SOCIA · CIBERSEGURIDAD

Fuerza bruta SSH sin acceso válido

Guía paso a paso de triaje, análisis, contención y cierre

Fecha de la práctica: 9 de agosto de 2026

Herramientas: TheHive, Cortex, Graylog, Malcolm, Arkime y OPNsense

Tiempo activo: 18:42

Contexto del caso:

Una alerta de Wazuh avisa de varios intentos de acceso SSH contra `vulnmachine`. El análisis parte de la IP `172.31.0.2` y busca responder tres preguntas: si hubo una autenticación válida, si el origen atacó otros equipos y qué medida de contención procede.

La práctica reparte el trabajo entre un analista N1 y otro N2. Este reparto sirve para mostrar una entrega ordenada, aunque cada organización puede asignar las tareas de otro modo.

Objetivos de aprendizaje

- Convertir una alerta en un caso trazable.
- Distinguir intentos fallidos de accesos válidos.
- Acotar el alcance con registros y sesiones de red.
- Documentar la contención, sus pruebas y sus límites.

Índice de contenidos

FASE 1

ANALISTA N1

Triaje y apertura del caso

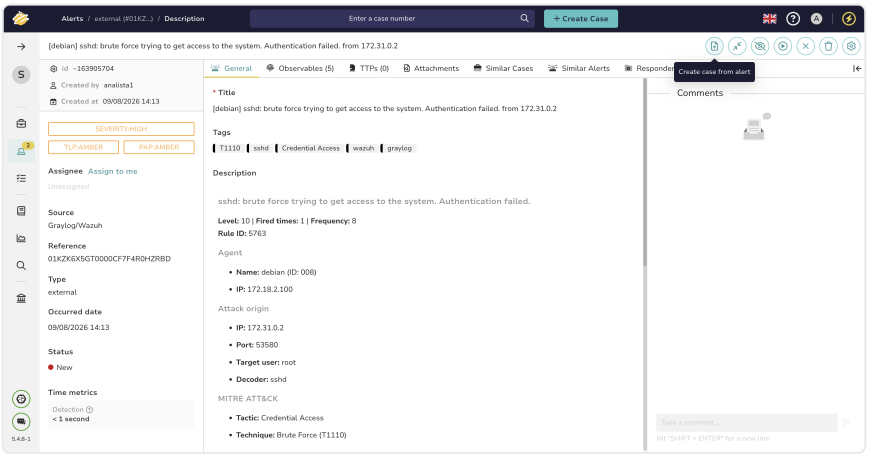
> TheHive es el punto central para seguir alertas y casos. Primero vamos a decidir qué alerta merece investigación, qué datos aporta y cómo convertirla en un expediente de trabajo.

Paso 1. Leer la alerta antes de actuar

Accedemos a **Alerts** y abrimos la alerta con más observables. Una alerta comunica una detección; un caso reúne el análisis, las decisiones y las pruebas que se generan después. Por eso no debemos saltar aún al bloqueo.

La descripción aporta el punto de partida: regla Wazuh `5763`, nivel 10, usuario objetivo `root`, origen `172.31.0.2`, puerto de origen `53580` y agente `debian` con IP `172.18.2.100`. El mensaje dice *Authentication failed*, lo que prueba un fallo concreto, pero no descarta otros accesos válidos dentro del mismo periodo.

> **Resultado:** Hipótesis inicial: intento de fuerza bruta SSH desde 172.31.0.2 contra 172.18.2.100; falta confirmar si algún acceso tuvo éxito.



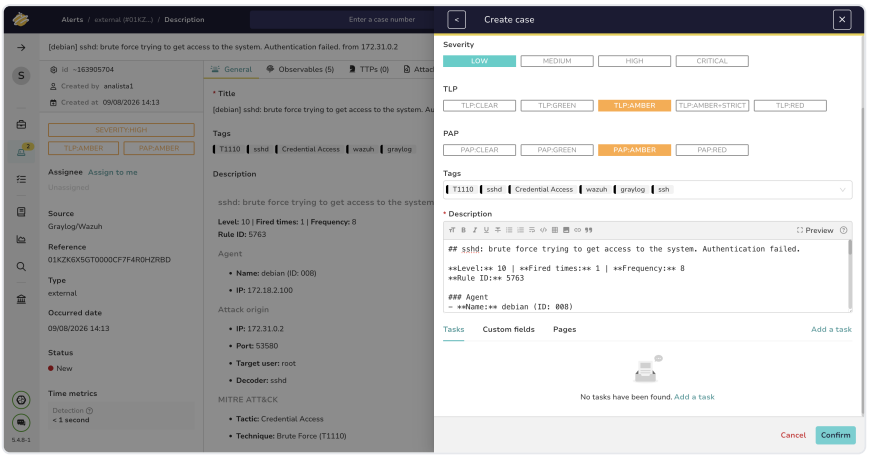
Detalle de la alerta en TheHive con el origen, el destino, el usuario y la regla 5763.

Paso 2. Crear un caso con datos útiles

Usamos **Create case from alert**. Si la organización dispone de una plantilla, conviene elegirla para mantener el mismo método en incidentes parecidos. La práctica no tiene plantilla, así que parte de un caso vacío y conserva la descripción de la alerta.

El título identifica el tipo de incidente, el activo y el origen: `Brute force SSH to from vulnmachine (172.18.2.100) from 172.31.0.2`. Bajamos la gravedad a **LOW** mientras no haya prueba de acceso válido o impacto. Se mantienen **TLP:AMBER** y **PAP:AMBER** según la política expuesta en clase, y se añade la etiqueta `ssh` junto a las que ya trae la alerta.

TLP limita con quién puede compartirse la información. PAP marca la cautela que debe seguirse al usarla o compartirla para el análisis. Ambos valores deben ajustarse a las reglas de la organización, no copiarse sin revisarlas.



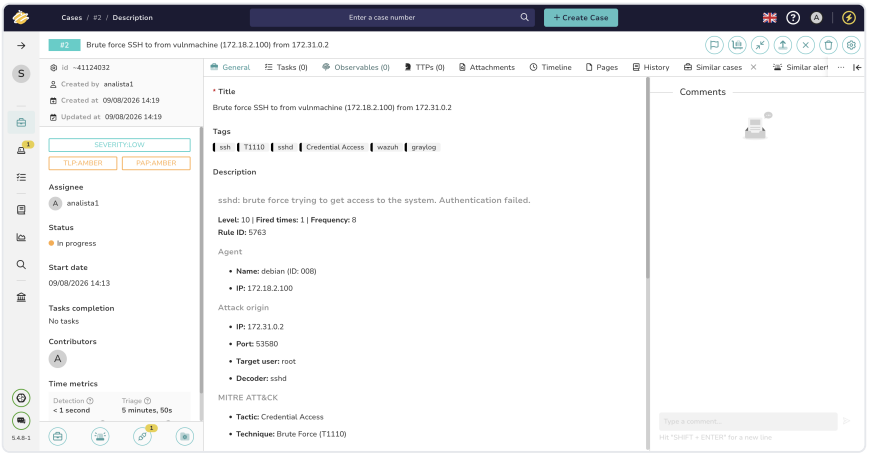
Formulario de creación con gravedad LOW, TLP/PAP AMBER y la etiqueta ssh.

Paso 3. Indicar que el caso está en curso

Tras confirmar, cambiamos el estado de **New** a **In progress** y guardamos. El cambio avisa al resto del equipo de que alguien ya trabaja en el caso.

La conversión importa los observables de la alerta. La vista general conserva el origen, el destino, la técnica MITRE ATT&CK `T1110` y la descripción original, mientras que el caso añade responsable, estado, comentarios y futuras pruebas.

> **Resultado:** El caso queda asignado a analista1, con gravedad LOW y estado In progress.



Caso abierto en TheHive, asignado a analista1 y marcado como In progress.

FASE 2

ANALISTA N1

Análisis N1 y entrega a N2

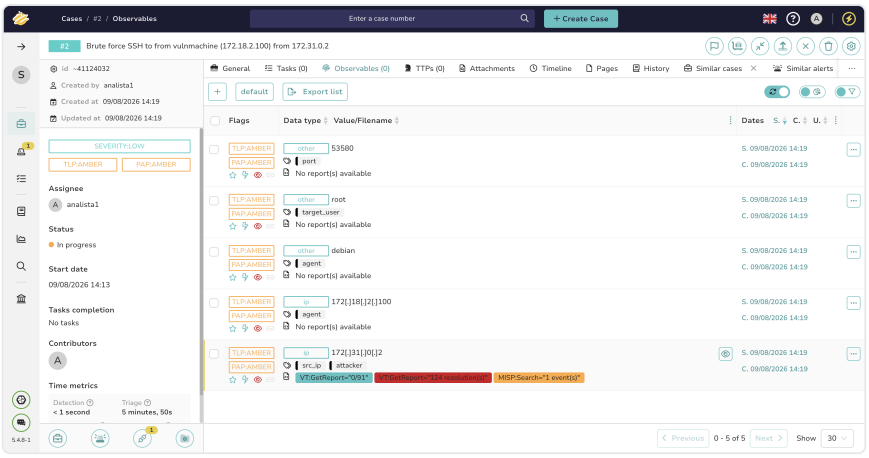
> N1 prepara los observables y obtiene contexto rápido antes de entregar el caso. La pregunta no es solo si una IP aparece en una fuente, sino qué aporta ese dato y qué trabajo sigue pendiente.

Paso 4. Clasificar y enriquecer la IP de origen

Abrimos `172.31.0.2`. Al tratarla como origen del ataque, debe quedar marcada como **IOC**, con **Sighted** para indicar que se vio en la infraestructura, y con la etiqueta `attacker`. Guardamos cada cambio antes de continuar.

Desde el menú del observable ejecutamos los analizadores de Cortex. Los resultados visibles incluyen `VT:GetReport=0/91`, resoluciones históricas y una coincidencia de MISP. La ausencia de detecciones en un motor no demuestra que la IP sea benigna; solo añade contexto al resto de las pruebas.

> **Resultado:** El comentario de entrega indica que Cortex no aportó nada relevante para cambiar la decisión.



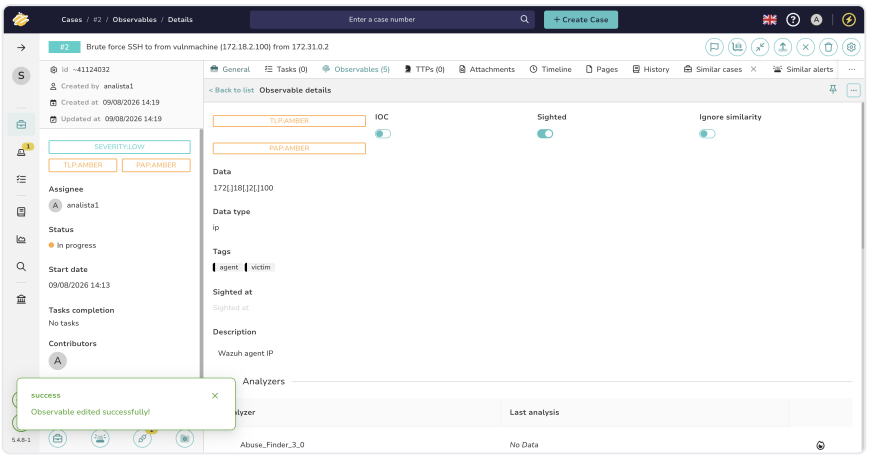
Observables del caso con las etiquetas de resultados que devolvieron los analizadores.

Paso 5. Distinguir la víctima del indicador atacante

Abrimos `172.18.2.100`, la IP del agente afectado, y añadimos la etiqueta `victim`. Esta dirección representa un activo conocido, no el origen de la amenaza. La decisión de marcarla también como IOC depende de la norma interna y del uso que la organización dé a ese campo.

La explicación oral recomienda no tratar esta IP propia como IOC por defecto, pero el control guardado en pantalla no permite confirmar esa recomendación. Debemos seguir la política del SOC y documentar el criterio elegido.

> **Nota:** La captura prueba las etiquetas agent y victim; no debe usarse como prueba de una regla general sobre el campo IOC.



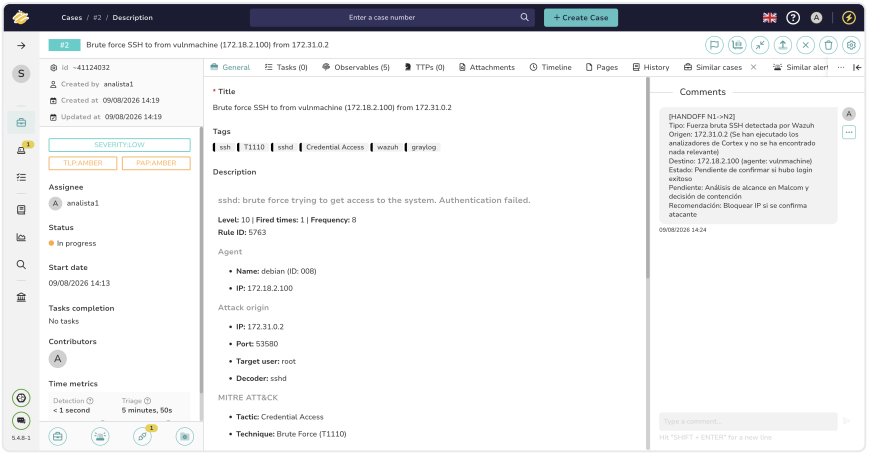
Observable 172.18.2.100 con las etiquetas agent y victim.

Paso 6. Entregar el caso con preguntas concretas

El comentario de entrega permite que N2 continúe sin repetir el triaje. Anotamos el tipo de alerta, el origen `172.31.0.2`, el destino `172.18.2.100`, el agente `vulnmachine` y el resultado de Cortex.

Después separamos lo conocido de lo pendiente: confirmar si hubo un acceso válido, revisar el alcance en Malcolm y decidir la contención. La recomendación de bloquear queda condicionada a confirmar que el origen es atacante.

Evidencia que conviene guardar: Comentario [HANDOFF N1→N2] con origen, destino, trabajo hecho, preguntas pendientes y recomendación.



Comentario de entrega de N1 a N2 con datos, estado y trabajo pendiente.

FASE 3

ANALISTA N2

Comprobación de autenticación

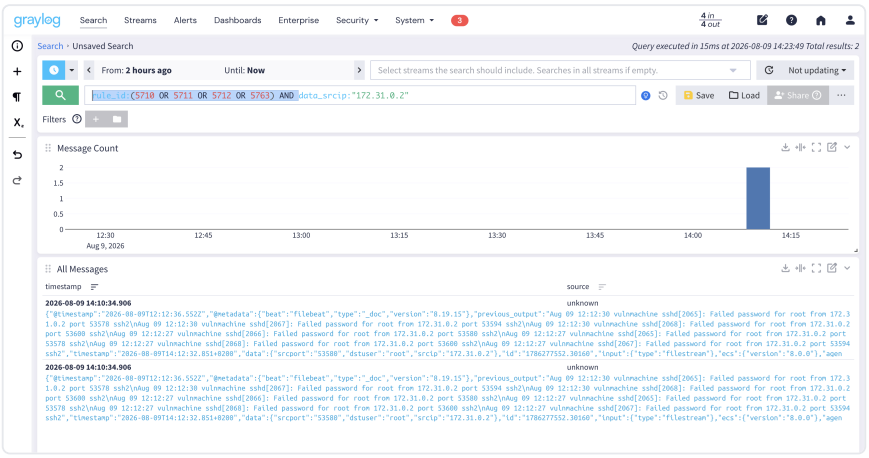
N2 usa Graylog para revisar los eventos que generó Wazuh. Primero busca la actividad SSH del origen y después formula una consulta separada para los accesos válidos.

Paso 7. Buscar los intentos SSH del origen

Elegimos una ventana de **dos horas** para cubrir la detección. La consulta `rule_id:(5710 OR 5711 OR 5712 OR 5763) AND data_srcip:"172.31.0.2"` combina varias reglas SSH del entorno con la IP investigada.

Graylog devuelve **2 resultados**. En los mensajes aparecen varias líneas *Failed password for root*. El total cuenta documentos de Graylog, no intentos únicos de autenticación, porque un documento puede incluir varias líneas previas. La lectura segura es que sí hubo fallos repetidos, no que solo hubo dos intentos.

Resultado: Hay actividad de fuerza bruta y fallos de contraseña asociados al origen investigado.



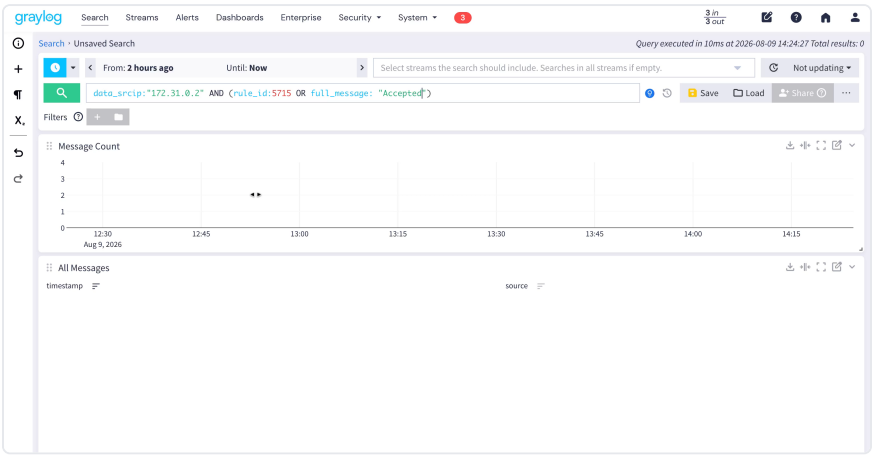
Consulta de reglas SSH en Graylog: dos resultados con mensajes de contraseña fallida.

Paso 8. Separar los accesos válidos de los fallos

Una búsqueda general por origen ayuda a ver otros eventos, pero para decidir el impacto necesitamos buscar de forma expresa una autenticación válida. Usamos `data_srcip:"172.31.0.2" AND (rule_id:5715 OR full_message:"Accepted")`. Los paréntesis hacen que la condición de origen se aplique a cualquiera de los dos indicios de éxito.

El resultado es **0** dentro de las dos horas revisadas. Podemos afirmar que Graylog no muestra un acceso SSH aceptado en esa ventana; no debemos convertir esa ausencia en una prueba universal fuera del periodo o de las fuentes indexadas.

Resultado: No aparece ningún inicio de sesión válido en la ventana analizada.



Búsqueda de accesos aceptados en Graylog con un total de cero resultados.

FASE 4

ANALISTA N2

Análisis del alcance en red

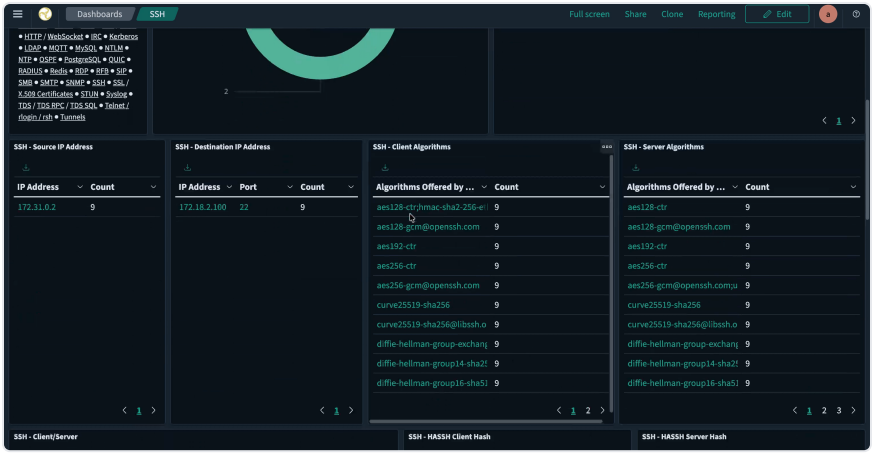
> Los registros indican que no hubo acceso válido. Ahora usamos los datos de red para comprobar el patrón SSH, el destino y si el origen contactó con otros equipos o servicios.

Paso 9. Resumir el patrón SSH con Malcolm

Abrimos **Dashboards** y elegimos el panel **SSH**. El panel agrupa los datos por origen, destino, puerto y algoritmos, lo que permite ver el patrón sin leer sesión por sesión.

La vista grabada muestra `172.31.0.2` como origen, `172.18.2.100` como destino, el puerto `22` y un recuento de **9** en esos paneles. También presenta los algoritmos ofrecidos por cliente y servidor. Este resumen confirma el par de comunicación, pero no prueba por sí solo una autenticación válida.

> **Resultado:** El tráfico SSH observado se concentra entre el origen investigado y vulnmachine por el puerto 22.



Panel SSH de Malcolm con origen 172.31.0.2, destino 172.18.2.100, puerto 22 y recuento 9.

Paso 10. Acotar y ampliar la búsqueda en Arkime

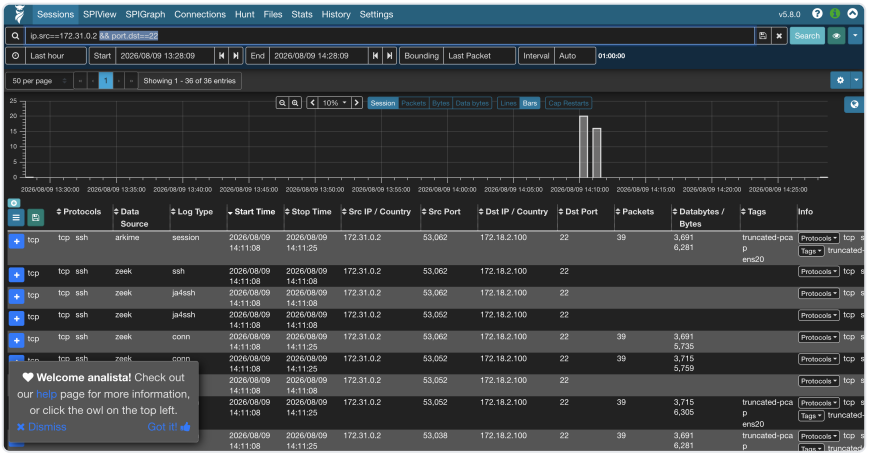
Arkime permite revisar las sesiones. Empezamos con `protocols==ssh`, añadimos `ip.src==172.31.0.2` y, cuando queremos fijar la víctima, sumamos `ip.dst==172.18.2.100`. La búsqueda por protocolo y

origen muestra **36 entradas** en la última hora; las filas visibles apuntan al destino esperado por el puerto 22.

Para comprobar el alcance cambiamos la pregunta. `ip.src==172.31.0.2 && port.dst==22` busca SSH hacia cualquier destino, mientras `ip.src==172.31.0.2` busca cualquier actividad del origen. La grabación propone ampliar además el periodo a 48 horas. Solo debemos concluir que no hubo otros objetivos si la búsqueda ampliada mantiene el mismo destino.

> **Decisión condicional**
>
> **Si aparecen otros destinos, puertos o periodos de actividad:** ampliamos el alcance del caso, revisamos esos activos y revaluamos gravedad y contención.
>
> **Si no:** mantenemos el alcance en vulnmachine y guardamos la consulta como prueba.

Evidencia que conviene guardar: Captura de Arkime con la consulta, el periodo, el total y las columnas de origen, destino y puerto.



Arkime muestra 36 entradas desde 172.31.0.2 hacia el puerto 22 de 172.18.2.100.

FASE 5

ANALISTA N2

Contención y verificación

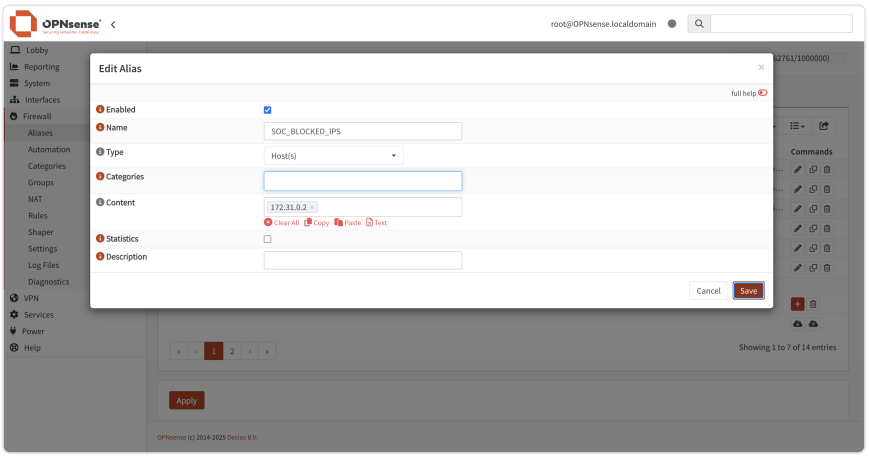
> Aunque no hubo acceso válido, el origen mantiene el patrón de un atacante. La práctica decide bloquearlo y después intenta comprobar el efecto en los registros del cortafuegos.

Paso 11. Añadir la IP al alias de bloqueo

En OPNsense abrimos **Firewall → Aliases**, editamos `SOC_BLOCKED_IPS` y añadimos `172.31.0.2` como host. Guardamos el alias y, si la interfaz muestra cambios pendientes, los aplicamos antes de dar la medida por activa.

La grabación prueba que la IP quedó escrita y que se pulsó **Save**. No muestra el uso de **Apply** ni el estado de la regla que consume el alias. Por ello debemos comprobar ambos puntos en el entorno real.

> **Resultado:** El alias queda configurado con 172.31.0.2; la captura no basta para probar que el cortafuegos ya descarta su tráfico.



Edición del alias SOC_BLOCKED_IPS con 172.31.0.2 antes de guardar.

Paso 12. Verificar el bloqueo o dejar constancia del límite

Abrimos **Firewall → Log Files → Live View** y filtramos por la IP investigada y por la acción de descarte. Si vuelve a generar tráfico, los eventos de bloqueo permiten relacionar la regla con la contención.

Durante la práctica la IP ya no tenía actividad, así que no aparecen descartes que prueben el efecto. No se sustituye esa falta con tráfico ajeno: se anota que la verificación quedó pendiente y se monitoriza por si el origen reaparece.

> **Decisión condicional**
>
> **Si Live View muestra paquetes descartados de 172.31.0.2:** guardamos una captura con hora, interfaz, origen, destino y regla.
>
> **Si no:** documentamos que no había tráfico y mantenemos la vigilancia.

FASE 6

ANALISTA N2

Documentación y cierre

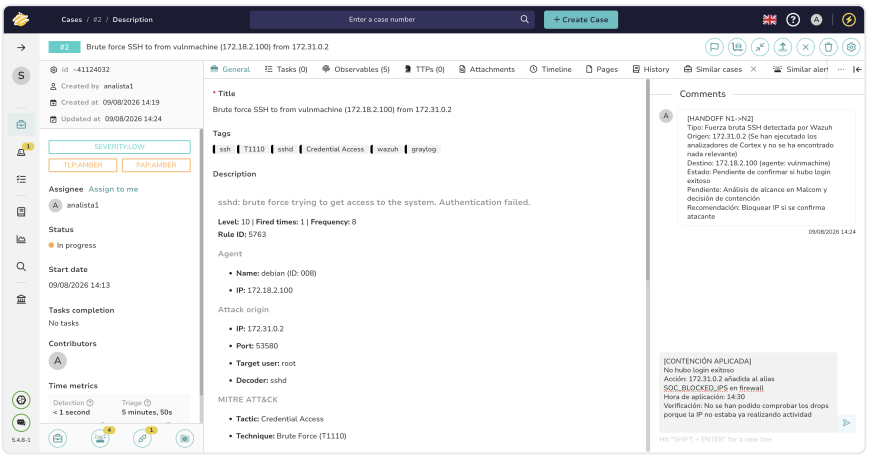
> El cierre debe permitir que otra persona reconstruya la decisión. Volvemos a TheHive para registrar la contención, adjuntar las pruebas disponibles y resolver la alerta relacionada.

Paso 13. Registrar la contención y sus pruebas

Añadimos un comentario con el resultado de autenticación, la acción, el alias, la hora y la verificación. La práctica registra: sin acceso válido; `172.31.0.2` añadida a `SOC_BLOCKED_IPS`; verificación no disponible porque el origen dejó de emitir tráfico.

La pestaña **Attachments** permite adjuntar capturas de Graylog, Malcolm, Arkime u OPNsense. La grabación abre el formulario, pero lo cancela, así que no consta ningún archivo adjunto. En un caso real debemos guardar las capturas que sustentan cada conclusión.

Evidencia que conviene guardar: Consulta de fallos, consulta de accesos válidos, alcance en Malcolm/Arkime, alias del cortafuegos y descartes si llegan a observarse.



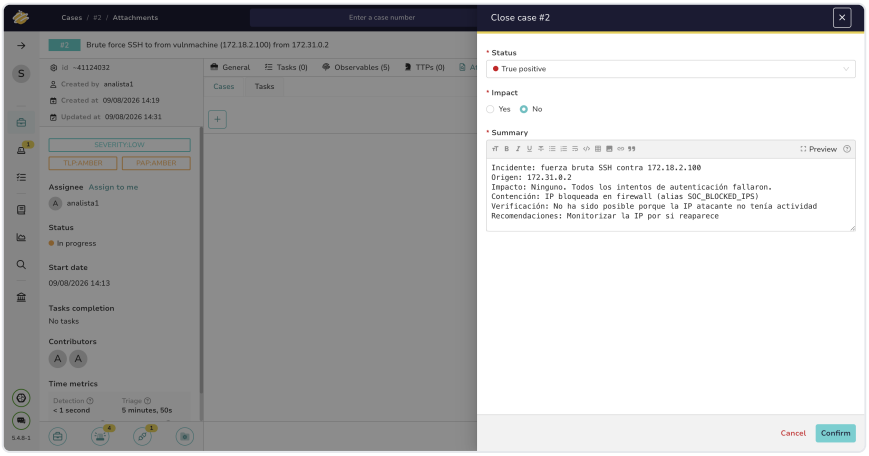
Comentario de contención con la acción tomada y el límite de la verificación.

Paso 14. Cerrar como positivo verdadero sin impacto

Elegimos **True positive** porque la fuerza bruta fue real, y **Impact: No** porque no se encontró una autenticación válida. El resumen debe incluir incidente, origen, impacto, contención, verificación y recomendación.

El texto final indica que todos los intentos observados fallaron, que la IP se añadió al alias de bloqueo, que no pudo comprobarse el descarte por falta de actividad y que debe vigilarse si reaparece. Cerramos solo cuando esas frases coinciden con las pruebas reunidas.

> **Resultado:** Caso cerrado como positivo verdadero, sin impacto y con recomendación de monitorizar el origen.



Formulario de cierre con estado True positive, impacto No y resumen del análisis.

Paso 15. Incorporar la alerta relacionada

Queda una segunda alerta del mismo origen. Puede cerrarse como duplicada o unirse al caso para conservar sus observables y su historia. La práctica usa **Merge alert into case**, busca por el número `2`, selecciona el caso cerrado y confirma.

Tras la unión, la lista de alertas queda a cero. Esta opción evita crear dos expedientes para el mismo incidente y mantiene la información relacionada en un único lugar.

> **Resultado:** La segunda alerta queda unida al caso #2.

Resumen del flujo de trabajo

Fase	Rol	Herramienta	Decisión principal
1	Analista N1	TheHive	Validar la alerta, abrir el caso y dejarlo en curso.
2	Analista N1	TheHive y Cortex	Enriquecer el origen, clasificar el activo y dejar una entrega clara.
3	Analista N2	Graylog	Encontrar fallos de contraseña y comprobar que no aparecen accesos aceptados.
4	Analista N2	Malcolm y Arkime	Confirmar el flujo SSH y acotar su alcance al activo afectado.
5	Analista N2	OPNsense	Añadir el origen al bloqueo y registrar qué parte de la contención puede probarse.
6	Analista N2	TheHive	Registrar pruebas, cerrar sin impacto y unir la alerta relacionada.

Conclusión:

El análisis confirma una fuerza bruta SSH desde `172.31.0.2` contra `172.18.2.100`. Graylog muestra fallos de contraseña y cero accesos aceptados en la ventana revisada; Malcolm y Arkime sitúan el tráfico SSH en el mismo par de equipos y el puerto 22. El caso se cierra como positivo verdadero sin impacto.

La práctica también deja dos lecciones de control de calidad. La clasificación IOC de un activo propio debe seguir una norma clara, y guardar un alias no basta para probar un bloqueo: hay que aplicar los cambios cuando proceda y verificar los descartes. Si no existe tráfico con el que comprobarlos, el expediente debe decirlo.

Guía generada automáticamente usando MENTORA. MENTORA graba las acciones del profesor sobre las herramientas reales del SOC y produce automáticamente este material didáctico, garantizando que la documentación coincide con la práctica actual del aula.

Grabación MENTORA del 9 de agosto de 2026 · Tiempo activo: 18:42 · 15 pasos