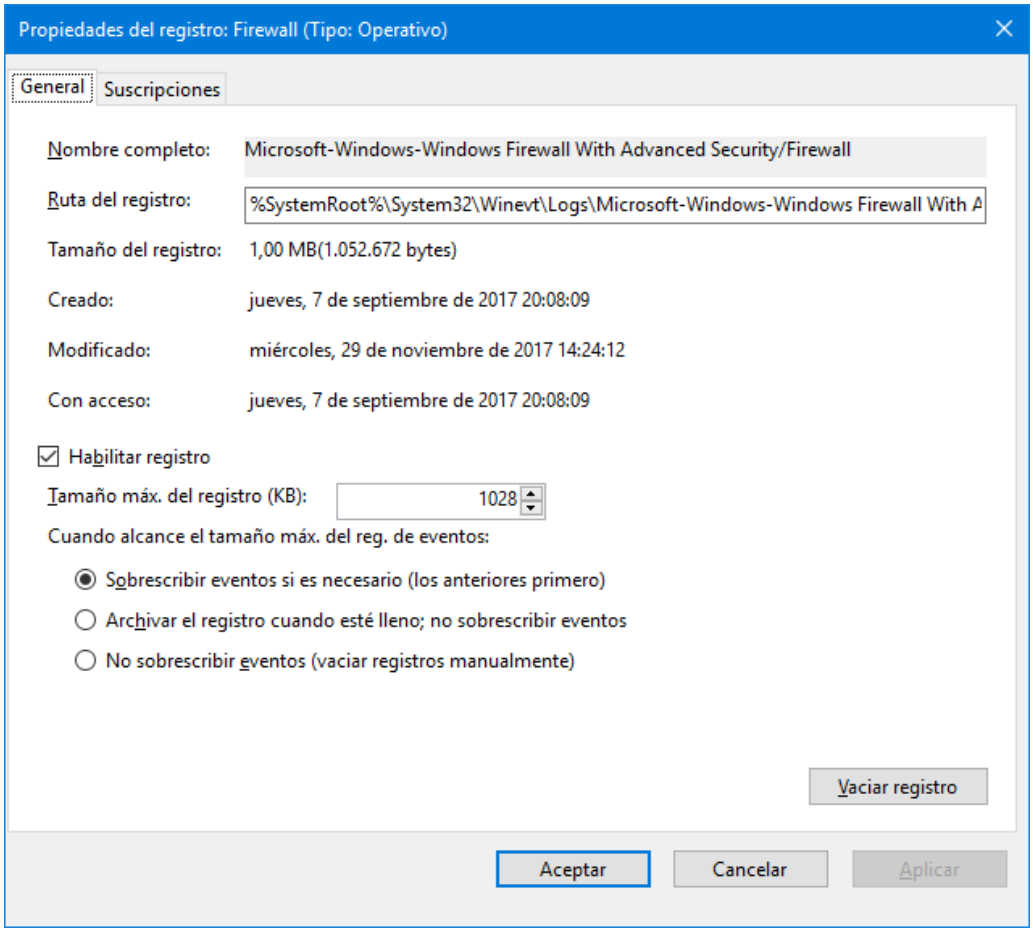
Parece que este registro no es muy útil, pues casi todos los cambios relativos al funcionamiento del Firewall se almacenan en el registro "Firewall".



El registro "Seguridad de conexión" contiene los eventos relacionados con la configuración de las reglas y los parámetros de IPsec.

El registro "Seguridad de conexión detallada" contiene los eventos relacionados con el funcionamiento de IPsec.

Selecciona el registro "Firewall" en el panel izquierdo. En el panel derecho de acciones selecciona la opción "Propiedades" y aparece la ventana siguiente:



Observar el nombre del registro y su ubicación en el sistema de ficheros.

Análisis de los eventos del registro

Utiliza el visor de eventos para ver los eventos del registro:
Registros de aplicaciones y servicios > Microsoft > Windows >
> Windows Firewall With Advanced Security / Firewall

Se puede comprobar que existen eventos con los siguientes IDs:

ID	Descripción del evento
2003	Cambió una configuración de Firewall de Windows Defender en el perfil Público.
2006	Se eliminó una regla en la lista de excepciones de Firewall de Windows Defender.
2010	Cambió el perfil de red en una interfaz.
2033	Se eliminaron todas las reglas de la configuración de Firewall de Windows Defender en este equipo.
2051	Actualización de directiva de restricciones de inquilino.
2097	Se ha agregado una regla a la lista de excepciones del Firewall de Windows Defender.
2099	Se modificó una regla en la lista de excepciones de Firewall de Windows Defender.

La mayoría de los eventos están relacionados con las **reglas** del Firewall de Windows: agregación (2097), modificación (2099) o eliminación (2006, 2033).

Pero también se puede comprobar que el evento 2003 se utiliza para registrar una gran variedad de cambios en la configuración del Firewall.

6. Realización de ejercicios de auditoría

Los aspectos mostrados en esta práctica son solo un ejemplo introductorio de las posibilidades de auditoría disponibles en el sistema operativo Windows 10. Las posibilidades reales deben ser exploradas con más detalle por cada alumno realizando ejercicios de auditoría.

Los ejercicios deben realizarse en la Máquina Virtual utilizada en las prácticas y los pasos que se van realizando se deben documentar imprimiendo ventanas (Alt+ImprPant) y copiándolas en un documento abierto en la MV con el programa WordPad, junto con un mínimo texto explicativo. Es muy conveniente guardar frecuentemente el documento en el formato "Documento XML abierto de Office (*.docx)".

Si se pide un ejercicio de auditoría en el examen de prácticas habrá que entregar un documento de este tipo en el Campus Virtual.

Generalmente, en los ejercicios de auditoría hay que realizar 4 tareas secuenciales:

1.-Activar y configurar los controles de seguridad.
Utilizar la herramienta secpol, el Firewall, o incluso propiedades del sistema de ficheros, como la concesión de permisos de acceso. Muchos de los controles que se pueden utilizar ya estarán activados y tendrán una configuración por defecto que aplica el propio sistema.

2.-Activar y configurar la auditoría de seguridad, para controlar los eventos que se generan.
El objetivo suele ser recopilar información sobre cuatro aspectos:

- Cuando se activa y se desactiva un control, por ejemplo el Firewall.
- Cuando se cambia la configuración de un control, por ejemplo las reglas del Firewall.
- Cuando el control detecta una violación de seguridad y cual, por ejemplo una denegación de acceso. Esto permite analizar los ataques que han fracasado.
- También se puede recabar información de la ausencia de violaciones de seguridad, por ejemplo todos los accesos que permite el Firewall. El volumen de información a tratar aumenta muchísimo. Pero esto permite analizar los ataques que han tenido éxito.

La herramienta fundamental para realizar esta tarea es secpol, usando la "Configuración de directiva de auditoría avanzada".

3.-Realizar pruebas para generar eventos.
En esta tarea los alumnos deben realizar algunas pruebas para generar eventos de auditoría. Por ejemplo, acceder al sistema dando contraseñas erróneas varias veces y finamente volviendo a entrar con la contraseña correcta. También pueden acceder a archivos cuyo acceso este auditado. Y por supuesto, pueden intentar escanear el computador usando cualquier herramienta de red, como por ejemplo Nmap o Nping.

4.-Analizar los registros de eventos de seguridad, para evaluar problemas con la seguridad.
Utilizar el Visor de eventos para analizar los eventos capturados. Se puede vaciar el registro para tener unos pocos eventos y localizarlos rápidamente, pero **es más realista no hacerlo**. Diseñar algún filtro o alguna consulta para localizar algún tipo de evento en particular y documentarlo. Finalmente, volcar los archivos de eventos para su análisis, utilizando alguna herramienta de análisis de registros o realizando algún tipo de programa que procese eventos.

7. Programa para la lectura y análisis de eventos

Crea una **solución de Visual Studio** para leer eventos y realizar un análisis forense posterior.

Observa en el visor de ayuda el espacio de nombres System.Diagnostics.Eventing.Reader que proporciona clases para leer y gestionar registros de eventos.

LECTURA DE EVENTOS DE UN FICHERO

Crea el método estático **LeeFicheroEventos()** que recibe un string con el nombre del fichero con eventos y devuelve una lista de eventos de la clase **List<EventRecord>**.

Para leer los eventos, declara el objeto **LectorEventos** de la clase **EventLogReader** indicando en el primer parámetro del constructor el nombre del fichero con los eventos y en el segundo parámetro el tipo del primer parámetro, que en este caso es **PathType.FilePath**.

Para almacenar los eventos leídos, declara el objeto **ListaEventos** de la clase **List<EventRecord>**.

Utiliza un bucle while para leer los eventos, leyendo en la propia cláusula del while un objeto de la clase **EventRecord** con método **ReadEvent()** de **LectorEventos**, mientras que el objeto devuelto no sea igual a **null**. Añade cada evento leído a **ListaEventos**.

Finalmente, retorna el objeto **ListaEventos**.

En el método Main() declara una Lista de Eventos Inicial de la clase **List<EventRecord>** y asígnale la lista devuelta por el método LeeFicheroEventos("Fichero.evtx").

Como comprobación mínima del funcionamiento correcto, escribe en la consola el número de elementos de la Lista de Eventos Inicial. Abre el fichero con **eventvwr** y comprueba que el número de eventos indicado es el mismo.

LECTURA DE EVENTOS DE UN REGISTRO DE WINDOWS

Crea el método estático **LeeRegistroEventos()** que recibe un string con el nombre del registro con eventos y devuelve una lista de eventos de la clase **List<EventRecord>**.

Para leer los eventos, declara el objeto **LectorEventos** de la clase **EventLogReader** indicando en el primer parámetro del constructor el nombre del registro con los eventos y en el segundo parámetro el tipo del primer parámetro, que en este caso es **PathType.LogName**.

Para almacenar los eventos leídos, declara el objeto **ListaEventos** de la clase **List<EventRecord>**.

Utiliza un bucle while para leer los eventos, leyendo en la propia cláusula del while un objeto de la clase **EventRecord** con método **ReadEvent()** de **LectorEventos**, mientras que el objeto devuelto no sea igual a **null**. Añade cada evento leído a **ListaEventos**.

Finalmente, retorna el objeto **ListaEventos**.

En el método Main() declara una Lista de Eventos Inicial de la clase **List<EventRecord>** y asígnale la lista devuelta por el método LeeRegistroEventos("Security").

Usa **eventvwr** para comprobar que el número de eventos leídos por el método coincide con los que hay en el registro Security de Windows.

NOTA: Para leer el registro de Seguridad de Windows, el programa debe ejecutarse en modo Administrador para que no genere una excepción.

FILTRADO DE EVENTOS

Para realizar un análisis basado en eventos, puede ser conveniente extraer de la lista con todos los eventos leídos, los eventos que sean de interés para el análisis a realizar. Por ejemplo, se puede generar una nueva lista que solo contenga eventos con unos determinados IDs.

Crea el método estático **FiltroIDs()** que recibe **ListaEnt**, una lista de eventos de la clase **List<EventRecord>,** y **ListaID**, un vector con los IDs de interés en un array de long. El método devuelve una lista de eventos de la clase **List<EventRecord>** con los eventos de interés.

Para almacenar los eventos de salida, declara el objeto **ListaSal** de la clase **List<EventRecord>**.

Usa un bucle for para recorrer los eventos de **ListaEnt**, y para cada evento usa otro bucle for para recorrer los IDs de **ListaID**. Si el ID del evento coincide con un ID de **ListaID** entonces añade el evento a **ListaSal**. Una vez añadido el evento, no es necesario seguir recorriendo el bucle for comprobando el resto de IDs. Utiliza una sentencia break para salir del bucle.

Finalmente, el método devuelve **ListaSal**.

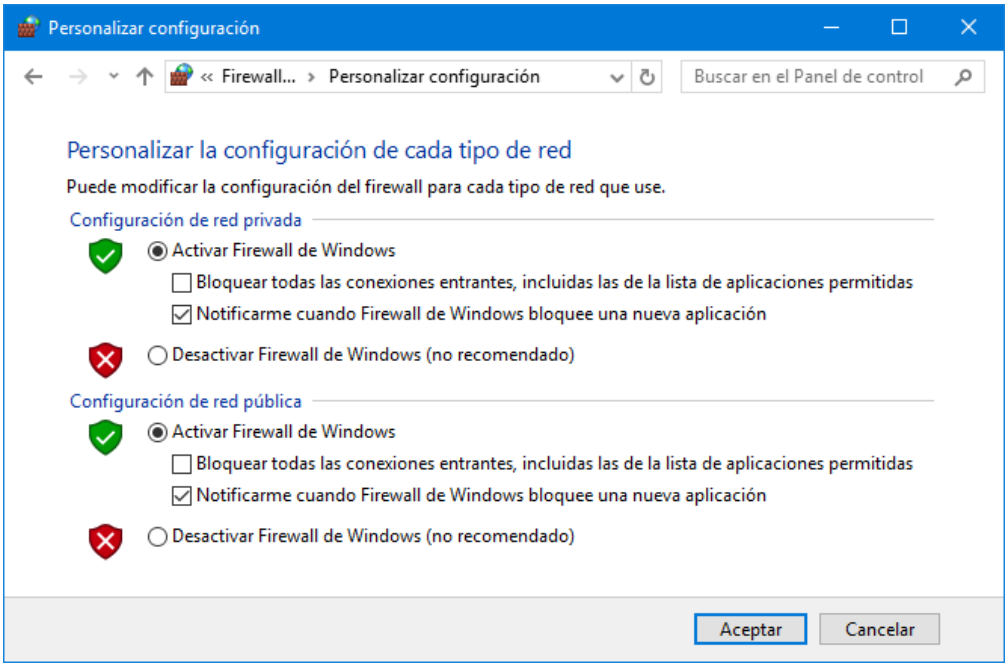
8. Ejercicio: Analizar los períodos de desactivación del firewall

El objetivo de este ejercicio es analizar los periodos en los que el Firewall de Windows ha estado desactivado. Para ello se realiza un pequeño programa que analiza los eventos necesarios.

Recordar que el FW de Windows trabaja con 3 perfiles diferentes (Público, Privado y de Dominio) en función del tipo de red a la que se conecte el computador. El análisis se debe realizar para uno cualquiera de los perfiles, independientemente de los otros dos.

A continuación se desarrollan las 4 tareas secuenciales de estos ejercicios

1.-Activar y configurar los controles de seguridad.
En este caso hay que hacer una secuencia de desactivaciones y activaciones del FW usando la ventana de configuración del propio FW:



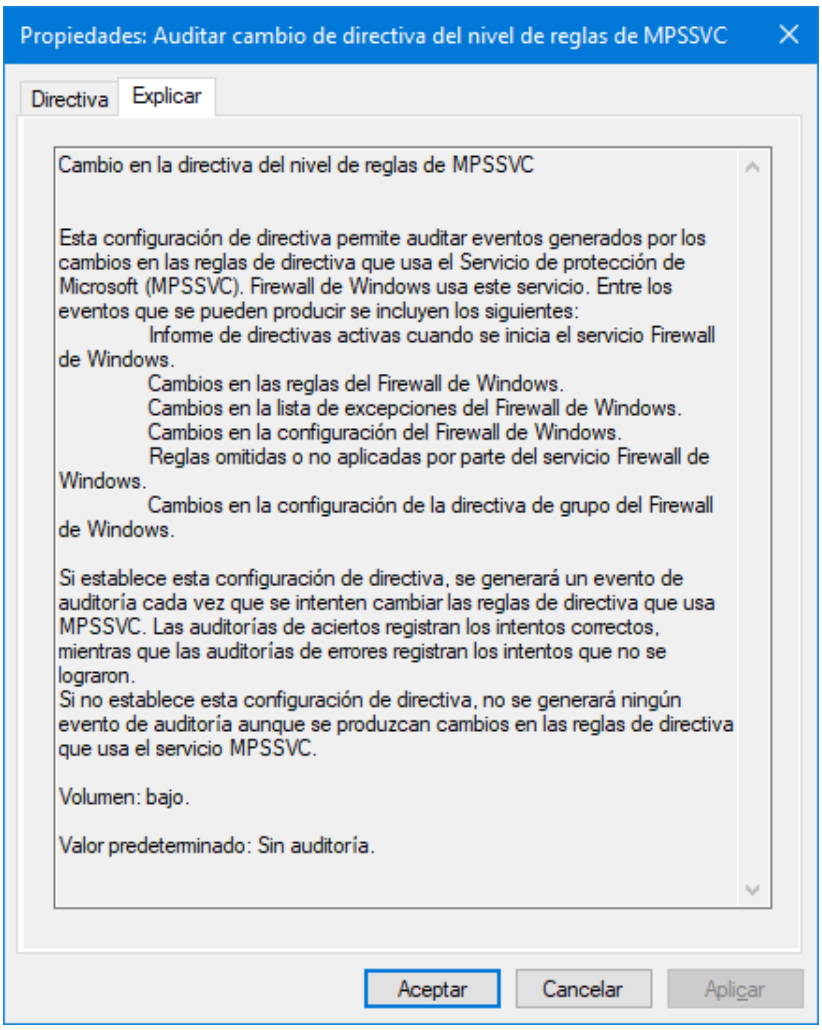
2.-Activar y configurar la auditoría de seguridad.
Para auditar los cambios en el estado del Firewall se puede usar la directiva “Audit MPSSVC Rule-Level Policy Change”.

En la ventana de SECPOL, navegar en el panel izquierdo así:

Configuración de seguridad > Configuración de directiva de auditoría avanzada > Directivas de auditoría del sistema – Objeto de directiva de grupo local > Cambio en directivas

Subcategoría: Auditar cambio de directiva del nivel de reglas de MPSSVC.

Mpssvc.exe es el Microsoft Protection Service, que es utilizado por Windows Firewall. Las actividades auditadas se pueden observar en la pestaña “Explicar” que se puede abrir en la ventana de activación/desactivación de esta directiva:



Si se configura esta directiva se generan los eventos siguientes:

4944: The following policy was active when the Windows Firewall started.

4945: A rule was listed when the Windows Firewall started.

4946: A change has been made to Windows Firewall exception list. A rule was added.

4947: A change has been made to Windows Firewall exception list. A rule was modified.

4948: A change has been made to Windows Firewall exception list. A rule was deleted.

4949: Windows Firewall settings were restored to the default values.

4950: A Windows Firewall setting has changed.

4951: A rule has been ignored because its major version number was not recognized by Windows Firewall.

4952: Parts of a rule have been ignored because its minor version number was not recognized by Windows Firewall. The other parts of the rule will be enforced.

4953: A rule has been ignored by Windows Firewall because it could not parse the rule.

4954: Windows Firewall Group Policy settings have changed. The new settings have been applied.

4956: Windows Firewall has changed the active profile.

4957: Windows Firewall did not apply the following rule:

4958: Windows Firewall did not apply the following rule because the rule referred to items not configured on this computer:

Estos eventos se generan en: Registros de Windows\Seguridad

Para obtener ayuda sobre los eventos generados se puede acceder a diversas páginas en Internet. A continuación se indican algunas:

https://www.ultimatewindowssecurity.com/securitylog/encyclopedia/

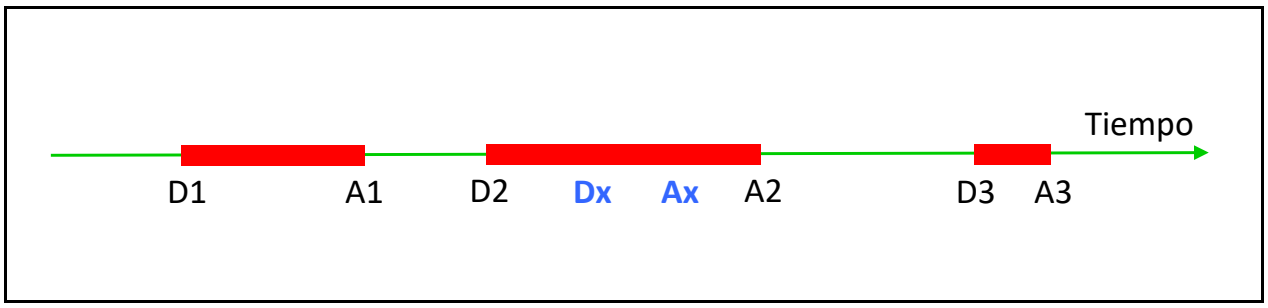
http://eventopedia.cloudapp.net/Events/?/Operating+System/Microsoft+Windows

https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4950

En esta última página se puede cambiar el número de evento para obtener información específica sobre cada evento.

3.-Realizar pruebas para generar eventos.

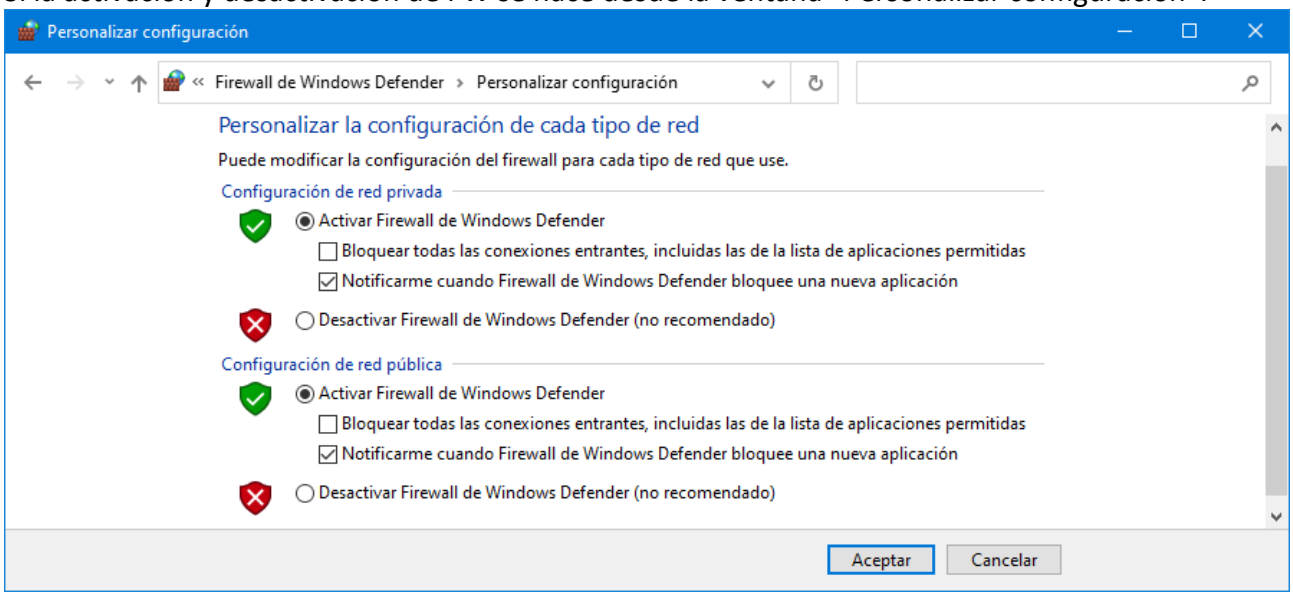
En esta tarea los alumnos deben realizar algunas pruebas para generar eventos de auditoría. Se sugiere generar tres períodos de desactivación del Firewall (perfil público), como se muestra en la figura siguiente:



El comienzo del período está determinado por un evento Dn de Desactivación y el final del período por un evento An de Activación. Además, mientras el Firewall del perfil público esta desactivado, se recomienda desactivar el Firewall del perfil privado (evento Dx) y volver a activarlo (evento Ax). De este modo hay eventos mezclados de ambos perfiles.

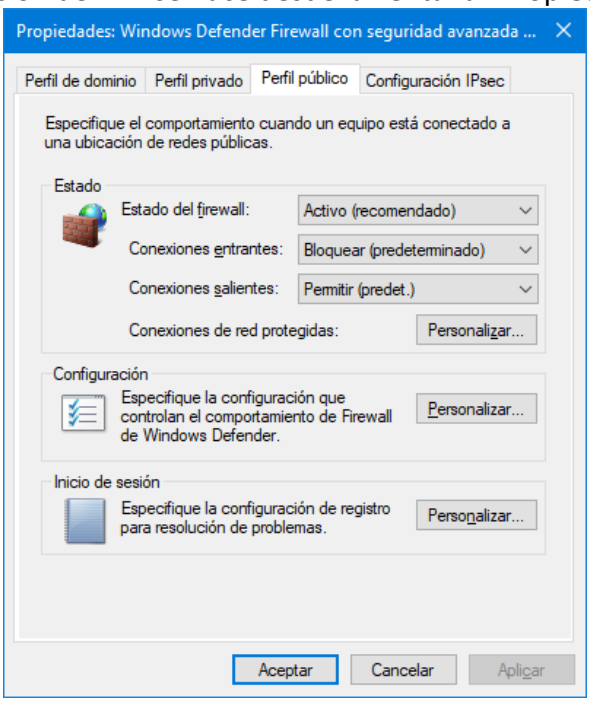
Dependiendo de la forma utilizada para activar y desactivar el FW se genera un número de eventos diferente. Observa las dos formas siguientes de activar y desactivar:

Si la activación y desactivación de FW se hace desde la ventana “Personalizar configuración”:



Entonces cada activación del FW genera un solo evento con ID 4950.

Si la activación y desactivación del FW se hace desde la ventana "Propiedades":



Seleccionando el estado Activo o Inactivo y después pulsando el botón Aplicar…
Entonces cada activación del FW genera CUATRO eventos con ID 4950.

Además del evento de interés:

- Habilitar Firewall de Windows Defender → Sí

Se generan los eventos:

- Acción saliente predeterminada → Permitir
- Modo blindado del FW → Sin configurar
- Acción saliente predeterminada → Bloquear

Utilizar el visor de eventos, para guardar un fichero con los eventos de la última hora. Una vez almacenado el fichero .evtx haz doble-clic sobre él para comprobar con el visor que están guardados los eventos de interés, que en este caso son los eventos con ID 4950.

4.-Analizar los registros de eventos de seguridad.

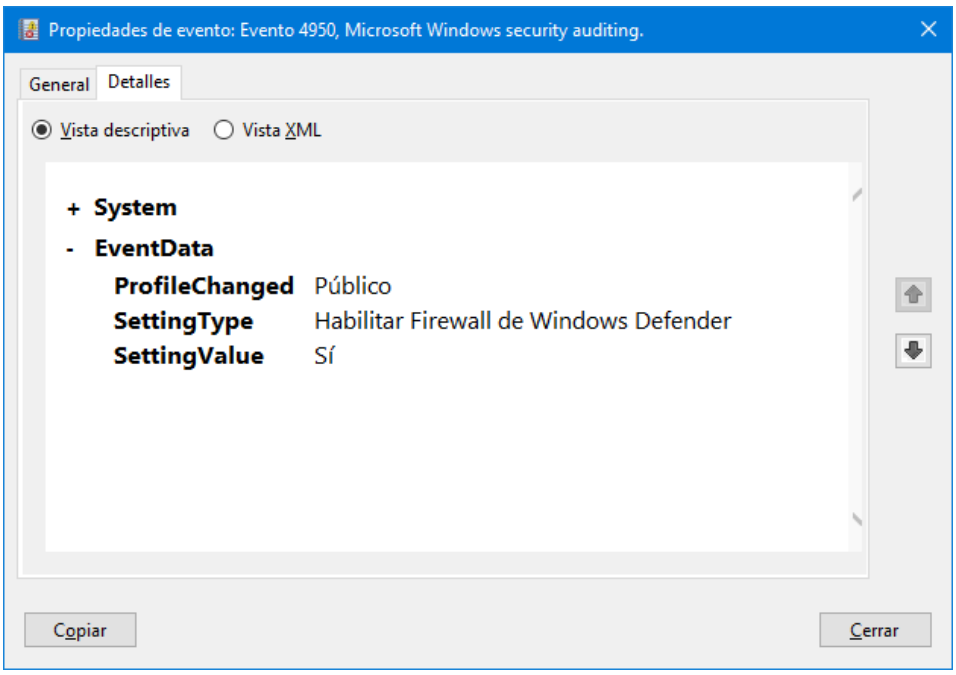
Crea una **nueva solución de Visual Studio** para analizar los periodos de desactivación del FW. Reutiliza el programa creado previamente que contiene el método de lectura de eventos de un fichero y el método de filtrado de los eventos por ID.

Copia el fichero con los eventos a analizar en el directorio ..\bin\Debug\ de la solución.

Tras generar una lista de eventos LE con los eventos de interés, escribe en la consola el número de eventos contenidos en LE y comprueba que es correcto comparándolos con los que proporciona el visor eventvwr.

Puedes comprobar que todos los eventos de interés tienen ID = 4950, tanto al activar como al desactivar el FW, por lo que es necesario utilizar las propiedades del evento para distinguirlos.

En el visor de eventos, esas propiedades se pueden ver en la ventana que muestra las "Propiedades de evento", en la pestaña detalles y en la Vista descriptiva en la sección EventData. Esto se muestra en la figura siguiente.



Para ver cómo acceder a estos datos del evento desde un programa, abre el Visor de Ayuda de Visual Studio y busca en el índice la clase EventRecord. Lee la ayuda sobre el método FormatDescription() y la propiedad Properties.

Desarrolla un programa para estudiar un poco estos eventos…

Desarrolla el método **VerPropEven()** que recibe una lista con eventos, y para cada evento muestra en la consola el ID del evento y la información devuelta por el método **FormatDescription()**. Comprueba que este método devuelve la información de forma poco estructurada.

Parece que es mejor usar la propiedad "Properties". Para ello declara la lista de propiedades **LisPro** de la clase IList<EventProperty> y asígnale el valor de la propiedad Properties de cada evento a analizar. Usa un bucle for para escribir el valor de cada propiedad en la consola.

Comprueba los valores que se espera que aparezcan en las propiedades:

Propiedad 0: Público, Privado, Dominio

Propiedad 1: Habilitar Firewall de Windows Defender

Propiedad 2: Sí, No

Para determinar los períodos de desactivación del FW…

Comienza creando la clase PeriodoDesactivacion para anotar cada uno de los periodos.

```
class PeriodoDesactivacion
{
    public DateTime Tini = new DateTime();
    public DateTime Tfin = new DateTime();
    public TimeSpan Duracion = new TimeSpan();
}
```

Ahora desarrolla el método **VerPerFwDesactivado()** que recibe **LE** de la clase List<EventRecord> con la lista de eventos a analizar y **PerfilFW** del tipo string con el tipo de perfil de Firewall a analizar. El método no devuelve nada, pero debe mostrar en la consola un listado de los períodos de desactivación del FW indicando en cada línea del listado los **campos** siguientes:

Período **N** desde **fecha-hora INI** hasta **fecha-hora FIN** y duración **D**.

Comienza el método haciendo estas dos cosas:

Declara el objeto **PerDes** de la clase PeriodoDesactivacion y asígnale el valor null.

Crea la lista LisPerDes de la clase List<PeriodoDesactivacion>.

Ahora usa un bucle (for int e) para recorrer los eventos de la lista de entrada LE. Dentro del for:

Descarta el evento si su ID no es el deseado y procesa el evento siguiente.

```
if (LE[e].Id != 4950) continue;
```

Ahora extrae las propiedades del evento. Para ello, declara tres strings: **Perfil**, **TipoConfig** y **ValorConfig** y asígnales las tres propiedades del evento.

Descarta el evento si su **Perfil** no coincide con el **PerfilFW** deseado y procesa el evento siguiente.

```
if (Perfil != PerfilFW) continue;
```

Descarta el evento si su **TipoConfig** no es una habilitación del FW y procesa el evento siguiente.

```
if (TipoConfig != "Habilitar Firewall de Windows Defender") continue;
```

Si el evento no ha sido descartado por las sentencias previas, es que el evento es de interés y debe procesarse en función del contenido de la cadena **ValorConfig** del modo siguiente:

SI **ValorConfig** == "No" el evento corresponde a una Desactivación del FW. Crea un nuevo objeto de la clase PeriodoDesactivacion y asígnalo a **PerDes**. Seguidamente, asigna a PerDes.Ini el instante de creación del evento.

SINO SI **ValorConfig** == "Sí" el evento corresponde a una Activación del FW. En este caso, solo si existe un objeto de la clase PeriodoDesactivacion se procesa el evento haciendo:

1) Asigna a PerDes.Fin el instante de creación del evento.

2) Asigna a PerDes.Duracion la duración del período de desactivación del FW.

3) Añade el objeto PerDes con todos sus campos rellenados a la lista LisPerDes.

4) Asigna a la variable PerDes el valor null.

SINO Termina con esta cláusula else que solo se ejecuta si **ValorConfig** es distinto de Sí y de No, cosa que no debería ocurrir nunca. Escribe un mensaje de error en la consola y termina la ejecución del programa.

Después del bucle de procesamiento de eventos y justo antes de terminar el método, utiliza un bucle for para recorrer la lista **LisPerDes** y escribe en la consola una línea por cada período de desactivación del FW.

Realiza una **prueba básica** con el programa procesando el fichero de eventos generado previamente con 4 periodos de desactivación. Comprueba que la lista generada es correcta para los 3 tipos de perfil de FW: Público (3 períodos), Privado (1 período) y Dominio (0 períodos).

Realiza algunas **pruebas adicionales** para ver cómo se comporta el programa. Por ejemplo prueba los casos elementales siguientes, en los que se explica el comportamiento esperado del método.

Caso 1: Borrar D1 Este escenario se produce cuando se inicia la auditoría con el FW desactivado. Entonces el primer evento es de Activación. El primer periodo de desactivación no tiene instante inicial. El método descarta el primer período de desactivación porque no hay un objeto creado para anotarlo.
Caso 2: Borrar A3 Este escenario se produce cuando se finaliza la auditoría con el FW desactivado. Entonces el último evento es de Desactivación. El último periodo de desactivación no tiene instante final. El método descarta el tercer período de desactivación porque al no procesar el evento A3, el objeto creado para anotar el período no se añade a la lista de periodos de desactivación.
Caso 3: Borrar D2 Este escenario se produce cuando se pierde un evento de Desactivación del FW. La consecuencia es que en el registro de eventos aparecen dos eventos de Activación sucesivos. El método descarta el segundo período de desactivación porque no procesa el evento A2 al no haber un objeto creado para anotarlo, debido a que D2 se ha perdido.
Caso 4: Borrar A2 Este escenario se produce cuando se pierde un evento de Activación del FW. La consecuencia es que en el registro de eventos aparecen dos eventos de Desactivación sucesivos. El método descarta el segundo período de desactivación porque al procesar el evento D3 crea un nuevo objeto para anotar el nuevo periodo de desactivación, descartando el objeto creado para anotar el periodo anterior, sin añadirlo a la lista de periodos de desactivación.

Para preparar registros para estas pruebas se recomienda seguir el procedimiento siguiente.

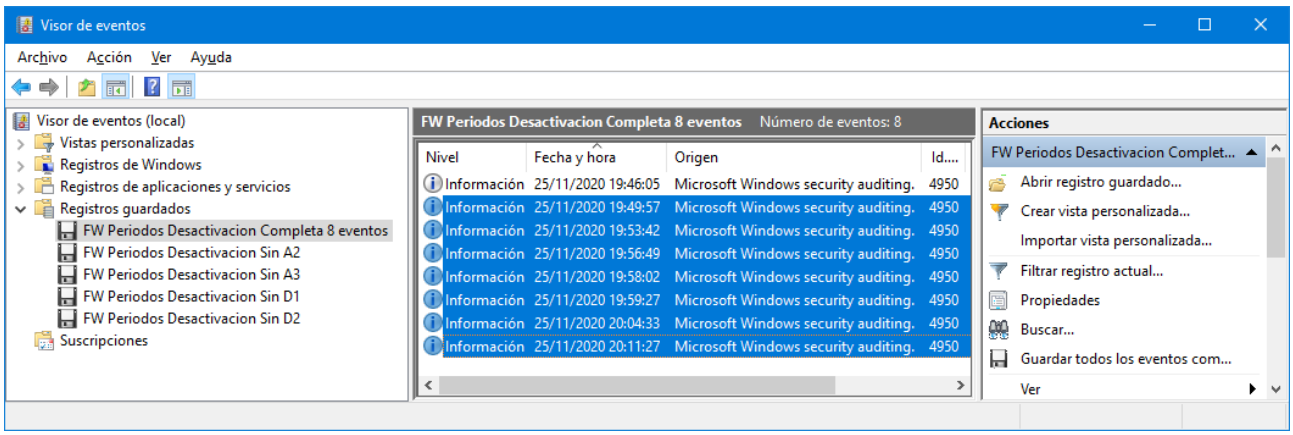
Abre la traza original con eventvwr y filtra los eventos con ID=4950, y también con un periodo de tiempo si es necesario, para quedarse solo con los 8 eventos a usar en las pruebas. Guarda un registro con los 8 eventos.

Por ejemplo denomínalo FW Periodos Desactivacion Completa 8 eventos.evtx

Carga ese registro en eventvwr y selecciona todos los eventos excepto el primero. Guarda un registro con los eventos seleccionados.

Por ejemplo denomínalo FW Periodos Desactivacion Sin D1.evtx

Y seleccionando siempre todos los eventos, excepto uno, genera el resto de registros de prueba. La figura siguiente muestra la selección de eventos realizada en eventvwr para excluir el primer evento del registro completo, D1. Antes de seleccionar eventos hay que ordenar los eventos por Fecha y hora.



Es muy recomendable crear dentro de la solución de Visual Studio un directorio denominado TEST en el que se almacenen todos los registros indicados anteriormente y se copia al directorio \bin\Debug el que sea necesario en cada momento.

Los siguientes ejercicios de esta práctica no son para realizarlos en la sesión presencial de prácticas. El alumno puede realizarlos cómo trabajos de extensión de la práctica.

Tan solo muestran ejemplos de las posibilidades de análisis que proporciona un sistema de registro de eventos como el de Windows para auditar la seguridad.

9. Ejercicio: Detección de escaneos

¿Se puede detectar un escaneo con los registros del Firewall?

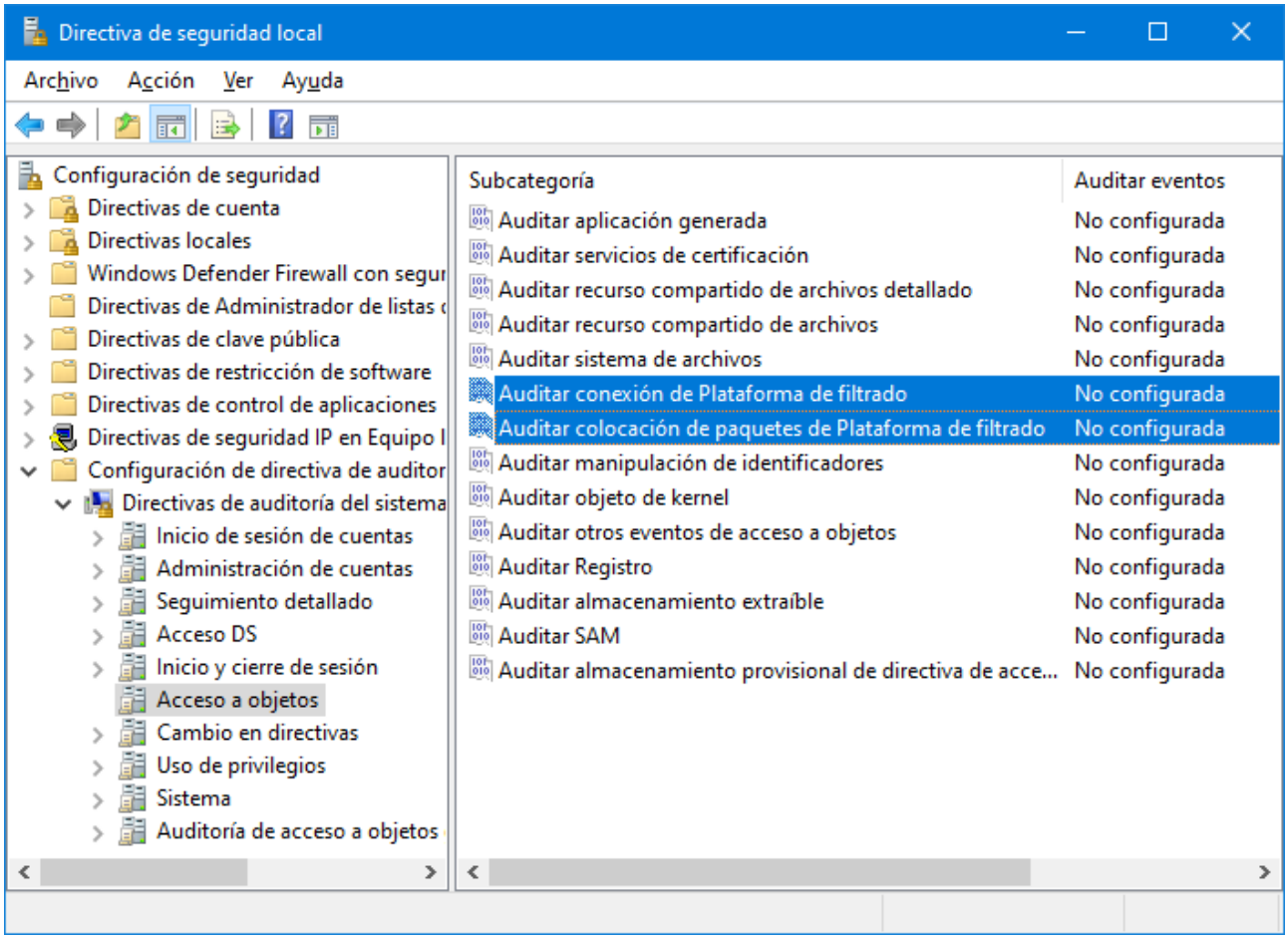
El primer objetivo del ejercicio es determinar si se generan registros de conexiones rechazadas y paquetes rechazados.

1.-Activar y configurar los controles de seguridad.

En este caso hay que configurar el cortafuegos de Windows. Pero se puede utilizar la configuración de reglas predeterminada y simplemente comprobar que el FW está activo para el perfil público, que es el utilizado en las prácticas.

2.-Activar y configurar la Auditoría de seguridad

Hay que seleccionar el Acceso a objetos, y configurar las dos subcategorías siguientes:



Configuración de "**Auditar conexión de Plataforma de filtrado**":

A continuación se copia la ayuda de esta directiva:

> Esta configuración de directiva permite auditar las conexiones permitidas o bloqueadas por la Plataforma de filtrado de Windows (WFP). Se incluyen los eventos siguientes:
>
> - El servicio Firewall de Windows impide que una aplicación acepte conexiones entrantes en la red.
> - WFP permite una conexión.
> - WFP bloquea una conexión.
> - WFP permite un enlace a un puerto local.
> - WFP bloquea un enlace a un puerto local.
> - WFP permite que una aplicación o servicio escuche conexiones entrantes en un puerto.
> - WFP impide que una aplicación o servicio escuche conexiones entrantes en un puerto.
>
> Si establece esta configuración de directiva, se generará un evento de auditoría cada vez que WFP permita o bloquee una conexión. Las auditorías de aciertos registran los eventos generados cuando se permiten las conexiones, mientras que las auditorías de errores registran eventos generados cuando se bloquean las conexiones.
>
> Si no establece esta configuración de directiva, no se generará ningún evento de auditoría cuando WFP permita o bloquee alguna conexión.

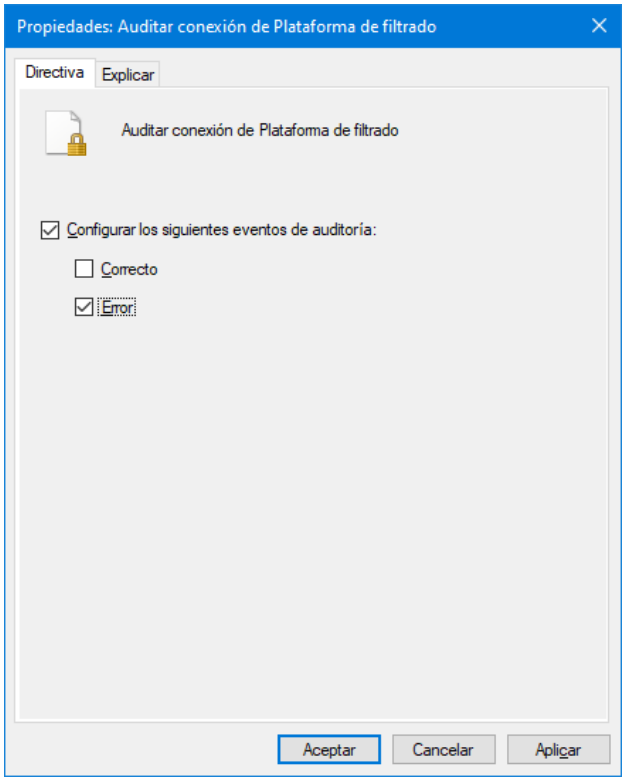
Información adicional en:

https://docs.microsoft.com/es-es/windows/security/threat-protection/auditing/audit-filtering-platform-connection

Al activar esta directiva se generan estos eventos:

- 5031(F): el servicio Firewall de Windows impidió que una aplicación aceptara conexiones entrantes en la red.
- 5150(-): la plataforma de filtrado de Windows bloqueó un paquete.
- 5151(-): un filtro de plataforma de filtrado de Windows más restrictivo bloqueó un paquete.
- 5154(S): la plataforma de filtrado de Windows permitió a una aplicación o servicio escuchar en un puerto las conexiones entrantes.
- 5155(F): la plataforma de filtrado de Windows ha bloqueado una aplicación o servicio para que no escuche en un puerto las conexiones entrantes.
- 5156(S): la plataforma de filtrado de Windows permitió una conexión.
- 5157(F): la plataforma de filtrado de Windows bloqueó una conexión.
- 5158(S): la plataforma de filtrado de Windows permitió enlazar un puerto local.
- 5159(F): la plataforma de filtrado de Windows bloqueó un enlace a un puerto local.

Preferentemente se configura la directiva así:



Para evitar la generación de muchos eventos debidos a las conexiones correctas.

Configuración de "**Auditar colocación de paquetes de Plataforma de filtrado**":

A continuación se copia la ayuda de esta directiva:

Pérdida de paquetes de Plataforma de filtrado de Windows Esta configuración de directiva permite auditar los paquetes que pierde la Plataforma de filtrado de Windows (WFP).

Información adicional en:

https://docs.microsoft.com/es-es/windows/security/threat-protection/auditing/audit-filtering-platform-packet-drop

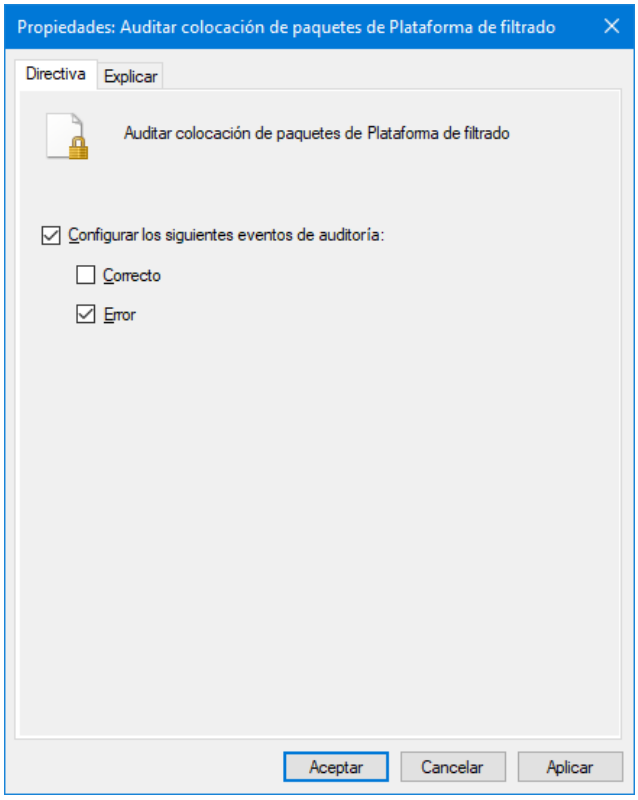
Indica: No se recomienda habilitar la auditoría de aciertos porque los eventos de éxito en esta subcategoría rara vez se producen.

Indica: Puede ser mejor auditar solo las conexiones bloqueadas, pues la información obtenida puede ser parecida pero con un volumen de eventos mucho menor.

Al activar esta directiva se generan estos eventos:

- 5152(F): la plataforma de filtrado de Windows bloqueó un paquete.
- 5153(S): un filtro de plataforma de filtrado de Windows más restrictivo bloqueó un paquete.

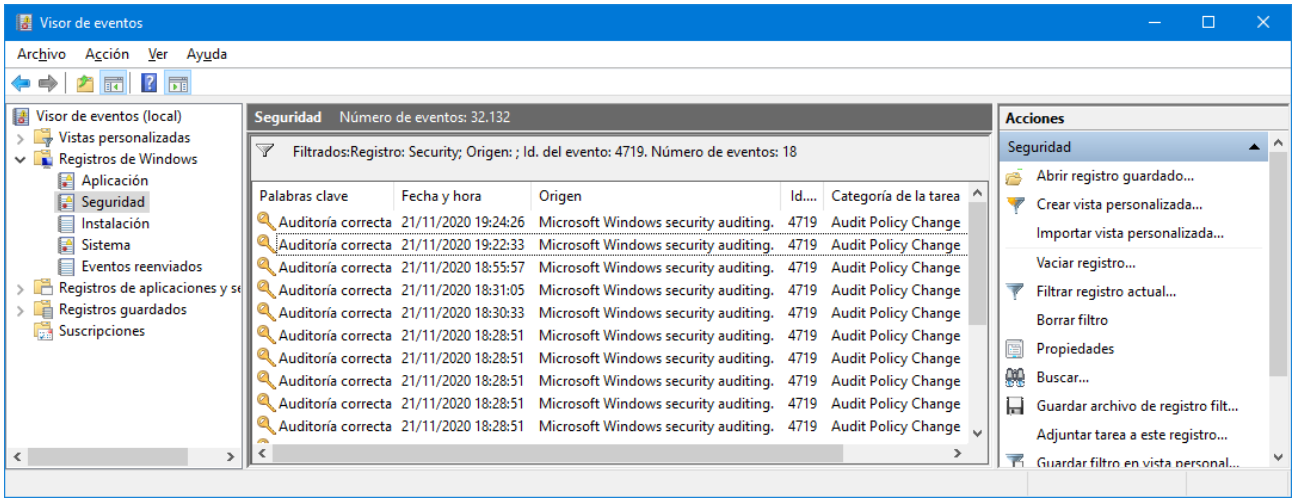
Preferentemente se configura la directiva así:



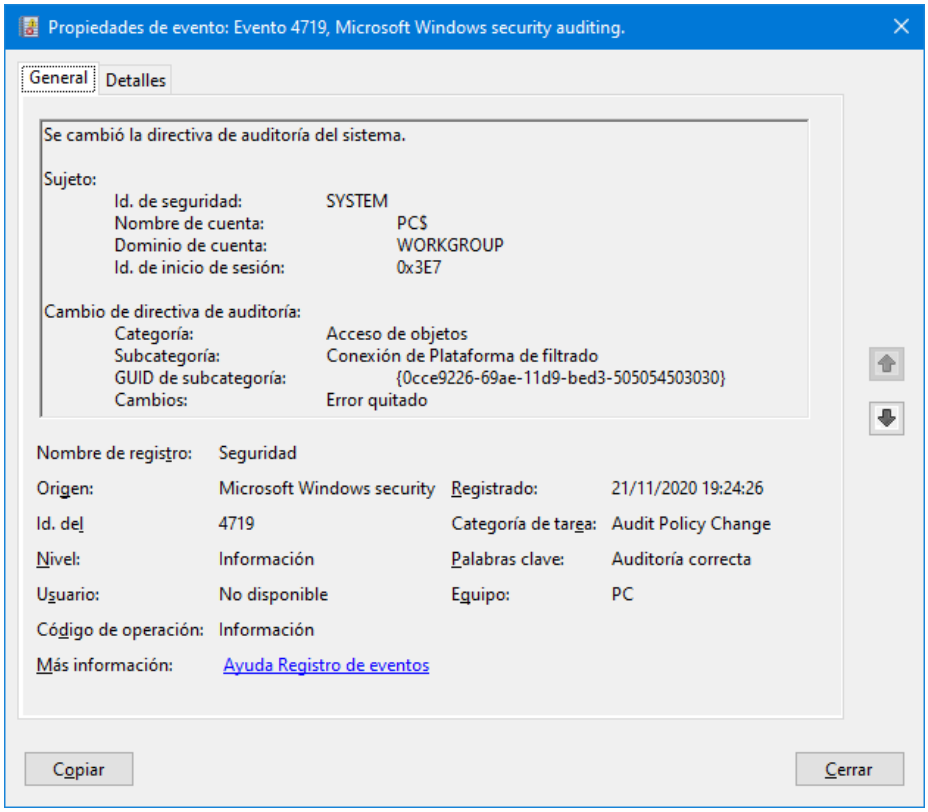
Para evitar la generación de muchos eventos debidos a los paquetes permitidos.

PROBLEMA: Muchos eventos. Tras activar las dos directivas se observa el Registro de Windows "Seguridad" y se comprueba que hay una secuencia de eventos 5152 y 5157 de entre 1 y 16 eventos cada segundo. Tras desactivar la auditoría del bloqueo de paquetes se comprueba que hay una secuencia de eventos 5157 de entre 1 y 8 eventos cada segundo.

Es interesante indicar que cada cambio que se realiza se audita generándose el evento "Audit Policy Change" ID 4719. Ver un ejemplo:



Evento generado al quitar la auditoría de errores en la conexión de Plataforma de filtrado:



Se analiza la posible realización de un entorno de red aislado para pruebas, pero no parece factible de forma sencilla. Ver VirtualBox UserManual sección 6. La única opción directa es usar MVs con la red en modo puente.

3.-Realizar pruebas para generar eventos + 4.-Analizar resultados

En esta sección se proponen tres escaneos de puertos. Para cada escaneo se prueban tres configuraciones de auditoría con el par de directivas explicadas previamente. Y adicionalmente se usa el registro del FW para realizar comparaciones.

Para configuración de auditoría se analizan los resultados del escaneo.

ESCANEO TCP Connect

nmap -sT -p 1-1000 -T3 -v -r 156.35.151.51

Se escanean los puertos del 1 al 1000 en secuencia.

● Auditar colocación de paquetes de Plataforma de filtrado: Correcto o Aciertos

Tras el escaneo NO hay ningún evento en el visor.

● Auditar colocación de paquetes de Plataforma de filtrado: Error

Tras el escaneo que dura 50 segundos aparecen en el visor 2388 eventos. Visualizando rápidamente la secuencia de eventos observando solo el puerto se puede apreciar claramente la secuencia de acceso a los puertos: 1,2,3,…, 1000 y un reintento de conexión en cada puerto.

SE SALTA EL PUERTO 80 que tiene una regla del FW permitiendo el tráfico de entrada.

● Auditar colocación de paquetes de Plataforma de filtrado: Correcto Y Error (ambos)

Los resultados son totalmente similares que auditando solo Errores

SE SALTA EL PUERTO 80 que tiene una regla del FW permitiendo el tráfico de entrada.

● Auditar conexión de Plataforma de filtrado: Correcto Y Error (ambos)

Tras el escaneo que dura 50 segundos aparecen en el visor 419 eventos. Nos centramos en los que contienen la IP del computador escáner y la IP de computador escaneado. Solo se han registrado conexiones en los puertos: 80 allows:31, 135 blocks:2, 139 blocks:2, 445 blocks:2.

También hay un solo evento con estas IPs al principio del escaneo que indica el puerto de destino 8 Allow. Pero el protocolo = 1 que es ICMP. Entonces el puerto carece de sentido.

En este escaneo Nmap reintenta continuamente la conexión en el puerto 80.

● Registro del FW: Paquetes descartados Y Conexiones correctas (ambos)

Tras el escaneo que dura 50 segundos se registran 126 eventos. Pero con la IP del computador escáner y la IP de computador escaneado solo se han registrado 3 acciones DROP en los puertos: 135, 139, 445, pero dos intentos en cada puerto. Hay 28 acciones ALLOW en el puerto 80.

Con ambas IPs solo hay una acción ALLOW ICMP con icmptype=8 y icmpcode=0.

ESCANEO TCP SYN

nmap -sT -p 1-1000 -T3 -v -r 156.35.151.51

Se escanean los puertos del 1 al 1000 en secuencia.

● Auditar colocación de paquetes de Plataforma de filtrado: Correcto Y Error (ambos)

Tras el escaneo, que dura 7 segundos, aparecen en el visor 2157 eventos. Visualizando rápidamente la secuencia de eventos observando solo el puerto se puede apreciar claramente la secuencia de acceso a los puertos: 1,2,3,…, 1000. Escanea 10 puertos seguidos, ej. 1>>10 y luego realiza un solo reintento de conexión en cada puerto, 10>>1, y así sucesivamente hasta el último puerto escaneado, el 1000.

SE SALTA EL PUERTO 80 que tiene una regla del FW permitiendo el tráfico de entrada.

● Auditar conexión de Plataforma de filtrado: Correcto Y Error (ambos)

Tras el escaneo que dura 7 segundos aparecen en el visor 46 eventos.

Pero con la IP del computador escáner y la IP de computador escaneado solo se han registrado conexiones en los puertos: 80 allows:3, 135 blocks:2, 139 blocks:2, 445 blocks:2. También hay un solo evento con estas IPs que indica el puerto de destino 8 Allow. Pero el protocolo = 1 que es ICMP. Entonces el puerto carece de sentido.

● Registro del FW: Paquetes descartados Y Conexiones correctas (ambos)

Tras el escaneo que dura 7 segundos se registran 18 eventos. Pero con la IP del computador escáner y la IP de computador escaneado solo se han registrado 3 acciones DROP en los puertos: 135, 139, 445, pero dos intentos en cada puerto.

Con ambas IPs solo hay una acción ALLOW ICMP con icmptype=8 y icmpcode=0.

ESCANEO UDP

nmap -sU -p 1-1000 -T3 -v -r 156.35.151.51

Se escanean los puertos del 1 al 1000 en secuencia.

● Auditar colocación de paquetes de Plataforma de filtrado: Correcto Y Error (ambos)

Tras el escaneo que dura 26 segundos aparecen en el visor 2365 eventos. Visualizando rápidamente la secuencia de eventos observando solo el puerto puede apreciar claramente la secuencia de acceso a los puertos: 1,2,3,…, 1000 y un reintento de conexión en cada puerto.

● Auditar conexión de Plataforma de filtrado: Correcto Y Error (ambos)

Tras el escaneo que dura 26 segundos aparecen en el visor 195 eventos. Pero con la IP del computador escáner y la IP de computador escaneado solo se han registrado conexiones en 3 puertos: 123 block, 137 block, 138 block. También hay un solo evento con estas IPs que indica el puerto de destino 8 Allow. Pero el protocolo = 1 que es ICMP. Entonces el puerto carece de sentido.

● Registro del FW: Paquetes descartados Y Conexiones correctas (ambos)

Tras el escaneo que dura 26 segundos se registran 238 eventos. Pero con la IP del computador escáner y la IP de computador escaneado solo se han registrado 3 acciones DROP en los puertos: 123, 137, 138, pero dos intentos en cada puerto.

Con ambas IPs solo hay una acción ALLOW ICMP con icmptype=8 y icmpcode=0.

CONCLUSIONES de las pruebas:

1) La directiva de secpol "Auditar la colocación de Paquetes de la plataforma de filtrado" permite registrar los paquetes descartados durante un escaneo de puertos realizado con Nmap de los tipos básicos (TCP Connect, TCP SYN, UDP). TODOS los eventos generados tienen ID=5152 (Filtering Platform Packet Drop). NO se registran los paquetes del escaneo permitidos por el FW.

Como la mayoría de los paquetes enviados en un escaneo son descartados, parece que es posible implementar un detector de escaneos analizando los eventos del Registro de Seguridad de Windows.

2) La directiva de secpol "Auditar conexión de Plataforma de filtrado" y el Registro del FW, registran ambos la misma información, aunque en formatos diferentes. Durante un escaneo, parece que solo generan eventos para servicios que están instalados y usan puertos concretos en el computador escaneado Y ADEMÁS servicios que tengan una regla de entrada en el FW para esos puertos.

Si la regla está habilitada la acción registrada será ALLOW.

Si la regla NO está habilitada la acción registrada será DROP.

La información registrada con esta directiva no permite implementar un detector de escaneos.

10. Ejercicio: Analizar las sesiones de los usuarios

El objetivo consiste en extraer de una secuencia de eventos la información de las sesiones de los usuarios. Cada sesión está caracterizada por un instante de inicio, un instante de finalización, y una duración.

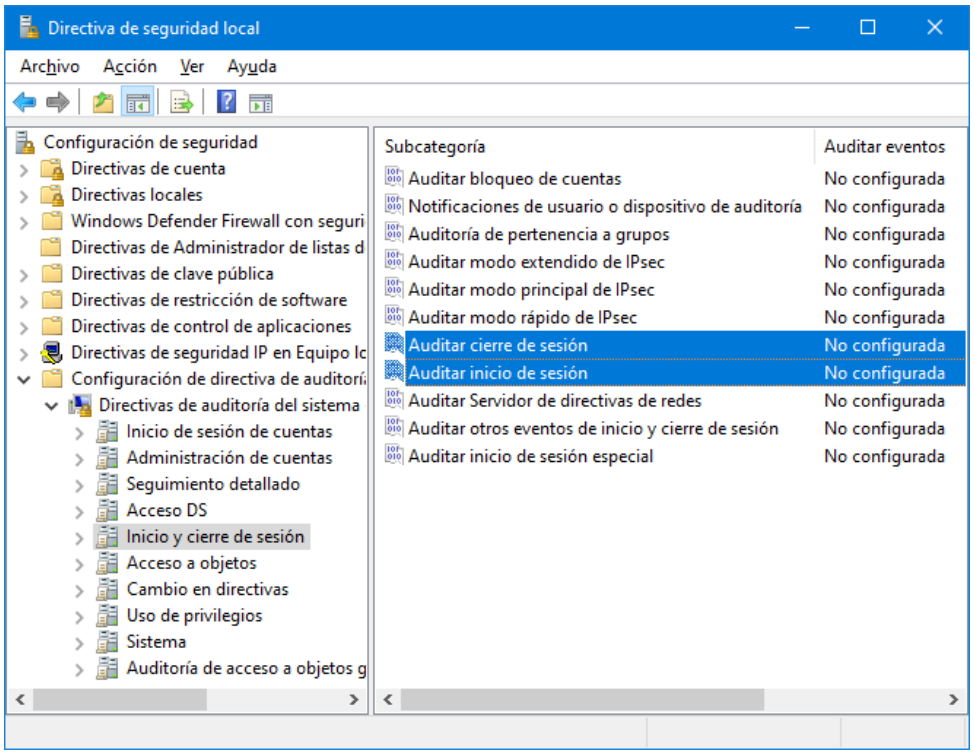
Adicionalmente se pueden anotar los intentos de inicio de sesión erróneos, que no generan una sesión.

1.-Activar y configurar los controles de seguridad.

En este caso no hay que configurar ningún control de seguridad. La utilización de sesiones de usuario para acceder al computador está integrada y activada en el sistema operativo.

2.-Activar y configurar la Auditoría de seguridad

Usa la herramienta secpol para acceder a las Directivas de auditoría del sistema. Hay que configurar las dos Subcategorías mostradas en la figura siguiente.



Configura los inicios de sesión para auditar: aciertos y errores.

Configura los cierres de sesión para auditar: aciertos y errores.

3.-Realizar pruebas para generar eventos.

Antes de comenzar las pruebas hay que conocer los eventos que se generan en los procesos de Inicio/Cierre (Logon/Logoff) de las sesiones de usuario. Consultar la página siguiente: https://www.ultimatewindowssecurity.com/securitylog/encyclopedia/

Seleccionar "Windows Audit" y obtener la información siguiente:

Category: Logon/Logoff Subcategory: Logon Windows 4624 An account was successfully logged on Windows 4625 An account failed to log on Windows 4626 User/Device claims information Windows 4648 A logon was attempted using explicit credentials Windows 4675 SIDs were filtered
Category: Logon/Logoff Subcategory: Logoff Windows 4634 An account was logged off Windows 4647 User initiated logoff

Estos eventos se almacenan en Registros de Windows\Seguridad.

3.-Realizar pruebas para generar eventos.

Antes de comenzar las pruebas se puede vaciar el Registro de Seguridad de Windows para facilitar la visualización de los eventos de interés.
Cierra la sesión, espera unos minutos, y vuelve a iniciar sesión. La espera se hace para que los instantes de tiempo (minutos) de los eventos de cierre e inicio sean claramente distintos.
Observa que no se genera un solo evento para el cierre y otro para el inicio, sino múltiples eventos en ambos casos.

Un **inicio de sesión** correcto genera TIPICAMENTE 8 eventos en el registro de Windows:

Nº	EvID	Características
1	4648	Se intentó iniciar sesión con credenciales explícitas. Nombre de cuenta: ADM
2	4624	Se inició sesión correctamente en una cuenta. Tipo de inicio de sesión: 2 Token elevado: Sí Nombre de cuenta: ADM
3	4624	Se inició sesión correctamente en una cuenta. Tipo de inicio de sesión: 2 Token elevado: No Nombre de cuenta: ADM
4 5 6 7 8	4624	Se inició sesión correctamente en una cuenta. Tipo de inicio de sesión: 5 Token elevado: Sí Nombre de cuenta: SYSTEM

En ocasiones puede haber menos o más eventos finales 4624 generados por SYSTEM.

Para una información detallada de los eventos 4648 y 4624:

https://learn.microsoft.com/es-es/windows/security/threat-protection/auditing/event-4648

https://learn.microsoft.com/es-es/windows/security/threat-protection/auditing/event-4624

Un **cierre de sesión** correcto genera TIPICAMENTE 4 eventos en el registro de Windows:

Nº	EvID	Características
1	4647	Cierre de sesión iniciado por el usuario. Nombre de cuenta: ADM
2	4634	Se cerró sesión en una cuenta. Tipo de inicio de sesión: 2 Nombre de cuenta: UMFD-2
3	4634	Se cerró sesión en una cuenta. Tipo de inicio de sesión: 2 Nombre de cuenta: DWM-2
4	4634	Se cerró sesión en una cuenta. Tipo de inicio de sesión: 2 Nombre de cuenta: DWM-2 Cambia el Id. de inicio de sesión del evento 3

Tras el primer o segundo evento del cierre de sesión puede aparecer una secuencia de eventos 4648 y 4624 correspondientes a un proceso de Logon de SYSTEM.

Para una información detallada de los eventos 4647 y 4634:

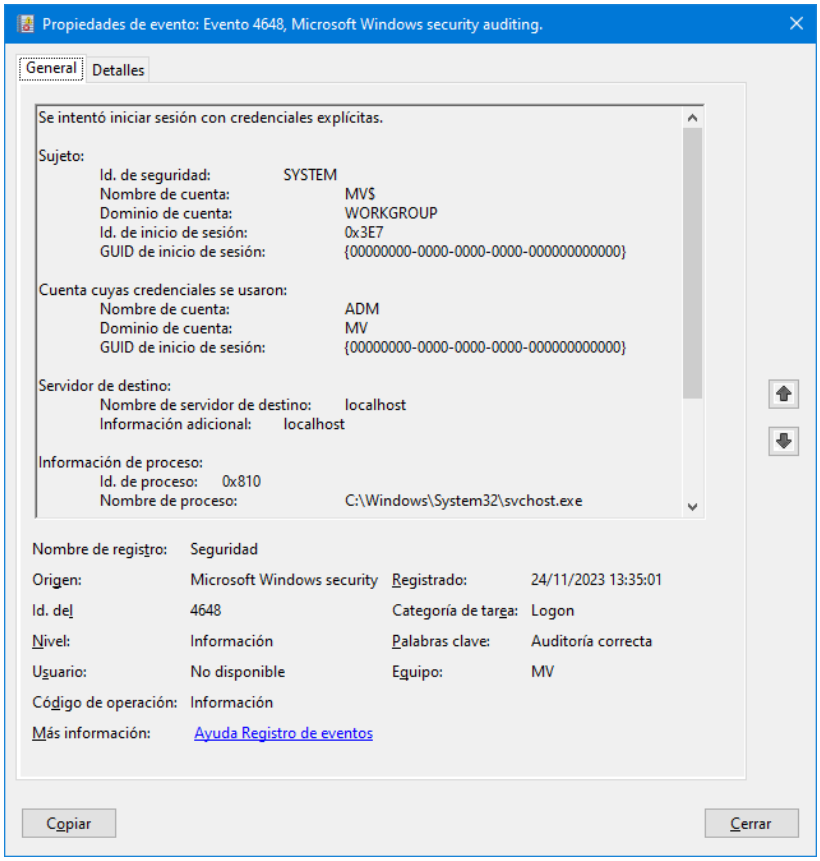
https://learn.microsoft.com/es-es/windows/security/threat-protection/auditing/event-4647

https://learn.microsoft.com/es-es/windows/security/threat-protection/auditing/event-4634

Del análisis previo, parece que el inicio de una sesión se puede definir mediante un evento 4648 y el cierre de la sesión mediante un evento 4647.

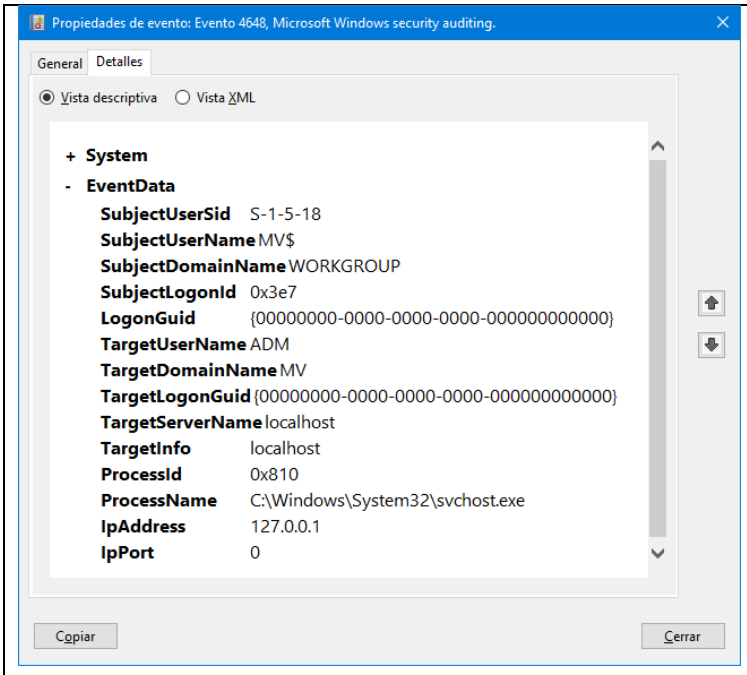
Por ello, analiza con un poco de detalle estos dos eventos usando la ventana de propiedades del visor de eventos.

Selecciona un evento 4648 y observa sus propiedades:



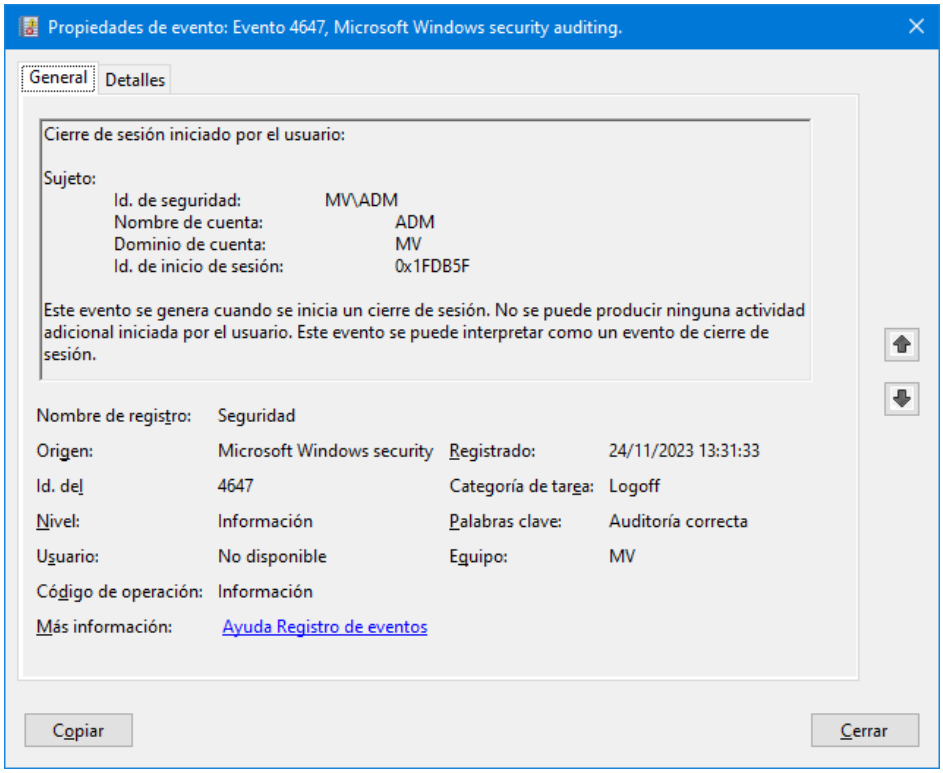
Observa el nombre de cuenta: ADM, pues será esencial para diferenciar un inicio de sesión del usuario ADM de otros eventos con ID 4648 pero con un nombre de cuenta correspondiente a un servicio del sistema operativo, como DWM-3 o UFMD-3.

En la pestaña de detalles del evento se pueden ver sus propiedades con los nombres que se deben usar en un programa que procese los eventos.



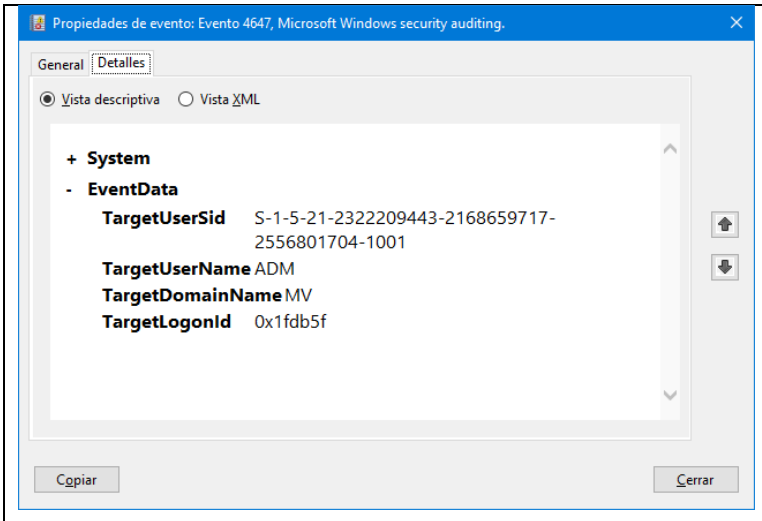
Observa que, empezando a contar desde cero, la propiedad (EventData) número 5 se denomina TargetUserName y toma el valor ADM.

Selecciona un evento 4647 y observa sus propiedades:



Observa el nombre de cuenta: ADM, pues será esencial para diferenciar un cierre de sesión del usuario ADM de otros posibles eventos con ID 4647.

En la pestaña de detalles del evento se pueden ver sus propiedades con los nombres que se deben usar en un programa que procese los eventos.



Observa que, empezando a contar desde cero, la propiedad (EventData) número 1 se denomina TargetUserName y toma el valor ADM.

Desarrolla un programa para estudiar un poco estos eventos…

Desarrolla el método **VerPropEven()** que recibe una lista con eventos, y para cada evento muestra en la consola el ID del evento y la información devuelta por el método **FormatDescription()**. Comprueba que este método devuelve la información de forma poco estructurada.

Es mejor usar la propiedad "Properties". Para ello declara la lista de propiedades **LisPro** de la clase IList<EventProperty> y asígnale el valor de la propiedad Properties de cada evento a analizar. Usa un bucle for para escribir el valor de cada propiedad en la consola.

Comprueba los valores que se espera que aparezcan en las propiedades:

Para los eventos con ID=4648 la propiedad 5 vale: UMFD-n, DWM-n, ADM.

Para los eventos con ID=4647 la propiedad 1 vale: ADM.

Los identificadores de los eventos y sus propiedades de interés se utilizarán en un bucle de procesamiento de eventos para extraer los periodos de las sesiones de un usuario. Por ello, es esencial su conocimiento.

Para determinar los períodos de duración de las Sesiones…

Comienza creando la clase Sesion para anotar cada uno de los periodos.

```
class Sesion
{
    public DateTime Tini = new DateTime();
    public DateTime Tfin = new DateTime();
    public TimeSpan Duracion = new TimeSpan();
}
```

Ahora desarrolla el método **VerSesiones()** que recibe **LE** de la clase List<EventRecord> con la lista de eventos a analizar y **NombreUsuario** del tipo string con el nombre del usuario cuyas sesiones se deben analizar. El método no devuelve nada, pero debe mostrar en la consola un listado de las sesiones del usuario indicando en cada línea del listado los **campos** siguientes:

Sesión **N** desde **fecha-hora INI** hasta **fecha-hora FIN** y duración **D**.

Comienza el método haciendo estas dos cosas:

Declara el objeto **Ses** de la clase Sesion y asígnale el valor null.

Crea la lista **ListaSes** de la clase List<Sesion>.

Ahora usa un bucle (for int e) para recorrer los eventos de la lista de entrada LE. Dentro del for procesa eventos solamente SI son inicios de sesión o SI son cierres de sesión.

Procesa el evento si su ID = Inicio de sesión así:

```
if (LE[e].Id == 4648) // Inicio sesión
{
    if (LE[e].Properties[5].Value.ToString() == NombreUsuario)
    {
        if (Ses == null) // Comentar o descomentar este if()
        {
            Ses = new Sesion();
            Ses.Tini = (DateTime)LE[e].TimeCreated;
        }
    }
}
```

Procesa el evento si su ID = Cierre de sesión así:

```
if (LE[e].Id == 4647) // Fin sesión
{
    if (LE[e].Properties[1].Value.ToString() == NombreUsuario)
    {
        if (Ses != null)
        {
            Ses.Tfin = (DateTime)LE[e].TimeCreated;
            Ses.Duracion = Ses.Tfin - Ses.Tini;
            ListaSes.Add(Ses);
            Ses = null;
        }
    }
}
```

Después del bucle de procesamiento de eventos y justo antes de terminar el método, utiliza un bucle for para recorrer la lista **ListaSes** y escribe en la consola una línea por cada sesión del usuario indicado.

11. Ejercicio: Analizar el registro de eventos del antivirus

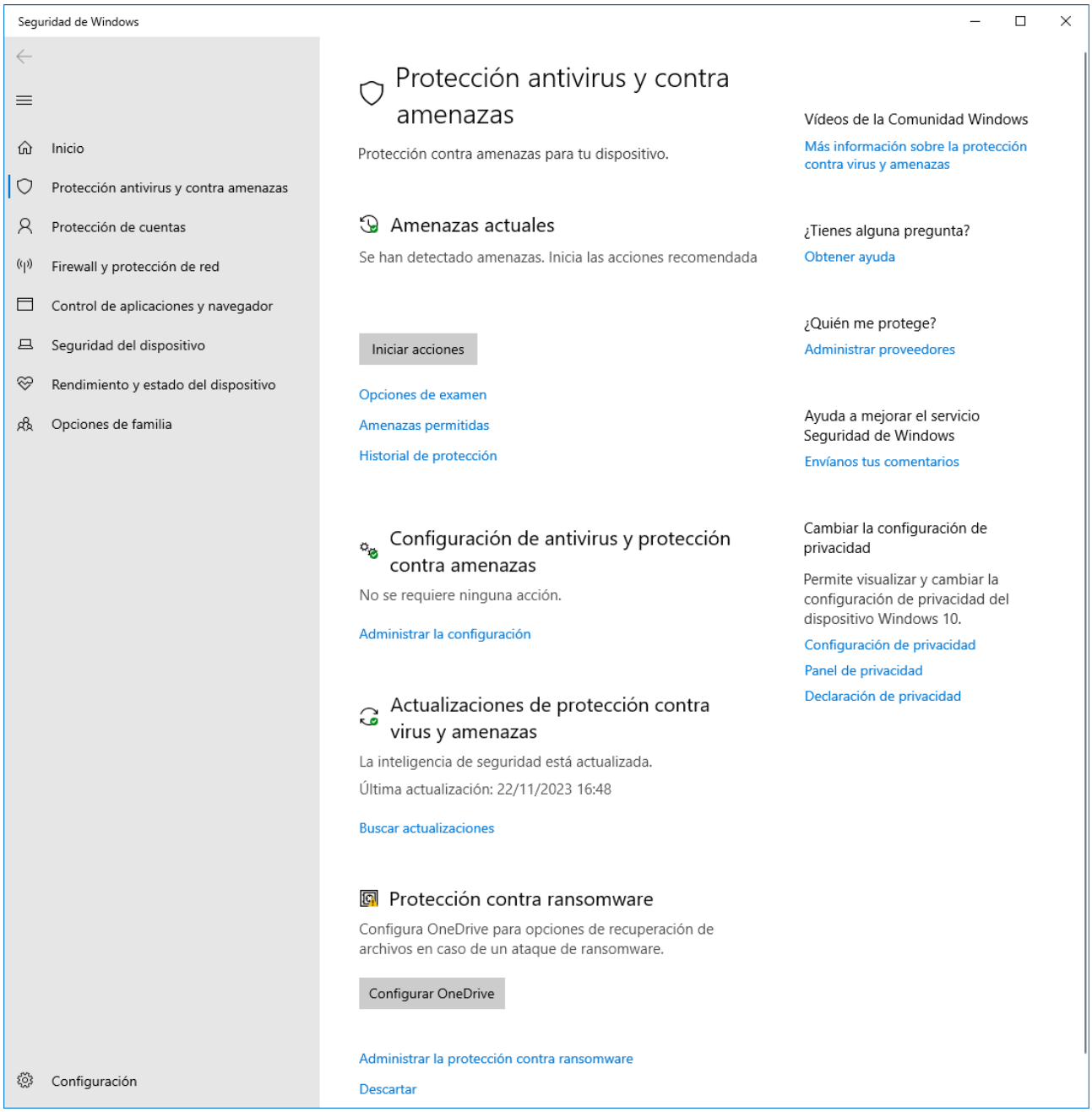
El objetivo consiste en realizar un análisis de los eventos básicos que genera el antivirus (AV).

Página con información detallada sobre los eventos que genera el antivirus Defender:

https://docs.microsoft.com/es-es/windows/security/threat-protection/microsoft-defender-antivirus/troubleshoot-microsoft-defender-antivirus

Accede al Antivirus:

Botón Configuración > Actualización y seguridad > En el panel izquierdo selecciona “Seguridad de Windows” y en el panel derecho “Protección antivirus y contra amenazas”



Accede al Visor de Eventos:

En el panel izquierdo debes seleccionar el registro del AV.

Visor > Registro de aplicaciones y servicios > Microsoft > Windows > Windows Defender

Selecciona el registro "Operational"

Ahora realiza pruebas en el AV y comprueba lo que se registra en el Visor:

Selecciona "Administrar la configuración"

Protección en tiempo real → Desactivar. Genera un evento 5001.

Protección en tiempo real → Activar. Genera un evento 5000.

Protección basada en la nube → Desactivar. Genera un evento 5007.

Protección basada en la nube → Activar. Genera un evento 5007.

Los valores anterior y nuevo de SpyNetReporting son diferentes.

Envío de muestras automático → Desactivar. Genera un evento 5007.

Envío de muestras automático → Activar. Genera otro evento 5007.

Los valores anterior y nuevo de SubmitSamplesConsent son diferentes.

Protección contra alteraciones → Desactivar. Genera un evento 5007.

Protección contra alteraciones → Activar. Genera un evento 5007.

Los valores anterior y nuevo de TamperProtection son diferentes.

Activa el Control del acceso a carpetas. Genera un evento 5007.

Agrega la carpeta C:\TEMP a las carpetas protegidas por defecto. Genera un evento 5007.

Elimina la carpeta C:\TEMP de las carpetas protegidas por defecto. Genera un evento 5007.

Desactiva el Control del acceso a carpetas. Genera un evento 5007.

Agrega una exclusión a los análisis que realiza el AV: la capeta C:\TEMP

Esto genera un evento 5007, con un Valor anterior vacío.

Elimina la exclusión a los análisis que realiza el AV: la capeta C:\TEMP

Esto genera un evento 5007, con un Valor nuevo vacío.

La desactivación/activación de notificaciones de Protección contra virus no genera eventos en el registro.

La búsqueda de actualizaciones, si son necesarias, puede generar varios eventos:

1150, 1151, 2000 (2 eventos) y 5007.

Pero si todo está actualizado no se generan eventos.

Realiza un examen rápido.

El inicio genera el evento 1000

La finalización genera el evento 1001.

Con el conocimiento adquirido se pueden generar secuencias de eventos que reflejen múltiples aspectos de la configuración y funcionamiento del antivirus, que se pueden usar para auditar posteriormente su funcionamiento.