Novell® Sentinel™

5.1.3

www.novell.com

Volumen IV: GUÍA DE REFERENCIA DE SENTINEL

7 de julio de 2006

Novell®

Aviso legal

Novell, Inc.
404 Wyman Street, Suite 500
Waltham, MA 02451
EE.UU.
www.novell.com

Documentación en línea: Para acceder a la documentación en línea de éste y otros productos de Novell y obtener actualizaciones, consulte www.novell.com/documentation.

Marcas comerciales de Novell

Para obtener información sobre marcas comerciales de Novell, consulte la lista de marcas comerciales y de marcas de servicio de Novell (http://www.novell.com/company/legal/trademarks/tmlist.html).

Materiales de otros fabricantes

Todas las marcas comerciales de otros fabricantes pertenecen a sus respectivos propietarios.

Avisos legales de otros fabricantes

Sentinel 5 contiene las siguientes tecnologías de otros fabricantes:

- Apache Axis y Apache Tomcat, Copyright © de 1999 a 2005, Apache Software Foundation. Para obtener más información y consultar las restricciones y renuncias de responsabilidad, visite http://www.apache.org/licenses/.
- ANTLR. Para obtener más información y consultar las restricciones y renuncias de responsabilidad, visite http://www.antlr.org/.
- Boost, Copyright © 1999, Boost.org.
- Bouncy Castle, Copyright © 2000-2004, the Legion of Bouncy Castle. Para obtener más información y consultar las restricciones y renuncias de responsabilidad, visite http://www.bouncycastle.org.
- Checkpoint. Copyright © Check Point Software Technologies Ltd.
- Concurrent, paquete de utilidades. Copyright © Doug Lea. Se utiliza sin las clases CopyOnWriteArrayList ni ConcurrentReaderHashMap.
- Crypto++ Compilation. Copyright © 1995-2003, Wei Dai, que incorpora los siguientes trabajos sujetos a copyright: mars.cpp de Brian Gladman y Sean Woods. Para obtener más información y consultar las restricciones y renuncias de responsabilidad, visite http://www.eskimo.com/~weidai/License.txt.
- Crystal Reports Developer y Crystal Reports Server. Copyright © 2004 Business Objects Software Limited.
- DataDirect Technologies Corp. Copyright © 1991-2003.
- edpFTPj, con licencia de Lesser GNU Public License. Para obtener más información y consultar las restricciones y renuncias de responsabilidad, visite http://www.enterprisedt.com/products/edtftpj/purchase.html.
- Enhydra Shark, con licencia de Lesser General Public License disponible en: http://shark.objectweb.org/license.html.
- ICEsoft ICEbrowser. ICEsoft Technologies, Inc. Copyright © 2003-2004.
- ILOG, Inc. Copyright © 1999-2004.
- Installshield Universal. Copyright © de 1996 a 2005, Macrovision Corporation y/o Macrovision Europe Ltd.
- Java 2 Platform, Standard Edition. Copyright © Sun Microsystems, Inc. Para obtener más información y consultar las restricciones y renuncias de responsabilidad, visite http://java.sun.com/j2se/1.4.2/j2re-1_4_2_10-license.txt.

  La plataforma Java 2 también contiene los siguientes productos de otros fabricantes:

  - CoolServlets © 1999
  - DES y 3xDES © 2000 de Jef Poskanzer
  - Crimson © 1999-2000 The Apache Software Foundation
  - Xalan J2 © 1999-2000 The Apache Software Foundation
  - NSIS 1.0j © 1999-2000 Nullsoft, Inc.

- Eastman Kodak Company © 1992
- Lucinda, marca comercial o marca comercial registrada de Bigelow and Holmes
- Taligent, Inc.
- IBM, algunas partes se encuentran disponibles en: http://oss.software.ibm.com/icu4j/

Para obtener más información acerca de estas tecnologías de otros fabricantes y consultar las restricciones y renuncias de responsabilidad correspondientes, visite: http://java.sun.com/j2se/1.4.2/j2se-1_4_2-thirdpartylicensereadme.txt.

- JavaBeans Activation Framework (JAF). Copyright © Sun Microsystems, Inc. Para obtener más información y consultar las restricciones y renuncias de responsabilidad, visite http://www.java.sun.com/products/javabeans/glasgow/jaf.html y haga clic en download > license.
- JavaMail. Copyright © Sun Microsystems, Inc. Para obtener más información y consultar las restricciones y renuncias de responsabilidad, visite http://www.java.sun.com/products/javamail/downloads/index.html y haga clic en download > license.
- Java Ace, de Douglas C. Schmidt y su grupo de investigación de la Universidad de Washington y Tao (con empaquetadores ACE) de Douglas C. Schmidt y su grupo de investigación en las universidades de Washington, California, Irvine y Vanderbilt. Copyright © de 1993 a 2005. Para obtener más información y consultar las restricciones y renuncias de responsabilidad, visite http://www.cs.wustl.edu/~schmidt/ACE-copying.html y http://www.cs.wustl.edu/~pjain/java/ace/JACE-copying.html.
- Módulos Java de servicios de autorización y autenticación, con licencia de Lesser General Public License. Para obtener más información y consultar las restricciones y renuncias de responsabilidad, visite http://free.tagish.net/jaas/index.jsp.
- Java Network Launching Protocol (JNLP). Copyright © Sun Microsystems, Inc. Para obtener más información y consultar las restricciones y renuncias de responsabilidad, visite http://www.java.sun.com/products/javawebstart/download-jnlp.html y haga clic en download > license.
- Java Service Wrapper. Partes con copyright como se indica a continuación: Copyright © 1999, 2004 Tanuki Software y Copyright © 2001 Silver Egg Technology. Para obtener más información y consultar las restricciones y renuncias de responsabilidad, visite http://wrapper.tanukisoftware.org/doc/english/license.html.
- JIDE. Copyright © de 2002 a 2005, JIDE Software, Inc.
- jTDS con licencia de Lesser GNU Public License. Para obtener más información y consultar las restricciones y renuncias de responsabilidad, visite http://jtds.sourceforge.net/.
- MDateSelector. Copyright © 2005, Martin Newstead, con licencia de Lesser General Public License. Para obtener más información y consultar las restricciones y renuncias de responsabilidad, visite http://web.ukonline.co.uk/mseries.
- Monarch Charts. Copyright © 2005, Singleton Labs.
- Net-SNMP. Partes del código están sujetas a copyright de varias entidades, las cuales se reservan todos los derechos. Copyright © 1989, 1991, 1992 de Carnegie Mellon University; Copyright © 1996, de 1998 a 2000, Junta de regentes de la Universidad de California; Copyright © de 2001 a 2003 Networks Associates Technology, Inc.; Copyright © de 2001 a 2003, Cambridge Broadband, Ltd.; Copyright © 2003 Sun Microsystems, Inc. y Copyright © de 2003 a 2004, Sparta, Inc. Para obtener más información y consultar las restricciones y renuncias de responsabilidad, visite http://net-snmp.sourceforge.net.
- The OpenSSL Project. Copyright © 1998-2004. the Open SSL Project. Para obtener más información y consultar las restricciones y renuncias de responsabilidad, visite http://www.openssl.org.
- Oracle Help for Java. Copyright © 1994-2006, Oracle Corporation.
- RoboHELP Office. Copyright © Adobe Systems Incorporated, antes conocido como Macromedia.

- Skin Look and Feel (SkinLF). Copyright © 2000-2006 L2FProd.com. Con la licencia Apache Software License. Para obtener más información y consultar las restricciones y renuncias de responsabilidad, visite https://skinlf.dev.java.net/.
- Sonic Software Corporation. Copyright © 2003-2004. El software de SSC contiene software de seguridad con licencia de RSA Security, Inc.
- Tinyxml. Para obtener más información y consultar las restricciones y renuncias de responsabilidad, visite http://grinninglizard.com/tinyxmldocs/index.html.
- SecurityNexus. Copyright © de 2003 a 2006. SecurityNexus, LLC. Reservados todos los derechos.
- Xalan y Xerces, ambos se otorgan con licencia de Apache Software Foundation Copyright © de 1999 a 2004. Para obtener más información y consultar las restricciones y renuncias de responsabilidad, visite http://xml.apache.org/dist/LICENSE.txt.
- yWorks. Copyright © de 2003 a 2006, yWorks.

NOTA: A la fecha de publicación de este documento, los enlaces indicados anteriormente están activos. En caso de que alguno de los enlaces anteriores esté dañado o la página a la que enlace esté inactiva, póngase en contacto con Novell, en la dirección Inc., 404 Wyman Street, Suite 500, Waltham, MA 02451 EE.UU.

Prólogo

La documentación técnica de Sentinel es una guía de referencia en la que se describen las funciones más generales. Esta documentación va dirigida a profesionales en seguridad de la información. El texto de esta documentación pretende servir como fuente de referencia para el sistema de gestión de seguridad empresarial de Novell. Existe documentación adicional en el portal Web de Novell.

La documentación técnica de Sentinel se divide en cinco volúmenes distintos. Son los siguientes:

- Volumen I: Guía de instalación de Sentinel™ 5
- Volumen II: Guía del usuario de Sentinel™ 5
- Volumen III: Guía del usuario del asistente de Sentinel™ 5
- Volumen IV: Guía de referencia del usuario de Sentinel™
- Volumen V: Guía de integración de productos de otros fabricantes en Sentinel

Volumen I: Guía de instalación de Sentinel

En esta guía se describe la instalación de:

- Servidor de Sentinel
- Consola de Sentinel
- Motor de correlación de Sentinel
- Crystal Reports de Sentinel
- Generador de recopiladores del asistente
- Gestor de recopiladores del asistente
- Asesor

Volumen II: Guía del usuario de Sentinel

En esta guía se tratan los temas siguientes:

- Funcionamiento de la consola de Sentinel
- Funciones de Sentinel
- Arquitectura de Sentinel
- Comunicación de Sentinel
- Apagado/inicio de Sentinel
- Valoración de vulnerabilidades
- Supervisión de eventos
- Filtrado de eventos
- Correlación de eventos
- Gestor de datos de Sentinel
- Configuración de eventos para relevancia empresarial
- Asignación de servicios
- Informes históricos
- Gestión del host del asistente
- Incidencias
- Casos
- Gestión del usuario
- Flujo de trabajo

Volumen III: Guía del usuario del asistente

En esta guía se tratan los temas siguientes:

- Funcionamiento del Generador de recopiladores del asistente
- Gestor de recopiladores del asistente
- Recopiladores
- Gestión del host del asistente
- Generación y mantenimiento de los recopiladores

Volumen IV: Guía de referencia del usuario de Sentinel

En esta guía se tratan los temas siguientes:

- Lenguaje de la secuencia de comandos del asistente
- Comandos de análisis del asistente
- Funciones de administrador del asistente
- Metaetiquetas de Sentinel y el asistente
- Permisos del usuario
- Motor de correlación de Sentinel
- Opciones de línea de comandos de correlaciones
- Esquema de la base de datos de Sentinel

Volumen V: Guía de integración de productos de otros fabricantes en Sentinel

- Remedy
- Operaciones de HP OpenView
- Servicio de asistencia técnica de HP

Contenido

7 Motor de correlación de Sentinel 7-1

1 Introducción a la Guía de referencia del usuario de Sentinel™ 5

NOTA: El término Agente puede intercambiarse con Recopilador. En adelante, los agentes se denominarán recopiladores.

La Guía de referencia del usuario de Sentinel se utiliza para consultar:

- Lenguaje para guiones del asistente
- Comandos de análisis del asistente
- Funciones de administrador del asistente
- Meta-etiquetas de Sentinel y el asistente
- Permisos de usuario de la consola de Sentinel
- Motor de correlación de Sentinel
- Opciones de línea de comandos de Sentinel
- Vistas de la base de datos del servidor de Sentinel

En esta guía se presupone que está familiarizado con la seguridad de red, la administración de la base de datos y con los sistemas operativos Windows y UNIX.

Contenido

Esta guía incluye los siguientes capítulos:

Convenciones usadas

Notas y precauciones

NOTA: Las notas proporcionan información adicional que puede resultar útil.

PRECAUCIÓN: Las precauciones proporcionan información adicional que puede ayudarle a evitar daños o pérdida de datos en su equipo.

Comandos

La fuente de los comandos es Courier. Por ejemplo:

```
useradd -g dba -d /export/home/oracle -m -s /bin/csh
   oracle
```

Otros materiales de consulta de Sentinel

Los manuales siguientes están disponibles en los CD de instalación de Sentinel.

- Guía de instalación de Sentinel™ 5
- Guía del usuario de Sentinel™ 5
- Guía del usuario del asistente de Sentinel™ 5
- Guía de referencia del usuario de Sentinel™ 5
- Guía de integración de productos de otros fabricantes en Sentinel™ 5
- Notas de revisión

Cómo ponerse en contacto con Novell

- Sitio Web: http://www.novell.com
- Asistencia técnica de Novell: http://www.novell.com/support/index.html
- Asistencia técnica internacional de Novell: http://support.novell.com/phone.html?sourceidint=suplnav4_phonesup
- Self Support (Autoasistencia técnica): http://support.novell.com/support_options.html?sourceidint=suplnav_supportprog
- Para obtener asistencia técnica las 24 horas del día los 7 días de la semana , llame al número 800-858-4000 (sólo para EE.UU.).

2 Lenguaje para guiones del asistente

NOTA: El término Agente puede intercambiarse con Recopilador. En adelante, los agentes se denominarán recopiladores.

En este capítulo y en el siguiente se describe cómo utilizar el lenguaje del asistente para generar guiones. Se analizan los operadores que se utilizan en las diversas cadenas y comandos de análisis de la generación de recopiladores.

Se tratan los temas siguientes:

Cadenas de decisión

Las cadenas distinguen entre mayúsculas y minúsculas.

A medida que se sondean los recopiladores, se recoge diversa información en el buffer de recepción interno. Las cadenas de tipo de decisión especifican que se tomará una decisión con respecto a los datos recibidos y almacenados en el buffer interno. Las cadenas de decisión se evalúan para determinar si son verdaderas o falsas. Si existe un error de sintaxis o si el cuadro Tipo de decisión se deja en blanco, la decisión es falsa.

La cadena de decisión sólo se evalúa si el Tipo de decisión se define en cadena o datos.

Manipulación del puntero Rx Buffer

Cada puerto del asistente posee su propio puntero Rx Buffer. El puntero Rx Buffer señala los bytes de datos en el buffer de recepción. Antes de cada cadena de decisión evaluada, el puntero Rx Buffer se restaura a su valor de retención, que normalmente es cero, excepto que sea modificado por una decisión que utilizó el operador de búsqueda (:).

- 0 no señala a ningún byte en el buffer de recepción.
- 1 señala al primer byte de datos, 2 al segundo byte de datos y así sucesivamente.

Formato

Una cadena de decisión adopta la forma de una secuencia de operadores lógicos (OL) y expresiones regulares.

No es necesario que aparezcan operadores lógicos y operadores de cadenas en cada secuencia. A continuación se mencionan algunas reglas con respecto a su uso:

- Los operadores lógicos generan expresiones booleanas (verdadero o falso) dentro de la cadena de decisión y se evalúan en función de la siguiente prioridad:

  ```
  ~ Not
  & And
  ```

- Un operador de cadena especifica una cadena de caracteres para buscar en el buffer de recepción. Con el operador de cadena se realiza una búsqueda de byte a byte desde la posición del puntero Rx Buffer en adelante.

NOTA: Dado que el cuadro Tipo de decisión se corta en el último carácter que se puede imprimir, se debe utilizar el equivalente hexadecimal de un espacio. El operador : no se puede utilizar con el operador NULL.

Nombres de parámetros

Para especificar un parámetro en una cadena de decisión, éste debe escribirse entre llaves ({ }). Cuando se genera el guión, el nombre del parámetro y las llaves se reemplazan con el valor del parámetro.

Si el nombre del parámetro especificado no existe en el archivo de parámetros a partir del cual se genera el guión, la expresión de nombre de parámetro y las llaves permanecen en los datos de la cadena de decisión.

Las expresiones de nombre de parámetro se pueden presentar en cualquier parte de la cadena de decisión. No pueden, sin embargo, anidarse (incluir otra expresión de nombre de parámetro en sí misma).

Jerarquía de operaciones en una cadena de decisión

Cada operación de una cadena de decisión se evalúa como verdadera (1) o falsa (0). Las operaciones de una cadena de decisión siempre respetan el orden que rige la sintaxis del operador lógico.

- Cuando se utiliza más de una operación, las evaluaciones de las cadenas se realizan de izquierda a derecha.
- Cuando se utilizan paréntesis, primero se evalúa el operador lógico de cada conjunto de paréntesis.
- Las siguientes operaciones lógicas que se deben evaluar son not (~), and (&).

También se sigue un orden de operación cuando se utiliza la sintaxis del operador de cadena:

- El puntero Rx buffer restablecido se evalúa primero.
- Todos los demás caracteres de sintaxis tienen la misma prioridad y se evalúan por orden, de izquierda a derecha.

Reglas del puntero Rx Buffer

Las reglas siguientes rigen el valor del puntero del buffer de recepción:

- Cuando la búsqueda de una cadena de caracteres es correcta, se considera que la búsqueda es verdadera y el puntero Rx Buffer se coloca en el primer byte de la cadena que se encontró.

  Cadena de decisión: DE

```
 A BCDE F GH
^

 A BCDE F GH
     ^
```

- Cuando la búsqueda de una cadena de caracteres no es correcta, se considera que la búsqueda es falsa y el puntero Rx Buffer se devuelve al valor de retención.

  Cadena de decisión: DEJ

  ```
   A BCDE F GH
  ^
   A BCDE F GH
  ^
  ```

Comprobación de un buffer de recepción vacío

Para comprobar un buffer de recepción vacío utilice la cadena de decisión siguiente:

```
NULL
```

Ejemplo de evaluaciones y resultados de una cadena de decisión

Cadenas de decisión alfanuméricas

Las siguientes son cadenas de decisión alfanuméricas para un buffer de recepción de muestra:

```
ABCDEFGHIJKLMNO (avance de línea) YZ<[&
```

Cadena de decisión	Expresión lógica	Resultado
A	1	1
P	0	0
\41\(HEX para A)	1	1
AB	1	1
\4142\(HEX para AB)	1	1
ABD	0	0
A&B	1 & 1	1
A&P	1 & 0	0
A+P	1 + 0	1
A\42\(HEX para B)	1	1
A&BC	1 & 1	1
DEF&ABC	1 & 0	0
ABC&DEF	1 & 1	1
ABC&BCD	1 & 1	1
ABC&ABC	1 & 0	0
\OA\(HEX para avance de línea)	1	1
NULL *	0	0

Si no se encuentran caracteres en el buffer de recepción, el resultado es TRUE.

Cadenas de decisión HEX

Las siguientes son cadenas de decisión HEX para un buffer de recepción de muestra (HEX):

```
02 0A 10 FF 1F 2E 3C 03
```

Cadena de decisión	Expresión lógica	Resultado
\020A\&\FF\	1 & 1	1
\02\	0	0
\02\&\03\	1 & 1	1
\03\&\02\	1 & 0	0

Expresiones regulares

Se utilizan caracteres y secuencias de caracteres especiales en la escritura de patrones para expresiones regulares.

Sentinel utiliza una biblioteca compatible con POSIX (Portable Operating System Interface for UNIX) para las expresiones regulares. POSIX es un conjunto de normas IEEE e ISO que permiten garantizar la compatibilidad entre sistemas operativos compatibles con POSIX, que incluye casi todas las variedades de UNIX.

Resumen de caracteres especiales para las expresiones regulares

La tabla siguiente resume los caracteres especiales que se pueden utilizar en las expresiones regulares para las funciones SEARCH y REPLACE.

Carácter	Uso/Ejemplo
\	Marca el carácter siguiente como especial. n coincide con el carácter “n”. La secuencia \n concuerda con un carácter de avance de línea o de nueva línea (final de la línea), pero para poder pasar el símbolo “\” por el analizador, se debe anteponer el carácter de excepción “/”; por lo tanto, para pasar \n, debe utilizar /\n.
^	Hace coincidir el inicio de una entrada o línea.
$	Hace coincidir el final de una entrada o línea.
*	Hace coincidir el carácter anterior ninguna o varias veces. go* coincide con “g” o “goo.”
+	Hace coincidir el carácter anterior una o más veces. go+ coincide con “goo” pero no con “g.”
?	Hace coincidir el carácter anterior una vez o ninguna. a?te? coincide con “te” en “eater”.
.	Hace coincidir cualquier carácter único excepto un carácter de línea nueva (final de línea).
x|y	Hace coincidir x o y. z|good? coincide con “goo” o “good” o “z”.
{n}	n es un entero no negativo. Coincide exactamente n veces. e{3} no concuerda con la “e” en “Ted", pero sí con las tres primeras “e” en “greeeeeed”.
{n,}	n es un entero no negativo. Coincide al menos n veces. e{3,} no concuerda con la “e” en “Ted” y concuerda con todas las “e” en “greeeeeed” e{1,} equivale a e+.
{n,m}	m y n son enteros no negativos. Coincide al menos n veces y como máximo m veces. e{1,3} concuerda con las tres primeras “e” en “greeeeeed”.

[xyz]	Un conjunto de caracteres. Concuerda con cualquiera de los caracteres entre corchetes. [xyz] coincide la “y” de “play”.
[^xyz]	Un conjunto de caracteres negativos. Concuerda con cualquier carácter que no esté entre corchetes. [^xyz]/ coincide la “v” en “vain”.
[0-9]	Concuerda con un carácter de dígito.
[^0-9]	Concuerda con un carácter que no sea un dígito.
[A-Za-z0-9_]	Concuerda con cualquier carácter de la palabra, incluido el guión bajo.
[^A-Za-z0-9_]	Concuerda con cualquier carácter que no pertenezca a la palabra.
/n/	Hace coincidir n, donde n es un valor de escape octal, hexadecimal o decimal. Permite incorporar códigos ASCII en expresiones regulares.

Espacios en blanco en las expresiones regulares

En las expresiones regulares, los espacios en blanco están formados por uno o más espacios vacíos, que puede ser uno de los caracteres siguientes:

Nombre simbólico	UCS	Descripción
<tabulación>	<U0009>	CHARACTER TABULATION (HT) (tabulación de caracteres)
<retorno de carro>	<U000D>	CARRIAGE RETURN (CR) (retorno de carro)
<línea nueva>	<U000A>	LINE FEED (LF) (avance de línea)
<tabulación vertical>	<U000B>	LINE TABULATION (VT) (tabulación de línea)
<avance de página>	<U000C>	FORM FEED (FF) (avance de página)
<espacio>	<U0020>	SPACE (espacio)

Comandos de análisis

El lenguaje de análisis del asistente está orientado por función. La mayoría de las funciones de análisis le permiten manipular las variables del asistente y su contenido. El lenguaje de análisis del asistente admite cuatro tipos de variables:

- Entero (el nombre de la variable comienza con i)
- Valores flotantes (el nombre de la variable comienza con f)
- Cadenas de longitud variable (el nombre de la variable comienza con cualquier letra que no sea i ni f)
- Matrices de variables (el nombre de la variable finaliza con [ ]). Los tipos de variables de matrices pueden ser matrices de enteros, valores flotantes o cadenas.

Estas variables son locales para cada puerto del asistente y no se comparten de manera global entre todos los puertos del asistente. Los comandos de análisis le permiten copiar datos del buffer de recepción a variables de cadenas.

El buffer de recepción contiene los datos que se han recibido del puerto de comunicación del asistente, puerto de zócalo, archivo o proceso.

La longitud de los bytes que se deben copiar, así como la posición desde donde se deben copiar los bytes, se puede controlar con los comandos de análisis siguientes:

- SEARCH()
- SKIP()
- SKIPWORD()
- NEGSEARCH()
- RESET()
- COPY()

Los datos del buffer de recepción se pueden añadir al final de una variable de cadena con el comando APPEND(). El lenguaje de análisis del asistente también le permite copiar o añadir al final datos de variables de cadena a otras variables de cadena.

Tipos de datos simples

número

Los numerales sólo pueden estar precedidos por los símbolos + o - en el caso de los comandos SKIP, SKIPWORD y SET. Por ejemplo:

```
0, 10, 2.5
```

ivar (Variables de enteros)

Las variables de enteros son números con signos de 32 bits. El nombre de la variable debe comenzar con I o i. Por ejemplo:

```
i_count, I_severity, i, i[55], i[index]
```

La variable de entero, i[55], es el índice 55 en la matriz de enteros, i[]. Asimismo, el índice de una matriz puede ser una variable de entero.

fvar (Variables de valores flotantes)

Las variables de valores flotantes son números de punto flotante de 32 bits. El nombre de la variable debe comenzar con F o f. Por ejemplo:

```
f_rate, F_queue, f, f[1], f[index]
```

svar (Variables de cadena)

Las variables de cadenas contienen cadenas de longitud de variable. Los nombres de variables de cadena no pueden comenzar con I, i, F o f. Por ejemplo:

```
resource, date, _message, string[1000], string[i_sev]
```

array (Matrices de variables)

Las matrices de variables pueden representar matrices de variables del tipo ivar, fvar y svar. Por ejemplo:

```
i_bits[], F_values[], s_resources[]
```

Las matrices se pueden indexar con cualquier índice numérico sin desperdiciar espacio de la memoria. El hecho de acceder a ivar[1000] no significa que se haya asignado memoria para 1.000 variables de enteros.

Una variable de matriz indexada se trata como cualquier otra variable (ivar, svar y fvar).

Por ejemplo, la siguiente sería la sintaxis legal para el comando POPUP:

```
POPUP(xterm_display[4], data[i_count])
```

Datos entre comillas

Los datos entre comillas se exploran y analizan de la siguiente manera:

- /=carácter de excepción: incluye el byte que le sigue a / independientemente de cualquier significado especial; para poder utilizar uno de los caracteres especiales de la cadena, se debe colocar / delante del carácter. Por ejemplo, corp/\router se utiliza para corp\router
- \xx x xx\=Datos hexadecimales (puede ser uno o dos caracteres por byte): \0ad\, \0a0d\, \a d\,\0a 0d\, y \0a d\ significan todos avance de línea/retorno de carro

Todos los demás caracteres se especifican directamente.

Tipos de datos de adición derivados

La tabla siguiente enumera los tipos de datos de adición derivados:

Tipo	Descripción
todos	número, ivar, fvar, svar, comillas
numérico	número, ivar, fvar, ivar[index], fvar[index]
cadena	svar, svar[index], comillas
variable	ivar, fvar, svar, ivar[index], fvar[index], svar[index]
numvar	ivar, fvar, ivar[index], fvar[index]
matriz	ivar[], fvar[], svar[]
matriz numvar	ivar[], fvar[]
matriz de variable de cadena	svar[]

Reglas especiales para variables

Las siguientes son reglas especiales para las variables.

- Los nombres de variables distinguen entre mayúsculas y minúsculas.
- Cuando se utiliza numvar por primera vez, excepto en los casos donde tiene el valor establecido, se define en cero.
- Cuando se utiliza svar por primera vez, excepto en los casos donde tiene el valor establecido, se define como nulo ("").
- Una matriz indexada se trata como cualquier otra variable de su tipo: ivar, fvar o svar.
- Para comentar uno o más comandos de análisis o colocar comentarios en textos de análisis, se deben escribir los comentarios entre /* */.

Por ejemplo:

```
/* esto es un comentario */
/* estos son comandos de análisis comentados
COPY(s: "test")
DISPLAY( )
*/
```

3 Comandos de análisis del asistente

NOTA: El término Agente puede intercambiarse con Recopilador. En adelante, los agentes se denominarán recopiladores.

En este capítulo, se enumeran los comandos de análisis del asistente utilizados en la generación de recopiladores en orden alfabético. A continuación, se incluye una lista de los comandos de análisis por función.

Función	**Comando de análisis**
Interacción de bases de datos	DBCLOSE DBDELETE DBGETROW DBINSERT DBOPEN DBSELECT
Depuración	BREAKPOINT DISPLAY POPUP
Interacción de archivos	FILEA FILEL FILER FILEW

Operaciones lógicas	COMPARE
	ELSE
	ENDFOR
	ENDIF
	ENDWHILE
	FOR
	IF
	LOOKUP
	WHILE

Interacción de redes	SOCKETW
Notificación	ALERT CLEARTAGS CONSTANTTAGS EVENT INDICATOR PAUSE

Función	Comando de análisis
Manipulación de datos sin formato	BITFIELD BYTEFIELD CONVERT CRC DECODE DELETE ENCODE ENDFOR HEXTONUM NUMTOHEX SETBYTES STRIP STRIP-ASCII-RANGE
Manipulación de cadenas	APPEND COPY COPY-FROM-RX-BUFF-UNTIL-SEARCH COPY-FROM-RX-BUFF COPY-FROM-STRING-TO-STRING-UNTIL-SEARCH COPY-STRING-TO-STRING LENGTH LENGTH-OPTION2 NEGSEARCH PARSER_ATTACHVARIABLE PARSER_CREATEBASIC PARSER_NEXT PARSER_PARSESTRING PRINTF REGEXPREPLACE REGEXPSEARCH REGEXPSEARCH_EXPLICIT REGEXPSEARCH REPLACE SEARCH SKIP SKIPWORD STONUM TOKENIZE TOLOWER TOUPPER TOKENIZE TRANSLATE
Utilidad	DATE DATETIME PAUSE SHELL TBOSSETCOMMAND TBOSSETREQUEST TIME

Función	Comando de análisis
Gestión de variables	CLEAR DELETE GETCONFIG GETENV INC RESET RXBUFF SET SETCONFIG
Análisis de vulnerabilidades	INFO_CLEARTAGS INFO_CLOSE INFO_CONSTANTTAGS INFO_CREATE INFO_DUMP INFO_PUSH INFO_SEND INFO_SETTAG

Formato de comandos y uso de matrices

Los formatos de los comandos de análisis utilizan determinados símbolos para expresar significados específicos. A continuación, se describen ejemplos de dichos símbolos:

Ejemplo de símbolo en uso	Ejemplo de significado del símbolo
`[parámetro]`	Los corchetes indican parámetros opcionales.
`<parámetro>`	Las llaves angulares indican los parámetros requeridos a proporcionar.
`a`	Debe introducir “a” literalmente aquí.
`a|b`	Utilice exactamente “a” o “b”, pero no las dos letras.
`<elemento> ::= <definición>`	Puede reemplazar el elemento con la definición.

`<varList>`
donde:
`<varList> ::= var [, <varList>]`
	Utilizado para las definiciones recursivas para describir una lista de variables en la que, al menos, una variable es necesaria.

`...`	Se permite repetir los parámetros anteriores.
`/`	La barra diagonal se utiliza como un “escape” para poder usar caracteres especiales, como la barra diagonal inversa (\).

Se permiten matrices en expresiones, por ejemplo:

Matriz determinada	Matrices equivalentes
`SET(i_var = 2)`	`i_arr[3]`

`SET(i_arr[3]=2`	`i_arr[i_var]`
	`i_arr[1+2]`
	`i_arr[1+1_var]`
	`i_arr[i_arr[3]]`

Comandos

ALERT

El comando ALERT envía mensajes de eventos a Sentinel.

- El primer parámetro necesario define el nombre del recurso.
- El segundo parámetro requerido define el texto del mensaje del evento.
- El tercer parámetro requerido define la gravedad del evento.
- La fecha y la hora del mensaje del evento pueden definirse como parámetros opcionales.
  - El parámetro de fecha puede utilizarse individualmente.
  - El parámetro de hora debe combinarse con el parámetro de fecha.

Formato

```
ALERT(resource, message, iseverity)
```

o

```
ALERT(resource, message, iseverity[, date[, time]])
```

No puede utilizar el parámetro de fecha, a menos que se combine con el parámetro de hora.

NOTA: Utilice el comando STONUM para convertir iseverity de una cadena a un número entero.

Tipos de datos

Argumento	Tipo	Descripción
resource	string (ENTRADA)	El recurso y, de manera opcional, el subrecurso, para enviar un evento (por ejemplo: `xterm:tcp_retransmits`).
message string	(ENTRADA)	El texto del mensaje para el mensaje del evento.
iseverity	numérico (ENTRADA)	La representación numérica de la prioridad de este mensaje de evento (0 - 5). 0 = carácter informativo 1 = carácter consultivo 2 = advertencia 3 = secundario 4 = importante 5 = crítico
date string	(ENTRADA) [OPCIONAL]	Define la fecha del mensaje de evento con el formato MM-DD-AAAA (por ejemplo: “12-01-2002”) (valor predeterminado = fecha actual).
time string	(ENTRADA) [OPCIONAL]	Define la hora del mensaje de evento con el formato HH:MM:SS (por ejemplo: “15:14:34”) (valor predeterminado = hora actual); debe utilizarse con el parámetro de fecha.

Por ejemplo:

```
ALERT("xterm:tcp_retransmits", mesg_txt,ivar[3])
ALERT("router_subnet_15", msg_txt, "c")
ALERT(resource, "El servidor no responde", iseverity)
ALERT("Mux184:card1", "C1 no funciona correctamente.", 4)
ALERT("Firewall", "Se ha perdido la conexión con el
   cortafuegos.", 5)
ALERT("CB5", "El banco de canales 5 se encuentra en
   mantenimiento", "Mant.")
ALERT(resource, message, isev, thedate, thetime)
ALERT("Switch3", oos_msg, 5, "07-30-1997", "07:03:23")
```

APPEND

El comando APPEND agrega datos del buffer de recepción, una variable de cadena o una cadena entrecomillada a una variable de cadena. Se aplica lo siguiente:

- Todos los parámetros APPEND son opcionales, excepto el parámetro de destino.
- El destino de los datos (variable de cadena) puede especificarse con los parámetros de APPEND.
- Puede especificarse un desplazamiento en el origen para controlar dónde se copian los datos a partir de los datos de origen.
- El número de bytes que se agregará a la variable de destino puede especificarse con el parámetro de longitud (ilen), o el valor predeterminado de la longitud será la longitud de los datos de origen.
- Además de especificar un parámetro de longitud numérico, puede utilizarse una cadena para definir la longitud.
- Si se utiliza una cadena como el parámetro de longitud, el parámetro de origen puede ser el buffer de recepción o una svar.
- Al utilizar una cadena como el parámetro de longitud, el motor del recopilador agrega bytes de los datos de origen (a partir del desplazamiento) a la variable de destino hasta, pero sin incluir, el primer carácter de la cadena (si se encuentra) (si no se encuentra la cadena, no se agrega ningún byte).
- Si los parámetros de desplazamiento o longitud se especifican a partir del rango de la variable de origen, se agregan todos los bytes posibles, hasta el final de los datos de origen.
- Si el desplazamiento es superior o igual a la longitud de los datos de origen, no se agrega ningún byte a la variable de destino (si no se especifica ningún desplazamiento, el valor predeterminado del desplazamiento es cero).

Formato

```
APPEND(<dest>: [source] [, [search] [, [ilen] [, [ioffset]
   ]]])
APPEND(<dest>: [source] [, [ilen] [, [ioffset] ]])
APPEND(<dest>: [ilen] [, [offset]])
```

Tipo de datos

Argumento	Tipo	Descripción
dest	svar (SALIDA)	La variable de cadena de datos a la que se agregan los bytes.
source	string (ENTRADA) [OPCIONAL] o svar	La cadena en la que se encuentran los bytes de origen que se agregarán a la cadena de destino. (valor predeterminado = buffer de recepción) Si se utiliza el parámetro de búsqueda.
search	string (ENTRADA) [OPCIONAL]	Cadena utilizada para especificar: la copia de los bytes que se buscarán en la cadena de origen.
ilen	numérico (ENTRADA) [OPCIONAL]	El número de bytes que se agregará del origen al destino.
ioffset	numérico (ENTRADA) [OPCIONAL]	El desplazamiento en el origen en el que se comenzará a agregar datos.

En los siguientes ejemplos, se agregan bytes del buffer de recepción a la svar de destino (dest). La posición del puntero Rx buffer se agrega al valor de desplazamiento para especificar la primera posición de los datos que se agregarán. El símbolo ^ indica la posición del puntero Rx buffer.

```
APPEND(svar:ilen)
APPEND(svar:3)
APPEND(svar:,ioffset)
APPEND(source:ilen,ioffset)
APPEND(svar: 10, 12)
```

Los ejemplos que se describen a continuación se basaron en los siguientes supuestos.

```
rxbuff="receive buffer"
^ (posición del puntero Rx buffer)
dest="A destination string"
source="A source string"
ilen=3
ioffset=3
```

Introduzca la información siguiente:

```
APPEND(dest:)
```

Resultado:

```
dest = "A destination stringreceive buffer"
```

O si ha introducido:

```
APPEND(dest:ilen)
```

Resultado:

```
dest = "A destination stringrec"
```

O si ha introducido:

```
APPEND(dest:,ioffset)
```

Resultado:

```
dest = "A destination stringreceive buffer"
```

En los siguientes ejemplos, se agregan bytes del buffer de recepción hasta, pero sin incluir, la cadena de búsqueda a la svar de destino (dest). Si no se encuentra la cadena de búsqueda en el buffer de recepción (después del puntero Rx buffer + la posición de desplazamiento), no se agrega ningún byte.

Introduzca la información siguiente:

```
APPEND(dest:,"buffer")
```

Resultado:

```
dest = "A destination stringreceive "
```

Introduzca la información siguiente:

```
APPEND(dest:,"buffer", 9)
```

Resultado:

```
dest = "A destination string"
```

En los siguientes ejemplos, se agrega una subcadena del buffer de recepción suponiendo que:

```
Rx Buffer = "Minor Alarm Firewall A"
```

Introduzca la información siguiente:

```
COPY(message:"Resource Name is: ")
APPEND(message:,6)
```

Resultado:

```
message = "Resource Name is: Alarm Firewall A"
```

BITFIELD

El comando BITFIELD convierte bytes en bits. Este comando convierte cada byte de una cadena de longitud arbitraria en 8 bits (0 ó 1) al colocarlos en una matriz de enteros, una matriz de valores flotantes o una cadena.

PRECAUCIÓN: La salida es 8 veces mayor que la entrada; por lo tanto, el comando de análisis bitfield puede afectar la capacidad de memoria, si se lo utiliza de modo

inadecuado. Por ejemplo, al utilizar cadenas de entrada que cuentan con un número de bytes demasiado grande.

Formato

```
BITFIELD(s_bytes, dest_var)
```

Tipos de datos

Argumento	Tipo	Descripción
s_bytes	string (ENTRADA)	Cualquier número de bytes hexadecimales o ASCII en una cadena.
dest_var	matriz numvar (SALIDA)	Matriz de enteros (definidos en 0 ó 1). El número de bits es equivalente al número de bytes en s_bytes por 8. Por cada conjunto de 8 bits, los bits se dividen en bits más importantes (MSB, Most Significant Bit) y bits menos importantes (LSB, Least Significant Bit). Por ejemplo: idest_var[0] = MSB de byte 1 idest_var[1] = MSB siguiente de byte 1 idest_var[2] = MSB siguiente de byte 1 idest_var[3] = MSB siguiente de byte 1 idest_var[4] = MSB siguiente de byte 1 idest_var[5] = MSB siguiente de byte 1 idest_var[6] = MSB siguiente de byte 1 idest_var[7] = LSB de byte 1 idest_var[8] = MSB de byte 2 idest_var[9] = MSB siguiente de byte 2 ... idest_var[n * 8 - 1] = LSB de byte n Una cadena que contiene un múltiplo de 8 bytes, donde cada byte representa un bit en los bytes de entrada. Los bytes de esta cadena siempre se definirán en un carácter ASCII 0 ó 1.
	O svar (SALIDA)	Para cada 8 bits consecutivos representados en cada cadena, los caracteres ASCII (0 y 1) se dividen en MSB y LSB. Por ejemplo: Si s_bytes = “\5AFE\” Entonces, dest_var= “0101101011111110”

NOTA: El segundo parámetro de bitfield (`dest_var`) debe ser una cadena (por ejemplo, `ivar[]` o `fvar[]`).

Por ejemplo:

```
BITFIELD("\00\", f_bit_array[])
BITFIELD(s_bytes, i_bit_array[])
BITFIELD(s_byte, string_out)
BITFIELD("Funcionará", i_bit_array[])
BITFIELD("\563F\", string_out)
```

En el siguiente ejemplo, la cadena sbyte está definida en un byte hexadecimal y se ha enviado al comando BITFIELD dos veces (una vez para una matriz de enteros y una vez para una cadena).

```
COPY(sbyte:"\AE\")
BITFIELD(sbyte, ibits[])
BITFIELD(sbyte, sbits)
```

Contenido de las variables de salida actuales

```
ibits[0] = 1
ibits[1] = 0
ibits[2] = 1
ibits[3] = 0
ibits[4] = 1
ibits[5] = 1
ibits[6] = 1
ibits[7] = 0
sbits = "10101110"
```

BREAKPOINT

El comando BREAKPOINT detiene la ejecución de un guión de análisis. Cuando se está ejecutando el depurador de guiones del asistente, el comando breakpoint detiene el analizador en espera de la intervención del usuario. Por ejemplo, en el panel de depurador del asistente, seleccione el botón Go (Ir) o Step (Paso a paso) para continuar con la depuración.

Formato

```
BREAKPOINT()
```

BYTEFIELD

El comando BYTEFIELD toma una representación binaria (0 ó 1) de bytes y los ubica en una variable de cadena.

La entrada puede ser una:

- cadena
- matriz de enteros
- matriz de valores flotantes

La salida es siempre una variable de cadena.

Formato

PRECAUCIÓN: Si el primer parámetro es una matriz de valores flotantes o enteros, no utilice valores superiores a 100 para i_num_bytes, ya que la matriz se inicializará para todas esas entradas (valores grandes de i_num_bytes podrían afectar la capacidad de memoria).

```
BYTEFIELD(source_var, s_bytes[, i_num_bytes])
```

NOTA: El primer parámetro de BYTEFIELD (`source_var`) debe ser `svar`, `ivar[]` o `fvar[]`.

Tipos de datos

Argumento	Tipo	Descripción
source_var	matriz numvar (ENTRADA)	Matriz de enteros (definidos en 0 ó 1). El número de bits es equivalente al número de bytes en s_bytes por 8. Por cada conjunto de 8 bits, los bits se dividen en bits más importantes (MSB, Most Significant Bit) y bit menos importantes (LSB, Least Significant Bit) (consulte los ejemplos que siguen a esta tabla).
	svar (ENTRADA)	Una cadena que contiene un múltiplo de 8 bytes, donde cada byte representa un bit en los bytes de entrada. Los bytes de esta cadena siempre deben definirse en un carácter ASCII 0 ó 1. Por cada 8 bits consecutivos representados en cada cadena, los caracteres ASCII (0 y 1) deben dividirse en MSB y LSB. Por ejemplo: Si source_var = “0101101011111110”, y i_num_bytes = 2, Entonces, s_bytes = “\5AFE\”
s_bytes	string (SALIDA)	Cualquier número de bytes de datos hexadecimales o ASCII en una cadena.
i_num_bytes	numérico (ENTRADA) [OPCIONAL]	El número de bytes que se colocará en los _bytes. Debido a que es opcional, el valor predeterminado es 1, a menos que se utilice cuando la entrada es de tipo CADENA. Si la entrada es de tipo CADENA, el valor predeterminado es el tamaño de la cadena dividido por 8.

A continuación, se describen ejemplos específicos de source_var:

```
ISOURCE_VAR[0] = MSB de byte 1
ISOURCE_VAR[1] = MSB siguiente de byte 1
ISOURCE_VAR[2] = MSB siguiente de byte 1
ISOURCE_VAR[3] = MSB siguiente de byte 1
ISOURCE_VAR[4] = MSB siguiente de byte 1
ISOURCE_VAR[5] = MSB siguiente de byte 1
ISOURCE_VAR[6] = MSB siguiente de byte 1
ISOURCE_VAR[7] = LSB de byte 1
ISOURCE_VAR[8] = MSB de byte 2
ISOURCE_VAR[9] = MSB siguiente de byte 2
...
ISOURCE_VAR[n * 8 - 1] = LSB de byte n
```

Algunos ejemplos de BYTEFIELD son los siguientes:

```
BYTEFIELD(i_bit_array[], s_bytes)
BYTEFIELD(string_bits_in, s_bytes)
BYTEFIELD(f_bit_array[], string_bytes, 2)
BYTEFIELD(i_bit_array[], string_bytes, i_num_bytes)
```

En el siguiente ejemplo, la cadena sbyte y la matriz de enteros ivar están definidas en una representación binaria de un byte hexadecimal y se han enviado al comando BYTEFIELD dos veces (una vez para la entrada de matriz de enteros y una vez para la entrada de cadenas).

```
SET(ivar[0] = 0)
SET(ivar[1] = 0)
SET(ivar[2] = 0)
SET(ivar[3] = 0)
SET(ivar[4] = 1)
SET(ivar[5] = 1)
SET(ivar[6] = 1)
SET(ivar[7] = 1)
COPY(sbits:"11110000")
BYTEFIELD(ivar[], sbyte1)
BYTEFIELD(sbits, sbyte2, 1)
```

Contenido de las variables de salida actuales:

```
sbyte1 = "\0F\"
sbyte2 = "\F0\"
```

CLEAR

El comando CLEAR convierte variables de cadena en cero bytes o define variables de enteros y de valores flotantes en cero. Se pueden especificar hasta 100 variables en un comando CLEAR.

Formato

```
CLEAR(<varlist>)
```

Donde:

```
varlist ::= var [, <varlist>]
Var ::= variable a eliminar (fvar, ivar o svar)
```

Número máximo de variables: 100

Tipos de datos

Argumento	Tipo	Descripción
var1	variable (ENTRADA/ SALIDA)	La variable a borrar (fvar, ivar o svar).
var2	variable (ENTRADA/ SALIDA) [OPCIONAL]	La variable a borrar (fvar, ivar o svar).
var3	variable (ENTRADA/ SALIDA) [OPCIONAL]	La variable a borrar (fvar, ivar o svar).
...	variable (ENTRADA/ SALIDA) [OPCIONAL]	Otras variables a borrar (fvar, ivar o svar).

Por ejemplo:

```
CLEAR(var1)
CLEAR(var1,var2)
CLEAR(var1,var2,var3)
CLEAR(svar[45])
CLEAR(imatrix[5][5])
CLEAR(ivar, fvar, i_len, data_string[i_var])
CLEAR(temp)
CLEAR(sdata[index_x][index_y])
CLEAR(f_bits[3], i_var_array[2])
CLEAR(i_counter, temp)
```

En los siguientes ejemplos, se asignan valores a las variables de cadena; se utilizan las variables de cadena en un mensaje de evento y se borran los valores de las variables de cadena.

```
COPY(res_var: "Cortafuegos")
COPY(msg_var: "Alarma secundaria de cortafuegos 116")
ALERT(res_var, msg_var, 4)
CLEAR(res_var, msg_var)
RESULTADO:
res_var = “”
msg_var = “”
```

CLEARTAGS

El comando CLEARTAGS ejecuta CLEAR en todas las variables reservadas de evento y variables reservadas de fecha / hora que no se encuentran protegidas por el comando CONSTANTTAGS.

Se debe llamar a este comando en el estado de inicialización (estado 4 en la plantilla estándar de Sentinel) del recopilador antes de analizar cualquier entrada en las variables reservadas.

El comando CLEARTAGS funciona en las variables de evento reservadas y en las variables reservadas de fecha / hora. El comando CLEARTAGS no necesita ningún parámetro. Las variables de cadena se definen en cadenas vacías “”; por ejemplo:

```
s_EVT y s_Sec.
```

La variable de enteros i_Severity se define en cero.

Formato

```
CLEARTAGS ()
```

Por ejemplo:

```
SET(i_Severity = 3)
COPY(s_BM:"Mensaje de base")
COPY(s_Example:"Prueba")
CLEARTAGS()
```

Resultado:

```
i_Severity = 0
s_BM = ""
s_Example = "Prueba"
```

NOTA: s_Example no es una variable reservada de fecha / hora, por eso, no se ejecutó CLEAR.

COMMENT

Este comando toma un argumento opcional, que representa una cadena. Se trata de un método para introducir comentarios en el archivo de plantilla del recopilador. Este comando permite introducir comentarios de un editor visual sin cambiar de editor de texto.

Formato

```
/*[cadena]*/
```

Por ejemplo:

```
/* INFORMACIÓN DEL RECOPILADOR
; ------------------------------------------------
Nombre_recopilador:             Plantilla estándar
Descripción_recopilador:        Plantilla sobre la que se
   basarán los nuevos recopiladores del asistente
Fabricante_recopilador:         N/D
Versión/Producto_recopilador:   N/D
Versión_recopilador:            versión 4.1
Fecha_recopilador:              Agosto de 2003
; ------------------------------------------------*/
```

COMPARE

El comando COMPARE examina dos argumentos y define una variable, según el resultado. El resultado de la comparación de tipo cadena o numérico puede almacenarse en una variable. Si la variable es de tipo ivar, fvar o de cadena, contendrá el valor -1, 0 ó 1.

- -1 se utiliza si arg1 es inferior a arg2.
- 0 se utiliza si arg1 es igual a arg2.
- 1 se utiliza si arg1 es superior a arg2.

Formato

```
COMPARE(arg1, arg2, dest)
```

Tipos de datos

Argumento	Tipo	Descripción
arg1	todos (ENTRADA)	Compare data 1. Debe ser una cadena o valores numéricos.
arg2	todos (ENTRADA)	Compare data 2. Debe ser del mismo tipo que Compare data 1.
dest	variable (SALIDA)	La variable en la que se ubicarán los resultados de compare: svar = “-1”, “0” ó “1” ivar = -1, 0 ó 1 fvar = -1.0, 0.0 ó 1.0

NOTA: Los dos tipos de arg1 y arg2 deben ser una cadena o valores numéricos.

Por ejemplo:

```
COMPARE(i_counter, 0, temp)
COMPARE(sdata, "ALM", i_sdata_cmp_val)
COMPARE(i_counter, i_counter2, temp)
COMPARE(i_counter, i_counter2, i_result[i_counter])
```

En el siguiente ejemplo, se compara el texto con el contenido de una variable de cadena, y el resultado de la comparación se almacena en la variable de entero. Se crea un evento si el texto no es el mismo que el valor de la variable de cadena.

```
COMPARE(s_data_var, "ALARMA", i_compare_var)
IF(i_compare_var = 0)
ALERT(res_var, "ALARMA importante", 5)
ENDIF()
```

NOTA: Los comandos IF( ), ELSE( ) y ENDIF( ) llevan a cabo la misma función que el comando COMPARE, salvo en la comparación de números negativos.

CONSTANTTAGS

El comando CONSTANTTAGS toma un número variable de parámetros de nombres de variables reservadas (evento y fecha / hora). Al determinar que una variable reservada es constante, evita que el comando CLEARTAGS la borre.

Un ejemplo de esta variable es s_PN, que lleva el nombre del producto que el recopilador está procesando. La variable s_PN debe determinarse constante y definirse una vez en el estado de instalación del recopilador.

Se debe llamar a este comando en el estado de instalación del recopilador (estado 1 en la plantilla estándar 4.1) para las variables reservadas que no se modifican cuando el recopilador procesa los eventos.

El comando CONSTANTTAGS funciona en las variables de evento reservadas y en las variables reservadas de fecha / hora.

Formato

```
CONSTANTTAGS (<variable_reservada> [, ...])
```

Tipos de datos

Argumento	Tipo	Descripción
reserved_variable		La lista de variables reservadas que se determinarán constantes y no se borrarán por el comando CLEARTAGS.

Por ejemplo:

```
COPY(s_PN:"PN")
COPY(s_ST:"ST")
COPY(s_BM:"BM")
CONSTANTTAGS(s_PN,s_ST)
CLEARTAGS()
```

Resultado:

```
s_PN = "PN"
s_ST = "ST"
s_BM = ""
```

De las tres variables de evento reservadas, CONSTANTTAGS no protegió a s_BM de CLEARTAGS; por lo tanto, se borró dicha variable.

CONVERT

El comando CONVERT transforma una cadena de entrada de tipo binario, octal, decimal, hexadecimal o sin formato en una variable de cadena de salida de tipo binario, octal, decimal, hexadecimal o sin formato.

Formato

```
CONVERT(string_in, type_in, svar_out, type_out)
```

Tipos de datos

Argumento	Tipo	Descripción
string_in	Cadena (ENTRADA)	La cadena de entrada que se convertirá.
type_in	Lista de selección Cadena Variable de cadena (ENTRADA)	El tipo de cadena de entrada (string_in): Binaria = “B” o “b” Octal = “O” u “o” Decimal = “D” o “d” Hexadecimal = “H” o “h” Sin formato = “R” o “r”
svar_out	svar (SALIDA)	La variable de cadena que contiene los datos de la cadena convertida.
type_out	Lista de selección Cadena Variable de cadena (ENTRADA)	El tipo al que se convertirán los datos (la cadena convertida se ubicará en svar_out): Binaria = “B” o “b” Octal = “O” u “o” Decimal = “D” o “d” Hexadecimal = “H” o “h” Sin formato = “R” o “r”

Por ejemplo:

```
CONVERT("10101010", "b", shex, "h")
CONVERT(sdata, "B", sraw, "r")
CONVERT("2356", "d", soctal, "o")
CONVERT("\3A\", "r", sbinary, "b")
CONVERT("2A3E", "h", sraw, "r")
CONVERT(data, "r", sdecimal, "d")
CONVERT(data, "o", shex, "H")
```

En el siguiente ejemplo, se llama al comando CONVERT para realizar diferentes conversiones.

```
CONVERT("\0afe\", "R", sdecimal, "D")
CONVERT("63", "d", sbinary, "b")
CONVERT("63", "d", shex, "h")
CONVERT("63", "d", soctal, "o")
CONVERT("1101010111110101", "b", sraw, "r")
```

Contenido de las variables de salida actuales:

```
sdecimal = "2814"
sbinary = "00111111"
shex = "3F"
soctal = "077"
sraw = "\d5 f5\"
```

COPY

El comando COPY duplica datos del buffer de recepción o de la cadena de origen, y los ubica en una variable de cadena o una cadena entrecomillada a una variable de cadena. El puntero Rx buffer no cambia al utilizar este comando.

El destino de los datos (svar) debe especificarse con los parámetros COPY.

> **NOTA:** En el Editor visual del Generador de recopiladores, COPY, COPY-FROM-RX-BUFF-UNTIL-SEARCH, COPY-FROM-RX-BUFF, COPY-FROM-STRING-TO-STRING-UNTIL-SEARCH y COPY-STRING-TO-STRING aparecen como comandos independientes. Se trata de los mismos comandos. Se proporcionan como descripciones de diferentes variaciones del mismo comando. Si va a utilizar alguna variación del comando COPY en el editor de texto, introducirá COPY.

Al utilizar este comando:

- Especifique un desplazamiento en el origen para controlar dónde se copian los datos a partir de los datos de origen.
- El número de bytes que se copiará a la variable de destino puede especificarse con el parámetro de longitud (ilen), o el valor predeterminado de la longitud puede ser la longitud de los datos de origen.
- Además de especificar un parámetro de longitud numérico, puede utilizarse una cadena. Al utilizar una cadena, el motor del recopilador copia bytes de los datos de origen (a partir del desplazamiento) en la variable de destino hasta, pero sin incluir, el primer carácter de la cadena (si se encuentra). Si no se encuentra la cadena, no se copia ningún byte.
- Si los parámetros de desplazamiento (ioffset) o de longitud (ilen) se especifican a partir del rango de la variable de origen, se copian todos los bytes posibles, hasta el final de los datos de origen.

Si el desplazamiento es superior o igual a la longitud de los datos de origen, no se copia ningún byte en la variable de destino.

Si no se especifica ningún desplazamiento, el valor predeterminado del desplazamiento es cero.

Formato

```
COPY(<dest>: [source] [, [search] [, [ilen] [, [ioffset]
   ]]])
COPY(<dest>: [source] [, [ilen] [, [ioffset] ]])
COPY(<dest>: [ilen] [, [offset]])
```

Tipos de datos

Argumento	Tipo	Descripción
dest	svar (SALIDA)	La variable de cadena de datos en la que se copiarán los bytes.
source string	(ENTRADA) [OPCIONAL] O svar	La cadena de la que se copian los datos (valor predeterminado = buffer de recepción). Si se utiliza el parámetro de búsqueda.
search	string (ENTRADA) [OPCIONAL]	Cadena utilizada para especificar: la copia de los bytes que se buscarán en la cadena de origen.
ilen	numérico (ENTRADA) [OPCIONAL]	El número de bytes que se copiará del origen al destino.
ioffset	numérico (ENTRADA) [OPCIONAL]	El desplazamiento en el origen en el que se comenzará a copiar datos; copia todos los caracteres del buffer de recepción al buffer de transmisión.

En los siguientes ejemplos, se copian bytes del buffer de recepción a la svar de destino (dest). La posición del puntero Rx buffer se agrega al valor de desplazamiento para especificar la primera posición de los datos que se copiarán. El símbolo ^ identifica la posición del puntero Rx buffer.

Estos ejemplos se basan en los siguientes supuestos:

```
rxbuff="receive buffer"
^ (posición del puntero Rx buffer)
dest=""
source="A source string"
ilen=3
ioffset=3
```

Comando	Resultado
`COPY(dest:)`	`dest = "receive buffer"`
`COPY(dest:5)`	`dest = "recei"`
`COPY(dest:,5)`	`dest = "ve.buffer"`

En los siguientes ejemplos, se copian bytes de una cadena de origen a la svar de destino (dest).

Comando	Resultado
`COPY(dest:source)`	`dest="A source string"`
`COPY(dest:source,5)`	`dest="A sou"`
`COPY(dest:source,5,6)`	`dest = "ce st"`

En los siguientes ejemplos, se copian bytes del buffer de recepción hasta, pero sin incluir, la cadena de búsqueda a la variable de cadena. Si no se encuentra la cadena de búsqueda en el buffer de recepción (después del puntero Rx buffer + la posición de desplazamiento), no se copia ningún byte.

NOTA: Para la sustitución de hexadecimales, \0000\ termina una cadena. Por lo tanto, "xxxx\0000\yyyy" se convierte en "xxxx".

En los siguientes ejemplos, se copian bytes del buffer de recepción hasta, pero sin incluir, la cadena de búsqueda a la svar de destino (dest). Si no se encuentra la cadena de búsqueda en el buffer de recepción (después del puntero Rx buffer + la posición de desplazamiento), no se copia ningún byte.

Comando	Resultado
`COPY(dest:,"buffer")`	`dest = "receive "`
`COPY(dest:,"receive")`	`dest = ""`

En los siguientes ejemplos, se copian bytes desde una cadena de origen (debe ser una variable de cadena) hasta, pero sin incluir, la cadena de búsqueda a la variable de cadena de destino (dest). Si no se encuentra la cadena de búsqueda en el buffer de recepción (después del puntero Rx buffer + la posición de desplazamiento), no se copia ningún byte.

Comando	Resultado
`COPY(dest:source," string")`	`dest = "a source"`
`COPY(dest:source," .string")`	`dest = ""`

CRC

El comando CRC calcula una comprobación de redundancia cíclica en una cadena de bytes (hexadecimal o ASCII).

Formato

```
CRC(source_data, dest_crc)
```

Tipo de datos

Argumento	Tipo	Descripción
source_data	string (ENTRADA)	Los datos de la cadena en los que se ejecutará el comando CRC.
dest_crc	svar (SALIDA)	La variable de cadena en la que se almacena el resultado de CRC de 2 bytes.

Por ejemplo:

En el siguiente ejemplo, el valor CRC calculado se compara con un valor guardado. Si los dos valores CRC son los mismos, se genera un mensaje de evento.

```
CRC(svar, s_crc_var)
IF(s_crc_var = "\0A5F\")
EVENT(res, "generado CRC correcto", 0)
ENDIF()
```

NOTA: Para la sustitución de hexadecimales, \0000\ termina una cadena; por lo tanto, “xxxx\0000\yyyy” se convierte en “xxxx”.

DATE

El comando DATE copia la fecha actual (con el formato MM-DD-AAAA) en una variable de cadena. De manera opcional, puede copiar el día de la semana actual en una variable de cadena, de enteros o de valores flotantes.

Formato

```
DATE(date_string [, day_of_week] [, i_day_of_week]
   [, f_day_of_week])
```

Tipo de datos

Argumento	Tipo	Descripción
date_string	svar (SALIDA)	La variable de cadena en la que se almacena la fecha (por ejemplo: svar = “11-18-2002”).
day_of_week	svar (SALIDA) [OPCIONAL] ivar (SALIDA) [OPCIONAL] O fvar (SALIDA) [OPCIONAL]	(De manera opcional) La variable de cadena en la que se almacena el día de la semana; escrito con el nombre completo del día (por ejemplo: svar = sábado) (De manera opcional) La variable de enteros o valores flotantes en la que se almacena el día de la semana; escrito con el nombre completo del día = número: lunes = 1 martes = 2 miércoles = 3 jueves = 4 viernes = 5 sábado = 6 domingo = 7 (por ejemplo: lunes es ivar = 1)

Por ejemplo:

En el siguiente ejemplo, la fecha del sistema se compara con una cadena de fecha. Si las dos fechas son las mismas, se genera un mensaje de evento.

```
DATE(date_var, day_of_week)
IF(date_var = "11-18-2002")
ALERT(res, "¡Feliz 23 aniversario!", 0)
ENDIF()
IF(day_of_week = "sábado")
ALERT(res, "Es hora de ir a la playa," 0)
ENDIF()
```

DATETIME

El comando DATETIME convierte una representación entera del número de segundos desde el 1 de enero de 1970 en las variables de cadena de fecha y hora. De manera opcional, puede copiar el día de la semana actual en una variable de cadena, de enteros o de valores flotantes.

Formato

```
DATETIME(itime_secs, svar_date, svar_time [, day_of_week]
   [, i_day_of_week] [, f_day_of_week])
```

Tipos de datos

Argumento	Tipo	Descripción
itime_secs	numérico (ENTRADA)	El número entero que contiene el número de segundos desde 1970.
svar_date	svar (SALIDA)	La variable de cadena en la que se almacena la fecha (por ejemplo: 02-19-96).
svar_time	svar (SALIDA)	La variable de cadena en la que se almacena la hora (por ejemplo: 15:14:33).
day_of_week	svar (SALIDA) [OPCIONAL] ivar (SALIDA) [OPCIONAL] O fvar (SALIDA) [OPCIONAL]	(De manera opcional) La variable de cadena en la que se almacena el día de la semana; escrito con el nombre completo del día (por ejemplo: svar = sábado) (De manera opcional) La variable de enteros o valores flotantes en la que se almacena el día de la semana; escrito con el nombre completo del día = número: lunes = 1 martes = 2 miércoles = 3 jueves = 4 viernes = 5 sábado = 6 domingo = 7 (por ejemplo: lunes es ivar = 1)

Por ejemplo:

En el siguiente ejemplo, el comando DATETIME convierte el número de segundos desde 1970 en variables de fecha y hora:

```
DATETIME(0, sdatevar, stimevar)
```

En el siguiente ejemplo, el comando DATETIME ofrece el día de la semana, así como la fecha y la hora:

```
DATETIME(946728000, sdate, stime, sday)
```

Contenido de las variables de salida actuales:

```
sdatevar = "01-01-70"
stimevar = "00:00:00"
sdate = "01-01-2000"
stime = "12:00:00"
sday = "sábado"
```

DBCLOSE

El comando DBCLOSE cierra la conexión con la base de datos. Requiere dos parámetros.

- El primer parámetro requerido es la referencia de la base de datos que devuelve el comando DBOPEN. Se trata de un número entero o de una variable de enteros.
- El segundo parámetro requerido es el estado del cierre. Se trata de una variable de enteros o de valores flotantes. Se devuelve un “1” al finalizar correctamente la operación.

Formato

```
DBCLOSE(i_dbhandle, i_closestatus)
```

DBDELETE

El comando DBDELETE elimina filas de la tabla seleccionada según los criterios de selección. Requiere cuatro parámetros.

- El primer parámetro requerido es la referencia de la base de datos que devuelve el comando DBOPEN. Se trata de un número entero o de una variable de enteros.
- El segundo parámetro requerido es el estado de la eliminación. Se trata de una variable de enteros o de valores flotantes. El número de filas eliminadas se devuelve al finalizar la operación correctamente, incluso 0.
- El tercer parámetro requerido es el nombre de la tabla de la que se eliminarán las filas. Puede ser una cadena o una variable de cadena.
- El cuarto parámetro opcional es la cláusula where (donde). Permite a los usuarios filtrar los datos no deseados por un criterio de selección. Si se deja en blanco, la eliminación borrará todas las filas de la tabla.

Los códigos de error para el comando DBDELETE son los siguientes:

```
>0No existe error
0No se eliminó ninguna fila
-1Referencia de BD no válido
```

Formato

```
DBDELETE(i_dbhandle, i_deletestatus, "nombre de tabla",
   "cláusula where")
```

Por ejemplo:

```
DBDELETE(i_dbhandle, i_deletestatus, "nombre de tabla")
DBDELETE(i_dbhandle, i_deletestatus, s_tablename, "cláusula
   where")
```

DBGETROW

El comando DBGETROW funciona junto con el comando DBSELECT. El usuario primero debe obtener una selección, mediante DBSELECT, antes de recuperar las filas con el comando DBGETROW. Este comando recuperará la siguiente fila disponible de una selección, deja el cursor abierto, para que este comando pueda llamarse en un bucle, y recuperar la siguiente fila en cada llamada. Requiere cuatro parámetros.

- El primer parámetro requerido es la referencia de la base de datos que devuelve el comando DBOPEN. Se trata de un número entero o una variable de enteros.
- El segundo parámetro requerido es la referencia para la selección. Puede ser una cadena o una variable de cadena. Se trata de la misma referencia que se asignó durante el comando DBSELECT.
- El tercer parámetro requerido es el estado de la obtención. Se trata de una variable de enteros o de valores flotantes. Se devuelve un “1” al finalizar correctamente la operación.
- El cuarto parámetro requerido y los parámetros opcionales siguientes son los datos de la columna que devuelve el comando. Estas columnas pueden ser variables de cadena, variables de valores flotantes o variables de enteros. Los datos de una columna de un tipo diferente al tipo de parámetro se convierten en el tipo de parámetro adecuado, si es posible. Por lo tanto, si la tabla contiene una columna de valores flotantes, pero el parámetro es una cadena, los datos se convierten de un valor flotante a una cadena. El usuario puede incluir hasta 48 de estos parámetros.

NOTA: El comando completará el menor número de parámetros definidos y el número de columnas actuales en la base de datos. Si la base de datos contiene 4 columnas, pero el usuario proporciona 7 de estos parámetros, sólo se completarán los primeros cuatro.

Los códigos de error para el comando DBGETROW son los siguientes:

```
1No existe error
-1Error al recuperar la fila
```

Formato

```
DBGETROW(i_dbhandle, "selección 1", i_selectstatus, s_col1,
   s_col2, s_col3, ..., s_col48)
```

Por ejemplo:

```
DBGETROW(i_dbhandle, s_selecthandle, i_selectstatus,
   s_col1, s_col2)
```

DBINSERT

El comando DBINSERT inserta una fila de datos en la base de datos para una tabla seleccionada. Requiere cuatro parámetros.

- El primer parámetro requerido es la referencia de la base de datos que devuelve el comando DBOPEN. Se trata de un número entero o de una variable de enteros.
- El segundo parámetro requerido es el estado de la inserción. Se trata de una variable de enteros o de valores flotantes. Se devuelve un "1" al finalizar correctamente la operación.
- El tercer parámetro requerido es el nombre de la tabla en la que se insertarán los datos.
- El cuarto parámetro requerido y los parámetros opcionales siguientes son los datos de las columnas que se insertarán. Las columnas pueden ser de cualquier tipo. El usuario puede incluir hasta 48 de estos parámetros.

El comando debe incluir el número exacto de parámetros necesarios para insertar una fila de datos. DBINSERT no agregará un nuevo registro si se viola una sola restricción.

Los códigos de error para el comando DBINSERT son los siguientes:

```
1 No existe error
-1 referencia de BD no válido / no se ha insertado ninguna
   fila
-2 No se puede crear la solicitud de datos
-7 Error de ejecución de SQL
-16 Error de sintaxis de SQL
```

Formato

```
DBINSERT(i_dbhandle, i_insertstatus, "el nombre de tabla",
   "datos1", "datos2", ..., "datos48")
```

Por ejemplo:

```
DBINSERT(i_dbhandle, i_insertstatus, s_theTableName,
   "datos1", I_data2, f_data3)
DBINSERT(i_dbhandle, i_insertstatus, "el nombre de tabla",
   s_data1, "datos2")
```

DBOPEN

El comando DBOPEN establece una conexión con una base de datos admitida.

Sólo en el recopilador de Microsoft Windows NT, DBOPEN no funcionará cuando el nombre de la base de datos se configure para indicar una “unidad asignada”. Debido a que el recopilador funciona como un servicio, suele ejecutarse bajo la cuenta “system”. Esta cuenta no dispone de permisos para acceder al uso compartido remoto, incluidas las unidades asignadas. Esto significa que cada conexión con la base de datos (incluso mediante OBDC) en un recopilador de Windows debe ser una base de datos totalmente local.

Requiere cinco parámetros.

- El primer parámetro requerido es el tipo de base de datos. Puede seleccionarse mediante una lista de selección o utilizando una cadena o una variable de cadena. El valor aceptable para este parámetro es Oracle9i.
- El segundo parámetro requerido es el nombre de la base de datos con la que se conectará. Puede ser una cadena o una variable de cadena.
- El tercer parámetro requerido es el nombre del usuario para la base de datos. Puede ser una cadena o una variable de cadena. Este campo puede contener cualquier tipo de texto, si los usuarios no se han configurado específicamente para acceder a la base de datos.
- El cuarto parámetro requerido es la contraseña para el usuario. Puede ser una cadena o una variable de cadena. Este campo puede contener cualquier tipo de texto, si los usuarios no se han configurado específicamente para acceder a la base de datos.
- El quinto parámetro requerido es la referencia de la base de datos, que devuelve este comando en la variable de enteros o en la variable de valores flotantes. La referencia de la base de datos es superior a 0 al finalizar la operación correctamente.

Formato

```
DBOPEN("oracle9i", "Nombre de base de datos", "nombre de
   usuario", "contraseña", i_dbhandle)
```

Por ejemplo:

```
DBOPEN(s_dbtype, s_dbname, s_username, s_password,
   i_dbhandle)
DBOPEN(s_dbtype, "nombre bd", s_username, "contraseña",
   i_dbhandle)
```

DBSELECT

El comando DBSELECT funciona junto con el comando DBGETROW. El comando DBSELECT abre un cursor de selección en la base de datos. Captura una instantánea de los registros actuales en la base de datos que cumplen los criterios de selección. Los registros introducidos después del comando DBSELECT no aparecerán en recuperaciones de registros hasta que se utilice otro comando DBSELECT para actualizar la selección.

Requiere siete parámetros.

- El primer parámetro requerido es la referencia de la base de datos que devuelve el comando DBOPEN. Se trata de un número entero o de una variable de enteros.
- El segundo parámetro requerido es el estado de la selección. Se trata de una variable de enteros o de valores flotantes. Se devuelve un "1" al finalizar correctamente la operación.
- El tercer parámetro requerido es el identificador de la selección. Puede ser una cadena o una variable de cadena. Debe ser único, si cuenta con más de un comando DBSELECT.
- El cuarto parámetro requerido es el número de filas que se omitirá después de realizar la selección. Permite al usuario ubicar el puntero en el comando DBGETROW para señalar los datos nuevos, y permite que se omitan los datos anteriores. Puede ser un número entero o una variable de enteros.
- El quinto parámetro requerido es la tabla de la que se obtendrán los datos. Puede ser una cadena o una variable de cadena.
- El sexto parámetro opcional es la cláusula where (donde). Permite a los usuarios filtrar los datos no deseados por un criterio de selección. Si se deja en blanco, la selección contendrá todas las filas de la tabla. El formato de la cláusula where (donde) es: where nombre-columna='datos'.
- El séptimo parámetro opcional pertenece a las columnas que devuelve el comando DBSELECT. Si se deja en blanco, la selección contendrá todas las columnas de la tabla.

Los códigos de error para el comando DBSELECT son los siguientes:

```
1 No existe error
-1 Identificador de BD no válido
-2 No se puede crear la solicitud de datos
-3 Error en la configuración de confirmación automática
-4 Error de asignación de memoria
-5 Error de sintaxis de SQL
-6 Error de ejecución de SQL
```

Formato

```
DBSELECT( i_dbhandle, i_selectstatus, "selección 1",
   i_rows_to_skip, "f_atom"<, "cláusula where"><,
   "columna1<columna2><...>">)
```

Por ejemplo:

```
DBSELECT(i_dbhandle, i_selectstatus, "selección1",
   i_rows_to_skip, "f_atom")
DBSELECT(i_dbhandle, i_selectstatus, s_select1, 23,
   S_TABLENAME, s_whereclause)
DBSELECT(i_dbhandle, i_selectstatus, s_select1, 23,
   S_TABLENAME, "where fname='BOB'")
DBSELECT(i_dbhandle, i_selectstatus, s_select1, 23,
   S_TABLENAME, "where fname='BOB'", "NOMBRE, APELLIDO,
   DIRECCIÓN")
```

DEC

El comando DEC disminuye una variable numérica en 1. Al utilizar DEC, debe especificar una ivar o una fvar.

Formato

```
DEC(i_numvar)
```

Tipos de datos

Argumento	Tipo	Descripción
i_numvar	numvar (ENTRADA/ SALIDA)	La variable que se disminuirá (ivar o fvar).

Por ejemplo:

```
SET(icounter = 2)
DEC(icounter)
DEC(icounter)
```

Resultado:

```
icounter = 0
```

DECODE

El comando DECODE recupera una cadena que se codificó para conservar la identificación del paquete. Este comando identifica los bytes (o caracteres) de coincidencia y los bytes (o caracteres) de escape para eliminar el carácter de escape. Elimina cada instancia de la cadena de escape que precede a los bytes de coincidencia cada vez que se encuentra en los datos.

Formato

```
DECODE(data_decode, match, escape)
```

Tipos de datos

Argumento	Tipo	Descripción
data_decode	svar (ENTRADA/ SALIDA)	La variable de cadena de datos que se decodificará. El resultado decodificado se vuelve a ubicar en esta variable.
match	string (ENTRADA)	La cadena de bytes que coincidirá en la variable de cadena data_decode.
escape	string (ENTRADA)	La cadena de escape que se eliminará de la variable data_decode.

Por ejemplo:

En el siguiente ejemplo, se codifica una cadena, se copia para guardar la versión codificada, y se decodifica con los mismos parámetros.

```
COPY(svar:"Esto es una prueba de decodificación")
ENCODE(svar, " ", "\00\")
COPY(svar_encode:svar)
DECODE(svar, " ", "\00\")
```

Contenido de las variables de salida actuales:

```
svar = "Esto es una prueba de decodificación")
svar_encode = "Esto\00\ es\00\ una\00\ prueba\00\ de\00\
   decodificación"
```

DECODEMIME

El comando DECODEMIME permite al usuario decodificar una cadena o una variable de cadena codificada base 64 mediante la decodificación base 64, y permite almacenar la cadena decodificada resultante en una variable de cadena. Si se produce un error, la longitud de la cadena de datos resultante sería cero y el evento de la variable de números opcional de define en 0. Si la decodificación se lleva a cabo correctamente, el evento de la variable de números se define en 1.

Formato

```
DECODEMIME(encoded_data, data, success)
```

Tipos de datos

Argumento	Tipo	Descripción
encoded_data	Cadena o variable de cadena (ENTRADA)	La cadena codificada base 64 que se decodificará.
data	Variable de cadena (SALIDA)	Datos decodificados resultantes.
success	Variable de enteros o de valores flotantes (SALIDA) [OPCIONAL]	Se define en uno si la decodificación se lleva a cabo correctamente. Si se produce un error, se define en cero.

Por ejemplo:

```
DECODEMIME("VGVzdGluZyBEYXRhIEVuY29kaW5n", s_data,
   i_success)
```

En el ejemplo anterior, el comando DECODEMIME decodifica la cadena entre comillas dobles mediante la decodificación base 64 y almacena la cadena decodificada resultante en s_data. S_data se rellena con los siguientes datos:

```
test encode64 command
```

Como la decodificación se llevó a cabo correctamente, se asigna 1 a la variable de enteros i_success.

Consulte el comando ENCODEMIME.

DELETE

El comando DELETE elimina variables del sistema para liberar memoria asignada para su almacenamiento (se utiliza principalmente para variables de cadena).

Se recomienda eliminar svars cuando haya finalizado, para ahorrar memoria. Se pueden especificar hasta 100 variables en un comando DELETE.

Formato

```
DELETE(<varlist>)
```

Donde:

```
varlist ::= var [, <varlist>]
Var ::= variable a eliminar (fvar, ivar o svar)
```

Número máximo de variables: 100

Tipos de datos

Argumento	Tipo	Descripción
var1	variable (ENTRADA/ SALIDA)	La variable a eliminar (fvar, ivar o svar).
var2	variable (ENTRADA/ SALIDA) [OPCIONAL]	La variable a eliminar (fvar, ivar o svar).
var3	variable (ENTRADA/ SALIDA) [OPCIONAL]	La variable a eliminar (fvar, ivar o svar).
...	variable (ENTRADA/ SALIDA) [OPCIONAL]	Otras variables a eliminar (fvar, ivar o svar).

Por ejemplo:

```
DELETE(ivar1)
DELETE(sdata, i_len, i_count, svar[22])
DELETE(imatrix3d[ix][iy][iz])
DELETE(f_array[i_count], svar[4], sdata)
DELETE(ichart[3][icount])
```

DISPLAY

El comando DISPLAY muestra las variables de cadena y sus valores actuales en una ventana emergente.

Tiene las opciones siguientes:

- Utilizarlo al depurar guiones.
- Si transfiere una cadena como parámetro, muestra el contenido de esa cadena.
- Las cadenas que contienen datos hexadecimales se muestran en formato hexadecimal (es decir, cadena="\0a 0d\").

En primer lugar, el programa intenta mostrar la cadena en ASCII. Si la cadena contiene datos hexadecimales imprimibles y no imprimibles, los caracteres hexadecimales imprimibles se muestran en ASCII y el resto de la cadena se muestra en formato hexadecimal. Para la sustitución de hexadecimales, \0000\ termina una cadena; por lo tanto, "xxxx\0000\yyyy" se convierte en "xxxx".

Formato

```
DISPLAY(string_data)
```

Tipos de datos

Argumento	Tipo	Descripción
string_data	string (ENTRADA) [OPCIONAL]	Cualquier cadena en particular a mostrar. Si deja este comando desactivado, se muestra el contenido de todas las variables (cadenas, números y matrices) para cada guión.

Por ejemplo:

```
DISPLAY( )
DISPLAY(sdata_var)
DISPLAY("Hola, éstos son los datos de la cadena")
DISPLAY(sdata_var)
```

ELSE

El comando ELSE marca el final de la parte verdadera del comando anterior if() relacionado. Los comandos de análisis después de ELSE() se ejecutan si el resultado del comando IF() es FALSO. Los comandos se ejecutan hasta el siguiente comando ENDIF() correspondiente.

Formato

```
ELSE()
```

Por ejemplo:

```
IF(i = 10)
ALERT("I es 10")
ELSE()
ALERT("I no es 10")
ENDIF()
```

No puede realizar una comparación directamente con un número negativo. Para ello, lleve a cabo uno de estos dos métodos:

- Utilice la función de análisis compare
- Realice una comparación indirectamente, según se muestra a continuación:

```
SET(i_compare_val=-10)
IF(ivar > i_compare_val)
ALERT("ivar es superior a -10")
endif()
```

ENCODE

Utilice el comando ENCODE para conservar la identificación del paquete. Este comando compara bytes (o caracteres) en datos y antepone aquellos bytes coincidentes con una cadena de escape. La cadena de escape se ubica delante de los bytes coincidentes donde se encuentren tales caracteres en los datos.

Formato

```
ENCODE(data_encode, match, escape)
```

Tipos de datos

Argumento	Tipo	Descripción
data_encode	svar (ENTRADA/ SALIDA)	La variable de cadena de datos que se codificará. El resultado codificado se vuelve a ubicar en esta variable.
match	string (ENTRADA)	La cadena de bytes que coincidirá en la variable data_encode.
escape	string (ENTRADA)	La cadena de escape que se ubicará delante de cada byte coincidente en la variable data_encode.

Por ejemplo:

En el siguiente ejemplo, dos cadenas de datos se codifican para anteponer todos los espacios con “#” y otra para anteponer todas las ‘t’ y las ‘p’ con “!!”.

```
COPY(data:"Anteponer todos los espacios con '#'")
ENCODE(data, " ", "#")
COPY(svar:"Anteponer 't' y 'p' con '!!'")
ENCODE(svar, "tp", "!!")
```

Resultado:

```
data = "Anteponer# todos# los# espacios# con# '#'")
svar = "Anteponer '!!t y !!p con '!!'"
```

ENCODEMIME

El comando ENCODEMIME permite al usuario codificar una cadena o una variable de cadena mediante la codificación base 64, y permite almacenar la cadena codificada resultante en una variable de cadena.

Formato

```
ENCODEMIME(data, encoded_data)
```

Tipos de datos

Argumento	Tipo	Descripción
data	Cadena o variable de cadena (ENTRADA)	La cadena de datos que se codificará.
encoded_data	Variable de cadena (SALIDA)	Datos codificados resultantes.

Por ejemplo:

```
COPY(s_data:"test encode64 command")
ENCODEMIME(s_data, s_encd_data)
```

En el ejemplo anterior, el comando ENCODEMIME codifica la cadena en la variable s_data mediante la codificación base 64 y almacena la cadena codificada resultante en s_encd_data. S_encd_data se rellena con los siguientes datos:

```
VGVzdGluZyBEYXRhIEVuY29kaW5n
```

Consulte el comando DECODEMIME.

ENDFOR

El comando ENDFOR marca el final del bloque anterior for().

Formato

```
ENDFOR()
Ejemplo
```

```
FOR(i=0,i<3,i=i+1)
ALERT("Todavía en el bucle")
ENDFOR()
```

ENDIF

El comando ENDIF marca el final del bloque anterior if().

Formato

```
ENDIF()
```

Por ejemplo:

```
IF(i = 10)
ALERT("I es 10")
ELSE()
ALERT("I no es 10")
ENDIF()
```

No puede realizar una comparación directamente con un número negativo. Para ello, lleve a cabo uno de estos métodos:

- Utilice la función de análisis compare
- Realice una comparación indirectamente, según se muestra a continuación:

```
SET(i_compare_val=-10)
IF(ivar >i_compare_val)
ALERT("ivar es superior a -10")
ENDIF()
```

ENDWHILE

El comando ENDWHILE marca el final del bloque anterior while().

Formato

```
ENDWHILE()
Ejemplo
WHILE(i<3)
SET(i=i+1)
ENDWHILE()
```

EVENT

El comando EVENT crea y envía un mensaje de alerta. No necesita ningún parámetro. El comando EVENT automáticamente crea el mensaje de alerta con el contenido de las variables reservadas.

La mayoría de las variables reservadas se asigna directamente a las meta-etiquetas de la plantilla del asistente versión 3.2. Sólo se envían las variables que se utilizan en el guión y que no se definen como "". Las variables, como i_Severity y c_Res, son necesarias para que el Gestor de recopiladores procese un mensaje de alerta.

Variables de evento reservadas

NOTA: Cuando una etiqueta es precedida de una 'e.', como e.crt, se refiere a eventos actuales. Si una etiqueta es precedida de una 'w.', como w.crt, se refiere a eventos históricos.

Variable	Descripción breve	Se asigna a meta-etiqueta (etiqueta)
s_BM	Mensaje de base	Message (msg)
i_Severity	Gravedad	Severity (sev)
s_Res	Recurso	Resource (res)
s_SubRes	Subrecurso	SubResource (sres)
s_ET	Hora de evento	EventTime (et)
s_P	Protocolo	Protocol (prot)
s_DP	Puerto de destino	DestinationPort (dp)
s_SP	Puerto de origen	SourcePort (sp)
s_EVT	Nombre de evento	EventName (evt)
s_SN	Nombre de sensor	SensorName (sn)
s_SIP	IP de origen	Source IP (sip)
s_DIP	IP de destino	DestinationIP (dip)
s_SHN	Nombre de host de origen	SourceHostName (shn)
s_DHN	Nombre de host de destino	DestinationHostName (dhn)
s_SUN	Nombre de usuario de origen	SourceUserName (sun)
s_DUN	Nombre de usuario de destino	DestinationUserName (dun)
s_FN	Nombre de archivo	FileName (fn)
s_EI	Información ampliada	ExtendedInformation (ei)
s_RN	Nombre de informador	ReporterName (rn)
s_ST	Tipo de sensor	Sensor Type (st)
s_PN	Nombre de producto	ProductName (pn)
s_CRIT	Importancia	Criticality (crt)
s_VULN	Vulnerabilidad	Vulnerability (vul)
s_CT1	Reservado a cliente 1	Ct1 (ct1)
s_CT2	Reservado a cliente 2	Ct2 (ct2)
s_CT3	Reservado a cliente 3	Ct3 (ct3)
s_RT1	Nombre de dispositivo de ataque (Reservado para Sentinel 1)	Rt1 (rt1)
s_RT2	Reservado para Sentinel 2	Rt2 (rt2)
s_RT3	Reservado para Sentinel 3	Rt3 (rt3)

Variable	Descripción breve	Se asigna a meta-etiqueta (etiqueta)
s_CV1 a s_CV100	Variable de cliente 1 a 100 **NOTA**: 1 a 10 es de tipo long (número) 11 a 20 es de tipo fecha 21 a 100 es de tipo cadena	Cv1 a Cv100 (cv1 a cv100)
s_RV1 a s_RV29	Valor reservado 1 a 29 **NOTA**: Reservado para uso de Novell.	Rv1 a Rv31 (rv1 a rv29)
s_RV30	AttackId	Rv30
s_RV31	DeviceName	Rv31
s_RV32	DeviceCategory	Rv32 (rv32)
s_RV33	EventContext	Rv33 (rv33)
s_RV34	SourceThreatLevel	Rv34 (rv34)
s_RV35	SourceUserContext	Rv35 (rv35)
s_RV36	DataContext	Rv36 (rv36)
s_RV37	SourceFunction	Rv37 (rv37)
s_RV38	SourceOperationalContext	Rv38 (rv38)
s_RV39	MSSPCustomerName	Rv39 (rv39)
s_RV40 a s_RV43	Valor reservado de 40 a 43 **NOTA:** Reservado para uso de Novell.	Rv40 a Rv43 (rv40 a rv43)
s_RV44	DestinationThreatLevel	Rv44 (rv44)
s_RV45	DestinationUserContext	Rv45 (rv45)
s_RV46	VirusStatus	Rv46 (rv46)
s_RV47	DestinationFunction	Rv47 (rv47)
s_RV48	DestinationOperationalContext	Rv48 (rv48)
s_RV49	ReservedVar49 **NOTA:** Reservado para uso de Novell.	Rv49 (rv49)
s_RV50	eSecTaxonomyLevel1	Rv50 (rv50)
s_RV51	eSecTaxonomyLevel2	Rv51 (rv51)
s_RV52	eSecTaxonomyLevel3	Rv52 (rv52)
s_RV53	eSecTaxonomyLevel4	Rv53 (rv53)
s_RV54 a s_RV100	Valor reservado de 54 a 100 **NOTA:** Reservado para uso de Novell.	Rv54 a Rv100 (rv54 a rv100)

Formato automático

Las variables reservadas s_DP, c_SP y s_P están definidas en minúsculas antes de enviar el mensaje de evento. Las variables reservadas s_ST y s_PN están definidas en mayúscula antes de enviar el mensaje de evento. La s_ET de la variable de fecha y hora de evento se define si se deja en blanco con el formato de hora estándar, de la siguiente manera:

```
s_Year-s_Month-s_Day~sHour:s_Min:s_Sec~s_AMPM24~s_TZ
```

Para anular esta función, defina la variable s_ET con otra información. Como mínimo, tanto s_Hour como s_Month deben definirse para crear la ET. Todos los campos vacíos aparecen en el campo ET como NULOS.

Variables reservadas de fecha y hora

La variable s_ET de la meta-etiqueta ET se rellena automáticamente si s_ET se deja en blanco y si s_Hour y s_Month no están vacíos. Las variables reservadas de fecha y hora deben definirse con valores. Todo campo vacío aparecerá como NULL (NULO). El formato del campo s_Day está compuesto por valores de dos dígitos 01-09. El escritor de guiones puede convertir el valor de mes en un número de dos dígitos a través del comando TRANSLATE y el archivo months.csv. Las etiquetas reservadas de fecha y hora son las siguientes:

- s_Year
- s_Month
- s_Day
- s_Hour
- s_Min
- s_Sec
- s_TZ
- s_AMPM24

Variables reservadas de control de eventos

Dos variables, s_SendEITag y s_SendETTag, se utilizan para determinar si el comando EVENT incluye los campos EI y ET, respectivamente, en un mensaje de alerta. Para desactivar el envío de cualquiera de los dos campos, las variables deben definirse como OFF (Inactivo).

Formato

```
EVENT ()
```

Por ejemplo:

```
COPY(s_Res:"Recurso")
SET(i_Severity = 3)
COPY(s_BM:"Alerta")
EVENT()
```

FILEA

El comando FILEA agrega el contenido de una cadena al final de un archivo llano en el disco. Al utilizar este comando:

- Especifique el nombre de archivo usando una cadena.
- Para Windows, el nombre de archivo hace referencia al archivo según se ha especificado, si el nombre de archivo comienza con una letra de la unidad, dos puntos y barra diagonal inversa (como c:\).
- Debe especificar toda la vía del archivo.
- Si el archivo no existe, se crea.
- Si no se puede crear el archivo, el comando FILEA no hace nada.
- El archivo se cierra una vez agregados los datos.

Si escribe este comando como parte de un guión que ejecutará un recopilador, asegúrese de utilizar la sintaxis de ruta adecuada, incluidas las barras diagonales (/). Recuerde incluir los caracteres de escape barra diagonal y barra diagonal inversa al especificar la vía. El cero final al final de la cadena no se escribe en el archivo.

Formato

```
FILEA("filename", data)
```

Tipos de datos

Argumento	Tipo	Descripción
filename	string (ENTRADA)	El nombre del archivo al que se aplicarán los datos.
data	string (ENTRADA)	La cadena de datos que se agregará al archivo.

Por ejemplo:

En el siguiente ejemplo, se crea el archivo \temp\mux_data y se agrega el contenido de s_variable a este archivo:

```
FILEA("c:/\temp/\mux_data", s_variable)
FILEA("mux_data", "literal")
FILEA("mux_data", s_variable)
```

En el siguiente ejemplo, se agrega una cadena al final de un archivo de registro de auditoría.

```
COPY(audit_str: "Se han enviado 20 alertas de gravedad 5.")
FILEA("h:/\temp/\audit.log", audit_str)
```

FILEL

El comando FILEL obtiene la longitud (en bytes) de un archivo llano y ubica el valor en una variable numérica. Al utilizar este comando:

- Especifique el nombre de archivo usando una cadena.
- Para Windows, el nombre de archivo hace referencia al archivo según se ha especificado, si el nombre de archivo comienza con una letra de la unidad, dos puntos y barra diagonal inversa (como c:\).
- Si el archivo no existe, el comando FILEL no hace nada y el contenido de numvar no se modifica.
- El archivo se cierra una vez leídos los datos.

Si escribe este comando como parte de un guión que ejecutará un recopilador, asegúrese de utilizar la sintaxis de ruta adecuada, incluidas las barras diagonales (/). Recuerde incluir los caracteres de escape barra diagonal y barra diagonal inversa al especificar la vía.

Formato

```
FILEL("filename", i_length)
```

Tipos de datos

Argumento	Tipo	Descripción
filename	string (ENTRADA)	El nombre del archivo, cuya longitud se determinará.
i_length	numvar (SALIDA)	La longitud del archivo, en bytes.

Por ejemplo:

```
FILEL("h:/\tmp/\onfotron.log", i_length)
```

Devuelve la longitud del archivo infotron.log, en bytes, por ejemplo:

```
i_length = 2390
```

FILER

El comando FILER copia el contenido de un archivo llano del disco en una variable de cadena. Al utilizar este comando:

- Especifique el nombre de archivo usando una cadena.
- Para Windows, el nombre de archivo hace referencia al archivo según se ha especificado, si el nombre de archivo comienza con una letra de la unidad, dos puntos y barra diagonal inversa (como c:\).
- Si el archivo no existe, el comando FILER no hace nada y el contenido de svar no se modifica.
- El archivo se cierra una vez leídos los datos.
- De manera opcional, introduzca el número máximo de bytes que se leerá. No puede utilizar el parámetro max_bytes, a menos que se combine con el parámetro i_offset.

Si escribe este comando como parte de un guión que ejecutará un recopilador, asegúrese de utilizar la sintaxis de ruta adecuada, incluidas las barras diagonales (/). Recuerde incluir los caracteres de escape barra diagonal y barra diagonal inversa al especificar la vía.

Formato

```
FILER("filename", dest, [i_offset [, i_max_bytes]])
```

NOTA: No puede utilizar el parámetro max_bytes, a menos que se combine con el parámetro i_offset.

Tipos de datos

Argumento	Tipo	Descripción
filename	string (ENTRADA)	El nombre del archivo desde el que se leerá la cadena de datos.
data	svar (SALIDA)	Los datos leídos del archivo se ubican en esta variable de cadena.
i_offset	entero (ENTRADA) [OPCIONAL]	Especifica un número de desplazamiento de caracteres desde el que se comenzará a leer.
max_bytes	entero (ENTRADA) [OPCIONAL]	De manera opcional, especifique el número máximo de bytes que se leerá. **NOTA:** Al utilizar este argumento, debe especificar el argumento i_offset.

Por ejemplo:

```
CLEAR(data)
FILER("filename", data, 0, 20)
if(data = "")
ALERT(s_res_var, "El archivo de datos no existe o está
   vacío.", 0)
ENDIF()
```

FILEW

El comando FILEW escribe el contenido de una cadena en un archivo sin formato del disco. Al utilizar este comando:

- Se sobrescribe el contenido anterior del archivo.
- Especifique el nombre de archivo usando una cadena.
- Para Windows, el nombre de archivo hace referencia al archivo según se ha especificado, si el nombre de archivo comienza con una letra de la unidad, dos puntos y barra diagonal inversa (como c:\).

- Si el archivo no existe, se crea.
- Si no se puede crear el archivo, el comando FILEW no hace nada.
- El archivo se cierra una vez escritos los datos.

Si escribe este comando como parte de un guión que ejecutará un recopilador, asegúrese de utilizar la sintaxis de ruta adecuada, incluidas las barras diagonales (/). Recuerde incluir los caracteres de escape barra diagonal y barra diagonal inversa al especificar la vía.

Formato

```
FILEW("filename", data)
```

Tipos de datos

Argumento	Tipo	Descripción
filename	string (ENTRADA)	El nombre del archivo en el que se escribirá la cadena de datos.
data	svar (SALIDA)	Los datos que se escribirán en el archivo.

Por ejemplo:

```
FILEW("filename", data)
FILEW("h:/\tmp/\infotron.stat", "EJEC CORRECTA")
```

FOR

El comando FOR permite ejecutar en bucle el flujo de control. Al utilizar este comando:

- Siempre se ejecuta la instrucción de inicialización.
- Si el resultado de la instrucción de comparación FOR() es verdadero, se ejecutan los comandos de análisis posteriores al comando FOR(), hasta el siguiente comando ENDFOR(). Se ejecuta la instrucción de incremento y el control del flujo se devuelve a la instrucción de comparación.
- Si el resultado de la instrucción de comparación FOR() es falso, no se ejecuta ningún comando de análisis entre los comandos FOR() y ENDFOR(). No se ejecuta la declaración de incremento.
- Si bien se permiten todos los tipos de datos a cada lado de la instrucción de comparación for(), los valores numéricos sólo pueden compararse con los valores numéricos, y las cadenas sólo pueden compararse con las cadenas.
- El operador de la instrucción de comparación FOR() puede ser <, =, >, <=, >=, <>, &, + o ^.

No puede realizar una comparación directamente con un número negativo. Para ello, lleve a cabo uno de estos métodos:

- Utilice la función de análisis COMPARE.
- Realice una comparación indirectamente, según se muestra a continuación:

```
SET(i_compare_val=-10)
FOR(ivar=0, ivar>i_compare_val, ivar=ivar-1)
ALERT("Todavía en el bucle")
ENDFOR()
```

Formato

```
FOR(initialization, compare, increment)
```

Tipos de datos

Argumento	Tipo	Descripción
initialization	SET() parameter	Todo parámetro válido que puede transferirse al comando SET(). Verifique la definición del comando SET().
conditional	IF() conditional	Todo parámetro válido que puede transferirse al comando IF(). Verifique la definición del comando IF().
increment	SET() parameter	Todo parámetro válido que puede transferirse al comando SET(). Verifique la definición del comando SET().

Por ejemplo:

```
FOR(i=0, i<3, i=i+1)
```

GETCONFIG

Recupera la configuración actual de una propiedad del sistema. Este comando se utiliza para recuperar las propiedades del sistema configuradas con el comando SETCONFIG. Estos comandos se utilizan para definir variables y recuperar los valores actuales de las propiedades del sistema que pueden modificarse con frecuencia, por ejemplo, un archivo de registro, cuyo nombre se cambia todos los días con la fecha actual.

Las propiedades del sistema disponibles son:

Propiedad del sistema	Ejemplos
▪ System.OS.Family	Solaris y Windows
▪ System.OS.Name	Windows 2000
▪ System.OS.Version.Major	5
▪ System.OS.Version.Minor	0
▪ System.Net.Hostname	ESECServer
▪ System.Net.IP_List	Lista de direcciones IP para este host separadas por un punto y coma, por ejemplo, “172.163.3.45;172.45.2.1”

Consulte el comando SETCONFIG.

Requiere dos parámetros.

- El primer parámetro requerido define la opción de configuración (FileConnector.InputFile) o (FileConnector.OutputFile).
- El segundo parámetro requerido define el valor de configuración que se recuperará.

Formato

```
GETCONFIG(Config Option, Value)
```

Tipos de datos

Argumento	Tipo	Descripción
Config Option	string (ENTRADA)	Nombre de la variable de configuración que se recuperará. Archivo de entrada = “FileConnector.InputFile” Archivo de salida = “FileConnector.OutputFile”
Valor	string (ENTRADA)	Valor de configuración que se recuperará.

Por ejemplo:

```
GETCONFIG(“FileConnector.InputFile”, s_inputfilename)
GETCONFIG(“FileConnector.OutputFile”, s_outputfilename)
```

Contenido de las variables de salida actuales

```
“C:/\nombrearchivo.txt”
```

GETENV

El comando GETENV recupera el valor de una variable de entorno.

Formato

```
GETENV(Environment Key, Variable to store value)
```

Tipo de datos

Argumento	Tipo	Descripción
Environment Key	string (ENTRADA)	Nombre de la variable de entorno.
Variable to store value	Variable de cadena (ENTRADA)	Destino del lugar en el que se guardará la variable de entorno.

Por ejemplo:

```
GETENV("ESEC_HOME", s_EsecHome)
```

HEXTONUM

El comando HEXTONUM convierte una cadena hexadecimal con hasta 4 bytes de datos hexadecimales en un número decimal y ubica este número en una variable de valores flotantes o entero. Más de 4 bytes da como resultado datos no válidos.

Formato

```
HEXTONUM(bytes_data, i_val [,[-]i_4] [, ioffset])
```

Tipos de datos

Argumento	Tipo	Descripción
bytes_data	string (ENTRADA)	Cadena de 1 a 4 bytes. (por ejemplo: "\FF\", "\FF FF\", "\3C 4A F2\", "\43 76 F3 FF\" o "PRUEBA"). El número hexadecimal representado por estos bytes se convertirá en un valor entero, i_val.
i_val	numvar (SALIDA)	El decimal equivalente al número hexadecimal se ubica en esta variable, ivar o fvar.
i_len	numérico (ENTRADA) [OPCIONAL]	Número de bytes hexadecimales que se convertirá en un entero (debe tener un rango de valor absoluto de 1 a 4). Si no define este parámetro, el valor predeterminado es el número de bytes en la cadena de entrada, bytes_data, hasta 4 bytes. Si i_len es positiva, los bytes se interpretan de izquierda a derecha (del byte más importante al byte menos importante). Si i_num_bytes es negativa, los bytes se interpretan de derecha a izquierda (del byte menos importante al byte más importante).
ioffset	numérico (ENTRADA) [OPCIONAL]	Número de desplazamiento de bytes que se omitirá en bytes_data.

Por ejemplo:

En el siguiente ejemplo, los datos de una cadena hexadecimal "\5A32\" se convierten a un valor entero; se interpretan del byte más importante al byte menos importante y, a continuación, del byte menos importante al byte más importante.

```
COPY(data:"\5A 32\")
HEXTONUM(data, ivar1)
HEXTONUM(data, ivar2, -2)
```

NOTA: Para la sustitución de hexadecimales, \0000\ termina una cadena; por lo tanto, "xxxx\0000\yyyy" se convierte en "xxxx".

Contenido de las variables de salida actuales:

```
ivar1 = 23090
ivar2 = 12890
```

IF

El comando IF compara dos valores.

- Si el resultado de la instrucción IF() es verdadero, se ejecutan los comandos de análisis posteriores al comando IF(), hasta el siguiente comando ELSE() o ENDIF().
- Si el resultado de la instrucción IF() es falso, se ejecutan los comandos de análisis posteriores al comando ELSE(), hasta ENDIF().
- Si no se utiliza ningún comando ELSE, cuando el resultado de la instrucción IF() es falso, no se ejecuta ningún comando de análisis entre IF() y ENDIF().
- Si bien se permiten todos los tipos de datos a cada lado de la instrucción IF(), los valores numéricos sólo pueden compararse con los valores numéricos, y las cadenas sólo pueden compararse con las cadenas.
- El operador de comparación de IF() puede ser <, =, >, <=, >=, <>, &, + o ^. No utilice el operador lógico NOT (^) junto con una variable de cadena. Si lo hace, se genera un error de sintaxis.

No puede realizar una comparación directamente con un número negativo. Para ello, lleve a cabo uno de estos métodos:

- Utilice la función de análisis COMPARE.
- Realice una comparación indirectamente, según se muestra a continuación:

```
SET(i_compare_val=-10)
IF(ivar > i_compare_val)
ALERT("ivar es superior a -10")
ENDIF()
```

Formato

```
IF(<expr>)
Donde:
expr ::= var
        | (<expr>)
        | ^ <expr>
```

Donde <expr> debe dar como resultado enteros o valores flotantes.

```
        | <expr> <|=|>|<=|>=|<>|&|+ <expr>
```

Donde las dos <expr> deben dar como resultado el mismo tipo.

Tipos de datos

Argumento	Tipo	Descripción
data1	variable (ENTRADA)	Los datos que se compararán con data2. Si data2 no se utiliza, se convierte en un valor lógico (0 = falso, cualquier otro = verdadero).
logical operator	< = > <= >= <> & + ^	Inferior a Igual a Superior a Inferior o igual a Superior o igual a No es igual a Operador lógico AND Operador lógico OR Operador lógico NOT
data2	todos (ENTRADA) [OPCIONAL]	Los datos que se compararán con data1. Debe ser el mismo tipo que data1.
…	igual que en el caso anterior	Utilice hasta 200 parámetros individuales para crear expresiones lógicas complejas.

Por ejemplo:

```
IF(s = "prueba" & i_count < 5)
script(test)
ELSE()
IF((i <= i_num) + (i_count <> 10) & (i_page))page("111")
ENDIF()
ENDIF()
```

INC

El comando INC incrementa una variable numérica en 1. Al utilizar este comando, debe especificar una variable de enteros o una variable de valores flotantes.

Formato

```
INC(i_counter)
```

Tipos de datos

Argumento	Tipo	Descripción
i_counter	numvar (ENTRADA/ SALIDA)	La variable numérica que se incrementará en 1.

Por ejemplo:

```
SET(icounter = 0)
INC(icounter)
INC(icounter)
```

Resultado:

```
icounter = 2
```

INDICATOR

El comando INDICATOR envía mensajes del indicador a Sentinel. Los mensajes contienen texto que se mostrará en el indicador especificado en Sentinel.

Formato

```
INDICATOR(name, value)
```

NOTA: En versiones anteriores a la versión 4.0, el comando INDICATOR tenía argumentos adicionales que ya no se utilizan. Para lograr la compatibilidad con otros recopiladores, estos argumentos llevan la etiqueta "Not Used" (no utilizado) en la ventana del editor de comandos del asistente.

Tipos de datos

Argumento	Tipo	Descripción
name	string (ENTRADA)	Nombre del indicador
value	string (ENTRADA)	El texto del indicador que se mostrará en la consola de Sentinel. Por ejemplo: `IMPRESORA ENCENDIDA`

Por ejemplo:

```
INDICATOR("memoria", "5 MB")
INDICATOR(name, value)
```

NOTA: El nombre del indicador en el comando de análisis debe coincidir con el nombre del indicador de Sentinel; si no coinciden, el indicador no se actualizará en la consola de Sentinel.

INFO_CLEARTAGS

Esta función quitará (o borrará, en el caso de las cadenas) todas las variables que formen parte del conjunto de bloque de información al que hace referencia el identificador. Utilice INFO_CONSTANTTAGS para que esto no ocurra con un subconjunto de tales etiquetas.

Formato

```
INFO_CLEARTAGS(<IN handle>)
```

Tipos de datos

Argumento	Tipo	Descripción
IN handle	string (ENTRADA)	tipo de bloque de información

INFO_CLOSE

Este comando se utiliza para cerrar una sesión del bloque de información. Al llamarlo, primero envía todos los bloques de información que no se han enviado, tal como lo haría el comando INFO_SEND. A continuación, envía un mensaje de cierre de sesión de bloque de información al configurar el atributo EOD (End Of Data) de los elementos infos en "verdadero". Después de enviar el mensaje de cierre, el número de segmento ("segnum") se incrementa en uno.

Formato

```
INFO_CLOSE(<IN handle>)
```

Tipos de datos

Argumento	Tipo	Descripción
IN handle	string (ENTRADA)	tipo de bloque de información

INFO_CONSTANTTAGS

Utilice este comando para identificar las etiquetas que no se borrarán cuando se solicite INFO_CLEARTAGS. Pase ninguno o más nombres de etiquetas para crear el conjunto de etiquetas constantes. Varias llamadas a esta función restablecerán la lista de etiquetas constantes.

Formato

```
INFO_CONSTANTTAGS(<IN handle>, [<IN tag name>, …])
```

Tipos de datos

Argumento	Tipo	Descripción
IN handle	string (ENTRADA)	tipo de bloque de información
IN tag name	string (ENTRADA)	nombre para remitirse a la referencia IN

INFO_CREATE

Este comando creará un nuevo conjunto de bloque de información. Debe pasar una referencia (que usará en todos los demás comandos para afectar a este conjunto de bloque de información). También debe pasar un tipo. Se trata de una cadena de su elección, pero debe formalizarse (consulte INFO_SEND).

Si solicita INFO_CREATE en una referencia ya existente, borrará el contenido en esa referencia como si hubiera comenzado una nueva referencia. Deberá solicitar INFO_SETTAG y INFO_CONSTANTTAGS de nuevo.

Formato

```
INFO_CREATE(<OUT handle>,<IN type>)
```

Tipos de datos

Argumento	Tipo	Descripción
OUT handle	string (SALIDA)	nombre para hacer referencia al tipo IN
IN type	string (ENTRADA)	tipo de bloque de información

INFO_DUMP

Este comando conservará el estado actual del conjunto de bloque de información en una variable de cadena. Este comando se incluyó para facilitar las tareas de comprobación, pero también se puede utilizar para reproducir conjuntos de bloque de información o guardarlos en un archivo de texto, o cualquier otro tipo de archivo. No tiene el efecto secundario que tiene INFO_SEND, ya que no borra el estado actual.

Formato

```
INFO_DUMP(<IN handle>, <OUT string-variable>)
```

Tipos de datos

Argumento	Tipo	Descripción
IN handle	string (ENTRADA)	tipo de bloque de información
OUT string-variable	string (SALIDA)	variable de cadena para remitirse a la referencia IN

INFO_PUSH

Este comando etiquetará los valores actuales de todos los nombres de etiquetas (a través de sus variables relacionadas) y los trasladará al final de una lista de bloques de información a los que hace referencia un identificador. Los bloques se seguirán acumulando en el conjunto hasta que se vacíe al ejecutar INFO_CREATE, INFO_SEND o INFO_CLOSE. Para INFO_CREATE, no se toma ninguna medida. Para INFO_SEND, los bloques de información se envían a Collectormanager. Para INFO_CLOSE, los bloques de información se envían a Collectormanager y se envía un mensaje de cierre de bloque de información (EndOfData o EOD).

Formato

```
INFO_PUSH(<IN handle>)
```

Tipos de datos

Argumento	Tipo	Descripción
IN handle	string (ENTRADA)	tipo de bloque de información

INFO_SEND

Este comando toma el conjunto actual de bloques de información y los envía a través de un canal de comunicación especificado por el tipo que se utilizó durante INFO_CREATE, agregado a la palabra “infoblock.”, incluido el punto. Por lo tanto, si el tipo era “vulnerability”, el nombre del canal por el que se enviaría el mensaje sería “infoblock.vulnerability”.

Además, este comando borrará el conjunto actual de bloques de información e incrementará el número de segmento (“segnum”) en uno.

Formato

```
INFO_SEND(<IN handle>)
```

Tipos de datos

Argumento	Tipo	Descripción
IN handle	string (ENTRADA)	tipo de bloque de información

INFO_SETTAG

Este comando unirá una variable de cadena con el nombre de un atributo. Cuando se llame a INFO_PUSH (consulte INFO_PUSH), todas las variables que se unieron a este comando se definirán como atributos en una entrada de bloque.

Formato

```
INFO_SETTAG(<IN handle, IN tag name, IN variable)
```

Tipos de datos

Argumento	Tipo	Descripción
IN handle	string (ENTRADA)	tipo de bloque de información
IN tag name	string (ENTRADA)	tipo de nombre de etiqueta
IN variable	string (ENTRADA)	tipo de variable

Etiquetas de vulnerabilidad de bloques de información

A continuación, se incluye una lista de etiquetas válidas de vulnerabilidad de bloques de información para el comando INFO_SETTAG. Las etiquetas que se identifican como requeridas deben definirse para que el bloque de información se guarde como una vulnerabilidad. Aunque el bloque de información no se guarde como una vulnerabilidad, las etiquetas que se identifican como constantes se extraerán del bloque de información, de todos modos. Si se define una etiqueta que no se incluye en esta lista, el sistema de apoyo de vulnerabilidades omitirá la etiqueta.

Nombre de etiqueta	Explicación	Tipo	Constante	Requerida
ScannerInstance	El nombre que el usuario le da a esta instancia del escáner. Suele definirse en los parámetros del Recopilador.	Cadena	X	
ProductName	Nombre del escáner.	Cadena	X	
ProductVersion	Versión del escáner.	Cadena	X	
ScannerType	El tipo de escáner.	Cadena	X	
Vendor	El nombre del fabricante del escáner.	Cadena	X	
ScanType	PARTIAL o FULL	Cadena	X	
ScanStartDate	La hora a la que comenzó la exploración.	Cadena		
ScanEndDate	La fecha y hora a la que finalizó la exploración.	Cadena		
IP	La dirección IP del recurso.	Cadena		X
HostName	El nombre de host del recurso.	Cadena		
Location	La ubicación del recurso.	Cadena		
Department	El departamento del recurso.	Cadena		
BusinessSystem	El sistema empresarial del recurso.	Cadena		
OperationalEnvironment	El entorno de operación del recurso.	Cadena		
Regulation	La regulación del recurso.	Cadena		
RegulationRating	La calificación de regulación del recurso.	Cadena		
Criticality	La importancia del recurso [1 – 25].	Núme-ro		
VulnModule	El módulo utilizado para detectar la vulnerabilidad.	Cadena		
PortNumber	El número de puerto de la vulnerabilidad.	Número		
PortName	El nombre del puerto de la vulnerabilidad.	Cadena		
NetworkProtocol	El protocolo de red de la vulnerabilidad.	Número		
ApplicationProtocol	El protocolo de aplicación de la vulnerabilidad.	Cadena		
AssignedVulnSeverity	La gravedad asignada de la vulnerabilidad.	Número		

Nombre de etiqueta	Explicación	Tipo	Constante	Requerida
ComputedVulnSeverity	La gravedad calculada de la vulnerabilidad.	Número		
VulnDescription	La descripción de la vulnerabilidad.	Cadena		
VulnSolution	La solución de la vulnerabilidad.	Cadena		
VulnSummary	La solución de la vulnerabilidad.	Cadena		
VulnCrossRefs	Una lista de códigos para la vulnerabilidad.	Cadena		
DetectedOs	El sistema operativo detectado al encontrar la vulnerabilidad.	Cadena		
DetectedOsVersion	La versión del sistema operativo detectado al encontrar la vulnerabilidad.	Cadena		
ScannedApp	La aplicación detectada al encontrar la vulnerabilidad.	Cadena		
ScannedAppVersion	La versión de la aplicación detectada al encontrar la vulnerabilidad.	Cadena		
VulnUserName	El nombre de usuario de la vulnerabilidad.	Cadena		
VulnUserDomain	El dominio del usuario de la vulnerabilidad.	Cadena		
VulnTaxonomy	La taxonomía de la vulnerabilidad.	Cadena		
ScannerClassification	La clasificación de vulnerabilidad determinada por el escáner.	Cadena		
ExtendedInformation	La información ampliada que se almacenará junto con esta vulnerabilidad.	Cadena		
VulnName	El nombre de la vulnerabilidad determinado por el escáner.	Cadena		

Ejemplo del comando INFO_*

Sentinel separa las exploraciones de vulnerabilidad en pequeñas porciones (sesiones de bloques de información) que pueden procesarse más fácilmente. Una sesión de bloque de información contiene varios conjuntos de bloques de información, cada uno consta de un número cada vez mayor de segmento (“segnum”) seguido de un mensaje de cierre de sesión de bloque de información. Se hace referencia a una sesión de bloque de información por su “id” único global. Cada vez que se llama a INFO_SEND, se envía un conjunto de bloques de información con los valores “introducidos” y el número de segmento actual (“segnum”). A continuación, el número de segmento se incrementa en uno. El comando INFO_SEND se llama para cada lote de datos, después del cual se solicita el comando INFO_CLOSE para cerrar la sesión de bloque de información. El mensaje de cierre de bloque de información consta de un conjunto de bloques de información con el atributo EOD definido en “verdadero”.

Por ejemplo:

```
INFO_CREATE(h_vuln,"vulnerabilidad")
INFO_SETTAG(h_vuln,"ALFA", s_alpha)
INFO_SETTAG(h_vuln,"BETA", i_beta)
INFO_SETTAG(h_vuln,"GAMMA", s_gamma)
INFO_SETTAG(h_vuln,"DELTA", i_delta)
INFO_SETTAG(h_vuln,"^1E*P$S I(L)O.N--", f_epsilon)
INFO_CONSTANTTAGS(h_vuln,"GAMMA","DELTA","^1E*P$S
   I(L)O.N--")
SET(i_beta=12345)
SET(i_delta=123456789)
SET(f_epsilon=1.234)
COPY(s_alpha:"a corresponde a azul")
COPY(s_gamma:"c corresponde a café")
INFO_PUSH(h_vuln)
INFO_CLEARTAGS(h_vuln)
INFO_PUSH(h_vuln)
INFO_DUMP(h_vuln, s_simulate)
INFO_SEND(h_vuln)
SET(i_beta=6789)
SET(i_delta=987654321)
SET(f_epsilon=3.1415926)
COPY(s_alpha:"a corresponde a agua")
COPY(s_gamma:"c corresponde a ciruela")
INFO_PUSH(h_vuln)
INFO_SEND(h_vuln)
INFO_CLOSE(h_vuln)
```

Resultados:

```
<?xml version="1.0" encoding="UTF-8"?>
<infos id="B008961E00CB1026B8F000065BBD13AB"
   type="vulnerabilidad" segnum="0" version="4.2.0.0"
   EOD="falso">
<info ALPHA="a corresponde a azul" BETA="12345"
   DELTA="123456789" GAMMA="c corresponde a café"
   _1EPSILON="1.234"/>
<info ALPHA="" BETA="0" DELTA="123456789" GAMMA="c
   corresponde a café" _1EPSILON="1.234"/>
</infos>
<?xml version="1.0" encoding="UTF-8"?>
<infos id="B008961E00CB1026B8F000065BBD13AB"
   type="vulnerabilidad" segnum="1" version="4.2.0.0"
   EOD="falso">
<info ALPHA="a corresponde a agua" BETA="6789"
   DELTA="987654321" GAMMA="c corresponde a ciruela"
   _1EPSILON="3.1415926"/>
</infos>
<?xml version="1.0" encoding="UTF-8"?>
<infos id="B008961E00CB1026B8F000065BBD13AB"
   type="vulnerabilidad" segnum="2" version="4.2.0.0"
   EOD="verdadero">
</infos>
```

IPTONUM

El comando IPTONUM convierte una representación de cadena de direcciones IPv4 en un número entero y ubica este número en una variable de enteros. Esta función sólo admite direcciones IPv4. Una dirección IPv4 que no encaje en el rango válido, resulta en datos no válidos.

Formato

```
IPTONUM(ip_address, i_integer, i_valid)
```

Tipos de datos

Argumento	Tipo	Descripción
ip_address	svar (ENTRADA)	Dirección IPv4 de cadena.
i_integer	numérico (SALIDA)	La dirección IPv4 de cadena se convierte en un valor entero. El valor entero se ubica en esta variable.
i_invalid	ivar (SALIDA) [OPTIONAL}	El valor 0 significa que la dirección IP no es válida. El valor 1 significa que la dirección IP es válida.

Por ejemplo:

En el siguiente ejemplo, la dirección IPv4 “10.10.10.255” se convierte en un número entero. i_valid se define en 1, lo que significa que el resultado es válido.

```
IPTONUM("10.10.10.255", i_y, i_valid)
```

Contenido de la variable de salida actual:

```
i_y = 168430335
i_valid = 1
```

En el siguiente ejemplo, la dirección IPv4 “10.10.10.258” no válida se convierte en un número entero 0. i_valid se define en 0, lo que significa que el resultado no es válido.

```
IPTONUM("10.10.10.258", i_y, i_valid)
```

Contenido de la variable de salida actual:

```
i_y = 0
i_valid = 0
```

El comando NUMTOIP convierte un número en una dirección IP. Para obtener más información, consulte NUMTOIP.

LENGTH o LENGTH-OPTION2

El comando LENGTH define una variable numérica a partir de la longitud en bytes de una variable de cadena (sin tener en cuenta el cero final).

NOTA: En el Editor visual del Generador de recopiladores, LENGTH y LENGTH-OPTION2 aparecen como comandos independientes. Se trata de los mismos comandos. Se proporcionan como descripciones de diferentes variaciones del mismo comando. Si va a utilizar LENGTH-OPTION2 en el editor de texto, introducirá LENGTH.

Formato

```
LENGTH(i_length, s_variable)
```

Tipos de datos

Argumento	Tipo	Descripción
s_variable	string (ENTRADA)	La cadena (por lo general, una variable de cadena) en la que se calcula la longitud.
i_length	numvar (SALIDA)	La longitud de la variable de cadena, s_variable, se ubica en esta variable numérica.

Por ejemplo:

```
LENGTH(i_length, source)
LENGTH(i_num_bytes, "No tiene sentido hacer esto, ya que
   conocemos la cadena, cuya longitud estamos comprobando")
```

Resultados:

```
i_num_bytes = 80
```

LOOKUP

El comando LOOKUP compara datos encontrados en el buffer de recepción o en una cadena con cadenas clave encontradas en un archivo de búsqueda clave especificado.

Si se encuentra un registro que coincide con los datos byte por byte, se procesan los comandos de análisis en el registro del archivo de búsqueda clave.

Si se especifica una cadena como el primer parámetro en el comando LOOKUP, el comando utiliza esa cadena al buscar el archivo de búsqueda clave.

Existen cinco argumentos o parámetros con este comando.

- compare: si se especifica un valor numérico como este parámetro, ese número de bytes (el valor numérico) de datos del buffer de recepción, a partir de la posición del puntero Rx buffer, se utiliza como cadena al realizar la comparación con las cadenas clave del archivo de búsqueda clave.
- lookup name: este parámetro especifica el nombre del archivo de búsqueda clave en relación con el directorio WORKBENCH_HOME.
- imatch: una variable de enteros opcional que puede especificarse, la cual devuelve el estado del comando LOOKUP. (0 = no se encontró ninguna coincidencia, 1 = se encontraron coincidencias).
- parameter file: un parámetro opcional que corresponde al nombre de un archivo de parámetros para utilizar otro archivo diferente al archivo de parámetros predeterminado. El nombre del archivo de parámetros predeterminado es <recopilador>.par. Este nombre de archivo no debe incluir el sufijo.par.
- column name: un parámetro opcional es la columna con el archivo de parámetros que se utilizará para los valores de búsqueda. El nombre de columna predeterminado es el nombre de la plantilla. Si especifica este parámetro, también debe utilizar un nombre de archivo de parámetros.

Formato

```
LOOKUP(compare, lookup filename [, imatch] [, [parameter
   filename] [, column name]])
```

Tipos de datos

Argumento	Tipo	Descripción
compare	string (ENTRADA) o numérico (ENTRADA)	Los datos que se utilizarán para realizar la comparación con los campos en el archivo de búsqueda clave. Se trata de una comparación byte por byte. El número de bytes del buffer de recepción, con la posición actual del puntero Rx buffer, que se utilizará para realizar la comparación con los campos del archivo de búsqueda clave. Se trata de una comparación byte por byte. **NOTA:** Sólo funcionará si rxbuff se utilizó para definir el buffer de recepción.
lookup filename	string (ENTRADA)	El nombre del archivo de búsqueda clave.
imatch	numvar (SALIDA) [OPCIONAL]	Se encontró una coincidencia. 0 = No 1 = Sí
parameter filename	string (ENTRADA)	El nombre del archivo de parámetros. Valor predeterminado: Collector.par
column name	string (ENTRADA)	La columna en el archivo de parámetros que se utilizará. Valor predeterminado: `Nombre del recopilador`

Por ejemplo:

```
LOOKUP(data, filename, imatch)
```

En el siguiente ejemplo, el nombre de archivo key_01 se determina a partir del nombre indicado en el archivo de parámetros, no el nombre del archivo de búsqueda clave.

```
LOOKUP(s_variable, {key_01})
LOOKUP(s_variable, {key_01}, imatch, "Enviar un alerta",
   "Elementos geo")
```

Si alguna definición de parámetros se encuentra en el archivo de búsqueda, búsquela en la columna GeoElements del archivo de parámetros Send One Alert (Enviar una alerta).

NEGSEARCH

El comando NEGSEARCH realiza una búsqueda hacia atrás de una cadena en el buffer de recepción. Existen dos parámetros con este comando.

- search: la búsqueda comienza en la posición actual del puntero Rx buffer y retrocede hasta que encuentra la cadena o hasta que llega al comienzo del buffer de recepción. Si la búsqueda encuentra la cadena, el puntero Rx buffer se actualiza para indicar el primer byte de la cadena de búsqueda. Si la búsqueda no encuentra la cadena, el puntero Rx buffer no se modifica.
- ifound: un parámetro opcional, se trata de una variable de enteros que se define en 1 si la búsqueda encuentra la cadena, y se define en 0 si la búsqueda no encuentra la cadena.

Formato

```
NEGSEARCH(search[, ifound])
```

Tipos de datos

Argumento	Tipo	Descripción
search	string (ENTRADA)	La cadena buscada en el buffer de recepción, a partir de la posición actual del puntero Rx buffer y mediante una búsqueda hacia atrás.
ifound	numvar (SALIDA) (OPCIONAL)	Devuelve si la cadena de búsqueda se encontró o no. 0 = no se encontró 1 = se encontró

Por ejemplo:

```
NEGSEARCH("ALARMA SECUNDARIA")
NEGSEARCH(search_string)
```

En el siguiente ejemplo, se busca un retorno de carro y un avance de línea:

```
NEGSEARCH("\0d0a\")
NEGSEARCH(data, ifound)
```

Otro ejemplo es el siguiente:

La letra subrayada representa la posición actual del puntero Rx buffer en el ejemplo.

NOTA: Para la sustitución de hexadecimales, \0000\ termina una cadena; por lo tanto, “xxxx\0000\yyyy” se convierte en “xxxx”.

```
Rx Buffer = "Radioalarma secundaria A"
NEGSEARCH("Ala")
```

Resultado:

```
Rx Buffer = "Radioalarma secundaria A"
```

NUMTOHEX

El comando NUMTOHEX convierte un número numérico en datos hexadecimales y ubica los bytes hexadecimales (hasta 4 bytes) en una cadena.

Formato

```
NUMTOHEX(i_decimal, hex_data)
```

Tipos de datos

Argumento	Tipo	Descripción
i_decimal	numérico (ENTRADA)	Valor entero que se convertirá en datos hexadecimales.
hex_data	svar (SALIDA)	Cadena de 1 a 4 que son los bytes hexadecimales determinados por el valor numérico, i_decimal.

Por ejemplo:

En el siguiente ejemplo, el número decimal 16777215 se convierte en datos hexadecimales.

```
SET(i_decimal = 16777215)
NUMTOHEX(i_decimal, shex)
```

Contenido de la variable de salida actual:

```
shex = "\ff ff ff\"
```

NUMTOIP

El comando NUMTOIP convierte un número numérico en una dirección IPv4 y ubica la dirección IP en una cadena.

Formato

```
NUMTOIP(i_integer, ip_address)
```

Tipos de datos

Argumento	Tipo	Descripción
i_integer	numérico (ENTRADA)	Valor entero que se convertirá en una dirección IPv4.
ip_address	svar (SALIDA)	Dirección IPv4 de cadena.

Por ejemplo:

En el siguiente ejemplo, el número decimal 16777215 se convierte en una dirección IPv4.

```
SET(i_integer = 167772161)
NUMTOIP(i_integer, s)
```

Contenido de la variable de salida actual:

```
s = "10.0.0.1"
```

El comando IPTONUM convierte una dirección IP en un número. Para obtener más información, consulte IPTONUM.

PARSER_ATTACHVARIABLE

El comando PARSER_ATTACHVARIABLE permite que el nombre de un par nombre - valor se relacione con una target_variable.

En la mayoría de los casos, se propone crear un analizador y adjuntar una variable en el estado de inicialización fuera del bucle. De este modo, puede volver a utilizar ese analizador al usarlo en el bucle de análisis.

Para obtener información sobre comandos de análisis relacionados, consulte los comandos PARSER_CREATEBASIC y PARSER_PARSESTRING.

Analizador NVP (par nombre - valor)

El siguiente fragmento de código muestra el analizador NVP:

```
PARSER_CREATEBASIC (h_nvp, "nvp", "separator==",
   "entry_separator= ", "value_quotes=/"",
   value_quotes_optional=yes")
PARSER_ATTACHVARIABLE (h_nvp,"esto",s_this)
PARSER_ATTACHVARIABLE (h_nvp,"yo",s_me)
PARSER_ATTACHVARIABLE (h_nvp,"hola",s_hello)
PARSER_PARSESTRING (h_nvp, "esto=/”aquel/” yo=/"usted = a
   ellos/" hola=/”adiós/”")
```

Parámetros

Los siguientes parámetros se reconocen cuando aparecen en este formato:

```
"<parámetro>=<valor>"
```

<parámetro> es uno de los siguientes elementos y <valor> es un valor adecuado para ese parámetro.

- separator: el carácter que utiliza para separar el nombre del valor.
- entry_separator: el carácter que utiliza para separar un par nombre - valor del siguiente.
- name_quotes: el carácter que utiliza para escribir el nombre (“o”, por ejemplo).
- value_quotes: el carácter que utiliza para escribir el valor.
- name_quoted: se define en yes (sí) para que el analizador NVP observe la opción name_quotes.
- value_quoted: se define en yes (sí) para que el analizador NVP observe la opción value_quotes.
- name_quotes_optional: se define en yes (sí) para admitir la opción de comillas en el nombre. Si se define en yes (sí) y se omiten las comillas, el espacio en blanco opcional seguido del separador termina el nombre.
- value_quotes_optional: se define en yes (sí) para admitir la opción de comillas en el nombre.

Si se define en yes (sí) y se omiten las comillas, el espacio en blanco opcional seguido de entry_separator termina el valor.

Formato

```
PARSER_ATTACHVARIABLE(<parser_handle>, <name>,
   <target_variable>)
```

Tipos de datos

Argumento	Tipo	Descripción
parser_handle	variable de cadena (ENTRADA)	La variable de referencia de un analizador creado.
name	cadena (ENTRADA)	El nombre de un par nombre - valor.
target_variable	cualquier variable (SALIDA)	La variable que se definirá con el valor relacionado con el nombre de un par nombre - valor.

A continuación, se incluye un ejemplo del analizador de Checkpoint.

```
ESTADO DE CONFIGURACIÓN DEL RECOPILADOR:
PARSER_CREATEBASIC(h_nvp,"nvp", "separator==",
   "entry_separator= ", "value_quotes=/"",
   "value_quotes_optional=yes")
PARSER_ATTACHVARIABLE(h_nvp,"action", s_EVT)
PARSER_ATTACHVARIABLE(h_nvp,"d_port", s_DP)
PARSER_ATTACHVARIABLE(h_nvp,"proto", s_P)
PARSER_ATTACHVARIABLE(h_nvp,"src", s_SIP)
PARSER_ATTACHVARIABLE(h_nvp,"dst", s_DIP

ESTADO DE ANÁLISIS:
PARSER_PARSESTRING(h_nvp,s_RXBufferString)
```

PARSER_CREATEBASIC

El comando PARSER_CREATEBASIC define un analizador y lo relaciona con un parser_handle. Para obtener más información, consulte el Analizador NVP (par nombre - valor) en PARSER_ATTACHVARIABLE.

En la mayoría de los casos, se propone crear un analizador y adjuntar una variable en el estado de inicialización fuera del bucle. De este modo, puede volver a utilizar ese analizador al usarlo en el bucle de análisis.

Para obtener información sobre comandos de análisis relacionados, consulte el comando PARSER_PARSESTRING.

Formato

```
PARSER_CREATEBASIC(<parser_handle>, <parser_name>, [, <nvp>
   [, ...]])
```

Tipos de datos

Argumento	Tipo	Descripción
parser_handle	variable de cadena (SALIDA)	La variable con la que hará referencia a este analizador de aquí en adelante.
parser_name	cadena (ENTRADA)	El nombre de cadena del analizador simple que está creando. **NOTA:** En este momento, sólo se reconoce nvp.
nvp	cadena (ENTRADA) (OPCIONAL)	El par nombre - valor. Ninguna o más cadenas que contienen un nombre de propiedad, seguido de un signo igual y un valor. Los parámetros reconocidos se determinan por el parser_name seleccionado. **NOTA**: Cuando el nombre de analizador se define en nvp, debe utilizar los siguientes argumentos: "separator==" "entry_separator= " "value_quotes=/"" "value_quotes_optional=yes"
nvp1	cadena (ENTRADA) (OPCIONAL)	El par nombre - valor 1.
nvp2	cadena (ENTRADA) (OPCIONAL)	El par nombre - valor 2.
…	cadena (ENTRADA) (OPCIONAL)	Otros pares nombre - valor.

Para obtener un ejemplo, consulte un ejemplo de analizador de Checkpoint en PARSER_ATTACHVARIABLE, Tipo de datos.

PARSER_NEXT

El comando PARSER_NEXT adelanta el analizador a la posición siguiente en la cadena de análisis y rellena las variables definidas por el comando PARSER_ATTACHVARIABLE.

Formato

```
PARSER_NEXT(<parser_handle>, <success_flag>)
```

Tipo de datos

Argumento	Tipo	Descripción
parser_handle	variable de cadena (ENTRADA)	La variable de referencia de un analizador creado.
success_flag	numvar (ENTRADA)	0: análisis incorrecto 1: análisis correcto

PARSER_PARSESTRING

El comando PARSER_PARSESTRING procesará la string_to_parse con el analizador creado al que hace referencia el parser_handle. Esto permite crear cualquier cadena arbitraria para su análisis, antes que insistir con un origen de secuencia o el buffer de recepción.

Para obtener más información, consulte los comandos PARSER_ATTACHVARIABLE y PARSER_CREATEBASIC.

La variable reservada s_RXBufferString puede utilizarse como una string_to_parse después del Estado de recepción para analizar la entrada del guión. Para obtener más información, consulte el Analizador NVP (par nombre - valor) en PARSER_ATTACHVARIABLE.

Formato

```
PARSER_PARSESTRING(<parser_handle>, <string_to_parse>)
```

Tipos de datos

Argumento	Tipo	Descripción
parser_handle	string variable (ENTRADA)	La variable de referencia de un analizador creado.
string_to_parse	string (ENTRADA)	La cadena única que se ejecutará a través de este analizador.

Para obtener un ejemplo, consulte un ejemplo de analizador de Checkpoint en PARSER_ATTACHVARIABLE, Tipo de datos.

PAUSE

El comando PAUSE provoca que el guión actual pause de inmediato “n” número de segundos. El comando PAUSE se ejecuta entre instrucciones en un estado de análisis y entre estados. El comando PAUSE es útil para definir los ciclos de sondeo o para garantizar que el usuario no realice sondeos demasiado rápido (por ejemplo, al sondear un registro de base de datos).

Puede especificar diferentes comandos PAUSE durante el análisis.

Formato

```
PAUSE(iseconds)
```

Argumento	Tipo	Descripción
iseconds	numérico (ENTRADA)	Número de segundos que permanecerá en pausa antes de pasar al siguiente estado.

Por ejemplo:

```
PAUSE(10)
PAUSE(iseconds)
```

O

```
IF(slowing=true)
pause(50)
ENDIF( )
```

POPUP

El comando POPUP muestra el contenido de una cadena a una pantalla en una ventana de texto desplazable.

Formato

```
POPUP(data [, title])
```

Tipos de datos

Argumento	Tipo	Descripción
data	string (ENTRADA)	El mensaje de cadena de datos que se colocará en la ventana emergente.
title	string (ENTRADA) [OPCIONAL]	La cadena que se utilizará como título de la ventana emergente (título predeterminado = “Popup DATA” (Datos de Popup)).

Por ejemplo:

```
POPUP(data)
POPUP("Hola mundo", "cadena de título")
POPUP(data, title)
```

PRINTF

El comando PRINTF copia datos con formato en una variable de cadena (svar). El comando PRINTF es un comando avanzado de análisis. Si es principiante con el lenguaje de comandos de análisis, considere utilizar el comando COPY y el comando APPEND hasta que se sienta cómodo con el lenguaje.

Al utilizar este comando:

- Especifique una svar como cadena de destino.
- Especifique una cadena de formato.
- Especifique cualquier otro parámetro opcional que explorar de acuerdo con la cadena de formato.

Cadena de formato

Para utilizar los datos HEXADECIMALES en la cadena de formato, utilice la siguiente convención:

```
\HX HX HX\
```

Si desea incluir una alimentación de línea al final de la cadena de formato, la cadena de formato debe aparecer como se muestra a continuación:

```
Format String\0a\
```

La cadena de formato para un retorno de carro es \0d0a\, por ejemplo:

```
PRINTF(mensaje,"El voltaje es %lf \0d0a\",f_volts)
```

La cadena de formato para una tabulación es \09\, por ejemplo:

```
PRINTF(mensaje,"Voltaje = \09\ %lf",f_volts)
```

Formato

```
PRINTF(dest, format [, <paramList>])
```

donde:

```
<paramList> ::= var [, <paramList>]
```

Tipos de datos

Argumento	Tipo	Descripción
dest	svar (SALIDA)	La variable de cadena de destino en la que se ubica la cadena con formato.
format	string (ENTRADA)	El formato de la cadena que se copiará en la variable de cadena de destino. Similar al formato del comando C printf; por ejemplo, “Ejecución en bucle de %d en %s” (consulte Caracteres % para el formato de salida).
parm1	todos (ENTRADA) [OPCIONAL]	Todos los tipos de datos, excepto matriz. Debe coincidir con la cadena de formato.
parm2	todos (ENTRADA) [OPCIONAL]	Todos los tipos de datos, excepto matriz. Debe coincidir con la cadena de formato.
…	todos (ENTRADA) [OPCIONAL]	Todos los tipos de datos, excepto matriz. Debe coincidir con la cadena de formato.

Formato

```
Caracteres % para el formato de salida
```

Carácter	Tipo	Formato de salida
%d	entero	Entero decimal con signo.
%le	flotante	Valor con signo con el formato [-]d.dddd e [signo]ddd ... donde d corresponde a un simple dígito decimal, dddd corresponde a uno o más dígitos decimales, ddd corresponde a exactamente tres dígitos decimales, y el signo corresponde a + o -.
%lf	flotante	Valor con signo con el formato [-]dddd.dddd ... donde dddd corresponde a uno o más dígitos decimales. El número de dígitos antes del punto decimal depende de la magnitud del número y, el número de dígitos después del punto decimal depende de la precisión solicitada.
%lg	flotante	Valor con signo impreso en formato f o e, el que sea más conciso para el valor y la precisión determinados. El formato e se utiliza sólo cuando el exponente del valor es inferior a -4 o superior o igual al argumento de precisión. Los ceros finales se truncan, el punto decimal aparece sólo si uno o más dígitos lo siguen.
%s	string	Impresión de una variable de cadena.

Visualización de dígitos de precisión

De forma predeterminada, el comando PRINTF muestra un número de punto flotante hasta seis dígitos de precisión. Los seis dígitos de precisión predeterminados también se aplican a los números de precisión doble.

Para mostrar más dígitos de precisión, especifique un valor para el campo de precisión en la especificación de formato de PRINTF():

```
%[<width>][.<precision>] type>
```

Por ejemplo:

```
PRINTF(dest, "%2.3lf", fvar)
```

generaría esta salida: 22.012, y representa 2 posiciones a la izquierda del punto decimal y 3 posiciones a la derecha del punto decimal.

En los siguientes ejemplos, se muestra cómo pasar las variables de enteros y de cadena.

```
PRINTF(dest,format_string) PRINTF(mystring,
   "val de matriz[%d][%d] = %s",
   index_x, index_y, matrix[index_x][index_y])
   PRINTF(dest,"Ejecución en bucle de %d en estado
   %s",iloop,state) PRINTF(dest,"Se ha creado el formato de
   %s datos en %s","string","dest")
```

En el siguiente ejemplo, se muestra cómo pasar una variable de punto flotante a una cadena.

```
PRINTF(message,"El voltaje es %lf",f_volts)
```

Para imprimir números de punto flotante, utilice %lf o %le.

REGEXPREPLACE

El comando REGEXPREPLACE busca y reemplaza cadenas, mediante expresiones regulares. Cuando la búsqueda encuentra la cadena, reemplaza la cadena regexpreplace. El comando REGEXPREPLACE hace un reemplazo global, no sólo un reemplazo de la primera instancia.

Formato

```
REGEXPREPLACE(dest_string, search, replace)
```

Tipos de datos

Argumento	Tipo	Descripción
dest_string	svar (ENTRADA/ SALIDA)	La variable de cadena en la que se reemplazarán los bytes.
search	string (ENTRADA) O svar (ENTRADA/ SALIDA)	La cadena de búsqueda que se reemplazará.
replace	string (ENTRADA) O svar (ENTRADA/ SALIDA)	La cadena de reemplazo; puede contener una longitud de valor cero para indicar que la cadena es nula.

Por ejemplo:

```
COPY(string:"La primera vez")
REGEXREPLACE(string, "primera", "segunda")
```

Resultado:

```
string = "La segunda vez"
```

NOTA: En este ejemplo, puede reemplazar una expresión regular por la "primera" cadena.

Para reemplazarla con la cadena nula

```
COPY(string:"La primera vez")
REGEXPREPLACE(string, "primera", "")
```

Resultado:

```
string="La vez"
```

Para obtener más información sobre expresiones comunes y el conjunto de caracteres transferibles, consulte Expresiones regulares.

Sentinel utiliza una biblioteca compatible con POSIX (Portable Operating System Interface for UNIX) para las expresiones regulares. POSIX es un conjunto de normas IEEE e ISO que permiten garantizar la compatibilidad entre sistemas operativos compatibles con POSIX, que incluye casi todas las variedades de UNIX.

REGEXPSEARCH, REGEXPSEARCH_EXPLICIT o REGEXPSEARCH_STRING

El comando REGEXPSEARCH realiza una búsqueda hacia adelante en el buffer de recepción o en la variable designada de cadena de entrada para una cadena, mediante expresiones regulares. También admite grupos de expresiones.

NOTA: En el Editor visual del Generador de recopiladores, REGEXPSEARCH, REGEXPSEARCH_EXPLICIT o REGEXPSEARCH_STRING aparecen como comandos independientes. Se trata de los mismos comandos. Se proporcionan como descripciones de diferentes variaciones del mismo comando. Si va a utilizar REGEXPSEARCH_EXPLICIT o REGEXPSEARCH_STRING en el editor de texto, introducirá REGEXPSEARCH.

Buffer de recepción

La búsqueda en el buffer de recepción se realiza de la siguiente forma:

- La búsqueda comienza en la posición actual del puntero Rx buffer y sigue buscando hacia adelante hasta que encuentra la cadena o hasta que llega al final del buffer de recepción.
- Si la búsqueda encuentra la cadena, el puntero Rx buffer se actualiza para indicar el primer byte de la cadena de búsqueda. Se conserva esta posición del puntero Rx buffer al pasar por los estados, a menos que se cambie explícitamente al utilizar el comando RESET.
- Si la búsqueda no encuentra la cadena, el puntero Rx buffer no se mueve.

Al utilizar este comando para buscar el buffer de recepción, el segundo parámetro opcional es una variable de enteros que se define en 1 si la búsqueda encuentra la cadena, y se define en 0 si la búsqueda no encuentra la cadena.

Variable de cadena

Las variables de cadena no admiten el puntero de análisis, por lo tanto, las dinámicas al realizar búsquedas en una variable de cadena son diferentes. El patrón de expresiones regulares coincidirá con algunas o todas las cadenas de entrada. Si el patrón de expresiones regulares se configura con grupos de expresiones, el contenido de la cadena de entrada que coincide con los grupos de expresiones puede almacenarse en las variables de salida. Existen dos opciones de salida de grupos de expresiones. Una es rellenar la lista de variables por orden de grupo de expresiones, y la otra es designar una matriz de cadenas.

Si la expresión regular coincide con la variable cadena - entrada, una lista designada de variables o una matriz de salida se define con los valores del grupo, y la variable encontrada se define en un número mayor que el número de grupos, o en cero si no se encuentra ninguna coincidencia.

Cuando la salida de los valores del grupo es una matriz de cadenas, el primer elemento indexado con “0” contendrá la cadena de coincidencia. La cadena de coincidencia contendrá el contenido que coincidió con toda la expresión regular independiente de los grupos de expresiones. Por lo tanto, el contenido del primer grupo de expresiones se almacenará en la posición de la matriz indexada con “1”. Al ejecutar un bucle a través de la matriz de salida, tenga en cuenta que el valor i_Found_Tokens compensa el primer elemento y representa la cadena de coincidencia al ser un número mayor que el número total de grupos. En un bucle “for”, se considerará la condición de parada de ser inferior al valor i_Found_Tokens, pero es posible que tenga que iniciar el índice en “1” en lugar de “0”.

Al designar los valores del grupo que se almacenarán en una lista de variables de salida en lugar de una matriz, el comando puede realizar la conversión de tipo. Si bien la cadena de entrada es de tipo cadena, los componentes de la cadena pueden ser números. Si la intención es tratar esos números como enteros o valores de punto flotante, el simple hecho de designar las variables de salida con el tipo adecuado dará como resultado una conversión.

Coincidencia simple REGEX

Expresión	Descripción
.	Cualquier carácter
\d	Cualquier dígito
\w	Cualquier carácter alfanumérico
\s	Cualquier espacio en blanco
+	1 o más de los caracteres anteriores
*	0 o más de los caracteres anteriores

Formato

Como buffer de recepción:

```
REGEXPSEARCH(search[, ifound])
```

Como variable de cadena:

```
REGEXPSEARCH(Input_String, s_Regular_Exp_Pattern,
   i_Found_Tokens[, s_Output_Results[]])
REGEXPSEARCH(s_Input_String, s_Regular_Exp_Pattern,
   i_Found_Tokens, s_Match[, var1, var2, ...)]
```

Tipos de datos

Argumento	Tipo	Descripción
s_Input_String	Cadena o variable de cadena (ENTRADA) [OPCIONAL]	La cadena o variable de cadena que se buscará para detectar coincidencias regex especificadas en regex.
s_Regular_Exp_Pattern	Cadena (ENTRADA)	La cadena que se buscará en el buffer de recepción (búsqueda a partir de la posición del puntero Rx buffer en adelante) o un literal de cadena de entrada, o una variable de cadena de entrada.
i_Found_Tokens	numvar (SALIDA) [OPCIONAL]	Devuelve si la cadena de búsqueda se encontró o no. 0: El patrón de expresiones regulares no coincide. 1: El patrón de expresiones regulares coincide, pero no se designaron grupos de expresiones. 2: El patrón de expresiones regulares coincide con 1 grupo de expresiones designado. N+1: El patrón de expresiones regulares coincide con N grupos de expresiones designados. **NOTA:** La variable I_found_tokens puede utilizarse como una prueba para buscar coincidencias, ya que el valor no será cero cuando la expresión regular coincida.
s_Match	Cadena (SALIDA) [CONDICIONAL]	Sólo se rellena al coincidir con el patrón y debe designarse cuando se utiliza una lista de variables de salida de grupo de expresiones. Cuando los valores del grupo se almacenan en una matriz de salida, s_Match NO es un parámetro válido.
Variable List O s_Output_Results[]	Son todos posibles. (SALIDA) [OPCIONAL] O Matriz de cadena (SALIDA) [OPCIONAL]	La lista de variables en la que se ubicarán los valores del grupo. El valor se asigna por orden de valores del grupo designados cuando se cumplen las reglas de prioridad.

En el siguiente ejemplo, se busca un retorno de carro y un avance de línea en el buffer de recepción:

```
REGEXPSEARCH("\0d0a\")
```

En el siguiente ejemplo, se busca la palabra alarma en el buffer de recepción:

```
REGEXPSEARCH("alarma")
```

NOTA: Para la sustitución de hexadecimales, \0000\ termina una cadena; por lo tanto, “xxxx\0000\yyyy” se convierte en “xxxx”.

A continuación, se describe un ejemplo detallado de búsqueda de un patrón en un valor de cadena literal:

```
REGEXPSEARCH("15 de ene. de 2003 13:34:20",
   "(/\d+)/\s+(/\w+)/\s+(/\d+)/\s+(/\d+):(/\d+):(/\d+)",
   i_Success, s_Match, s_Year, s_Month, s_Day, s_Hour,
   s_Minute, s_Second)
```

Donde,

```
i_Success = 7
s_Match = 15 de enero de 2003 13:34:20
s_Year = 2003
s_Month = ene.
s_Day = 15
s_Hour = 13
s_Minute = 34
s_Second = 20
```

Para obtener más información sobre expresiones regulares y el conjunto de caracteres transferibles, consulte la sección Expresiones regulares del capítulo 2.

Sentinel utiliza una biblioteca compatible con POSIX (Portable Operating System Interface for UNIX) para las expresiones regulares. POSIX es un conjunto de normas IEEE e ISO que permiten garantizar la compatibilidad entre sistemas operativos compatibles con POSIX, que incluye casi todas las variedades de UNIX.

REPLACE

El comando REPLACE busca y reemplaza cadenas.

Cuando la búsqueda encuentra la cadena, reemplaza la cadena de reemplazo. El comando REPLACE hace un reemplazo global, no sólo un reemplazo de la primera instancia.

Formato

```
REPLACE(dest_string, search, replace)
```

Tipos de datos

Argumento	Tipo	Descripción
dest_string	svar (ENTRADA/ SALIDA)	La variable de cadena en la que se reemplazarán los bytes.
search	string (ENTRADA)	La cadena de búsqueda que se reemplazará.
replace	string (ENTRADA)	La cadena de reemplazo.

Por ejemplo:

```
COPY(string:"La primera vez")
REPLACE(string, "primera", "segunda")
```

Resultado:

```
string = "La segunda vez"
```

NOTA: En este ejemplo, puede reemplazar una expresión regular por la cadena “primera”.

RESET

El comando RESET restablece el puntero Rx buffer a cero.

Formato

```
RESET()
```

Por ejemplo, el símbolo ^ identifica la posición del puntero Rx buffer.

```
rxbuff = "abcdefg"
                ^
RESET()
```

Resultado:

```
"abcdefg"
 ^
```

RXBUFF

El comando RXBUFF sobrescribe el buffer de recepción con el contenido de una cadena entrecomillada o una variable de cadena. El contenido del buffer de recepción se modificará de inmediato; el puntero Rx buffer y el valor de retención se restablecerán a cero.

Formato

```
RXBUFF(s_data)
```

Tipos de datos

Argumento	Tipo	Descripción
s_data	string (ENTRADA)	La cadena de datos que se escribirá en el buffer de recepción. Esta cadena pasará a ser la nueva cadena del buffer de recepción.

Por ejemplo:

En el siguiente ejemplo, el comando FILER lee un archivo llamado alert.data y coloca el contenido de ese archivo en una variable de cadena llamada s_data. Este ejemplo se basa en el siguiente supuesto:

```
alert.data: "Minor Alarm Xterminal A")
```

A continuación, el comando RXBUFF coloca los datos en el buffer de recepción, como si los datos se recibieran desde un puerto.

```
FILER("alert.data", s_data)
RXBUFF(s_data)
//copia datos del BUFFER de recepción en S_Alarm_Priority,
   se detiene antes de la cadena "Alarm")
COPY(S_Alarm_Priority:," Alarm")
```

Resultado:

```
S_Alarm_Priority= "Minor"
```

SEARCH

El comando SEARCH realiza una búsqueda hacia adelante de una cadena en el buffer de recepción.

La búsqueda se realiza de la siguiente forma:

- La búsqueda comienza en la posición actual del puntero Rx buffer y sigue buscando hacia adelante hasta que encuentra la cadena o hasta que llega al final del buffer de recepción.
- Si la búsqueda encuentra la cadena, el puntero Rx buffer se actualiza para indicar el primer byte de la cadena de búsqueda. Se conserva esta posición del puntero Rx buffer al pasar por los estados, a menos que se cambie explícitamente al utilizar el comando RESET.
- Si la búsqueda no encuentra la cadena, el puntero Rx buffer no se mueve.

Al utilizar este comando, el segundo parámetro opcional es una variable de enteros que se define en 1 si la búsqueda encuentra la cadena, y se define en 0 si la búsqueda no encuentra la cadena.

Formato

```
SEARCH(search[, ifound])
```

Tipos de datos

Argumento	Tipo	Descripción
search	string (ENTRADA)	La cadena que se buscará en el buffer de recepción (a partir de la posición actual del puntero Rx buffer hacia adelante).
ifound	numvar (SALIDA) [OPCIONAL]	Devuelve si la cadena de búsqueda se encontró o no. 0 = no se encontró 1 = se encontró

Por ejemplo:

En el siguiente ejemplo, se busca un retorno de carro y un avance de línea.

```
SEARCH("\0d0a\")
SEARCH(data, ifound)
```

En el siguiente ejemplo, se busca la palabra alarma:

```
SEARCH("alarma")
```

NOTA: Para la sustitución de hexadecimales, \0000\ termina una cadena; por lo tanto, “xxxx\0000\yyyy” se convierte en “xxxx”.

SET

El comando SET procesa una expresión matemática y actualiza un valor numérico (numvar) con el resultado de la evaluación.

Al utilizar este comando:

- Especifique una numvar de destino, seguida de un signo igual, seguida de una combinación de ( ) - + * /, números y variables numéricos.
- Debe especificar al menos un valor numérico a la derecha del signo igual.
- No hay ninguna restricción para el número de paréntesis incorporados.
- Todos los argumentos se convierten en un valor flotante; el resultado se convierte en el tipo (entero o flotante) de la numvar de destino.
- Se puede introducir hasta 98 entradas después del signo igual; estas entradas incluyen: (,), *, /, +, -, cualquier valor numérico y variables numéricas.
- Cuando las operaciones tienen el mismo orden de nivel de operación, se gestionan de izquierda a derecha; el orden de operación se describe en la siguiente tabla.

Nivel 1	:	()	por ejemplo: paréntesis
Nivel 2	:	*/	por ejemplo: multiplicación, división
Nivel 3	:	+ -	por ejemplo: suma, resta

Formato

```
SET(idest = <expr>) o SET(fdest = <expr>)
```

Donde:

```
set_command ::= SET(<idest>=<expr>) | SET(<fdest>=<expr>)
expr ::= (<expr>)
         | expr ( '+' | '-' | '*' | '/' ) expr
         | ivar | fvar | número
```

Tipo de datos

Argumento	Tipo	Descripción
idest	numvar (SALIDA)	La variable numérica (fvar o ivar) en la cual se guardará el valor.
inum1	numérico (ENTRADA)	Una fvar, ivar o número.
inum2	numérico (ENTRADA) [OPCIONAL]	Una fvar, ivar o número.

Argumento	Tipo	Descripción
inum3	numérico (ENTRADA) [OPCIONAL]	Una fvar, ivar o número.
…	numérico (ENTRADA) [OPCIONAL]	Una fvar, ivar o número.

Por ejemplo:

```
SET(idest=inum1)
SET(i_loop=10)
SET(idest=inum1+inum2)
SET(idest=(inum1+inum2) * inum3)
SET(i_counter=i_counter+1)
SET(i_val = (ivar)*(ivar/3) + 15/fvar - (5 + 20/iloop))
```

SETBYTES

El comando SETBYTES permite definir bytes en una variable de cadena en un valor específico, si se transfirió como entero o como cadena. Si se transfirió como entero, los rangos válidos oscilan entre 0 y 255. Si se utilizó una cadena como parámetro de reemplazo, la cadena se ubica a partir de la posición de índice en la variable de cadena de destino.

Formato

```
SETBYTES(dest_string, index, replace)
```

Tipos de datos

Argumento	Tipo	Descripción
dest_string	svar (ENTRADA/ SALIDA)	La variable de cadena en la que se reemplazarán los bytes.
index	numérico (ENTRADA)	El índice (se cuentan los bytes a partir de 0 para el primer byte) en dest_string en el cual se utilizarán los bytes para realizar el reemplazo.
replace	string (ENTRADA) O entero (ENTRADA)	Los bytes de la cadena que se escribirán en la dest_string. El valor que se definirá para el byte #n del índice en la cadena de destino.

Por ejemplo:

```
COPY(string:"Uso del ancho de banda = 22 %")
SETBYTES(string, 18, "44")
```

Contenido de las variables de salida actuales:

```
string = "Uso del ancho de banda = 44 %"
```

SETCONFIG

Este comando define una propiedad del sistema. La configuración actual de la propiedad del sistema puede recuperarse con el comando GETCONFIG. Estos comandos se utilizan para definir propiedades del sistema y recuperar los valores actuales de las propiedades del sistema que pueden modificarse con frecuencia, por ejemplo, un archivo de registro, cuyo nombre se cambia todos los días con la fecha actual.

Las propiedades del sistema disponibles son:

Propiedad del sistema	Ejemplos
▪ System.OS.Family	Solaris y Windows
▪ System.OS.Name	Windows 2000
▪ System.OS.Version.Major	5
▪ System.OS.Version.Minor	0
▪ System.Net.Hostname	ESECServer
▪ System.Net.IP_List	lista de direcciones IP para este host separadas por un punto y coma, por ejemplo, “172.163.3.45;172.45.2.1”

Consulte el comando GETCONFIG.

Existen dos parámetros con este comando.

- El primer parámetro requerido define la opción de configuración (“FileConnector.InputFile” o “FileConnector.OutputFile”) que se establecerá.
- El segundo parámetro requerido define el valor de configuración que se establecerá.

Formato

```
SETCONFIG(Config Option, Value)
```

Tipos de datos

Argumento	Tipo	Descripción
Config Option	cadena (ENTRADA)	Nombre de la variable de configuración que se establecerá. Archivo de entrada = “FileConnector.InputFile” Archivo de salida = “FileConnector.OutputFile”
Valor	string svar (ENTRADA)	Valor de configuración.

Por ejemplo:

```
SETCONFIG(“FileConnector.InputFile”, s_inputfilename)
SETCONFIG(“FileConnector.OutputFile”, s_outputfilename)
```

Contenido de las variables de salida actuales:

```
“C:/\prueba.dat”
```

SHELL

El comando SHELL ejecuta una secuencia de comandos shell o un comando.

Formato

```
SHELL(command [, wait_parameter][, wait_return_status])
```

Tipos de datos

Argumento	Tipo	Descripción
command	string (ENTRADA)	La ruta y el nombre de archivo del comando que se ejecutará. De forma predeterminada, se utiliza la variable de entorno PATH.
wait/no_wait	numvar [OPCIONAL]	Permite que el comando SHELL espere (o no espere) a que se inicie el programa para completar la ejecución antes de seguir con el proceso. 0 = sin espera 1 = espera a que finalice el programa
return_status	numvar [OPCIONAL]	Valor numérico al utilizar la opción wait/no_wait. EJECUCIÓN CORRECTA = 1 EJECUCIÓN INCORRECTA = 0

En el siguiente ejemplo, se inicia un archivo por lote de PC o una secuencia de comandos shell de UNIX:

```
SHELL("device_poll")
```

En el siguiente ejemplo, se inicia Notepad:

```
SHELL("c:/\winnt/\system32/\notepad.exe")
```

En el siguiente ejemplo, se espera que el comando clock finalice la ejecución:

```
SHELL("clock",1)
```

En el siguiente ejemplo, se espera que un archivo por lote de PC o una secuencia de comandos shell de UNIX finalice la ejecución, luego se obtiene el estado de devolución:

```
SHELL("device_poll",1,i_ret)
```

En el siguiente ejemplo, se ejecuta el proceso de clock y no espera a que finalice:

```
SHELL("clock",0)
```

SKIP

El comando SKIP agrega un número al valor del puntero Rx buffer.

El número puede ser positivo o negativo. Si la posición resultante del puntero Rx buffer es inferior a cero, el puntero Rx buffer se define en cero. Si la posición resultante del puntero Rx buffer sobrepasa el final del buffer de recepción, el puntero Rx buffer se define para indicar el último byte en el buffer de recepción.

Formato

```
SKIP([+ | -] iskip_amount)
```

Tipos de datos

Argumento	Tipo	Descripción
iskip_amount	numérico (ENTRADA)	El número de bytes que se moverá el Rx.

Por ejemplo:

```
SKIP(iskip_amount)
SKIP(+iskip_amount)
SKIP(-iskip_amount)
SKIP(5)
SKIP(-1)
```

En los siguientes ejemplos, se muestra la posición del puntero Rx buffer después de un comando skip, para los datos:

```
aaaaaa bbbbb c d ee
      ^

SKIP(-2)
aaaaaa bbbbb c d ee
    ^

SKIP(-1)
aaaaaa bbbbb c d ee
     ^

SKIP(0)
aaaaaa bbbbb c d ee
      ^

SKIP(1)
aaaaaa bbbbb c d ee
       ^
SKIP(2)
aaaaaa bbbbb c d ee
        ^

SKIP(3)
aaaaaa bbbbb c d ee
         ^
```

```
SKIP(4)
aaaaaa bbbbb c d ee
          ^

SKIP(8)
aaaaaa bbbbb c d ee
              ^
```

SKIPWORD

El comando SKIPWORD modifica el puntero Rx buffer para que indique el comienzo de una palabra.

Este comando considera que una palabra es cada secuencia de bytes continuos imprimibles separados por, al menos, un byte no imprimible. Los bytes imprimibles se definen como ASCII y ASCII-0-255 extendido (según la norma ISO 8859-1).

Al utilizar valores de skip negativos y positivos, el puntero Rx buffer omite hacia adelante y hacia atrás a través del buffer de recepción hasta el primer o el siguiente byte imprimible en el buffer de recepción.

El puntero Rx buffer no sobrepasará el final del buffer de recepción ni se ubicará antes del comienzo del buffer de recepción, aunque el comando SKIPWORD lo provocara.

Un valor cero no provoca que el puntero Rx buffer cambie. El comando SKIPWORD trata todos los caracteres inferiores a 33, y entre 126 y 161 como espacios en blanco.

Formato

```
SKIPWORD([+ | -] iwords)
```

Tipos de datos

Argumento	Tipo	Descripción
iwords	numérico (ENTRADA)	El número de palabras que moverá el puntero Rx buffer en el buffer de recepción.

Por ejemplo:

```
SKIPWORD(iwords)
SKIPWORD(3)
SKIPWORD(+iwords)
SKIPWORD(-iwords)
SKIPWORD(-4)
```

En los siguientes ejemplos, se muestra la posición del puntero Rx buffer después de un comando SKIPWORD, para los datos:

```
aaaaaa bbbbb c d ee
      ^

SKIPWORD(-2)
aaaaaa bbbbb c d ee
^

SKIPWORD(-1)
aaaaaa bbbbb c d ee
^

SKIPWORD(0)
aaaaaa bbbbb c d ee
      ^

SKIPWORD(1)
aaaaaa bbbbb c d ee
       ^

SKIPWORD(2)
aaaaaa bbbbb c d ee
             ^

SKIPWORD(3)
aaaaaa bbbbb c d ee
               ^

SKIPWORD(4)
aaaaaa bbbbb c d ee
                 ^

SKIPWORD(5)
aaaaaa bbbbb c d ee
                  ^
```

SOCKETW

El comando SOCKETW abre, conecta, escribe datos SIN BLOQUEO (socket STREAM por byte de red) en un socket (puerto IP y TCP) y cierra el socket. De manera opcional, devuelve el estado del intento de escritura en el socket.

Formato

```
SOCKETW(address, i_port, data [, istat])
```

Tipos de datos

Argumento	Tipo	Descripción
address	string (ENTRADA)	Dirección IP del socket.
i_port	numérico (ENTRADA)	Número de puerto TCP del socket.
data	string (ENTRADA)	La cadena de datos que se escribirá en el socket.
istat	numvar (SALIDA)	Estado opcional devuelto. istat = número de bytes escritos; > 0 (EJECUCIÓN CORRECTA) istat = 0 (EJECUCIÓN INCORRECTA)

Ejemplos:

```
SOCKETW("192.168.15.25", 5051, "Datos que se escribirán en
   el socket")
SOCKETW("192.168.15.25", i_port, "Datos al socket\0d\")
SOCKETW(s_ip_address, i_port, "\54AF0D0B91\", i_status)
SOCKETW(s_ip_address, i_port, "\54AF0D0B91\", f_status)
SOCKETW(s_ip_address, 6004, "\54AF0D0B91\", f_status)
SOCKETW(s_ip_address, 6004,sdata, f_status)
```

STONUM

El comando STONUM (cadena a número) convierte una variable de cadena (svar) en una variable numérica (numvar).

PRECAUCIÓN: Las variables de cadena que constan de algún valor diferente a la representación de cadena de un entero o coma flotante pueden causar resultados inesperados. Todos los valores enteros se limitan a 2147483647; los valores superiores a este se convierten en 2147483647.

Formato

```
STONUM(string, ivar)
```

Tipos de datos

Argumento	Tipo	Descripción
inum	numvar (SALIDA)	La variable numérica en la cual se guardará el número (ivar o fvar).
string	string (ENTRADA)	La representación de cadena de un número (por ejemplo: “306”).

Por ejemplo:

```
STONUM(source, idest)
STONUM(string_number, ivar)
STONUM("6512", ivar)
```

STRIP o STRIP-ASCII-RANGE


El comando STRIP quita todas las instancias de una cadena strip o un rango ASCII de la svar. El comando STRIP siempre realiza eliminaciones a paso múltiple hasta que la cadena strip o rango ASCII no se puede encontrar en la variable de cadena de destino.

Al utilizar este comando, especifique la variable de cadena de la cual se quitarán los caracteres. Los demás parámetros pueden ser una cadena o un valor inicial o final del rango numérico.

NOTA: En el Editor visual del Generador de recopiladores, STRIP y STRIP-ASCII-RANGE aparecen como comandos independientes. Se trata de los mismos comandos. Se proporcionan como descripciones de diferentes variaciones del mismo comando. Si está por utilizar STRIP-ASCII-RANGE en el editor de texto, introducirá STRIP.

Formato

```
STRIP(dest, strip)
STRIP(dest, start ASCII range, stop ASCII range)
```

Tipos de datos

Argumento	Tipo	Descripción
dest	svar (ENTRADA/ SALIDA)	La variable de cadena que contiene los datos de cadena de la que se eliminarán los bytes de acuerdo con el segundo argumento.
strip or start ASCII range	cadena o valor numérico (ENTRADA)	La cadena o valor ASCII de inicio que se quitará de la cadena de destino.
stop ASCII range	numérico (ENTRADA [opcional])	valor ASCII de detención **NOTA**: Si se especifica el rango ASCII de inicio, se necesita este parámetro.

Los siguientes ejemplos son eliminaciones a paso múltiple.

```
COPY(test:"THHELLOE")
STRIP(test, "HELLO")
```

Después del comando STRIP(), la variable test tiene el valor THE.

```
COPY(test2:"ABCDDEDDDFGDDH")
STRIP(test2, "D")
```

Después del comando STRIP(), la variable test2 tiene el valor ABCEFGH.

```
COPY(test3:"ABCDDEDDDFGDDH")
STRIP(test3, 68, 69)
```

Después del comando STRIP(), la variable test3 tiene el valor ABCFGH.

TBOSSETCOMMAND

El comando TBOSSETCOMMAND crea un paquete de comandos TBOS de 3 bytes que puede transferirse a un dispositivo a través del protocolo TBOS.

El tipo de comando, el número de comando y el número visualizado de TBOS se utilizan para colocar el paquete correcto de comandos TBOS (3 bytes) en la variable de cadena de salida. El formato del paquete TBOS creado con este comando de análisis se describe en las siguientes tablas de solicitud de comandos remotos.

Carácter 1		
Número de bits	**Valor**	**Significado**
8 7	0 1	Código de operación: 01 = solicitud de comando remoto (carácter 1)
6 5 4	MSB LSB	Número visualizado: 000 = N.º 1 001 = N.º 2 ... 111 = N.º 7
3	0	Sin significado
2 1	MSB LSB	Tipo: 00 = momentáneo 01 = conectar (latch) 10 = desconectar (unlatch)

Carácter 2		
Número de bits	**Valor**	**Significado**
8 7	1 0	Código de operación: 10 = solicitud de comando remoto (carácter 2)

Carácter 2		
Número de bits	**Valor**	**Significado**
6 5 4 3 2 1	MSB LSB	Número de comando remoto: 000000 = N.º 1 000001 = N.º 2 ... 111111 = N.º 63

Carácter 3		
Número de bits	**Valor**	**Significado**
8 7 6 5 4 3 2 1	1 1 0 0 1 1 0 0	Eco de carácter: La respuesta del comando remoto es el eco de este byte en el puerto.

Formato

```
TBOSSETCOMMAND(cmd_bytes, idisp_num, icmd_num, tipo)
```

Tipos de datos

Argumento	Tipo	Descripción
cmd_bytes	svar (SALIDA)	Los bytes de datos hexadecimales (3 bytes en total) que se colocarán en esta variable de cadena y que pueden utilizarse para transmisiones a un dispositivo TBOS en el casillero Next State Transmit (Transmisión del estado siguiente).
idisp_num	numérico (ENTRADA)	El número visualizado (o dirección) de TBOS del dispositivo (1 - 8). **NOTA:** Los rangos válidos de idisp_num oscilan solamente entre 1 y 8; si se utiliza cualquier otro valor, la salida (cmd_bytes), se define en ceros, "\00 00 00\".
i_cmd_num	numérico (ENTRADA)	El número de comando de TBOS (1 - 64). **NOTA:** Los rangos válidos de i_cmd_num oscilan solamente entre 1 y 64; si se utiliza cualquier otro valor, la salida (cmd_bytes), se define en ceros, "\00 00 00\".

Argumento	Tipo	Descripción
tipo	numérico (ENTRADA) O string (ENTRADA)	El tipo de comando de TBOS (0 - 2): 0 = momentáneo 1 = conectar (latch) 2 = desconectar (unlatch) **NOTA:** Los rangos válidos del tipo oscilan solamente entre 0 y 2; si se utiliza cualquier otro valor, el tipo se define en 0 = "momentáneo" de forma predeterminada. El tipo de comando de TBOS en formato de cadena. "momentáneo" o "m" = momentáneo "latch" (conectar) o "l" = latch (conectar) "unlatch" (desconectar) o "u" = latch (desconectar) Esta cadena no distingue entre mayúsculas y minúsculas.

Por ejemplo:

```
TBOSSETCOMMAND(string_cmd_bytes, 1, 1, 0)
TBOSSETCOMMAND(s_bytes, 1, 1, "latch (conectar)")
TBOSSETCOMMAND(s_bytes, i_display, i_cmd_num, "U")
TBOSSETCOMMAND(s_bytes, i_display, i_cmd_num, 2)
TBOSSETCOMMAND(s_bytes, 1, 1, "momentáneo")
TBOSSETCOMMAND(s_bytes, 1, 1, "latch (conectar)")
```

Recuerde comprobar que la salida cmd_bytes esté definida en "\00 00 00\" para verificar la existencia de errores en entradas que no se especifiquen dentro del rango. Por ejemplo:

```
TBOSSETCOMMAND(cmd_bytes, i_display, i_cmd_num, "M")
IF(cmd_bytes = "\00 00 00\") /* ENTRADAS QUE NO SE
   ESPECIFICAN DENTRO DEL RANGO */
...
ENDIF()
```

En el siguiente ejemplo, se crea un comando tbos para el número visualizado 5, el número de comando 33 y el tipo desconectado.

```
TBOSSETCOMMAND(sbytes, 5, 33, 2)
```

Contenido de las variables de salida actuales:

```
sbytes = "\ba0 cc\"
```

TBOSSETREQUEST

El comando TBOSSETREQUEST crea un paquete de solicitudes TBOS de 1 byte que puede transferirse a un dispositivo a través del protocolo TBOS. El número de solicitud y el número visualizado de TBOS se utilizan para colocar el byte de solicitud de exploración TBOS correcto en la variable de cadena de salida. El formato del paquete TBOS creado con este comando de análisis se describe en las siguientes tablas de solicitud y respuesta de exploración de caracteres.

Carácter 1 – Solicitud de exploración de caracteres		
Número de bits	**Valor**	**Significado**
8 7	0 0	Código de operación: 00 = Solicitud de exploración de caracteres
6 5 4	MSB LSB	Número visualizado: 000 = N.º 1 001 = N.º 2 ... 111 = N.º 3
3 2 1	MSB LSB	Tipo: 000 = N.º 1 001 = N.º 2 ... 111 = N.º 8

Carácter 1 – Respuesta de exploración de caracteres		
Número de bits	**Valor**	**Significado**
8 7 6 5 4 3 2 1	MSB LSB	Cada bit en este byte de respuesta tiene un significado especial según el número de carácter enviado (1 - 8) y el protocolo del dispositivo del número visualizado enviado (1 - 8).

Formato

```
TBOSSETREQUEST(cmd_bytes, idisp_num, irequest_num)
```

Tipos de datos

Argumento	Tipo	Descripción
cmd_bytes	svar (SALIDA)	El byte de datos hexadecimales se coloca en esta variable de cadena y puede utilizarse para transmisiones a un dispositivo TBOS en el casillero Next State Transmit. (Transmisión del estado siguiente).
idisp_num	numérico (ENTRADA)	El número visualizado (o dirección) de TBOS del dispositivo (1 - 8). **NOTA:** Los rangos válidos de idisp_num oscilan solamente entre 1 y 8; si se utiliza cualquier otro valor, la salida cmd_bytes se define en ceros, "\00\."
irequest_num	numérico (ENTRADA)	El número de caracteres de exploración de TBOS (1 - 8). **NOTA:** Los rangos válidos de irequest_num oscilan solamente entre 1 y 8; si se utiliza cualquier otro valor, la salida cmd_bytes se define en ceros, "\00\."

Por ejemplo:

```
TBOSSETREQUEST(string_request_byte, 1, 1)
TBOSSETREQUEST(s_byte, idisp_num, i_scan_number)
```

En el siguiente ejemplo, se crea un carácter de solicitud de exploración de TBOS para el número visualizado 2 y el número de solicitud 1.

```
TBOSSETREQUEST(sbytes, 2, 1)
```

Contenido de las variables de salida actuales:

```
sbytes = "\08\"
```

TIME

El comando TIME copia la hora actual (con el formato HH-MM-SS) en una variable de cadena, ivar o fvar.

Formato

```
TIME(dest)
```

Tipos de datos

Argumento	Tipo	Descripción
dest	svar (SALIDA) numvar (SALIDA)	La representación de cadena de la hora se coloca en esta variable de cadena (por ejemplo: "23-11-55"). El número de segundos desde 00:00:00 UTC, 1 de enero de 1970, se coloca en esta variable numérica (puede ser una fvar).

Por ejemplo:

```
TIME(time_of_day)
TIME(i_num_seconds)
TIME(f_num_seconds)
```

NOTA: Si utiliza una fvar, la hora se devolverá con precisión de microsegundos.

TOKENIZE

El comando TOKENIZE copia cada componente de una cadena entre los limitadores en una matriz de cadena. Puede ser útil al leer los datos delimitados de un archivo y al pasar datos a un guión que se ejecutará a pedido.

Cada carácter de la cadena se considera un testigo separador potencial. Por ejemplo, el uso del testigo separador "THE END" no utilizaría toda la cadena como separador. En cambio, los caracteres individuales se utilizarían como posibles separadores:

```
"T"
"H"
"E"
"E"
"N"
"D"
```

Formato

```
TOKENIZE(data, delimiter, tokens[], itokens)
```

Tipos de datos

Argumento	Tipo	Descripción
data	svar (ENTRADA)	Los datos que se colocarán entre testigos (por ejemplo: "xterm|subres|33|50").
delimiter	string (ENTRADA)	Los delimitadores que separan los testigos.
token	matriz (SALIDA)	La matriz de testigos tal como se encuentra en la cadena de datos de entrada delimitados.
itokens	numvar (SALIDA)	El número de testigos colocados en la matriz de cadena de testigos.

Por ejemplo:

```
COPY(data:"This|Data|Is|Tokenized")
TOKENIZE(data, "|",tokens[], inumtokens)
```

Contenido de las variables de salida actuales:

```
inumtokens = 4
tokens[0]= "This"
tokens[1]= "Data"
tokens[2]= "Is"
tokens[3]= "Tokenized"
```

En el siguiente ejemplo, los datos pasados al guión son:

```
"There#are|several*fields|in*this#string".
```

Existen tres separadores diferentes de símbolos que vamos a utilizar: #, | y *.

Contenido de las variables de salida actuales:

```
i_tokens = 7
messages[0] = "There"
messages[1] = "are"
messages[2] = "several"
messages[3] = "fields"
messages[4] = "in"
messages[5] = "this"
messages[6] = "string"
```

En el siguiente ejemplo, los datos en el buffer de recepción son:

```
"Firewall Alarm - Major;Denial of Service Alarm - Major;"
COPY(rxbuff:)
TOKENIZE(rxbuff,";",msgs[],i_msgs)
```

Contenido de las variables de salida actuales:

```
i_msgs = 2
msjs[0] = "Firewall Alarm - Major")
msgs[1] = "Denial of Service Alarm - Major"
```

TOLOWER

El comando TOLOWER convierte el contenido de una variable de cadena en caracteres en minúscula. El contenido de la variable de cadena que pasa por este comando se convierte a minúsculas.

Formato

```
TOLOWER(stringvar)
```

Tipos de datos

Argumento	Tipo	Descripción
stringvar	string (ENTRADA/ SALIDA)	La variable de cadena que contiene la cadena que se convertirá a minúsculas.

Por ejemplo:

```
s_var = "Caracteres En Minúscula"
TOLOWER(s_var)
```

Resultado:

```
s_var = "caracteres en minúscula"
```

TOUPPER

El comando TOUPPER convierte el contenido de una variable de cadena en caracteres en mayúscula. El contenido de una variable de cadena que pasa por este comando se convierte a mayúsculas.

Formato

```
TOUPPER(stringvar)
```

Tipos de datos

Argumento	Tipo	Descripción
stringvar	string (ENTRADA/ SALIDA)	La variable de cadena que contiene la cadena que se convertirá a mayúscula.

Por ejemplo:

```
s_var = "Caracteres En Mayúscula"
TOLOWER(s_var)
```

Resultado:

```
s_var = "CARACTERES EN MAYÚSCULA"
```

TRANSLATE

El comando TRANSLATE carga un archivo de valores separados por coma (csv) en la memoria, permite realizar una búsqueda rápida para verificar si la entrada de claves se encuentra en el archivo o no, y permite recuperar otros datos relacionados con la clave.

La siguiente información está relacionada con el comando TRANSLATE.

- Valor separado por coma (CSV)
- Búsquedas de claves que no distinguen entre mayúsculas y minúsculas.
- Estado encontrado
- Variables de datos

Archivo de valores separados por coma (CSV)

El archivo csv es una ruta relativa desde un directorio de guión del recopilador. El Generador de recopiladores no admite la edición de estos archivos; por lo tanto, Novell sugiere crearlos a través de Microsoft Excel. El nombre de archivo puede ser una cadena o una variable.

El formato del archivo csv se muestra en el siguiente ejemplo de un archivo llamado amigos.csv:

```
clave1,datos1,datos2,datos3
Roberto,azul,25,210
Alicia,verde,19,110
Patricia,violeta,36,145
```

Para buscar si algún amigo en particular se encuentra en el archivo amigos.csv, el comando TRANSLATE aparecerá de la siguiente forma:

```
TRANSLATE("Roberto","amigos.csv",i_found)
```

O

```
COPY(s_Name:"Roberto")
TRANSLATE(s_Name,"amigos.csv",i_found)
```

Búsquedas de claves que no distinguen entre mayúsculas y minúsculas.

El parámetro clave puede ser una cadena o una variable de cadena. Además, se admite un número entero o una variable. Como el archivo csv se carga en la memoria, la clave de cada entrada se define en minúsculas. La clave en el comando TRANSLATE también se define internamente en minúsculas para permitir las búsquedas de claves sin distinguir entre mayúsculas y minúsculas.

Siguiendo con el ejemplo del archivo csv:

```
TRANSLATE("roBerto","amigos.csv",i_found)
```

De esta forma, también se habría encontrado el nombre Roberto en el archivo csv.

Estado encontrado

El estado encontrado se define en 1 si la clave se incluye en el archivo csv, y se define en cero si la clave no se incluye en el archivo csv. Un archivo csv sólo con entradas de claves puede utilizarse con el comando TRANSLATE sólo para determinar si la clave forma parte de ese archivo. Para fines de seguridad, un archivo csv puede contener una lista de direcciones IP hostiles conocidas o nombres de usuario válidos con otra información de políticas, como permisos y cantidad de veces permitidas que se accede a éste.

NOTA: No se admiten claves que expresan rangos: rangos numéricos y direcciones IP.

Variables de datos

Así como se determina si una entrada de clave se incluye o no en el archivo csv, se pueden recuperar los datos relacionados con esa clave. Un número variable de variables de guión se pueden utilizar para indicar en qué variables se almacenarán los datos. Se admiten variables de cadena, números enteros y de valores flotantes. Todas las entradas de datos se almacenan como cadenas y se convertirán al tipo de variable proporcionada en el comando TRANSLATE.

Siguiendo con el ejemplo de amigos.csv:

```
Roberto,azul,25,210
Alicia,verde,19,110
Patricia,violeta,36,145
```

Puede obtener los datos relacionados con:

```
TRANSLATE(s_friend, "amigos.csv", i_found, s_color, i_age,
   i_weight)
```

Donde:

- Si s_friend contiene Alicia, i_found equivaldría a 1, s_color equivaldría a verde, i_age equivaldría a 19 e i_weight equivaldría a 110.
- Si no se encuentra la entrada de clave, las variables no se modifican (s_color, i_age, i_weight).
- Si la entrada para Alicia fuera:
  `Alicia,verde,19,`

  Al utilizar el mismo comando TRANSLATE, la variable i_weight se borraría (0 para enteros, 0.0 para valores flotantes y cadenas “”). s_color sería verde e i_age sería 19.
- Si la entrada para Alicia fuera:
  `Alicia,verde,delgada,Ford`

  Al utilizar el mismo comando TRANSLATE, la variable i_age se borraría, delgada se convertiría en un entero (0) y se colocaría en i_weight. s_color sería verde y Ford se omitiría.
- Si la entrada para Alicia fuera:
  `Alicia,25,19,110`

  Al utilizar el mismo comando TRANSLATE, la variable s_color contendría 25. i_age sería 19 e i_weight sería 110.

Formato

```
TRANSLATE(<key>, <csv_file>, <found_status>
   [, <variable>, ...])
```

Tipos de datos

Argumento	Tipo	Descripción
key		La clave que se buscará en el archivo csv.
csv_file		El nombre del archivo csv.
found_status		La variable de enteros se define en 1 si la clave se incluye en el archivo csv, y se define en cero si la clave no se incluye en el archivo csv.
variable		La lista de variables en la que se colorarán los datos relacionados con la clave.

TRIM

Quita todos los espacios en blanco de los extremos de una cadena y reemplaza múltiples espacios en blanco por espacios simples. Los espacios en blanco incluyen los siguientes caracteres:

- <tabulación>
- <retorno de carro>
- <línea nueva>
- <tabulación vertical>
- <avance de página>
- <espacio>

Formato

```
TRIM(svar)
```

Tipos de datos

Argumento	Tipo	Descripción
string	svar (ENTRADA)	Cadena de la que se quitarán los espacios en blanco. La cadena resultante se almacena en la variable de entrada.

Por ejemplo:

```
COPY(s_var:" Hola    Mundo  "
TRIM(s_var)
```

Contenido de las variables de salida actuales:

```
s_var = " Hola Mundo "
```

WHILE

El comando WHILE permite ejecutar en bucle el flujo de control.

El comando While se ejecuta de la siguiente forma:

- Si el resultado de la instrucción WHILE() es verdadero, se ejecutan los comandos de análisis posteriores al comando WHILE(), hasta el siguiente comando ENDWHILE().
- Si el resultado de WHILE() es falso, no se ejecuta ningún comando de análisis entre los comandos WHILE() y ENDWHILE().

Si bien se permiten todos los tipos de datos a cada lado del operador para la instrucción WHILE(),los valores numéricos sólo pueden compararse con valores numéricos, y las cadenas sólo pueden compararse con cadenas.

El operador de la comparación WHILE() puede ser <, =, >, <=, >=, <>, &, +, o ^.

PRECAUCIÓN: No utilice el operador lógico NOT (^) junto con una variable de cadena. Si lo hace, se genera un error de sintaxis.

No puede realizar una comparación directamente con un número negativo. Lleve a cabo uno de estos métodos:

- Utilice la función de análisis COMPARE.
- Realice una comparación indirectamente, según se muestra a continuación:

```
SET(i_compare_val=-10)
WHILE(ivar >i_compare_val)
SET(ivar=ivar-1)
ENDWHILE()
```

Formato

```
WHILE(<expr>)
```

Donde:

```
expr ::= var
       | (<expr>)
       | ^ <expr>
```

Donde <expr> debe dar como resultado enteros o valores flotantes.

```
       | <expr> <|=|>|<=|>=|<>|&|+ <expr>
```

Donde las dos <expr> deben dar como resultado el mismo tipo.

Tipos de datos

Argumento	Tipo	Descripción
data1	todos (ENTRADA)	Los datos que se compararán con data2. Si data2 no se utiliza, se convierte en un valor lógico (0 = falso, cualquier otro = verdadero).
logical operator	< = > <= >= <> & + ^	Inferior a Igual a Superior a Inferior o igual a Superior o igual a No es igual a Operador lógico AND Operador lógico OR Operador lógico NOT
data2	todos (ENTRADA) [OPCIONAL]	Los datos que se compararán con data1. Deben ser del mismo tipo que data1.
…	igual que en el caso anterior	Utilice hasta 200 parámetros individuales para crear expresiones lógicas complejas.

Por ejemplo:

```
WHILE(i<3)
SET(i=i+1)
ALERT("Todavía en el bucle")
ENDWHILE()
ALERT("Salió del bucle")
```

4 Funciones del administrador del asistente

NOTA: El término Agente puede intercambiarse con Recopilador. En adelante, los agentes se denominarán recopiladores.

Este capítulo va dirigido al administrador del sistema del asistente. Describe las diversas funciones administrativas que realiza el administrador del sistema y brinda información con respecto a los procesos en segundo plano del asistente.

NOTA: La primera vez que el Generador de recopiladores del asistente se ejecuta, aparece el siguiente mensaje: "El directorio recopiladores no existe por lo que se creará de forma automática." Puede perderse alguna información." Seleccione Aceptar. El directorio se creará y aparecerá el asistente del generador de recopiladores. Si el mensaje se mostrase una vez que el generador de recopiladores se ejecute, el directorio de recopiladores puede haber sido eliminado por error, por lo que deberá revisar si ha perdido alguna información.

Utilidades y aplicaciones del asistente

El asistente consta de una interfaz de usuario (Generador de recopiladores) y varias utilidades adicionales que funcionan con el generador para realizar la supervisión de la red.

Generador de recopiladores

La interfaz de usuario del asistente es el generador de recopiladores. El generador de recopiladores le permite configurar los recopiladores de la red así como los puertos y guiones que se utilizan para comunicarse con los hosts. El generador de recopiladores se ejecuta solamente en Windows.

NOTA: Si tiene problemas con la forma en que se muestran las ventanas del asistente después de arrastrar una ventana a otra posición, revise la configuración de la pantalla en el panel de control de Microsoft Windows. En la pestaña Efectos, desmarque la casilla Mostrar contenido de la ventana mientras se arrastra.

Puerto

En el asistente, los puertos permiten que un recopilador ubique los datos de eventos de seguridad en la red al proporcionar la dirección IP y otra información acerca del origen (dispositivo de seguridad [router, IDS, conmutador, etc…]). Cada fila de la tabla Configuración del puerto ejecuta un guión del recopilador en un origen del evento.

Gestor de recopiladores

El gestor de recopiladores inicia y detiene el procesamiento del puerto.

Motor del recopilador

El motor del recopilador procesa la lógica de la plantilla para cada puerto. Se ejecuta un motor del recopilador para cada puerto activo.

popup.exe

El motor del recopilador emplea la utilidad popup.exe para asistir en el procesamiento de comandos de análisis emergentes o de pantalla.

popup.cfg

El archivo popup.cfg es un archivo opcional que se utiliza para controlar el tiempo límite de los comandos de análisis emergentes y de pantalla. Si no posee un archivo popup.cfg, los comandos de análisis de pantalla y emergentes no tendrán tiempo límite.

Para configurar un tiempo límite para el comando de pantalla, escriba la declaración:

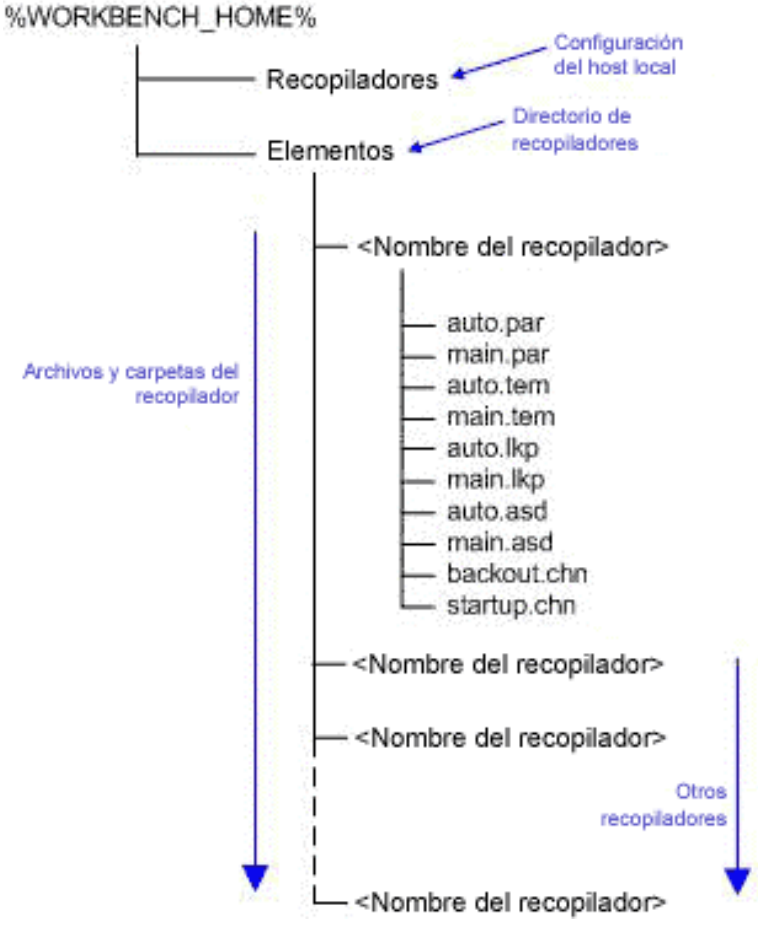
```
displaytimeout <verdadero/falso>.
```

El tiempo límite de pantalla se configura en 20 segundos.

Para configurar un tiempo límite para el comando emergente, escriba la declaración:

```
timeout <tiempo límite en segundos>.
```

Estructura de directorio del asistente



Clave

Recopiladores	Archivos de configuraciones de puertos (Hosts del asistente)
Elements	Archivos de recopiladores
.par	Archivos de parámetros
.tem	Archivos de plantillas
.lkp	Archivos de búsquedas
.asd	Archivos de descripción de estado activo
backout.chn	Archivos de guiones de restitución
startup.chn	Archivos de guiones de inicio

5 Meta-etiquetas de Sentinel y del asistente

NOTA: El término Agente puede intercambiarse con Recopilador. En adelante, los agentes se denominarán recopiladores.

NOTA: Para los usuarios de MS SQL 2000, el tamaño del evento no puede ser superior a 8KB.

Las meta-etiquetas almacenan metadatos. Los metadatos son información sobre datos, nombres de variables predefinidas para los metadatos. Por ejemplo, la dirección IP de origen de un ataque se almacena en la meta-etiqueta IP de origen. Los nombres de producto se almacenan en la meta-etiqueta Nombre de producto. Los datos que se utilizan para completar las meta-etiquetas se extrae de los datos de registro del dispositivo o se configuran como parte del procesamiento del recopilador.

Para acceder a la función de Configuración de eventos y asignación en el Gestor de datos de Sentinel, haga clic en la pestaña Eventos.

NOTA: En el lenguaje de la regla de Correlación RuleLg sin formato, cuando una etiqueta es precedida de una ‘e.’, como e.crt, se refiere a eventos actuales. Si una etiqueta es precedida de una ‘w.’, como w.crt, se refiere a eventos históricos.

El valor de la columna Variable del recopilador es el nombre de la variable del recopilador que se debe configurar para completar la meta-etiqueta correspondiente. Para obtener más información acerca de los comandos de análisis, consulte el Capítulo 3 y la documentación de los recopiladores específicos que se encuentra en

```
%ESEC_HOME%\wizard\elements\<Nombre del
   recopilador>\docs.
```

NOTA: En la tabla de abajo, se utilizan etiquetas y meta-etiquetas en el Centro de control de Sentinel. En el análisis de recopiladores se utilizan variables del recopilador. No todas las meta-etiquetas poseen una variable del recopilador que le corresponda.

Los tipos que se especifican en la columna Tipo tienen las siguientes propiedades:

- cadena: máximo de 255 (salvo que se especifique lo contrario)
- entero: entero con signos de 32 bits
- UUID: cadena hexadecimal de 36 caracteres (con guiones) o 32 caracteres (sin guiones) en el formato XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX (p. ej., 6A5349DA-7CBF-1028-9795-000BCDFFF482)
- fecha: la variable del recopilador se debe configurar con fecha como un número de milisegundos desde el 1 de enero de 1970 00:00:00 GMT. Cuando se muestran en el Centro de control de Sentinel, las meta-etiquetas de tipo fecha se mostrarán en un formato de fecha regular.
- IPv4: dirección IP en notación decimal con puntos (p. ej., xxx.xxx.xxx.xxx)

Etiqueta	Meta-etiqueta	Tipo	Descripción	Variable del recopilador
CorrelatedEventUuids	ceu	string	Lista de UUID de evento asociados con este evento correlacionado. Sólo relevante para eventos correlacionados.	
Criticality	crt	integer	La importancia del activo identificado en este evento.	s_CRIT
Ct1 a Ct2 (Cliente reservado)	ct1 a ct2	string	Reservado para uso de clientes, para datos específicos del cliente (cadena).	s_CT1 y s_CT2
Ct3 (Cliente reservado 3)	ct3	integer	Reservado para uso de clientes, para datos específicos del cliente (número).	s_CT3
CustomerVar1 a CustomerVar10	cv1 a cv10	integer	Reservado para uso de clientes, para datos específicos del cliente (número).	s_CV1 a s_CV10
CustomerVar11 a CustomerVar20	cv11 a cv20	date	Reservado para uso de clientes, para datos específicos del cliente (fecha).	s_CV11 a s_CV20
CustomerVar21 a CustomerVar29	cv21 a cv29	cadena	Reservado para uso de clientes, para datos específicos del cliente (cadena).	s_CV21 a s_CV29
CustomerVar30 a CustomerVar34	cv30 a cv34	cadena	Reservado para uso de clientes, para datos específicos del cliente (cadena). Puede manipular longitudes de cadena de hasta 4000 caracteres.	s_CV30 a s_CV34
CustomerVar35 a CustomerVar89	cv35 a cv89	string	Reservado para uso de clientes, para datos específicos del cliente (cadena).	s_CV35 a s_CV89
SARBOX	cv90	string	Datos específicos Sarbanes Oxley.	s_CV90
HIPAA	cv91	string	Datos específicos de la HIPAA (Ley de Portabilidad y Responsabilidad de Seguro México).	s_CV91

Etiqueta	Meta-etiqueta	Tipo	Descripción	Variable del recopilador
GLBA	cv92	string	Datos específicos de la Ley Gramm-Leach-Bliley (GLBA).	s_CV92
FISMA	cv93	string	Datos específicos de la FISMA (Ley federal de administración de la seguridad de la información).	s_CV93
NISPOM	cv94	string	Datos específicos de NISPOM (Manual de operaciones del programa nacional de seguridad industrial).	s_CV94
SIPCountry	cv95	string	País de IP de origen.	s_CV95
DIPCountry	cv96	string	País de IP de destino.	s_CV96
CustomerVar97 a CustomerVar100	cv97 a cv100	string	Reservado para uso de clientes, para datos específicos del cliente (cadena).	s_CV97 a s_CV100
DateTime	dt	date	La fecha y hora normalizadas del evento, según la indica el recopilador.	
DestinationHostName	dhn	string	El nombre de host de destino al que apuntó el evento.	s_DHN
DestinationIP	dip	IPv4	La dirección IP de destino a la que apuntó el evento.	s_DIP
DestinationPort	dp	string (32)	El puerto de destino al que apuntó el evento.	s_DP
DestinationUserName	dun	string	El nombre de usuario de destino en el que se intentó la acción. Ejemplo: Intentos para restaurar la contraseña de root.	s_DUN
EventID	id	UUID	Identificador único para este evento.	
EventTime	et	string	La hora normalizada del evento, según la informa el sensor; analizada en el formato: Y-M-D-H:M:S~AMPM24~TZ.	s_ET

Etiqueta	Meta-etiqueta	Tipo	Descripción	Variable del recopilador
EventName	evt	string	El nombre descriptivo del evento según lo informa (o brinda) el sensor. Ejemplo “Exploración de puerto”.	s_EVT
ExtendedInformation	ei	string (1000)	Almacena más información recopilada del recopilador. Los valores de esta variable se separan con punto y coma (;). Ejemplo: Un dominio para una ID o nombres de archivo.	s_EI
FileName	fn	string (1000)	El nombre del programa que se ejecuta o el archivo que se accede, modifica o afecta. Ejemplo: El nombre de un archivo infectado con virus o de un programa detectado por un IDS.	s_FN
Message	msg	string (4000)	Texto de mensaje sin formato para el evento.	s_BM
Protocol	prot	string	El protocolo de red del evento.	s_P
ProductName	pn	string	Indica el nombre del tipo, proveedor o código de producto del sensor a partir del cual se genera el evento. Ejemplo: Cortafuegos de Check Point=CPFW.	s_PN
ReporterName	rn	string	El nombre de host o dirección IP del dispositivo al cual ingresó un evento o desde el cual se envía la notificación del evento.	s_RN
ReservedVar1 a ReservedVar10	rv1 a rv10	integer	Reservado por Novell para expansión (Número).	s_RV1 a s_RV10
ReservedVar11 a ReservedVar20	rv11 a rv20	date	Reservado por Novell para expansión (Fecha).	s_RV11 a s_RV20

Etiqueta	Meta-etiqueta	Tipo	Descripción	Variable del recopilador
ReservedVar21 a ReservedVar25	rv21 a rv25	UUID	Reservado por Novell para expansión (UUID).	s_RV21 a s_RV25
ControlPack	rv26	string	Categorización de control de Sentinel nivel 1	s_RV26
ControlMonitor	rv27	string	Categorización de control de Sentinel nivel 2	s_RV27
ReservedVar28	rv28	string	Reservado por Novell para expansión (Cadena).	s_RV28
SourceIPCountry	rv29	string	País de dirección IP de origen.	s_RV29
AttackID	rv30	string	ID de ataque normalizado (ID de ataque del asesor)	s_RV30
DeviceName	rv31	string	Nombre del dispositivo de seguridad	s_RV31
Categoría del dispositivo	rv32	string	Categoría del dispositivo (AV, DB, ESEC, FW, IDS, OS). AV: Antivirus BD: base de datos ESEC: evento del sistema FW: cortafuegos IDS: detección de intrusiones OS: sistema operativo	s_RV32
EventContext	rv33	string	Contexto del evento (nivel de amenaza).	s_RV33
SourceThreatLevel	rv34	string	Nivel de amenaza de origen.	s_RV34
SourceUserContext	rv35	string	Contexto de usuario de origen.	s_RV35
DataContext	rv36	string	Contexto de los datos.	s_RV36
SourceFunction	rv37	string	Función de origen.	s_RV37
SourceOperationalContext	rv38	string	Contexto operativo de origen.	s_RV38
MSSPCustomerName	rv39	string	Nombre del cliente MSSP.	s_RV39
ReservedVar40 a ReservedVar43	rv40 a rv43	string	Reservado por Novell para expansión (Cadena).	s_RV40 a s_RV43

Etiqueta	Meta-etiqueta	Tipo	Descripción	Variable del recopilador
DestinationThreatLevel	rv44	string	Nivel de amenaza de destino.	s_RV44
DestinationUserContext	rv45	string	Contexto de usuario de destino.	s_RV45
VirusStatus	rv46	string	Estado de virus.	s_RV46
DestinationFunction	rv47	string	Función de destino.	s_RV47
DestinationOperationalContext	rv48	string	Contexto operativo de destino.	s_RV48
ReservedVar49	rv49	string	Reservado por Novell para expansión (Cadena).	s_RV49
eSecTaxonomyLevel1	rv50	string	Categorización de código de evento de Sentinel - nivel 1	s_RV50
eSecTaxonomyLevel2	rv51	string	Categorización de código de evento de Sentinel - nivel 2	s_RV51
eSecTaxonomyLevel3	rv52	string	Categorización de código de evento de Sentinel - nivel 3	s_RV52
eSecTaxonomyLevel4	rv53	string	Categorización de código de evento de Sentinel - nivel 4	s_RV53
ReservedVar54 a ReservedVar55	rv54 a rv55	string	Reservado por Novell para expansión (Cadena).	s_RV54 a s_RV55
SourceAssetName	rv56	string	Origen (Administración de activo Mgmt) – Nombre de activo	s_RV56
SourceMacAddress	rv57	string	Origen (Administración de activo) – Dirección MAC	s_RV57
SourceNetworkIdentity	rv58	string	Origen (Administración de activo) - Identidad de red	s_RV58
SourceAssetCategory	rv59	string	Origen (Administración de activo) – Categoría de activo	s_RV59
SourceEnvironmentIdentity	rv60	string	Origen (Administración de activo) - Identidad de entorno	s_RV60
SourceAssetValue	rv61	string	Origen (Administración de activo Mgmt) – Valor de activo	s_RV61
SourceCriticality	rv62	string	Origen (Administración de activo) – Importancia	s_RV62

Etiqueta	Meta-etiqueta	Tipo	Descripción	Variable del recopilador
SourceSensitivity	rv63	string	Origen (Administración de activo) – Sensibilidad	s_RV63
SourceBuilding	rv64	string	Origen (Administración de activo) – Edificio	s_RV64
SourceRoom	rv65	string	Origen (Administración de activo) – Sala	s_RV65
SourceRackNumber	rv66	string	Origen (Administración de activo) – Número de bastidor	s_RV66
SourceCity	rv67	string	Origen (Administración de activo) – Ciudad	s_RV67
SourceState	rv68	string	Origen (Administración de activo) – Estado	s_RV68
SourceCountry	rv69	string	Origen (Administración de activo) – País	s_RV69
SourceZipCode	rv70	string	Origen (Administración de activo) – Código postal	s_RV70
SourceAssetOwner	rv71	string	Origen (Administración de activo) – Propietario del activo	s_RV71
SourceAssetMaintainer	rv72	string	Origen (Administración de activo) - Mantenedor de activo	s_RV72
SourceBusinessUnit	rv73	string	Origen (Administración de activo) – Unidad empresarial	s_RV73
SourceLineOfBusiness	rv74	string	Origen (Administración de activo) – Líneas de negocio	s_RV74
SourceDivision	rv75	string	Origen (Administración de activo) – División	s_RV75
SourceDepartment	rv76	string	Origen (Administración de activo) – Departamento	s_RV76
SourceAssetId	rv77	string	Origen (Administración de activo) – ID de activo de origen	s_RV77
DestinationAssetName	rv78	string	Destino (Administración de activo) – Nombre de activo	s_RV78
DestinationMacAddress	rv79	string	Destino (Administración de activo) – Dirección MAC	s_RV79

Etiqueta	Meta-etiqueta	Tipo	Descripción	Variable del recopilador
DestinationEnvironmentIdentity	rv80	string	Destino (Administración de activo) - Identidad de red	s_RV80
DestinationAssetCategory	rv81	string	Destino (Administración de activo) – Categoría de activo	s_RV81
DestinationEnvironmentIdentity	rv82	string	Destino (Administración de activo) - Identidad de entorno	s_RV82
DestinationAssetValue	rv83	string	Destino (Administración de activo) – Valor de activo	s_RV83
DestinationCriticality	rv84	string	Destino (Administración de activo) – Importancia	s_RV84
DestinationSensitivity	rv85	string	Destino (Administración de activo) – Sensibilidad	s_RV85
DestinationBuilding	rv86	string	Destino (Administración de activo) – Edificio	s_RV86
DestinationRoom	rv87	string	Destino (Administración de activo) – Sala	s_RV87
DestinationRackNumber	rv88	string	Destino (Administración de activo) – Número de bastidor	s_RV88
DestinationCity	rv89	string	Destino (Administración de activo) – Ciudad	s_RV89
DestinationState	rv90	string	Destino (Administración de activo) – Estado	s_RV90
DestinationCountry	rv91	string	Destino (Administración de activo) – País	s_RV91
DestinationZipCode	rv92	string	Destino (Administración de activo) – Código postal	s_RV92
DestinationAssetOwner	rv93	string	Destino (Administración de activo) – Propietario del activo	s_RV93
DestinationAssetMaintainer	rv94	string	Destino (Administración de activo) - Mantenedor de activo	s_RV94
DestinationBusinessUnit	rv95	string	Destino (Administración de activo) – Unidad empresarial	s_RV95
DestinationLineOfBusiness	rv96	string	Destino (Administración de activo) – Líneas de negocio	s_RV96
DestinationDivision	rv97	string	Destino (Administración de activo) – División	s_RV97

Etiqueta	Meta-etiqueta	Tipo	Descripción	Variable del recopilador
DestinationDepartment	rv98	string	Destino (Administración de activo) – Departamento	s_RV98
DestinationAssetId	rv99	string	Destino (Administración de activo) - ID de activo de destino	s_RV99
ReservedVar100	rv100	string	Reservado por Novell para expansión (Cadena).	s_RV100
Resource	res	string	El nombre del recurso.	s_Res
DeviceAttackName	rt1	string	Para uso con el asesor. Nombre del ataque desde el dispositivo de seguridad	s_RT1
Rt2	rt2	string	Se completa con el nombre de la regla de correlación cuando una regla de correlación genera un evento.	s_RT2
Rt3	rt3	entero	Reservado por Novell para expansión (Número).	s_RT3
SourceHostName	shn	string	El nombre del host de origen desde el que se origina el evento.	s_SHN
SourceID	src	UUID	Identificador único del proceso de Sentinel que generó este evento.	
SourceIP	sip	IPv4	La dirección IP de origen desde la que se origina el evento	s_SIP
SensorName	sn	string	El nombre del “detector definitivo” del evento cuando se reciben datos sin procesar. Por ejemplo, “FW1” para un cortafuegos.	s_SN
Severity	sev	entero	La gravedad normalizada del evento (0-5).	i_Severity
SourcePort	sp	string (32)	El puerto de origen desde el que se origina el evento.	s_SP

Etiqueta	Meta-etiqueta	Tipo	Descripción	Variable del recopilador
SensorType	st	string (5)	El designador de carácter simple para el tipo de sensor single (N, H, I, O, P, V, C, W). C: Correlación H: de host I: internos (evento del sistema) N: de red O: Otros P: rendimiento (evento del sistema) V: Antivirus W: Lista de vigilancia	s_ST
SourceUserName	sun	string	El nombre de usuario de origen utilizado para iniciar un evento. Por ejemplo, “jdoe” durante un intento de “su”.	s_SUN
SubResource	sres	string	El nombre del subrecurso.	s_SubRes
Vulnerability	vul	entero	La vulnerabilidad del activo identificado en este evento.	s_VULN
WizardAgent	agent	string (64)	Recopilador de Sentinel que generó este evento. Para los eventos del sistema, el recopilador será Rendimiento o Interno.	
WizardPort	port	string (64)	Descripción del puerto del recopilador de Sentinel.	

6 Permisos de usuario del Centro de control de Sentinel

NOTA: El término Agente puede intercambiarse con Recopilador. En adelante, los agentes se denominarán recopiladores.

Los permisos de usuario se dividen de la manera siguiente:

- General
  - Filtros públicos
  - Filtros privados
  - Acciones de integración
- Active View
  - Elementos de menú
  - Pantallas de resumen
- iTRAC
  - Gestión de plantillas
  - Gestión de procesos
- Incidencias
- Gestión de recopiladores
- Análisis
- Asesor
- Administración
  - Correlación
  - Estadísticas DAS
  - Información del archivo de eventos
  - Vistas del servidor
  - Filtros globales
  - Gestión de funciones iTRAC
  - Configuración de menús
  - Gestión de usuarios
  - Gestión de sesiones de usuario

Usuarios por defecto

El programa de instalación creará los usuarios por defecto siguientes en el servidor de Sentinel:

Autenticación de Oracle y MS SQL: usuarios por defecto: Consulte usuario por defecto

- esecdba: propietario del esquema (se puede configurar durante la instalación).
- esecadm: usuario Administrador de Sentinel (se puede configurar durante la instalación).

NOTA: Para UNIX, el programa de instalación también crea el usuario del sistema operativo con el mismo nombre de usuario y contraseña.

- esecrpt: usuario que genera informes, contraseña como usuario admin.
- ESEC_CORR: usuarios del motor de correlación, se utilizan para crear incidencias.
- esecapp: nombre de usuario de la aplicación Sentinel para la conexión a la base de datos.

Autenticación de Windows:

- Administrador de la base de datos de Sentinel: propietario del esquema (se puede configurar durante la instalación).
- Administrador de Sentinel: usuario administrador de Sentinel (se puede configurar durante la instalación).

- Usuario de generación de informes de Sentinel: usuario de generación de informes, contraseña como el usuario admin.
- Usuario de la base de datos de aplicaciones Sentinel: nombre de usuario de la aplicación Sentinel para la conexión a la base de datos.

General

Nombre del permiso	Descripción
Guardar área de trabajo	Permite al usuario guardar las preferencias. Si este permiso no está disponible, nunca se le solicitará al usuario que guarde los cambios de las preferencias cuando cierre la sesión o salga del Centro de control de Sentinel.
Gestión de columnas	Permite al usuario administrar las columnas en las tablas de Active View.
Instantánea	Permite al usuario crear una instantánea de las tablas de Active View.

General: Filtros públicos

Nombre del permiso	Descripción
Crear Filtros públicos	Permite al usuario crear un filtro con un ID de propietario PUBLIC. Si el usuario no tiene este permiso, el valor PUBLIC no aparecerá como uno de los ID de propietario para el cual el usuario puede crear un filtro.
Modificar Filtros públicos	Permite al usuario modificar un filtro público.
Eliminar Filtros públicos	Permite al usuario eliminar un filtro público.

General: Filtros privados

Nombre del permiso	Descripción
Crear Filtros privados	Permite al usuario crear filtro privados para sí mismo o para otros usuarios.
Modificar Filtros privados	Permite al usuario modificar sus propios filtros privados y aquéllos creados por otros usuarios.
Eliminar Filtros privados	Permite al usuario eliminar sus propios filtros privados y aquéllos creados por otros usuarios.
Ver/usar Filtros privados	Permite al usuario ver sus propios filtros privados y aquéllos creados por otros usuarios.

General: Acciones de integración

Nombre del permiso	Descripción
Enviar a HP Open View	Permite al usuario enviar eventos, incidencias y objetos asociados a HP-OVO.
Enviar eventos a HP Service Desk	Permite al usuario enviar eventos, incidencias y objetos asociados a HP Service Desk.
Enviar a Remedy Help Desk	Permite al usuario enviar eventos, incidencias y objetos asociados a Remedy.

Active Views

Nombre del permiso	Descripción
Ver pestaña Active View	Permite al usuario ver y utilizar la pestaña Active View, el menú y otras funciones relacionadas que estén asociadas con la pestaña Active View.

Active Views: Elementos de menú

Nombre del permiso	Descripción
Usar elementos de menú asignados	Permite al usuario utilizar elementos de menú asignados en la tabla Eventos de Active View (el menú emergente que aparece al hacer clic con el botón derecho del ratón).
Añadir a incidencia existente	Permite al usuario añadir eventos a incidencias existentes mediante la tabla Eventos de Active View (el menú emergente que aparece al hacer clic con el botón derecho del ratón).
Eliminar de la incidencia	Permite al usuario eliminar eventos de una incidencia existente mediante la tabla Eventos de Active View (el menú emergente que aparece al hacer clic con el botón derecho del ratón).
Enviar eventos por correo electrónico	Permite al usuario enviar eventos por correo electrónico mediante la tabla Eventos de Active View (el menú emergente que aparece al hacer clic con el botón derecho del ratón).
Ver datos de ataque del asesor	Permite al usuario ver el flujo de datos de ataque del asesor.
Ver vulnerabilidades	Permite al usuario ver la salida de una exploración Nessus.

Active Views: Pantallas de resumen

Nombre del permiso	Descripción
Usar/Ver pantallas de resumen	Permite al usuario acceder a los diagramas de Active View.

iTRAC

Nombre del permiso	Descripción
Ver pestaña iTRAC	Permite al usuario ver y utilizar la pestaña iTRAC, el menú y otras funciones relacionadas que estén asociadas con la pestaña iTRAC.
Gestión de actividades	Permite al usuario acceder al Gestor de actividades.

Gestión de plantillas

Nombre del permiso	Descripción
Ver/Usar el Gestor de plantillas	Permite al usuario acceder al Gestor de plantillas.
Crear/Modificar plantillas	Permite al usuario crear y modificar plantillas.

Gestión de procesos

Nombre del permiso	Descripción
Ver/Usar el Gestor de procesos	Permite al usuario acceder al Gestor de vistas del proceso.
Procesos del control	Permite al usuario utilizar el Gestor de vistas del proceso.

Incidencias

Nombre del permiso	Descripción
Ver pestaña Ver incidencias	Permite al usuario ver y utilizar la pestaña Ver incidencias, el menú y otras funciones relacionadas que estén asociadas con esta pestaña.
Administración de incidencias	Permite al usuario modificar una incidencia.
Ver incidencias	Permite al usuario ver los detalles de una incidencia. Si el usuario no tiene este permiso, la ventana Información de incidencias no se mostrará cuando el usuario haga doble clic en una incidencia en la ventana del navegador o en una incidencia en la pestaña respectiva de un caso.
Crear incidencias	Permite al usuario crear incidencias en el menú Eventos al que se accede haciendo clic con el botón derecho del ratón en un evento.
Modificar incidencias	Permite al usuario modificar una incidencia en la ventana Información de incidencias.
Eliminar incidencias	Permite al usuario eliminar incidencias.
Asignar incidencias	Permite al usuario asignar una incidencia en la ventana de modificación y creación de incidencias.
Enviar incidencia por correo electrónico	Permite al usuario enviar por correo electrónico las incidencias de interés.
Acciones de la incidencia	Permite al usuario habilitar o inhabilitar la configuración o ejecución de la acción de la incidencia.

Gestión de recopiladores

Nombre del permiso	Descripción
Ver recopiladores	▪ Permite ver la pestaña “Recopiladores” en el Centro de control de Sentinel ▪ Permite ver la pestaña 'Hosts del asistente' en el generador de recopiladores
Controlar recopiladores	▪ Incluye todas las capacidades como el permiso de “Ver recopiladores” ▪ Permitido para el comando y el control de recopiladores desde el Centro de control de Sentinel ▪ Permite el comando y el control de recopiladores desde el generador de recopiladores del asistente

Nombre del permiso	Descripción
Administración de recopiladores	▪ Incluye todas las capacidades como el permiso de 'Recopiladores del comando' ▪ En el generador de recopiladores, el recopilador edita y desarrolla ▪ En el generador de recopiladores, crea, edita, compila y depura los recopiladores. ▪ En el generador de recopiladores, carga y descarga los recopiladores. ▪ En el generador de los recopiladores, se exporta el hosts del asistente ▪ En el generador de recopiladores, se añaden, editan y eliminan los puertos ▪ En el generador de los recopiladores, se configuran las opciones del puerto

El comando y el control consisten de:

- encender/apagar los puertos individuales
- encender/apagar todos los puertos
- restaurar los hosts
- renombrar los hosts

Análisis

Nombre del permiso	Descripción
Ver pestaña Ver análisis	Permite al usuario ver y utilizar la pestaña Ver análisis, el menú y otras funciones relacionadas que estén asociadas con la pestaña Ver descripción general del sistema.

Asesor

Nombre del permiso	Descripción
Ver pestaña Ver asesor	Permite al usuario ver y utilizar la pestaña Ver asesor, el menú y otras funciones relacionadas que estén asociadas con esta pestaña.

Administración

Nombre del permiso	Descripción
Ver pestaña Ver administración	Permite al usuario ver y utilizar la pestaña Ver administración, el menú y otras funciones relacionadas que estén asociadas con esta pestaña.

Administración: Correlación

Nombre del permiso	Descripción
Usar/Ver el gestor de motores de correlación	Permite al usuario ver y utilizar el motor de correlación.
Usar/Ver reglas de correlación	Permite al usuario iniciar o detener las reglas de correlación.

Administración: Filtros globales

Nombre del permiso	Descripción
Ver/usar Filtros globales	Permite al usuario acceder a la ventana de configuración de filtros globales.
Modificar Filtros globales	Permite al usuario modificar la configuración de filtros globales. **NOTA:** Para acceder a esta función, también se debe asignar el permiso Ver filtros globales.

Administración: Configuración de menú

Nombre del permiso	Descripción
Configuración del menú	Permite al usuario acceder a la ventana de configuración del menú y añadir opciones nuevas que se muestran en el menú Eventos al hacer clic con el botón derecho del ratón en un evento.

Administración: Estadísticas DAS

Nombre del permiso	Descripción
Estadísticas de DAS	Permite al usuario ver actividades de DAS (DAS binario y de consulta).

Administración: Información del archivo de eventos

Nombre del permiso	Descripción
Información del archivo de eventos	Permite al usuario ver el estado de archivos de eventos.

Administración: Vistas del servidor

Nombre del permiso	Descripción
Ver servidores	Permite al usuario supervisar el estado de todos los procesos.
Control de servidores	Permite al usuario iniciar, restaurar y detener procesos.

Administración: Gestión de usuarios

Nombre del permiso	Descripción
Usar/Ver una cuenta de usuario	Permite al usuario ver y utilizar cuentas de usuario.
Crear una cuenta de usuario	Permite al usuario crear una cuenta de usuario. **NOTA:** Para acceder a esta función, también se debe asignar el permiso Ver/Usar cuentas de usuario.

Nombre del permiso	Descripción
Modificar una cuenta de usuario existente	Permite al usuario modificar una cuenta de usuario existente. **NOTA:** Para acceder a esta función, también se debe asignar el permiso Ver/Usar cuentas de usuario.
Eliminar una cuenta de usuario	Permite al usuario eliminar una cuenta de usuario existente. **NOTA:** Para acceder a esta función, también se debe asignar el permiso Ver/Usar cuentas de usuario.

Administración: Gestión de sesiones de usuario

Nombre del permiso	Descripción
Gestión de sesiones de usuario	Permite al usuario ver, bloquear y anular a usuarios activos (entradas en el Centro de control de Sentinel).

Administración: Gestión de funciones iTRAC

Nombre del permiso	Descripción
Gestión de funciones iTRAC	Permite ver y utilizar el Gestor de funciones en la pestaña Admin.

7 Motor de correlación de Sentinel

NOTA: El término Agente puede intercambiarse con Recopilador. En adelante, los agentes se denominarán recopiladores.

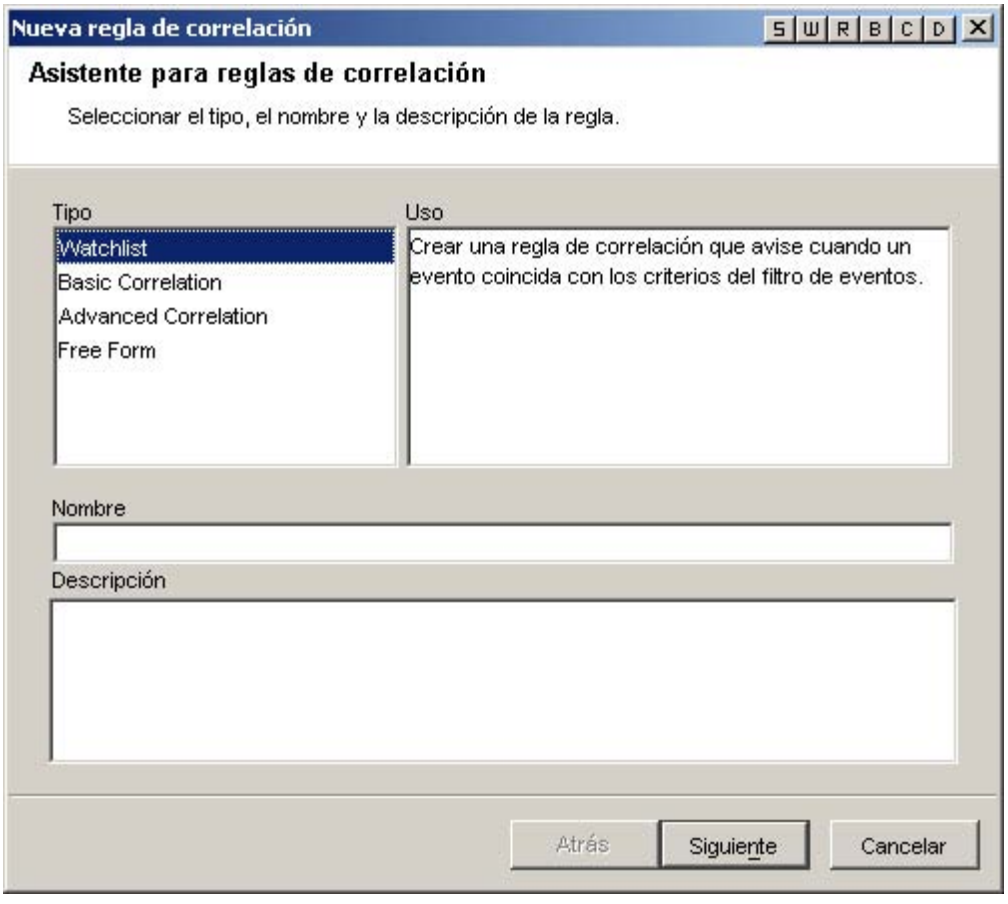
El Motor de correlación de Sentinel es una aplicación residente en memoria de múltiples hilos de ejecución. Los múltiples subprocesos permiten que el motor de correlación aproveche las ventajas de equipos multiprocesadores, como máquinas SMP (multiprocesamiento simétrico).

El motor de correlación ha sido diseñado para recibir datos desde dispositivos de seguridad, dispositivos de red y otras fuentes de aplicaciones y para buscar patrones significativos, por lo general, dentro de un determinado plazo de tiempo. Estos patrones podrían indicar ataques, intrusiones, uso abusivo o incumplimiento. Cuando se genera un evento correlacionado, el campo rt2 se completará con el nombre de la regla de correlación.

El Motor de correlación de Sentinel ofrece una implementación que permite la expansión. Esta arquitectura permite la implementación en una red distribuida de motores de correlación que funcionan en conjunto para correlacionarse en tiempo real con los datos relevantes de seguridad, incluso eventos de seguridad supervisados en tiempo real, resultados de exploraciones de vulnerabilidades en sistemas potencialmente apuntados e información de activos que indiquen la importancia relativa de esos sistemas con respecto a los procesos críticos del negocio y su asociación con otros sistemas de la organización.

El Motor de correlación de Sentinel funciona en base a reglas. Usted dirige el procesamiento del motor de correlación a través de reglas que crea en el editor respectivo del Centro de control de Sentinel. El editor de reglas se basa en un conjunto de asistentes que ofrecen varias opciones para la creación de reglas. Los asistentes para reglas son:

- Lista de vigilancia
- Correlación básica
- Correlación avanzada
- RuleLg sin formato



Tipos de filtros de correlación

Para Lista de vigilancia, Correlación básica y Correlación avanzada, puede elegir entre cuatro tipos de filtros diferentes: Son los siguientes:

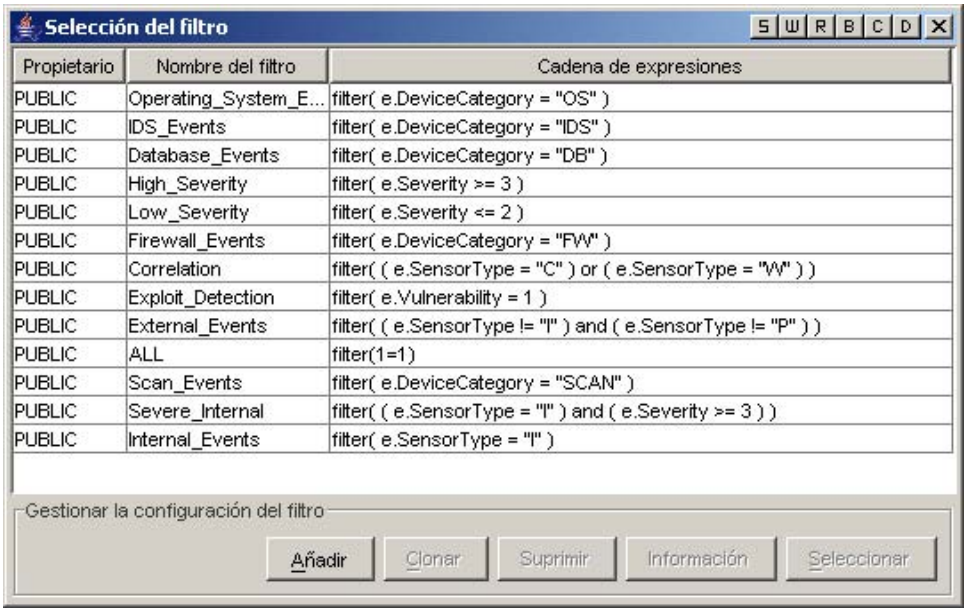
- Mostrar todo: equivale a ejecutar una gravedad de filtro superior o igual a cero.
- Patrón: cualquier expresión regular con una sintaxis tipo grep. Una regla puede buscar una dirección IP de origen específica de un pirata informático y notificarle cada vez que esa dirección IP aparece en un mensaje de evento.
- Gestor de filtros: lista desplegable que muestra el Gestor de filtros para seleccionar o crear un filtro nuevo.
- Generador: permite crear criterios para la inclusión y exclusión de eventos basados en álgebra booleana.

Filtro de correlación tipo Patrón

Un filtro de correlación tipo Patrón utiliza cualquier expresión regular con una sintaxis tipo grep. La coincidencia con una expresión regular se realiza con una concatenación de todas las meta-etiquetas presentes en cada evento entrante. Por ejemplo, virusXYZ observará la presencia de la cadena virusXYZ en todas las meta-etiquetas en cada evento entrante.

Filtro de correlación tipo Gestor de filtros

Esta opción le permite seleccionar un filtro existente o crear un filtro para usar en la correlación mediante la ventana Gestor de filtros.



Filtro de correlación tipo Generador

El filtro de correlación tipo Generador presenta dos partes. Una parte comprende los criterios de inclusión (qué eventos deben incluirse en la coincidencia de patrón) y la otra parte comprende la exclusión (qué eventos deben excluirse de la coincidencia de patrón).

- Qué eventos deben incluirse en la coincidencia de patrón - Utilice esta tabla para especificar las condiciones para limitar qué eventos activarán la correlación.
  - Etiqueta de evento: la columna Etiqueta de evento es una lista desplegable que detalla las etiquetas de eventos (también llamadas meta-etiquetas) que se pueden correlacionar.
  - Condición: la columna Condición es una lista desplegable que detalla los operadores utilizados en la generación de una condición de correlación.
  - Valor: la columna Valor es un campo sin formato que puede utilizar para especificar valores si se eligen las condiciones =, !=, <, >, <= o >=. Si se selecciona =Meta-etiqueta o !=Meta-etiqueta en la columna Condición, la columna Valor presentará una lista desplegable de las meta-etiquetas disponibles para elegir. Puede escribir lo que desee con las siguientes restricciones: y pueden aparecer en cualquier parte de la cadena si se utiliza regex.
  - y/o: cambie entre uno y otro haciendo clic en uno de estos cuadros. Cuando se especifican varias condiciones en la tabla, los botones 'y' y 'o' le permiten especificar si se deben cumplir todas las condiciones o si se debe cumplir solo una. Elija “y” para indicar que se deben cumplir todas las condiciones. Elija “o” para indicar que se debe cumplir sólo una de las condiciones.

  NOTA: La selección es válida sólo si la tabla contiene una segunda fila o más. Todas las filas de la tabla tomarán este operador lógico como predeterminado excepto la última. No se admiten combinaciones de 'y' y 'o' entre filas dentro de la tabla.

  - Botones +/- El botón + añadirá una fila adicional al final de la tabla. El botón - eliminará la fila seleccionada de la tabla independientemente de su posición.
- Qué eventos deben excluirse de la coincidencia de patrón - Utilice esta tabla para especificar las condiciones para limitar qué eventos no activarán la regla de correlación.
  - Etiqueta de evento: una lista de las etiquetas de evento disponibles que se pueden correlacionar.
  - Condición: la columna Condición es una lista desplegable que detalla los operadores utilizados en la elaboración de una condición de correlación.
  - Valor: la columna Valor es un campo sin formato que puede utilizar para especificar valores si se eligen las condiciones =, !=, <, >, <= o >=. Si se selecciona =Meta-etiqueta o !=Meta-etiqueta en la columna Condición, la columna Valor presentará una lista desplegable de las meta-etiquetas disponibles para elegir. Puede escribir lo que desee con las siguientes restricciones: Y pueden aparecer en cualquier parte de la cadena si se utiliza regex.
  - y/o: cambie entre uno y otro haciendo clic en uno de estos cuadros. Cuando se especifican varias condiciones en la tabla, los botones 'y' y 'o' le permiten especificar si se deben cumplir todas las condiciones o si se debe cumplir solo una. Elija “y” para indicar que se deben cumplir todas las condiciones. Elija “o” para indicar que se debe cumplir sólo una de las condiciones.

  NOTA: La selección es válida sólo si la tabla contiene una segunda fila o más. Todas las filas de la tabla tomarán este operador lógico como predeterminado excepto la última. No se admiten combinaciones de 'y' y 'o' entre filas dentro de la tabla.

  - Botones +/-: El botón + añadirá una fila adicional al final de la tabla. El botón - eliminará la fila seleccionada de la tabla independientemente de su posición.

Definición de regla de correlación

Asistentes para reglas de correlación: Vigilante, Correlación básica y Correlación avanzada le permiten añadir rápidamente un tipo de regla predefinido, según lo que desee lograr. El asistente para cada tipo de regla maneja la generación de la regla de correlación en el lenguaje nativo de la regla del motor de correlación. Cada una de estas reglas se crea mediante la ventana Reglas de correlación en la pestaña Admin.

El Asistente para reglas comprende un editor sin formato que le permite usar el lenguaje de definición de correlación RuleLg para añadir la regla directamente en el lenguaje nativo de la regla del motor de correlación.

Lista de vigilancia

Se puede elegir entre cuatro tipos diferentes de filtros. Son los siguientes:

- Mostrar todo: equivale a ejecutar una gravedad de filtro superior a cero o igual a cero.
- Patrón: cualquier expresión regular con una sintaxis tipo grep.
- Gestor de filtros: lista desplegable que muestra el Gestor de filtros para seleccionar o crear un filtro nuevo.
- Generador: permite crear criterios para la inclusión y exclusión de eventos basados en álgebra booleana.

Para obtener más información, consulte Creación de una regla de lista de vigilancia.

Correlación básica

Se puede elegir entre cuatro tipos diferentes de filtros. Son los siguientes:

- Mostrar todo: equivale a ejecutar una gravedad de filtro superior a cero o igual a cero.
- Patrón: cualquier expresión regular con una sintaxis tipo grep.
- Gestor de filtros: lista desplegable que muestra el Gestor de filtros para seleccionar o crear un filtro nuevo.
- Generador: permite crear criterios para la inclusión y exclusión de eventos basados en el álgebra booleana.

Esta regla permite contar el número de veces que se satisfacen determinadas condiciones en un plazo de tiempo específico.

Por ejemplo, una regla de correlación básica puede buscar la misma dirección IP de origen informada cinco veces en cinco minutos, incluso si los eventos se informan desde productos distintos como, por ejemplo, un sistema de detección de intrusiones (IDS) y un cortafuegos.

Para obtener más información, consulte Creación de una regla de correlación básica.

Correlación avanzada

Se puede elegir entre cuatro tipos diferentes de filtros. Son los siguientes:

- Mostrar todo: equivale a ejecutar una gravedad de filtro superior a cero o igual a cero.
- Patrón: cualquier expresión regular con una sintaxis tipo grep.
- Gestor de filtros: lista desplegable que muestra el Gestor de filtros para seleccionar o crear un filtro nuevo.
- Generador: permite crear criterios para la inclusión y exclusión de eventos basados en el álgebra booleana.

Esta regla le permite:

- Contar el número de veces que se satisfacen determinadas condiciones en un plazo de tiempo específico.
- Incorporar todas las funciones de la regla de correlación básica, así como evaluar eventos en comparación con eventos anteriores.

Por ejemplo, una regla de correlación avanzada puede buscar eventos desde la misma dirección IP de origen a la misma dirección IP de destino con el mismo nombre de evento y que se producen tanto dentro como fuera de un cortafuegos (lo que significa que puede que un ataque haya atravesado el cortafuegos).

Para obtener más información, consulte Creación de una regla de correlación avanzada.

Correlación RuleLg de regla sin formato

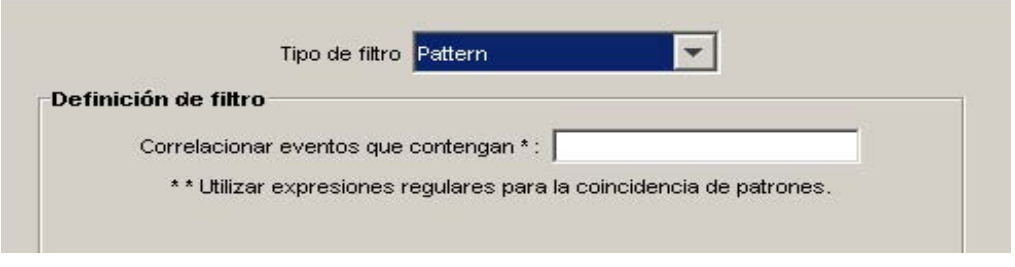
El lenguaje de definición de reglas de correlación RuleLg permite un control completo de la definición de reglas de correlación. Antes de utilizar este tipo de regla de correlación, deberá familiarizarse con el lenguaje de definición de reglas de correlación RuleLg.

Para obtener más información, consulte Creación de una regla de correlación RuleLg sin formato.

Creación de una regla de lista de vigilancia

Cree una regla de lista de vigilancia cuando desee especificar una cadena que el Motor de correlación observará en cada evento entrante. Para crear una regla de lista de vigilancia:

- Seleccione la regla de lista de vigilancia en la ventana Asistente para reglas de correlación. Complete la información relativa a:
  - Nombre de regla: nombre que aparecerá en la lista de reglas. La cantidad máxima de caracteres es 255 y no se permiten puntos. No se permiten los caracteres ASCII extendidos. El Nombre de regla distingue entre mayúscula y minúscula.
  - Descripción: breve descripción. La longitud máxima del texto descriptivo es de 1024 caracteres.
- Tipo de filtro
  - Mostrar todo -
  - Patrón – Observa eventos que contengan *

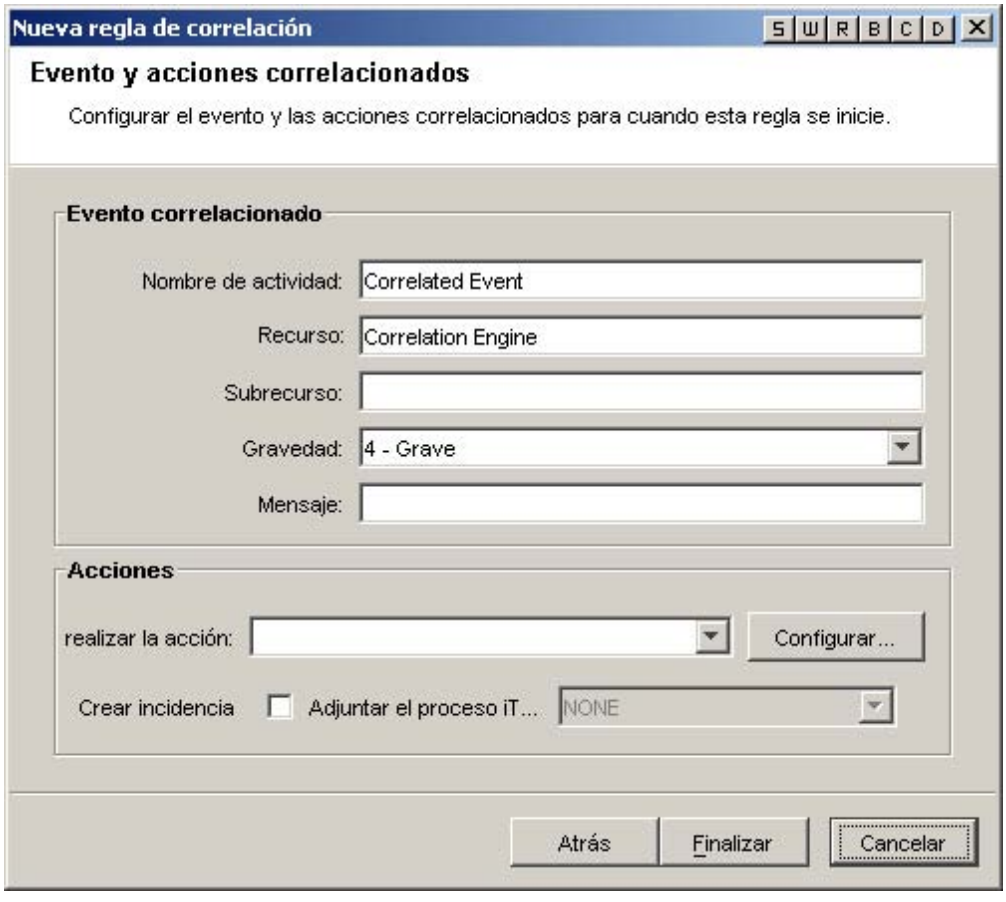


- Gestor de filtro: ({ownerid}:{Filter name}:<Nombre de filtro>

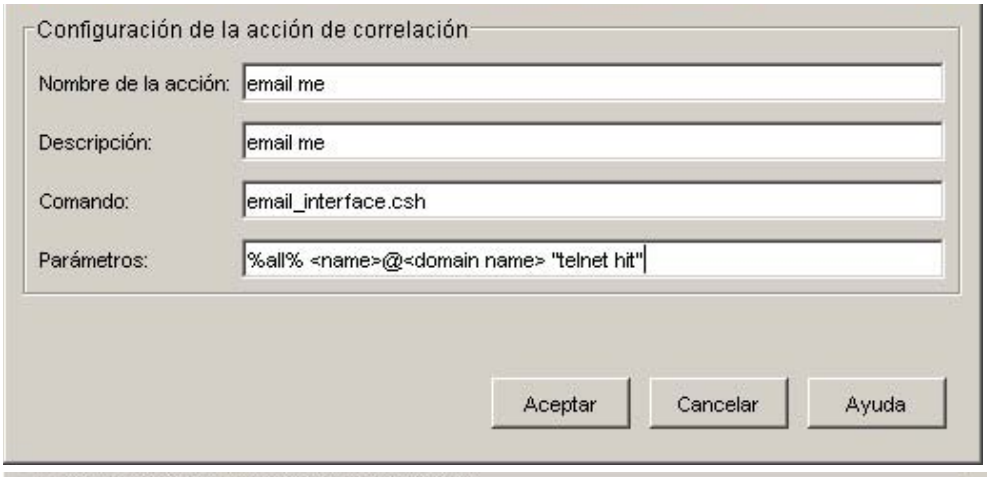
- Generador

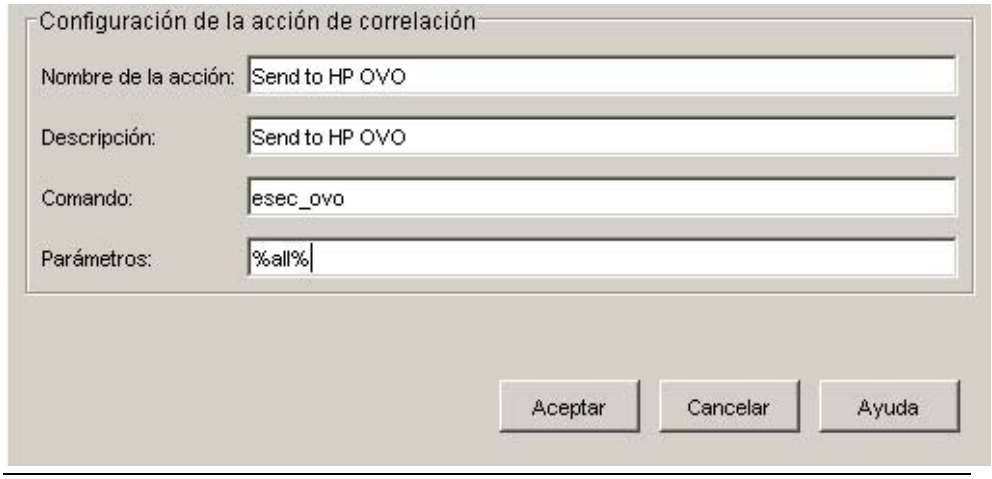
- Página Evento y acciones correlacionadas: en este panel se define qué acción se realizará automáticamente cuando los eventos coincidan con esta regla de correlación. La única entrada obligatoria corresponde al nivel de gravedad cuyo valor predeterminado es 4.
  - Nombre de evento - por defecto: Evento correlacionado. Éste es el nombre de texto del evento correlacionado.
  - Recurso por defecto: Motor de correlación. Éste es el nombre de texto de un recurso en el sistema.
  - Subrecurso - por defecto: <ninguno>. Éste es el nombre correspondiente a los recursos con varios subrecursos.
  - Configurar el nivel de gravedad en - por defecto: 4, éste es el nivel de gravedad que se asignará al evento. Los valores válidos son 0, 1, 2, 3, 4 (por defecto) y 5. Se ofrece una lista desplegable con los niveles de gravedad válidos.
  - Texto del mensaje personalizado - por defecto: <ninguno>. Éste es el texto que aparecerá junto con el evento. Es práctico para identificar la condición que activó la regla de lista de vigilancia. La cantidad máxima de caracteres es de 4 000. El texto que escriba en este cuadro precede al texto del evento de correlación con un separador de conducto. Por ejemplo, la entrada de “Mensaje nuevo” resultaría en un mensaje correlacionado de “Mensaje nuevo|Tres instancias de....”.

- Realizar la acción (Oracle solamente) - por defecto: <ninguno>. Éste es el nombre de un archivo ejecutable que se ejecuta cuando se activa la regla de lista de vigilancia. El archivo debe encontrarse en el directorio $ESEC_HOME/sentinel/exec y ser ejecutable por el usuario esecadm. No se valida la entrada en este tipo de cuadro de texto sin formato. Puede especificar las meta-etiquetas que desee enviar al ejecutable.
- Realizar la acción (MSSQL solamente) - por defecto: <ninguno>. Éste es el nombre de un archivo ejecutable que se ejecuta cuando se activa la regla. El archivo debe encontrarse en el directorio %ESEC_HOME%\sentinel\bin y ser ejecutable por el usuario esecadm. No se valida la entrada. Puede especificar las meta-etiquetas que desee enviar al ejecutable. A continuación se ilustran dos ejemplos de una regla de correlación que envía un mensaje por correo electrónico y una regla de correlación que envía un evento de correlación a HP OVO.



La línea de comando y la línea de parámetro se alimentan como una cadena. En el análisis aplican las mismas reglas que \ (barra diagonal inversa) es un carácter de escape. Los caracteres \, % y " se pueden utilizar para escape. Por ejemplo, \%\"\\ es equivalente a %"\. Si necesita un comando que contenga una barra diagonal inversa, es decir, ejecutar un comando de Windows en un subdirectorio de sentinel\bin, introduzca dos barraas diagonales inversas (\\) para cada barra de directorio. Por ejemplo, para ejecutar un archivo por lotes llamado run.bat en %esec_home%\sentinel\bin\batchfiles\, debería introducir batchfiles\\run.bat. Recuerde que todos los ejecutables deben aparecer debajo de %esec_home%\sentinel\bin\.





NOTA: Para obtener más información sobre Comandos y parámetros, consulte el capítulo 5 – Meta-etiquetas de Sentinel y el asistente en la Guía de referencia del usuario y Sección Salida de correlación

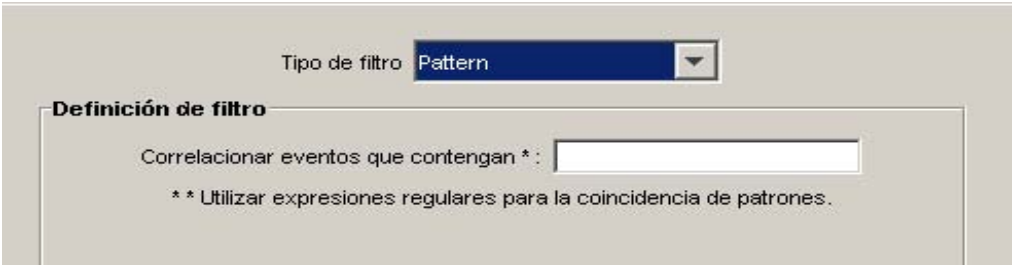
- Crear incidencia: una acción del evento correlacionado también puede ser la creación de una incidencia.
- Adjuntar el proceso iTrac: la incidencia creada puede tener un proceso iTrac adjunto.

Creación de una regla de correlación básica

Cree una regla de correlación básica cuando desee contar el número de veces que se satisfacen determinadas condiciones en un plazo de tiempo. Los pasos en cuestión son:

- Seleccione la regla de correlación básica en la primera ventana Asistente para reglas de correlación. Complete la información relativa a:
  - Nombre de regla: nombre que aparecerá en la lista de reglas. La cantidad máxima de caracteres es 255 y no se permiten puntos. No se permiten los caracteres ASCII extendidos. El Nombre de regla distingue entre mayúscula y minúscula.
  - Descripción: breve descripción. La longitud máxima del texto descriptivo es de 1024 caracteres.

- Tipo de filtro
  - Mostrar todo
  - Patrón



  - Gestor de filtro: ({ownerid}:{Filter name}:<Nombre de filtro>

  - Generador

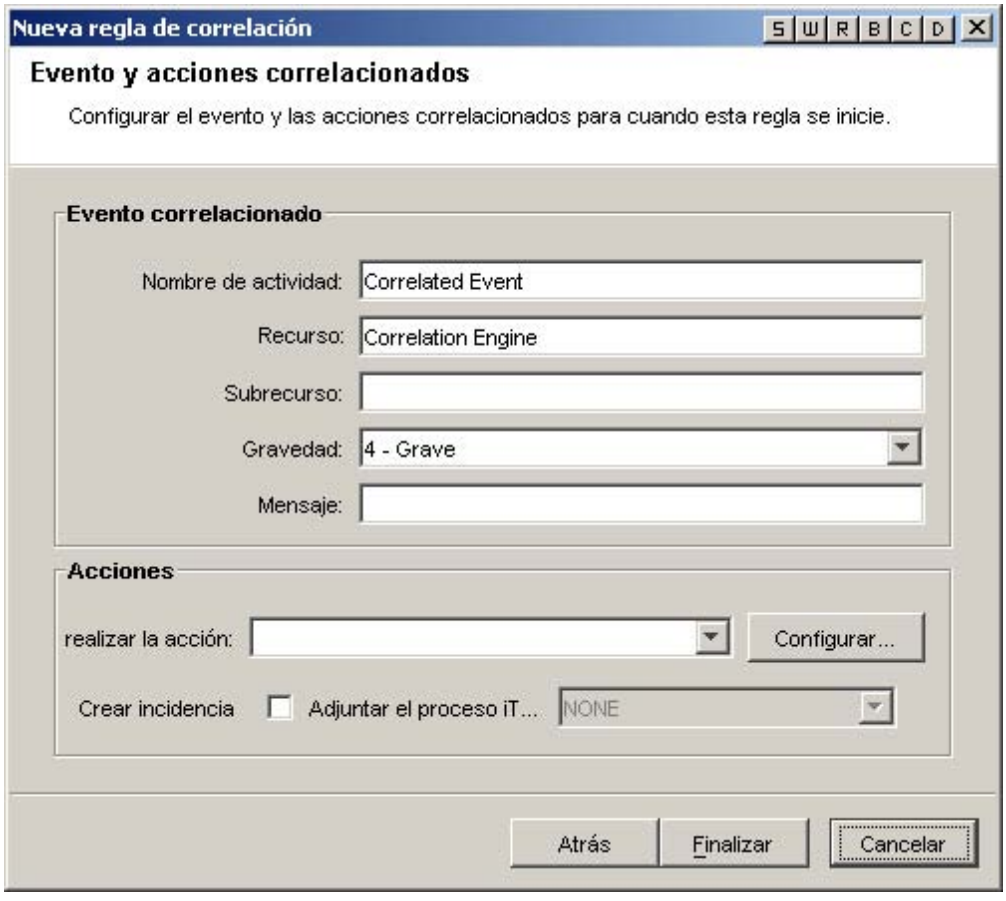
- Criterios de grupo y umbral (mitad superior de la ventana) - Activar regla: esta opción le permite introducir criterios “de coincidencia” para varios eventos que ingresan en el sistema durante un período de tiempo dado.
  - Cuando se cumple la condición _veces - por defecto: 1. Se activa una regla sólo después de que se ha detectado el número de veces especificado. El rango válido de entradas para este valor de umbral es 1 o mayor.
  - dentro (del plazo de tiempo) - por defecto: 60 segundos. Esto enlazará la condición con el plazo de tiempo. Es una combinación de entrada de variable y lista desplegable. Las opciones de la lista desplegable son: segundo, minutos, horas y días.

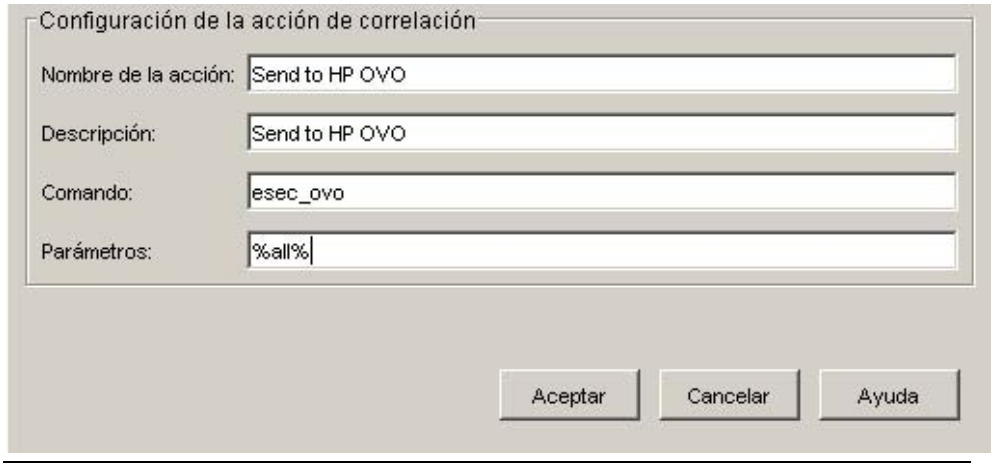
NOTA: Cuando el plazo de tiempo es 0, la activación se considera instantánea. En el caso de correlación básica, el evento sucederá como máximo una vez durante un plazo de tiempo instantáneo.

- Página Criterios de grupo y umbral (mitad inferior de la ventana) - Se correlaciona con distintas combinaciones de las meta-etiquetas siguientes - Seleccione las meta-etiquetas para usar combinadas con la correlación. Los eventos se colocan en grupos en función de las meta-etiquetas seleccionadas.

- Página Evento y acciones correlacionadas: en este panel se define qué acción se realizará automáticamente cuando los eventos coincidan con esta regla de correlación. La única entrada obligatoria corresponde al nivel de gravedad cuyo valor predeterminado es 4.
  - Nombre de evento - por defecto: Evento correlacionado. Éste es el nombre de texto del evento correlacionado.
  - Recurso por defecto: Motor de correlación. Éste es el nombre de texto de un recurso en el sistema.
  - Subrecurso - por defecto: <ninguno>. Éste es el nombre correspondiente a los recursos con varios subrecursos
  - Configurar el nivel de gravedad en - por defecto: 4. Éste es el nivel de gravedad que se asignará al evento. Los valores válidos son 0, 1, 2, 3, 4 (por defecto) y 5. Se ofrece una lista desplegable con los niveles de gravedad válidos.

- Texto del mensaje personalizado - por defecto: <ninguno>. Éste es el texto que aparecerá junto con el evento. Es práctico para identificar la condición que activó la regla de lista de vigilancia. La cantidad máxima de caracteres es de 4 000. El texto que escriba en este cuadro precede al texto del evento de correlación con un separador de conducto. Por ejemplo, la entrada de “Mensaje nuevo” resultaría en un mensaje correlacionado de “Mensaje nuevo|Tres instancias de....”.
- Ejecute este comando (Oracle solamente) - por defecto: <ninguno>. Éste es el nombre de un archivo ejecutable que se ejecuta cuando se activa la regla de lista de vigilancia. El archivo debe encontrarse en el directorio $ESEC_HOME/sentinel/exec y ser ejecutable por el usuario esecadm. No se valida la entrada en este tipo de cuadro de texto sin formato. Puede especificar las meta-etiquetas que desee enviar al ejecutable.
- Realizar la acción (MSSQL solamente) - por defecto: <ninguno>. Éste es el nombre de un archivo ejecutable que se ejecuta cuando se activa la regla. El archivo debe encontrarse en el directorio %ESEC_HOME%\sentinel\bin y ser ejecutable por el usuario esecadm. No se valida la entrada. Puede especificar las meta-etiquetas que desee enviar al ejecutable. A continuación se ilustran dos ejemplos de una regla de correlación que envía un mensaje por correo electrónico y una regla de correlación que envía un evento de correlación a HP OVO.





NOTA: Para obtener más información sobre Comandos y parámetros, consulte el capítulo 5 – Meta-etiquetas de Sentinel y el asistente en la Guía de referencia del usuario y Sección Salida de correlación.

- Crear incidencia: una acción del evento correlacionado también puede ser la creación de una incidencia.
- Adjuntar el proceso iTrac: la incidencia creada puede tener un proceso iTrac adjunto

Creación de una regla de correlación avanzada

Una regla de correlación avanzada le permite añadir mayor complejidad a la regla mediante la incorporación de una condición adicional en la ventana Criterios adicionales; en esencia, añadir un nivel de adición lógica (ANDing) a la definición de la regla.

Cree una regla de correlación avanzada cuando desee no sólo contar el número de veces que se satisfacen determinadas condiciones, sino también desee recibir una alerta cuando los eventos también satisfacen criterios que comprenden a eventos pasados. Los pasos en cuestión son:

- Seleccione la regla de correlación avanzada en la primera ventana Asistente para reglas de correlación. Complete la información relativa a:
  - Nombre de regla: nombre que aparecerá en la lista de reglas. La cantidad máxima de caracteres es 255 y no se permiten puntos. No se permiten los caracteres ASCII extendidos. El Nombre de regla distingue entre mayúscula y minúscula.

- Descripción: breve descripción. La longitud máxima del texto descriptivo es de 1024 caracteres.

- Tipo de filtro
  - Mostrar todo
  - Patrón

  - Gestor de filtro: ({ownerid}:{Filter name}:<Nombre de filtro>

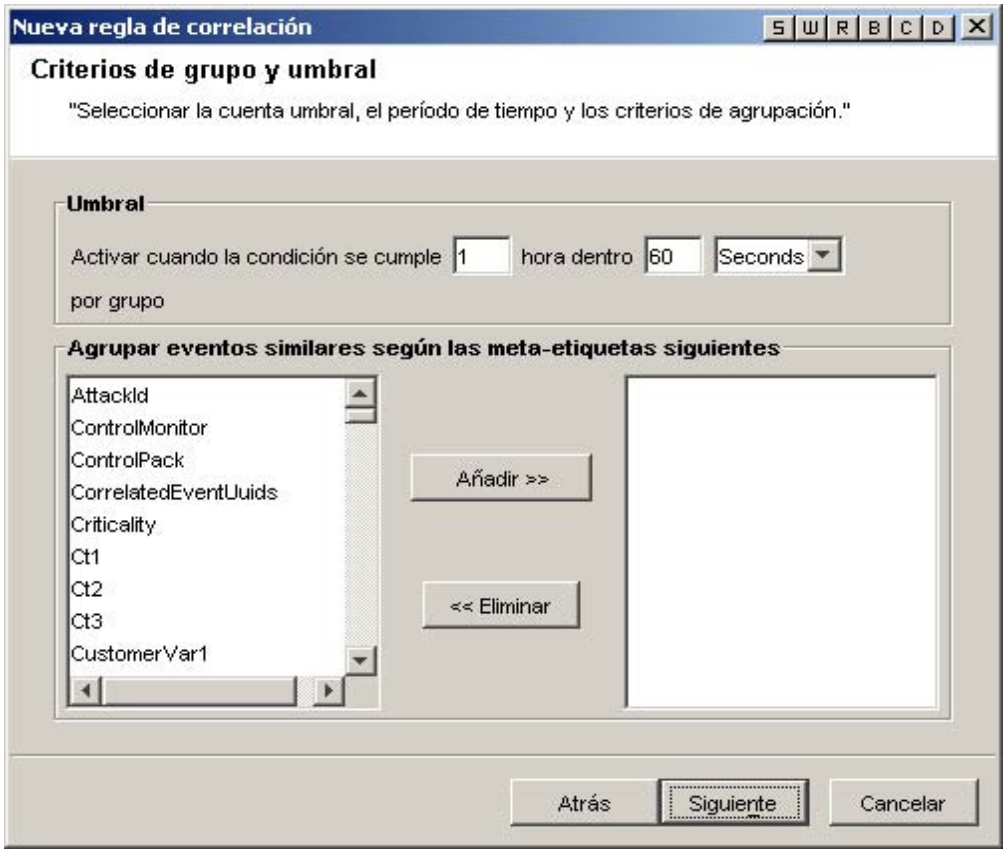
  - Generador

- Criterios adicionales: esta opción le permite introducir criterios “de coincidencia” para varios eventos que ingresan en el sistema durante un período de tiempo dado. El tiempo por defecto es de 60 segundos. Es una combinación de entrada de variable y lista desplegable. Las opciones de la lista desplegable son: segundo, minutos, horas y días.

- Criterios de grupo y umbral (mitad superior de la ventana) - Activar regla: esta opción le permite introducir criterios “de coincidencia” para varios eventos que ingresan en el sistema durante un período de tiempo dado.
  - Cuando se cumple la condición _veces - por defecto: 1. Se activa una regla sólo después de que se ha detectado el número de veces especificado. El rango válido de entradas para este valor de umbral es 1 o mayor.
  - dentro (del plazo de tiempo) - por defecto: 60 segundos. Esto enlazará la condición con el plazo de tiempo. Es una combinación de entrada de variable y lista desplegable. Las opciones de la lista desplegable son: segundo, minutos, horas y días.

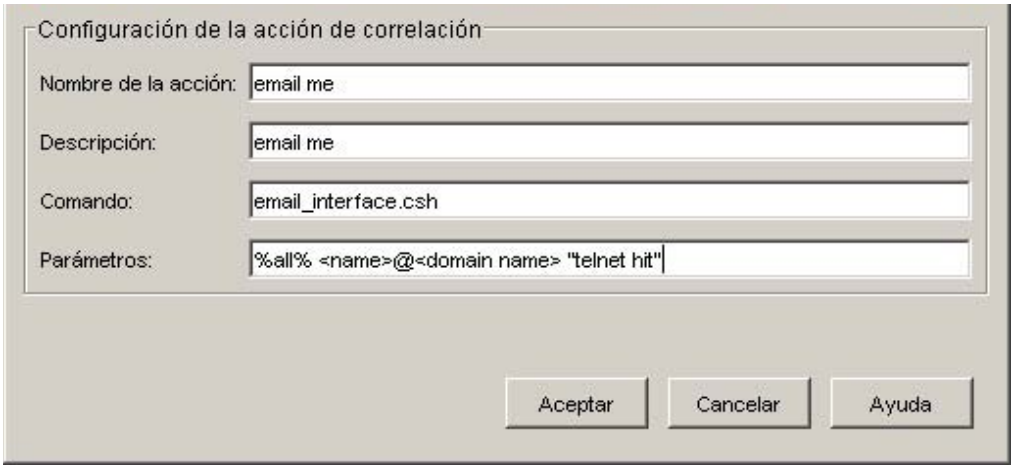
NOTA: Cuando el plazo de tiempo es 0, la activación se considera instantánea. En el caso de correlación básica, el evento sucederá como máximo una vez durante un plazo de tiempo instantáneo.

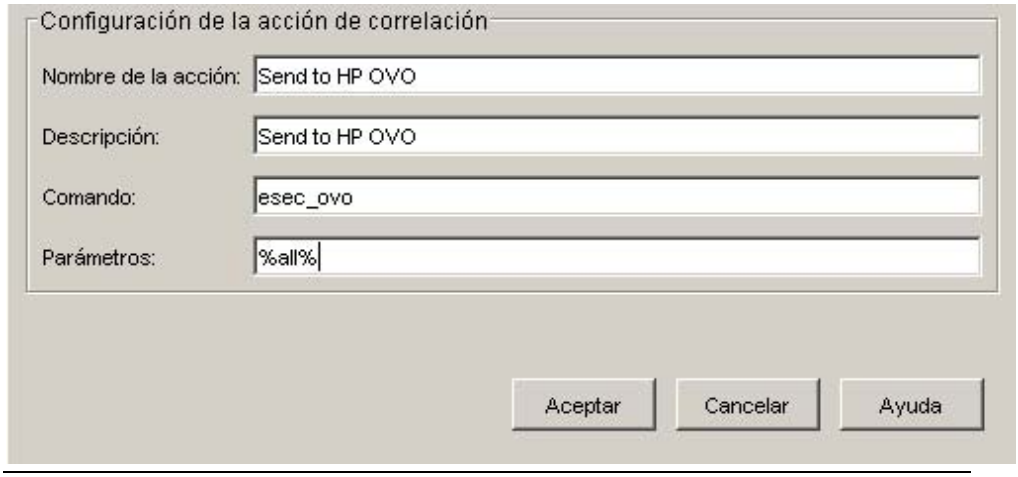
- Página Criterios de grupo y umbral (mitad inferior de la ventana) - Se correlaciona con distintas combinaciones de las meta-etiquetas siguientes - Seleccione las meta-etiquetas para usar combinadas con la correlación. Los eventos se colocan en grupos en función de las meta-etiquetas seleccionadas.



- Página Evento y acciones correlacionadas: en este panel se define qué acción se realizará automáticamente cuando los eventos coincidan con esta regla de correlación. La única entrada obligatoria corresponde al nivel de gravedad cuyo valor predeterminado es 4.
  - Nombre de evento - por defecto: Evento correlacionado. Éste es el nombre de texto del evento correlacionado.
  - Recurso por defecto: Motor de correlación. Éste es el nombre de texto de un recurso en el sistema.

- Subrecurso - por defecto: <ninguno>. Éste es el nombre correspondiente a los recursos con varios subrecursos
- Configurar el nivel de gravedad en - por defecto: 4. Éste es el nivel de gravedad que se asignará al evento. Los valores válidos son 0, 1, 2, 3, 4 (por defecto) y 5. Se ofrece una lista desplegable con los niveles de gravedad válidos.
- Texto del mensaje personalizado - por defecto: <ninguno>. Éste es el texto que aparecerá junto con el evento. Es práctico para identificar la condición que activó la regla de lista de vigilancia. La cantidad máxima de caracteres es de 4 000. El texto que escriba en este cuadro precede al texto del evento de correlación con un separador de conducto. Por ejemplo, la entrada de "Mensaje nuevo" resultaría en un mensaje correlacionado de "Mensaje nuevo|Tres instancias de....".
- Ejecute este comando (Oracle solamente) - por defecto: <ninguno>. Éste es el nombre de un archivo ejecutable que se ejecuta cuando se activa la regla de lista de vigilancia. El archivo debe encontrarse en el directorio $ESEC_HOME/sentinel/exec y ser ejecutable por el usuario esecadm. No se valida la entrada en este tipo de cuadro de texto sin formato. Puede especificar las meta-etiquetas que desee enviar al ejecutable.
- Realizar la acción (MSSQL solamente) - por defecto: <ninguno>. Éste es el nombre de un archivo ejecutable que se ejecuta cuando se activa la regla. El archivo debe encontrarse en el directorio %ESEC_HOME%\sentinel\bin y ser ejecutable por el usuario esecadm. No se valida la entrada. Puede especificar las meta-etiquetas que desee enviar al ejecutable. A continuación se ilustran dos ejemplos de una regla de correlación que envía un mensaje por correo electrónico y una regla de correlación que envía un evento de correlación a HP OVO.





NOTA: Para obtener más información sobre Comandos y parámetros, consulte el capítulo 5 – Meta-etiquetas de Sentinel y el asistente en la Guía de referencia del usuario y en la Sección Salida de correlación.

- Crear incidencia: una acción del evento correlacionado también puede ser la creación de una incidencia.
- Adjuntar el proceso iTrac: la incidencia creada puede tener un proceso iTrac adjunto.

Creación de una regla de correlación RuleLg sin formato

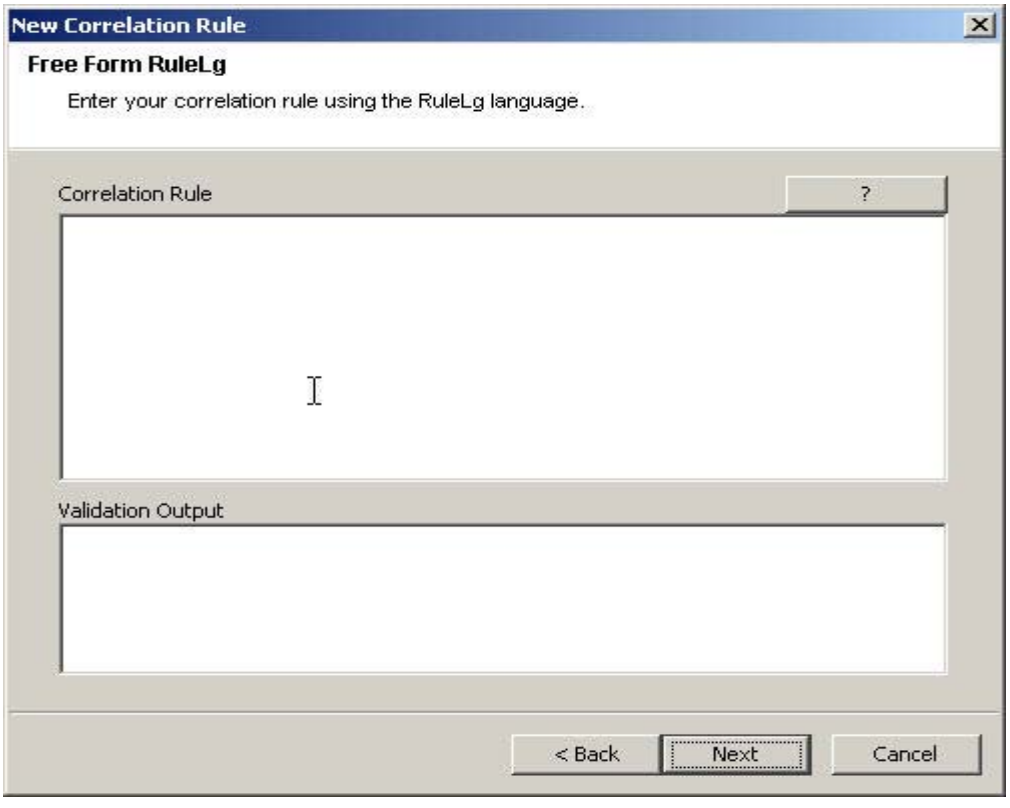
El Motor de correlación ha sido desarrollado en función de tres operaciones fundamentales. Estas operaciones se combinan para formar una regla con operadores de flujo, unión e inserción. Las tres operaciones fundamentales son:

- Operación de filtro
- Operación de ventana
- Operación de activador

> **PRECAUCIÓN:** Si ha cambiado el nombre de una etiqueta, no utilice el nombre original al crear la regla de correlación.

El lenguaje de la regla refleja directamente estas operaciones y cómo se pueden combinar de modo intuitivo para definir reglas de correlación. Cada operación ha sido específicamente diseñada e implementada para ofrecer un óptimo rendimiento y funcionar en un conjunto de eventos: recibir como entrada un conjunto de eventos y devolver un nuevo conjunto de eventos. El evento actual procesado por una regla normalmente tiene un significado especial para la semántica del lenguaje. El evento actual siempre forma parte del conjunto de eventos que se encuentran dentro y fuera de una operación, a menos que el conjunto esté vacío. Si el conjunto de entrada de una operación está vacío, esta operación no se evalúa.

Desde un punto de vista simplificado, una regla de correlación procesa en serie los eventos que ingresan en el motor de correlación, es decir, uno por uno. En realidad, el Motor de correlación es capaz de procesar múltiples eventos y evaluar varias reglas con respecto a un evento simultáneamente.



Operación de filtro

Una operación de filtro (expresión booleana) permite el filtrado en función del contenido del evento actual; es decir, sus valores de meta-etiqueta y la expresión booleana especificada en el filtro. La salida de un filtro es el conjunto vacío (si el evento actual no concuerda con el filtro) o un conjunto que contiene el evento actual y todos los demás eventos del conjunto entrante.

- Los filtros funcionan sobre el evento actual y evalúan la expresión booleana del evento actual:
  - La operación de filtro devuelve el conjunto de entrada si la expresión booleana lo evalúa como verdadero.
  - La operación de filtro devuelve el conjunto vacío si la expresión booleana lo evalúa como falso.
- La expresión booleana es una composición de instrucciones de comparación y de instrucciones de concordancia con los operadores booleanos 'and', 'or' y 'not':

Operación de filtro - Asociaciones y prioridad de los operadores RuleLg

Prioridad de los operadores booleanos de filtrado (desde del más alto [arriba] al más bajo [abajo]).

Operador	Significado	Tipo de operador	Asociabilidad
not	“not” lógico	unario	ninguna
and	“and” lógico	binario	de izquierda a derecha
or	“or” lógico	binario	de izquierda a derecha

Se aplica lo siguiente:

- Las instrucciones de comparación permiten la evaluación de valores de meta-etiqueta con otros valores de meta-etiqueta de eventos o restricciones.
- Los operadores de comparación disponibles son =, !=, >, <, >=, <=:
- Las instrucciones de concordancia disponibles son expresiones regulares de concordancia, match regex() o subredes de concordancia, match subnet().
- Las instrucciones de comparación y concordancia pueden anidarse con paréntesis en tantos niveles como se desee.
- Los nombres de meta-etiquetas en las instrucciones de comparación y concordancia siempre deben ir precedidos por “e.”. para especificar el evento actual.
- Si un filtro es la última operación o la única operación de una regla de correlación, el conjunto de salida del filtro se utiliza para construir un evento correlacionado (siendo los eventos correlacionados el conjunto de salida de los eventos de las operaciones de filtro con el evento actual en primer lugar).
- Si un filtro no es la última operación de una regla de correlación (es decir, existe un operador de flujo a su derecha), el conjunto de salida de un filtro se utiliza como el conjunto de entrada de otras operaciones (a través del operador de flujo).

Por ejemplo: si el evento actual posee una gravedad de 4 y la meta-etiqueta de recursos contiene “FW” o “Comm”, se envía un evento correlacionado con el evento actual (evento único) enumerado como el evento correlacionado.

```
filter(e.sev = 4 and (e.res match regex ("FW") or e.res
   match regex ("Comm")))
```

Otro ejemplo es si alguna de las meta-etiquetas del evento actual contiene “ABC”, se envía un evento correlacionado con el evento actual (evento único) enumerado como el evento correlacionado.

```
filter(e.all match regex("ABC"))
```

Operación de ventana

Una operación de ventana (expresión booleana simple[, expresión de filtro], duración del int) actúa sobre el evento actual con relación a una ventana de eventos pasados. El propio funcionamiento de la ventana mantiene los eventos pasados. La salida de una ventana puede ser el conjunto vacío (si el evento actual no coincidió con la expresión booleana simple) o un conjunto que contenga el evento actual y todos los eventos pasados para los que la expresión booleana simple es verdadera.

La expresión booleana simple puede ser una instrucción de comparación o una instrucción de coincidencia única de un valor de meta-etiqueta de un evento pasado con un valor de meta-etiqueta de un evento actual o una constante. Para expresiones booleanas:

- El nombre de una meta-etiqueta debe ir precedido por “e.” para especificar el evento actual, o por “w.”, para especificar los eventos pasados
- Los operadores de comparación disponibles son =, !=, , , =, =, “in” y “not in”.
- Las instrucciones de concordancia disponibles son expresiones regulares de concordancia, match regex() o subredes de concordancia, match subnet()
- Debe aparecer una “w.[meta-etiqueta]” en una expresión booleana simple de ventana
- Si se evalúa un evento pasado como verdadero con el evento actual de la expresión booleana simple, el conjunto de salida es el evento entrante más todas las concordancias de la ventana
- Si ningún evento de la ventana coincide con el evento actual de la expresión booleana simple, se produce un conjunto vacío

Los eventos pasados se mantienen durante la operación de ventana.

El parámetro opcional de la expresión de filtro de una ventana le permite controlar qué eventos mantiene la ventana. Esta expresión es cualquier filtro válido.

- Cada evento que entra en el Motor de correlación que pase este filtro se sitúa en la ventana de eventos pasados.
- Si no existe una expresión de filtro, la ventana mantiene todos los eventos que entran en el Motor de correlación.
- El evento actual no se coloca en la ventana hasta que no haya finalizado la evaluación de la ventana de evento actual.
- En realidad, la ventana sólo mantiene las partes relevantes de los eventos pasados (para disminuir el uso de memoria)

Si una ventana es la última operación o la única operación de la regla de correlación, el conjunto de salida de la ventana se utiliza para construir un evento correlacionado (siendo los eventos correlacionados el conjunto de salida de los eventos de las operaciones de ventana con el evento actual en primer lugar).

Ejemplo 1

```
window(e.sip = w.sip, filter(e.sip match subnet
   (<xxx.xxx.x.x/yy>)), 60)
```

En el ejemplo anterior, el evento actual tiene una dirección IP de origen en la dirección xxx.xxx.x.x/yy especificada con máscara de subred CIDR y coincide con uno o varios eventos que tuvieron lugar en los últimos 60 segundos, se envía un evento correlacionado con el evento actual y con cualquier evento de concordancia pasado como los eventos correlacionados (el evento actual en primer lugar).

Ejemplo 2

```
window(e.sip = w.dip, 3600) intersection
window(e.dp = w.dp, 3600) intersection
window(e.evt = w.evt, 3600)
```

El anterior es un tipo de regla dominó. Un atacante explota un sistema vulnerable y lo utiliza como plataforma de ataque.

Ejemplo 3

```
filter(e.sev > 3) flow (window(e.sip = w.sip, filter
   (e.sev >3), 5) intersection window(e.evt = w.evt,
   filter(e.sev >3), 5) intersection window(e.dip =
   w.dip, filter(e.sev >3), 5) intersection window(e.sn!
   = w.sn, filter(e.sev > 3),5)
```

El ejemplo anterior es un tipo de regla dentro/fuera. Puede verse una firma de ataque en dos sistemas de detección de intrusión: uno en el lado interno de un cortafuegos y el otro en el lado externo; y el ataque tiene una gravedad superior a 3.

Operación de activador

La función principal de una operación de activador es contar el número total de eventos para una duración específica. Si se alcanza el total especificado dentro de la duración dada, devuelve un conjunto de eventos que contiene todos los eventos que el activador mantiene; si no es así, devuelve un conjunto vacío.

- La operación de activador recibe como entrada un conjunto de eventos para que se devuelva como parte del conjunto de salida de los eventos si el total, la duración y los discriminadores especificados de los conjuntos de entrada anteriores y el conjunto de entrada actual cumplen con los criterios definidos por la operación de activador.
- El total es un valor entero que especifica el número de eventos que deben tener lugar dentro de la ventana de duración para que devuelva un conjunto no vacío.
- La duración es un valor entero en segundos que especifica el tiempo durante el cual la operación de activador mantiene los eventos.
- Si la duración es igual a cero, la operación de activador sólo compara el número de eventos del conjunto de entrada con el total y emite el evento actual si el número es superior o igual al total.
- Cuando se recibe un nuevo conjunto de eventos de entrada, en primer lugar, un activador descarta los eventos obsoletos (eventos que se han mantenido después de la duración) y, a continuación, inserta el eventos actual. Si el número de eventos resultante es superior o igual al total especificado, el activador devuelve un conjunto que contiene todos los eventos.

- Si un activador es la última operación (o la única) de una regla de correlación, el conjunto de salida del activador se utiliza para construir un evento correlacionado (siendo los eventos correlacionados el conjunto de salida de los eventos de la operación activador con el evento actual en primer lugar).
- Si un activador no es la última operación de una regla de correlación (es decir, existe un operador de flujo a su derecha), el conjunto de salida de un activador se utiliza como el conjunto de entrada de otras operaciones (a través del operador de flujo).
- Tras la primera vez que se cumplen los criterios de la operación de activador (y, por tanto, la operación de activador devuelve un conjunto de eventos), si se vuelven a cumplir los criterios y contienen como mínimo uno de los eventos previamente devueltos y el activador es la última operación (o la única), el Motor de correlación no construye un nuevo evento correlacionado, sino que construye una actualización del evento correlacionado anterior.
- El discriminador (lista de meta-etiquetas) es una lista de meta-etiquetas delimitada por comas. Una operación de activador mantiene totales o contadores distintos para cada una de las combinaciones de las meta-etiquetas del discriminador.

Por ejemplo, si 5 eventos con la misma dirección IP de origen tienen lugar en un periodo de tiempo de 10 segundos, envía un evento correlacionado junto con los cinco eventos como eventos correlacionados (en primer lugar, el evento actual).

```
trigger(5,10,discriminator(e.sip))
```

Si bien el uso de la opción de regla sin formato le permite crear expresiones de complejidad ilimitada, estas reglas podrían carecer de sentido. La forma normal admitida de una expresión RuleLg consta de tres partes: la sección de filtro, la sección de ventana y la sección de activador. Las tres secciones se conectan con un operador de flujo.

La sección de filtro puede contener varios filtros conectados.

Ejemplo:

```
(filter(e.sev = 5) union filter(e.sev =4))
(filter(e.sev = 5 or e.sev =4))
```

NOTA: Esta sección es opcional. Si se omite, equivale a filter(1=1).

La sección de ventana puede contener varias ventanas que se intersecan.

Ejemplo:

(window(w.sev = e.sev,10) intersection window(w.sip = e.sip,10))

NOTA: Esta sección es opcional.

La sección de activador puede contener una operación de activador.

Ejemplo

```
(trigger(5,10))
```

NOTA: Esta sección es opcional. Si se omite, la regla se comporta como si finalizase con trigger(1,0).

Operadores que se combinan con operaciones para formar reglas

Los operadores que se combinan con operaciones para formar reglas son:

- Operador de flujo
- Operador de unión
- Operador de intersección

La prioridad de los operadores de operación de filtro, ventana y de activador (desde del más alto (arriba) al más bajo (abajo)) es:

Operador	Significado	Tipo de operador	Asociabilidad
flow	El conjunto de salida se convierte en el conjunto de entrada	binario	de izquierda a derecha
intersection	establecimiento de intersección (eliminación de duplicados)	binario	de izquierda a derecha
union	establecimiento de unión (eliminación de duplicados)	binario	de izquierda a derecha

Operador de flujo

El conjunto de eventos de salida de la operación del lado izquierdo es el conjunto de eventos de entrada para la operación del lado derecho.

Por ejemplo:

```
filter(e.sev = 5) flow trigger(3, 60)
```

La salida de la operación de filtro corresponde a la entrada de la operación de activador. El activador sólo cuenta eventos con gravedad equivalente a 5.

Operador de unión

Unión del conjunto de salida de las operaciones del lado izquierdo y del conjunto de salida de las operaciones del lado derecho. El conjunto de salida obtenido contiene eventos tanto del conjunto de salida de las operaciones del lado izquierdo como del conjunto de salida de las operaciones del lado derecho sin duplicados.

Por ejemplo:

```
filter(e.sev = 5) union filter(e.sip = 192.168.0.1)
```

es equivalente a

```
filter(e.sev = 5 or e.sip = 192.168.0.1)
```

Operador de intersección

Intersección del conjunto de salida de las operaciones del lado izquierdo y del conjunto de salida de las operaciones del lado derecho. El conjunto de salida obtenido contiene eventos que son comunes tanto en el conjunto de salida de las operaciones del lado izquierdo como en el conjunto de salida de las operaciones del lado derecho sin duplicados.

Por ejemplo:

```
filter(e.sev = 5) intersection filter(e.sip =
   192.17.16.32)
```

es equivalente a

```
filter(e.sev = 5 and e.sip = 192.17.16.32)
```

Reglas de correlación ilustrativas

En este documento se brinda un conjunto de reglas de correlación ilustrativas en función de reglas, junto con los requisitos previos (requerimientos) necesarios para que las reglas entren en efecto. Es posible que las reglas varíen en función de la configuración del sistema.

Las etiquetas de e.rv50 a e.rv53 que aparecen en los ejemplos de RuleLg corresponden a asignaciones configuradas en los archivos de asignación del recopilador. Por ejemplo, si abre el archivo windows_v2000_mapv*.csv o snort_v20_mapv*.csv, la:

- Columna Cultura corresponde a e.rv50
- Columna Comunidad corresponde a e.rv51
- Columna Familia corresponde a e.rv52
- Columna Evento corresponde a e.rv53

Por ejemplo:

```
filter (e.rv52 = "Worm") flow trigger (3, 300)
```

Esta regla se refiere a la taxonomía de NIDS. Si busca en la columna Familia en el archivo de asignación snort, encontrará más de cuarenta instancias del término Worm. Esta regla activará más de cuarenta ataques de gusanos distintos si se producen tres veces durante un período de cinco minutos.

Se detallan las reglas de correlación siguientes a modo de ejemplo de tipos de ataques.

- Brute Force – same source and target
- Buffer Overflow - same source to same target
- Desbordamiento de buffer e interrupción del servicio
- Denegación del servicio
- Login Failures - any source to any destination
- Login Failures - same source to same destination
- Microsoft - entrada anónima
- Microsoft - general windows authentication
- Microsoft - IE
- Microsoft – IIS
- Microsoft - Autenticación LAN Manager
- Microsoft – MDAC
- Microsoft - acceso remoto al registro
- Microsoft– SQL Server
- Microsoft - NETBIOS
- Microsoft - secuencia de comandos de Windows
- Múltiples Backdoor (puerta de atrás) – distintos orígenes
- Múltiples Backdoor (puerta de atrás) – único origen
- Caballo troyano
- UNIX - Apache Web server
- UNIX - BIND/DNS
- FTP de UNIX
- UNIX - autenticación general
- UNIX - line printer daemon
- UNIX - remote procedure call
- UNIX - remote services
- UNIX - secure shell
- UNIX - sendmail
- UNIX – SNMP
- Propagación de virus
- Propagación de un gusano

Ataque por desbordamiento de buffer e interrupción del servicio

Esta regla identificará una potencial violación de la seguridad tras un ataque por desbordamiento de buffer. Esta regla emitirá un alerta si el destino de un ataque por desbordamiento de buffer ha interrumpido un servicio a los 60 segundos del ataque. Un Recopilador de host, HIDS/OS, puede detectar si se interrumpe un servicio. El ataque por desbordamiento de buffer puede detectarse mediante un Recopilador OS, NIDS o HIDS.

Si un sistema se viese afectado por un ataque por desbordamiento de buffer, este evento debería investigarse.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información) ▪ Plataformas IDS de host que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía HIDS y OS para obtener más información)	NIDS HIDS/OS

RuleLg para esta regla

```
filter ((e.rv51 = "Service" and e.rv52 = "Stop" ) and
   (e.st = "H")) flow window (w.dip = e.sip, filter
   (e.rv52 = "Buffer_Overflow"), 60) flow trigger(1, 0)
```

Ataque de denegación del servicio e interrupción del servicio

Esta regla identificará una potencial violación de la seguridad tras un ataque de denegación del servicio. Esta regla emitirá un alerta si el destino de un ataque de denegación de servicio ha interrumpido un servicio a los 60 segundos del ataque. Un Recopilador de host, es decir, HIDS/OS, puede detectar si se interrumpe el servicio. El ataque por desbordamiento de buffer puede detectarse mediante Recopiladores OS, NIDS o HIDS.

Si un sistema se viese afectado por un ataque de denegación de servicio, debería investigarse con mayor detenimiento.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información) ▪ Plataformas IDS de host que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía HIDS y OS para obtener más información)	NIDS HIDS/OS

RuleLg para esta regla

```
filter ((e.rv51 = "Service" and e.rv52 = "Stop" ) and
   (e.st = "H")) flow window (w.dip = e.sip, filter
   (e.rv52 ="DoS" ), 60) flow trigger(1, 0)
```

Detección de propagación de virus

Esta regla identificará si un virus conocido ataca un sistema dentro de una infraestructura.

Cuando un virus ataca, por lo general, un sistema o varios se ven negativamente afectados y es necesario volver a cargar completamente el sistema y los datos de aplicaciones o se pierden completamente los activos de la empresa. La identificación de un virus mientras está en progreso puede reducir significativamente el daño o bien impedirlo.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
3 veces en 5 minutos	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv52 ="Virus") flow trigger (3, 300)
```

Detección de propagación de un gusano

Esta regla identificará si un gusano conocido ataca un sistema dentro de una infraestructura.

Cuando un gusano ataca, por lo general, un sistema o varios se ven negativamente afectados y es necesario volver a cargar completamente el sistema y los datos de aplicaciones o se pierden completamente los activos de la empresa. La identificación de un gusano mientras está en progreso puede reducir significativamente el riesgo de la empresa.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
3 veces en 5 minutos	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv52 = "Worm") flow trigger (3, 300)
```

Detección de caballo troyano

Esta regla identificará si un caballo troyano conocido se ha implantado en un sistema dentro de una infraestructura.

Cuando un caballo troyano logra su cometido, el sistema apuntado puede verse totalmente comprometido.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
3 veces en 5 minutos	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información) ▪ Plataformas IDS de host que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía HIDS y OS para obtener más información)	NIDS HIDS/OS

RuleLg para esta regla

```
filter (e.rv52 ="Trojan") flow trigger (3, 500)
```

Múltiples intentos de backdoor (puerta trasera) desde un único origen

Esta regla correlacionará múltiples intentos de insertar o ejecutar un ataque indirecto desde un único origen.

Un programa indirecto normalmente se utiliza para obtener el control completo de un sistema apuntado y luego se utiliza para lanzar otros ataques. Por lo general, esta regla identificará los intentos de un intruso de buscar en un sistema infectado o de procurar infectar un sistema.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
5 veces en 2 minutos	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información) ▪ Plataformas IDS de host que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía HIDS y OS para obtener más información)	NIDS HIDS/OS

RuleLg para esta regla

```
filter (e.rv50 = "Attack" and e.rv52 = "Backdoor" ) flow
   trigger(5, 120, discriminator (e.sip))
```

Múltiples intentos de backdoor (puerta trasera) desde distintos orígenes

Esta regla correlacionará múltiples intentos de insertar o ejecutar un ataque indirecto coordinado desde diferentes sistemas que apunten a un único destino.

Un programa indirecto normalmente se utiliza para obtener el control completo de un sistema apuntado y luego se utiliza para lanzar otros ataques. Por lo general, esta regla identifica que:

- el sistema de destino haya sido comprometido
- el atacante intente explotar el sistema comprometido
- el atacante intenta enmascararse con un ataque coordinado
- o que el atacante haya tomado conocimiento de que el destino es vulnerable a este tipo de ataque. Si este es el caso, esto podría indicar que el atacante ha tomado conocimiento ha partir de un origen interno.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
5 veces en 2 minutos	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información) ▪ Plataformas IDS de host que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía HIDS y OS para obtener más información)	NIDS HIDS/OS

RuleLg para esta regla

```
filter (e.rv50 = "Attack" and e.rv52 = "Backdoor" ) flow
   trigger( 5, 120, discriminator(e.dip))
```

Múltiples errores de entrada desde cualquier origen a cualquier destino

Esta regla identificará errores de entrada en los mismos tipos de sistemas.

Los errores de entrada al mismo tipo de cuenta o sistema pueden indicar que el atacante tenía un conocimiento previo de la red y de los sistemas críticos ubicados en la red. Esto provocaría una alarma. Cuanta mayor información tenga el atacante, más fácil será para él encontrar un sistema que pueda explotar.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
5 veces en 2 minutos	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter ((e.rv52 = "Access" or e.rv52 = "Brute_Force") and
   e.rv51 = "User" and e.rv50 = "Attack") flow trigger
   (5, 120)
```

Múltiples errores de entrada desde el mismo origen al mismo destino

Esta regla identificará múltiples errores de entrada del mismo origen al mismo destino.

Los errores de entrada al mismo tipo de cuenta o sistema pueden indicar que el atacante tenía un conocimiento previo de la red y de los sistemas críticos ubicados en la red. Esto provocaría una alarma. Cuanta mayor información tenga el atacante, más fácil será para él encontrar un sistema que pueda explotar.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
3 veces en 5 minutos	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter ((e.rv52 = "Access" or e.rv52 = "Brute_Force") and
   e.rv51 = "User" and e.rv50 = "Attack") flow trigger
   (5, 120, discriminator (e.sip, e.dip))
```

Ataque por debordamiento de buffer desde el mismo origen al mismo destino

Esta regla identificará un ataque por desbordamiento de buffer desde la misma dirección IP de origen a la misma dirección IP de destino.

Un ataque por desbordamiento de buffer es el ataque más común en la red y se utiliza para deshabilitar un sistema. Estos tipos de ataques sólo se pueden bloquear en el perímetro. El hecho de conocer un sistema atacante puede contribuir a bloquear ese sistema.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
5 veces en 3 minutos	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información) ▪ Plataformas IDS de host que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía HIDS y OS para obtener más información)	NIDS HIDS/OS

RuleLg para esta regla

```
filter (e.rv52 ="Buffer_Overflow" ) flow trigger (5, 180,
   discriminator (e.sip, e.dip))
```

Ataque de fuerza bruta satisfactorio con mismo origen y destino

Esta regla identificará un sistema posiblemente comprometido con una contraseña convertida.

Un intento constante de utilizar combinaciones de nombres de usuario y contraseñas para poder acceder, seguido de una eventual entrada satisfactoria podría indicar que un atacante ha logrado el acceso a través de un ataque de fuerza bruta. Si este ataque se logra con éxito, se debe cerrar la cuenta a la que se accede.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez en 3 minutos	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información) ▪ Plataformas IDS de host que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía HIDS y OS para obtener más información)	NIDS HIDS/OS

RuleLg para esta regla

```
filter (e.rv53="Other" and rv52="Access" e.rv51 ="User"
   and e.rv50="Prob" and e.st = "H") flow window (w.dip =
   e.sip, filter (e.rv52="Brute Force" and
   e.rv50="Compromise"), 180) flow trigger(1, 180,
   discriminator(e.sip, e.dip))
```

Verificación de ataques de Internet Information Server (IIS) de Microsoft

Esta regla admite los 10 ataques principales de SANS Microsoft en un ataque de Internet Information Service (IIS). Si ejecuta la aplicación IIS de Microsoft, puede ser vulnerable a un ataque.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv53 = "Sans_MS_IIS") flow trigger(1,60)
```

Verificación de ataques de servicios de datos remotos - Data Access Connector (MDAC) de Microsoft

Esta regla admite los 10 ataques principales de SANS Microsoft en MDAC. El uso de productos de Microsoft puede aumentar la vulnerabilidad a ataques. MDAC es una herramienta subyacente que se utiliza para integrar productos de Microsoft.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv53 = "Sans_MS_MDAC") flow trigger(1,60)
```

Verificación de ataques de SQL Server - Ataques SQL Server de Microsoft

Esta regla admite los 10 ataques principales de SANS Microsoft en SQL Server de Microsoft. El uso de SQL Server de Microsoft puede ser vulnerable a ataques. Existen varias vulnerabilidades graves que permiten que atacantes remotos obtengan información sensible, alerten sobre el contenido de base de datos, comprometan servidores SQL y comprometan hosts de servidores.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv53 = "Sans_MS_SQLServer") flow trigger(1,60)
```

Verificación de ataques de intercambios de red de Windows sin protección de NETBIOS de Microsoft

Esta regla admite los 10 ataques principales de SANS Microsoft en NETBIOS. El uso de interconexión con NETBIOS de Microsoft puede aumentar la vulnerabilidad a un ataque. NETBIOS era el software de comunicaciones de interconexión original de Microsoft. Las redes actuales de Microsoft no confían en NETBIOS como medio de transporte.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv53 = "Sans_MS_NETBIOS") flow trigger(1,60)
```

Verificación de ataques Null Sessions - Entrada anónima de Microsoft

Esta regla admite los principales 10 ataques de Microsoft SANS en Null Sessions. Si está usando Null Session de Microsoft, puede ser vulnerable a un ataque. El usuario anónimo puede recuperar información a través de la red o conectarse sin autenticación.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv53 = "Sans_MS_NullSessions") flow
   trigger(1,60)
```

Verificación de ataques de Weak LM Hashing (Parcialización LM débil) - Autenticación LAN Manager (LM) de Microsoft

Esta regla admite los 10 ataques principales de SANS Microsoft de un ataque en parcialización LM débil LM utiliza un esquema de cifrado mucho más débil que los protocolos de autenticación actuales de Microsoft (NTLM y NTLMv2) y las contraseñas de LM pueden dañarse en un período de tiempo breve.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv53 = "Sans_MS_LM") flow trigger(1,60)
```

Verificación de ataque de autenticación de General Windows de Microsoft

Esta regla admite los 10 ataques principales de SANS Microsoft en contraseñas. Cuando se detectan contraseñas débiles, deben ser reemplazadas.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv53 = "Sans_MS_WeakPasswords") flow
   trigger(1,60)
```

Verificación de ataques de Internet Explorer (IE) de Microsoft

Esta regla admite los 10 ataques principales de SANS Microsoft en IE. Las últimas versiones de Microsoft han incorporado esta aplicación en la interfaz de usuario del sistema operativo. Los ataques conocidos con IE podrían comprometer cualquier entorno de Microsoft posterior a Windows 2000.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv53 = "Sans_MS_IE") flow trigger(1,60)
```

Verificación de ataque de acceso remoto al registro de Microsoft

Esta regla admite los 10 ataques principales de SANS Microsoft en el registro de Microsoft. El registro de un sistema operativo de Microsoft es donde se alojan todas las variables definidas por el sistema. La capacidad de modificar o reemplazarlas puede afectar negativamente a la operación o a la seguridad de una plataforma de Microsoft.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv53 = "Sans_MS_Registry") flow trigger(1,60)
```

Verificación de ataque de secuencia de comandos de Windows de Microsoft

Esta regla admite los 10 ataques principales de SANS Microsoft en la secuencia de comandos de Windows. Una serie de aplicaciones de Microsoft se desarrollan a partir de lenguaje de programación de Visual Basic. La capacidad para ejecutar comandos mediante una secuencia de comandos le permite al atacante acceder y controlar un sistema Microsoft.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv53 = "Sans_MS_Scripting") flow trigger(1,60)
```

Verificación de ataque Remote Procedure Call (RPC) de UNIX

Esta regla admite los 10 ataques principales de SANS UNIX en RPC. Las Remote Procedure Calls (llamadas de procedimiento remoto) constituyen un método dentro de un entorno UNIX para acceder o ejecutar algunas aplicaciones o archivos en un sistema remoto sin autenticación. El hecho de dejar RPC abierto permite que cualquier usuario remoto ejecute comandos de privilegio en un sistema sin autenticación. RPC puede dar lugar a ataques remotos.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv53 = "Sans_Unix_RPC") flow trigger(1,60)
```

Verificación de ataque de servidor Web Apache de UNIX

Esta regla admite los 10 ataques principales de SANS UNIX en servidores Web Apache. El servidor Web Apache es una aplicación gratuita que admite servidores Web. El hecho de ejecutar un servidor Web Apache puede dejarlo librado a este tipo de ataque.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv53 = "Sans_Unix_Apache") flow trigger(1,60)
```

Verificación de ataque Secure Shell de UNIX

Esta regla admite los 10 principales ataques de SANS UNIX en Secure Shell. Con la cantidad de problemas con telnet y FTP, Secure Shell se desarrolló para cifrar el tráfico entre dos equipos. Esta aplicación permite la transferencia de datos o la interacción con un sistema remoto a través de un método seguro. Sin embargo, se han identificado una serie de errores con versiones de esta aplicación que permiten a un atacante tomar el control completo del sistema apuntado.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv53 = "Sans_Unix_SSH") flow trigger(1,60)
```

Verificación de ataque de Simple Network Management Protocol (SNMP) de UNIX

Esta regla admite los 10 principales ataques de SANS UNIX en SNMP. SNMP se diseñó originalmente para administrar nodos en una red. Nunca se implementaron medidas de seguridad en SNMP V 1.0 y unas pocas se implementaron en SNMP V 3.0. En consecuencia, SNMP está sujeto a una serie de ataques.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv53 = "Sans_Unix_SNMP") flow trigger(1,60)
```

Verificación de ataque de File Transfer Protocol (FTP o protocolo de transferencia de archivos) de UNIX

Esta regla admite los 10 ataques principales de SANS UNIX en FTP. Este protocolo constituye una parte vital de la comunicación dentro de Internet. Como tal, es un objetivo primordial de los atacantes para redirigir el acceso hacia Internet y desde ella.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv53 = "Sans_Unix_FTP") flow trigger(1,60)
```

Verificación de ataque de Remote Services (Servicios remotos) de UNIX

Esta regla admite los 10 ataques principales de SANS UNIX en Remote Services. Remote Services constituye un método dentro de un entorno UNIX para acceder o ejecutar algunas aplicaciones o archivos en un sistema remoto sin autenticación. El hecho de dejar Remote Services abierto permite que cualquier usuario remoto ejecute comandos de privilegio en un sistema sin autenticación. lo que permite posibles ataques remotos.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv53 = "Sans_Unix_RemoteServices") flow
   trigger(1,60)
```

Verificación de ataque Line Printer Daemon de UNIX

Esta regla admite los 10 ataques principales de SANS UNIX en Line Printer Daemon. Line Printer Daemon es el mecanismo que utiliza UNIX para imprimir archivos. Esta aplicación se ejecuta en un entorno UNIX en una cuenta raíz. Los numerosos errores hallados en esta aplicación permiten que un atacante tome el control completo de un entorno UNIX.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv53 = "Sans_Unix_LPD") flow trigger(1,60)
```

Verificación de ataque Sendmail de UNIX

Esta regla admite los 10 ataques principales de SANS UNIX en Sendmail. La aplicación Sendmail utiliza Simple Mail Transport Protocol (SMTP o protocolo de transporte de correo simple). Esta aplicación constituye una parte vital de la comunicación dentro de Internet. Como tal, es un objetivo primordial de los atacantes para redirigir el acceso hacia Internet y desde ella.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv53 = "Sans_Unix_SendMail") flow trigger(1,60)
```

Verificación de ataque BIND/DNS de UNIX

Esta regla admite los 10 ataques principales de UNIX SANS en ataques DNS. El Domain Name Service (DNS o servicio de nombre de dominio) constituye una parte vital de la comunicación dentro de Internet. Como tal, es un objetivo primordial de los atacantes para redirigir el acceso hacia Internet y desde ella.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv53 = "Sans_Unix_DNS") flow trigger(1,60)
```

Verificación de ataque de autenticación de General UNIX de UNIX

Esta regla admite los 10 ataques principales de SANS UNIX en contraseñas débiles. Cuando se detectan contraseñas débiles, deben ser reemplazadas.

Frecuencia de regla	Requisitos de la regla	Taxonomía de regla
1 vez	Defina lo siguiente antes de implementar esta regla: ▪ Plataformas IDS de red que la taxonomía de Sentinel pueda convertir (consulte la tabla Taxonomía NIDS para obtener más información)	NIDS

RuleLg para esta regla

```
filter (e.rv53 = "Sans_Unix_WeakPasswords") flow
   trigger(1,60)
```

Tablas de taxonomía

Esta sección contiene dos tablas. Estas tablas son:

- Taxonomía NIDS
- Taxonomía HIDS y OS

Ofrecen una lista de distintos valores de e.rv50 a e.rv53 para los ejemplos de RuleLg ilustrados.

Tabla de taxonomía de NIDS

Acción – Nivel1 (e.rv50)	**Sistema– Nivel2 (e.rv51)**	**Detalle – Nivel3 (e.rv52)**	**Resultados – Nivel4 (e.rv53)**
Ataque	Charla	Access	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
	DNS	Access	Sans_Unix_DNS
		Buffer_Overflow	Sans_Unix_DNS
		Backdoor	
		Brute_Force	
		DoS	
	Correo	Access	Sans_Unix_SendMail
		Buffer_Overflow	Sans_Unix_SendMail Sans_MS_IE
		Backdoor	
		Brute_Force	
		DoS	
	Telnet	Access	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
	Archivo	Access	Sans_Unix_FTP Sans_MS_WeakPasswords Sans_MS_NETBIOS
		Buffer_Overflow	Sans_Unix_FTP
		Backdoor	Sans_Unix_FTP
		Brute_Force	
		DoS	
	Web	Access	Sans_Unix_Apache Sans_MS_NETBIOS Sans_MS_WeakPasswords Sans_MS_IIS Sans_MS_Scripting Sans_MS_SQLServer Sans_MS_IE SANS_MS_MDAC

Acción – Nivel1 (e.rv50)	Sistema– Nivel2 (e.rv51)	Detalle – Nivel3 (e.rv52)	Resultados – Nivel4 (e.rv53)
		Buffer_Overflow	Sans_Unix_Apache Sans_MS_IIS
		Backdoor	
		Brute_Force	Sans_MS_IIS
		DoS	Sans_Unix_Apache Sans_MS_IIS
	PC	Virus	Sans_MS_IE Sans_MS_IIS
		Script	
		Worm	Sans_MS_SQLServer
		Trojan	
	Servidor	Access	Scan_MS_IIS Sans_MS_Registry Sans_MS_SQLServer Sans_MS_NETBIOS Sans_Unix_remoteServices Sans_Unix_RPC Sans_Unix_SSH
		Buffer_Overflow	Sans_Unix_RemoteServices Sans_Unix_WeakPasswords Sans_Unix_RPC Sans_Unix_LPD Sans_MS_SQLServer Sans_MS_MDAC Sans_MS_NETBIOS Sans_Unix_SSH
		Backdoor	Sans_Unix_RPC
		Brute_Force	Sans_MS_SQLServer Sans_MS_WeakPasswords
		DoS	
	Protocolo	IP	
		TCP	
		UDP	
		ICMP	
		HTTP	
		Route	
		Talk	
		XFS	
		SSH	
		IGMP	
		Tiempo	
		News	
		Windows	
		RIP	
		IDS	
		SNMP	Sans_Unix_SNMP
		BGP	

Acción – Nivel1 (e.rv50)	Sistema– Nivel2 (e.rv51)	Detalle – Nivel3 (e.rv52)	Resultados – Nivel4 (e.rv53)
	Usuario	Access	Sans_Unix_WeakPasswords Sans_Unix_RemoteServices
		Buffer_Overflow	Sans_Unix_RemoteServices Sans_MS_NETBIOS
		Backdoor	
		Brute_Force	
		DoS	
Prueba	Charla		
	DNS		
	Correo		
	Archivo		Sans_Unix_FTP
	Web		Sans_MS_IIS Sans_Unix_Apache
	PC		
	Servidor		Sans_MS_NullSessions Sans_MS_Registry
	Protocolo	IP	
		TCP	
		RIP	
		SNMP	Sans_Unix_SNMP
		SSH	
		Talk	
		Time	
		Windows	
		UDP	
		ICMP	
		DHCP	
	Exploración		
	Telnet		Sans_MS_LM
	Usuario		Sans_MS_LM
	IDS		
Directiva	Porno		
Comprometido	Chat	Access	
		Buffer_Overflow	Sans_Unix_Weak_Passwords
		Backdoor	
		Brute_Force	
		DoS	
	DNS	Access	Sans_Unix_DNS
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
	Correo	Access	
		Buffer_Overflow	
		Backdoor	Sans_Unix_SendMail

Acción – Nivel1 (e.rv50)	Sistema– Nivel2 (e.rv51)	Detalle – Nivel3 (e.rv52)	Resultados – Nivel4 (e.rv53)
		Brute_Force	
		DoS	
	Telnet	Access	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
	Archivo	Access	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
	Web	Access	Sans_Unix_Apache
		Buffer_Overflow	Sans_MS_IIS
		Backdoor	Sans_Unix_Apache Sans_MS_Registry
		Brute_Force	
		DoS	
	PC	Virus	
		Script	
		Worm	
		Trojan	
	Servidor	Access	Sans_MS_SQLServer
		Buffer_Overflow	Sans_Unix_RPC
		Backdoor	Sans_MS_WeakPasswords Sans_MS_Registry Sans_Unix_SNMP Sans_Unix_WeakPasswords
		Brute_Force	
		DoS	
	Usuario	Access	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	

Tabla de taxonomía de HIDS y OS

Acción – Nivel1 (e.rv50)	Sistema– Nivel2 (e.rv51)	Detalle – Nivel3 (e.rv52)	Resultados – Nivel4 (e.rv53)
Ataque	Archivo	Delete	App OS
		Execute	App OS
		Create	App OS

Acción – Nivel1 (e.rv50)	Sistema– Nivel2 (e.rv51)	Detalle – Nivel3 (e.rv52)	Resultados – Nivel4 (e.rv53)
		Modify	App OS
		Access	App OS
	Servicio	Delete	App OS
		Detener	App OS
		Start	App OS
		Create	App OS
		Access	App OS Priv Correo ID Red Archivo Sistema
		Buffer_Overflow	
		Backdoor	
		DoS	
	Config	Delete	App OS
		Modify	App OS
		Create	App OS
		Enable	App OS
		Access	App OS
	Usuario	Create	ID Auth Param Priv
		Modify	ID Auth Param Priv
		Delete	ID Auth Param Priv

Acción – Nivel1 (e.rv50)	Sistema– Nivel2 (e.rv51)	Detalle – Nivel3 (e.rv52)	Resultados – Nivel4 (e.rv53)
		Access	Guest Priv Root Otros
	Grupo	Create	Member Group
		Modify	Member Group
		Delete	Member Group
	Sistema	Información	
		Memoria	
		Debug	
	Anomalía		
	Telnet	Access	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
	Web	Access	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
	PC	Virus	
		Script	
		Backdoor	
		Worm	
		Trojan	
	DNS	Access	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
	Correo	Access	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
Prueba	Archivo	Delete	App OS
		Execute	App OS
		Create	App OS
		Modify	App OS

Acción – Nivel1 (e.rv50)	Sistema– Nivel2 (e.rv51)	Detalle – Nivel3 (e.rv52)	Resultados – Nivel4 (e.rv53)
		Access	App OS
	Servicio	Delete	App OS
		Detener	App OS
		Start	App OS
		Create	App OS
		Access	App OS Archivo ID Correo Priv Red Sistema
	Config	Delete	App OS
		Modify	App OS
		Create	App OS
		Enable	App OS
		Access	App OS
	Usuario	Create	ID Auth Param Priv
		Modify	ID Auth Param Priv
		Delete	ID Auth Param Priv
		Access	Guest Root Otros
	Grupo	Create	Member Group
		Modify	Member Grupo

Acción – Nivel1 (e.rv50)	**Sistema– Nivel2 (e.rv51)**	**Detalle – Nivel3 (e.rv52)**	**Resultados – Nivel4 (e.rv53)**
		Delete	Member Group
	Sistema	Información	
		Memoria	
		Debug	
	Anomalía		
	Web	Access	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
	Correo	Access	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
	Protocolo	IP	
		TCP	
		UDP	
		ICMP	
		HTTP	
		Route	
		Talk	
		XFS	
		SSH	
		IGMP	
		Time	
		News	
		Windows	
		RIP	
		IDS	
		SNMP	
		BGP	

Salida de correlación

La estructura de salida del Motor de correlación permite la clasificación, el filtrado y la generación de informes sobre datos elaborados como parte de una regla de Lista de vigilancia o de correlación.

Estructura de salida de una regla de correlación

Los valores de salida por defecto son:

- Configuración de RES en “Correlación” excepto que lo haya configurado el usuario
- SubRes configurado en "<regla>.<nombrederegla> excepto que lo haya configurado el usuario
- Sev configurado en 4 excepto que lo haya configurado el usuario
- ST (tipo de sensor - C)
- EI (patrón de regla - SIP=’1.2.3.4.’ luego punto y coma, luego umbral de regla en formato de 3-2-m (total de 3 en 2 minutos, por ejemplo)
- RT2 (nombre de regla)

Parámetros de guión transferidos

Los parámetros de guión aprobados afectan tanto las reglas de lista de vigilancia como a las reglas de correlación. Los parámetros de guión son específicos en el cuadro de entrada Realizar la acción de la pestaña Criterios de activación con el formato %xyz% donde xyz representa el nombre del parámetro. Los nombres de los parámetros que representan las meta-etiquetas pueden ser un nombre abreviado (como sip) o un nombre extendido (como SourceIP). Los nombres de parámetros distinguen entre mayúscula y minúscula.

Parámetros

Los primeros once parámetros son parámetros especiales. No son meta-etiquetas. Corresponden a eventos correlacionados. Los parámetros doce a cuarenta y siete son parámetros de meta-etiquetas.

1. `%RuleName%` - El nombre de la regla que activó (el formato es rule.rulename).
2. `%RuleType%` - El tio de la regla que se ha activado. C corresponde a correlación. W corresponde a lista de vigilancia.
3. `%RuleDescription%` - La descripción que se introdujo cuando se creó la regla.
4. `%RuleSeverity%` - La gravedad de la regla que se activó.
5. `%RuleResource%` - El nombre del recurso de la regla que se ha activado.
6. `%RuleSubResource%` - El nombre del subrecurso de la regla que se ha activado.
7. `%RuleLg%` - La regla en el lenguaje de regla del Motor de correlación (RuleLg).
8. `%RuleCount%` - El total de la regla que se ha activado.
9. `%RuleDuration%` - La duración (en segundos) de la regla que se ha activado.
10. `%RulePattern%` - Una lista de todas las etiquetas en el lenguaje de la regla y el valor de la etiqueta extraído del último evento que activó la regla . El formato es tsn1=‘value1=’value2’tsn3=’value3’, donde:
    - tns1 corresponde al nombre corto de la etiqueta 1
    - tns2 corresponde al nombre corto de la etiqueta 2

Por ejemplo:

```
sip='192.168.0.3'dip='2.168.0.2'
```

11. `%CorrelatedEventID%` - El identificador del evento del evento correlacionado generado por la regla que se ha activado.
12. %MessageText% - El texto del mensaje de la regla que se ha activado.
13. %EventName% - El nombre del evento de la regla que se ha activado.

Las etiquetas restantes corresponden al campo del último evento que activó el evento correlacionado.

14. %sev% - Gravedad: La gravedad normalizada del evento (0-5).
15. %vul% - Vulnerabilidad: La vulnerabilidad del activo identificado en este evento.
16. %crt% - Importancia: La importancia del activo identificado en este evento.
17. %dt% - Fecha y hora: La fecha y hora normalizadas del evento, según la indica el recopilador.
18. %sip% - IP de origen: La dirección IP de origen desde la que se origina el evento.
19. %dip% - IP de destino: La dirección IP de destino a la que se apuntó el evento.
20. %id% - ID de evento: Identificador único (UUID) para este evento
21. %src% - ID de origen: Identificador único (UUID) del proceso de Sentinel que generó este evento.
22. %port% - Puerto de asistente: Descripción del puerto del recopilador de Sentinel.
23. %agent% - Recopilador del asistente: Descripción del puerto del recopilador de Sentinel.
24. %res% - Recurso: El nombre del recurso.
25. %sres% - Subrecurso: El nombre del subrecurso.
26. %evt% - Nombre de evento: el nombre descriptivo del evento según lo informa (o brinda) el sensor. Ejemplo: “Port Scan”.
27. %sn% - Nombre de sensor: El nombre del “detector definitivo” del evento cuando se reciben datos sin procesar. Por ejemplo, “FW1” para un cortafuegos.
28. %st% - Tipo de sensor: El designador de carácter simple para el tipo de sensor single (N, H, O, V, C, W). H: de host, N: de red, O: Otro, V: Antivirus, C: Correlación y W: Lista de vigilancia.
29. %et% - Hora de evento: La hora normalizada del evento, según la informa el sensor; analizada en el formato: Y-M-D-H:M:S~AMPM24~TZ.
30. %prot% - Protocolo: El protocolo de red del evento.
31. %shn% - Nombre de host de origen: El nombre del host de origen desde el que se origina el evento.
32. %sp% - Puerto de origen: El puerto del origen desde el que se origina el evento.
33. %dhn% - Nombre de host de destino: El nombre de host de destino al que se apuntó el evento.
34. %dp% - Puerto de destino: El puerto de destino al que se apuntó el evento.
35. %sun% - Nombre de usuario de origen: El nombre de usuario de origen utilizado para iniciar un evento. Por ejemplo, “jdoe” durante un intento de “su”.

36. %dun% - Nombre de usuario de destino: El nombre de usuario de destino en el que se intentó la acción. Ejemplo: Intentos para reiniciar la contraseña de la raíz.
37. %fn% - Nombre de archivo: El nombre del programa que se ejecuta o el archivo que se accede, modifica o afecta. Ejemplo: El nombre de un archivo infectado con virus o de un programa detectado por un IDS.
38. %ei% - Información ampliada: Almacena más información recopilada del recopilador. Los valores de esta variable se separan con punto y coma (;). Ejemplo: Un dominio para una ID o nombres de archivo.
39. %rn% - Nombre de informador: El nombre del host y la dirección IP del dispositivo en el que se registró el evento o desde el que se envía la notificación del evento.
40. %pn% - Nombre de producto: Indica el nombre del tipo, proveedor o código de producto del sensor a partir del cual se genera el evento. Ejemplo: Check Point FireWall=CPFW.
41. %msg% - Mensaje: Texto de mensaje sin formato para el evento:.
42. %rt1% -Reservado por Novell para expansión. Para uso con Asesor (cadena).
43. %rt2% - Reservado por Novell para expansión (Cadena).
44. %ct1% - Reservado para uso de clientes, para datos específicos del cliente (cadena).
45. %ct2% - Reservado para uso de clientes, para datos específicos del cliente (cadena).
46. %rt3% - Reservado por Novell para expansión (Número).
47. %ct3% - Reservado para uso de clientes, para datos específicos del cliente (Número).
48. Parámetros de 46 a 145

    %rv1% thru %rv100%

    Estas son meta-etiquetas del evento actual que representan variables reservadas.
49. Parámetros de 146 a 245

    %cv1% thru %cv100%

    Estas son meta-etiquetas del evento actual que representan variables del cliente.

NOTA: Para obtener más información sobre Comandos y parámetros, consulte el capítulo 5 – Meta-etiquetas de Sentinel y el asistente en la Guía de referencia del usuario y el capítulo 9 Pestaña Admin, Reglas de correlación en la Guía del usuario.

Al utilizar el comando %all%:

- Si el valor del parámetro está vacío o es nulo, este valor será E_NULL o <tag absent>. De este modo, siempre habrá 45 parámetros independientemente de si algunos de los campos están en blanco.
- Cuando configure el motor de correlación para iniciar el guión de la interfaz HP OVO, deberá especificar el nombre del guión junto con la etiqueta de parámetro %all%:

  ```
  esec_ovo %all%
  ```

- Cuando configure el motor de correlación para iniciar el guión de la interfaz BMC, deberá especificar el nombre del guión junto con el parámetro %all%:

  ```
  bmc_interface.csh %all%
  ```

- Cuando configure el motor de correlación para enviar un mensaje por correo electrónico, deberá especificar el nombre del guión del mensjae de correo electrónico junto con el parámetro %all% y la dirección de correo y el asunto (opcional):

  ```
  email_interface.csh %all% <name>@<domain name> "My
     Subject"
  ```

- Todos los guiones o aplicaciones que puede ejecutar el motor de correlación deben hallarse en el directorio $ESEC_HOME/sentinel/exec (UNIX) %ESEC_HOME%\sentinel\bin (Windows).
- Por defecto, el motor de correlación NO transferirá ningún parámetro a los guiones que ejecuta. Debe utilizar las %tags% descriptas anteriormente si desea transferir algún parámetro a los guiones.
- Cuando especifique parámetros para un guión, para agruparlos puede utilizar comillas dobles. A continuación, se dan algunos ejemplos:

  ```
  %sip% %dip% - se considerará como dos parámetros.

  "%sip% %dip%" - se considerará como un solo
     parámetro.

  "Hello World" %sip% - se considerará como dos
     parámetros.

  "The message is %msg%" - se considerará como un
     parámetro.

  %msg% - se considerará como un parámetro (aun si el
     mensaje sustituido tiene espacios.)

  "%msg%" - también se considerará como un solo
     parámetro (aun si el mensaje sustituido tiene
     espacios.)
  ```

8 Opciones de línea de comando de correlaciones de Sentinel

NOTA: El término Agente puede intercambiarse con Recopilador. En adelante, los agentes se denominarán recopiladores.

Las opciones de línea de comando deben ser utilizadas por usuarios avanzados. Los usuarios normales no deben realizar modificaciones basadas en el uso de estas opciones. Para acceder a las opciones de línea de comando, vaya a:

En UNIX:

```
$ESEC_HOME/sentinel/bin
```

En Windows:

```
%ESEC_HOME%\sentinel\bin
```

Para ejecutar la opción de línea de comando, escriba:

```
correlation_engine <opción de línea de comando de
   correlaciones>
```

Opción de línea de comando de correlaciones	**Descripción**
-debug	Modo de depuración (imprime información de depuración extensa)
-noErrorLogging	Inhabilitar el registro de errores en el registro de eventos de Windows.
-ruleFile <archivo>	Especificar el archivo de texto que contiene las reglas que debe procesar la instancia del motor de correlación
-xmlruleFile <archivo>	Especificar el archivo de configuraciones para guardar una copia local de las reglas contenidas en la base de datos. Por defecto: startup_correlation_rules.xml
-inputChannel <cadena>	Especificar el canal de entrada del nivel de comunicaciones del motor de correlación. Por defecto: ewizard_binary_event
-outputChannel <cadena>	Especificar el canal de salida del nivel de comunicaciones del motor de correlación. Por defecto: correlation_binary_event.

Opción de línea de comando de correlaciones	Descripción
-outputUpdateChannel <cadena>	Especificar el canal de actualización de salida del nivel de comunicaciones del motor de correlación. Por defecto: correlation_binary_event_update
-outputExecuteChannel <cadena>	Especificar el canal de ejecución de salida del nivel de comunicaciones del motor de correlación. Por defecto: ejecutar
-outputIncidentChannel <cadena>	Especificar el canal de incidencia de salida del nivel de comunicaciones del motor de correlación. Por defecto: app_incident_req
-service <cadena>	Especificar el servicio de comunicaciones (parámetro de configuración) del motor de correlación. Por defecto: correlation_engine
-mgmtInputChannel <cadena>	Especificar el canal de entrada de gestión del nivel de comunicaciones del motor de correlación. Por defecto: correlation_mgmt_input_channel
-mgmtOutputChannel <cadena>	Especificar el canal de salida de gestión del nivel de comunicaciones del motor de correlación. Por defecto: correlation_mgmt_output_channel
-mgmtService <cadena>	Especificar el servicio de gestión de comunicaciones (parámetro de configuración) del motor de correlación. Por defecto: correlation_engine_mgmt
-configurationFile <archivo>	Especificar el archivo para anular los parámetros de configuración de inicio por defecto del motor de correlación. Por defecto: ± 30 segundos del tiempo del servidor de Sentinel.
-noStartupRules	Configurar el motor de correlación para que se ejecute sin recuperar las reglas almacenadas en la base de datos. La opción -ruleFile también omite la recuperación de la base de datos.
-dbTimeout <tiempo límite en milisegundos>	Definir el valor de tiempo límite para recuperar las reglas almacenadas en la base de datos. Por defecto: 5000 milisegundos
-dbRetries <número>	Definir el número de reintentos para conectarse con la base de datos. Por defecto: 6

Opción de línea de comando de correlaciones	Descripción
-name <nombre del motor>	Configurar el nombre del informador de este motor de correlación. Por defecto: Motor de correlación.
-affinityOneProcessor	Configurar el motor de correlación para que se ejecute en un solo procesador.
-useEventTime	Esta opción es para pruebas y no debe utilizarse.
-useNullOutput	Esta opción es para pruebas y no debe utilizarse.
-logFile <nombre de archivo>	Esta opción dirige el estado a un archivo.
-logPeriod <segundos>	Esta opción controla la frecuencia con que se escribe el estado en un archivo.
-version	Mostrar el número de versión y salir.
-help	Mostrar esta ayuda y salir.

9 Servicio de acceso a los datos de Sentinel

NOTA: El término Agente puede intercambiarse con Recopilador. En adelante, los agentes se denominarán recopiladores.

El proceso del Servicio de acceso a los datos (DAS, Data Access Service) es un servicio permanente del servidor de Sentinel y proporciona una interfaz MOM (bus de mensajes) a la base de datos. Ofrece acceso basado en datos al sistema de apoyo de la base de datos. Recibe peticiones XML desde los diferentes procesos de Sentinel, las convierte en una consulta a la base de datos, procesa el resultado de la base de datos y lo vuelve a convertir en una respuesta XML. Admite peticiones para recuperar eventos para una consulta rápida y para el detalle de eventos, para recuperar información de vulnerabilidades e información del asesor y para modificar la información de configuración. DAS también gestiona la entrada de todos los eventos que se reciben desde el Gestor de recopiladores del asistente y peticiones para recuperar y almacenar información de configuración.

Archivos del contenedor de DAS

DAS es un contenedor, formado por cinco procesos diferentes. Cada proceso es responsable de diferentes tipos de operaciones de la base de datos. Estos procesos son controlados por los siguientes archivos:

- das_binary.xml: se utiliza para la operación de inserción de eventos y eventos correlacionados.
- das_query.xml: se utiliza para el resto de operaciones de la base de datos.
- das_aggregation.xml: se utiliza para la operación de adición
- das_itrac.xml: se utiliza para ejecutar y configurar el servicio de actividades y para configurar el servicio de flujo de trabajo
- das_rt.xml: se utiliza para configurar la función Active Views en la consola de control de Sentinel

PRECAUCIÓN: No edite manualmente los archivos xml. Use la utilidad dbconfig para cambiar cualquier valor en los archivos xml.

Cada uno de estos procesos tiene un archivo de registro activo en %ESEC_HOME%\Sentinel\log o $ESEC_HOME/Sentinel/log. Son los siguientes:

- das_query0.*.log: todos los registros das_query
- das_binary0.*.log: todos los registros das_binary
- das_itrac0_*.log: registros de actividades y flujo de trabajo
- das_aggregation0.*.log: registros de adición
- das_rt0.*.log: registros de Active View

Los archivos xml especifican:

- ConnectionManager
  - username
  - password
  - hostname
  - portnumber
  - database (nombre de la base de datos)
  - server (oracle o mssql)
  - maxConnections
  - batchSize
  - loadSize
- DispatchManager — Especifica los canales del bus de mensajes que debe escuchar DAS. También especifica qué clase java utilizar para convertir las peticiones xml en objetos java e indica a qué gestor se debe enviar el objeto java para el procesamiento del mensaje. Por ejemplo: una petición de consulta de eventos se convierte en un objeto java a través del convertidor esecurity.cracker.QuickQueryRequestCracker. A continuación, el convertidor lo envía al gestor esecurity.event.request y éste lo envía a uno de los servicios para su ejecución.
- Y otros componentes que proporcionan servicios de DAS importantes.

Use la utilidad dbconfig para reconfigurar las propiedades de conexión de la base de datos para Windows.

Reconfiguración de las propiedades de conexión de la base de datos

Este procedimiento se debe ejecutar para cada uno de los nombres de archivo de contenedor siguientes (containerFilename):

- das_binary.xml
- das_query.xml
- das_rt.xml
- das_aggregation.xml
- das_itrac.xml

Reconfiguración de las propiedades de conexión de la base de datos para Windows

NOTA: En intervalos de 10 segundos, se revisará el archivo de propiedades del registro para ver si se han producido cambios desde la última vez que se leyó. Si el archivo ha cambiado, el LogManagerRefreshService volverá a leer el archivo de propiedades del registro.

1. Entre como un usuario con derechos administrativos a la ubicación donde está instalada la base de datos.
2. Vaya a:

   En Windows:

   ```
   %ESEC_HOME%\sentinel\config
   ```

   En UNIX:

   ```
   $ESEC_HOME/sentinel/config
   ```

3. Enter the following command:

```
dbconfig -n <nombre de archivo del contenedor> [-u
   username] [-p password] [-h hostname] [-t port
   number] [-d database] [-s server(mssql or oracle)]
   [-help] [-version]
```

Archivos de configuración de DAS

Para configurar el registro del proceso DAS se utilizan los archivos siguientes.

- das_query_log.prop
- das_binary_log.prop
- das_rt_log.prop
- das_itrac_log.prop
- das_aggregation_log.prop

  Su ubicación es la siguiente:

  En Windows:

  `%ESEC_HOME%\sentinel\config`

  En UNIX:

  `$ESEC_HOME/sentinel/config`

Estos archivos contienen la información de configuración del gestor de la consola, que imprime mensajes en una salida estándar y del gestor de archivos, que imprime mensajes en un archivo. La configuración de cada gestor permite que uno especifique las opciones disponibles para cada uno. Estos archivos le permiten especificar la configuración de los mensajes de registro que deben imprimirse. Los niveles son los siguientes:

- OFF: inhabilita todo el registro
- SEVERE (valor más alto): indica que un componente presenta un mal funcionamiento o que existe una pérdida/corrupción de datos importantes.
- WARNING: indica si una acción puede causar un mal funcionamiento de un componente en el futuro o si existen pérdidas/corrupción de datos no importantes.
- INFO: información de auditoría
- CONFIG
- FINE: para depuración
- FINER: para depuración
- FINEST (valor más bajo): para depuración
- ALL: registrará todos los niveles

Cuando uno especifica un nivel de registro, se registrarán todos los mensajes de registro de ese nivel y superiores (en la lista de arriba). Por ejemplo, si uno especifica el nivel INFO, se registrarán todos los mensajes con nivel INFO, WARNING y SEVERE.

Si se realiza un cambio en los archivos, deberá reiniciar DAS para que los cambios tengan efecto.

El registro se escribe en:

En Windows:

```
%ESEC_HOME%\sentinel\log\das_query_0.*.log
%ESEC_HOME%\sentinel\log\das_binary_0.*.log
%ESEC_HOME%\sentinel\log\das_itrac_0.*.log
%ESEC_HOME%\sentinel\log\das_aggregation0.*.log
```

En UNIX:

```
$ESEC_HOME/sentinel/log/das_query0.*.log
$ESEC_HOME/sentinel/log/das_binary0.*.log
$ESEC_HOME/sentinel/log/das_itrac_0.*.log
$ESEC_HOME/sentinel/log/das_aggregation0.*.log
```

El * indica el número exclusivo para solucionar conflictos y el número de generación para distinguir los registros rotados. Por ejemplo, das_query0.0.log es el registro con el archivo de índice 0 (primero) en un conjunto de archivos de registro del proceso DAS.

Conectores de BD nativos para la inserción de eventos

Los conectores de BD nativos ofrecen un mayor rendimiento para la inserción de eventos. El conector que debe utilizar depende de la plataforma de base de datos que utiliza.

Conector de BD nativo MS SQL

Usar el almacenamiento de eventos nativo ADO.Net.

Cómo configurar el conector nativo MS SQL

1. En el equipo en el que está instalado DAS, instale la estructura.Net.
2. En el archivo das_binary.xml, cambie la propiedad “insert.strategy” de EventStoreService > Persistor a:

   ```
   esecurity.ccs.comp.event.jdbc.ADOLoadStrategy
   ```

Conector de BD nativo Oracle

Usar el almacenamiento de eventos nativo OCI. Como mínimo, Oracle Client debe estar instalado en la máquina de DAS.

Cómo configurar el conector nativo Oracle

1. Cree un archivo “.profile” en el directorio personal de esecadm. Inluya el texto siguiente en dicho archivo (modifique ORACLE_HOME para su instalación):

   ```
   ORACLE_HOME=/build/home/oracle/OraHome
   export ORACLE_HOME
   LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$ORACLE_HOME/lib
   export LD_LIBRARY_PATH
   ```

2. En el archivo das_binary.xml, cambie la propiedad “insert.strategy” de EventStoreService > Persistor a:

   ```
   esecurity.ccs.comp.event.jdbc.OCILoadStrategy
   ```

10 Cambio de las contraseñas de usuario por defecto

NOTA: El término Agente puede intercambiarse con Recopilador. En adelante, los agentes se denominarán recopiladores.

En este capítulo se describe cómo cambiar las contraseñas de los usuarios por defecto de Sentinel:

Autenticación de Oracle y MS SQL:

- esecadm
- esecapp
- esecdba
- esecrpt

Autenticación de Windows:

- Administrador de Sentinel
- Usuario de BD de aplicación de Sentinel
- Administrador de BD de Sentinel
- Usuario de informes de Sentinel

Cambio de las contraseñas de usuario por defecto para la autenticación de Oracle y MS SQL

NOTA: Para cambiar las contraseñas es preciso tener derechos administrativos.

Cambio de la contraseña de esecadm

Cambio de la contraseña de esecadm

1. Entre a la Consola de control de Sentinel y haga clic en la pestaña *Admin*.
2. Abra la ventana *Gestor de usuarios*.
3. Haga doble clic en la cuenta de usuario esecadm o haga clic con el botón derecho del ratón en > *Información del usuario*.
4. Modifique la contraseña de la cuenta.
5. Haga clic en *Aceptar*.

Cambio de la contraseña de esecapp

Cambio de la contraseña de esecapp

1. Para MS SQL, utilice MS SQL Enterprise Manager y cambie la contraseña de esecapp.
2. En Oracle, utilice Oracle Enterprise Manager y cambie la contraseña de esecapp.

3. Con la utilidad dbconfig, actualice todos los archivos xml del contenedor. Esto es necesario dado que los archivos xml almacenan la contraseña de esecapp (cifrada) para permitir que DAS y el Asesor se conecten con la base de datos.

   - das_binary.xml
   - das_query.xml
   - activity_container.xml
   - workflow_container.xml
   - das_rt.xml

   Los archivos xml del contenedor se encuentran en las ubicaciones siguientes:

   En Windows:

   ```
   %ESEC_HOME%\sentinel\config
   ```

   En Oracle:

   ```
   $ESEC_HOME/sentinel/config
   ```

   Para obtener más información acerca del uso de la utilidad dbconfig, consulte la Guía de referencia de Sentinel, Capítulo 9 - Servicio de acceso a los datos de Sentinel.

   ```
   dbconfig –a <DirectorioContenedor> -p <contraseña>
   ```

Cambio de la contraseña de esecdba

Cambio de la contraseña de esecdba

1. Para MS SQL, utilice MS SQL Enterprise Manager y cambie la contraseña de esecdba.
2. En Oracle, utilice Oracle Enterprise Manager y cambie la contraseña de esecdba.
3. Para que las tareas automatizadas de SDM continúen funcionando (p. ej., añadir partición, archivar partición) actualice la dbPass en el archivo sdm.connect con la contraseña de esecdba nueva utilizando la GUI de SDM o la línea de comando. Para obtener más información, consulte la Guía del usuario de Sentinel, Capítulo 10 – Gestor de datos de Sentinel.

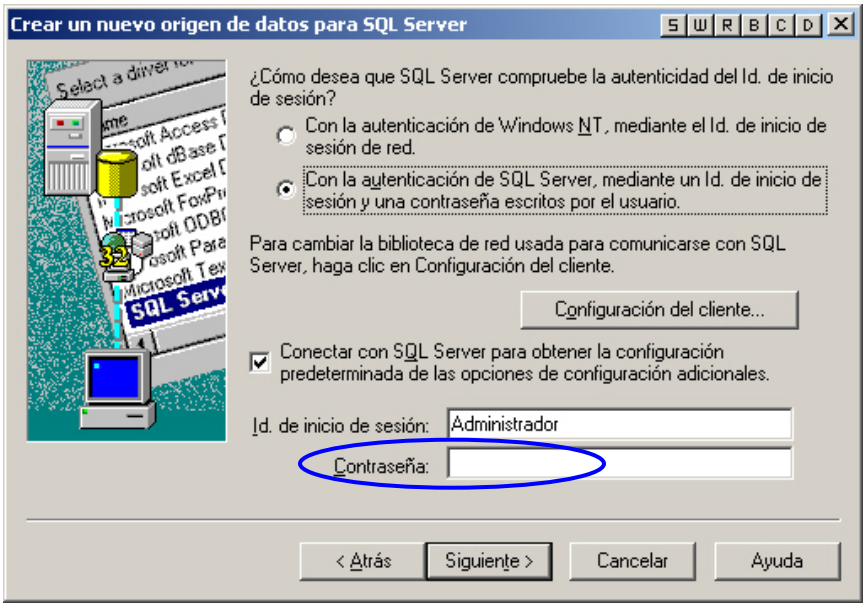
   ```
   sdm -action saveConnection -server <oracle/mssql> -
      host <IPhost/NombreHost> -port <NúmPuerto> -
      database <NombreBDD/SID> [-driverProps
      <ArchivoPropiedades>] {-user <UsuarioBDD> -password
      <contraseñaBDD>} -connectFile
      <NombreArchivoGuardarConexión>
   ```

Cambio de la contraseña de esecrpt

Cambio de la contraseña de esecrpt

1. Para la base de datos MS SQL de Sentinel, utilice MS SQL Enterprise Manager y cambie la contraseña de esecrpt.
2. Para la base de datos Oracle de Sentinel, utilice Oracle Enterprise Manager y cambie la contraseña de esecrpt.

3. Crystal Server para MS SQL de Sentinel, si corresponde, en la máquina de Crystal Server actualice ODBC DSN (*Panel de control> Herramientas administrativas > Orígenes de datos (ODBC)*).
   a. En la pestaña DSN del sistema, seleccione sentineldb y haga clic en *Configurar*.
   b. Haga clic en *Siguiente*. Actualice la contraseña.
   c. Haga clic en *Siguiente* hasta que aparezca el botón Finalizar. Haga clic en *Finalizar*.



4. Crystal Server para Oracle de Sentinel, no se requieren cambios.

Cambio de las contraseñas de usuario por defecto para la autenticación de Windows

Cambio de la contraseña de Administrador de Sentinel

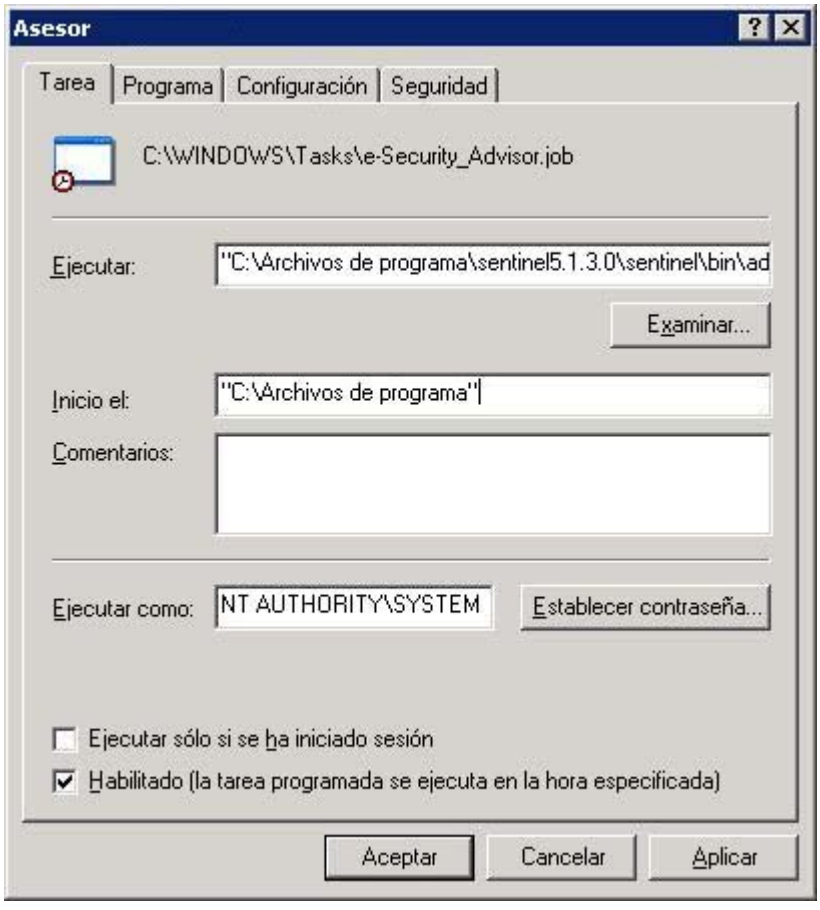
Cambio de la contraseña de administrador de Sentinel

1. Utilice el sistema operativo Windows para cambiar la contraseña.

Cambio de la contraseña de Administrador de la base de datos de Sentinel

Cambio de la contraseña de administrador de BD de Sentinel

1. Utilice el sistema operativo Windows para cambiar la contraseña.
2. Si está ejecutando una tarea de SDM programada (p. ej., para añadir o archivar particiones), deberá actualizar la propiedad “Ejecutar como” (*Panel de control > Tareas programadas > hacer clic con el botón derecho del ratón en Propiedades*).

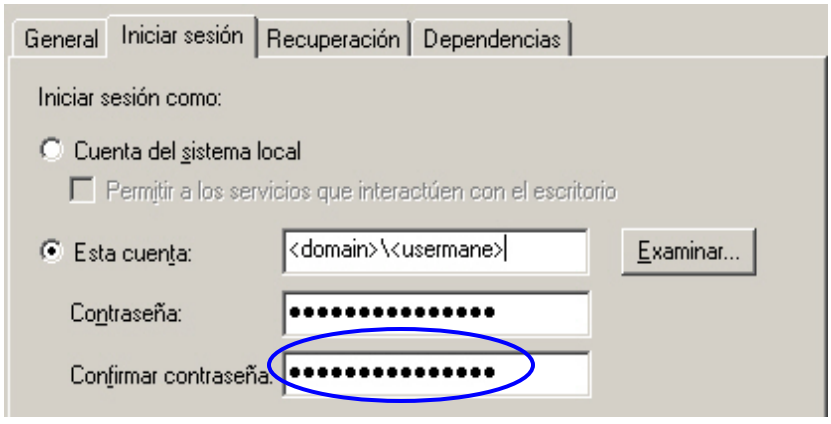


3. Haga clic en *Establecer contraseña*. Introduzca la contraseña nueva dos veces y haga clic en *Aceptar*. Haga clic en *Aplicar* y, a continuación, en *Aceptar*.

Cambio de la contraseña de Administrador de la base de datos de la aplicación de Sentinel

Cambio de la contraseña de administrador de BD de la aplicación de Sentinel

1. Utilice el sistema operativo Windows para cambiar la contraseña.
2. En la máquina DAS, abra Servicios de Windows (*Panel de control* > *Herramientas administrativas* > *Servicios*).
3. Haga clic con el botón derecho del ratón en *Sentinel* > *Propiedades*. Haga clic en la pestaña *Iniciar sesión* y actualice la contraseña en *Iniciar sesión como*. Haga clic en *Aplicar* y, a continuación, en *Aceptar*.



4. Si tiene el asesor instalado, deberá actualizar la propiedad “Ejecutar como” (*Panel de control > Tareas programadas > hacer clic con el botón derecho del ratón en Propiedades*) en las tareas programadas del asesor.
5. Haga clic en *Establecer contraseña*. Introduzca la contraseña nueva dos veces y haga clic en *Aceptar*. Haga clic en *Aplicar* y, a continuación, en *Aceptar*.

Cambio de la contraseña de usuario de informes de Sentinel

Cambio de la contraseña de usuario de informes de Sentinel

1. Utilice el sistema operativo Windows para cambiar la contraseña.

11 Vistas de la base de datos de Sentinel para Oracle

NOTA: El término Agente puede intercambiarse con Recopilador. En adelante, los agentes se denominarán recopiladores.

En este capítulo se enumeran las vistas del esquema de Sentinel para Oracle. Las vistas proporcionan información para desarrollar sus propios informes (Crystal Reports).

Vistas

ADV_ALERT_CVE_RPT_V

La vista hace referencia a la tabla ADV_ALERT_CVE que almacena el número de identificación de alerta del Asesor.

Nombre de la columna	Tipo de datos	Comentario
ALERT_ID	number	Identificador de anotación - número de secuencia
CVE	varchar2	
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acción
MODIFIED_BY	number	ID del usuario que realiza la acción

ADV_ALERT_PRODUCT_RPT_V

La vista hace referencia a la tabla ADV_ALERT_PRODUCT que almacena información de productos del Asesor como, por ejemplo, número de identificación de Service Pack, versión y fecha de creación.

Nombre de la columna	Tipo de datos	Comentario
ALERT_ID	number	Identificador de anotación - número de secuencia
SERVICE_PACK_ID	number	
VENDOR	varchar2	
PRODUCT	varchar2	
VERSION	varchar2	Contiene el número de versión
SERVICE_PACK	varchar2	
PRIMARY_FLAG	number	
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acción
MODIFIED_BY	number	ID del usuario que realiza la acción

ADV_ALERT_RPT_V

La vista hace referencia a la tabla ADV_ALERT que almacena información de alertas del Asesor como, por ejemplo, nombre, tipo de amenaza y fecha de publicación.

Nombre de la columna	Tipo de datos	Comentario
ALERT_ID	number	Identificador de anotación - número de secuencia
VERSION	number	Contiene el número de versión
TEMPLATE_ID	number	
TEMPLATE_NAME	varchar2	
THREAT_CATEGORY_NAME	varchar2	
THREAT_TYPE_NAME	varchar2	
HEADLINE	clob	
FIRST_PUBLISHED	date	
LAST_PUBLISHED	date	
STATUS	varchar2	
URGENCY_ID	number	
CREDIBILITY_ID	number	
SEVERITY_ID	number	
SUMMARY	clob	
LEGAL_DISCLAIMER	clob	
COPYRIGHT	varchar2	
BEGIN_EFFECTIVE_DATE	date	
END_EFFECTIVE_DATE	date	
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acción
MODIFIED_BY	number	ID del usuario que realiza la acción

ADV_ATTACK_ALERT_RPT_V

La vista hace referencia a la tabla ADV_ATTACK_ALERT que almacena información de ataques del Asesor como, por ejemplo, nombre, tipo de amenaza y fecha de publicación.

Nombre de la columna	Tipo de datos	Comentario
ATTACK_ID	number	
ALERT_ID	number	
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acción
MODIFIED_BY	number	ID del usuario que realiza la acción

ADV_ATTACK_CVE_RPT_V

La vista hace referencia a la tabla ADV_ATTACK_CVE que almacena información de CVE del Asesor.

Nombre de la columna	Tipo de datos	Comentario
ATTACK_ID	number	
CVE	varchar2	
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acción
MODIFIED_BY	number	ID del usuario que realiza la acción

ADV_ATTACK_MAP_RPT_V

La vista hace referencia a la tabla ADV_ATTACK_MAP que almacena información de asignación del Asesor.

Nombre de la columna	Tipo de datos	Comentario
ATTACK_KEY	number	
ATTACK_ID	number	
SERVICE_PACK_ID	number	
ATTACK_NAME	varchar2	
ATTACK_CODE	varchar2	
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acción
MODIFIED_by	number	ID del usuario que realiza la acción

ADV_ATTACK_PLUGIN_RPT_V

La vista hace referencia a la tabla ADV_ATTACK_PLUGIN que almacena información de módulos auxiliares (plug-ins) del Asesor.

Nombre de la columna	Tipo de datos	Comentario
PLUGIN_KEY	number	
ATTACK_ID	number	
SERVICE_PACK_ID	number	
PLUGIN_ID	varchar2	
PLUGIN_NAME	varchar2	
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acción
MODIFIED_BY	number	ID del usuario que realiza la acción

ADV_ATTACK_RPT_V

La vista hace referencia a la tabla ADV_ATTACK que almacena información de ataques del Asesor.

Nombre de la columna	Tipo de datos	Comentario
ALERT_ID	number	
TRUSECURE_ATTACK_NAME	number	
FEED_DATE_CREATED	date	
FEED_DATE_UPDATED	date	
ATTACK_CATEGORY	varchar2	
URGENCY_ID	number	
SEVERITY_ID	number	
LOCAL	number	
REMOTE	number	
BEGIN_EFFECTIVE_DATE	date	
END_EFFECTIVE_DATE	date	
DESCRIPTION	clob	
SCENARIO	clob	
IMPACT	clob	
SAFEGUARDS	clob	
PATCHES	clob	
FALSE_POSITIVES	clob	
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acción
MODIFIED_BY	number	ID del usuario que realiza la acción

ADV_CREDIBILITY_RPT_V

Ver la tabla de referencias ADV_CREDIBILITY que almacena información de credibilidad del Asesor.

Nombre de la columna	Tipo de datos	Comentario
CREDIBILITY_ID	number	
CREDIBILITY_RATING	varchar2	
CREDIBILITY_EXPLANATION	varchar2	
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acción
MODIFIED_BY	number	ID del usuario que realiza la acción

ADV_FEED_RPT_V

La vista hace referencia a la tabla ADV_FEED que almacena información de los datos del Asesor como, por ejemplo, el nombre y la fecha.

Nombre de la columna	Tipo de datos	Comentario
FEED_NAME	varchar2	
FEED_FILE	varchar2	
BEGIN_DATE	date	
END_DATE	date	
FEED_INSERT	number	
FEED_UPDATE	number	
FEED_EXPIRE	number	

ADV_PRODUCT_RPT_V

La vista hace referencia a la tabla ADV_PRODUCT que almacena información de productos del Asesor como, por ejemplo, proveedor e ID de producto.

Nombre de la columna	Tipo de datos	Comentario
PRODUCT_ID	number	
VENDOR_ID	number	
PRODUCT_CATEGORY_ID	number	
PRODUCT_CATEGORY_NAME	varchar2	
PRODUCT_TYPE-ID	number	
PRODUCT_TYPE_NAME	varchar2	
PRODUCT_NAME	varchar2	
PRODUCT_DESCRIPTION	varchar2	
FEED_DATE_CREATED	date	
FEED_DATE_UPDATED	date	
ACTIVE_FLAG	number	
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acción
MODIFIED_BY	number	ID del usuario que realiza la acción

ADV_PRODUCT_SERVICE_PACK_RPT_V

La vista hace referencia a la tabla ADV_PRODUCT_SERVICE _PACK que almacena información de Service Pack del asesor como, por ejemplo, nombre del Service Pack, ID de la versión y fecha.

Nombre de la columna	Tipo de datos	Comentario
SERVICE_PACK_ID	number	
VERSION_ID	number	Contiene el número de ID de la versión
SERVICE_PACK_NAME	varchar2	
FEED_DATE_CREATED	date	
FEED_DATE_UPDATED	date	
ACTIVE_FLAG	number	
BEGIN_EFFECTIVE_DATE	date	

Nombre de la columna	Tipo de datos	Comentario
END_EFFECTIVE_DATE	date	
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acción
MODIFIED_BY	number	ID del usuario que realiza la acción

ADV_PRODUCT_VERSION_RPT_V

La vista hace referencia a la tabla ADV_PRODUCT_VERSION que almacena información de la versión de productos del Asesor como, por ejemplo, nombre de la versión, producto e ID de la versión.

Nombre de la columna	Tipo de datos	Comentario
VERSION_ID	number	Contiene el número de ID de la versión
PRODUCT_ID	number	
VERSION_NAME	varchar2	
FEED_DATE_CREATED	date	
FEED_DATE_UPDATED	date	
ACTIVE_FLAG	number	
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	number	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acción
MODIFIED_BY	number	ID del usuario que realiza la acción

ADV_SEVERITY_RPT_V

La vista hace referencia a la tabla ADV_SEVERITY que almacena información de valoraciones de gravedad del Asesor.

Nombre de la columna	Tipo de datos	Comentario
SEVERITY_ID	number	
SEVERITY_RATING	varchar2	
SEVERITY_EXPLANATION	varchar2	
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acción
MODIFIED_BY	number	ID del usuario que realiza la acción

ADV_SUBALERT_RPT_V

La vista hace referencia a la tabla ADV_SUBALERT.

Nombre de la columna	Tipo de datos	Comentario
ALERT_ID	number	
SUBALERT_ID	number	
CHANGED_SECTIONS	varchar2	
VARIANTS	clob	
VIRUS_NAME	clob	
DESCRIPTION	clob	
IMPACT	clob	

Nombre de la columna	Tipo de datos	Comentario
WARNING_INDICATORS	clob	
TECHNICAL_INFO	clob	
TRUSECURE_COMMENTS	clob	
VENDOR_ANNOUNCEMENTS	clob	
SAFEGUARDS	clob	
PATCHES_SOFTWARE	clob	
ALERT_HISTORY	clob	
BACKGROUND_INFO	clob	
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acción
MODIFIED_BY	number	ID del usuario que realiza la acción

ADV_URGENCY_RPT_V

La vista hace referencia a la tabla ADV_URGENCY.

Nombre de la columna	Tipo de datos	Comentario
URGENCY_ID	number	
URGENCY_RATING	varchar2	
URGENCY_EXPLANATION	varchar2	
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acción
MODIFIED_BY	number	ID del usuario que realiza la acción

ADV_VENDOR_RPT_V

La vista hace referencia a la tabla ADV_VENDOR que almacena información de direcciones del Asesor.

Nombre de la columna	Tipo de datos	Comentario
VENDOR_ID	number	
VENDOR_NAME	varchar2	
CONTACT_PERSON	varchar2	
ADDRESS_LINE_1	varchar2	
ADDRESS_LINE_2	varchar2	
ADDRESS_LINE_3	varchar2	
ADDRESS_LINE_4	varchar2	
CITY	varchar2	
STATE	varchar2	
COUNTRY	varchar2	
ZIP_CODE	varchar2	
URL	varchar2	
PHONE	varchar2	
FAX	varchar2	
EMAIL	varchar2	
PAGER	varchar2	
FEED_DATE_CREATED	date	

Nombre de la columna	Tipo de datos	Comentario
FEED_DATE_UPDATED	date	
ACTIVE_FLAG	number	
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acción
MODIFIED_BY	number	ID del usuario que realiza la acción

ADV_VULN_PRODUCT_RPT_V

La vista hace referencia a la tabla ADV_VULN_PRODUCT que almacena la ID de ataque de vulnerabilidad y la ID de Service Pack del Asesor.

Nombre de la columna	Tipo de datos	Comentario
ATTACK_ID	number	
SERVICE_PACK_ID	number	
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acción
MODIFIED_BY	number	ID del usuario que realiza la acción

ANNOTATIONS_RPT_V

La vista hace referencia a la tabla ANNOTATIONS que almacena documentación o notas que pueden asociarse con objetos en el sistema Sentinel como, por ejemplo, incidencias.

Nombre de la columna	Tipo de datos	Comentario
ANN_ID	NUMBER	Identificador de anotación - número de secuencia
TEXT	VARCHAR2(4000)	Documentación o notas
DATE_CREATED	DATE	Fecha de inserción
DATE_MODIFIED	DATE	Fecha de la última actualización
MODIFIED_BY	NUMBER	ID del usuario que realizó la última actualización
CREATED_BY	NUMBER	ID del usuario que realiza la inserción
ACTION	Varchar2(255)	Acción

ASSET_CTGRY_RPT_V

La vista hace referencia a la tabla ASSET_CTGRY que almacena información acerca de las categorías de activos (p. ej., hardware, software, SO, base de datos, etc.).

Nombre de la columna	Tipo de datos	Comentario
ASSET_CATAGORY_ID	number	Identificador de categoría de activo
ASSET_CATAGORY_NAME	varchar2(100)	Nombre de categoría de activo
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

ASSET_HOSTNAME_RPT_V

La vista hace referencia a la tabla ASSET_HOSTNAME que almacena información acerca de nombres de host alternativos de los activos.

Nombre de la columna	Tipo de datos	Comentario
ASSET_HOSTNAME_ID	Varchar2(36)	Identificador de nombre de host alternativo del activo
PHYSICAL_ASSET_ID	varchar2(36)	Identificador de activo físico
HOST_NAME	Varchar2(255)	Nombre del host
CUSTOMER_ID	number	Identificador de cliente
DATE_CREATED	date	Fecha de la última actualización
DATE_MODIFIED	date	ID del usuario que realizó la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

ASSET_IP_RPT_V

La vista hace referencia a la tabla ASSET_IP que almacena información acerca de direcciones IP alternativas de los activos.

Nombre de la columna	Tipo de datos	Comentario
ASSET_IP_ID	Varchar2(36)	Identificador de IP alternativo de activo
PHYSICAL_ASSET_ID	varchar2(36)	Identificador de activo físico
IP_ADDRESS	number	Dirección IP de activo
CUSTOMER_ID	number	Identificador de cliente
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

ASSET_LOCATION_RPT_V

La vista hace referencia a la tabla ASSET_LOC que almacena información acerca de las ubicaciones de activos.

Nombre de la columna	Tipo de datos	Comentario
LOCATION_ID	number	Identificación de ubicación
CUSTOMER_ID	number	Identificador de cliente
BUILDING_NAME	varchar2(255)	Nombre de edificio
ADDRESS_LINE_1	varchar2(255)	Línea de dirección 1
ADDRESS_LINE_2	varchar2(255)	Línea de dirección 2
CITY	varchar2(100)	Ciudad
STATE	varchar2(100)	Estado
COUNTRY	varchar2(100)	País
ZIP_CODE	varchar2(50)	Código postal
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

ASSET_RPT_V

La vista hace referencia a la tabla ASSET que almacena información acerca de los activos físicos y de software.

Nombre de la columna	Tipo de datos	Comentario
ASSET_ID	varchar2(36)	Identificador de activo
CUSTOMER_ID	number	Identificador de cliente
ASSET_NAME	varchar2(255)	Nombre de activo
PHYSICAL_ASSET_ID	varchar2(36)	Identificador de activo físico
PRDT_ID	number	Identificador de producto
ASSET_CATEGORY_ID	number	Identificador de categoría de activo
ENVIRONMENT_IDENTITY_CD	varchar2(5)	Código de identificación de entorno
PHYSICAL_ASSET_IND	number(1)	Indicador de activo físico
ASSET_VALUE_CODE	varchar2(5)	Código de valor de activo
CRITICALITY_CODE	varchar2(5)	Código de importancia de activo
SENSITIVITY_CODE	varchar2(5)	Código de sensibilidad de activo
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

ASSET_VALUE_RPT_V

La vista hace referencia a la tabla ASSET_VAL_LKUP que almacena información acerca del valor de activos.

Nombre de la columna	Tipo de datos	Comentario
ASSET_VALUE_CODE	varchar2(5)	Código de valor de activo
ASSET_VALUE_NAME	varchar2(50)	Nombre de valor de activo
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

ASSET_X_ENTITY_X_ROLE_RPT_V

La vista hace referencia a la tabla ASSET_X_ENTITY_X_ROLE que asocia a una persona u organización con un activo.

Nombre de la columna	Tipo de datos	Comentario
PERSON_ID	varchar2(36)	Identificador de persona
ORGANIZATION_ID	varchar2(36)	Identificador de organización
ROLE_CODE	varchar2(5)	Código de función
ASSET_ID	varchar2(36)	Identificador de activo
ENTITY_TYPE_CODE	varchar2(5)	Código de tipo de entidad
PERSON_ROLE_SEQUENCE	number	Orden de las personas bajo una función en particular
DATE_CREATED	date	Fecha de inserción

Nombre de la columna	Tipo de datos	Comentario
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	Usuario de última actualización

ASSOCIATIONS_RPT_V

La vista hace referencia a la tabla ASSOCIATIONS que asocia usuarios con incidencias, incidencias con anotaciones, etc.

Nombre de la columna	Tipo de datos	Comentario
TABLE1	VARCHAR2(64)	Nombre de tabla 1. Por ejemplo, incidencias.
ID1	VARCHAR2(36)	ID1. Por ejemplo, ID de incidencias.
TABLE2	VARCHAR2(64)	Nombre de tabla 2. Por ejemplo, usuarios.
ID2	VARCHAR2(36)	ID2. Por ejemplo, ID de usuario.
DATE_CREATED	DATE	Fecha de inserción
DATE_MODIFIED	DATE	Fecha de la última actualización
CREATED_BY	NUMBER	ID del usuario que realiza la inserción
MODIFIED_BY	NUMBER	ID del usuario que realizó la última actualización

ATTACHMENTS_RPT_V

La vista hace referencia a la tabla ATTACHMENTS que almacena información de datos adjuntos.

Nombre de la columna	Tipo de datos	Comentario
ATTACHMENT_ID	number	Identificador del adjunto
NAME	varchar2(255)	Nombre del adjunto
SOURCE_REFERENCE	varchar2(64)	Referencia de origen
TYPE	varchar2(32)	Tipo de adjunto
SUB_TYPE	varchar2(32)	Subtipo de adjunto
FILE_EXTENSION	varchar2(32)	Extensión de archivo
ATTACHMENT_DESCRIPTION	varchar2(255)	Descripción del adjunto
DATA	clob	Datos del adjunto
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

CONFIGS_RPT_V

La vista hace referencia a la tabla CONFIGS que almacena información general de configuración de la aplicación.

Nombre de la columna	Tipo de datos	Comentario
USR_ID	VARCHAR2(32)	Nombre de usuario
APPLICATION	VARCHAR2(255)	Identificador de la aplicación
UNIT	VARCHAR2(64)	Unidad de la aplicación
VALUE	VARCHAR2(255)	Valor del texto, si corresponde
DATA	CLOB	Datos XML
DATE_CREATED	DATE	Fecha de inserción
DATE_MODIFIED	DATE	Fecha de la última actualización
CREATED_BY	NUMBER	ID del usuario que realiza la inserción
MODIFIED_BY	NUMBER	ID del usuario que realizó la última actualización

CONTACTS_RPT_V

La vista hace referencia a la tabla CONTACTS que almacena información de contactos.

Nombre de la columna	Tipo de datos	Comentario
CNT_ID	NUMBER	ID del contacto - Número de secuencia
FIRST_NAME	VARCHAR2(20)	Nombre del contacto
LAST_NAME	VARCHAR2(30)	Apellido del contacto
TITLE	VARCHAR2(128)	Cargo del contacto
DEPARTMENT	VARCHAR2(128)	Departamento
PHONE	VARCHAR2(64)	Teléfono del contacto
EMAIL	VARCHAR2(255)	Correo electrónico del contacto
PAGER	VARCHAR2(64)	Buscapersonas del contacto
CELL	VARCHAR2(64)	Móvil del contacto
DATE_CREATED	DATE	Fecha de inserción
DATE_MODIFIED	DATE	Fecha de la última actualización
CREATED_BY	NUMBER	ID del usuario que realiza la inserción
MODIFIED_BY	NUMBER	ID del usuario que realizó la última actualización

CORRELATED_EVENTS_RPT_V

La vista hace referencia a las tablas CORRELATED_EVENTS_* que almacenan información de eventos correlacionados.

Nombre de la columna	Tipo de datos	Comentario
PARENT_EVT_ID	varchar2	UUID (Identificador exclusivo universal de eventos) del evento padre
CHILD_EVT_ID	varchar2	UUID (Identificador exclusivo universal de eventos) de evento hijo
PARENT_EVT_TIME	DATE	Fecha y hora del evento padre
CHILD_EVT_TIME	DATE	Fecha y hora del evento hijo
DATE_CREATED	DATE	Fecha de inserción creada por DAS
DATE_MODIFIED	DATE	Fecha de la última actualización
CREATED_BY	NUMBER	ID del usuario que realiza la inserción
MODIFIED_BY	NUMBER	ID del usuario que realizó la última actualización

CORRELATED_EVENTS_RPT_V1

La vista contiene eventos correlacionados actuales e históricos (eventos correlacionados importados de archivos).

Nombre de la columna	Tipo de datos	Comentario
PARENT_EVT_ID	varchar2	UUID (Identificador exclusivo universal de eventos) del evento padre
CHILD_EVT_ID	varchar2	UUID (Identificador exclusivo universal de eventos) de evento hijo
PARENT_EVT_TIME	DATE	Fecha y hora del evento padre
CHILD_EVT_TIME	DATE	Fecha y hora del evento hijo
DATE_CREATED	DATE	Fecha de inserción creada por DAS
DATE_MODIFIED	DATE	Fecha de la última actualización
CREATED_BY	NUMBER	ID del usuario que realiza la inserción
MODIFIED_BY	NUMBER	ID del usuario que realizó la última actualización

CRITICALITY_RPT_V

La vista hace referencia a la tabla CRIT_LKUP que contiene información acerca de la importancia de activos.

Nombre de la columna	Tipo de datos	Comentario
CRITICALITY_CODE	varchar2(5)	Código de importancia de activo
CRITICALITY_NAME	varchar2(50)	Nombre de importancia de activo
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acción
MODIFIED_BY	number	ID del usuario que realiza la acción

CUST_RPT_V

La vista hace referencia a la tabla CUST que almacena información de clientes para MSSPs.

Nombre de la columna	Tipo de datos	Comentario
CUSTOMER_ID	number	Identificador de cliente
CUSTOMER_NAME	varchar2(255)	Nombre del cliente
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

ENTITY_TYPE_RPT_V

La vista hace referencia a la tabla ENTITY_TYP que almacena información acerca de los tipos de entidades (personas, organizaciones).

Nombre de la columna	Tipo de datos	Comentario
ENTITY_TYPE_CODE	varchar2(5)	Código de tipo de entidad
ENTITY_TYPE_NAME	varchar2(50)	Nombre de tipo de entidad
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acciónusuario de inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

ENV_IDENTITY_RPT_V

La vista hace referencia a la tabla ENV_IDENTITY_LKUP que almacena información acerca de la identidad del entorno de activos.

Nombre de la columna	Tipo de datos	Comentario
ENVIRONMENT_IDENTITY_CODE	varchar2(5)	Código de identidad del entorno
ENVIRONMENT_IDENTITY_NAME	varchar2(255)	Nombre de identidad del entorno
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acciónusuario de inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

ESEC_DISPLAY_RPT_V

La vista hace referencia a la tabla ESEC_DISPLAY que almacena las propiedades de los objetos que se pueden mostrar. Se utiliza actualmente para renombrar las meta-etiquetas. Se utiliza con la Configuración de eventos (Relevancia empresarial).

Nombre de la columna	Tipo de datos	Comentario
DISPLAY_OBJECT	VARCHAR2(32)	El objeto padre de la propiedad
TAG	VARCHAR2(32)	El nombre de etiqueta nativa de la propiedad
LABEL	VARCHAR2(32)	La cadena de visualización de la etiqueta
POSITION	NUMBER	Posición de la etiqueta en la pantalla
WIDTH	NUMBER	El ancho de la columna
ALIGNMENT	NUMBER	La alineación horizontal
FORMAT	NUMBER	El formato enumerado para mostrar la propiedad
ENABLED	VARCHAR2(1)	Indica si se muestra la etiqueta.
TYPE	NUMBER	Indica el tipo de dato de la etiqueta. 1 = string 2 = ulong 3 = date 4 = uuid 5 = ipv4

Nombre de la columna	Tipo de datos	Comentario
DESCRIPTION	VARCHAR2(255)	Descripción textual de la etiqueta
DATE_CREATED	DATE	Fecha de inserción
DATE_MODIFIED	DATE	Fecha de la última actualización
CREATED_BY	NUMBER	ID del usuario que realiza la inserción
MODIFIED_BY	NUMBER	ID del usuario que realizó la última actualización
REF_CONFIG	VARCHAR2(4000)	Configuración de datos referenciales

ESEC_PORT_REFERENCE_RPT_V

La vista hace referencia a la tabla ESEC_PORT_REFERENCE que almacena números de puerto estándares de la industria asignados.

Nombre de la columna	Tipo de datos	Comentario
PORT_NUMBER	NUMBER	Según http://www.iana.org/assignments/port-numbers, la representación numérica del puerto. Este número de puerto comúnmente se asocia con el nivel de Protocolo de transporte en la pila TCP/IP.
PROTOCOL_NUMBER	NUMBER	Según http://www.iana.org/assignments/protocol-numbers, los identificadores numéricos utilizados para representar protocolos que están encapsulados en un paquete IP.
PORT_KEYWORD	VARCHAR2(64)	Según http://www.iana.org/assignments/port-numbers, la palabra clave que representa al puerto.
PORT_DESCRIPTION	VARCHAR2(512)	Descripción del puerto
DATE_CREATED	DATE	Fecha de inserción
DATE_MODIFIED	DATE	Fecha de la última actualización
CREATED_BY	NUMBER	ID del usuario que realiza la inserción.
MODIFIED_BY	NUMBER	ID de usuario de la última modificación

ESEC_PROTOCOL_REFERENCE_RPT_V

La vista hace referencia a la tabla ESEC_PROTOCOL_REFERENCE que almacena números de protocolo estándares de la industria asignados.

Nombre de la columna	Tipo de datos	Comentario
PROTOCOL_NUMBER	NUMBER	Según http://www.iana.org/assignments/protocol-numbers, los identificadores numéricos utilizados para representar protocolos que están encapsulados en un paquete IP.
PROTOCOL_KEYWORD	VARCHAR2(64)	Según http://www.iana.org/assignments/protocol-numbers, la palabra clave utilizada para representar protocolos que están encapsulados en un paquete IP.
PROTOCOL_DESCRIPTION	VARCHAR2(512)	Descripción del protocolo del paquete IP
DATE_CREATED	DATE	Fecha de inserción
DATE_MODIFIED	DATE	Fecha de la última actualización
CREATED_BY	NUMBER	ID del usuario que realiza la inserción.
MODIFIED_BY	NUMBER	ID del usuario que realizó la última actualización

ESEC_SEQUENCE_RPT_V

La vista hace referencia a la tabla ESEC_SEQUENCE que se utiliza para generar números de secuencia de clave principal para las tablas de Sentinel.

Nombre de la columna	Tipo de datos	Comentario
TABLE_NAME	VARCHAR2(32)	Nombre de la tabla
COLUMN_NAME	VARCHAR2(32)	Nombre de la columna
SEED	NUMBER	Valor actual del campo de clave principal
DATE_CREATED	DATE	Fecha de inserción
DATE_MODIFIED	DATE	Fecha de la última actualización
CREATED_BY	NUMBER	ID del usuario que realiza la inserción
MODIFIED_BY	NUMBER	ID del usuario que realizó la última actualización

EVENTS_ALL_RPT_V (provisto para fines de compatibilidad con versiones anteriores)

La vista contiene eventos actuales e históricos (eventos importados de archivos).

Nombre de la columna	Tipo de datos	Comentario
EVENT_ID	varchar2	Identificador de evento
RESOURCE_NAME	varchar2(255)	Nombre de recurso
SUB_RESOURCE	varchar2(255)	Nombre de subrecurso
SEVERITY	number	Gravedad del evento
EVENT_PARSE_TIME	date	Fecha y hora del evento
EVENT_DATE_TIME	date	Fecha y hora del evento
BASE_MESSAGE	varchar2(4000)	Mensaje de base
EVENT_NAME	varchar2(255)	Nombre del evento según lo informado por el sensor.
EVENT_TIME	varchar2(255)	Fecha y hora del evento según lo informado por el sensor.
SENSOR_NAME	varchar2(255)	Nombre del sensor
SENSOR_TYPE	varchar2(5)	Tipo de sensor: H – de host N – de red V – virus O – otro
PROTOCOL	varchar2(255)	Nombre de protocolo
SOURCE-IP	number	Dirección IP de origen en formato numérico
SOURCE_HOST_NAME	varchar2(255)	Nombre de host de origen
SOURCE_PORT	varchar2(32)	Puerto de origen
DESTINATION_IP	number	Dirección IP de destino en formato numérico
DESTINATION_HOST_NAME	varchar2(255)	Nombre de host de destino
DESTINATION_PORT	varchar2(32)	Puerto de destino
SOURCE_USER_NAME	varchar2(255)	Nombre de usuario de origen
DESTINATION_USER_NAME	varchar2(255)	Nombre de usuario de destino
FILE_NAME	varchar2(1000)	Nombre de archivo
EXTENDED_INFO	varchar2(1000)	Información ampliada
REPORT_NAME	varchar2(255)	Nombre del informador
PRODUCT_NAME	varchar2(255)	Nombre del producto de generación de informes
CUSTOM_TAG_1	varchar2(255)	Etiqueta de cliente 1
CUSTOM_TAG_2	varchar2(255)	Etiqueta de cliente 2
CUSTOM_TAG_3	number	Etiqueta de cliente 3
RESERVED_TAG_1	VARCHAR2(255)	Etiqueta reservada 1 Reservado para uso futuro de Novell. Este campo se utiliza para información de Asesor relativa a descripciones de ataques.

Nombre de la columna	Tipo de datos	Comentario
RESERVED_TAG_2	varchar2(255)	Reservado para uso futuro de Novell. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RESERVED_TAG_3	number	Reservado para uso futuro de Novell. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
SOURCE_UUID	varchar(36)	UUID de origen
PORT	varchar(64)	Puerto del recopilador
AGENT	varchar2(64)	Nombre del recopilador
VULNERABILITY_RATING	number	Puntuación de vulnerabilidad
CRITICALITY_RATING	number	Valoración de la importancia
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acción
MODIFIED_BY	number	ID del usuario que realiza la acción
RV01 - 10	NUMBER	Valor reservado de 1 a 10 Reservado para uso futuro de Novell. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV11 - 20	DATE	Valor reservado de 11 a 20 Reservado para uso futuro de Novell. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV21 - 25	varchar2	Valor reservado de 21 a 25 Reservado para uso futuro de Novell para almacenar UUID. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV26 - 31	VARCHAR2(255)	Valor reservado de 26 a 31 Reservado para uso futuro de Novell. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.

Nombre de la columna	Tipo de datos	Comentario
RV32	VARCHAR2(255)	Valor reservado 32 Reservado para DeviceCategory El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV33	VARCHAR2(255)	Valor reservado 33 Reservado para EventContex El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV34	VARCHAR2(255)	Valor reservado 34 Reservado para SourceThreatLevel El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV35	VARCHAR2(255)	Valor reservado 35 Reservado para SourceUserContext. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV36	VARCHAR2(255)	Valor reservado 36 Reservado para DataContext. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV37	VARCHAR2(255)	Valor reservado 37 Reservado para SourceFunction. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV38	VARCHAR2(255)	Valor reservado 38 Reservado para SourceOperationalContext. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.

Nombre de la columna	Tipo de datos	Comentario
RV39	VARCHAR2(255)	Valor reservado 39 Reservado para MSSPCustomerName. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV40 - 43	VARCHAR2(255)	Valor reservado de 40 a 43 Reservado para uso futuro de Novell. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV44	VARCHAR2(255)	Valor reservado 44 Reservado para DestinationThreatLevel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV45	VARCHAR2(255)	Valor reservado 45 Reservado para DestinationUserContext. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV46	VARCHAR2(255)	Valor reservado 46 Reservado para VirusStatus. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV47	VARCHAR2(255)	Valor reservado 47 Reservado para uso futuro de Novell. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV48	VARCHAR2(255)	Valor reservado 48 Reservado para DestinationOperationalContext. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.

Nombre de la columna	Tipo de datos	Comentario
RV49	VARCHAR2(255)	Valor reservado 49 Reservado para uso futuro de Novell. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV50	VARCHAR2(255)	Taxonomía nivel 1
RV51	VARCHAR2(255)	Taxonomía nivel 2
RV52	VARCHAR2(255)	Taxonomía nivel 3
RV53	VARCHAR2(255)	Taxonomía nivel 4
CV01 - 10	NUMBER	Valor personalizado de 1 a 10 Reservado para uso del cliente, normalmente para asociación de datos comerciales relevantes
CV11 - 20	DATE	Valor personalizado de 11 a 20 Reservado para uso del cliente, normalmente para asociación de datos comerciales relevantes
CV21 - 100	VARCHAR2(255)	Valor personalizado de 21 a 100 Reservado para uso del cliente, normalmente para asociación de datos comerciales relevantes

EVENTS_ALL_RPT_V1 (provisto para fines de compatibilidad con versiones anteriores)

La vista contiene eventos actuales. Tiene las mismas columnas que EVENT_ALL_RPT_V.

EVENTS_RPT_V (provisto para fines de compatibilidad con versiones anteriores)

La vista contiene eventos actuales e históricos. Tiene las mismas columnas que EVENT_ALL_RPT_V.

EVENTS_RPT_V1 (provisto para fines de compatibilidad con versiones anteriores)

La vista contiene eventos actuales. Tiene las mismas columnas que EVENT_ALL_RPT_V.

EVENTS_RPT_V2 (todos los informes nuevos de Sentinel 5 deben utilizar esta vista)

La vista contiene eventos actuales y eventos históricos.

Nombre de la columna	Tipo de datos	Comentario
EVENT_ID	varchar2	Identificador de evento
RESOURCE_NAME	varchar2(255)	Nombre de recurso
SUB_RESOURCE	varchar2(255)	Nombre de subrecurso
SEVERITY	number	Gravedad del evento
EVENT_PARSE_TIME	date	Fecha y hora del evento
EVENT_DATETIME	date	Fecha y hora del evento
BASE_MESSAGE	varchar2(4000)	Mensaje de base
EVENT_NAME	varchar2(255)	Nombre del evento según lo informado por el sensor.
EVENT_TIME	varchar2(255)	Fecha y hora del evento según lo informado por el sensor.
TAXONOMY_ID	number	Identificador de taxonomía
PROTOCOL_ID	number	Identificador de protocolo
AGENT_ID	number	Identificador de recopilador
SOURCE_IP	number	Dirección IP de origen en formato numérico
SOURCE_HOST_NAME	varchar2(255)	Nombre de host de origen
SOURCE_PORT	varchar2(32)	Puerto de origen
DESTINATION_IP	number	Dirección IP de destino en formato numérico
DESTINATION_HOST_NAME	varchar2(255)	Nombre de host de destino
DESTINATION_PORT	varchar2(32)	Puerto de destino
SOURCE_USER_NAME	varchar2(255)	Nombre de usuario de origen
DESTINATION_USER_NAME	varchar2(255)	Nombre de usuario de destino
FILE_NAME	varchar2(1000)	Nombre de archivo
EXTENDED_INFO	varchar2(1000)	Información ampliada
CUSTOM_TAG_1	varchar2(255)	Etiqueta de cliente 1
CUSTOM_TAG 2	varchar2(255)	Etiqueta de cliente 2
CUSTOM_TAG 3	number	Etiqueta de cliente 3
RESERVED_TAG_1	VARCHAR2(255)	Etiqueta reservada 1 Reservado para uso futuro de Novell. Este campo se utiliza para información de Asesor relativa a descripciones de ataques.
RESERVED_TAG_2	varchar2(255)	Reservado para uso futuro de Novell. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.

Nombre de la columna	Tipo de datos	Comentario
RESERVED_TAG_3	number	Reservado para uso futuro de Novell. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
VULNERABILITY_RATING	number	Puntuación de vulnerabilidad
CRITICALITY_RATING	number	Valoración de la importancia
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización
RV01 - 10	NUMBER	Valor reservado de 1 a 10 Reservado para uso futuro de Novell. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV11 - 20	DATE	Valor reservado de 1 a 31 Reservado para uso futuro de Novell. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV21 - 25	varchar2	Valor reservado de 21 a 25 Reservado para uso futuro de Novell para almacenar UUID. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV26 31	VARCHAR2(255)	Valor reservado de 26 a 31 Reservado para uso futuro de Novell. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV33	VARCHAR2(255)	Valor reservado 33 Reservado para EventContex El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.

Nombre de la columna	Tipo de datos	Comentario
RV34	VARCHAR2(255)	Valor reservado 34 Reservado para SourceThreatLevel El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV35	VARCHAR2(255)	Valor reservado 35 Reservado para SourceUserContext. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV36	VARCHAR2(255)	Valor reservado 36 Reservado para DataContext. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV37	VARCHAR2(255)	Valor reservado 37 Reservado para SourceFunction. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV38	VARCHAR2(255)	Valor reservado 38 Reservado para SourceOperationalContext. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV40 - 43	VARCHAR2(255)	Valor reservado de 40 a 43 Reservado para uso futuro de Novell. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV44	VARCHAR2(255)	Valor reservado 44 Reservado para DestinationThreatLevel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.

Nombre de la columna	Tipo de datos	Comentario
RV45	VARCHAR2(255)	Valor reservado 45 Reservado para DestinationUserContext. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV46	VARCHAR2(255)	Valor reservado 46 Reservado para VirusStatus. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV47	VARCHAR2(255)	Valor reservado 47 Reservado para uso futuro de Novell. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV48	VARCHAR2(255)	Valor reservado 48 Reservado para DestinationOperationalContext. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV49	VARCHAR2(255)	Valor reservado 49 Reservado para uso futuro de Novell. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
REFERENCE_ID 01 - 20	number	Reservado para uso futuro de Novell. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
CV01 - 10	NUMBER	Valor personalizado de 1 a 10 Reservado para uso del cliente, normalmente para asociación de datos comerciales relevantes
CV11 - 20	DATE	Valor personalizado de 11 a 20 Reservado para uso del cliente, normalmente para asociación de datos comerciales relevantes
CV21 - 100	VARCHAR2(255)	Valor personalizado de 21 a 100 Reservado para uso del cliente, normalmente para asociación de datos comerciales relevantes

EVT_AGENT_RPT_V

La vista hace referencia a la tabla EVT_AGENT que almacena información acerca de recopiladores.

Nombre de la columna	Tipo de datos	Comentario
AGENT_ID	number	Identificador de recopilador
AGENT	varchar2(64)	Nombre del recopilador
PORT	varchar2(64)	Puerto del recopilador
REPORT_NAME	varchar2(255)	Nombre del informador
PRODUCT_NAME	varchar2(255)	Nombre de producto
SENSOR_NAME	varchar2(255)	Nombre del sensor
SENSOR_TYPE	varchar2(5)	Tipo de sensor: H - de host N - de red V - virus O - otro
DEVICE_CTGRY	varchar2(255)	Categoría de dispositivo
SOURCE_UUID	varchar2	UUID (Identificador exclusivo universal de componente de origen)
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

EVT_ASSET_RPT_V

La vista hace referencia a la tabla EVT_ASSET que almacena información de activos.

Nombre de la columna	Tipo de datos	Comentario
EVENT_ASSET_ID	number	Identificador de activo de evento
ASSET_NAME	varchar2(255)	Nombre de activo
PHYSICAL_ASSET_NAME	varchar2(255)	Nombre de activo físico
REFERENCE_ASSET_ID	varchar2(100)	Identificador de activo de referencia, enlaza con el sistema de administración de activos de origen.
MAC_ADDRESS	varchar2(100)	Dirección MAC
RACK_NUMBER	varchar2(50)	Número de bastidor
ROOM_NAME	varchar2(100)	Nombre de sala
BUILDING_NAME	varchar2(255)	Nombre de edificio
CITY	varchar2(100)	Ciudad
STATE	varchar2(100)	Estado
COUNTRY	varchar2(100)	País
ZIP_CODE	varchar2(50)	Código postal
ASSET_CATEGORY_NAME	varchar2(100)	Nombre de categoría de activo
NETWORK_IDENTITY_NAME	varchar2(255)	Nombre de identidad de red de activo
ENVIRONMENT_IDENTITY_NAME	varchar2(255)	Nombre de entorno
ASSET_VALUE_NAME	varchar2(50)	Nombre de valor de activo

Nombre de la columna	Tipo de datos	Comentario
CRITICALITY_NAME	varchar2(50)	Nombre de importancia de activo
SENSITIVITY_NAME	varchar2(50)	Nombre de sensibilidad de activo
CONTACT_NAME_1	varchar2(255)	Nombre de organización/persona de contacto 1
CONTACT_NAME_2	varchar2(255)	Nombre de organización/persona de contacto 2
ORGANIZATION_NAME_1	varchar2(100)	Organización propietaria del activo nivel 1
ORGANIZATION_NAME_2	varchar2(100)	Organización propietaria del activo nivel 2
ORGANIZATION_NAME_3	varchar2(100)	Organización propietaria del activo nivel 3
ORGANIZATION_NAME_4	varchar2(100)	Organización propietaria del activo nivel 4
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

EVT_DEST_EVT_NAME_SMRY_1_RPT_V

La vista resume el total de eventos por destino, taxonomía, nombre de evento, gravedad y fecha y hora de evento.

Nombre de la columna	Tipo de datos	Comentario
DESTINATION_IP	number	Dirección IP de destino
DESTINATION_EVENT_ASSET_ID	number	Identificador de activo de evento
TAXONOMY_ID	number	Identificador de taxonomía
EVENT_NAME_ID	number	Identificador de nombre de evento
SEVERITY	number	Gravedad del evento
CUSTOMER_ID	number	Identificador de cliente
EVT_TIME	date	Fecha y hora del evento
EVT_COUNT	number	Total de eventos
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

EVT_DEST_SMRY_1_RPT_V

La vista contiene información de resumen de destino de eventos

Nombre de la columna	Tipo de datos	Comentario
DESTINATION_IP	number	Dirección IP de destino
DESTINATION_EVENT_ASSET_ID	number	Identificador de activo de evento
DESTINATION_PORT	varchar2(32)	Puerto de destino
DESTINATION_USR_ID	number	Identificador de usuario de destino
TAXONOMY_ID	number	Identificador de taxonomía
EVENT_NAME_ID	number	Identificador de nombre de evento
RESOURCE_ID	number	Identificador de recurso
AGENT_ID	number	Identificador de recopilador
PROTOCOL_ID	number	Identificador de protocolo
SEVERITY	number	Gravedad del evento
CUSTOMER_ID	number	Identificador de cliente
EVENT_TIME	date	Fecha y hora del evento
EVENT_CNT	number	Total de eventos
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

EVT_DEST_TXNMY_SMRY_1_RPT_V

La vista resume el total de eventos por destino, taxonomía, gravedad y fecha y hora de evento.

Nombre de la columna	Tipo de datos	Comentario
DESTINATION_IP	number	Dirección IP de destino
DESTINATION_EVENT_ASSET_ID	number	Identificador de activo de evento
TAXONOMY_ID	number	Identificador de taxonomía
SEVERITY	number	Gravedad del evento
CUSTOMER_ID	number	Identificador de cliente
EVENT_TIME	date	Fecha y hora del evento
EVENT_COUNT	number	Total de eventos
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

EVT_NAME_RPT_V

La vista hace referencia a la tabla EVT_NAME que almacena información de nombres de eventos.

Nombre de la columna	Tipo de datos	Comentario
EVENT_NAME_ID	number	Identificador de nombre de evento
EVENT_NAME	varchar2(255)	Nombre de evento
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

EVT_PORT_SMRY_1_RPT_V

La vista resume el total de eventos por puerto de destino, gravedad y fecha y hora de evento.

Nombre de la columna	Tipo de datos	Comentario
DESTINATION_PORT	Varchar2(32)	Puerto de destino
SEVERITY	number	Gravedad del evento
CUSTOMER_ID	number	Identificador de cliente
EVENT_TIME	date	Fecha y hora del evento
EVENT_COUNT	number	Total de eventos
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

EVT_PRTCL_RPT_V

La vista hace referencia a la tabla EVT_PRTCL que almacena información de protocolo de eventos.

Nombre de la columna	Tipo de datos	Comentario
PROTOCOL_ID	number	Identificador de protocolo
PROTOCOL_NAME	varchar2(255)	Nombre de protocolo
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

EVT_RSRC_RPT_V

La vista hace referencia a la tabla EVT_RCRS que almacena información de recursos de eventos.

Nombre de la columna	Tipo de datos	Comentario
RESOURCE_ID	number	Identificador de recurso
RESOURCE_NAME	varchar2(255)	Nombre de recurso
SUBRESOURCE_NAME	varchar2(255)	Nombre de subrecurso
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

EVT_SEV_SMRY_1_RPT_V

La vista resume el total de eventos por gravedad y fecha y hora de evento.

Nombre de la columna	Tipo de datos	Comentario
SEVERITY	number	Gravedad del evento
CUSTOMER_ID	number	Identificador de cliente
EVENT_TIME	date	Fecha y hora del evento
EVENT_COUNT	number	Total de eventos
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

EVT_SRC_SMRY_1_RPT_V

La vista contiene información de resumen de origen y destino del evento.

Nombre de la columna	Tipo de datos	Comentario
SOURCE_IP	number	Dirección IP de origen
SOURCE_EVENT_ASSET_ID	number	Identificador de activo de evento de origen
SOURCE_PORT	varchar2(32)	Puerto de origen
SOURCE_USER_ID	number	Identificador de usuario de origen
TAXONOMY_ID	number	Identificador de taxonomía
EVENT_NAME_ID	number	Identificador de nombre de evento
RESOURCE_ID	number	Identificador de recurso
AGENT_ID	number	Identificador de recopilador
PROTOCOL_ID	number	Identificador de protocolo
SEVERITY	number	Gravedad del evento
CUSTOMER_ID	number	Identificador de cliente
EVENT_TIME	date	Fecha y hora del evento
EVENT_COUNT	number	Total de eventos
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

EVT_TXNMY_RPT_V

La vista hace referencia a la tabla EVT_TXNMY que almacena información de taxonomía del evento.

Nombre de la columna	Tipo de datos	Comentario
TAXONOMY_ID	number	Identificador de taxonomía
TAXONOMY_LEVEL_1	varchar2(100)	Taxonomía nivel 1
TAXONOMY_LEVEL_2	varchar2(100)	Taxonomía nivel 2
TAXONOMY_LEVEL_3	varchar2(100)	Taxonomía nivel 3
TAXONOMY_LEVEL_4	varchar2(100)	Taxonomía nivel 4
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

EVT_USR_RPT_V

La vista hace referencia a la tabla EVT_USR que almacena información de usuarios de eventos.

Nombre de la columna	Tipo de datos	Comentario
USER_ID	number	Identificador de usuario
USER_NAME	varchar2(255)	Nombre de usuario
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

EXTERNAL_DATA_RPT_V

La vista hace referencia a la tabla EXTERNAL_DATA que almacena datos externos.

Nombre de la columna	Tipo de datos	Comentario
EXTERNAL_DATA_ID	number	Identificador de datos externos
SOURCE_NAME	varchar2(50)	Nombre de origen
SOURCE_DATA_ID	varchar2(255)	Identificador de datos de origen
EXTERNAL_DATA	text	Datos externos
EXTERNAL_DATA_TYPE	varchar2(10)	Tipo de datos externos
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

HIST_EVENTS_RPT_V

Vista de eventos históricos (eventos restaurados de archivos).

HIST_INCIDENTS_RPT_V

Vista de eventos históricos (eventos restaurados de archivos).

IMAGES_RPT_V

La vista hace referencia a la tabla IMAGES que almacena información de imágenes descriptivas generales del sistema.

Nombre de la columna	Tipo de datos	Comentario
NAME	VARCHAR2(128)	Nombre de imagen
TYPE	VARCHAR2(64)	Tipo de imagen
DATA	CLOB	Datos de imagen
DATE_CREATED	DATE	Fecha de inserción
DATE_MODIFIED	DATE	Fecha de la última actualización
CREATED_BY	NUMBER	ID del usuario que realiza la inserción
MODIFIED_BY	NUMBER	ID del usuario que realizó la última actualización

INCIDENTS_ASSETS_RPT_V

La vista hace referencia a la tabla INCIDENTS_ASSETS que almacena información sobre activos que componen incidencias creadas en la consola de Sentinel.

Nombre de la columna	Tipo de datos	Comentario
INC_ID	NUMBER	Identificador de incidencia - número de secuencia
ASSET_ID	varchar2	UUID (Identificador exclusivo universal de activo)
DATE_CREATED	DATE	Fecha de inserción
DATE_MODIFIED	DATE	Fecha de la última actualización
CREATED_BY	NUMBER	ID del usuario que realiza la inserción
MODIFIED_BY	NUMBER	ID del usuario que realizó la última actualización

INCIDENTS_EVENTS_RPT_V

La vista hace referencia a la tabla INCIDENTS_EVENTS que almacena información sobre eventos que componen incidencias creadas en la consola de Sentinel.

Nombre de la columna	Tipo de datos	Comentario
INC_ID	NUMBER	Identificador de incidencia - número de secuencia
EVT_ID	varchar2	UUID (Identificador exclusivo universal de evento)
EVT_TIME	DATE	Fecha y hora del evento
DATE_CREATED	DATE	Fecha de inserción
DATE_MODIFIED	DATE	Fecha de la última actualización
CREATED_BY	NUMBER	ID del usuario que realiza la inserción
MODIFIED_BY	NUMBER	ID del usuario que realizó la última actualización

INCIDENTS_RPT_V

La vista hace referencia a la tabla INCIDENTS que almacena información que describe los detalles de incidencias creadas en la consola de Sentinel.

Nombre de la columna	Tipo de datos	Comentario
INC_ID	NUMBER	Identificador de incidencia - número de secuencia
NAME	VARCHAR2(255)	Nombre de incidencia
SEVERITY	NUMBER	Gravedad de la incidencia
STT_ID	NUMBER	ID de estado de incidencia
SEVERITY_RATING	VARCHAR2(32)	Media de la gravedad de todos los eventos que componen una incidencia.
VULNERABILITY_RATING	VARCHAR2(32)	Reservado para uso futuro de Novell. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
CRITICALITY_RATING	VARCHAR2(32)	Reservado para uso futuro de Novell. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
DATE_CREATED	DATE	Fecha de inserción
DATE_MODIFIED	DATE	Fecha de la última actualización
CREATED_BY	NUMBER	ID del usuario que realiza la inserción
MODIFIED_BY	NUMBER	ID del usuario que realizó la última actualización
INC_DESC	varchar2(4000)	Descripción de la incidencia
INC_PRIORITY	number	Prioridad de la incidencia
INC_CAT	varchar2(255)	Categoría de la incidencia
INC_RES	varchar2(4000)	Resolución de la incidencia

INCIDENTS_VULN_RPT_V

La vista hace referencia a la tabla INCIDENTS_VULN que almacena información sobre vulnerabilidades que componen incidencias creadas en la consola de Sentinel.

Nombre de la columna	Tipo de datos	Comentario
INC_ID	NUMBER	Identificador de incidencia - número de secuencia
VULN_ID	varchar2(36)	UUID (Identificador exclusivo universal de vulnerabilidad)
DATE_CREATED	DATE	Fecha de inserción
DATE_MODIFIED	DATE	Fecha de la última actualización
CREATED_BY	NUMBER	ID del usuario que realiza la inserción
MODIFIED_BY	NUMBER	ID del usuario que realizó la última actualización

L_STAT_RPT_V

La vista hace referencia a la tabla L_STAT que almacena información estadística.

Nombre de la columna	Tipo de datos	Comentario
RES_NAME	VARCHAR2(32)	Nombre de recurso
STATS_NAME	VARCHAR2(32)	Nombre de estadística
STATS_VALUE	VARCHAR2(32)	Valor de la estadística
OPEN_TOT_SECS	NUMERIC	Número de segundos desde 1970.

LOGS_RPT_V

La vista hace referencia a la tabla LOGS_RPT que almacena información de registro.

Tabla LOGS		
Nombre de la columna	**Tipo de datos**	**Comentario**
LOG_ID	NUMBER	Número de secuencia
TIME	DATE	Fecha de registro
MODULE	VARCHAR2(64)	Módulo al que corresponde el registro
TEXT	VARCHAR2(4000)	Texto de registro

NETWORK_IDENTITY_RPT_V

La vista hace referencia a la tabla NETWORK_IDENTITY_LKUP que almacena información acerca de la identidad de red de activos.

Nombre de la columna	Tipo de datos	Comentario
NETWORK_IDENTITY_CD	varchar2(5)	Código de identidad de red
NETWORK_IDENTITY_NAME	varchar2(255)	Nombre de identidad de red
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

ORGANIZATION_RPT_V

La vista hace referencia a la tabla ORGANIZATION que almacena información acerca de organizaciones (activos).

Nombre de la columna	Tipo de datos	Comentario
ORGANIZATION_ID	varchar2	Identificador de organización
ORGANIZATION_NAME	varchar2(100)	Nombre de organización
CUSTOMER_ID	number	Identificador de cliente
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

PERSON_RPT_V

La vista hace referencia a la tabla PERSION que almacena información de personas (activos).

Nombre de la columna	Tipo de datos	Comentario
PERSON_ID	varchar2	Identificador de persona
FIRST_NAME	varchar2(255)	Nombre
LAST_NAME	varchar2(255)	Apellidos
CUSTOMER_ID	number	Identificador de cliente
PHONE_NUMBER	varchar2(50)	Número de teléfono
EMAIL_ADDRESS	varchar2(255)	Dirección de correo electrónico
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

PHYSICAL_ASSET_RPT_V

La vista hace referencia a la tabla PHYSICAL_ASSET que almacena información de activos físicos.

Nombre de la columna	Tipo de datos	Comentario
PHYSICAL_ASSET_ID	varchar2	Identificador de activo físico
CUSTOMER_ID	number	Identificador de cliente
LOCATION_ID	number	Identificación de ubicación
HOST_NAME	varchar2(255)	Nombre del host
IP_ADDRESS	number	Dirección IP
NETWORK_IDENTITY_CD	varchar2(5)	Código de identidad de red
MAC_ADDRESS	varchar2(100)	Dirección MAC
RACK_NUMBER	varchar2(50)	Número de bastidor
ROOM_NAME	varchar2(100)	Nombre de sala
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

PRODUCT_RPT_V

La vista hace referencia a la tabla PRDT que almacena información de productos de activos.

Nombre de la columna	Tipo de datos	Comentario
PRODUCT_ID	number	Identificador de producto
PRODUCT_NAME	varchar2(255)	Nombre de producto
PRODUCT_VERSION	varchar2(100)	Versión de producto
VENDOR_ID	number	Identificador de proveedor
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

ROLE_RPT_V

La vista hace referencia a la tabla ROLE_LKUP que almacena información de funciones de usuarios (activos).

Nombre de la columna	Tipo de datos	Comentario
ROLE_CODE	varchar2(5)	Código de función
ROLE_NAME	varchar2(255)	Nombre de la función
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

SENSITIVITY_RPT_V

La vista hace referencia a la tabla SENSITIVITY_LKUP que almacena información acerca de la sensibilidad de activos.

Nombre de la columna	Tipo de datos	Comentario
SENSITIVITY_CODE	varchar2(5)	Código de sensibilidad de activo
SENSITIVITY_NAME	varchar2(50)	Nombre de sensibilidad de activo
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la acción
MODIFIED_BY	number	ID del usuario que realiza la acción

STATES_RPT_V

La vista hace referencia a la tabla STATES que almacena definiciones de los estados determinados por aplicaciones o contexto.

Nombre de la columna	Tipo de datos	Comentario
STT_ID	NUMBER	ID de estado - número de secuencia
CONTEXT	VARCHAR2(64)	Contexto del estado Es decir caso, incidencia, usuario
NAME	VARCHAR2(64)	Nombre del estado.
TERMINAL_FLAG	VARCHAR2(1)	Indica si el estado de la incidencia es resuelto.
DATE_CREATED	DATE	Fecha de inserción
DATE_MODIFIED	DATE	Fecha de la última actualización
MODIFIED_BY	NUMBER	ID del usuario que realiza la inserción
CREATED_BY	NUMBER	ID del usuario que realizó la última actualización

UNASSIGNED_INCIDENTS_RPT_V.

La vista hace referencia a las tablas CASES e INCIDENTS para informar sobre casos no asignados.

Nombre	Tipo de datos
INC_ID	NUMBER
NAME	VARCHAR2(255)
SEVERITY	NUMBER
STT_ID	NUMBER
SEVERITY_RATING	VARCHAR2(32)
VULNERABILITY_RATING	VARCHAR2(32)
CRITICALITY_RATING	VARCHAR2(32)
DATE_CREATED	DATE
DATE_MODIFIED	DATE
CREATED_BY	NUMBER
MODIFIED_BY	NUMBER
INC_DESC	VARCHAR2(4000)
INC_PRIORITY	NUMBER
INC_CAT	VARCHAR2(255)
INC_RES	VARCHAR2(4000)

USERS_RPT_V

La vista hace referencia a la tabla USERS que detalla todos los usuarios de la aplicación. Los usuarios también se crearán como usuarios de base de datos para adaptarse a herramientas de generación de informes de terceros.

Nombre de la columna	Tipo de datos	Comentario
USR_ID	NUMBER	Identificador de usuario - número de secuencia
NAME	VARCHAR2(64)	Corto, nombre de usuario exclusivo utilizado como inicio de sesión
CNT_ID	NUMBER	ID de contacto - Número de secuencia
STT_ID	NUMBER	ID de estado. El estado puede ser activo o inactivo.
DESCRIPTION	VARCHAR2(512)	Comentarios
DATE_CREATED	DATE	Fecha de inserción
DATE_MODIFIED	DATE	Fecha de la última actualización
CREATED_BY	NUMBER	ID del usuario que realiza la inserción
MODIFIED_BY	NUMBER	ID del usuario que realizó la última actualización
PERMISSIONS	VARCHAR2(4000)	Permisos actualmente asignados al usuario de Sentinel
FILTER	VARCHAR2(128)	Filtro de seguridad actual asignado al usuario de Sentinel
UPPER_NAME	VARCHAR2(64)	Nombre de usuario en mayúsculas
DOMAIN_AUTH_IND	NUMBER	Indicación de autenticación de dominio

VENDOR_RPT_V

La vista hace referencia a la tabla VNDR que almacena información de proveedores de productos de activos.

Nombre de la columna	Tipo de datos	Comentario
VENDOR_ID	number	Identificador de proveedor
VENDOR_NAME	varchar2(255)	Nombre de proveedor
DATE_CREATED	date	Fecha de inserción
DATE_MODIFIED	date	Fecha de la última actualización
CREATED_BY	number	ID del usuario que realiza la inserción
MODIFIED_BY	number	ID del usuario que realizó la última actualización

VULN_CALC_SEVERITY_RPT_V

La vista hace referencia a VULN_RSRC y VULN para calcular la calificación de gravedad de vulnerabilidad de Sentinel en función de las vulnerabilidades actuales.

Nombre de la columna	Tipo de datos
RSRC_ID	VARCHAR22(36)
IP	VARCHAR22(32)
HOST_NAME	VARCHAR22(255)
CRITICALITY	NUMBER
ASSIGNED_VULN_SEVERITY	NUMBER
VULN_COUNT	Total de vulnerabilidades para el recurso especificado
CALC_SEVERITY	Gravedad calculada en función de ASSIGNED_VULN_SEVERITY y CRITICALITY

VULN_CODE_RPT_V

La vista hace referencia a la tabla VULN_CODE que almacena códigos de vulnerabilidad asignados por la industria como CVE y CAN de Mitre.

Nombre de la columna	Tipo de datos
VULN_CODE_ID	VARCHAR2(36)
VULN_ID	VARCHAR2(36)
VULN_CODE_TYPE	VARCHAR2(64)
VULN_CODE_VALUE	VARCHAR2(255)
URL	VARCHAR2(512)
DATE_CREATED	DATE
DATE_MODIFIED	DATE
CREATED_BY	NUMBER
MODIFIED_BY	NUMBER

VULN_INFO_RPT_V

La vista hace referencia a la tabla VULN_INFO que almacena información adicional recogida durante una exploración.

Nombre de la columna	Tipo de datos
VULN_INFO_ID	VARCHAR2(36)
VULN_ID	VARCHAR2(36)
VULN_INFO_TYPE	VARCHAR2(36)
VULN_INFO_VALUE	VARCHAR2(2000)
DATE_CREATED	DATE
DATE_MODIFIED	DATE
CREATED_BY	NUMBER
MODIFIED_BY	NUMBER

VULN_RPT_V

La vista hace referencia a la tabla VULN que almacena información del sistema explorado. Cada escáner tendrá su propia entrada para cada sistema.

Nombre de la columna	Tipo de datos
VULN_ID	VARCHAR2(36)
RSRC_ID	VARCHAR2(36)
PORT_NAME	VARCHAR2(64)
PORT_NUMBER	NUMBER
NETWORK_PROTOCOL	NUMBER
APPLICATION_PROTOCOL	VARCHAR2(64)
ASSIGNED_VULN_SEVERITY	NUMBER
COMPUTED_VULN_SEVERITY	NUMBER
VULN_DESCRIPTION	CLOB
VULN_SOLUTION	CLOB
VULN_SUMMARY	VARCHAR2(1000)
BEGIN_EFFECTIVE_DATE	DATE
END_EFFECTIVE_DATE	DATE
DETECTED_OS	VARCHAR2(64)
DETECTED_OS_VERSION	VARCHAR2(64)
SCANNED_APP	VARCHAR2(64)
SCANNED_APP_VERSION	VARCHAR2(64)
VULN_USER_NAME	VARCHAR2(64)
VULN_USER_DOMAIN	VARCHAR2(64)
VULN_TAXONOMY	VARCHAR2(1000)
SCANNER_CLASSIFICATION	VARCHAR2(255)
VULN_NAME	VARCHAR2(300)
VULN_MODULE	VARCHAR2(64)
DATE_CREATED	DATE
DATE_MODIFIED	DATE
CREATED_BY	NUMBER
MODIFIED_BY	NUMBER

VULN_RSRC_RPT_V

La vista hace referencia a la tabla VULN_RSRC que almacena cada recurso explorado para una exploración en particular.

Nombre de la columna	Tipo de datos
RSRC_ID	VARCHAR2(36)
SCANNER_ID	VARCHAR2(36)
IP	VARCHAR2(32)
HOST_NAME	VARCHAR2(255)
LOCATION	VARCHAR2(128)
DEPARTMENT	VARCHAR2(128)
BUSINESS_SYSTEM	VARCHAR2(128)
OPERATIONAL_ENVIRONMENT	VARCHAR2(64)
CRITICALITY	NUMBER
REGULATION	VARCHAR2(128)
REGULATION_RATING	VARCHAR2(64)
DATE_CREATED	DATE
DATE_MODIFIED	DATE
CREATED_BY	NUMBER
MODIFIED_BY	NUMBER

VULN_RSRC_SCAN_RPT_V

La vista hace referencia a la tabla VULN_RSRC_SCAN que almacena cada recurso explorado para una exploración en particular.

Nombre de la columna	Tipo de datos
RSRC_ID	VARCHAR2(36)
SCAN_ID	VARCHAR2(36)
DATE_CREATED	DATE
DATE_MODIFIED	DATE
CREATED_BY	NUMBER
MODIFIED_BY	NUMBER

VULN_SCAN_RPT_V

La vista hace referencia a la tabla que almacena información perteneciente a las exploraciones.

Nombre de la columna	Tipo de datos
SCAN_ID	VARCHAR2(36)
SCANNER_ID	VARCHAR2(36)
SCAN_TYPE	VARCHAR2(10)
SCAN_START_DATE	DATE
SCAN_END_DATE	DATE
CONSOLIDATION_SERVER	VARCHAR2(64)
DATE_CREATED	DATE
DATE_MODIFIED	DATE
CREATED_BY	NUMBER
MODIFIED_BY	NUMBER

VULN_SCAN_VULN_RPT_V

La vista hace referencia a la tabla VULN_SCAN_VULN que almacena las vulnerabilidades detectadas durante las exploraciones.

Nombre de la columna	Tipo de datos
SCAN_ID	VARCHAR2(36)
VULN_ID	VARCHAR2(36)
DATE_CREATED	DATE
DATE_MODIFIED	DATE
CREATED_BY	NUMBER
MODIFIED_BY	NUMBER

VULN_SCANNER_RPT_V

La vista hace referencia a la tabla VULN_SCANNER que almacena información de escáneres de vulnerabilidades.

Nombre de la columna	Tipo de datos
SCANNER_ID	VARCHAR2(36)
PRODUCT_NAME	VARCHAR2(100)
PRODUCT_VERSION	VARCHAR2(64)
SCANNER_TYPE	VARCHAR2(64)
VENDOR	VARCHAR2(100)
SCANNER_INSTANCE	VARCHAR2(64)
DATE_CREATED	DATE
DATE_MODIFIED	DATE
CREATED_BY	NUMBER
MODIFIED_BY	NUMBER

12 Vistas de la base de datos de Sentinel para Microsoft SQL Server

NOTA: El término Agente puede intercambiarse con Recopilador. En adelante, los agentes se denominarán recopiladores.

En este capítulo se enumeran las vistas del esquema de Sentinel para Microsoft SQL Server. Las vistas proporcionan información para desarrollar sus propios informes (Crystal Reports).

Vistas

ADV_ALERT_CVE_RPT_V

La vista hace referencia a la tabla ADV_ALERT_CVE que almacena el número de identificación de alerta del Asesor.

Nombre de la columna	Tipo de datos	Comentario
ALERT_ID	int	Identificador de anotación - número de secuencia.
CVE	varchar	
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la acción.
MODIFIED_BY	int	ID del usuario que realiza la acción.

ADV_ALERT_PRODUCT_RPT_V

La vista hace referencia a la tabla ADV_ALERT_PRODUCT que almacena información de productos del Asesor como, por ejemplo, número de identificación de Service Pack, versión y fecha de creación.

Nombre de la columna	Tipo de datos	Comentario
ALERT_ID	int	Identificador de anotación - número de secuencia.
SERVICE_PACK_ID	int	
VENDOR	varchar	
PRODUCT	varchar	
VERSION	varchar	Contiene el número de versión.
SERVICE_PACK	varchar	
PRIMARY_FLAG	int	
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la acción.
MODIFIED_BY	int	ID del usuario que realiza la acción.

ADV_ALERT_RPT_V

La vista hace referencia a la tabla ADV_ALERT que almacena información de alertas del Asesor como, por ejemplo, nombre, tipo de amenaza y fecha de publicación.

Nombre de la columna	Tipo de datos	Comentario
ALERT_ID	int	Identificador de anotación - número de secuencia
VERSION	int	Contiene el número de versión
TEMPLATE_ID	int	
TEMPLATE_NAME	varchar	
THREAT_CATEGORY_NAME	varchar	
THREAT_TYPE_NAME	varchar	
HEADLINE	text	
FIRST_PUBLISHED	datetime	
LAST_PUBLISHED	datetime	
STATUS	varchar	
URGENCY_ID	int	
CREDIBILITY_ID	int	
SEVERITY_ID	int	
SUMMARY	text	
LEGAL_DISCLAIMER	text	
COPYRIGHT	varchar	
BEGIN_EFFECTIVE_DATE	datetime	
END_EFFECTIVE_DATE	datetime	
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la acción.
MODIFIED_BY	int	ID del usuario que realiza la acción.

ADV_ATTACK_ALERT_RPT_V

La vista hace referencia a la tabla ADV_ATTACK_ALERT que almacena información de ataques del Asesor como, por ejemplo, nombre, tipo de amenaza y fecha de publicación.

Nombre de la columna	Tipo de datos	Comentario
ATTACK_ID	int	
ALERT_ID	int	
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la acción.
MODIFIED_BY	int	ID del usuario que realiza la acción.

ADV_ATTACK_CVE_RPT_V

La vista hace referencia a la tabla ADV_ATTACK_CVE que almacena información de CVE del Asesor.

Nombre de la columna	Tipo de datos	Comentario
ATTACK_ID	int	
CVE	varchar	
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la acción.
MODIFIED_BY	int	ID del usuario que realiza la acción.

ADV_ATTACK_MAP_RPT_V

La vista hace referencia a la tabla ADV_ATTACK_MAP que almacena información de asignación del Asesor.

Nombre de la columna	Tipo de datos	Comentario
ATTACK_KEY	int	
ATTACK_ID	int	
SERVICE_PACK_ID	int	
ATTACK_NAME	varchar	
ATTACK_CODE	varchar	
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la acción.
MODIFIED_by	int	ID del usuario que realiza la acción.

ADV_ATTACK_PLUGIN_RPT_V

La vista hace referencia a la tabla ADV_ATTACK_PLUGIN que almacena información de módulos auxiliares (plug-ins) del Asesor.

Nombre de la columna	Tipo de datos	Comentario
PLUGIN_KEY	int	
ATTACK_ID	int	
SERVICE_PACK_ID	int	
PLUGIN_ID	varchar	
PLUGIN_NAME	varchar	
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la acción.
MODIFIED_BY	int	ID del usuario que realiza la acción.

ADV_ATTACK_RPT_V

La vista hace referencia a la tabla ADV_ATTACK que almacena información de ataques del Asesor.

Nombre de la columna	Tipo de datos	Comentario
ALERT_ID	int	
TRUSECURE_ATTACK_NAME	int	
FEED_DATE_CREATED	datetime	
FEED_DATE_UPDATED	datetime	
ATTACK_CATEGORY	varchar	
URGENCY_ID	int	
SEVERITY_ID	int	
LOCAL	int	
REMOTE	int	
BEGIN_EFFECTIVE_DATE	datetime	
END_EFFECTIVE_DATE	datetime	
DESCRIPTION	text	
SCENARIO	text	
IMPACT	text	
SAFEGUARDS	text	
PATCHES	text	
FALSE_POSITIVES	text	
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la acción.
MODIFIED_BY	int	ID del usuario que realiza la acción.

ADV_CREDIBILITY_RPT_V

Ver la tabla de referencias ADV_CREDIBILITY que almacena información de credibilidad del Asesor.

Nombre de la columna	Tipo de datos	Comentario
CREDIBILITY_ID	int	
CREDIBILITY_RATING	varchar	
CREDIBILITY_EXPLANATION	varchar	
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la acción.
MODIFIED_BY	int	ID del usuario que realiza la acción.

ADV_FEED_RPT_V

La vista hace referencia a la tabla ADV_FEED que almacena información de los datos del Asesor como, por ejemplo, nombre y fecha.

Nombre de la columna	Tipo de datos	Comentario
FEED_NAME	varchar	
FEED_FILE	varchar	
BEGIN_DATE	datetime	
END_DATE	datetime	
FEED_INSERT	int	
FEED_UPDATE	int	
FEED_EXPIRE	int	

ADV_PRODUCT_RPT_V

La vista hace referencia a la tabla ADV_PRODUCT que almacena información de productos del Asesor como, por ejemplo, proveedor e ID de producto.

Nombre de la columna	Tipo de datos	Comentario
PRODUCT_ID	int	
VENDOR_ID	int	
PRODUCT_CATEGORY_ID	int	
PRODUCT_CATEGORY_NAME	varchar	
PRODUCT_TYPE-ID	int	
PRODUCT_TYPE_NAME	varchar	
PRODUCT_NAME	varchar	
PRODUCT_DESCRIPTION	varchar	
FEED_DATE_CREATED	datetime	
FEED_DATE_UPDATED	datetime	
ACTIVE_FLAG	int	
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la acción.
MODIFIED_BY	int	ID del usuario que realiza la acción.

ADV_PRODUCT_SERVICE_PACK_RPT_V

La vista hace referencia a la tabla ADV_PRODUCT_SERVICE _PACK que almacena información de Service Pack del asesor como, por ejemplo, nombre del Service Pack, ID de la versión y fecha.

Nombre de la columna	Tipo de datos	Comentario
SERVICE_PACK_ID	int	
VERSION_ID	int	Contiene el número de ID de la versión.
SERVICE_PACK_NAME	varchar	
FEED_DATE_CREATED	datetime	
FEED_DATE_UPDATED	datetime	
ACTIVE_FLAG	int	
BEGIN_EFFECTIVE_DATE	datetime	

Nombre de la columna	Tipo de datos	Comentario
END_EFFECTIVE_DATE	datetime	
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la acción.
MODIFIED_BY	int	ID del usuario que realiza la acción.

ADV_PRODUCT_VERSION_RPT_V

La vista hace referencia a la tabla ADV_PRODUCT_VERSION que almacena información de la versión de productos del Asesor como, por ejemplo, nombre de la versión, producto e ID de la versión.

Nombre de la columna	Tipo de datos	Comentario
VERSION_ID	int	Contiene el número de ID de la versión.
PRODUCT_ID	int	
VERSION_NAME	varchar	
FEED_DATE_CREATED	datetime	
FEED_DATE_UPDATED	datetime	
ACTIVE_FLAG	int	
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	int	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la acción.
MODIFIED_BY	int	ID del usuario que realiza la acción.

ADV_SEVERITY_RPT_V

La vista hace referencia a la tabla ADV_SEVERITY que almacena información de valoraciones de gravedad del Asesor.

Nombre de la columna	Tipo de datos	Comentario
SEVERITY_ID	int	
SEVERITY_RATING	varchar	
SEVERITY_EXPLANATION	varchar	
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la acción.
MODIFIED_BY	int	ID del usuario que realiza la acción.

ADV_SUBALERT_RPT_V

La vista hace referencia a la tabla ADV_SUBALERT.

Nombre de la columna	Tipo de datos	Comentario
ALERT_ID	int	
SUBALERT_ID	int	
CHANGED_SECTIONS	varchar	
VARIANTS	text	
VIRUS_NAME	text	
DESCRIPTION	text	
IMPACT	text	

Nombre de la columna	Tipo de datos	Comentario
WARNING_INDICATORS	text	
TECHNICAL_INFO	text	
TRUSECURE_COMMENTS	text	
VENDOR_ANNOUNCEMENTS	text	
SAFEGUARDS	text	
PATCHES_SOFTWARE	text	
ALERT_HISTORY	text	
BACKGROUND_INFO	text	
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la acción.
MODIFIED_BY	int	ID del usuario que realiza la acción.

ADV_URGENCY_RPT_V

La vista hace referencia a la tabla ADV_URGENCY.

Nombre de la columna	Tipo de datos	Comentario
URGENCY_ID	int	
URGENCY_RATING	varchar	
URGENCY_EXPLANATION	varchar	
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la acción.
MODIFIED_BY	int	ID del usuario que realiza la acción.

ADV_VENDOR_RPT_V

La vista hace referencia a la tabla ADV_VENDOR que almacena información de direcciones del Asesor.

Nombre de la columna	Tipo de datos	Comentario
VENDOR_ID	int	
VENDOR_NAME	varchar	
CONTACT_PERSON	varchar	
ADDRESS_LINE_1	varchar	
ADDRESS_LINE_2	varchar	
ADDRESS_LINE_3	varchar	
ADDRESS_LINE_4	varchar	
CITY	varchar	
STATE	varchar	
COUNTRY	varchar	
ZIP_CODE	varchar	
URL	varchar	
PHONE	varchar	
FAX	varchar	
EMAIL	varchar	
PAGER	varchar	
FEED_DATE_CREATED	datetime	

Nombre de la columna	Tipo de datos	Comentario
FEED_DATE_UPDATED	datetime	
ACTIVE_FLAG	int	
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la acción.
MODIFIED_BY	int	ID del usuario que realiza la acción.

ADV_VULN_PRODUCT_RPT_V

La vista hace referencia a la tabla ADV_VULN_PRODUCT que almacena la ID de ataque de vulnerabilidad y la ID de Service Pack del Asesor.

Nombre de la columna	Tipo de datos	Comentario
ATTACK_ID	int	
SERVICE_PACK_ID	int	
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la acción.
MODIFIED_BY	int	ID del usuario que realiza la acción.

ANNOTATIONS_RPT_V

La vista hace referencia a la tabla ANNOTATIONS que almacena documentación o notas que pueden asociarse con objetos en el sistema Sentinel como, por ejemplo, los casos y las incidencias.

Nombre de la columna	Tipo de datos	Comentario
ANN_ID	INT	Identificador de anotación - número de secuencia.
TEXT	VARCHAR(4000)	Documentación o notas.
DATE_CREATED	DATETIME	Fecha de inserción.
DATE_MODIFIED	DATETIME	Fecha de la última actualización.
MODIFIED_BY	INT	ID del usuario que realizó la última actualización.
CREATED_BY	INT	ID del usuario que realiza la inserción.
ACTION	Varchar(255)	Acción.

ASSET_CTGRY_RPT_V

La vista hace referencia a la tabla ASSET_CTGRY que almacena información acerca de las categorías de activos (p. ej., hardware, software, SO, base de datos, etc.).

Nombre de la columna	Tipo de datos	Comentario
ASSET_CATEGORY_ID	bigint	Identificador de categoría de activo.
ASSET_CATEGORY_NAME	varchar(100)	Nombre de categoría de activo.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

ASSET_HOSTNAME_RPT_V

La vista hace referencia a la tabla ASSET_HOSTNAME que almacena información acerca de nombres de host alternativos de los activos.

Nombre de la columna	Tipo de datos	Comentario
ASSET_HOSTNAME_ID	Uniqueidentifier	Identificador de nombre de host alternativo del activo
PHYSICAL_ASSET_ID	uniqueidentifier	Identificador de activo físico.
HOST_NAME	Varchar(255)	Nombre del host.
CUSTOMER_ID	bigint	Identificador de cliente.
DATE_CREATED	datetime	Fecha de la última actualización.
DATE_MODIFIED	datetime	ID del usuario que realizó la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

ASSET_IP_RPT_V

La vista hace referencia a la tabla ASSET_IP que almacena información acerca de direcciones IP alternativas de los activos.

Nombre de la columna	Tipo de datos	Comentario
ASSET_IP_ID	Uniqueidentifier	Identificador de IP alternativo de activo.
PHYSICAL_ASSET_ID	uniqueidentifier	Identificador de activo físico.
IP_ADDRESS	int	Dirección IP de activo.
CUSTOMER_ID	bigint	Identificador de cliente.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

ASSET_LOCATION_RPT_V

La vista hace referencia a la tabla ASSET_LOC que almacena información acerca de las ubicaciones de activos.

Nombre de la columna	Tipo de datos	Comentario
LOCATION_ID	bigint	Identificación de ubicación.
CUSTOMER_ID	bigint	Identificador de cliente.
BUILDING_NAME	varchar(255)	Nombre de edificio.
ADDRESS_LINE_1	varchar(255)	Línea de dirección 1.
ADDRESS_LINE_2	varchar(255)	Línea de dirección 2.
CITY	varchar(100)	Ciudad.
STATE	varchar(100)	Estado.
COUNTRY	varchar(100)	País
ZIP_CODE	varchar(50)	Código postal.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

ASSET_RPT_V

La vista hace referencia a la tabla ASSET que almacena información acerca de los activos físicos y de software.

Nombre de la columna	Tipo de datos	Comentario
ASSET_ID	uniqueidentifier	Identificador de activo.
CUSTOMER_ID	bigint	Identificador de cliente.
ASSET_NAME	varchar(255)	Nombre de activo.
PHYSICAL_ASSET_ID	uniqueidentifier	Identificador de activo físico.
PRODUCT_ID	bigint	Identificador de producto.
ASSET_CATEGORY_ID	bigint	Identificador de categoría de activo.
ENVIRONMENT_IDENTITY_CD	varchar(5)	Código de identificación de entorno.
PHYSICAL_ASSET_IND	bit	Indicador de activo físico.
ASSET_VALUE_CD	varchar(5)	Código de valor de activo.
CRITICALITY_CODE	varchar(5)	Código de importancia de activo.
SENSITIVITY_CODE	varchar(5)	Código de sensibilidad de activo.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

ASSET_VALUE_RPT_V

La vista hace referencia a la tabla ASSET_VAL_LKUP que almacena información acerca del valor de activos.

Nombre de la columna	Tipo de datos	Comentario
ASSET_VALUE_CODE	varchar(5)	Código de valor de activo.
ASSET_VALUE_NAME	varchar(50)	Nombre de valor de activo.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

ASSET_X_ENTITY_X_ROLE_RPT_V

La vista hace referencia a la tabla ASSET_X_ENTITY_X_ROLE que asocia a una persona u organización con un activo.

Nombre de la columna	Tipo de datos	Comentario
PERSON_ID	uniqueidentifier	Identificador de persona.
ORGANIZATION_ID	uniqueidentifier	Identificador de organización.
ROLE_CODE	varchar(5)	Código de función.
ASSET_ID	uniqueidentifier	Identificador de activo.
ENTITY_TYPE_CODE	varchar(5)	Código de tipo de entidad.
PERSON_ROLE_SEQUENCE	int	Orden de las personas bajo una función en particular.
DATE_CREATED	datetime	Fecha de inserción.

Nombre de la columna	Tipo de datos	Comentario
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

ASSOCIATIONS_RPT_V

La vista hace referencia a la tabla ASSOCIATIONS que asocia usuarios con incidencias, incidencias con anotaciones, etc.

Nombre de la columna	Tipo de datos	Comentario
TABLE1	VARCHAR(64)	Nombre de tabla 1. Por ejemplo, incidencias.
ID1	VARCHAR(36)	ID1. Por ejemplo, ID de incidencias.
TABLE2	VARCHAR(64)	Nombre de tabla 2. Por ejemplo, usuarios.
ID2	VARCHAR(36)	ID2. Por ejemplo, ID de usuario.
DATE_CREATED	DATETIME	Fecha de inserción.
DATE_MODIFIED	DATETIME	Fecha de la última actualización.
CREATED_BY	INT	ID del usuario que realiza la inserción.
MODIFIED_BY	INT	ID del usuario que realizó la última actualización.

ATTACHMENTS_RPT_V

La vista hace referencia a la tabla ATTACHMENTS que almacena información de datos adjuntos.

Nombre de la columna	Tipo de datos	Comentario
ATTACHMENT_ID	int	Identificador del adjunto.
NAME	varchar(255)	Nombre del adjunto.
SOURCE_REFERENCE	varchar(64)	Referencia de origen.
TYPE	varchar(32)	Tipo de adjunto.
SUB_TYPE	varchar(32)	Subtipo de adjunto.
FILE_EXTENSION	varchar(32)	Extensión de archive.
ATTACHMENT_DESCRIPTION	varchar(255)	Descripción del adjunto.
DATA	clob	Datos del adjunto.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

CONFIGS_RPT_V

La vista hace referencia a la tabla CONFIGS que almacena información general de configuración de la aplicación.

Nombre de la columna	Tipo de datos	Comentario
USR_ID	VARCHAR(32)	Nombre de usuario.
APPLICATION	VARCHAR(255)	Identificador de la aplicación.
UNIT	VARCHAR(64)	Unidad de la aplicación.
VALUE	VARCHAR(255)	Valor del texto, si corresponde.
DATA	TEXT	Datos XML.
DATE_CREATED	DATETIME	Fecha de inserción.
DATE_MODIFIED	DATETIME	Fecha de la última actualización.
CREATED_BY	INT	ID del usuario que realiza la inserción.
MODIFIED_BY	INT	ID del usuario que realizó la última actualización.

CONTACTS_RPT_V

La vista hace referencia a la tabla CONTACTS que almacena información de contactos.

Nombre de la columna	Tipo de datos	Comentario
CNT_ID	INT	ID del contacto - Número de secuencia.
FIRST_NAME	VARCHAR(20)	Nombre del contacto.
LAST_NAME	VARCHAR(30)	Apellido del contacto.
TITLE	VARCHAR(128)	Cargo del contacto.
DEPARTMENT	VARCHAR(128)	Departamento.
PHONE	VARCHAR(64)	Teléfono del contacto.
EMAIL	VARCHAR(255)	Correo electrónico del contacto.
PAGER	VARCHAR(64)	Buscapersonas del contacto.
CELL	VARCHAR(64)	Móvil del contacto.
DATE_CREATED	DATETIME	Fecha de inserción.
DATE_MODIFIED	DATETIME	Fecha de la última actualización.
CREATED_BY	INT	ID del usuario que realiza la inserción.
MODIFIED_BY	INT	ID del usuario que realizó la última actualización.

CORRELATED_EVENTS_RPT_V

La vista hace referencia a las tablas CORRELATED_EVENTS_* que almacenan información de eventos correlacionados.

Nombre de la columna	Tipo de datos	Comentario
PARENT_EVT_ID	uniqueidentifier	UUID (Identificador exclusivo universal de eventos) del evento padre.
CHILD_EVT_ID	uniqueidentifier	UUID (Identificador exclusivo universal de eventos) de evento hijo.
PARENT_EVT_TIME	DATETIME	Fecha de creación del evento padre.
CHILD_EVT_TIME	DATETIME	Fecha de creación del evento hijo.
DATE_CREATED	FECHA	Fecha de inserción. generada por DAS.
DATE_MODIFIED	DATETIME	Fecha de la última actualización.
CREATED_BY	INT	ID del usuario que realiza la inserción.
MODIFIED_BY	INT	ID del usuario que realizó la última actualización.

CORRELATED_EVENTS_RPT_V1

La vista contiene eventos correlacionados actuales e históricos (eventos correlacionados importados de archivos).

Nombre de la columna	Tipo de datos	Comentario
PARENT_EVT_ID	uniqueidentifier	UUID (Identificador exclusivo universal de eventos) del evento padre.
CHILD_EVT_ID	uniqueidentifier	UUID (Identificador exclusivo universal de eventos) de evento hijo.
PARENT_EVT_TIME	DATETIME	Fecha y hora del evento. padre.
CHILD_EVT_TIME	DATETIME	Fecha y hora del evento. hijo.
DATE_CREATED	DATETIME	Fecha de inserción. Generada por DAS.
DATE_MODIFIED	DATETIME	Fecha de la última actualización.
CREATED_BY	INT	ID del usuario que realiza la inserción.
MODIFIED_BY	INT	ID del usuario que realizó la última actualización.

CRITICALITY_RPT_V

La vista hace referencia a la tabla CRIT_LKUP que contiene información acerca de la importancia de activos.

Nombre de la columna	Tipo de datos	Comentario
CRITICALITY_CODE	varchar(5)	Código de importancia de activo.
CRITICALITY_NAME	varchar(50)	Nombre de importancia de activo.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la acción.
MODIFIED_BY	int	ID del usuario que realiza la acción.

CUST_RPT_V

La vista hace referencia a la tabla CUST que almacena información de clientes para MSSP.

Nombre de la columna	Tipo de datos	Comentario
CUSTOMER_ID	bigint	Identificador de cliente.
CUSTOMER_NAME	varchar(255)	Nombre del cliente.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

ENTITY_TYPE_RPT_V

La vista hace referencia a la tabla ENTITY_TYP que almacena información acerca de los tipos de entidades (personas, organizaciones).

Nombre de la columna	Tipo de datos	Comentario
ENTITY_TYPE_CODE	varchar(5)	Código de tipo de entidad.
ENTITY_TYPE_NAME	varchar(50)	Nombre de tipo de entidad.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

ENV_IDENTITY_RPT_V

La vista hace referencia a la tabla ENV_IDENTITY_LKUP que almacena información acerca de la identidad del entorno de activos.

Nombre de la columna	Tipo de datos	Comentario
ENVIRONMENT_IDENTITY_CODE	varchar(5)	Código de identidad del entorno.
ENVIRONMENT_IDENTITY_NAME	varchar(255)	Nombre de identidad del entorno.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

ESEC_DISPLAY_RPT_V

La vista hace referencia a la tabla ESEC_DISPLAY que almacena las propiedades de los objetos que se pueden mostrar. Se utiliza actualmente para renombrar las meta-etiquetas. Se utiliza con la Configuración de eventos (Relevancia empresarial).

Nombre de la columna	Tipo de datos	Comentario
DISPLAY_OBJECT	VARCHAR(32)	El objeto padre de la propiedad.
TAG	VARCHAR(32)	El nombre de etiqueta nativa de la propiedad.
LABEL	VARCHAR(32)	La cadena de visualización de la etiqueta.
POSITION	INT	Posición de la etiqueta en la pantalla.
WIDTH	INT	El ancho de la columna.
ALIGNMENT	INT	La alineación horizontal.
FORMAT	INT	El formato enumerado para mostrar la propiedad.
ENABLED	BIT	Indica si se muestra la etiqueta.
TYPE	INT	Indica el tipo de dato de la etiqueta. 1 = string 2 = ulong 3 = fecha 4 = uuid 5 = ipv4
DESCRIPTION	VARCHAR(255)	Descripción textual de la etiqueta.

Nombre de la columna	Tipo de datos	Comentario
DATE_CREATED	DATETIME	Fecha de inserción.
DATE_MODIFIED	DATETIME	Fecha de la última actualización.
CREATED_BY	INT	ID del usuario que realiza la inserción.
MODIFIED_BY	INT	ID del usuario que realizó la última actualización.
REF_CONFIG	VARCHAR(4000)	Configuración de datos referenciales

ESEC_PORT_REFERENCE_RPT_V

La vista hace referencia a la tabla ESEC_PORT_REFERENCE que almacena números de puerto estándares de la industria asignados.

Nombre de la columna	Tipo de datos	Comentario
PORT_NUMBER	INT	Según http://www.iana.org/assignments/port-numbers, la representación numérica del puerto. Este número de puerto comúnmente se asocia con el nivel de Protocolo de transporte en la pila TCP/IP.
PROTOCOL_NUMBER	INT	Según http://www.iana.org/assignments/protocol-numbers, los identificadores numéricos utilizados para representar protocolos que están encapsulados en un paquete IP.
PORT_KEYWORD	VARCHAR(64)	Según http://www.iana.org/assignments/port-numbers, la palabra clave que representa al puerto.
PORT_DESCRIPTION	VARCHAR(512)	Descripción del puerto.
DATE_CREATED	DATETIME	Fecha de inserción.
DATE_MODIFIED	DATETIME	Fecha de la última actualización.
CREATED_BY	INT	ID del usuario que realiza la inserción.
MODIFIED_BY	INT	ID de usuario de la última modificación.

ESEC_PROTOCOL_REFERENCE_RPT_V

La vista hace referencia a la tabla ESEC_PROTOCOL_REFERENCE que almacena números de protocolo estándares de la industria asignados.

Nombre de la columna	Tipo de datos	Comentario
PROTOCOL_NUMBER	INT	Según http://www.iana.org/assignments/protocol-numbers, los identificadores numéricos utilizados para representar protocolos que están encapsulados en un paquete IP.
PROTOCOL_KEYWORD	VARCHAR(64)	Según http://www.iana.org/assignments/protocol-numbers, la palabra clave utilizada para representar protocolos que están encapsulados en un paquete IP.
PROTOCOL_DESCRIPTION	VARCHAR(512)	Descripción del protocolo del paquete IP.
DATE_CREATED	DATETIME	Fecha de inserción.
DATE_MODIFIED	DATETIME	Fecha de la última actualización.
CREATED_BY	INT	ID del usuario que realiza la inserción.
MODIFIED_BY	INT	ID del usuario que realizó la última actualización.

ESEC_SEQUENCE_RPT_V

La vista hace referencia a la tabla ESEC_SEQUENCE que se utiliza para generar números de secuencia de clave principal para las tablas de Sentinel.

Nombre de la columna	Tipo de datos	Comentario
TABLE_NAME	VARCHAR(32)	Nombre de la tabla.
COLUMN_NAME	VARCHAR(32)	Nombre de la columna.
SEED	INT	Valor actual del campo de clave principal.
DATE_CREATED	DATETIME	Fecha de inserción.
DATE_MODIFIED	DATETIME	Fecha de la última actualización.
CREATED_BY	INT	ID del usuario que realiza la inserción.
MODIFIED_BY	INT	ID del usuario que realizó la última actualización.

EVENTS_ALL_RPT_V (provisto para fines de compatibilidad con versiones anteriores)

La vista contiene eventos actuales e históricos (eventos importados de archivos).

Nombre de la columna	Tipo de datos	Comentario
EVENT_ID	uniqueidentifier	Identificador de evento.
RESOURCE_NAME	varchar(255)	Nombre de recurso.
SUB_RESOURCE	varchar(255)	Nombre de subrecurso.
SEVERITY	int	Gravedad del evento.
EVENT_PARSE_TIME	datetime	Fecha y hora del evento.
EVENT_DATETIME	datetime	Fecha y hora del evento.
BASE_MESSAGE	varchar(4000)	Mensaje de base.
EVENT_NAME	varchar(255)	Nombre del evento según lo informado por el sensor.
EVENT_TIME	varchar(255)	Fecha y hora del evento. según lo informado por el sensor.
SENSOR_NAME	varchar(255)	Nombre del sensor.
SENSOR_TYPE	varchar(5)	Tipo de sensor: H – de host N – de red V – virus O – otro
PROTOCOL	varchar(255)	Nombre de protocolo.
SOURCE_IP	int	Dirección IP de origen en formato numérico.
SOURCE_HOST_NAME	varchar(255)	Nombre de host de origen.
SOURCE_PORT	varchar(32)	Puerto de origen.
DESTINATION_IP	int	Dirección IP de destino. en formato numérico.
DESTINATION_HOST_NAME	varchar(255)	Nombre de host de destino.
DESTINATION_PORT	varchar(32)	Puerto de destino.
SOURCE_USER_NAME	varchar(255)	Nombre de usuario de origen.
DESTINATION_USER_NAME	varchar(255)	Nombre de usuario de destino.
FILE_NAME	varchar(1000)	Nombre de archivo.
EXTENDED_INFO	varchar(1000)	Información ampliada.
REPORT_NAME	varchar(255)	Nombre del informador.
PRODUCT_NAME	varchar(255)	Nombre del producto de generación de informes.
CUSTOM_TAG_1	varchar(255)	Etiqueta de cliente 1.
CUSTOM_TAG 2	varchar(255)	Etiqueta de cliente 2.
CUSTOM_TAG 3	int	Etiqueta de cliente 3.
RESERVED_TAG_1	VARCHAR(255)	Etiqueta reservada 1 Reservado para uso futuro de Sentinel. Este campo se utiliza para información de Asesor relativa a descripciones de ataques.

Nombre de la columna	Tipo de datos	Comentario
RESERVED_TAG_2	varchar(255)	Reservado para uso futuro de Sentinel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RESERVED_TAG_3	int	Reservado para uso futuro de Sentinel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
SOURCE_UUID	uniqueidentifier	UUID de origen.
PORT	varchar(64)	Puerto del recopilador.
AGENT	varchar(64)	Nombre del recopilador.
VULNERABILITY_RATING	int	Puntuación de vulnerabilidad.
CRITICALITY_RATING	int	Valoración de la importancia.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.
RV01 - 10	INT	Valor reservado de 1 a 10 Reservado para uso futuro de Sentinel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV11 - 20	DATETIME	Valor reservado de 11 a 20 Reservado para uso futuro de Sentinel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV21 - 25	uniqueidentifier	Valor reservado de 21 a 25 Reservado para uso futuro de Sentinel para almacenar UUID. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.

Nombre de la columna	Tipo de datos	Comentario
RV26 - 31	VARCHAR(255)	Valor reservado de 26 a 31 Reservado para uso futuro de Sentinel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV32	VARCHAR(255)	Valor reservado 32 Reservado para DeviceCategory El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV33	VARCHAR(255)	Valor reservado 33 Reservado para EventContex El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV34	VARCHAR(255)	Valor reservado 34 Reservado para SourceThreatLevel El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV35	VARCHAR(255)	Valor reservado 35 Reservado para SourceUserContext. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV36	VARCHAR(255)	Valor reservado 36 Reservado para DataContext. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.

Nombre de la columna	Tipo de datos	Comentario
RV37	VARCHAR(255)	Valor reservado 37 Reservado para SourceFunction. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV38	VARCHAR(255)	Valor reservado 38 Reservado para SourceOperationalContext. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV39	VARCHAR(255)	Valor reservado 39 Reservado para MSSPCustomerName. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV40 - 43	VARCHAR(255)	Valor reservado de 40 a 43 Reservado para uso futuro de Sentinel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV44	VARCHAR(255)	Valor reservado 44 Reservado para DestinationThreatLevel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV45	VARCHAR(255)	Valor reservado 45 Reservado para DestinationUserContext. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.

Nombre de la columna	Tipo de datos	Comentario
RV46	VARCHAR(255)	Valor reservado 46 Reservado para VirusStatus. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV47	VARCHAR(255)	Valor reservado 47 Reservado para uso futuro de Sentinel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV48	VARCHAR(255)	Valor reservado 48 Reservado para DestinationOperationalContext. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV49	VARCHAR(255)	Valor reservado 49 Reservado para uso futuro de Sentinel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV50	VARCHAR(255)	Taxonomía nivel 1.
RV51	VARCHAR(255)	Taxonomía nivel 2.
RV52	VARCHAR(255)	Taxonomía nivel 3.
RV53	VARCHAR(255)	Taxonomía nivel 4.
CV01 - 10	INT	Valor personalizado de 1 a 10 Reservado para uso del cliente, normalmente para asociación de datos comerciales relevantes.
CV11 - 20	DATETIME	Valor personalizado de 11 a 20 Reservado para uso del cliente, normalmente para asociación de datos comerciales relevantes.
CV21 100	VARCHAR(255)	Valor personalizado de 21 a 100 Reservado para uso del cliente, normalmente para asociación de datos comerciales relevantes.

EVENTS_ALL_RPT_V1 (provisto para fines de compatibilidad con versiones anteriores)

La vista contiene eventos actuales. Tiene las mismas columnas que EVENT_ALL_RPT_V.

EVENTS_RPT_V (provisto para fines de compatibilidad con versiones anteriores)

La vista contiene eventos actuales e históricos. Tiene las mismas columnas que EVENT_ALL_RPT_V.

EVENTS_RPT_V1 (provisto para fines de compatibilidad con versiones anteriores)

La vista contiene eventos actuales. Tiene las mismas columnas que EVENT_ALL_RPT_V.

EVENTS_RPT_V2 (provisto para fines de compatibilidad con versiones anteriores)

La vista contiene eventos actuales y eventos históricos.

Nombre de la columna	Tipo de datos	Comentario
EVENT_ID	uniqueidentifier	Identificador de evento.
RESOURCE_NAME	varchar(255)	Nombre de recurso.
SUB_RESOURCE	varchar(255)	Nombre de subrecurso.
SEVERITY	int	Gravedad del evento.
EVENT_PARSE_TIME	datetime	Fecha y hora del evento.
EVENT_DATETIME	datetime	Fecha y hora del evento.
BASE_MESSAGE	varchar(4000)	Mensaje de base.
EVENT_NAME	varchar(255)	Nombre del evento según lo informado por el sensor.
EVENT_TIME	varchar(255)	Fecha y hora del evento. según lo informado por el sensor.
TAXONOMY_ID	bigint	Identificador de taxonomía.
PROTOCOL_ID	bigint	Identificador de protocolo.
AGENT_ID	bigint	Identificador de recopilador.
SOURCE_IP	int	Dirección IP de origen en formato numérico.
SOURCE_HOST_NAME	varchar(255)	Nombre de host de origen.
SOURCE_PORT	varchar(32)	Puerto de origen.
DESTINATION_IP	int	Dirección IP de destino. en formato numérico.
DESTINATION_HOST_NAME	varchar(255)	Nombre de host de destino.
DESTINATION_PORT	varchar(32)	Puerto de destino.
SOURCE_USER_NAME	varchar(255)	Nombre de usuario de origen.
DESTINATION_USER_NAME	varchar(255)	Nombre de usuario de destino.
FILE_NAME	varchar(1000)	Nombre de archivo.
EXTENDED_INFO	varchar(1000)	Información ampliada.
CUSTOM_TAG_1	varchar(255)	Etiqueta de cliente 1.

Nombre de la columna	Tipo de datos	Comentario
CUSTOM_TAG 2	varchar(255)	Etiqueta de cliente 2.
CUSTOM_TAG 3	int	Etiqueta de cliente 3.
RESERVED_TAG_1	VARCHAR(255)	Etiqueta reservada 1. Reservado para uso futuro de Sentinel. Este campo se utiliza para información de Asesor relativa a descripciones de ataques.
RESERVED_TAG_2	varchar(255)	Reservado para uso futuro de Sentinel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RESERVED_TAG_3	int	Reservado para uso futuro de Sentinel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
VULNERABILITY_RATING	int	Puntuación de vulnerabilidad.
CRITICALITY_RATING	int	Valoración de la importancia.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.
RV01 - 10	INT	Valor reservado de 1 a 10 Reservado para uso futuro de Sentinel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV11 - 20	DATETIME	Valor reservado de 1 a 31 Reservado para uso futuro de Sentinel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV21 - 25	uniqueidentifier	Valor reservado de 21 a 25 Reservado para uso futuro de Sentinel para almacenar UUID. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV26 31	VARCHAR(255)	Valor reservado de 26 a 31 Reservado para uso futuro de Sentinel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.

Nombre de la columna	Tipo de datos	Comentario
RV33	VARCHAR(255)	Valor reservado 33 Reservado para EventContex El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV34	VARCHAR(255)	Valor reservado 34 Reservado para SourceThreatLevel El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV35	VARCHAR(255)	Valor reservado 35 Reservado para SourceUserContext. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV36	VARCHAR(255)	Valor reservado 36 Reservado para DataContext. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV37	VARCHAR(255)	Valor reservado 37 Reservado para SourceFunction. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV38	VARCHAR(255)	Valor reservado 38 Reservado para SourceOperationalContext. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV40 - 43	VARCHAR(255)	Valor reservado de 40 a 43 Reservado para uso futuro de Sentinel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV44	VARCHAR(255)	Valor reservado 44 Reservado para DestinationThreatLevel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.

Nombre de la columna	Tipo de datos	Comentario
RV45	VARCHAR(255)	Valor reservado 45 Reservado para DestinationUserContext. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV46	VARCHAR(255)	Valor reservado 46 Reservado para VirusStatus. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV47	VARCHAR(255)	Valor reservado 47 Reservado para uso futuro de Sentinel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV48	VARCHAR(255)	Valor reservado 48 Reservado para DestinationOperationalContext. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
RV49	VARCHAR(255)	Valor reservado 49 Reservado para uso futuro de Sentinel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
REFERENCE_ID 01 - 20	BIGINT	Reservado para uso futuro de Sentinel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
CV01 - 10	INT	Valor personalizado de 1 a 10 Reservado para uso del cliente, normalmente para asociación de datos comerciales relevantes
CV11 - 20	DATETIME	Valor personalizado de 11 a 20 Reservado para uso del cliente, normalmente para asociación de datos comerciales relevantes
CV21 100	VARCHAR(255)	Valor personalizado de 21 a 100 Reservado para uso del cliente, normalmente para asociación de datos comerciales relevantes

EVT_AGENT_RPT_V

La vista hace referencia a la tabla EVT_AGENT que almacena información acerca de recopiladores.

Nombre de la columna	Tipo de datos	Comentario
AGENT_ID	bigint	Identificador de recopilador.
AGENT	varchar(64)	Nombre del recopilador.
PORT	varchar(64)	Puerto del recopilador.
REPORT_NAME	varchar(255)	Nombre del informador.
PRODUCT_NAME	varchar(255)	Nombre de producto.
SENSOR_NAME	varchar(255)	Nombre del sensor.
SENSOR_TYPE	varchar(5)	Tipo de sensor: H - de host N - de red V - virus O – otro
DEVICE_CTGRY	varchar(255)	Categoría de dispositivo.
SOURCE_UUID	uniqueidentifier	UUID (Identificador exclusivo universal de componente de origen).
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

EVT_ASSET_RPT_V

La vista hace referencia a la tabla EVT_ASSET que almacena información de activos.

Nombre de la columna	Tipo de datos	Comentario
EVENT_ASSET_ID	bigint	Identificador de activo de evento.
ASSET_NAME	varchar(255)	Nombre de activo.
PHYSICAL_ASSET_NAME	varchar(255)	Nombre de activo físico.
REFERENCE_ASSET_ID	varchar(100)	Identificador de activo de referencia, enlaza con el sistema de administración de activos de origen.
MAC_ADDRESS	varchar(100)	Dirección MAC.
RACK_NUMBER	varchar(50)	Número de bastidor.
ROOM_NAME	varchar(100)	Nombre de sala.
BUILDING_NAME	varchar(255)	Nombre de edificio.
CITY	varchar(100)	Ciudad.
STATE	varchar(100)	Estado.
COUNTRY	varchar(100)	País.
ZIP_CODE	varchar(50)	Código postal.
ASSET_CATEGORY_NAME	varchar(100)	Nombre de categoría de activo.
NETWORK_IDENTITY_NAME	varchar(255)	Nombre de identidad de red de activo.
ENVIRONMENT_IDENTITY_NAME	varchar(255)	Nombre de entorno.

Nombre de la columna	Tipo de datos	Comentario
ASSET_VALUE_NAME	varchar(50)	Nombre de valor de activo.
CRITICALITY_NAME	varchar(50)	Nombre de importancia de activo.
SENSITIVITY_NAME	varchar(50)	Nombre de sensibilidad de activo.
CONTACT_NAME_1	varchar(255)	Nombre de organización/persona de contacto 1.
CONTACT_NAME_2	varchar(255)	Nombre de organización/persona de contacto 2.
ORGANIZATION_NAME_1	varchar(100)	Organización propietaria del activo nivel 1.
ORGANIZATION_NAME_2	varchar(100)	Organización propietaria del activo nivel 2.
ORGANIZATION_NAME_3	varchar(100)	Organización propietaria del activo nivel 3.
ORGANIZATION_NAME_4	varchar(100)	Organización propietaria del activo nivel 4.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

EVT_DEST_EVT_NAME_SMRY_1_RPT_V

La vista resume el total de eventos por destino, taxonomía, nombre de evento, gravedad y fecha y hora de evento.

Nombre de la columna	Tipo de datos	Comentario
DESTINATION_IP	int	Dirección IP de destino.
DESTINATION_EVENT_ASSET_ID	bigint	Identificador de activo de evento.
TAXONOMY_ID	bigint	Identificador de taxonomía.
EVENT_NAME_ID	bigint	Identificador de nombre de evento
SEVERITY	int	Gravedad del evento.
CUSTOMER_ID	bigint	Identificador de cliente.
EVT_TIME	datetime	Fecha y hora del evento.
EVT_COUNT	int	Total de eventos.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

EVT_DEST_SMRY_1_RPT_V

La vista contiene información de resumen de destino de eventos

Nombre de la columna	Tipo de datos	Comentario
DESTINATION_IP	int	Dirección IP de destino.
DESTINATION_EVENT_ASSET_ID	bigint	Identificador de activo de evento.
DESTINATION_PORT	varchar(32)	Puerto de destino
DESTINATION_USR_ID	bigint	Identificador de usuario de destino
TAXONOMY_ID	bigint	Identificador de taxonomía.
EVENT_NAME_ID	bigint	Identificador de nombre de evento
RESOURCE_ID	bigint	Identificador de recurso
AGENT_ID	bigint	Identificador de recopilador
PROTOCOL_ID	bigint	Identificador de protocolo.
SEVERITY	int	Gravedad del evento.
CUSTOMER_ID	bigint	Identificador de cliente.
EVENT_TIME	datetime	Fecha y hora del evento.
EVENT_COUNT	int	Total de eventos.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

EVT_DEST_TXNMY_SMRY_1_RPT_V

La vista resume el total de eventos por destino, taxonomía, gravedad y fecha y hora de evento.

Nombre de la columna	Tipo de datos	Comentario
DESTINATION_IP	int	Dirección IP de destino.
DESTINATION_EVENT_ASSET_ID	bigint	Identificador de activo de evento.
TAXONOMY_ID	bigint	Identificador de taxonomía.
SEVERITY	int	Gravedad del evento.
CUSTOMER_ID	bigint	Identificador de cliente.
EVENT_TIME	datetime	Fecha y hora del evento.
EVENT_COUNT	int	Total de eventos.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

EVT_NAME_RPT_V

La vista hace referencia a la tabla EVT_NAME que almacena información de nombres de eventos.

Nombre de la columna	Tipo de datos	Comentario
EVENT_NAME_ID	bigint	Identificador de nombre de evento
EVENT_NAME	varchar(255)	Nombre de evento
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

EVT_PORT_SMRY_1_RPT_V

La vista resume el total de eventos por puerto de destino, gravedad y fecha y hora de evento.

Nombre de la columna	Tipo de datos	Comentario
DESTINATION_PORT	Varchar(32)	Puerto de destino
SEVERITY	int	Gravedad del evento.
CUSTOMER_ID	bigint	Identificador de cliente.
EVENT_TIME	datetime	Fecha y hora del evento.
EVENT_COUNT	int	Total de eventos.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

EVT_PRTCL_RPT_V

La vista hace referencia a la tabla EVT_PRTCL que almacena información de protocolo de eventos.

Nombre de la columna	Tipo de datos	Comentario
PROTOCOL_ID	bigint	Identificador de protocolo.
PROTOCOL_NAME	varchar(255)	Nombre de protocolo
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

EVT_RSRC_RPT_V

La vista hace referencia a la tabla EVT_RCRS que almacena información de recursos de eventos.

Nombre de la columna	Tipo de datos	Comentario
RESOURCE_ID	bigint	Identificador de recurso
RESOURCE_NAME	varchar(255)	Nombre de recurso
SUB_RESOURCE_NAME	varchar(255)	Nombre de subrecurso
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

EVT_SEV_SMRY_1_RPT_V

La vista resume el total de eventos por gravedad y fecha y hora de evento.

Nombre de la columna	Tipo de datos	Comentario
SEVERITY	int	Gravedad del evento.
CUSTOMER_ID	bigint	Identificador de cliente.
EVENT_TIME	datetime	Fecha y hora del evento.
EVENT_COUNT	int	Total de eventos.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

EVT_SRC_SMRY_1_RPT_V

La vista contiene información de resumen de origen y destino del evento.

Nombre de la columna	Tipo de datos	Comentario
SOURCE_IP	int	Dirección IP de origen
SOURCE_EVENT_ASSET_ID	bigint	Identificador de activo de evento.
SOURCE_PORT	varchar(32)	Puerto de origen
SOURCE_USER_ID	bigint	Identificador de usuario
TAXONOMY_ID	bigint	Identificador de taxonomía.
EVENT_NAME_ID	bigint	Identificador de nombre de evento
RESOURCE_ID	bigint	Identificador de recurso
AGENT_ID	bigint	Identificador de recopilador
PROTOCOL_ID	bigint	Identificador de protocolo.
SEVERITY	int	Gravedad del evento.
CUSTOMER_ID	bigint	Identificador de cliente.
EVENT_TIME	datetime	Fecha y hora del evento.
EVENT_COUNT	int	Total de eventos.

Nombre de la columna	Tipo de datos	Comentario
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

EVT_TXNMY_RPT_V

La vista hace referencia a la tabla EVT_TXNMY que almacena información de taxonomía del evento.

Nombre de la columna	Tipo de datos	Comentario
TAXONOMY_ID	bigint	Identificador de taxonomía.
TAXONOMY _ LEVEL _1	varchar(100)	Taxonomía nivel 1.
TAXONOMY _ LEVEL _2	varchar(100)	Taxonomía nivel 2.
TAXONOMY _ LEVEL _3	varchar(100)	Taxonomía nivel 3.
TAXONOMY _ LEVEL _4	varchar(100)	Taxonomía nivel 4.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.
TAXONOMY_ID	bigint	Identificador de taxonomía.

EVT_USR_RPT_V

La vista hace referencia a la tabla EVT_USR que almacena información de usuarios de eventos.

Nombre de la columna	Tipo de datos	Comentario
USER_ID	bigint	Identificador de usuario.
USER_NAME	varchar(255)	Nombre de usuario.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.
USER_ID	bigint	Identificador de usuario.

EXTERNAL_DATA_RPT_V

La vista hace referencia a la tabla EXTERNAL_DATA que almacena datos externos.

Nombre de la columna	Tipo de datos	Comentario
EXTERNAL_DATA_ID	int	Identificador de datos externos.
SOURCE_NAME	varchar(50)	Nombre de origen.
SOURCE_DATA_ID	varchar(255)	Identificador de datos de origen.
EXTERNAL_DATA	text	Datos externos.
EXTERNAL_DATA_TYPE	varchar(10)	Tipo de datos externos.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

HIST_EVENTS_RPT_V

Vista de eventos históricos (eventos restaurados de archivos).

HIST_INCIDENTS_RPT_V

Vista de incidencias históricas (incidencias restauradas de archivos).

IMAGES_RPT_V

La vista hace referencia a la tabla IMAGES que almacena información de imágenes descriptivas generales del sistema.

Nombre de la columna	Tipo de datos	Comentario
NAME	VARCHAR(128)	Nombre de imagen.
TYPE	VARCHAR(64)	Tipo de imagen.
DATA	TEXT	Datos de imagen.
DATE_CREATED	DATETIME	Fecha de inserción.
DATE_MODIFIED	DATETIME	Fecha de la última actualización.
CREATED_BY	INT	ID del usuario que realiza la inserción.
MODIFIED_BY	INT	ID del usuario que realizó la última actualización.

INCIDENTS_ASSETS_RPT_V

La vista hace referencia a la tabla INCIDENTS_ASSETS que almacena información sobre activos que componen incidencias creadas en la consola de Sentinel.

Nombre de la columna	Tipo de datos	Comentario
INC_ID	INT	Identificador de incidencia - número de secuencia.
ASSET_ID	uniqueidentifier	UUID (Identificador exclusivo universal de activo).
DATE_CREATED	DATETIME	Fecha de inserción.
DATE_MODIFIED	DATETIME	Fecha de la última actualización.
CREATED_BY	INT	ID del usuario que realiza la inserción.
MODIFIED_BY	INT	ID del usuario que realizó la última actualización.

INCIDENTS_EVENTS_RPT_V

La vista hace referencia a la tabla INCIDENTS_EVENTS que almacena información sobre eventos que componen incidencias creadas en la consola de Sentinel.

Nombre de la columna	Tipo de datos	Comentario
INC_ID	INT	Identificador de incidencia - número de secuencia.
EVT_ID	uniqueidentifier	UUID (Identificador exclusivo universal de evento).
EVT_TIME	DATETIME	Fecha y hora del evento.
DATE_CREATED	DATETIME	Fecha de inserción.
DATE_MODIFIED	DATETIME	Fecha de la última actualización.
CREATED_BY	INT	ID del usuario que realiza la inserción.
MODIFIED_BY	INT	ID del usuario que realizó la última actualización.

INCIDENTS_RPT_V

La vista hace referencia a la tabla INCIDENTS que almacena información que describe los detalles de incidencias creadas en la consola de Sentinel.

Nombre de la columna	Tipo de datos	Comentario
INC_ID	INT	Identificador de incidencia – número de secuencia.
NAME	VARCHAR(255)	Nombre de incidencia.
SEVERITY	INT	Gravedad de la incidencia.
STT_ID	INT	ID de estado de incidencia.
SEVERITY_RATING	VARCHAR(32)	Media de la gravedad de todos los eventos que componen una incidencia.
VULNERABILITY_RATING	VARCHAR(32)	Reservado para uso futuro de Sentinel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
CRITICALITY_RATING	VARCHAR(32)	Reservado para uso futuro de Sentinel. El uso de este campo para cualquier otro propósito puede resultar en que una funcionalidad posterior sobrescriba los datos.
DATE_CREATED	DATETIME	Fecha de inserción.
DATE_MODIFIED	DATETIME	Fecha de la última actualización.
CREATED_BY	INT	ID del usuario que realiza la inserción.
MODIFIED_BY	INT	ID del usuario que realizó la última actualización.
INC_DESC	varchar(4000)	Descripción de la incidencia.
INC_PRIORITY	int	Prioridad de la incidencia
INC_CAT	varchar(255)	Categoría de la incidencia.
INC_RES	varchar(4000)	Resolución de la incidencia.

INCIDENTS_VULN_RPT_V

La vista hace referencia a la tabla INCIDENTS_VULN que almacena información sobre vulnerabilidades que componen incidencias creadas en la consola de Sentinel.

Nombre de la columna	Tipo de datos	Comentario
INC_ID	INT	Identificador de incidencia - número de secuencia.
VULN_ID	uniqueidentifier	UUID (Identificador exclusivo universal de vulnerabilidad).
DATE_CREATED	DATETIME	Fecha de inserción.
DATE_MODIFIED	DATETIME	Fecha de la última actualización.
CREATED_BY	INT	ID del usuario que realiza la inserción.
MODIFIED_BY	INT	ID del usuario que realizó la última actualización.

L_STAT_RPT_V

La vista hace referencia a la tabla L_STAT que almacena información estadística.

Nombre de la columna	Tipo de datos	Comentario
RES_NAME	VARCHAR(32)	Nombre de recurso.
STATS_NAME	VARCHAR(32)	Nombre de estadística.
STATS_VALUE	VARCHAR(32)	Valor de la estadística
OPEN_TOT_SECS	NUMERIC	Número de segundos desde 1970.

LOGS_RPT_V

La vista hace referencia a la tabla LOGS_RPT que almacena información de registro.

Tabla LOGS		
Nombre de la columna	**Tipo de datos**	**Comentario**
LOG_ID	NUMBER	Número de secuencia
TIME	FECHA	Fecha de registro.
MODULE	VARCHAR(64)	Módulo al que corresponde el registro.
TEXT	VARCHAR(4000)	Texto de registro.

NETWORK_IDENTITY_RPT_V

La vista hace referencia a la tabla NETWORK_IDENTITY_LKUP que almacena información acerca de la identidad de red de activos.

Nombre de la columna	Tipo de datos	Comentario
NETWORK_IDENTITY_CD	varchar(5)	Código de identidad de red.
NETWORK_IDENTITY_NAME	varchar(255)	Nombre de identidad de red.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

ORGANIZATION_RPT_V

La vista hace referencia a la tabla ORGANIZATION que almacena información acerca de organizaciones (activos).

Nombre de la columna	Tipo de datos	Comentario
ORGANIZATION_ID	uniqueidentifier	Identificador de organización.
ORGANIZATION_NAME	varchar(100)	Nombre de organización.
CUSTOMER_ID	bigint	Identificador de cliente.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

PERSON_RPT_V

La vista hace referencia a la tabla PERSION que almacena información de personas (activos).

Nombre de la columna	Tipo de datos	Comentario
PERSON_ID	uniqueidentifier	Identificador de persona.
FIRST_NAME	varchar(255)	Nombre.
LAST_NAME	varchar(255)	Apellidos.
CUSTOMER_ID	bigint	Identificador de cliente.
PHONE_NUMBER	varchar(50)	Número de teléfono.
EMAIL_ADDRESS	varchar(255)	Dirección de correo electrónico.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

PHYSICAL_ASSET_RPT_V

La vista hace referencia a la tabla PHYSICAL_ASSET que almacena información de activos físicos.

Nombre de la columna	Tipo de datos	Comentario
PHYSICAL_ASSET_ID	uniqueidentifier	Identificador de activo físico.
CUSTOMER_ID	int	Identificador de cliente.
LOCATION_ID	bigint	Identificación de ubicación.
HOST_NAME	varchar(255)	Nombre del host.
IP_ADDRESS	int	Dirección IP.
NETWORK_IDENTITY_CD	varchar(5)	Código de identidad de red.
MAC_ADDRESS	varchar(100)	Dirección MAC.
RACK_NUMBER	varchar(50)	Número de bastidor.
ROOM_NAME	varchar(100)	Nombre de sala.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

PRODUCT_RPT_V

La vista hace referencia a la tabla PRDT que almacena información de productos de activos.

Nombre de la columna	Tipo de datos	Comentario
PRODUCT_ID	bigint	Identificador de producto.
PRODUCT_NAME	varchar(255)	Nombre de producto.
PRODUCT_VERSION	varchar(100)	Versión de producto.
VENDOR_ID	bigint	Identificador de proveedor.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

ROLE_RPT_V

La vista hace referencia a la tabla ROLE_LKUP que almacena información de funciones de usuarios (activos).

Nombre de la columna	Tipo de datos	Comentario
ROLE_CODE	varchar(5)	Código de función.
ROLE_NAME	varchar(255)	Nombre de la función.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

SENSITIVITY_RPT_V

La vista hace referencia a la tabla SENSITIVITY_LKUP que almacena información acerca de la sensibilidad de activos.

Nombre de la columna	Tipo de datos	Comentario
SENSITIVITY_CODE	varchar(5)	Código de sensibilidad de activo.
SENSITIVITY_NAME	varchar(50)	Nombre de sensibilidad de activo.
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la acción.
MODIFIED_BY	int	ID del usuario que realiza la acción.

STATES_RPT_V

La vista hace referencia a la tabla STATES que almacena definiciones de los estados determinados por aplicaciones o contexto.

Nombre de la columna	Tipo de datos	Comentario
STT_ID	INT	ID de estado - número de secuencia.
CONTEXT	VARCHAR(64)	Contexto del estado Es decir caso, incidencia, usuario.
NAME	VARCHAR(64)	Nombre del estado.

Nombre de la columna	Tipo de datos	Comentario
TERMINAL_FLAG	VARCHAR(1)	Indica si el estado de la incidencia es resuelto.
DATE_CREATED	DATETIME	Fecha de inserción.
DATE_MODIFIED	DATETIME	Fecha de la última actualización.
MODIFIED_BY	INT	ID del usuario que realiza la inserción.
CREATED_BY	INT	ID del usuario que realizó la última actualización.

UNASSIGNED_INCIDENTS_RPT_V

La vista hace referencia a las tablas CASES e INCIDENTS para informar sobre casos no asignados.

Nombre	Tipo de datos
INC_ID	INT
NAME	VARCHAR(255)
SEVERITY	INT
STT_ID	INT
SEVERITY_RATING	VARCHAR(32)
VULNERABILITY_RATING	VARCHAR(32)
CRITICALITY_RATING	VARCHAR(32)
DATE_CREATED	DATETIME
DATE_MODIFIED	DATETIME
CREATED_BY	INT
MODIFIED_BY	INT
INC_DESC	VARCHAR(4000)
INC_PRIORITY	INT
INC_CAT	VARCHAR(255)
INC_RES	VARCHAR(4000)

USERS_RPT_V

La vista hace referencia a la tabla USERS que detalla todos los usuarios de la aplicación. Los usuarios también se crearán como usuarios de base de datos para adaptarse a herramientas de generación de informes de terceros.

Nombre de la columna	Tipo de datos	Comentario
USR_ID	INT	Identificador de usuario - número de secuencia.
NAME	VARCHAR(64)	Corto, nombre de usuario exclusivo utilizado como inicio de sesión.
CNT_ID	INT	ID de contacto - Número de secuencia
STT_ID	INT	ID de estado. El estado puede ser activo o inactivo.
DESCRIPTION	VARCHAR(512)	Comentarios.
DATE_CREATED	DATETIME	Fecha de inserción.
DATE_MODIFIED	DATETIME	Fecha de la última actualización.
CREATED_BY	INT	ID del usuario que realiza la inserción.
MODIFIED_BY	INT	ID del usuario que realizó la última actualización.

Nombre de la columna	Tipo de datos	Comentario
PERMISSIONS	VARCHAR(4000)	Permisos actualmente asignados al usuario de Sentinel.
FILTER	VARCHAR(128)	Filtro de seguridad actual asignado al usuario de Sentinel.
UPPER_NAME	VARCHAR(64)	Nombre de usuario en mayúsculas.
DOMAIN_AUTH_IND	Bit	Indicación de autenticación de dominio.

VENDOR_RPT_V

La vista hace referencia a la tabla VNDR que almacena información de proveedores de productos de activos.

Nombre de la columna	Tipo de datos	Comentario
VENDOR_ID	bigint	Identificador de proveedor.
VENDOR_NAME	varchar(255)	Nombre de proveedor
DATE_CREATED	datetime	Fecha de inserción.
DATE_MODIFIED	datetime	Fecha de la última actualización.
CREATED_BY	int	ID del usuario que realiza la inserción.
MODIFIED_BY	int	ID del usuario que realizó la última actualización.

VULN_CALC_SEVERITY_RPT_V

La vista hace referencia a VULN_RSRC y VULN para calcular la calificación de gravedad de vulnerabilidad de eSecurity en función de las vulnerabilidades actuales.

Nombre de la columna	Tipo de datos
RSRC_ID	uniqueidentifier
IP	VARCHAR(32)
HOST_NAME	VARCHAR(255)
CRITICALITY	int
ASSIGNED_VULN_SEVERITY	int
VULN_COUNT	Total de vulnerabilidades para el recurso especificado.
CALC_SEVERITY	Gravedad calculada en función de ASSIGNED_VULN_SEVERITY y CRITICALITY.

VULN_CODE_RPT_V

La vista hace referencia a la tabla VULN_CODE que almacena códigos de vulnerabilidad asignados por la industria como CVE y CAN de Mitre.

Nombre de la columna	Tipo de datos
VULN_CODE_ID	VARCHAR(36)
VULN_ID	VARCHAR(36)
VULN_CODE_TYPE	VARCHAR(64)
VULN_CODE_VALUE	VARCHAR(255)
URL	VARCHAR(512)
DATE_CREATED	DATETIME
DATE_MODIFIED	DATETIME
CREATED_BY	INT
MODIFIED_BY	INT

VULN_INFO_RPT_V

La vista hace referencia a la tabla VULN_INFO que almacena información adicional recogida durante una exploración.

Nombre de la columna	Tipo de datos
VULN_INFO_ID	VARCHAR(36)
VULN_ID	VARCHAR(36)
VULN_INFO_TYPE	VARCHAR(36)
VULN_INFO_VALUE	VARCHAR(2000)
DATE_CREATED	DATETIME
DATE_MODIFIED	DATETIME
CREATED_BY	INT
MODIFIED_BY	INT

VULN_RPT_V

La vista hace referencia a la tabla VULN que almacena información del sistema explorado. Cada escáner tendrá su propia entrada para cada sistema.

Nombre de la columna	Tipo de datos
VULN_ID	VARCHAR(36)
RSRC_ID	VARCHAR(36)
PORT_NAME	VARCHAR(64)
PORT_NUMBER	INT
NETWORK_PROTOCOL	INT
APPLICATION_PROTOCOL	VARCHAR(64)
ASSIGNED_VULN_SEVERITY	INT
COMPUTED_VULN_SEVERITY	INT
VULN_DESCRIPTION	CLOB
VULN_SOLUTION	CLOB
VULN_SUMMARY	VARCHAR(1000)
BEGIN_EFFECTIVE_DATE	DATETIME
END_EFFECTIVE_DATE	DATETIME
DETECTED_OS	VARCHAR(64)
DETECTED_OS_VERSION	VARCHAR(64)
SCANNED_APP	VARCHAR(64)
SCANNED_APP_VERSION	VARCHAR(64)
VULN_USER_NAME	VARCHAR(64)
VULN_USER_DOMAIN	VARCHAR(64)
VULN_TAXONOMY	VARCHAR(1000)
SCANNER_CLASSIFICATION	VARCHAR(255)
VULN_NAME	VARCHAR(300)
VULN_MODULE	VARCHAR(64)
DATE_CREATED	DATETIME
DATE_MODIFIED	DATETIME
CREATED_BY	INT
MODIFIED_BY	INT

VULN_RSRC_RPT_V

La vista hace referencia a la tabla VULN_RSRC que almacena cada recurso explorado para una exploración en particular.

Nombre de la columna	Tipo de datos
RSRC_ID	VARCHAR(36)
SCANNER_ID	VARCHAR(36)
IP	VARCHAR(32)
HOST_NAME	VARCHAR(255)
LOCATION	VARCHAR(128)
DEPARTMENT	VARCHAR(128)
BUSINESS_SYSTEM	VARCHAR(128)
OPERATIONAL_ENVIRONMENT	VARCHAR(64)
CRITICALITY	INT
REGULATION	VARCHAR(128)
REGULATION_RATING	VARCHAR(64)
DATE_CREATED	DATETIME
DATE_MODIFIED	DATETIME
CREATED_BY	INT
MODIFIED_BY	INT

VULN_RSRC_SCAN_RPT_V

La vista hace referencia a la tabla VULN_RSRC_SCAN que almacena cada recurso explorado para una exploración en particular.

Nombre de la columna	Tipo de datos
RSRC_ID	VARCHAR(36)
SCAN_ID	VARCHAR(36)
DATE_CREATED	DATETIME
DATE_MODIFIED	DATETIME
CREATED_BY	INT
MODIFIED_BY	INT

VULN_SCAN_RPT_V

La vista hace referencia a la tabla que almacena información perteneciente a las exploraciones.

Nombre de la columna	Tipo de datos
SCAN_ID	VARCHAR(36)
SCANNER_ID	VARCHAR(36)
SCAN_TYPE	VARCHAR(10)
SCAN_START_DATE	DATETIME
SCAN_END_DATE	DATETIME
CONSOLIDATION_SERVER	VARCHAR(64)
DATE_CREATED	DATETIME
DATE_MODIFIED	DATETIME
CREATED_BY	INT
MODIFIED_BY	INT

VULN_SCAN_VULN_RPT_V

La vista hace referencia a la tabla VULN_SCAN_VULN que almacena las vulnerabilidades detectadas durante las exploraciones.

Nombre de la columna	Tipo de datos
SCAN_ID	VARCHAR(36)
VULN_ID	VARCHAR(36)
DATE_CREATED	DATETIME
DATE_MODIFIED	DATETIME
CREATED_BY	INT
MODIFIED_BY	INT

VULN_SCANNER_RPT_V

La vista hace referencia a la tabla VULN_SCANNER que almacena información de escáneres de vulnerabilidades.

Nombre de la columna	Tipo de datos
SCANNER_ID	VARCHAR(36)
PRODUCT_NAME	VARCHAR(100)
PRODUCT_VERSION	VARCHAR(64)
SCANNER_TYPE	VARCHAR(64)
VENDOR	VARCHAR(100)
SCANNER_INSTANCE	VARCHAR(64)
DATE_CREATED	DATETIME
DATE_MODIFIED	DATETIME
CREATED_BY	INT
MODIFIED_BY	INT

A Lista de verificación para la resolución de problemas en Sentinel

NOTA: El término Agente puede intercambiarse con Recopilador. En adelante, los agentes se denominarán recopiladores.

Se ofrece esta lista de verificación para facilitar el diagnóstico de un problema. Al completar esta lista de verificación, los problemas más comunes pueden resolverse en menos tiempo. Los problemas cuya resolución demande más tiempo ya tendrán recopilada la información de diagnóstico, para eliminar la repetición del trabajo de este tipo de trabajo.

Elemento de lista de verificación	Información	Ejemplo
Versión de Novell:		v5.1.3
Versión del SO y plataforma Novell:		Win2003 Server sp1
Versión del SO y plataforma de base de datos:		MS SQL 2000 sp3a
Configuración del hardware del servidor de Sentinel ▪ Procesador ▪ Memoria ▪ Otros		5 GB de RAM 4 CPU 3.0 GHz
Configuración del hardware del servidor de la base de datos ▪ Procesador ▪ Memoria ▪ Otros (si existe un cuadro independiente)		8 GB de RAM 4 CPU 3.0 GHz
Configuración de almacenamiento de la base de datos (NAS, SAN, Local, etc.)		Local con copia de seguridad externa
Configuración y SO del servidor de informes (Crystal Server)		Crystal XI Win2003 Server sp1 Autenticación Windows

NOTA: En función de cómo esté configurado (distribuido) el sistema Sentinel, es posible que deba expandir la tabla anterior. Por ejemplo, se puede necesitar información adicional para DAS, el Asesor, el Centro de control del Sentinel, el Generador de recopiladores y el nivel de comunicaciones.

1. Consulte el portal de Asistencia técnica al cliente para obtener más detalles sobre el problema en particular:
   - ¿Es un problema conocido con una solución?
   - ¿Este problema tiene solución en la versión de revisión o Hot Fix más reciente?
   - ¿Se ha programado una resolución para este problema en una versión posterior?
2. Determinar la naturaleza del problema.
   - ¿Se puede reproducir? ¿Se pueden enumerar los pasos para reproducir el problema?
   - ¿Qué acción del usuario, si la hubiera, causará el problema?
   - ¿El problema es periódico por naturaleza?
3. Determinar la gravedad de este problema.
   - ¿El sistema puede seguir utilizándose?
4. Comprender el entorno y los sistemas involucrados.
   - ¿Qué plataformas y versiones de productos están involucrados?
   - ¿Hay componentes personalizados o no estándar involucrados?
   - ¿Es un entorno con alta frecuencia de eventos?
   - ¿A qué velocidad se recopilan los eventos?
   - ¿A qué velocidad se insertan los eventos en la base de datos?
   - ¿Cuántos usuarios simultáneos hay?
   - ¿Se utilizan los informes de Crystal? ¿Cuándo se generan los informes?
   - ¿Se utilizan las correlaciones? ¿Cuántas reglas se distribuyen?

   Recopile archivos de configuración, archivos de registro e información del sistema. Reúna esta información para posibles intercambios de conocimiento futuros. Si desea obtener información sobre la ubicación de los archivos de registro, consulte la Guía de instalación de Sentinel, Capítulo 2: Prácticas recomendadas.
5. Comprobar la actividad del sistema.
   - ¿Puede iniciar sesión en la consola de Sentinel?
   - ¿Se generan e insertan eventos en la base de datos? (si aún está configurado, ejecute SendOneEvent y observe estos eventos)
   - ¿Se pueden ver los eventos en la consola de Sentinel?
   - ¿Se pueden recuperar los eventos de la base de datos mediante una consulta rápida?
   - Compruebe el uso de memoria RAM, el espacio en disco, la actividad de procesos, el uso de CPU y la conectividad de red de los hosts involucrados.
   - Compruebe que todos los procesos de Sentinel esperados estén en ejecución. Los guiones como hp_checkprocess en Solaris enumerarán los procesos y su estado. Se puede utilizar el gestor de tareas de Microsoft en un entorno de Windows.
   - Compruebe si existen volcados del núcleo en alguno de los subdirectorios de ESEC_HOME. Averigüe en qué proceso se produjo el volcado del núcleo. (cd $ESEC_HOME, buscar . –nombre del núcleo –imprimir)
   - Compruebe el acceso de red sqlplus. Compruebe los espacios de tabla.

- Asegúrese de que el intermediario de Sonic esté en ejecución. Para verificar la conectividad se puede utilizar la consola de gestión de Sonic. Compruebe que las diversas conexiones se encuentren activas desde los procesos de Novell. Asegúrese de que no haya un archivo de bloqueo que impida el inicio de Sonic. De manera opcional, intente realizar una operación de telnet a ese servidor en el puerto de Sonic (p. ej. realice una operación de telnet en sentinel.company.com 10012)
- Compruebe si se está ejecutando un paquete de vigilancia en el servidor. (paquete de vigilancia ps –ef | grep)
- Compruebe si los procesos del asistente están funcionando. ¿Se está ejecutando el Gestor de recopiladores? ¿El Gestor de recopiladores aparece activo en el Generador de recopiladores o en la consola de Sentinel? ¿Se están ejecutando los recopiladores? ¿Cuántos por máquina? ¿Qué conectores se están utilizando (archivos, procesos, syslog, cortafuegos, registro de eventos, etc.)? ¿Cuál es el consumo de recursos del sistema operativo?

6. ¿Hay algún problema con la base de datos?
   - Si utiliza sqlplus, ¿puede iniciar sesión en la base de datos?
   - ¿La base de datos admite una entrada sqlplus con la cuenta dba de Novell en el esquema ESEC?
     - ¿Las consultas en una de las tablas se realizan correctamente?
   - ¿Se realiza correctamente una petición de selección en una tabla de la base de datos?
   - Compruebe los controladores JDBC, las ubicaciones y los valores de ruta de clase.
   - Si es Oracle, ¿están instaladas y se utilizan las particiones (escriba “seleccionar * de v$versión;”)?
   - ¿La base de datos la mantiene un administrador? ¿Alguna persona?
   - ¿La base de datos ha sido modificada por un administrador?
   - ¿Se utiliza SDM para mantener las particiones y archivar o eliminar las particiones a fin de ganar más espacio en la base de datos?
   - ¿Cuál es la partición actual al utilizar SDM? ¿Es PMAX?
7. Revisar si los valores de entorno del producto son correctos.
   - Comprobar la actividad de los guiones de shell de entrada de usuario, variables de entorno, configuraciones y valores java home.
   - ¿Las variables de entorno están configuradas para ejecutar la opción jvm correcta?
   - Verifique los permisos apropiados en las carpetas para el producto instalado.
   - Compruebe si hay tareas cron configuradas que ocasionan interferencias con la funcionalidad de nuestro producto.
   - Si el producto está instalado en montajes NFS, compruebe la actividad de los montajes NFS y los servicios NFS/NIS.

8. ¿Hay una posible pérdida de memoria?
   - Obtenga estadísticas sobre la velocidad con la que se utiliza la memoria y cuáles son los procesos.
   - Reúna la medición del rendimiento de eventos por recopilador.
   - Ejecute el comando prstat en Solaris. De esta manera obtendrá las estadísticas del tiempo de ejecución del proceso.
   - En Windows puede comprobar el tamaño del proceso y manejar los recuentos en el gestor de tareas.

   Este problema, si aún no está resuelto, está listo para elevación. Los posibles resultados de la elevación son:
   - Mejoras
   - Revisiones Hot Fix
   - Soluciones temporales

B Configuración de la cuenta de inicio de sesión del servicio de Sentinel como NT AUTHORITY\NetworkService

NOTA: El término Agente puede intercambiarse con Recopilador. En adelante, los agentes se denominarán recopiladores.

El propósito de este documento es describir en detalle cómo configurar la cuenta de inicio de sesión del servicio de Sentinel como NT AUTHORITY\NetworkService en lugar de cómo cuenta de usuario de dominio. Se ha comprobado su funcionamiento con la plataforma Windows 2003 únicamente.

Un servicio debe iniciar sesión en una cuenta para acceder a los recursos y objetos del sistema operativo. Si se selecciona una cuenta que no tiene permiso para iniciar sesión como un servicio, los módulos integrados de servicios automáticamente otorgan a dicha cuenta los derechos de usuario requeridos para iniciar sesión como un servicio en el equipo que está administrando. Sin embargo, esto no garantiza que el servicio se inicie. Se recomienda que las cuentas de usuario que se utilizan para iniciar sesión como un servicio tenga activada la casilla de verificación **Contraseña sin caducidad** en el cuadro de diálogo de propiedades y que las contraseñas sean seguras. Si la directiva de bloqueo de cuenta se encuentra habilitada y la cuenta se bloquea, el servicio no funcionará correctamente

En la tabla siguiente se describen las cuentas de inicio de sesión del servicio y cómo se utilizan.

Cuenta de inicio de sesión	Descripción
Cuenta del sistema local	La cuenta del sistema local es una cuenta potente con acceso completo al sistema, incluidos los Servicios de Directorio en los controladores de dominio. Si un servicio inicia sesión en la Cuenta del sistema local en un controlador de dominio, dicho servicio tiene acceso a todo el dominio. Algunos servicios están configurados por defecto para iniciar sesión en la Cuenta del sistema local. No cambie el valor de servicio por defecto. La cuenta del sistema local es una cuenta local predefinida que se utiliza para iniciar un servicio y brindar el contexto de seguridad para dicho servicio. El nombre de la cuenta es NT AUTHORITY\System. Esta cuenta no posee contraseña y cualquier información de contraseña que se ingrese se ignora. La cuenta del sistema local posee acceso completo al sistema, incluidos los Servicios de Directorio en los controladores de dominio. Dado que la cuenta del sistema local funciona como un equipo en la red, posee acceso a los recursos de red.

Cuenta de inicio de sesión	Descripción
Cuenta del servicio local	La cuenta del servicio local es una cuenta especial incorporada similar a una cuenta de usuario autenticada. La cuenta del servicio local posee el mismo nivel de acceso a los recursos y objetos que los miembros del grupo Usuarios. Este acceso limitado permite proteger el sistema si los servicios o procesos individuales se encuentran comprometidos. Los servicios que se ejecutan como la cuenta del servicio local acceden a los recursos de red como una sesión nula sin credenciales. La cuenta del servicio local es una cuenta local predefinida que se utiliza para iniciar un servicio y brindar el contexto de seguridad para dicho servicio. El nombre de la cuenta es NT AUTHORITY\LocalService. La cuenta del servicio local posee acceso limitado al equipo local y acceso Anónimo a los recursos de red.
Cuenta del servicio de red	La cuenta del servicio de red es una cuenta especial incorporada similar a una cuenta de usuario autenticada. La cuenta del servicio de red posee el mismo nivel de acceso a los recursos y objetos que los miembros del grupo Usuarios. Este acceso limitado permite proteger el sistema si los servicios o procesos individuales se encuentran comprometidos. Los servicios que se ejecutan como la cuenta del servicio de red acceden a los recursos de red utilizando las credenciales de la cuenta del equipo. La cuenta del servicio de red es una cuenta local predefinida que se utiliza para iniciar un servicio y brindar el contexto de seguridad para dicho servicio. El nombre de la cuenta es NT AUTHORITY\NetworkService. La cuenta del servicio de red posee acceso limitado al equipo local y acceso autenticado (como la cuenta del equipo) a los recursos de red.

La ejecución de un servicio en el contexto de una cuenta de inicio de sesión de usuario presenta las siguientes desventajas:

1. La cuenta se debe crear antes de que se pueda ejecutar el servicio. Si el programa de configuración del servicio crea la cuenta, la Configuración se debe ejecutar desde una cuenta que posea credenciales administrativas suficientes para crear cuentas en el Servicio de Directorio.
2. Los nombres de cuentas y contraseñas del servicio se almacenan en cada equipo en el que está instalado el servicio. Si la contraseña de una cuenta de servicio en un equipo cambia o vence, el servicio no se puede iniciar en dicho equipo hasta que se defina la contraseña nueva para dicho servicio. Se recomienda utilizar el servicio local y el servicio de red en lugar de usar una cuenta que requiera una contraseña: esto simplifica la administración de contraseñas.

3. Si se cambia el nombre, bloquea, deshabilita o elimina una cuenta de servicio, el servicio no se puede iniciar en dicho equipo hasta que se restaure la cuenta.

Debido a las desventajas anteriores, Novell ha probado la ejecución del servicio de Sentinel bajo la cuenta NT AUTHORITY\NetworkService. La cuenta NT AUTHORITY\LocalService no posee privilegios suficientes para este fin, ya que los procesos de DAS deben comunicarse con el servidor de la base de datos en la red.

Para configurar NT AUTHORITY\NetworkService como la cuenta de inicio de sesión para el servicio de Sentinel

Para configurar NT AUTHORITY\NetworkService como la cuenta de inicio de sesión para el servicio de Sentinel, deberá realizar lo siguiente:

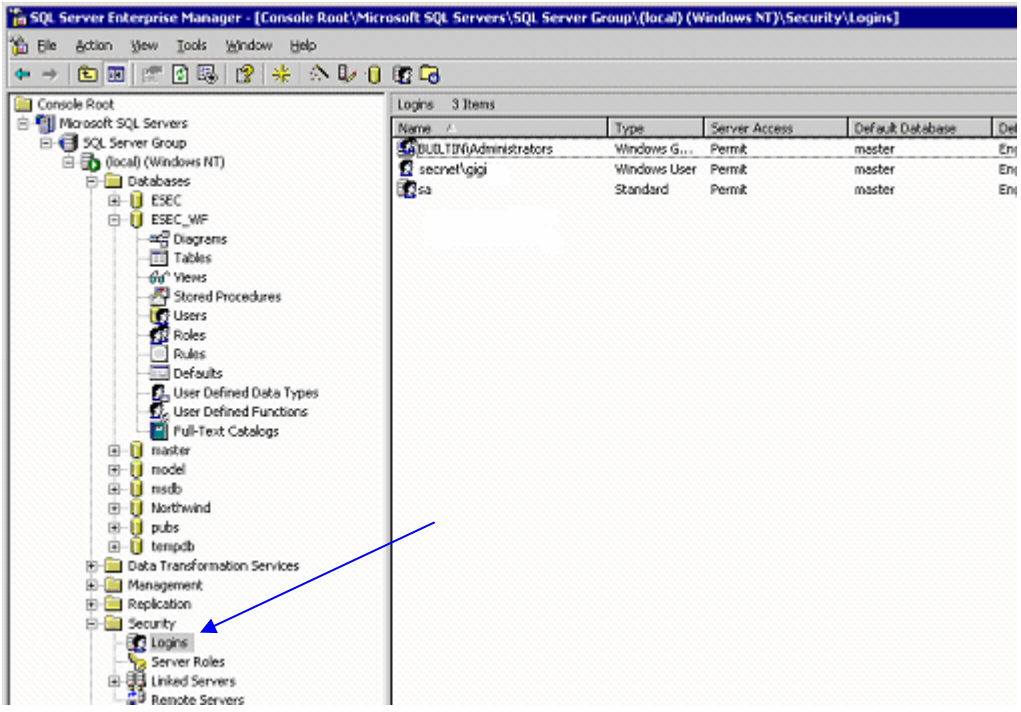
- Agregar la máquina que ejecuta el servicio de Sentinel como la cuenta de entrada a las instancias de la base de datos ESEC y ESEC_WF (ejecutadas en la máquina de la base de datos)
- Cambiar la cuenta de inicio de sesión para el servicio de Sentinel a NT AUTHORITY\NetworkService (que se ejecuta en la máquina remota)
- Configurar el inicio de Sentinel (que se ejecuta en la máquina remota)

Adición del servicio de Sentinel como una cuenta de inicio de sesión en las instancias de la base de datos ESEC y ESEC_WF

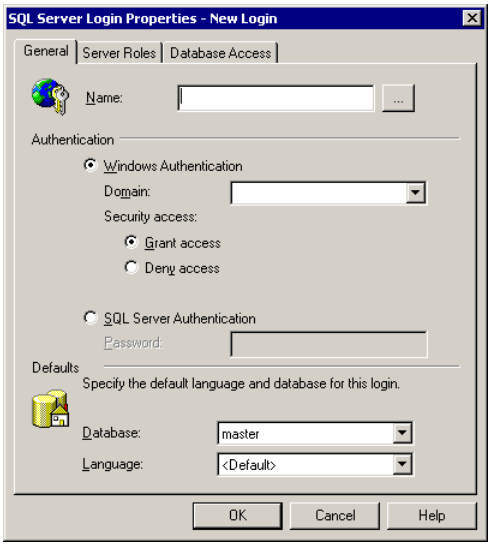
Adición de una entrada de máquina remota al servidor de la base de datos

NOTA: A modo de ejemplo, a continuación se indican los pasos para añadir secnet\case1 como entrada al servidor de la base de datos.

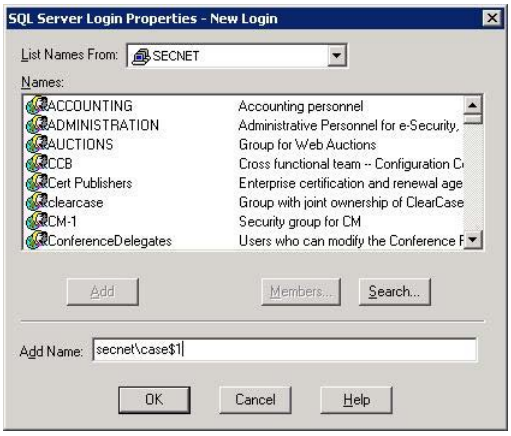
1. En la máquina de la base de datos, abra SQL Server Enterprise Manager. En el panel de navegación, bajo SQL Server Group, expanda la carpeta Seguridad y seleccione Entradas.



2. Haga clic con el botón derecho del ratón en *Entradas* > *Nueva entrada...*

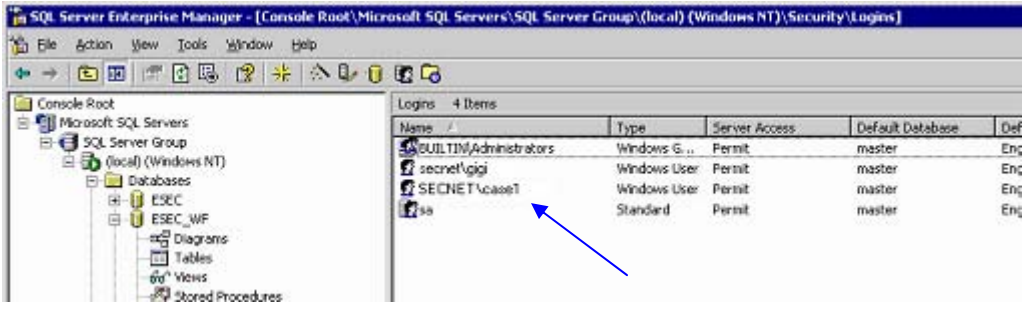


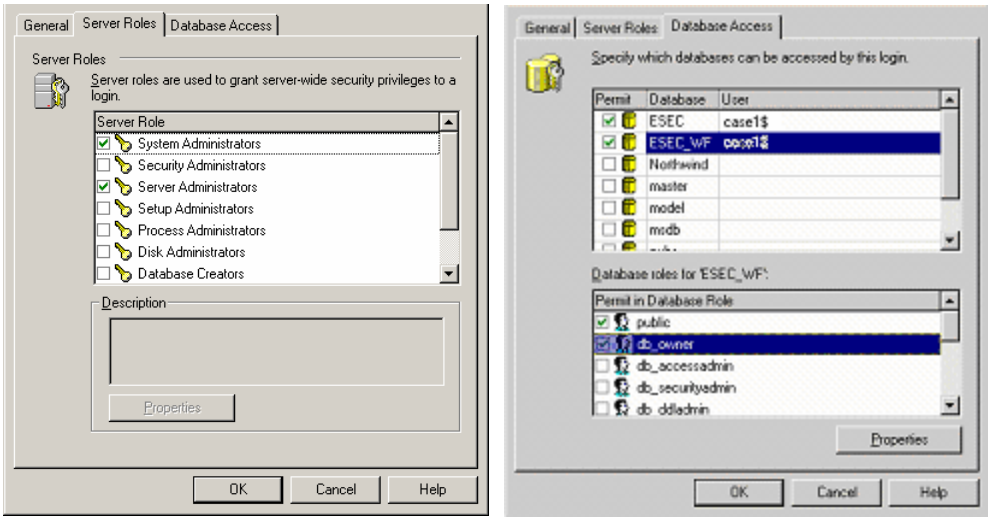
3. Haga clic en el botón Examinar que se encuentra junto al campo Nombre y aparecerá lo siguiente.



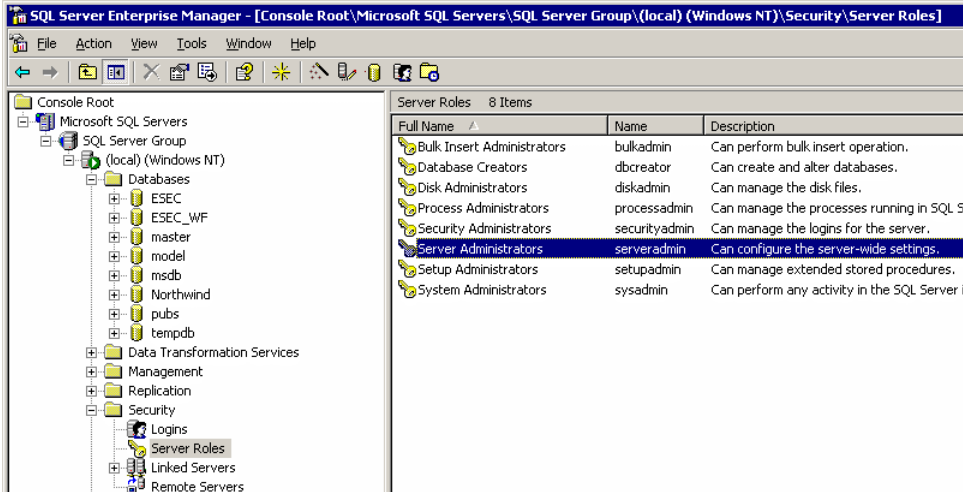
En el campo ‘Añadir nombre’, escriba un nombre de dominio y un nombre de usuario (secnet\case1$ se ofrece como ejemplo). Este es el <nombre de dominio>\<nombre de la máquina>$ de la máquina que añade como entrada al servidor de la base de datos. Haga clic en *Aceptar*.

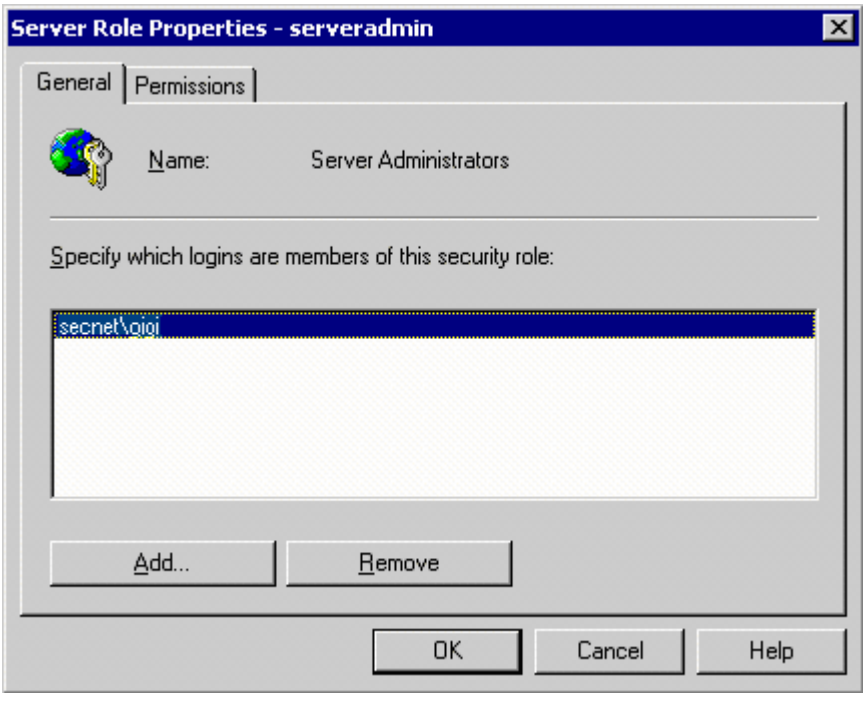
4. Haga clic con el botón derecho del ratón en Propiedades del nombre (el [<nombre de dominio>\<nombre de la máquina>$] de la máquina que añade como entrada al servidor de la base de datos) para cambiar Funciones del servidor y Acceso a bases de datos. Seleccione ‘Administradores del sistema’ y ‘Administradores del servidor’ como Funciones del servidor. Seleccione el acceso a ESEC como ‘public’ y ‘db_owner’. Seleccione el acceso a ESEC_WF como ‘public’ y ‘db_owner’.



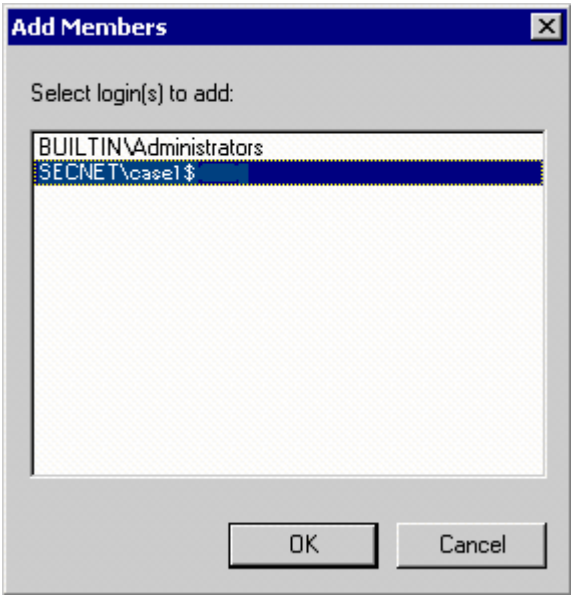


5. En ‘Funciones del servidor’, seleccione ‘Administradores del servidor’, haga clic con el botón derecho del ratón en > *Propiedades*.





6. Haga clic en el botón *Añadir*.

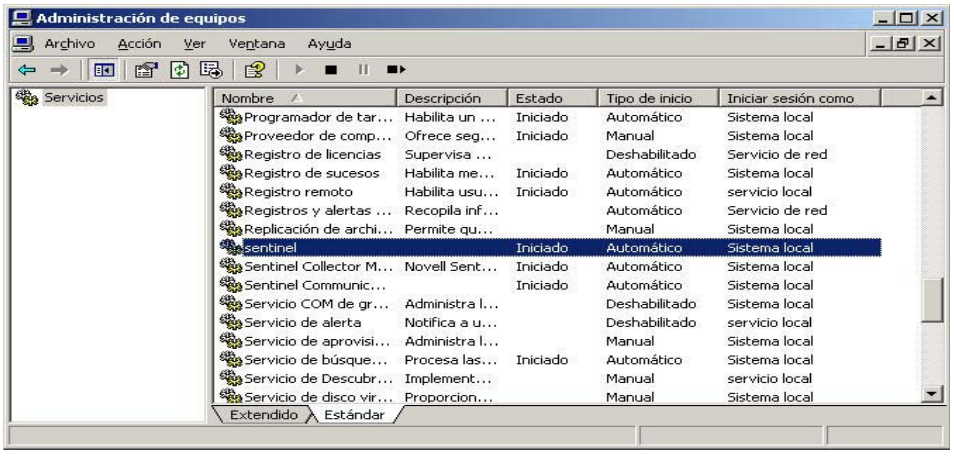


Haga clic en Aceptar. Secnet\case1$ se añade.

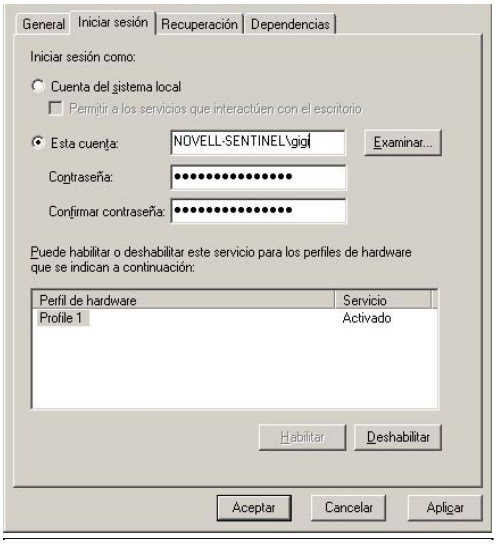
Cambio de la cuenta de inicio de sesión del servicio de Sentinel a NT AUTHORITY\NetworkService

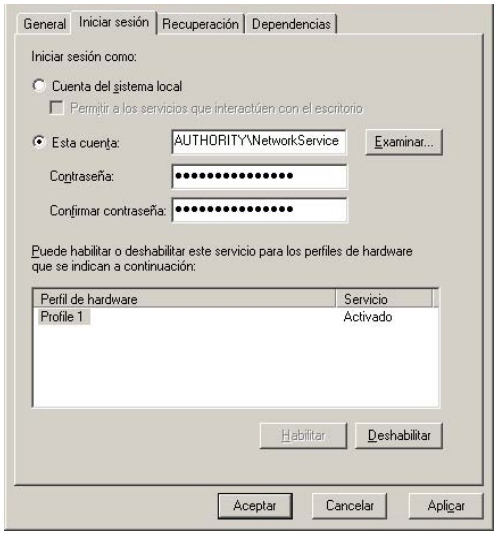
Cambio del inicio de sesión del servicio de Sentinel a NT AUTHORITY\NetworkService

1. En la máquina remota con la que se conecta a la base de datos, haga clic en *Inicio* > *Programas* >*Herramientas administrativas* > *Servicios*.

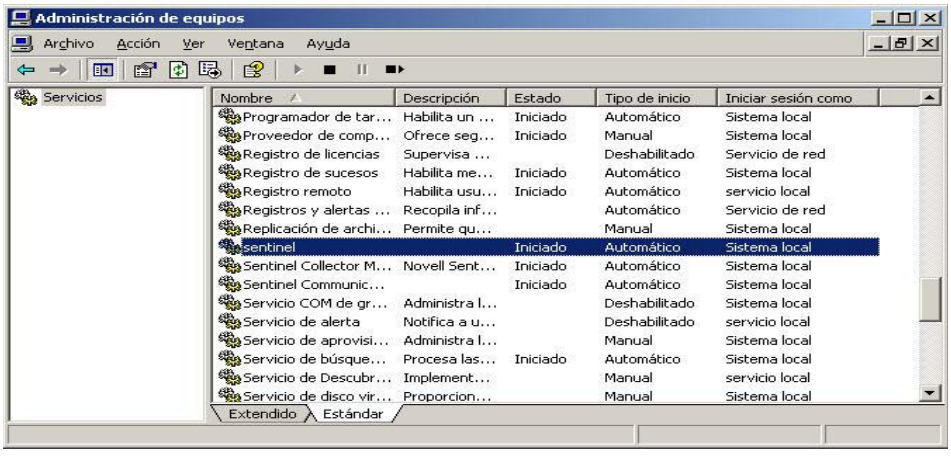


2. Detenga el servicio de Sentinel, haga clic con el botón derecho del ratón en la pestaña > *Propiedades* > *Iniciar sesión*.
3. Haga clic en 'Esta cuenta' y escriba en el campo 'NT AUTHORITY\NetworkService'. Borre los campos 'Contraseña' y 'Confirmar contraseña'.





Haga clic en *Aceptar*. La ventana Servicios del servicio de Sentinel debe indicar 'Servicio de red' bajo la columna 'Entrar como'.



Configuración del servicio de Sentinel para un inicio correcto

Para que el servicio de Sentinel se inicie correctamente, la cuenta NT AUTHORITY\NetworkService debe tener permiso de escritura para %ESEC_HOME%. Según la documentación de Microsoft, la cuenta NetworkService posee los privilegios siguientes:

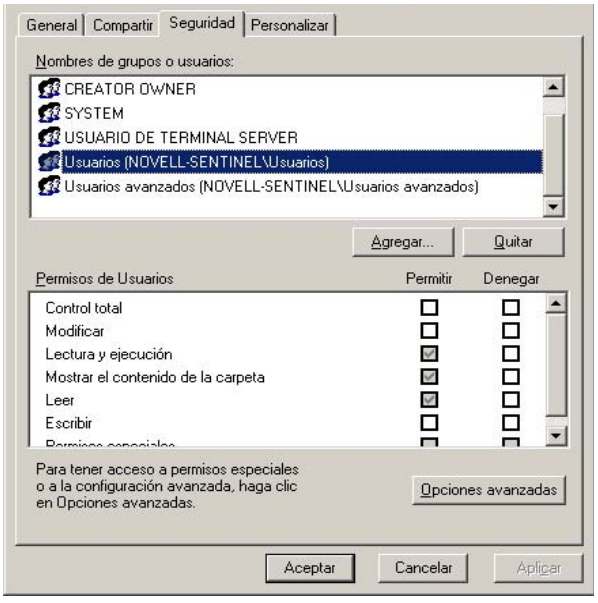
- SE_AUDIT_NAME
- SE_CHANGE_NOTIFY_NAME
- SE_UNDOCK_NAME
- Todo privilegio asignado a usuarios y usuarios autenticados.

Deberá otorgar al grupo Usuarios acceso de escritura a %ESEC_HOME%.

Configuración del servicio de Sentinel para un inicio correcto

1. Abra el explorador de Windows y desplácese a %ESEC_HOME%.
2. Haga clic con el botón derecho del ratón en la carpeta principal de Sentinel (comúnmente denominada sentinel5.1.3), pestaña > *Propiedades* > *Seguridad*.

3. Seleccione el grupo Usuarios. Active los permisos Leer y ejecutar, Listar contenido de carpetas, Lectura, Escritura.



Haga clic en *Aceptar*.

4. En la ventana Servicios, reinicie el servicio de Sentinel.

C Usuarios, funciones y permisos de acceso de la base de datos de Sentinel

NOTA: El término Agente puede intercambiarse con Recopilador. En adelante, los agentes se denominarán recopiladores.

El propósito de este documento es proporcionar un desglose detallado de los usuarios, las funciones y los permisos de acceso de la base de datos de Sentinel.

Instancia de la base de datos de Sentinel

ESEC

Usuarios:

- esecadm
- esecapp
- esecdba
- esecrpt
- otros usuarios

NOTA: Los usuarios arriba mencionados se crean a través del Gestor de usuarios. Consulte la sección Usuarios de la base de datos de Sentinel para conocer los permisos de acceso detallados.

Funciones:

- ESEC_APP: El mismo permiso que db_owner
- ESEC_ETL: esta función no se utiliza actualmente, está reservada para una actualización posterior. Consulte la sección Funciones de la base de datos de Sentinel para conocer los permisos de acceso detallados.
- ESEC_USER: consulte la sección Funciones de la base de datos de Sentinel para conocer los permisos de acceso detallados.

ESEC_WF

- Usuarios: esecapp: consulte la sección Usuarios de la base de datos de Sentinel para conocer los permisos de acceso detallados.
- Funciones: ESEC_APP: consulte la sección Funciones de la base de datos de Sentinel para conocer los permisos de acceso detallados.

Usuarios de la base de datos de Sentinel

Resumen

Nombre de usuario	Nombre de grupo	Nombre de entrada	Nombre de la BD por defecto
esecadm	ESEC_USER	esecadm	ESEC
esecapp	ESEC_APP	esecapp	ESEC
esecapp	ESEC_ETL	esecapp	ESEC
esecdba	db_owner	esecdba	ESEC
esecrpt	ESEC_USER	esecrpt	ESEC

esecadm

Nombre de entrada	Nombre de la BD	Nombre de usuario	Usuario de alias
esecadm	ESEC	ESEC_USER	MemberOf
esecadm	ESEC	esecadm	Usuario

esecapp

Nombre de entrada	Nombre de la BD	Nombre de usuario	Usuario de alias
esecapp	ESEC	ESEC_APP	MemberOf
esecapp	ESEC	ESEC_ETL	MemberOf
esecapp	ESEC	esecapp	Usuario
esecapp	ESEC_WF	ESEC_APP	MemberOf
esecapp	ESEC_WF	esecapp	Usuario

esecdba

Nombre de entrada	Nombre de la BD	Nombre de usuario	Usuario de alias
esecdba	ESEC	db_owner	MemberOf
esecdba	ESEC	esecdba	Usuario

esecrpt

Nombre de entrada	Nombre de la BD	Nombre de usuario	Usuario de alias
esecrpt	ESEC	ESEC_USER	MemberOf
esecrpt	ESEC	esecrpt	Usuario

Funciones de la base de datos de Sentinel

Resumen

- ESEC_APP: Es una función de la base de datos para ESEC y ESEC_WF. Posee el mismo permiso que db_owner para la instancia ESEC. Consulte la sección ESEC_APP para conocer los permisos detallados.
- ESEC_ETL: Es una función de la base de datos para la instancia ESEC. No se utiliza actualmente y está reservada para un desarrollo posterior. Consulte la sección Funciones de la base de datos de Sentinel para conocer los permisos de acceso detallados.
- ESEC_USER: Una función para la instancia ESEC. Consulte la sección Funciones de la base de datos de Sentinel para conocer los permisos de acceso detallados.

ESEC_APP

Para la instancia ESEC, ESEC_APP posee el mismo permiso que db_owner. ESEC_APP realiza las actividades de todas las funciones de la base de datos, así como otras actividades de mantenimiento y configuración en la base de datos. Los permisos de esta función abarcan todas las demás funciones fijas de la base de datos.

Para la instancia ESEC_WF, es el permiso para la función ESEC_APP.

Nombre de la función	Nombre del objeto	Acción	Tipo
ESEC_APP	Actividades	193 SELECT	Tabla de usuario U
ESEC_APP	Actividades	195 INSERT	Tabla de usuario U
ESEC_APP	Actividades	196 DELETE	Tabla de usuario U
ESEC_APP	Actividades	197 UPDATE	Tabla de usuario U
ESEC_APP	ActivityData	193 SELECT	Tabla de usuario U
ESEC_APP	ActivityData	195 INSERT	Tabla de usuario U
ESEC_APP	ActivityData	196 DELETE	Tabla de usuario U
ESEC_APP	ActivityData	197 UPDATE	Tabla de usuario U
ESEC_APP	ActivityStateEventAudits	193 SELECT	Tabla de usuario U
ESEC_APP	ActivityStateEventAudits	195 INSERT	Tabla de usuario U
ESEC_APP	ActivityStateEventAudits	196 DELETE	Tabla de usuario U
ESEC_APP	ActivityStateEventAudits	197 UPDATE	Tabla de usuario U
ESEC_APP	ActivityStates	193 SELECT	Tabla de usuario U
ESEC_APP	ActivityStates	195 INSERT	Tabla de usuario U
ESEC_APP	ActivityStates	196 DELETE	Tabla de usuario U
ESEC_APP	ActivityStates	197 UPDATE	Tabla de usuario U
ESEC_APP	AndJoinTable	193 SELECT	Tabla de usuario U
ESEC_APP	AndJoinTable	195 INSERT	Tabla de usuario U
ESEC_APP	AndJoinTable	196 DELETE	Tabla de usuario U
ESEC_APP	AndJoinTable	197 UPDATE	Tabla de usuario U
ESEC_APP	AssignmentEventAudits	193 SELECT	Tabla de usuario U
ESEC_APP	AssignmentEventAudits	195 INSERT	Tabla de usuario U
ESEC_APP	AssignmentEventAudits	196 DELETE	Tabla de usuario U
ESEC_APP	AssignmentEventAudits	197 UPDATE	Tabla de usuario U
ESEC_APP	AssignmentsTable	193 SELECT	Tabla de usuario U
ESEC_APP	AssignmentsTable	195 INSERT	Tabla de usuario U
ESEC_APP	AssignmentsTable	196 DELETE	Tabla de usuario U
ESEC_APP	AssignmentsTable	197 UPDATE	Tabla de usuario U
ESEC_APP	Counters	193 SELECT	Tabla de usuario U
ESEC_APP	Counters	195 INSERT	Tabla de usuario U
ESEC_APP	Counters	196 DELETE	Tabla de usuario U
ESEC_APP	Counters	197 UPDATE	Tabla de usuario U
ESEC_APP	CreateProcessEventAudits	193 SELECT	Tabla de usuario U
ESEC_APP	CreateProcessEventAudits	195 INSERT	Tabla de usuario U
ESEC_APP	CreateProcessEventAudits	196 DELETE	Tabla de usuario U
ESEC_APP	CreateProcessEventAudits	197 UPDATE	Tabla de usuario U
ESEC_APP	DataEventAudits	193 SELECT	Tabla de usuario U
ESEC_APP	DataEventAudits	195 INSERT	Tabla de usuario U

Nombre de la función	Nombre del objeto	Acción	Tipo
ESEC_APP	DataEventAudits	196 DELETE	Tabla de usuario U
ESEC_APP	DataEventAudits	197 UPDATE	Tabla de usuario U
ESEC_APP	Deadlines	193 SELECT	Tabla de usuario U
ESEC_APP	Deadlines	195 INSERT	Tabla de usuario U
ESEC_APP	Deadlines	196 DELETE	Tabla de usuario U
ESEC_APP	Deadlines	197 UPDATE	Tabla de usuario U
ESEC_APP	EventTypes	193 SELECT	Tabla de usuario U
ESEC_APP	EventTypes	195 INSERT	Tabla de usuario U
ESEC_APP	EventTypes	196 DELETE	Tabla de usuario U
ESEC_APP	EventTypes	197 UPDATE	Tabla de usuario U
ESEC_APP	GroupGroupTable	193 SELECT	Tabla de usuario U
ESEC_APP	GroupGroupTable	195 INSERT	Tabla de usuario U
ESEC_APP	GroupGroupTable	196 DELETE	Tabla de usuario U
ESEC_APP	GroupGroupTable	197 UPDATE	Tabla de usuario U
ESEC_APP	GroupTable	193 SELECT	Tabla de usuario U
ESEC_APP	GroupTable	195 INSERT	Tabla de usuario U
ESEC_APP	GroupTable	196 DELETE	Tabla de usuario U
ESEC_APP	GroupTable	197 UPDATE	Tabla de usuario U
ESEC_APP	GroupUser	193 SELECT	Tabla de usuario U
ESEC_APP	GroupUser	195 INSERT	Tabla de usuario U
ESEC_APP	GroupUser	196 DELETE	Tabla de usuario U
ESEC_APP	GroupUser	197 UPDATE	Tabla de usuario U
ESEC_APP	GroupUserPackLevelParticipant	193 SELECT	Tabla de usuario U
ESEC_APP	GroupUserPackLevelParticipant	195 INSERT	Tabla de usuario U
ESEC_APP	GroupUserPackLevelParticipant	196 DELETE	Tabla de usuario U
ESEC_APP	GroupUserPackLevelParticipant	197 UPDATE	Tabla de usuario U
ESEC_APP	GroupUserProcLevelParticipant	193 SELECT	Tabla de usuario U
ESEC_APP	GroupUserProcLevelParticipant	195 INSERT	Tabla de usuario U
ESEC_APP	GroupUserProcLevelParticipant	196 DELETE	Tabla de usuario U
ESEC_APP	GroupUserProcLevelParticipant	197 UPDATE	Tabla de usuario U
ESEC_APP	LockTable	193 SELECT	Tabla de usuario U
ESEC_APP	LockTable	195 INSERT	Tabla de usuario U
ESEC_APP	LockTable	196 DELETE	Tabla de usuario U
ESEC_APP	LockTable	197 UPDATE	Tabla de usuario U
ESEC_APP	NewEventAuditData	193 SELECT	Tabla de usuario U
ESEC_APP	NewEventAuditData	195 INSERT	Tabla de usuario U
ESEC_APP	NewEventAuditData	196 DELETE	Tabla de usuario U
ESEC_APP	NewEventAuditData	197 UPDATE	Tabla de usuario U
ESEC_APP	NextXPDLVersions	193 SELECT	Tabla de usuario U
ESEC_APP	NextXPDLVersions	195 INSERT	Tabla de usuario U
ESEC_APP	NextXPDLVersions	196 DELETE	Tabla de usuario U
ESEC_APP	NextXPDLVersions	197 UPDATE	Tabla de usuario U
ESEC_APP	NormalUser	193 SELECT	Tabla de usuario U
ESEC_APP	NormalUser	195 INSERT	Tabla de usuario U
ESEC_APP	NormalUser	196 DELETE	Tabla de usuario U
ESEC_APP	NormalUser	197 UPDATE	Tabla de usuario U

Nombre de la función	Nombre del objeto	Acción	Tipo
ESEC_APP	ObjectId	193 SELECT	Tabla de usuario U
ESEC_APP	ObjectId	195 INSERT	Tabla de usuario U
ESEC_APP	ObjectId	196 DELETE	Tabla de usuario U
ESEC_APP	ObjectId	197 UPDATE	Tabla de usuario U
ESEC_APP	OldEventAuditData	193 SELECT	Tabla de usuario U
ESEC_APP	OldEventAuditData	195 INSERT	Tabla de usuario U
ESEC_APP	OldEventAuditData	196 DELETE	Tabla de usuario U
ESEC_APP	OldEventAuditData	197 UPDATE	Tabla de usuario U
ESEC_APP	PackLevelParticipant	193 SELECT	Tabla de usuario U
ESEC_APP	PackLevelParticipant	195 INSERT	Tabla de usuario U
ESEC_APP	PackLevelParticipant	196 DELETE	Tabla de usuario U
ESEC_APP	PackLevelParticipant	197 UPDATE	Tabla de usuario U
ESEC_APP	PackLevelXPDLApp	193 SELECT	Tabla de usuario U
ESEC_APP	PackLevelXPDLApp	195 INSERT	Tabla de usuario U
ESEC_APP	PackLevelXPDLApp	196 DELETE	Tabla de usuario U
ESEC_APP	PackLevelXPDLApp	197 UPDATE	Tabla de usuario U
ESEC_APP	PackLevelXPDLAppTAAppDetail	193 SELECT	Tabla de usuario U
ESEC_APP	PackLevelXPDLAppTAAppDetail	195 INSERT	Tabla de usuario U
ESEC_APP	PackLevelXPDLAppTAAppDetail	196 DELETE	Tabla de usuario U
ESEC_APP	PackLevelXPDLAppTAAppDetail	197 UPDATE	Tabla de usuario U
ESEC_APP	PackLevelXPDLAppTAAppDetailUsr	193 SELECT	Tabla de usuario U
ESEC_APP	PackLevelXPDLAppTAAppDetailUsr	195 INSERT	Tabla de usuario U
ESEC_APP	PackLevelXPDLAppTAAppDetailUsr	196 DELETE	Tabla de usuario U
ESEC_APP	PackLevelXPDLAppTAAppDetailUsr	197 UPDATE	Tabla de usuario U
ESEC_APP	PackLevelXPDLAppTAAppUser	193 SELECT	Tabla de usuario U
ESEC_APP	PackLevelXPDLAppTAAppUser	195 INSERT	Tabla de usuario U
ESEC_APP	PackLevelXPDLAppTAAppUser	196 DELETE	Tabla de usuario U
ESEC_APP	PackLevelXPDLAppTAAppUser	197 UPDATE	Tabla de usuario U
ESEC_APP	PackLevelXPDLAppToolAgentApp	193 SELECT	Tabla de usuario U
ESEC_APP	PackLevelXPDLAppToolAgentApp	195 INSERT	Tabla de usuario U
ESEC_APP	PackLevelXPDLAppToolAgentApp	196 DELETE	Tabla de usuario U
ESEC_APP	PackLevelXPDLAppToolAgentApp	197 UPDATE	Tabla de usuario U
ESEC_APP	ProcessData	193 SELECT	Tabla de usuario U
ESEC_APP	ProcessData	195 INSERT	Tabla de usuario U
ESEC_APP	ProcessData	196 DELETE	Tabla de usuario U
ESEC_APP	ProcessData	197 UPDATE	Tabla de usuario U
ESEC_APP	ProcessDefinitions	193 SELECT	Tabla de usuario U
ESEC_APP	ProcessDefinitions	195 INSERT	Tabla de usuario U
ESEC_APP	ProcessDefinitions	196 DELETE	Tabla de usuario U
ESEC_APP	ProcessDefinitions	197 UPDATE	Tabla de usuario U
ESEC_APP	Processes	193 SELECT	Tabla de usuario U
ESEC_APP	Processes	195 INSERT	Tabla de usuario U
ESEC_APP	Processes	196 DELETE	Tabla de usuario U
ESEC_APP	Processes	197 UPDATE	Tabla de usuario U
ESEC_APP	ProcessRequesters	193 SELECT	Tabla de usuario U
ESEC_APP	ProcessRequesters	195 INSERT	Tabla de usuario U

Nombre de la función	Nombre del objeto	Acción	Tipo
ESEC_APP	ProcessRequesters	196 DELETE	Tabla de usuario U
ESEC_APP	ProcessRequesters	197 UPDATE	Tabla de usuario U
ESEC_APP	ProcessStateEventAudits	193 SELECT	Tabla de usuario U
ESEC_APP	ProcessStateEventAudits	195 INSERT	Tabla de usuario U
ESEC_APP	ProcessStateEventAudits	196 DELETE	Tabla de usuario U
ESEC_APP	ProcessStateEventAudits	197 UPDATE	Tabla de usuario U
ESEC_APP	ProcessStates	193 SELECT	Tabla de usuario U
ESEC_APP	ProcessStates	195 INSERT	Tabla de usuario U
ESEC_APP	ProcessStates	196 DELETE	Tabla de usuario U
ESEC_APP	ProcessStates	197 UPDATE	Tabla de usuario U
ESEC_APP	ProcLevelParticipant	193 SELECT	Tabla de usuario U
ESEC_APP	ProcLevelParticipant	195 INSERT	Tabla de usuario U
ESEC_APP	ProcLevelParticipant	196 DELETE	Tabla de usuario U
ESEC_APP	ProcLevelParticipant	197 UPDATE	Tabla de usuario U
ESEC_APP	ProcLevelXPDLApp	193 SELECT	Tabla de usuario U
ESEC_APP	ProcLevelXPDLApp	195 INSERT	Tabla de usuario U
ESEC_APP	ProcLevelXPDLApp	196 DELETE	Tabla de usuario U
ESEC_APP	ProcLevelXPDLApp	197 UPDATE	Tabla de usuario U
ESEC_APP	ProcLevelXPDLAppTAAppDetail	193 SELECT	Tabla de usuario U
ESEC_APP	ProcLevelXPDLAppTAAppDetail	195 INSERT	Tabla de usuario U
ESEC_APP	ProcLevelXPDLAppTAAppDetail	196 DELETE	Tabla de usuario U
ESEC_APP	ProcLevelXPDLAppTAAppDetail	197 UPDATE	Tabla de usuario U
ESEC_APP	ProcLevelXPDLAppTAAppDetailUsr	193 SELECT	Tabla de usuario U
ESEC_APP	ProcLevelXPDLAppTAAppDetailUsr	195 INSERT	Tabla de usuario U
ESEC_APP	ProcLevelXPDLAppTAAppDetailUsr	196 DELETE	Tabla de usuario U
ESEC_APP	ProcLevelXPDLAppTAAppDetailUsr	197 UPDATE	Tabla de usuario U
ESEC_APP	ProcLevelXPDLAppTAAppUser	193 SELECT	Tabla de usuario U
ESEC_APP	ProcLevelXPDLAppTAAppUser	195 INSERT	Tabla de usuario U
ESEC_APP	ProcLevelXPDLAppTAAppUser	196 DELETE	Tabla de usuario U
ESEC_APP	ProcLevelXPDLAppTAAppUser	197 UPDATE	Tabla de usuario U
ESEC_APP	ProcLevelXPDLAppToolAgentApp	193 SELECT	Tabla de usuario U
ESEC_APP	ProcLevelXPDLAppToolAgentApp	195 INSERT	Tabla de usuario U
ESEC_APP	ProcLevelXPDLAppToolAgentApp	196 DELETE	Tabla de usuario U
ESEC_APP	ProcLevelXPDLAppToolAgentApp	197 UPDATE	Tabla de usuario U
ESEC_APP	ResourcesTable	193 SELECT	Tabla de usuario U
ESEC_APP	ResourcesTable	195 INSERT	Tabla de usuario U
ESEC_APP	ResourcesTable	196 DELETE	Tabla de usuario U
ESEC_APP	ResourcesTable	197 UPDATE	Tabla de usuario U
ESEC_APP	StateEventAudits	193 SELECT	Tabla de usuario U
ESEC_APP	StateEventAudits	195 INSERT	Tabla de usuario U
ESEC_APP	StateEventAudits	196 DELETE	Tabla de usuario U
ESEC_APP	StateEventAudits	197 UPDATE	Tabla de usuario U
ESEC_APP	ToolAgentApp	193 SELECT	Tabla de usuario U
ESEC_APP	ToolAgentApp	195 INSERT	Tabla de usuario U
ESEC_APP	ToolAgentApp	196 DELETE	Tabla de usuario U
ESEC_APP	ToolAgentApp	197 UPDATE	Tabla de usuario U

Nombre de la función	Nombre del objeto	Acción	Tipo
ESEC_APP	ToolAgentAppDetail	193 SELECT	Tabla de usuario U
ESEC_APP	ToolAgentAppDetail	195 INSERT	Tabla de usuario U
ESEC_APP	ToolAgentAppDetail	196 DELETE	Tabla de usuario U
ESEC_APP	ToolAgentAppDetail	197 UPDATE	Tabla de usuario U
ESEC_APP	ToolAgentAppDetailUser	193 SELECT	Tabla de usuario U
ESEC_APP	ToolAgentAppDetailUser	195 INSERT	Tabla de usuario U
ESEC_APP	ToolAgentAppDetailUser	196 DELETE	Tabla de usuario U
ESEC_APP	ToolAgentAppDetailUser	197 UPDATE	Tabla de usuario U
ESEC_APP	ToolAgentAppUser	193 SELECT	Tabla de usuario U
ESEC_APP	ToolAgentAppUser	195 INSERT	Tabla de usuario U
ESEC_APP	ToolAgentAppUser	196 DELETE	Tabla de usuario U
ESEC_APP	ToolAgentAppUser	197 UPDATE	Tabla de usuario U
ESEC_APP	ToolAgentUser	193 SELECT	Tabla de usuario U
ESEC_APP	ToolAgentUser	195 INSERT	Tabla de usuario U
ESEC_APP	ToolAgentUser	196 DELETE	Tabla de usuario U
ESEC_APP	ToolAgentUser	197 UPDATE	Tabla de usuario U
ESEC_APP	UserGroupTable	193 SELECT	Tabla de usuario U
ESEC_APP	UserGroupTable	195 INSERT	Tabla de usuario U
ESEC_APP	UserGroupTable	196 DELETE	Tabla de usuario U
ESEC_APP	UserGroupTable	197 UPDATE	Tabla de usuario U
ESEC_APP	UserPackLevelParticipant	193 SELECT	Tabla de usuario U
ESEC_APP	UserPackLevelParticipant	195 INSERT	Tabla de usuario U
ESEC_APP	UserPackLevelParticipant	196 DELETE	Tabla de usuario U
ESEC_APP	UserPackLevelParticipant	197 UPDATE	Tabla de usuario U
ESEC_APP	UserProcLevelParticipant	193 SELECT	Tabla de usuario U
ESEC_APP	UserProcLevelParticipant	195 INSERT	Tabla de usuario U
ESEC_APP	UserProcLevelParticipant	196 DELETE	Tabla de usuario U
ESEC_APP	UserProcLevelParticipant	197 UPDATE	Tabla de usuario U
ESEC_APP	UserTable	193 SELECT	Tabla de usuario U
ESEC_APP	UserTable	195 INSERT	Tabla de usuario U
ESEC_APP	UserTable	196 DELETE	Tabla de usuario U
ESEC_APP	UserTable	197 UPDATE	Tabla de usuario U
ESEC_APP	XPDLApplicationPackage	193 SELECT	Tabla de usuario U
ESEC_APP	XPDLApplicationPackage	195 INSERT	Tabla de usuario U
ESEC_APP	XPDLApplicationPackage	196 DELETE	Tabla de usuario U
ESEC_APP	XPDLApplicationPackage	197 UPDATE	Tabla de usuario U
ESEC_APP	XPDLApplicationProcess	193 SELECT	Tabla de usuario U
ESEC_APP	XPDLApplicationProcess	195 INSERT	Tabla de usuario U
ESEC_APP	XPDLApplicationProcess	196 DELETE	Tabla de usuario U
ESEC_APP	XPDLApplicationProcess	197 UPDATE	Tabla de usuario U
ESEC_APP	XPDLData	193 SELECT	Tabla de usuario U
ESEC_APP	XPDLData	195 INSERT	Tabla de usuario U
ESEC_APP	XPDLData	196 DELETE	Tabla de usuario U
ESEC_APP	XPDLData	197 UPDATE	Tabla de usuario U
ESEC_APP	XPDLHistory	193 SELECT	Tabla de usuario U
ESEC_APP	XPDLHistory	195 INSERT	Tabla de usuario U

Nombre de la función	Nombre del objeto	Acción	Tipo
ESEC_APP	XPDLHistory	196 DELETE	Tabla de usuario U
ESEC_APP	XPDLHistory	197 UPDATE	Tabla de usuario U
ESEC_APP	XPDLHistoryData	193 SELECT	Tabla de usuario U
ESEC_APP	XPDLHistoryData	195 INSERT	Tabla de usuario U
ESEC_APP	XPDLHistoryData	196 DELETE	Tabla de usuario U
ESEC_APP	XPDLHistoryData	197 UPDATE	Tabla de usuario U
ESEC_APP	XPDLParticipantPackage	193 SELECT	Tabla de usuario U
ESEC_APP	XPDLParticipantPackage	195 INSERT	Tabla de usuario U
ESEC_APP	XPDLParticipantPackage	196 DELETE	Tabla de usuario U
ESEC_APP	XPDLParticipantPackage	197 UPDATE	Tabla de usuario U
ESEC_APP	XPDLParticipantProcess	193 SELECT	Tabla de usuario U
ESEC_APP	XPDLParticipantProcess	195 INSERT	Tabla de usuario U
ESEC_APP	XPDLParticipantProcess	196 DELETE	Tabla de usuario U
ESEC_APP	XPDLParticipantProcess	197 UPDATE	Tabla de usuario U
ESEC_APP	XPDLReferences	193 SELECT	Tabla de usuario U
ESEC_APP	XPDLReferences	195 INSERT	Tabla de usuario U
ESEC_APP	XPDLReferences	196 DELETE	Tabla de usuario U
ESEC_APP	XPDLReferences	197 UPDATE	Tabla de usuario U
ESEC_APP	XPDLS	193 SELECT	Tabla de usuario U
ESEC_APP	XPDLS	195 INSERT	Tabla de usuario U
ESEC_APP	XPDLS	196 DELETE	Tabla de usuario U
ESEC_APP	XPDLS	197 UPDATE	Tabla de usuario U

ESEC_ETL

Nombre de la función	Nombre del objeto	Acción	Tipo
ESEC_ETL	ACTVY	193 SELECT	Tabla de usuario U
ESEC_ETL	ACTVY_NAMESPACE	193 SELECT	Tabla de usuario U
ESEC_ETL	ACTVY_PARM	193 SELECT	Tabla de usuario U
ESEC_ETL	ACTVY_REF	193 SELECT	Tabla de usuario U
ESEC_ETL	ACTVY_REF_PARM_VAL	193 SELECT	Tabla de usuario U
ESEC_ETL	ADV_ALERT	193 SELECT	Tabla de usuario U
ESEC_ETL	ADV_ALERT_CVE	193 SELECT	Tabla de usuario U
ESEC_ETL	ADV_ALERT_PRODUCT	193 SELECT	Tabla de usuario U
ESEC_ETL	ADV_ATTACK	193 SELECT	Tabla de usuario U
ESEC_ETL	ADV_ATTACK_ALERT	193 SELECT	Tabla de usuario U
ESEC_ETL	ADV_ATTACK_CVE	193 SELECT	Tabla de usuario U
ESEC_ETL	ADV_ATTACK_MAP	193 SELECT	Tabla de usuario U
ESEC_ETL	ADV_ATTACK_PLUGIN	193 SELECT	Tabla de usuario U
ESEC_ETL	ADV_CREDIBILITY	193 SELECT	Tabla de usuario U
ESEC_ETL	ADV_FEED	193 SELECT	Tabla de usuario U
ESEC_ETL	ADV_PRODUCT	193 SELECT	Tabla de usuario U
ESEC_ETL	ADV_PRODUCT_SERVICE_PACK	193 SELECT	Tabla de usuario U
ESEC_ETL	ADV_PRODUCT_VERSION	193 SELECT	Tabla de usuario U
ESEC_ETL	ADV_SEVERITY	193 SELECT	Tabla de usuario U
ESEC_ETL	ADV_SUBALERT	193 SELECT	Tabla de usuario U

Nombre de la función	Nombre del objeto	Acción	Tipo
ESEC_ETL	ADV_URGENCY	193 SELECT	Tabla de usuario U
ESEC_ETL	ADV_VENDOR	193 SELECT	Tabla de usuario U
ESEC_ETL	ADV_VULN_PRODUCT	193 SELECT	Tabla de usuario U
ESEC_ETL	ANNOTATIONS	193 SELECT	Tabla de usuario U
ESEC_ETL	ASSET	193 SELECT	Tabla de usuario U
ESEC_ETL	ASSET_CTGRY	193 SELECT	Tabla de usuario U
ESEC_ETL	ASSET_HOSTNAME	193 SELECT	Tabla de usuario U
ESEC_ETL	ASSET_IP	193 SELECT	Tabla de usuario U
ESEC_ETL	ASSET_LOC	193 SELECT	Tabla de usuario U
ESEC_ETL	ASSET_VAL_LKUP	193 SELECT	Tabla de usuario U
ESEC_ETL	ASSET_X_ENTITY_X_ROLE	193 SELECT	Tabla de usuario U
ESEC_ETL	ASSOCIATIONS	193 SELECT	Tabla de usuario U
ESEC_ETL	ATTACHMENTS	193 SELECT	Tabla de usuario U
ESEC_ETL	CONFIGS	193 SELECT	Tabla de usuario U
ESEC_ETL	CONTACTS	193 SELECT	Tabla de usuario U
ESEC_ETL	CORRELATED_EVENTS_P_MAX	193 SELECT	Tabla de usuario U
ESEC_ETL	CORRELATED_EVENTS_P_MIN	193 SELECT	Tabla de usuario U
ESEC_ETL	CRIT_LKUP	193 SELECT	Tabla de usuario U
ESEC_ETL	CUST	193 SELECT	Tabla de usuario U
ESEC_ETL	ENTITY_TYP_LKUP	193 SELECT	Tabla de usuario U
ESEC_ETL	ENV_IDENTITY_LKUP	193 SELECT	Tabla de usuario U
ESEC_ETL	ESEC_ARCHIVE_CONFIG	193 SELECT	Tabla de usuario U
ESEC_ETL	ESEC_ARCHIVE_LOG_FILES	193 SELECT	Tabla de usuario U
ESEC_ETL	ESEC_ARCHIVE_LOGS	193 SELECT	Tabla de usuario U
ESEC_ETL	ESEC_DB_PATCHES	193 SELECT	Tabla de usuario U
ESEC_ETL	ESEC_DB_VERSION	193 SELECT	Tabla de usuario U
ESEC_ETL	ESEC_DISPLAY	193 SELECT	Tabla de usuario U
ESEC_ETL	ESEC_PARTITION_CONFIG	193 SELECT	Tabla de usuario U
ESEC_ETL	ESEC_PARTITIONS_TEMP	193 SELECT	Tabla de usuario U
ESEC_ETL	ESEC_PORT_REFERENCE	193 SELECT	Tabla de usuario U
ESEC_ETL	ESEC_PROTOCOL_REFERENCE	193 SELECT	Tabla de usuario U
ESEC_ETL	ESEC_SDM_LOCK	193 SELECT	Tabla de usuario U
ESEC_ETL	ESEC_SEQUENCE	193 SELECT	Tabla de usuario U
ESEC_ETL	EVENTS_P_MAX	193 SELECT	Tabla de usuario U
ESEC_ETL	EVENTS_P_MIN	193 SELECT	Tabla de usuario U
ESEC_ETL	EVT_AGENT	193 SELECT	Tabla de usuario U
ESEC_ETL	EVT_ASSET	193 SELECT	Tabla de usuario U
ESEC_ETL	EVT_DEST_EVT_NAME_SMRY_1_P_MAX	193 SELECT	Tabla de usuario U
ESEC_ETL	EVT_DEST_EVT_NAME_SMRY_1_P_MAX	195 INSERT	Tabla de usuario U
ESEC_ETL	EVT_DEST_EVT_NAME_SMRY_1_P_MAX	196 DELETE	Tabla de usuario U
ESEC_ETL	EVT_DEST_EVT_NAME_SMRY_1_P_MAX	197 UPDATE	Tabla de usuario U
ESEC_ETL	EVT_DEST_EVT_NAME_SMRY_1_P_MIN	193 SELECT	Tabla de usuario U
ESEC_ETL	EVT_DEST_SMRY_1_P_MAX	193 SELECT	Tabla de usuario U
ESEC_ETL	EVT_DEST_SMRY_1_P_MAX	195 INSERT	Tabla de usuario U
ESEC_ETL	EVT_DEST_SMRY_1_P_MAX	196 DELETE	Tabla de usuario U
ESEC_ETL	EVT_DEST_SMRY_1_P_MAX	197 UPDATE	Tabla de usuario U

Nombre de la función	Nombre del objeto	Acción	Tipo
ESEC_ETL	EVT_DEST_SMRY_1_P_MIN	193 SELECT	Tabla de usuario U
ESEC_ETL	EVT_DEST_TXNMY_SMRY_1_P_MAX	193 SELECT	Tabla de usuario U
ESEC_ETL	EVT_DEST_TXNMY_SMRY_1_P_MAX	195 INSERT	Tabla de usuario U
ESEC_ETL	EVT_DEST_TXNMY_SMRY_1_P_MAX	196 DELETE	Tabla de usuario U
ESEC_ETL	EVT_DEST_TXNMY_SMRY_1_P_MAX	197 UPDATE	Tabla de usuario U
ESEC_ETL	EVT_DEST_TXNMY_SMRY_1_P_MIN	193 SELECT	Tabla de usuario U
ESEC_ETL	EVT_NAME	193 SELECT	Tabla de usuario U
ESEC_ETL	EVT_NAME	195 INSERT	Tabla de usuario U
ESEC_ETL	EVT_NAME	196 DELETE	Tabla de usuario U
ESEC_ETL	EVT_NAME	197 UPDATE	Tabla de usuario U
ESEC_ETL	EVT_PORT_SMRY_1_P_MAX	193 SELECT	Tabla de usuario U
ESEC_ETL	EVT_PORT_SMRY_1_P_MAX	195 INSERT	Tabla de usuario U
ESEC_ETL	EVT_PORT_SMRY_1_P_MAX	196 DELETE	Tabla de usuario U
ESEC_ETL	EVT_PORT_SMRY_1_P_MAX	197 UPDATE	Tabla de usuario U
ESEC_ETL	EVT_PORT_SMRY_1_P_MIN	193 SELECT	Tabla de usuario U
ESEC_ETL	EVT_PRTCL	193 SELECT	Tabla de usuario U
ESEC_ETL	EVT_RSRC	193 SELECT	Tabla de usuario U
ESEC_ETL	EVT_SEV_SMRY_1_P_MAX	193 SELECT	Tabla de usuario U
ESEC_ETL	EVT_SEV_SMRY_1_P_MAX	195 INSERT	Tabla de usuario U
ESEC_ETL	EVT_SEV_SMRY_1_P_MAX	196 DELETE	Tabla de usuario U
ESEC_ETL	EVT_SEV_SMRY_1_P_MAX	197 UPDATE	Tabla de usuario U
ESEC_ETL	EVT_SEV_SMRY_1_P_MIN	193 SELECT	Tabla de usuario U
ESEC_ETL	EVT_SRC_SMRY_1_P_MAX	193 SELECT	Tabla de usuario U
ESEC_ETL	EVT_SRC_SMRY_1_P_MAX	195 INSERT	Tabla de usuario U
ESEC_ETL	EVT_SRC_SMRY_1_P_MAX	196 DELETE	Tabla de usuario U
ESEC_ETL	EVT_SRC_SMRY_1_P_MAX	197 UPDATE	Tabla de usuario U
ESEC_ETL	EVT_SRC_SMRY_1_P_MIN	193 SELECT	Tabla de usuario U
ESEC_ETL	EVT_TXNMY	193 SELECT	Tabla de usuario U
ESEC_ETL	EVT_USR	193 SELECT	Tabla de usuario U
ESEC_ETL	EVT_USR	195 INSERT	Tabla de usuario U
ESEC_ETL	EVT_USR	196 DELETE	Tabla de usuario U
ESEC_ETL	EVT_USR	197 UPDATE	Tabla de usuario U
ESEC_ETL	EXT_DATA	193 SELECT	Tabla de usuario U
ESEC_ETL	HIST_CORRELATED_EVENTS_P_MAX	193 SELECT	Tabla de usuario U
ESEC_ETL	HIST_EVENTS_P_MAX	193 SELECT	Tabla de usuario U
ESEC_ETL	IMAGES	193 SELECT	Tabla de usuario U
ESEC_ETL	INCIDENTS	193 SELECT	Tabla de usuario U
ESEC_ETL	INCIDENTS_ASSETS	193 SELECT	Tabla de usuario U
ESEC_ETL	INCIDENTS_EVENTS	193 SELECT	Tabla de usuario U
ESEC_ETL	INCIDENTS_VULN	193 SELECT	Tabla de usuario U
ESEC_ETL	L_STAT	193 SELECT	Tabla de usuario U
ESEC_ETL	LOGS	193 SELECT	Tabla de usuario U
ESEC_ETL	MD_CONFIG	193 SELECT	Tabla de usuario U
ESEC_ETL	MD_EVT_FILE_STS	193 SELECT	Tabla de usuario U
ESEC_ETL	MD_EVT_FILE_STS	195 INSERT	Tabla de usuario U
ESEC_ETL	MD_EVT_FILE_STS	196 DELETE	Tabla de usuario U

Nombre de la función	Nombre del objeto	Acción	Tipo
ESEC_ETL	MD_EVT_FILE_STS	197 UPDATE	Tabla de usuario U
ESEC_ETL	MD_SMRY_STS	193 SELECT	Tabla de usuario U
ESEC_ETL	MD_SMRY_STS	195 INSERT	Tabla de usuario U
ESEC_ETL	MD_SMRY_STS	196 DELETE	Tabla de usuario U
ESEC_ETL	MD_SMRY_STS	197 UPDATE	Tabla de usuario U
ESEC_ETL	MD_VIEW_CONFIG	193 SELECT	Tabla de usuario U
ESEC_ETL	NETWORK_IDENTITY_LKUP	193 SELECT	Tabla de usuario U
ESEC_ETL	OBJ_STORE	193 SELECT	Tabla de usuario U
ESEC_ETL	ORGANIZATION	193 SELECT	Tabla de usuario U
ESEC_ETL	PERSON	193 SELECT	Tabla de usuario U
ESEC_ETL	PHYSICAL_ASSET	193 SELECT	Tabla de usuario U
ESEC_ETL	PRDT	193 SELECT	Tabla de usuario U
ESEC_ETL	ROLE_LKUP	193 SELECT	Tabla de usuario U
ESEC_ETL	SENSITIVITY_LKUP	193 SELECT	Tabla de usuario U
ESEC_ETL	STATES	193 SELECT	Tabla de usuario U
ESEC_ETL	USERS	193 SELECT	Tabla de usuario U
ESEC_ETL	VNDR	193 SELECT	Tabla de usuario U
ESEC_ETL	VULN	193 SELECT	Tabla de usuario U
ESEC_ETL	VULN_CODE	193 SELECT	Tabla de usuario U
ESEC_ETL	VULN_INFO	193 SELECT	Tabla de usuario U
ESEC_ETL	VULN_RSRC	193 SELECT	Tabla de usuario U
ESEC_ETL	VULN_RSRC_SCAN	193 SELECT	Tabla de usuario U
ESEC_ETL	VULN_SCAN	193 SELECT	Tabla de usuario U
ESEC_ETL	VULN_SCAN_VULN	193 SELECT	Tabla de usuario U
ESEC_ETL	VULN_SCANNER	193 SELECT	Tabla de usuario U
ESEC_ETL	WORKFLOW_DEF	193 SELECT	Tabla de usuario U
ESEC_ETL	WORKFLOW_INFO	193 SELECT	Tabla de usuario U

ESEC_USER

Nombre de la función	Nombre del objeto	Acción	Tipo
ESEC_USER	ADV_ALERT_CVE_RPT_V	193 SELECT	V View
ESEC_USER	ADV_ALERT_PRODUCT_RPT_V	193 SELECT	V View
ESEC_USER	ADV_ALERT_RPT_V	193 SELECT	V View
ESEC_USER	ADV_ATTACK_ALERT_RPT_V	193 SELECT	V View
ESEC_USER	ADV_ATTACK_CVE_RPT_V	193 SELECT	V View
ESEC_USER	ADV_ATTACK_MAP_RPT_V	193 SELECT	V View
ESEC_USER	ADV_ATTACK_PLUGIN_RPT_V	193 SELECT	V View
ESEC_USER	ADV_ATTACK_RPT_V	193 SELECT	V View
ESEC_USER	ADV_CREDIBILITY_RPT_V	193 SELECT	V View
ESEC_USER	ADV_FEED_RPT_V	193 SELECT	V View
ESEC_USER	ADV_PRODUCT_RPT_V	193 SELECT	V View
ESEC_USER	ADV_PRODUCT_SERVICE_PACK_RPT_V	193 SELECT	V View
ESEC_USER	ADV_PRODUCT_VERSION_RPT_V	193 SELECT	V View
ESEC_USER	ADV_SEVERITY_RPT_V	193 SELECT	V View
ESEC_USER	ADV_SUBALERT_RPT_V	193 SELECT	V View

Nombre de la función	Nombre del objeto	Acción	Tipo
ESEC_USER	ADV_URGENCY_RPT_V	193 SELECT	V View
ESEC_USER	ADV_VENDOR_RPT_V	193 SELECT	V View
ESEC_USER	ADV_VULN_PRODUCT_RPT_V	193 SELECT	V View
ESEC_USER	ANNOTATIONS_RPT_V	193 SELECT	V View
ESEC_USER	ASSET_CATEGORY_RPT_V	193 SELECT	V View
ESEC_USER	ASSET_HOSTNAME_RPT_V	193 SELECT	V View
ESEC_USER	ASSET_IP_RPT_V	193 SELECT	V View
ESEC_USER	ASSET_LOCATION_RPT_V	193 SELECT	V View
ESEC_USER	ASSET_RPT_V	193 SELECT	V View
ESEC_USER	ASSET_VALUE_RPT_V	193 SELECT	V View
ESEC_USER	ASSET_X_ENTITY_X_ROLE_RPT_V	193 SELECT	V View
ESEC_USER	ASSOCIATIONS_RPT_V	193 SELECT	V View
ESEC_USER	ATTACHMENTS_RPT_V	193 SELECT	V View
ESEC_USER	CONFIGS_RPT_V	193 SELECT	V View
ESEC_USER	CONTACTS_RPT_V	193 SELECT	V View
ESEC_USER	CORRELATED_EVENTS	193 SELECT	V View
ESEC_USER	CORRELATED_EVENTS_RPT_V	193 SELECT	V View
ESEC_USER	CORRELATED_EVENTS_RPT_V1	193 SELECT	V View
ESEC_USER	CRITICALITY_RPT_V	193 SELECT	V View
ESEC_USER	CUST_RPT_V	193 SELECT	V View
ESEC_USER	ENTITY_TYPE_RPT_V	193 SELECT	V View
ESEC_USER	ENV_IDENTITY_RPT_V	193 SELECT	V View
ESEC_USER	ESEC_DISPLAY_RPT_V	193 SELECT	V View
ESEC_USER	ESEC_PORT_REFERENCE_RPT_V	193 SELECT	V View
ESEC_USER	ESEC_PROTOCOL_REFERENCE_RPT_V	193 SELECT	V View
ESEC_USER	ESEC_SEQUENCE_RPT_V	193 SELECT	V View
ESEC_USER	esec_toBase	224 EXECUTE	NULL
ESEC_USER	esec_toDecimal	224 EXECUTE	NULL
ESEC_USER	esec_toIpChar	224 EXECUTE	NULL
ESEC_USER	EVENTS	193 SELECT	V View
ESEC_USER	EVENTS_ALL_RPT_V	193 SELECT	V View
ESEC_USER	EVENTS_ALL_RPT_V1	193 SELECT	V View
ESEC_USER	EVENTS_ALL_V	193 SELECT	V View
ESEC_USER	EVENTS_RPT_V	193 SELECT	V View
ESEC_USER	EVENTS_RPT_V1	193 SELECT	V View
ESEC_USER	EVENTS_RPT_V2	193 SELECT	V View
ESEC_USER	EVT_AGENT_RPT_V	193 SELECT	V View
ESEC_USER	EVT_ASSET_RPT_V	193 SELECT	V View
ESEC_USER	EVT_DEST_EVT_NAME_SMRY_1	193 SELECT	V View
ESEC_USER	EVT_DEST_EVT_NAME_SMRY_1_RPT_V	193 SELECT	V View
ESEC_USER	EVT_DEST_SMRY_1	193 SELECT	V View
ESEC_USER	EVT_DEST_SMRY_1_RPT_V	193 SELECT	V View
ESEC_USER	EVT_DEST_TXNMY_SMRY_1	193 SELECT	V View
ESEC_USER	EVT_DEST_TXNMY_SMRY_1_RPT_V	193 SELECT	V View
ESEC_USER	EVT_NAME_RPT_V	193 SELECT	V View
ESEC_USER	EVT_PORT_SMRY_1	193 SELECT	V View

Nombre de la función	Nombre del objeto	Acción	Tipo
ESEC_USER	EVT_PORT_SMRY_1_RPT_V	193 SELECT	V View
ESEC_USER	EVT_PRTCL_RPT_V	193 SELECT	V View
ESEC_USER	EVT_RSRC_RPT_V	193 SELECT	V View
ESEC_USER	EVT_SEV_SMRY_1	193 SELECT	V View
ESEC_USER	EVT_SEV_SMRY_1_RPT_V	193 SELECT	V View
ESEC_USER	EVT_SRC_SMRY_1	193 SELECT	V View
ESEC_USER	EVT_SRC_SMRY_1_RPT_V	193 SELECT	V View
ESEC_USER	EVT_TXNMY_RPT_V	193 SELECT	V View
ESEC_USER	EVT_USR_RPT_V	193 SELECT	V View
ESEC_USER	EXTERNAL_DATA_RPT_V	193 SELECT	V View
ESEC_USER	HIST_CORRELATED_EVENTS	193 SELECT	V View
ESEC_USER	HIST_CORRELATED_EVENTS_RPT_V	193 SELECT	V View
ESEC_USER	HIST_EVENTS	193 SELECT	V View
ESEC_USER	HIST_EVENTS_RPT_V	193 SELECT	V View
ESEC_USER	IMAGES_RPT_V	193 SELECT	V View
ESEC_USER	INCIDENTS_ASSETS_RPT_V	193 SELECT	V View
ESEC_USER	INCIDENTS_EVENTS_RPT_V	193 SELECT	V View
ESEC_USER	INCIDENTS_RPT_V	193 SELECT	V View
ESEC_USER	INCIDENTS_VULN_RPT_V	193 SELECT	V View
ESEC_USER	L_STAT_RPT_V	193 SELECT	V View
ESEC_USER	LOGS_RPT_V	193 SELECT	V View
ESEC_USER	NETWORK_IDENTITY_RPT_V	193 SELECT	V View
ESEC_USER	ORGANIZATION_RPT_V	193 SELECT	V View
ESEC_USER	PERSON_RPT_V	193 SELECT	V View
ESEC_USER	PHYSICAL_ASSET_RPT_V	193 SELECT	V View
ESEC_USER	PRODUCT_RPT_V	193 SELECT	V View
ESEC_USER	ROLE_RPT_V	193 SELECT	V View
ESEC_USER	SENSITIVITY_RPT_V	193 SELECT	V View
ESEC_USER	STATES_RPT_V	193 SELECT	V View
ESEC_USER	UNASSIGNED_INCIDENTS_RPT_V	193 SELECT	V View
ESEC_USER	USERS_RPT_V	193 SELECT	V View
ESEC_USER	VENDOR_RPT_V	193 SELECT	V View
ESEC_USER	VULN_CALC_SEVERITY_RPT_V	193 SELECT	V View
ESEC_USER	VULN_CODE_RPT_V	193 SELECT	V View
ESEC_USER	VULN_INFO_RPT_V	193 SELECT	V View
ESEC_USER	VULN_RPT_V	193 SELECT	V View
ESEC_USER	VULN_RSRC_RPT_V	193 SELECT	V View
ESEC_USER	VULN_RSRC_SCAN_RPT_V	193 SELECT	V View
ESEC_USER	VULN_SCAN_RPT_V	193 SELECT	V View
ESEC_USER	VULN_SCAN_VULN_RPT_V	193 SELECT	V View
ESEC_USER	VULN_SCANNER_RPT_V	193 SELECT	V View

Funciones del servidor de Sentinel

Función del servidor	Descripción	Usuario de Sentinel
sysadmin	Administradores del sistema	esecdba
securityadmin	Administradores de seguridad	esecapp
serveradmin	Administradores del servidor	esecdba
setupadmin	Administradores de configuración	
processadmin	Administradores de procesos	
diskadmin	Administradores de disco	
dbcreator	Creadores de bases de datos	
bulkadmin	Administradores para inserción masiva	

Usuarios y permisos de base de datos con autenticación de dominio de Windows

Se asociará un usuario de dominio con el usuario esecadm, esecapp, esecdba y esecrpt según la configuración en el momento de la instalación. Tales usuarios de dominio tendrán los mismos privilegios que los especificados en las secciones anteriores.

D Tablas de permisos de servicios de Sentinel

NOTA: El término Agente puede intercambiarse con Recopilador. En adelante, los agentes se denominarán recopiladores.

Servidor de Sentinel (Motor de correlación)

Componente de Sentinel	Aplicación de Sentinel	Servicio de Sentinel	Proceso de Sentinel	Resumen de función	Permisos requeridos	Explicación del permiso
Servidor de Sentinel	-	Sentinel / WatchDog.exe	correlation_engine.exe	El proceso del motor de correlación (correlation_engine) recibe eventos del Gestor de recopiladores del asistente y publica los eventos correlacionados en función de las reglas de correlación definidas por el usuario.	Acceso de red; acceso de lectura a archivos de configuración modificados.	Se comunica con Sonic para procesos de configuración y eventos y generación de eventos correlacionados. Necesita acceso a archivos si se utiliza un archivo de configuración modificado.

Gestor de recopiladores

Componente de Sentinel	Aplicación de Sentinel	Servicio de Sentinel	Proceso de Sentinel	Resumen de función	Tipo de conector	Permisos requeridos	Explicación del permiso
Asistente de Sentinel / Gestor de recopiladores	-	Gestor de recopiladores	agentengine.exe	El proceso del Gestor de recopiladores gestiona Motores de recopiladores (genera los procesos del Motor del recopilador), publica mensajes de estado del sistema, realiza el filtrado global de eventos y asignaciones referenciales. El proceso del Motor del recopilador ejecuta guiones de un Recopilador, que normaliza los eventos sin procesar de los dispositivos y sistemas de seguridad.	**NOTA**: En función de los tipos de conexión, el Gestor de recopiladores necesitará diferentes permisos.		
					Serie: datos leídos de un puerto en serie RS-232C	Permiso de lectura/escritura a un puerto serie	Lectura/escritura del Motor del recopilador a un puerto serie
					Zócalo: una conexión por zócalo TPC	Acceso de red: lectura/escritura desde un zócalo de red; Permiso para iniciar una conexión	El Motor del recopilador inicia una conexión a un punto final de la red, y tiene permiso de lectura/escritura a dicho zócalo
					Nuevo en archivo: lee sólo datos de eventos de seguridad que se añaden a un archivo después de que el guión se haya iniciado (lee desde el final del archivo).	Acceso de lectura/escritura a archivo	El Motor del recopilador lee desde el primer archivo especificado y escribe en el segundo archivo especificado
					Todo en archivo: lee todos los datos de eventos de seguridad en un archivo.	Acceso de lectura/escritura a archivo	El Motor del recopilador lee desde el primer archivo especificado y escribe en el segundo archivo especificado

Componente de Sentinel	Aplicación de Sentinel	Servicio de Sentinel	Proceso de Sentinel	Resumen de función	Tipo de conector	Permisos requeridos	Explicación del permiso
					Proceso permanente: lanza un proceso permanente cuando el puerto se ha iniciado, se comunica con el recopilador asignado a ese puerto y a una aplicación externa a través de estados de recepción y transmisión y continúa en ejecución durante la vida activa del puerto.	Permiso para ejecutar el proceso permanente definido. (Nota: Si utiliza EventLog.exe como el proceso permanente para recopilar el registro NT utilizando WMI, el Gestor de recopiladores necesita permiso para acceder a WMI)	El Motor del recopilador ejecuta el proceso definido en el nivel de permiso actual
					Proceso transitorio: comunica al Recopilador asignado con el puerto y con la aplicación externa a través de estados de recepción y de transmisión. El proceso transitorio puede iniciarse en varias ocasiones.	Permiso para ejecutar el proceso transitorio definido	El Motor del recopilador ejecuta el proceso definido en el nivel de permiso actual
					SNMP: Recibe mensajes de alerta SNMP v1, v2 y v3.	Acceso de red: lectura/escritura desde el zócalo de red	El Gestor de recopiladores envía / recibe mensajes de alerta SNMP
					Ninguno	NA	NA

Componente de Sentinel	Aplicación de Sentinel	Servicio de Sentinel	Proceso de Sentinel	Resumen de función	Tipo de conector	Permisos requeridos	Explicación del permiso
Asistente de Sentinel / Generador de recopiladores	Generador de recopiladores	-	agentbuilder.exe	Una GUI que permite generar, configurar y controlar Recopiladores. La GUI puede utilizarse para ejecutar Recopiladores locales o controlar Recopiladores en sistemas de Asistente remotos.	Acceso de lectura/escritura a archivo		Generador de recopiladores lee/escribe Guiones del recopilador en %WORKBENCH_HOME%/ Elements
					Acceso de lectura/escritura a archivo		Generador de recopiladores lee/escribe archivo de config de puerto en %WORKBENCH_HOME%/ Agents
					Acceso de lectura/escritura a archivo		Generador de recopiladores accede a %ESEC_HOME%/.uuid
					Acceso de red: lectura/escritura desde un zócalo de red; Permiso para iniciar una conexión		Generador de recopiladores carga/descarga los Recopiladores y recibe mensajes de actividad del Gestor de recopiladores

Comunicación de Sentinel

Componente de Sentinel	Aplicación de Sentinel	Servicio de Sentinel	Proceso de Sentinel	Resumen de función	Permisos requeridos	Explicación del permiso
iSCALE / MOM	SonicMQ	Comunicación de Sentinel	sonicmf.exe	Para Windows, Sentinel Communication es un servicio y se denomina iSCALE: un Software de mensajería corporativa (MOM). El componente iSCALE proporciona una estructura de JMS (Java Message Service, Servicio de mensajes (o mensajería) de Java) para la comunicación entre procesos. Los procesos se comunican a través de un intermediario, que se ocupa del encaminamiento y buffer de los mensajes. Los diversos intermediarios pueden comunicarse entre sí para atravesar los cortafuegos y para balancear la carga. Los procesos de Sentinel utilizan un mecanismo de editor/suscriptor para comunicarse entre sí. Esto permite que un proceso publique un mensaje en un canal de temas que utilizan varios suscriptores, sin que el proceso de publicación sepa que procesos se han suscrito. Los suscriptores pueden recibir mensajes publicados de los editores sin saber qué editores están disponibles. De esta manera, se reduce la configuración y se aumenta la estabilidad y capacidad de ampliación del sistema. Por ejemplo, cuando se añade un asistente nuevo al sistema, no se requiere ninguna configuración en Sentinel. El proceso del editor publica mensajes en temas (canal), y los procesos del suscriptor se suscriben a los temas El intermediario de mensajes encamina los mensajes de los editores a los suscriptores en función de los temas a los que se han registrado.	Permisos para acceder a su propia base de datos incrustada, directorio de instalación (%ESEC_HOME%\3rdparty\ SonicMQ) y archivos	Sonic accede a su propia base de datos incrustada y directorio de instalación (%ESEC_HOME%\3rdparty \SonicMQ) y archivos

Servidor de la base de datos (sin DAS)

Componente de Sentinel	Aplicación de Sentinel	Servicio de Sentinel	Proceso de Sentinel	Resumen de función	Permisos requeridos	Explicación del permiso
-	-	-	-	Configurar la base de datos de Sentinel	-	El controlador odbc o el controlador de Oracle debe apuntar a la base de datos de Sentinel

Servidor de la base de datos (con DAS)

Para obtener un resumen o desglose de los permisos de acceso a la base de datos de Sentinel, consulte los detalles en la documentación siguiente:
Apéndice A - Usuarios, funciones y permisos de acceso de la base de datos de Sentinel

Componente de Sentinel	Aplicación de Sentinel	Servicio de Sentinel	Proceso de Sentinel	Resumen de función	Permisos requeridos	Explicación del permiso
-	-	-	-	Configurar la base de datos de Sentinel	-	El controlador odbc o el controlador de Oracle debe apuntar a la base de datos de Sentinel
Servidor de Sentinel	-	Sentinel / WatchDog.exe	das_binary	operaciones de inserción de eventos y eventos correlacionados	Acceso de red; requiere acceso a BD para instancia ESEC como ESECAPP	Se comunica con Sonic. Se comunica con la BD a través de JDBC para la recuperación de datos y con ADO para la inserción de eventos si se ha definido la estrategia de carga de ADO
			das_query	se utiliza para el resto de operaciones de la base de datos.	Acceso de red; requiere acceso a BD para instancia ESEC como ESECAPP; requiere permiso para ejecutar procesos	Se comunica con Sonic. Se comunica con la BD a través de JDBC para la recuperación de datos.
			activity_container	ejecución y configuración de servicio de actividades	Acceso de red; requiere acceso a BD para instancia ESEC como ESECAPP; requiere permiso para ejecutar procesos	Se comunica con Sonic. Se comunica con la BD a través de JDBC para la recuperación e inserción de datos.
			workflow_container	configuración del servicio de flujo de trabajo (iTRAC)	Acceso de red; requiere acceso a BD para instancia ESEC_WF como ESECAPP; requiere permiso para ejecutar procesos	Se comunica con Sonic. Se comunica con la BD a través de JDBC para la recuperación e inserción de datos.

Para obtener un resumen o desglose de los permisos de acceso a la base de datos de Sentinel, consulte los detalles en la documentación siguiente:
Apéndice A - Usuarios, funciones y permisos de acceso de la base de datos de Sentinel

Componente de Sentinel	Aplicación de Sentinel	Servicio de Sentinel	Proceso de Sentinel	Resumen de función	Permisos requeridos	Explicación del permiso
			das_rt	configuración de la función Active Views en la consola de control de Sentinel	Acceso de red; requiere acceso a BD para instancia ESEC como ESECAPP	Se comunica con Sonic. Se comunica con la BD a través de JDBC para la recuperación de datos.

Servidor de informes

Componente de Sentinel	Aplicación de Sentinel	Servicio de Sentinel	Proceso de Sentinel	Resumen de función	Permisos requeridos	Explicación del permiso
-	-	-	-	Crystal Reports XI o Crystal Enterprise 9 Standard es una de las herramientas de generación de informes de Sentinel.	-	El controlador odbc o el controlador de Oracle debe apuntar a la base de datos de Sentinel

Glosario

NOTA: El término Agente puede intercambiarse con Recopilador. En adelante, los agentes se denominarán recopiladores.

Agente

Consulte Recopilador

Análisis

En el Centro de control de Sentinel, permite la generación de informes. Los informes históricos y de vulnerabilidades se publican en un servidor Web Crystal®, éstos se ejecutan directamente en la base de datos y aparecen en las pestañas Análisis y Asesor de la barra del navegador del Centro de control de Sentinel.

archivo de guión

En el asistente, un archivo compilado (*.asd) que consta del archivo de plantilla del recopilador, el archivo de parámetros, el archivo de búsquedas y el archivo de asignación.

Archivos de asignación

Para los recopiladores, los archivos de asignación son archivos opcionales (.csv) que permiten la búsqueda rápida de entradas clave. El archivo csv es una ruta relativa desde un directorio de guión del recopilador. La edición de estos archivos no está disponible actualmente en el Generador de recopiladores, pero los archivos se pueden editar con Excel.

Archivos de búsqueda

Para los recopiladores, los archivos de búsqueda son tablas opcionales (.lkp files) que se comparan con los valores recibidos para determinar qué acciones realizar, si corresponde, como respuesta a eventos de seguridad. Los archivos de búsqueda contienen cláusulas de concordancia que se utilizan para comparar cadenas individuales. El comando LOOKUP determinará si la cadena de búsqueda se encuentra o no basándose en las cláusulas de un archivo de búsqueda específico y en los datos recibidos de los sensores.

De manera opcional, los comandos de análisis pueden estar asociados a la cadena de concordancia. Los comandos de análisis se ejecutan si se encuentra una concordancia.

Archivos de parámetro

Para los recopiladores, los archivos de parámetros (archivos .par) son tablas que se utilizan para definir nombres de parámetros en los archivos de guiones de ejecución asociados. Se utilizan cuando se hace referencia a ellos en el código de análisis. Los parámetros son equivalentes a las variables. Los parámetros se almacenan como cadenas. Cualquier valor numérico tiene que convertirse en una cadena para su manipulación. Cuando se introducen nuevos valores para los parámetros, éstos se activan después de generar el guión. Se fusionan con el archivo de plantilla cuando se crea el guión.

Los nombres de los archivos de guiones de ejecución se muestran en la primera fila de la tabla y los nombres y niveles de los parámetros se muestran en la primera columna de la tabla. La segunda fila de la tabla se utiliza para definir los iconos que aparecen en el árbol del recopilador. La fila restante define las variables o los valores de los parámetros que se utilizan para el parámetro ya que la fila está relacionada con un guión en concreto.

Los valores dentro del archivo de parámetros son:

- Meta-etiquetas, información y comentarios; existen más de 200 meta-etiquetas disponibles, de las cuales 100 son configurables por el usuario y el resto están reservadas.
- Regla: los nombres de los archivos configurados aparecen en la fila del encabezado de la tabla, mientras que los parámetros aparecen en la primera columna de la tabla.
- Mapa de bits: la segunda fila de la tabla define el mapa de bits utilizado para ese archivo. El mapa de bits aparecerá en la lista de recopiladores.

Archivos de plantilla

En los recopiladores, puede crear y añadir estados, así como editar y eliminar plantillas. Las plantillas determinan cómo se procesarán los registros. La mayoría de las decisiones sobre plantillas se centran en los tipos de registros con los que está trabajando y su formato. Existe un archivo de plantilla equivalente con la extensión .tem.

Los archivos de plantilla están basados en estados. Un estado es un punto de decisión dentro del flujo lógico o ruta de una plantilla. Cada punto (estado) contiene información sobre el próximo proceso que se va a realizar. Los estados incluyen parámetros, cuando la plantilla se fusiona con un archivo de parámetros, valores específicos reemplazan a los parámetros. Cuando los parámetros se reemplazan por valores específicos, se crea uno o más archivos de guión.

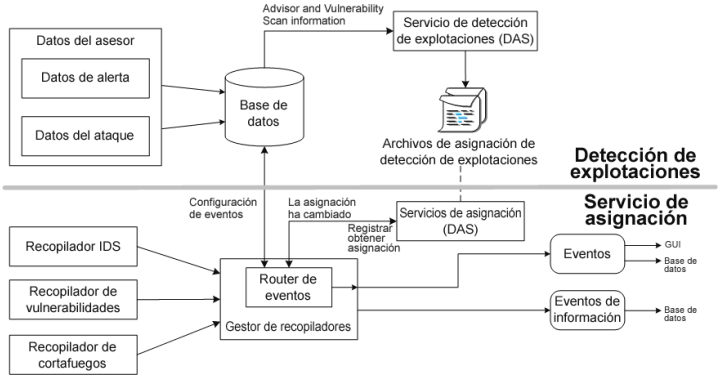
Al insertar un estado a una plantilla, se asigna un número que permanece con él aunque la plantilla cambie de ubicación.

Asesor

Un sistema integrado con base de datos de vulnerabilidades SecurityNexus que ofrece referencias cruzadas entre eventos en tiempo real y vulnerabilidades conocidas.

Asignación de servicios

El servicio de asignación de Sentinel permite la notificación inmediata y accionable de ataques en sistemas vulnerables. Ofrece un enlace en tiempo real entre los eventos y los resultados de la exploración de vulnerabilidad, de manera que los usuarios reciban una notificación automática e inmediata cuando un ataque intenta explotar un sistema vulnerable. Esto mejora la eficiencia y la efectividad de la respuesta en caso de incidencias, y resulta en una mayor disponibilidad de los sistemas críticos y una seguridad altamente rentable.



Asistente

El Generador de recopiladores y el Gestor de recopiladores.

Centro de control de Sentinel

El Centro de control de Sentinel es la consola de gestión central que se utiliza para ver pantallas de resumen, informes históricos, eventos de filtros en tiempo real y crear incidencias. El Centro de control de Sentinel ofrece la visualización de eventos en tiempo real, información general del sistema sobre cambios en la actividad activados por valores definidos en los recopiladores, administración de archivos, generación de informes, reglas correlacionadas, así como gestión de filtros globales y eventos de seguridad a través de incidencias.

comando de análisis

En el asistente, una interfaz de guiones de alto nivel que permite la manipulación de los datos. El análisis es el proceso de desglosar un evento en sus componentes.

Configuración de eventos

La configuración de eventos (parte del servicio de asignación) permite:

- Habilitar la monitorización del cumplimiento con las normas reguladoras
- Habilitar la conformidad con directivas
- Habilitar la definición de prioridades de las respuestas
- Habilitar el análisis de datos de seguridad en relación con las operaciones comerciales
- Mejorar de la responsabilidad

La configuración de eventos consiste en la asignación de nombres a las etiquetas existentes. Por ejemplo, cambiar el nombre Ct2 por Ciudad. Los cambios se transmiten a filtros y reglas de correlación.

Consulta rápida

Consulte Gestor de consultas

Controlador de datos

Consulte Proceso del sincronizador de datos

Correlación

El proceso de analizar eventos de seguridad para identificar relaciones potenciales entre dos o más eventos. La correlación permite establecer una asociación rápida de ataques de prioridad según elementos comunes de datos de evento. Las tendencias o patrones de eventos de nivel inferior diseñados para operar por debajo de umbrales de seguridad se pueden identificar más efectivamente con la correlación.

Sentinel ofrece cinco tipos de reglas de correlación. Son los siguientes:

- Lista de vigilancia
- Correlación básica
- Correlación avanzada
- RuleLg sin formato

CorrelatedEventUUID

El identificador de evento del evento correlacionado generado por la regla que se ha activado.

das_aggregation.xml

Se utiliza para la operación de adición.

das_binary.xml

Se utiliza para la operación de inserción de eventos y de eventos correlacionados.

das_itrac.xml

Se utiliza para ejecutar y configurar el servicio de actividades y para configurar el servicio de flujo de trabajo.

das_query.xml

Especifica los parámetros de configuración para el Servicio de acceso a los datos (DAS), un componente de la base de datos de Sentinel.

das_rt.xml

Especifica la configuración de la función Active Views en la consola de control de Sentinel

Detección de explotaciones

Consulte Servicio de asignación

evento

Un evento es una acción o un acontecimiento detectado por un dispositivo de seguridad (evento externo) o un proceso (interno). Los eventos pueden estar relacionados con la seguridad, con el rendimiento o con la información. Por ejemplo, un evento externo podría ser un ataque detectado por un sistema de detección de intrusos (IDS), un inicio de sesión correcto detectado por un sistema operativo o una situación definida por el usuario como, por ejemplo, un usuario que accede a un archivo. Los eventos relacionados con la información son eventos internos. Los eventos internos indican un cambio en el estado de un proceso. Por ejemplo, el cierre de un puerto.

Eventos del sistema

Los eventos internos o del sistema son un medio para informar sobre el estado y el cambio de estado del sistema. Existen dos tipos de eventos generados por el sistema, estos son:

- Eventos internos
- Eventos de rendimiento

Los eventos internos son informativos y describen un único estado o cambio de estado en el sistema. Generan un informe cuando un usuario inicia sesión o no puede autenticar, cuando se inicia un proceso o se activa una regla de correlación. Los eventos de rendimiento se generan periódicamente y describen los recursos medios utilizados por diferentes partes del sistema.

Eventos internos

Consulte Eventos del sistema

Filtros

Los filtros de Sentinel permiten procesar datos según criterios específicos tanto para los eventos que ingresan al sistema como para los usuarios del sistema. Existen varios niveles de filtros:

- Recopilador: que se realiza a través del guión utilizando el Generador de recopiladores.
- Filtro global: se aplica equitativamente en todos los eventos generados por todos los asistentes el sistema. Sólo los eventos que atraviesan los filtros globales se envían a todos los procesos de Sentinel.
- Filtro de seguridad: se aplica a los usuarios activos. Estos filtros restringen los eventos que pueden observar los usuarios activos y son asignados por el administrador.
- Filtro de visualización: se aplican a las vistas de la interfaz. Estos filtros permiten que el usuario defina las ventanas de eventos para realizar análisis en tiempo real. Estos filtros se aplican por cada usuario.

Existen dos tipos de filtros:

- public (públicos): los filtros públicos son propiedad del sistema y se pueden utilizar como filtros de seguridad o de visualización. Los filtros de seguridad se basan en los permisos de usuario y los filtros de visualización determinan los eventos que se mostrarán en las tablas, los diagramas y los gráficos de eventos en tiempo real.
- private (privados): los filtros privados son propiedad del usuario y son filtros de visualización que se pueden compartir si el usuario tiene el permiso de visualización de filtros privados.

flujo de trabajo

Consulte iTRAC™

Generador de agentes

Consulte Generador de recopiladores

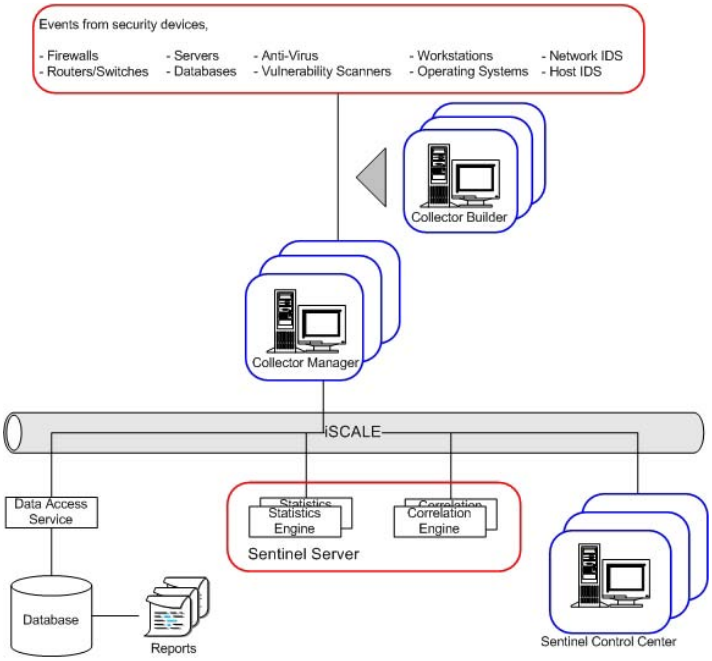
Generador de recopiladores

Una GUI que le permite crear recopiladores basados en reglas para recopilar, filtrar y normalizar datos provenientes de distintos orígenes y comunicar de manera segura información importante al servidor de Sentinel que se puede utilizar para supervisar el tráfico.

Gestión de activos	El propósito de la gestión de activos es vincular uno o más eventos a información de activos y vulnerabilidades a fin de desarrollar un método para proteger los activos de la organización con eficiencia. Existen dos tipos de activos, físicos y de software. Los activos físicos son el hardware y los activos de software son los servicios y las aplicaciones.
Gestor de agentes	Gestor de recopiladores
Gestor de recopiladores	El sistema de apoyo del asistente que administra los recopiladores y los mensajes de estado del sistema.
Host del asistente	Toda máquina que tenga instalado el software del Gestor de recopiladores.
Incidencias	La agrupación de un conjunto de eventos en su totalidad que representa algo de interés (grupo de eventos similares o conjunto de eventos diferentes que indiquen un patrón de interés como un ataque).

iSCALE™

El bus de mensajes proporciona una estructura JMS (Java Message Service) para la comunicación entre procesos. Los procesos se comunican a través de un intermediario, que se ocupa del encaminamiento y buffer de los mensajes. Los diversos intermediarios pueden comunicarse entre sí para atravesar los cortafuegos y para balancear la carga.



Los siguientes procesos se comunican entre sí mediante el bus de mensajes.

- Paquete de vigilancia
- Rendimiento del evento (motor de filtros)
- Recuento de eventos a lo largo del tiempo (motor de estadísticas)
- Sincronizador de datos (Controlador de datos)
- Correlation Engine
- Verificador RuleLg (verificador de reglas de correlación)
- Servicio de acceso a los datos (DAS)
- Query Manager

iTRAC™

iTRAC implica la automatización de los procedimientos, la capacidad para responder a las incidencias. Sentinel ofrece un sistema de gestión de flujo de trabajo que brinda la automatización de los procedimientos del proceso de gestión de incidencias de SANS. Las partes principales de iTRAC son:

- Gestor de lista de trabajo: aplicación que se utiliza para desplazarse de una actividad a otra.
- Generador de actividades: aplicación que se utiliza para crear su propio iTRAC personalizado.
- Monitor de procesos: supervisa las actividades (pasos) que se realizan para completar un proceso.

metadatos

Los metadatos son información sobre datos, nombres de variables predefinidas para los metadatos. Por ejemplo, la dirección IP de origen de un ataque se almacena en la meta-etiqueta IP de origen. Los nombres de producto se almacenan en la meta-etiqueta Nombre de producto. Los datos que se utilizan para completar las meta-etiquetas se extraen de los datos del evento o se configuran como parte del procesamiento del recopilador.

meta-etiqueta

Las meta-etiquetas almacenan metadatos.

Middleware orientado a mensajes

Consulte iSCALE™

MOM

Consulte iSCALE™

Motor de agentes

Consulte Motor del recopilador

Motor de correlación

El motor de correlación realiza un análisis de los eventos entrantes para buscar patrones de interés y búsquedas detalladas en los eventos de correlación a fin de determinar los detalles que activan una regla.

Motor de estadísticas

Consulte Proceso de recuento de eventos a lo largo del tiempo

Motor de filtros

Consulte Proceso de rendimiento del evento

Motor del recopilador

Procesa la lógica de la plantilla para cada puerto. Un motor del recopilador ejecuta un puerto correspondiente.

Normalización de adiciones y eventos

La adición es el proceso por el cual se toman elementos de datos individuales de baja importancia y se los combina, lo que da como resultado un elemento de datos que podría ser de mayor importancia. Las partes individuales de un evento, como el nombre del evento, la fecha, la dirección IP de destino y origen, el UUID, el tipo de sensor y demás, pueden no tener mucho sentido por sí mismas. No obstante, si se las agrupa se crea un evento que podría ser de interés y que podría ser un ataque en la red que resulte en una posible explotación de un activo. El hecho de guardar un evento completo provoca el almacenamiento de información duplicada. Por ejemplo, en un sistema que no sea de adición, en el caso de diez eventos que sean idénticos, excepto por la fecha, se guardará cada evento, lo que resultará en elementos de datos idénticos (nombre del evento, tipo de sensor, etc...) que se guardan diez veces. La adición guardará los elementos de datos idénticos sólo una vez y posteriormente, mantendrá una cuenta en ejecución durante una hora.

Los datos de eventos se transforman, resumen y guardan en tablas de resumen. Los informes de resumen se pueden luego ejecutar sobre resúmenes previamente computados, lo que hace que las consultas tengan menos contenido en las tablas de eventos en tiempo real. El motor de adición de eventos captura datos de eventos binarios, los transforma en una estructura de eventos normalizada y los resume en función de un conjunto predefinido de definiciones de resumen. El motor de adición de eventos procesa los eventos prácticamente en tiempo real y con una sobrecarga mínima en el tiempo real del sistema.

Normalización de eventos

Consulte Adición

Número del ID de evento

Un número asignado a un evento.

Proceso de vigilancia

El proceso del vigilante es un proceso de Sentinel que gestiona todos los demás procesos de Sentinel. Si se detiene un proceso que no sea el del vigilante, éste reiniciará el proceso.

Proceso del Gestor de consultas (query_manager)

El gestor de consultas (query_manager) recibe peticiones de consulta rápida y detalle desde el Centro de control de Sentinel y las remite a la base de datos mediante DAS. Las peticiones del Centro de control de Sentinel definen los eventos necesarios a través de un criterio o un filtro. Si se utiliza un filtro, el Gestor de consultas recupera la definición del filtro y convierte el filtro en un criterio xml. A continuación, el gestor de consultas envía la petición a la base de datos. No todos los filtros pueden convertirse completamente en xml. Si el filtro se convierte completamente, el gestor de consultas ordena a DAS que envíe la respuesta directamente al Centro de control de Sentinel. Si el filtro contiene expresiones regulares que no pueden convertirse a xml, el gestor de consultas convierte todo lo posible y genera un criterio xml conservador que devuelve un superconjunto de los eventos requeridos. En ese caso, el Gestor de consultas solicita a DAS que devuelva el resultado al Gestor de consultas. Cuando la respuesta vuelve al gestor de consultas, éste la filtra en la memoria y envía los eventos que pasan el filtro al Centro de control de Sentinel.

Proceso del motor de correlación (correlation_engine)

El proceso del motor de correlación (correlation_engine) recibe eventos del Gestor de recopiladores del asistente y publica los eventos correlacionados en función de las reglas de correlación definidas por el usuario.

Proceso del Servicio de acceso a los datos (DAS)

El proceso del Servicio de acceso a los datos (DAS) es un servicio permanente del servidor de Sentinel y proporciona una interfaz de bus de mensajes (iSCALE) a la base de datos. Ofrece acceso basado en datos al sistema de apoyo de la base de datos. Recibe peticiones XML desde los diferentes procesos de Sentinel, las convierte en una consulta a la base de datos, procesa el resultado de la base de datos y lo vuelve a convertir en una respuesta XML. Admite peticiones para recuperar eventos para una consulta rápida y para el detalle de eventos, para recuperar información de vulnerabilidades e información del asesor y para modificar la información de configuración. DAS también gestiona la entrada de todos los eventos que se reciben desde el Gestor de recopiladores del asistente y peticiones para recuperar y almacenar información de configuración.

Proceso del sincronizador de datos (Controlador de datos)

El proceso del sincronizador de datos (data_synchronizer) gestiona la modificación de los datos de configuración por múltiples usuarios. Cuando un usuario solicita modificar los datos a través del Centro de control de Sentinel, data_synchronizer bloquea el registro de datos. Los detalles acerca de quién ha bloqueado los datos se publican en los otros centros de control de Sentinel activos y ningún otro usuario puede modificarlos. Si un Centro de control de Sentinel se cierra antes de desbloquear los datos que ha bloqueado, los bloqueos excederán el tiempo límite.

Proceso del verificador RuleLg (rulelg_checker)

El proceso del verificador RuleLg (rulelg_checker) valida las expresiones de filtros y reglas de correlación. El Centro de control de Sentinel utiliza estos resultados para determinar si se puede guardar un filtro o una regla de correlación.

Puerto

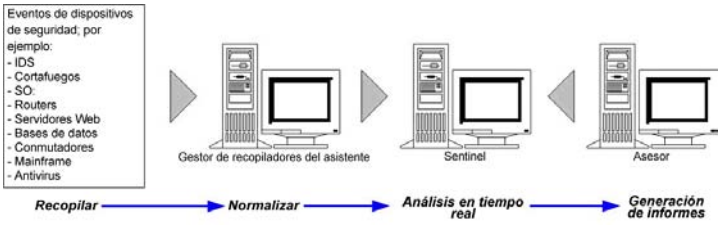
En el asistente, los puertos permiten que un recopilador ubique los datos de eventos de seguridad en la red al brindar la dirección IP y otra información acerca del origen (dispositivo de seguridad [router, IDS, conmutador, etc…]). Cada fila de la tabla Configuración del puerto ejecuta un guión del recopilador en un origen de evento.

Puntero Rx Buffer (Buffer de recepción)

El puntero Rx Buffer (Buffer de recepción) señala los bytes de datos en el buffer de recepción. Antes de cada cadena de decisión evaluada, el puntero Rx Buffer (Buffer de recepción) se restaura a su valor de retención (normalmente cero).

Recopilador

Un recopilador es el receptor que recopila y normaliza los eventos sin formato de los dispositivos y programas de seguridad y que devuelve los eventos normalizados que se pueden correlacionar, informar y utilizar en caso de repuesta a incidencias.



Existen tres niveles de recopiladores:

- Recopiladores admitidos (T1)
- Recopiladores documentados (T2)
- Recopiladores de muestra (T3)

Los recopiladores constan de:

- archivos de plantillas
- archivos de parámetros
- archivos de búsquedas
- archivos de asignaciones

Regla de correlación avanzada

Permite crear una regla de correlación que incorpora todas las funciones de la regla de correlación básica, así como enviar un evento cuando un conjunto de eventos posee valores de meta-etiquetas que son diferentes, por ejemplo un sensor que está dentro o fuera del cortafuegos. Por ejemplo, una regla de correlación avanzada puede buscar eventos desde la misma dirección IP de origen hasta la misma dirección IP de destino con el mismo nombre de evento y que se producen tanto dentro como fuera de un cortafuegos (lo que significa que puede que un ataque haya atravesado el cortafuegos).

Regla de correlación básica

Le permite seleccionar cualquiera de las meta-etiquetas para crear una regla de correlación que le permita contar el número de veces que se satisfacen determinadas condiciones en un plazo de tiempo específico. Por ejemplo, una regla de correlación básica puede buscar la misma dirección IP de origen informada cinco veces en cinco minutos, incluso si los eventos se informan desde productos distintos como, por ejemplo, un sistema de detección de intrusiones (IDS) y un cortafuegos.

Regla de correlación de lista de vigilancia

Permite especificar una cadena de texto que el motor de correlación vigilará en cada meta-etiqueta para cada evento entrante. Por ejemplo, una regla de lista de vigilancia puede buscar una dirección IP de origen específica de un pirata informático y notificarle cada vez que esa dirección IP aparece en un mensaje de evento.

Relevancia empresarial

Consulte Servicio de asignación

Router de eventos

El Router de eventos se encarga de la transformación y el filtrado de la asignación de eventos.

Rx Buffer (Buffer de recepción)

Parte del Gestor de recopiladores, el tamaño por defecto es de 50.000 eventos. El buffer de recepción es un parámetro que se puede editar. El tamaño mínimo es 5.000.

Secuencias (inicio y restitución)

Las secuencias de inicio y de restitución se asignan a un puerto que ejecuta las series de guiones que contiene cuando se inicia o detiene. Se debe incluir un guión en la secuencia de inicio o restitución para que el puerto lo pueda utilizar. Los puertos permiten al recopilador ubicar los hosts del asistente en la red a través de la dirección IP o el nombre de archivo del host. También proporcionan a Sentinel información sobre la ubicación de los sensores y del recopilador que se utiliza para gestionar datos para esos sensores. Se pueden configurar las opciones siguientes para los puertos:

- Tipo de conexión
- Nombre del proceso
- Información de zócalo
- Información de SNMP
- Nombres de archivos de entrada/salida
- Nombre del recopilador

Secuencias de inicio

Consulte Secuencias

Secuencias de restitución

Consulte Secuencias

Servidor de Sentinel

El servidor de Sentinel recibe información normalizada de eventos recopilada por los recopiladores del Gestor de recopiladores del asistente. El servidor de Sentinel correlaciona estos eventos para buscar patrones e identificar amenazas y genera en tiempo real informes de datos e históricos que pueden verse en el Centro de control de Sentinel.

Tiempo real del evento

Capacidad para supervisar los eventos a medida que se producen y realizar consultas en los mismos. Estos eventos se pueden supervisar en forma de tabla o a través de una representación gráfica tridimensional.

Visualización de vulnerabilidades

La representación gráfica de datos de eventos en tiempo real frente a sistemas vulnerables y está disponible en un evento para la vulnerabilidad del tiempo del evento y la hora actual.