Cisco XDR Forensics Knowledge Base

XDR Forensics Knowledge Base

Our mission is to deliver solutions that reduce incident response investigation times through unparalleled forensic-level visibility, automation, speed, and collaboration at scale.

XDR Forensics

XDR Forensics Platform

XDR Forensics is an Automated Investigation and Response platform offering the most comprehensive feature set for remotely acquiring 698 types of digital evidence in just minutes.

Welcome to XDR Forensics's Documentation

This Knowledge Base will guide you through all the features of XDR Forensics.

What is XDR Forensics?

Automated Investigation and Response platform

XDR Forensics is an Automated Investigation and Response platform that delivers deep forensic visibility and end-to-end investigation capabilities at speed.

XDR Forensics combines the rapid remote acquisition of 698 evidence types with intelligent, efficiency-driven automation to drastically reduce investigation time, simplify workflows, and empower SOC and incident responders with accurate, collaborative insights, thereby boosting long-term cyber resilience.

XDR Forensics further accelerates investigations through **DRONE**—a powerful suite of integrated analyzers that automatically assess collected evidence. DRONE's findings across multiple assets are consolidated and visualized in a single pane of glass: the **XDR Forensics Investigation Hub**.

XDR Forensics will perform **simultaneous triage on thousands of assets using YARA, Sigma, and osquery rules**.

XDR Forensics protects employee privacy with targeted collections when required. It also captures the 'forensic state' of multiple assets and presents this information in an Investigation Hub.

The **Investigation Hub** serves as an all-encompassing, user-friendly DFIR intelligence resource. This unifying Investigation Hub consolidates Acquisition and Triage data from all assets, presenting it in an easily digestible format. It also integrates DRONE analyzer findings through intuitive graphical visualizations, thereby identifying the most critical machines that warrant further immediate, focused investigation or remediation. The **Investigation Hub** streamlines the investigative process by:

- Providing actionable findings to prioritize and guide investigators,
- Offering comprehensive listings of all evidential artifacts,
- Including a range of advanced filtering options, and
- Featuring a powerful global search capability.

The XDR Forensics platform integrates with your existing SIEM, SOAR solutions, and many EDR products. This is done via Webhooks and an open API that empowers analysts to automate the response phase of IR.

So, all forensic collections can be scheduled, automated, remote, and scalable.

With evidence hashing, AES256 encryption, and RFC3161 time-stamping, the **Chain of Custody** for evidence handling by XDR Forensics is completely secure.

Other key features include our patent-pending **Baseline Comparison technology**. This allows you to be more proactive and focused in the way you target your efforts. Here, you can compare acquisitions against one another and easily identify additions, changes, and deletions to key system areas often exploited by attackers.

XDR Forensics helps you cut through the noise of security data with live YARA, Sigma, and osquery scanning combined with rapid keyword searching, automated post-acquisition analysis, and Event Scoring.

These features all combine to enable **most digital forensics investigations to be concluded in less than 4 hours** - which is a dramatic improvement over what is commonly achieved today with other solutions.

Terminology

A brief overview of XDR Forensics terminology

Acquisition Profile

A collection of evidence types, application artifacts, network captures, and custom content items grouped into reusable sets known as Acquisition Profiles. While several profiles are provided out of the box, users can also create and customize their own, adding them to the Acquisition Profile Library, accessible via the Main Menu.

XDR Forensics Console

The XDR Forensics Console is a web-based management interface that enables users to efficiently manage assets, assign tasks, and oversee the entire investigation lifecycle. From evidence acquisition to in-depth analysis, report generation, and case management, all activities are streamlined through the Investigation Hub. Users can also personalize their experience by switching between light and dark modes via the main menu, enhancing usability and visual comfort.

Asset and Asset Status

In XDR Forensics, **an asset** is defined as any entity, whether a device or system, physical or virtual, that operates on a supported operating system such as Windows, macOS, Linux, Chrome, IBM AIX, and ESXi. Assets are the foundational elements on which XDR Forensics performs various actions, including evidence collection and task execution, crucial for responding to and hunting cyber threats. Examples of assets include computers, servers, hosts, cloud accounts, and disk images.

In XDR Forensics, an Asset can be in one of 3 states:

1. **Managed**: The asset's responder has been successfully deployed to the device and is ready to collect tasking assignments from the console.
2. **Unmanaged**: An asset is categorized as Unmanaged under two specific conditions:
   - **Discovery without Deployment**: The asset is identified through Active Directory or Cloud Account scans, but does not have the XDR Forensics responder installed.
   - **Unreachable with No Data**: The asset has been disconnected from the XDR Forensics console for over 30 days (Unreachable), and no forensic data from that asset is stored in the XDR Forensics console.
3. **Off-Network:** An asset is classified as Off-Network under two specific scenarios:
   - **Data Supplied**: The asset has previously provided data through methods such as an **Off-Network Acquisition or a Triage task**.
   - **Unreachable with Stored Data**: The asset holds forensic data within the console but is currently inaccessible for further data collection or task assignments.

   For both scenarios, investigation of the existing data is possible, and additional data can be manually imported as required.

The Assets Summary window on the home page can also report the asset as:

1. **Unreachable:** The asset's responder is currently unreachable. If an Asset's Responder fails to connect to the XDR Forensics console for over 30 days, its status changes to "unreachable." Until then, its status will be managed as online or offline.
2. **Update Required:** The responder on the asset requires an update to function correctly.
3. **Update Advised:** The responder is still functional, but for full functionality, an update is recommended.
4. **Isolated:** The asset is currently isolated from the network, except for communication with the XDR Forensics console.

Asset Management - Using Persistent Saved Filters

Persistent Saved Filters enable users to create and store custom asset filters, making it easier to locate and manage assets without having to reapply filter conditions in each session.

Evidence Item or Artifact?

Evidence Item:

In the context of XDR Forensics and cybersecurity generally, an **evidence item** refers to data extracted from various components of a computer operating system and associated system areas crucial for **recording**, **managing**, or **operating** the system. These items often produce digital evidence that can be analyzed to uncover details of user activity and potential security incidents or anomalies.

Artifact:

On the other hand, in XDR Forensics, **artifacts** are files produced by applications during their execution. These files contain valuable information about the activities performed by the application, including **logs**, **configuration files**, **temporary files**, and other artifacts of potential interest for forensic analysis and investigation.

Evidence Repository

A remote location for saving evidence collected as a result of an XDR Forensics tasking. These include:

1. SMB
2. SFTP
3. FTPS
4. Amazon S3
5. Azure Blob
6. Network Shares

To create a New Repository, go to Settings in the Main Menu and select Evidence Repositories from the secondary Menu. From the window 'New Repository' complete the mandatory fields and select the type of repository you wish to add.

Read more details about Evidence Repositories here.

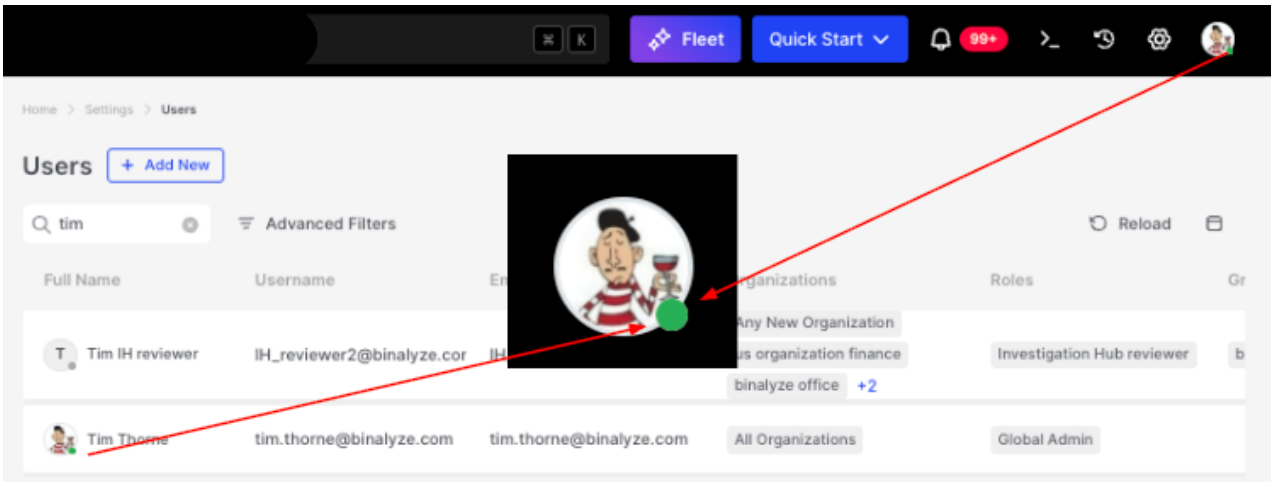
Organizations

In XDR Forensics, an **organization** is a structural entity that allows for the separation of assets, users, and cases within a multi-tenant environment. The multi-tenancy capability of XDR Forensics enables a single console to manage multiple organizations, each with its own isolated environment. Here's how it works:

1. **Asset Management**: An asset (e.g., a device or endpoint) can belong to only one organization, ensuring clear boundaries between different organizational environments. However, within that organization, the same asset can be assigned to multiple cases.
2. **Case Management**: Cases could perhaps also be called 'investigations' or 'incidents', and they are also aligned to a specific organization. Access to cases can be restricted based on user privileges within that organization.
3. **Global and Organization-Specific Settings**: Certain settings, such as policies and evidence repositories, can be configured globally across all organizations or individually for each organization. This flexibility allows administrators to enforce global standards while still providing the ability to customize configurations at the organizational level when required.
4. **Policies and Evidence Repositories**: Policies can be applied either globally or on an organization-by-organization basis. For example, evidence repositories, which store collected data, can be aligned to all organizations (global) or set up uniquely for each organization, allowing for localized data control.

This multi-tenant architecture in XDR Forensics enables organizations to operate independently within the same platform, benefiting from both shared resources and isolated environments, depending on their specific needs.

Real-Time User Online Status Indicators



Terminology: Real-Time User Online Status Indicators

User real-time status is shown throughout the XDR Forensics via colored dots—green for online, gray for offline—helping teams coordinate more effectively across the platform.

Responder

The XDR Forensics **responder** is a compact, cross-platform, zero-dependency, and zero-configuration package that functions as a virtual incident responder, delivering Level 3–4 SOC expertise directly to your assets.

Unlike 'agents' that constantly monitor systems and consume significant resources, XDR Forensics responders only activate to perform precise, user-defined DFIR tasks on demand. This approach enables the deployment of thousands of virtual responders across your IT ecosystem, ready to execute proactive and reactive incident response activities, such as evidence collection, threat hunting, and forensic-level analysis, as needed.

XDR Forensic's approach prioritizes efficient security enhancement, marrying minimal asset impact with maximum readiness and incident response capability.

Read more here: Responder Deployment

Task

Operations are assigned to the assets by the XDR Forensics console, either manually or automatically via a trigger. A task can be assigned to multiple assets, and this is managed through 'task assignments.' Each individual assignment, known as a 'task assignment,' creates a one-to-one correspondence between the task assigned by the console and the specific asset on which the task assignment is executed, ensuring precise management and tracking across all assigned tasks.

Tasks could be either:

1. Manual: Assigned manually by users,
2. Scheduled: Created by users to start at a future date. Scheduled tasks can be either one-time or recurring (daily, weekly, or monthly).
3. Triggered: Assigned to the assets as a response to a trigger request, which is sent by a SIEM/SOAR/EDR solution.

Triggers (Webhooks)

Triggers are the primary extensibility mechanism for XDR Forensics to receive alerts from other security suites, such as SIEM, SOAR, and EDRs.

A trigger is the combination of a parser, an acquisition profile, and a destination for saving the collected evidence (either local or remote).

XDR Forensics takes this to the next level by allowing the trigger to further automate the post-acquisition analysis by leveraging DRONE and MITRE&CK scanners. In effect, the alert from your security tools can launch XDR Forensics into the collection of relevant forensic data, facilitate the analysis of that data, and deliver any DFIR findings into the Intelligence Hub with no analyst intervention required.

Triage

Searching for pieces of evidence such as a file hash, process, or malicious domain at scale. XDR Forensics provides you with 'out-of-box' examples for YARA, Sigma, and osquery, making it fast and easy to start sweeping your environment.

Architecture

A brief overview of system architecture

Components

XDR Forensics is an on-premise or cloud-based, client-server solution that allows you to remotely perform various tasks on assets such as collecting forensic evidence and performing triage with YARA, Sigma, or osquery.

1. Management Console

Management Console is a web-based application that can be viewed from any device with an up-to-date browser.

2. XDR Forensics Responders

Assets are connected to the management console via a lightweight "passive" responder that can be deployed manually or via other mechanisms such as SCCM.

2.1. Passive Responder Explained

XDR Forensics responders;

- **DO NOT** scan anything on the asset that may cause slowdowns (e.g. your Antivirus),
- **DO NOT** block anything on the asset that may cause false positives (e.g. your DLP),
- **DO NOT** create any alerts that may cause "alert fatigue".

XDR Forensics Responder Architecture; overview and performance analysis

What is an XDR Forensics responder?

The XDR Forensics Responder is a compact, cross-platform, zero-dependency, and zero-configuration package that functions as a virtual incident responder, delivering Level 3–4 SOC expertise directly to your assets. It interfaces with the XDR Forensics Console to execute precise, user-defined tasks with minimal resource consumption. This design offers comprehensive investigative coverage without the overhead of continuous monitoring, thereby enhancing both cyber resilience and operational efficiency.

The XDR Forensics responder maintains regular communication with the XDR Forensics Console via what, in its simplest form, is known as HTTP polling, and what we like to call ‘a visit’. The visit interval is typically around 30 seconds for environments with fewer than 1,000 assets. For larger environments, the interval is calculated using the following formula:

```
intervalSeconds = MANAGED_ENDPOINT_COUNT / 100
```

For instance, in a scenario with 5000 assets, the calculated visit interval would be 50 seconds.

The responder sends these visit requests to inform the XDR Forensics console that it is online and ready to receive any task assignments awaiting action.

If the responder does not make a visit at the required interval, it will be shown as offline in the XDR Forensics console.

If the responder does not make a visit for 30 days, it will be marked as unreachable. This status will be resolved immediately once the asset is back online.

If the responder does not collect a task assignment within 30 days of its creation, it will expire and will not be actioned even when the asset reconnects and the responder visits next.

How does the XDR Forensics responder work?

Simply put, when the XDR Forensics responder collects a task assignment from the XDR Forensics console, it carries out the task and provides a report back to the XDR Forensics console upon completion. On the other hand, when the XDR Forensics responder is in an idle state, it periodically (as discussed above) sends visit requests to the XDR Forensics console to check if any new tasks have been assigned to it. During these visit requests, the XDR Forensics responder only checks for task assignments and does not perform any other operations.

The XDR Forensics responder is capable of executing various tasks when assigned by the XDR Forensics Console. These tasks include:

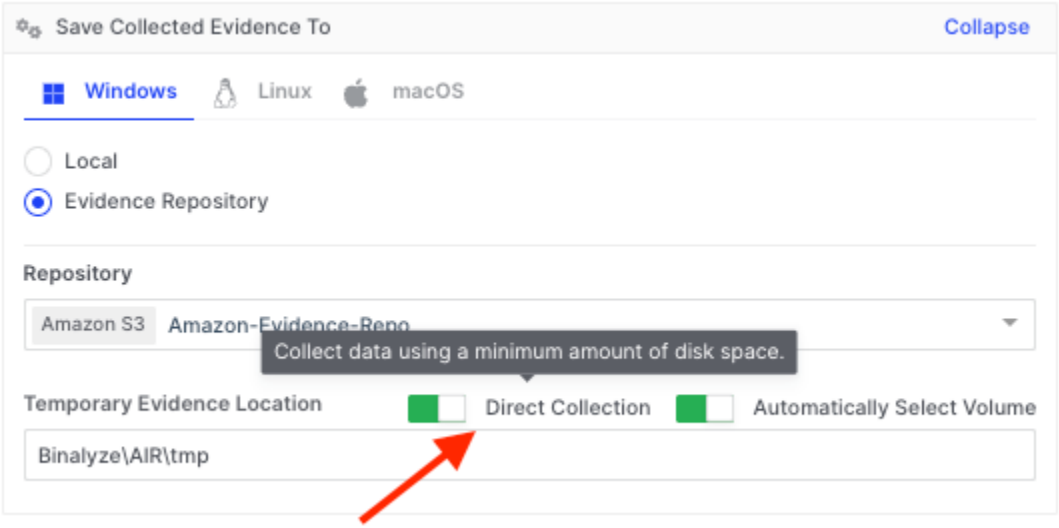
- Acquisition
- Triage scanning (YARA, Sigma, osquery, MITRE ATT&CK)
- Isolation
- interACT sessions
- Auto Tagging
- Disk/Volume Imaging
- Investigation (Timeline)
- Baseline
- Log Retrieval
- Certificate Authority Update
- Migration
- Reboot
- Shutdown
- Update
- Uninstall

Windows Safe Mode: XDR Forensics, like most forensic tools, has limitations in Safe Mode, where minimal drivers—including network access—restrict functionality. We're working on a solution and hope to have a solution in XDR Forensics soon (Q1 2025)

Both the Acquisition and Disk Image tasks support uploading collected evidence to external repositories, including Amazon S3, Azure Blob Storage, FTPS, SFTP, and SMB. These tasks enable the XDR Forensics responder to securely transfer the acquired evidence or disk images to the designated repositories for storage and further analysis.

By utilizing the supported protocols and repositories, the XDR Forensics responder ensures that the collected evidence or disk images are transmitted and stored securely in the designated locations. This enables efficient storage, accessibility, and collaboration, facilitating the management and analysis of acquired data in a secure and scalable manner.

XDR Forensics has an option for Windows, macOS, and Linux XDR Forensics responders to transmit evidential collections directly to external evidence repositories, thereby efficiently minimizing the utilization of local disk space:



XDR Forensics Responder Architecture; overview and performance analysis: Tool Tip explaining Direct Collection for Evidence Repositories

How is the XDR Forensics responder secured?

The XDR Forensics responder maintains robust security by implementing a range of measures, including:

Encrypted Traffic: The traffic between the XDR Forensics responder and the XDR Forensics Console, as well as between the XDR Forensics responder and any evidence repositories, is encrypted using TLS 1.2 and, if available on the server, TLS 1.3. If neither of these two TLS protocols is available, the connection will not be established. This ensures that data in transit is protected against interception and unauthorized access.

Communication: The XDR Forensics Console does not initiate the sending of task assignments to the XDR Forensics responder; rather, it is the XDR Forensics responder that initiates the interaction by asking the XDR Forensics Console if it has any tasking assignments ready for it to run. This approach significantly reduces the risk of various security attacks, as it controls the communication flow and reduces the XDR Forensics responder's exposure to external threats.

Privileged Account Usage: On macOS and Linux, the XDR Forensics responder uses the root account, while on Windows, it uses the system account. This level of access control makes it difficult for other users to tamper with the application, thereby enhancing its security.

Regular Internal Penetration Testing: Before every release, our internal penetration testing team conducts thorough penetration testing. This proactive approach helps identify and mitigate potential vulnerabilities.

Secure Libraries and Third-Party Applications: We consistently use updated and vulnerability-free libraries and third-party applications. This precaution in maintaining up-to-date software components protects against known security vulnerabilities.

Supply Chain Attack Prevention: Measures are in place to protect against supply chain attacks, and our DevOps team continuously improves these. This is crucial to prevent threats that could compromise the software development and deployment process.

Continuous Source Code Scanning: The source code is regularly scanned by security tools to identify potential vulnerabilities. This constant monitoring enables the quick identification and resolution of any security issues that arise in the codebase.

Digital Signing: The use of digital signatures adds a layer of security, ensuring the authenticity and integrity of our software. This helps prevent tampering and verifies that the software has not been altered after it was signed.

Blackbox Analysis: The binary undergoes Blackbox analysis, a method of testing the software's external functioning without delving into its internal structure. This type of analysis has been performed on the XDR Forensics responder. It helps in identifying security vulnerabilities from an outsider's perspective, providing a critical view of the system's external defenses.

Graybox Analysis: For the XDR Forensics responder project, Graybox analysis has been conducted. This testing method combines both the internal and external examination of the software, providing a more comprehensive security overview.

Are databases used by the XDR Forensics responder?

Functioning like a server application, the XDR Forensics responder does not use databases. Instead, it operates by saving reports as individual files using SQLite. These reports are subsequently forwarded to the XDR Forensics Console. This approach simplifies the data handling process, enabling the efficient and secure storage and transfer of information.

What is the Cisco XDR Forensics design process?

We continuously advance our development process by implementing the SCRUM methodology, complemented by unit and integration testing. The use of both unit and integration testing is crucial for maintaining high-quality standards and ensuring that each component of our product functions seamlessly individually and as part of the whole system.

Resource monitoring for the XDR Forensics responder

After the initial installation, it is normal to observe a small amount of memory being allocated, typically around 30MB to 40MB, with no significant CPU or disk usage during idle states. This behavior is expected and can be attributed to the necessary resources required for the XDR Forensics responder to function correctly.

During idle states, the XDR Forensics responder remains in standby mode, pending its next call to the Console to collect any new tasking assignments. The allocated memory is utilized to maintain the XDR Forensics responder's core functionality and to ensure prompt responsiveness when new tasks are assigned.

XDR Forensics Responder Architecture; overview and performance analysis: Responder standby mode

When the XDR Forensics responder receives an acquisition task, the evidence collection process is carried out by a subprocess called Tactical (or Incident Response Evidence Collector on Windows). During the acquisition process, it is normal to observe increased CPU and memory usage as the Tactical sub-process actively collects and processes the evidence.

The increase in CPU and memory usage is a result of the intensive data gathering and analysis performed by the Tactical sub-process. It utilizes system resources to capture and process the required evidence efficiently, ensuring the integrity and completeness of the collected data.

The extent of CPU and memory usage during the acquisition task may vary depending on factors such as the size and complexity of the evidence being collected. Once the acquisition is completed, the CPU and memory usage will typically return to normal levels, reflecting the completion of the resource-intensive task.

If you prefer to limit CPU usage during acquisition or triage tasks, you have the option to set a CPU policy that restricts maximum CPU usage to a specified percentage. This setting, adjustable in the XDR Forensics Console before execution, allows you to limit CPU usage. Setting a lower percentage may extend task completion times.

XDR Forensics Responder Architecture; overview and performance analysis: The idle state of the XDR Forensics responder after the acquisition is complete.

A **Triage task** does not involve running the Tactical sub-process for evidence collection. Instead, the Triage task is executed within the XDR Forensics responder, utilizing its internal capabilities to analyze and evaluate the collected data.

Although the CPU usage for a Triage task is typically low, it is still possible to set a CPU policy for the task.

XDR Forensics Responder Architecture; overview and performance analysis: CPU and Memory Usage during a Triage Task

The log file of the running XDR Forensics responder provides valuable information about CPU usage, memory usage, and other system resources. Here is an example of the log entries about system and service resources:

1. `INFO 2024-01-04 18:45:25+03:00 2.31.2 triage: resmon: SysStats{GoHeapAlloc: 2.3 MB, GoHeapSys: 12 MB, NumGoroutines: 27, NumCPU: 16} file:pkg/resmon/handlers.go:16 func:resmon.(*LoggingStatsHandler).HandleSysStats`
2. `INFO 2024-01-04 18:45:26+03:00 2.31.2 triage: resmon: PidStats{PID: 9460, Name: AIR.exe, CPU: 14.7%, AvgCPU: 25.9%, Mem: 56 MB, NumFDs: 341, NumCPU: 16} file:pkg/resmon/handlers.go:21 func:resmon.(*LoggingStatsHandler).HandlePidStats`

The log file for the XDR Forensics responder can be found at the following location:

`C:\Program Files\Cisco\Forensics\AIR\AIR.log.txt`

You can navigate this path on your system to access the log file and view the relevant information about CPU usage, memory usage, and other resources as logged by the XDR Forensics responder during its operation.

> **These usages were observed on a system equipped with an Intel Core i7-10875H running at 2.30GHz (with 16 processors) and 32GB of memory (Windows 10 Pro).**

Similar scenarios can be observed on macOS with the built-in Activity Monitor application. To access detailed process information, simply click on the (i) button within the Activity Monitor.

XDR Forensics Responder Architecture; overview and performance analysis: Activity Monitor (filtered air)

On Linux, an alternative option for resource monitoring is to use `htop` instead of the built-in app top. The `htop` option offers enhanced capabilities and can be installed by following these steps:

1. Open the terminal.
2. Run the command: sudo apt-get install `htop` (for Ubuntu/Debian-based distributions) or sudo yum install `htop` (for CentOS/Fedora-based distributions).
3. Once installed, type `htop` in the terminal and press Enter to launch the application.

Using `htop` provides a more comprehensive and user-friendly interface for monitoring system resources on Linux.

XDR Forensics Responder Architecture; overview and performance analysis: htop (filtered for XDR Forensics)

Resource Monitoring with `resmon`

There is also a CLI tool named `resmon` specifically developed for internal usage. It can be used to gather resource usage data related to the XDR Forensics responder and its subprocesses, storing them in a local database.

By default, `resmon` will monitor the XDR Forensics responder if no flags are given. However, you can monitor other processes by providing a PID flag or a process name flag. For more detailed information on its usage, a usage document for `resmon` can be provided upon request.

The information collected by `resmon` is stored in a local database, which includes numerous entries for the monitored process and its subprocesses. Due to the abundance of entries with comprehensive details, reading and interpreting the data can be a challenging task.

To address this, a script has been developed alongside `resmon` to visualize these outputs. It displays the CPU and memory usage of the processes (including subprocesses) monitored by `resmon` in a graphical format.

In the following section, we will share the `resmon` results as it monitored various task assignments being executed by the XDR Forensics responder. Throughout the tasks, `resmon` will continuously monitor the XDR Forensics responder and its subprocesses, generating a comprehensive local database that captures the output of resource monitoring.

For easy visualization, we will utilize a feature of a `resmon` designed to focus on visualizing its output by presenting the CPU and memory usage in intuitive graphical representations. These visualizations offer valuable insights into the resource utilization of the XDR Forensics responder and its subprocesses throughout each tasking assignment, from start to finish.

The following scenarios were observed on a system equipped with an Intel Core i7-10875H running at 2.30GHz (with 16 processors) and 32GB of memory (Windows 10 Pro).

Analysis of an Acquisition Task

Below, you will find two graphs illustrating the CPU and Memory usage of the XDR Forensics responder. These graphs represent the resource utilization from the moment an acquisition task is started through to its completion.

XDR Forensics Responder Architecture; overview and performance analysis: Memory Usage on an Acquisition Task

XDR Forensics Responder Architecture; overview and performance analysis: CPU Usage on an Acquisition Task

Duration	Report Size (Zipped)	Database Size	Event Record Count	Drone	Total Disk Space	Use Spa
06m29s	199KB	38KB	10091	Enabled	512 GB	176

Analysis of an Acquisition Task (with CPU limit of 50%)

In this scenario, we will examine the CPU and Memory usage of the XDR Forensics responder while running tasks received from the XDR Forensics Console, with a specific condition: the CPU usage of the XDR Forensics responder is limited to 50%.

This limitation is possible due to the XDR Forensics responder's capability to control and restrict its CPU utilization during task execution.

The visualized graphs provided below depict the resource utilization, explicitly focusing on the CPU and Memory usage of the XDR Forensics responder. These graphs illustrate the performance of the XDR Forensics responder, highlighting its ability to effectively manage CPU allocation while executing tasks received from the XDR Forensics Console.

The script can occasionally display temporary CPU usage spikes that surpass the process's CPU limit as a result of aggregating subprocesses.

XDR Forensics Responder Architecture; overview and performance analysis:Memory Usage on an Acquisition Task

XDR Forensics Responder Architecture; overview and performance analysis: CPU Usage on an Acquisition Task

Duration	Report Size (Zipped)	Database Size	Event Record Count	Drone	Total Disk Space	Use Spa
06m48s	200KB	39KB	10102	Enabled	512 GB	176

Analysis of a Triage Task

Let's examine the resource usage of the XDR Forensics responder when a Triage task is received from the XDR Forensics Console.

XDR Forensics Responder Architecture; overview and performance analysis: CPU Usage on a Triage Task

XDR Forensics Responder Architecture; overview and performance analysis: Memory Usage on a Triage Task

Duration	Triage Rule Type	Total Disk Space	Used Disk Space	CPU Limit
19m33s	YARA	512 GB	176 GB	100%

Analysis of a Triage Task (with CPU limit of 50%)

Similar to an acquisition task, a Triage task can also be configured with a CPU limit for executing the XDR Forensics responder. The following graphs illustrate the resource usage of a Triage task running with a CPU limit of 50%.

XDR Forensics Responder Architecture; overview and performance analysis: CPU Usage on a Triage Task

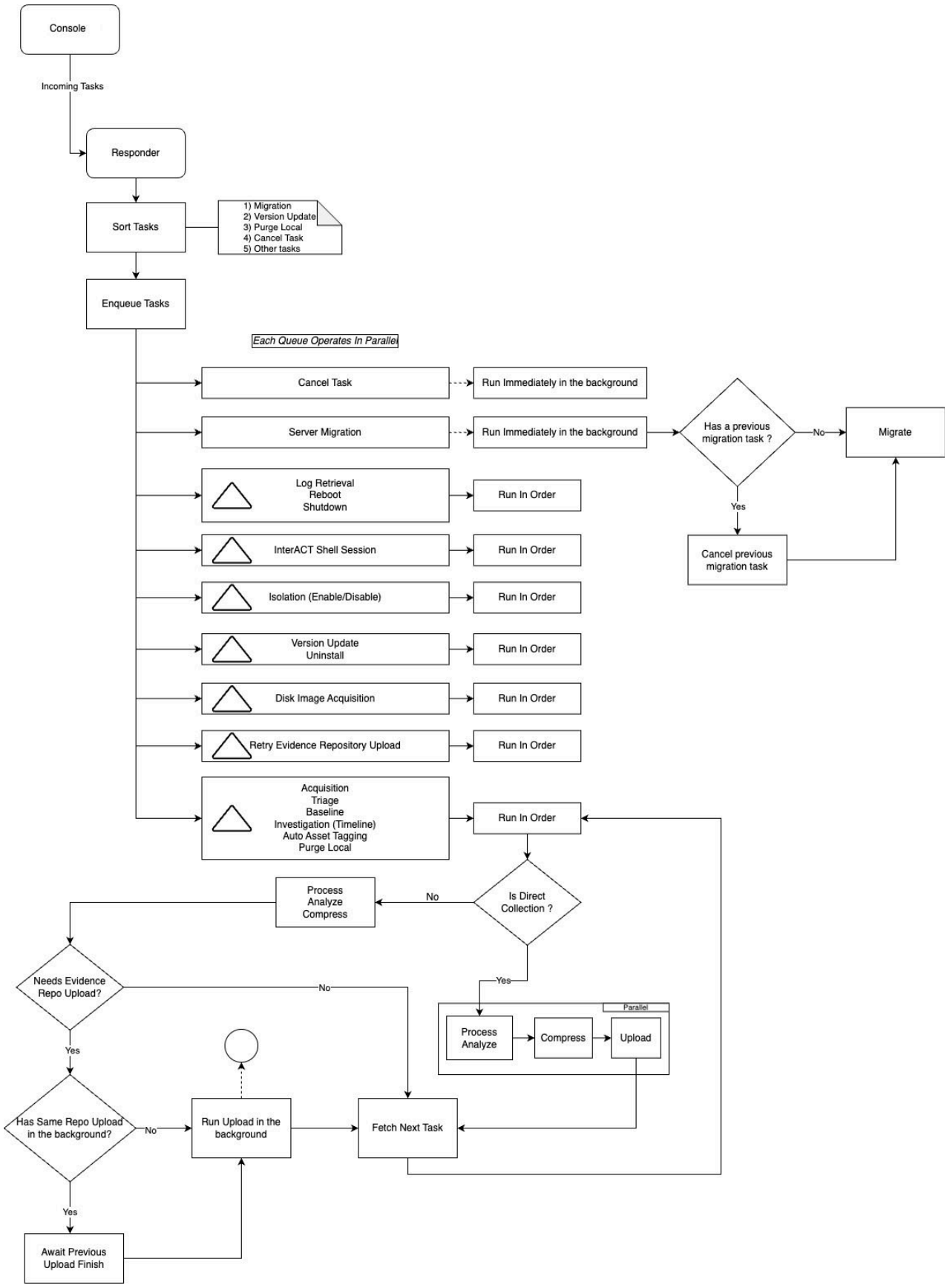
XDR Forensics Responder Architecture; overview and performance analysis: Memory Usage on a Triage Task

Duration	Triage Rule Type	Total Disk Space	Used Disk Space	CPU Limit
27m09s	YARA	512 GB	176 GB	50%

XDR Forensics Task Flow and Management

Introduction

In today's dynamic digital environment, managing tasks efficiently within a software system is crucial for reliability, flexibility, and optimal performance. This guide explores a sophisticated task management system designed to handle a wide range of operational scenarios, focusing on task retrieval, execution, prioritization, and system resilience against failures and network disruptions.



Architecture: XDR Forensics Task Flow and Management

Task Retrieval and Execution

The Role of the XDR Forensics Console

The XDR Forensics platform features an intuitive web-based console designed to orchestrate and dispatch tasks to designated remote XDR Forensics responders effectively. Serving as the nerve center for task allocation, this console guarantees that each task is accurately assigned for execution, optimizing operational efficiency. Within this ecosystem, assigning a specific task to a particular asset is termed a 'task assignment,' ensuring a clear, one-to-one correspondence between tasks and assets for precise management and tracking.

Mechanisms for Task Checking

To accommodate diverse operational needs and customer network policies, the system employs two primary mechanisms for task checking:

1. Regular Interval Checks: Tasks are checked at predefined intervals, which can be dynamically adjusted based on the system's current configuration and operational demands.
2. The NATS Protocol: For immediate task fetching or near real-time communications with assets, the system incorporates a specialized protocol named "NATS." This protocol is designed to bypass the standard checking intervals, allowing for urgent tasks to be retrieved and executed with minimal delay

> XDR Forensics does not use TLS for NATS traffic because no sensitive data is transmitted over it. NATS is only used to send a lightweight “ping” message when the XDR Forensics Console assigns a task to a Responder. After receiving this signal, the Responder immediately connects back to the XDR Forensics Console over HTTPS (port 443) to securely download the full task details

Task Checking Intervals

Task-checking intervals are not static; they vary dynamically from seconds to hours, influenced by the system's configuration. This flexibility ensures the system can adapt to changing workloads and priorities efficiently.

Task Prioritization and Execution Order

Prioritization of Critical Tasks

Certain tasks, such as "cancel tasks," receive priority in the execution queue. This prioritization is crucial to prevent delays in the cancellation process, ensuring tasks are halted promptly when required.

Execution Order and FIFO Queue Model

The system employs a first-in, first-out (FIFO) queue model for task execution. This model ensures that tasks are processed in the order received, with special consideration given to tasks that might block or delay subsequent operations unnecessarily.

Handling of Failed Tasks and Network Disruptions

Tasking Assignment interruptions

If a Tasking Assignment has been collected by the Responder but is interrupted before the completion of collection, triage, or analysis, the task will not resume where it left off. Instead, this interruption will result in a task failure. Such failures are automatically recorded within the console's tasking details.

When this occurs, the status of the task in the XDR Forensics console will reflect the failure, and it will be necessary to manually restart or initiate a new task to ensure that the intended data collection and analysis are completed. This approach ensures clarity and accuracy in managing task assignments, even in cases of unexpected interruptions.

Retry Mechanisms for File Uploads

For tasks that require file uploads, such as uploading to an evidence repository, the system includes built-in retry mechanisms. These mechanisms are activated to re-attempt uploads if network issues interrupt the process. The number of retries and the specific procedures for handling these retries vary depending on the task type and the destination of the file.

Additionally, if "direct collection" is enabled for an acquisition task and a failure occurs, the user must restart the acquisition process from the beginning. This ensures that all necessary data is properly collected without partial or corrupt files.

Data Purging and Task Cancellation

A specialized "purge local" task type exists for efficiently cleaning up local data related to completed or failed tasks. This function is crucial for maintaining optimal disk space utilization and efficient system resource allocation.

System Flexibility and Customer Policies

This guide emphasizes the importance of a flexible system that can adapt to diverse customer policies, including specific network configurations and security requirements. The choice of protocols and mechanisms for task management is influenced by these diverse operational needs.

Documentation and System Improvements

Continuous improvement is a cornerstone of system development. The commitment to updating documentation reflects ongoing efforts to refine task management processes and system functionalities based on operational insights and technical advancements.

Technical Implementation Details

The guide offers an in-depth examination of the system's technical underpinnings, including the utilization of the "NATS" protocol, dynamic adjustment of task-checking intervals, and the logic behind task prioritization and queue management. These details offer a comprehensive understanding of the system's operational logic and its capability to handle various scenarios efficiently.

Conclusion

Efficient task management is pivotal in ensuring the reliability and performance of software systems. Through innovative mechanisms, such as the air console and NATS protocol, alongside dynamic task-checking intervals and a robust FIFO queue model, the system outlined in this guide represents a state-of-the-art solution for managing tasks in complex software environments. The emphasis on flexibility, resilience, and continuous improvement underscores the system's readiness to meet the evolving demands of modern digital operations.

Network Communication

How do assets communicate with the console?

How Do Assets Communicate with the Console?

All routine communication between assets and the XDR Forensics console is **initiated by the assets**—they do not receive incoming requests from external sources. Communication occurs through various protocols and channels:

Primary Communication Channels

- **HTTPS (TCP 443)** – The main communication channel from assets to the console (e.g., `<tenantname>.cisco-<region>.binalyze.io`).
- **WebSocket over HTTPS (TCP 443)** – Used for **interACT** features.
- **NATS (TCP 4222) (Optional)** – Supports real-time task pushes to assets. If this port is unavailable, XDR Forensics defaults to HTTP(S) polling for task retrieval.
- **DNS (UDP/TCP 53)** – Required for name resolution services.

External Communication

- **HTTPS to `<tenantname>.cisco-<region>.binalyze.io`** – Used for **responder updates** and installation packages. If the CDN is unavailable, the XDR Forensics console acts as a fallback source.

Evidence Repository Communication (When Configured)

- **Cloud Storage:** HTTPS communication to services like **Amazon S3 and Azure**.
- **Traditional Storage:** Supported via **SFTP, FTPS, or SMB**.

Proxy Support

If a **proxy** is configured in your environment, assets can communicate using:

- **HTTP**
- **HTTPS**
- **SOCKS5**

Cloud Forensics

Investigators and analysts can use the XDR Forensics platform to conduct investigations on machines located in **cloud platforms**. Our platform supports cloud-based virtual machines, as well as on-premise and off-network devices. Investigators and analysts can install responders on virtual machines located on the cloud infrastructure for investigations and analysis. **Amazon Web Services and Microsoft Azure are both supported.**

Cloud Forensics: Tornado

We understand the unique challenges of investigating cloud-based attacks, such as **Business Email Compromise (BEC)**. That's why we have introduced the Tornado preview version, a standalone desktop application designed to simplify evidence collection from Google Workspace and Microsoft Office 365. **Learn all about Tornado here ↗.**

Investigators and analysts can easily and quickly deploy responders to their cloud assets and immediately initiate investigations, compromise assessments, and threat-hunting activities. By leveraging the automation advantages of cloud platforms, users can easily deploy multiple responders using a single authorized cloud platform account.

After adding the authorized account to the Console, it enumerates the cloud platform to discover and list assets. Then, investigators and analysts can deploy responders to individual or multiple cloud assets with one click.

Add Authorized User

Since different cloud platforms utilize distinct identity and access management infrastructures and employ different working mechanisms, their requirements may vary; however, ultimately, all we need is an authorized account with list and control permissions on cloud assets.

Investigators and analysts can add a cloud account to the Console by using the Assets page:

1. From the Main menu, select Assets
2. Click Add New and click Cloud Account
3. Then click the Add Account button according to the cloud platform you want to add on the appearing Cloud Platforms window

The configurations that need to be performed according to the cloud platforms are listed below.

Amazon Web Services Compute EC2

Either of the two methods mentioned above will redirect investigators and analysts to similar pages, allowing them to enter their account details. They can either enter their existing account details, which are given below, or use the cloud formation link provided by XDR Forensics to create a new account with enough permissions.

Account Name: Optional field.

Access key ID: Mandatory field and it must be filled with the value provided by AWS

Secret access key: Mandatory field and it must be filled with the value provided by AWS

Organization: Mandatory field and it must be selected from the Organization created on the XDR Forensics console. Every cloud account can be assigned to only one organization.

Cloud account needs the following permissions to deploy virtual machines XDR Forensics responder.

arn:aws:iam::aws:policy/AmazonEC2ReadOnlyAccess
arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy
arn:aws:iam::aws:policy/AmazonSSMManagedInstanceCore
arn:aws:iam::aws:policy/AmazonSSMDirectoryServiceAccess
arn:aws:iam::aws:policy/service-role/AmazonSSMMaintenanceWindowRole
arn:aws:iam::aws:policy/AmazonSSMReadOnlyAccess ssm:CancelCommand

The creation of an AWS Account with sufficient permissions flow is explained below.

1. Click on the URL and Create an Account
2. Open AWS Console → IAM → Users
3. Select the User → Security Credentials → Create Access Key
4. Fill out the Account Details Form

Microsoft Azure Virtual Machines

Either two different ways mentioned above will redirect investigators and analysts to similar pages allowing them to enter account details. They can either enter their existing account details, which are given below or create a new account with enough permissions.

Account Name: Optional field.

Application (client) ID: Mandatory field and it must be filled with the value provided by Azure

Subscription ID: Mandatory field and it must be filled with the value provided by Azure

Tenant ID: Mandatory field and it must be filled with the value provided by Azure

Key (Client Secret): Mandatory field and it must be filled with the value provided by Azure

Organization: Mandatory field and it must be selected from the Organization created on the XDR Forensics console. Every cloud account can be assigned to only one organization.

Cloud accounts need the following permissions to deploy the virtual machine responder.

Reader VirtualMachine Contributor

The creation of an Azure Account with sufficient permissions flow is explained below.

1. Azure portal → App Registrations → New Registration
2. Assign required roles to the new app registration for the subscription
3. App Registrations → Open the created App Registration
4. Certificates & Secrets → Create a new client secret
5. Fill out the Account Details Form

Google Cloud Platform

Coming soon

Synchronization and Enumeration

The Console immediately starts to enumerate the cloud platform and retrieves the assets list and asset details after the cloud account is added. It discovers the assets depending on the permissions and authorizations of the cloud accounts. All discovered assets will be shown under the Amazon AWS category under the associated organization.

The assets and their details are shown on the right side of the Secondary Menu in the Assets page as a list. Assets in which the XDR Forensics responder is deployed are shown in blue, and the assets in which the XDR Forensics responder is not deployed are shown in grey in that list.

> If investigators or analysts do not sync the cloud account manually, XDR Forensics Console automatically syncs in 30 minutes and updates the asset list.

Responder Deployment

All deployment actions are considered tasks by the XDR Forensics Console and listed under the Tasks as responder Deployment tasks. Therefore, all responder **deployment actions** and their **status** can be seen on the **Tasks list**.

The primary advantage of responder deployment in a cloud platform is **automation**. Analysts and investigators don't need to choose the operating systems and their versions. They only assign deployment tasks to the associated devices, and all deployment processes are performed quicker and easier automatically.

Investigators and analysts can deploy responders to cloud assets by using the Endpoint page:

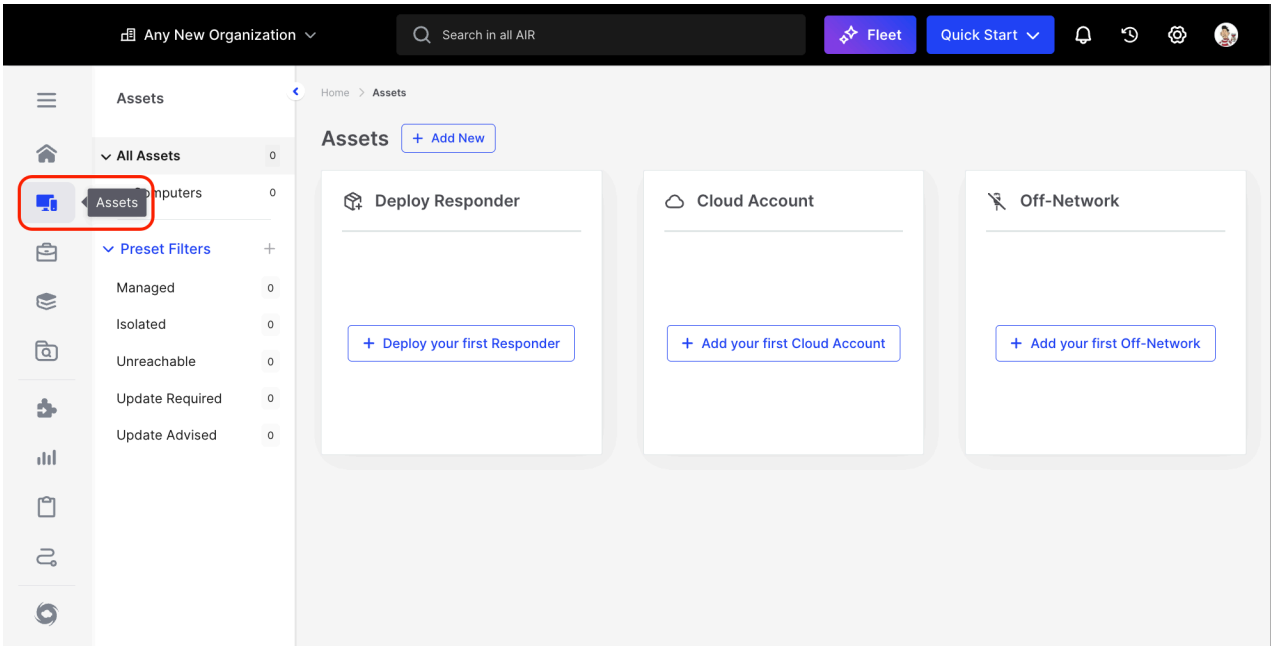
1. From Assets in the Main Menu: All cloud assets are listed here in the Secondary Menu. Investigators and analysts can search, filter and see the details of the assets on this page.
2. Investigators and analysts can deploy the responders individually, with multiple selections or all of them with one click.
    a. Individual deploy: Click the assets and then click the Deploy button
    b. Multiple selections: Select the assets in the list by clicking a checkbox at the beginning of the asset line. Then Actions button immediately appears at the top of the page. Click Deploy Responder the under the Actions menu.
    c. Deploy to All Assets: Click the three-dot on the right side of the Amazon AWS or Tag, which includes associated cloud assets. Then click Deploy Responder

Setup

XDR Forensics setup instructions

Responder

Installation of the XDR Forensics Responder on your assets is managed via the Assets button in the Main Menu:



Responder Deployment: Assets button in the Main Menu

The XDR Forensics Responder installer is a zero-configuration package that contains the console address already embedded in it.

You can deploy the XDR Forensics Responder in multiple ways:

1. Downloading an installation package (Windows, macOS, Linux, Chrome, and ESXi)
2. Copying a PowerShell Command (Windows)
3. Copying a CURL Command (macOS and Linux)
4. Copying a WGET Command (macOS and Linux)
5. Downloading a PowerShell Script (Windows)
6. Downloading the Asset installer (macOS and Linux)
7. Manual installation via Active Directory/SCCM.
8. Generation of a shareable Deployment Link (Windows, macOS, Linux, Chrome, and ESXi)

The cards below show the default location paths for the XDR Forensics Responder:

Windows

C:\Program Files\Cisco\Forensics\AIR

macOS

/opt/cisco/forensics/air

Linux

//opt/cisco/forensics/air

In the sections that follow, we will look at the deployment of XDR Forensics Responders to Windows, Linux, and Mac operating systems.

The XDR Forensics Responder is a 'zero-config' deployment, as the file name has all the information you need for quickly deploying a Responder.

This level of detail in the filename provides all the information needed as a digitally signed binary - this prevents issues with security solutions, and to date, not one issue has arisen.

The file name example shown here has 4 main components:

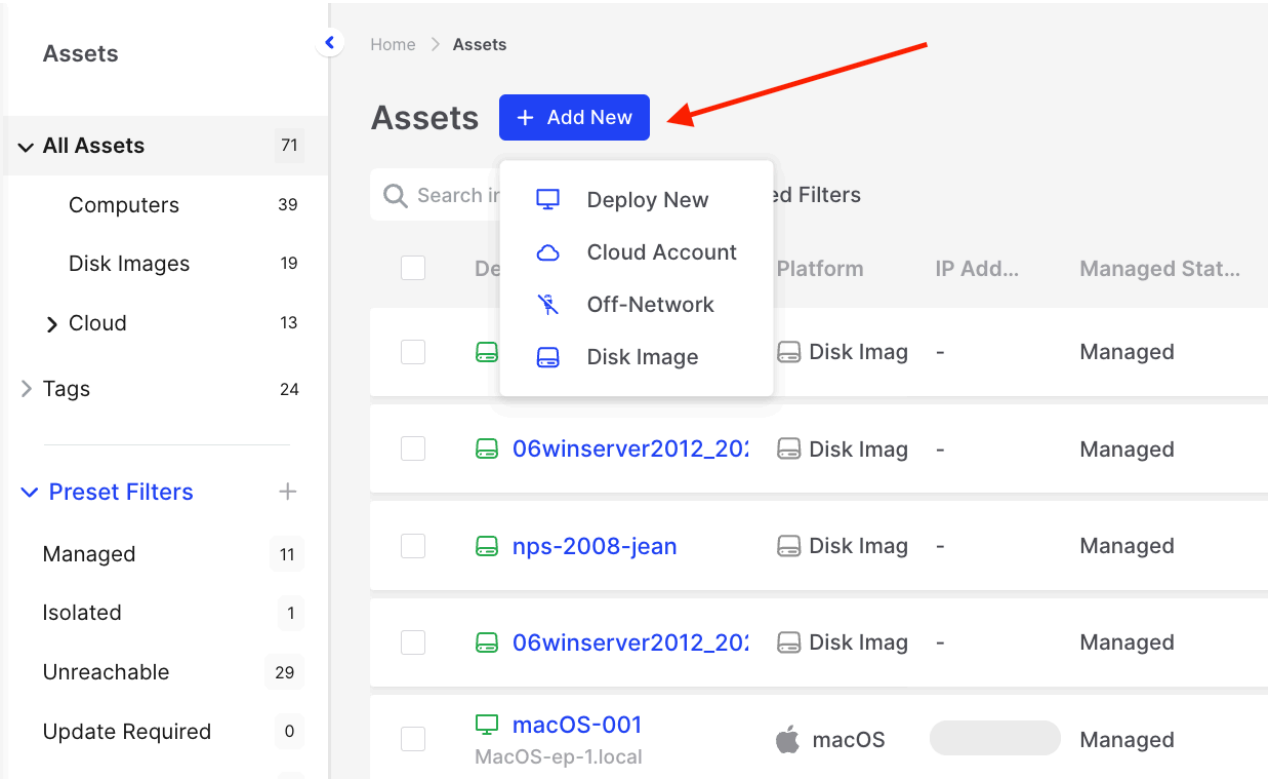
```
AIR.Responder_2.38.7_air-demo.ACME.com_176_9df51c56a73341f4_386_.msi
```

1. **2.38.7** - is the Responder version number.
2. **air-demo.ACME.com** - is the address of the console with which the Responder will be communicating
3. **176** - is the console's internal organization number ID.
4. And the apparently random mixture of letters and numbers, **9df51c56a73341f4**, is the - Deployment Token.
5. **386** - describes the processor architecture of the machine on which the Responder will run.

There are multiple ways of deploying the responder, all of which are designed to be quick and scalable. Let's take a look at the different ways in which you can deploy the XDR Forensics Responder to your assets:

From the Main Menu, select 'Assets' and then 'All Assets' from the Secondary Menu. Now you will see the page name 'Assets' and next to that is the **Action Button,** which for the Assets page is labeled **'+ Add New.'**

When this **'+ Add New'** button is selected, three deployment options are offered in a drop-down menu:



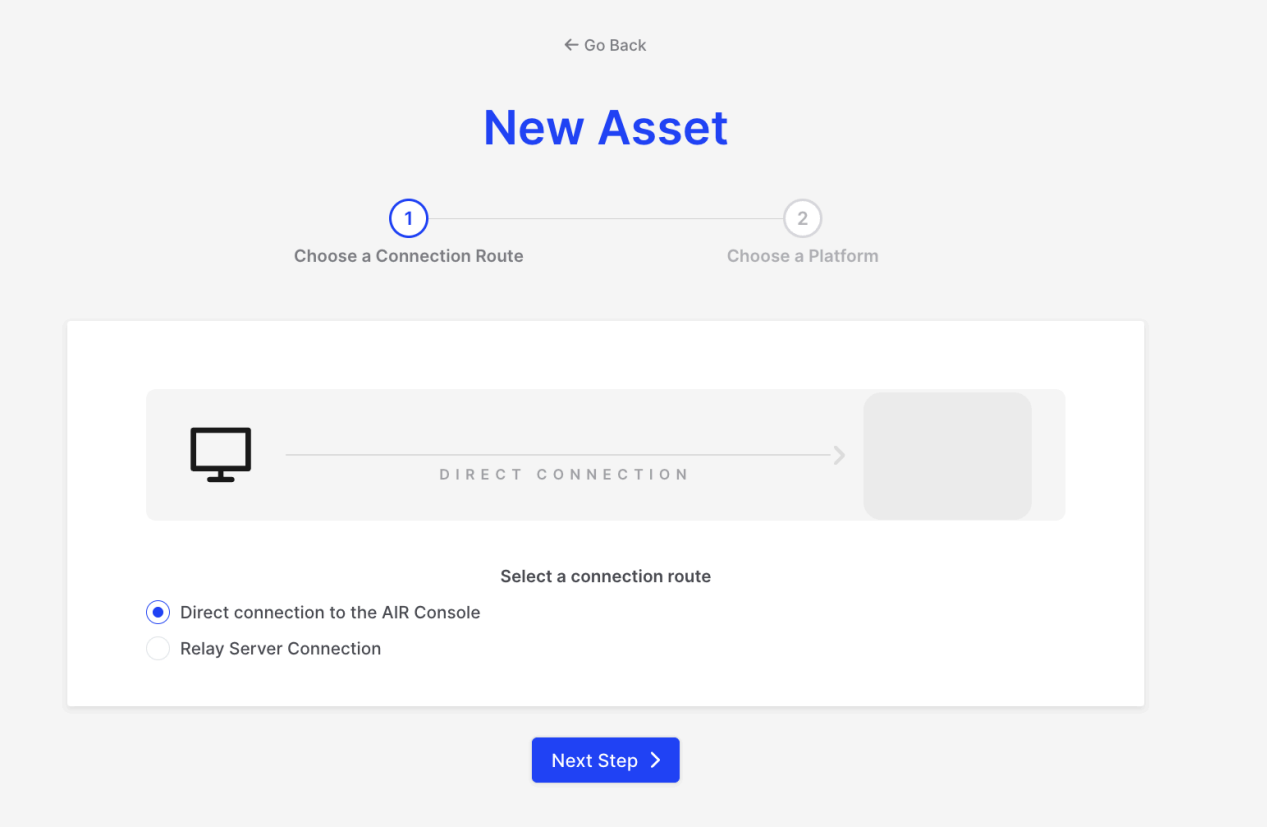
Responder Deployment: Three deployment options for adding XDR Forensics Responders to assets.

Each one of the options will present the user with a wizard that will walk through the options needed for the chosen deployment method:

1. **Deploy New** - For assets that are attached to a network that is visible to the XDR Forensics console
2. **Cloud Account** - For assets that reside in AWS EC2, and Virtual Machines in Microsoft Azure.
3. **Off-Network** - To generate triage and collection packages for assets that are not connected to a visible network.

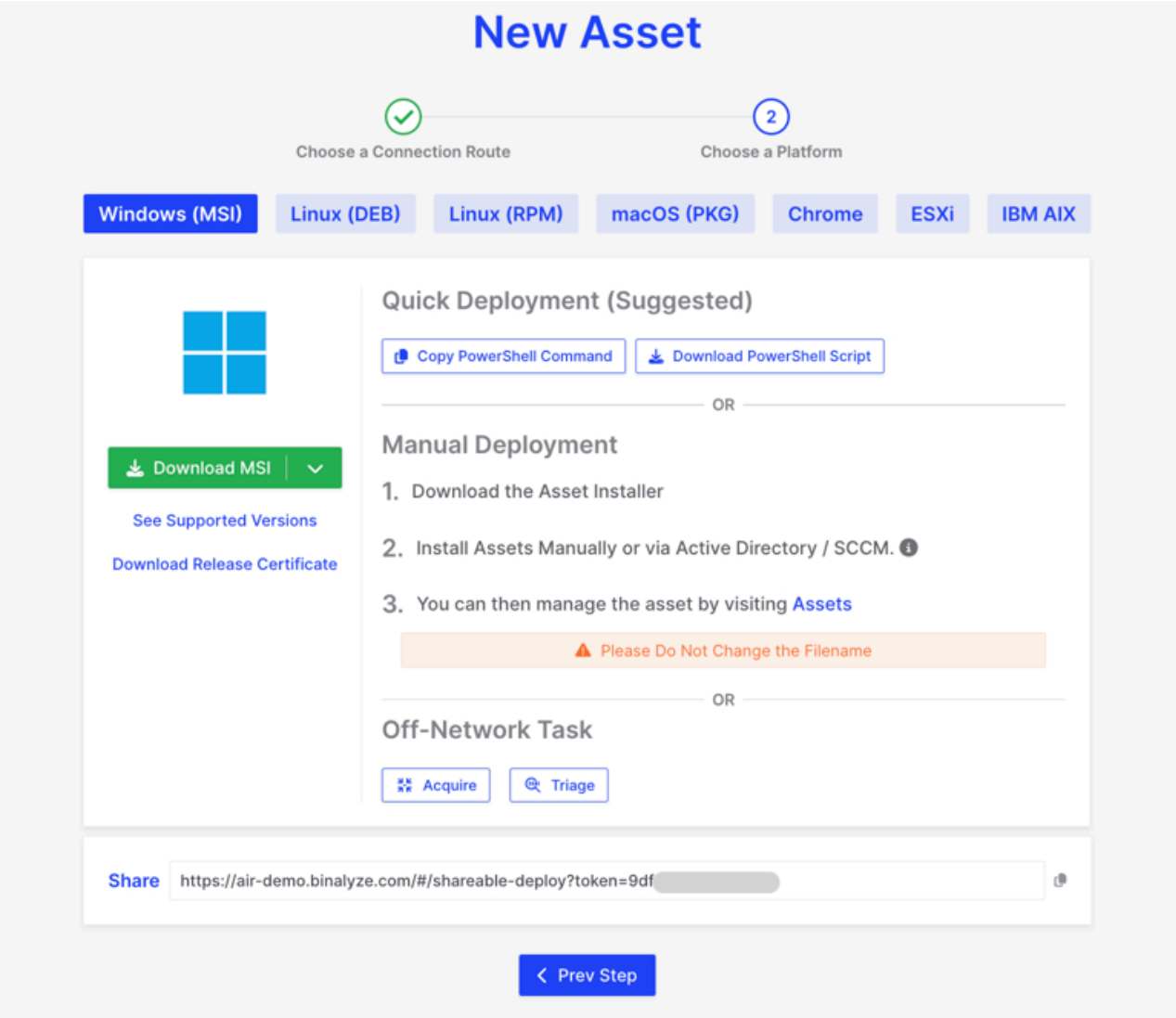
Deploy Responder to New Asset Wizard

1. When you choose 'Deploy New', you'll be prompted via a wizard to determine if the Responder should establish a direct connection to the XDR Forensics console or if utilizing a Relay Server connection would be more suitable for your environment. Relay Server is explained here. ↗



Responder Deployment: Select a connection route

2. The second step of the deployment wizard offers distinct deployment options for all currently supported network-attached operating systems: Windows, Linux, and macOS.



Responder Deployment: Choose Platform

Windows PowerShell Command:

- The command varies based on the Organization affiliation. An example PowerShell command to copy is provided below:

```
[System.Net.ServicePointManager]::ServerCertificateValidationCallback =
{$true}
(New-Object System.Net.WebClient).DownloadFile("https://air-
demo.ACME.com/api/endpoints/download/0/deploy/windows?deployment-
token=d297145XXXXXXXX", "$PWD\deploy-responder.ps1")
.\deploy-responder.ps1
```

⚠ **This command is specific to your console address and Organization.**

Windows PowerShell Script:

- This script can be downloaded from your XDR Forensics Console. Ensure you select or are working in the appropriate Organization before downloading.

```
<#
2022-2024 (c) XDR Forensics
XDR Forensics Responder Powershell Script for Windows
PLEASE DO NOT EDIT! This file is automatically generated at 2024-05-
02T13:49:58
VERSION 2.39.9
#>

<#
.SYNOPSIS
    This script installs the XDR Forensics Responder using given
parameters or default values are used.
    This script requires administrator privileges!

    MSI file is temporarily stored in %LOCALAPPDATA%\Cisco\Forensics\air

.DESCRIPTION
    Powershell script to deploy the XDR Forensics Responder.
.PARAMETER Version
    The version of the XDR Forensics Responder to be deployed.
.PARAMETER ConsoleAddress
    The address of the XDR Forensics Console without https:// prefix,
only domain address.
.PARAMETER OrganizationId
    The organization id to register the XDR Forensics Responder.
.PARAMETER DeploymentToken
    A Valid deployment token to deploy the XDR Forensics Responder.
.PARAMETER ConnectionRouteID
    Set Connection Route Id for the XDR Forensics Responder.
.PARAMETER ConnectionRouteAddress
    Set Connection Route Address for the XDR Forensics Responder.
.PARAMETER AllowInsecureTlsVersion
    Allow insecure TLS version for the XDR Forensics Responder.
#>

Param ([string]$Version="2.39.9",
       [string]$ConsoleAddress="air-demo.ACME.com",
       [string]$OrganizationId="0",
       [string]$DeploymentToken="d297145dXXXXXXXX",
       [string]$ConnectionRouteID="{{.AIR_CONNECTION_ROUTE_ID}}",
       [string]$ConnectionRouteAddress="
{{.AIR_CONNECTION_ROUTE_ADDRESS}}",
       [switch]$AllowInsecureTlsVersion)

$downloadDir = "$env:LOCALAPPDATA\Cisco\Forensics\air


Remove-Item $downloadDir -Force -Recurse -ErrorAction Ignore
```

```
New-Item -Path $downloadDir -ItemType Directory
Push-Location
Set-Location -Path $downloadDir

$arch = "386"
if ([Environment]::Is64BitProcess) {
    $arch = "amd64"
}

if ($ConnectionRouteID -like '{*') {
    $ConnectionRouteID = ""
}
if ($ConnectionRouteAddress -like '{*') {
    $ConnectionRouteAddress = ""
}

$fileSuffix = ""
if ($ConnectionRouteID) {
```

SCCM Deployment for Windows Responder:

- If you prefer, the Windows responder can be deployed using SCCM with the following command:

```
msiexec /i AIR.responder_2.24.2_air-demo.ACME.com_0_d297145XXXXXXXX_.msi
/qn /norestart
```

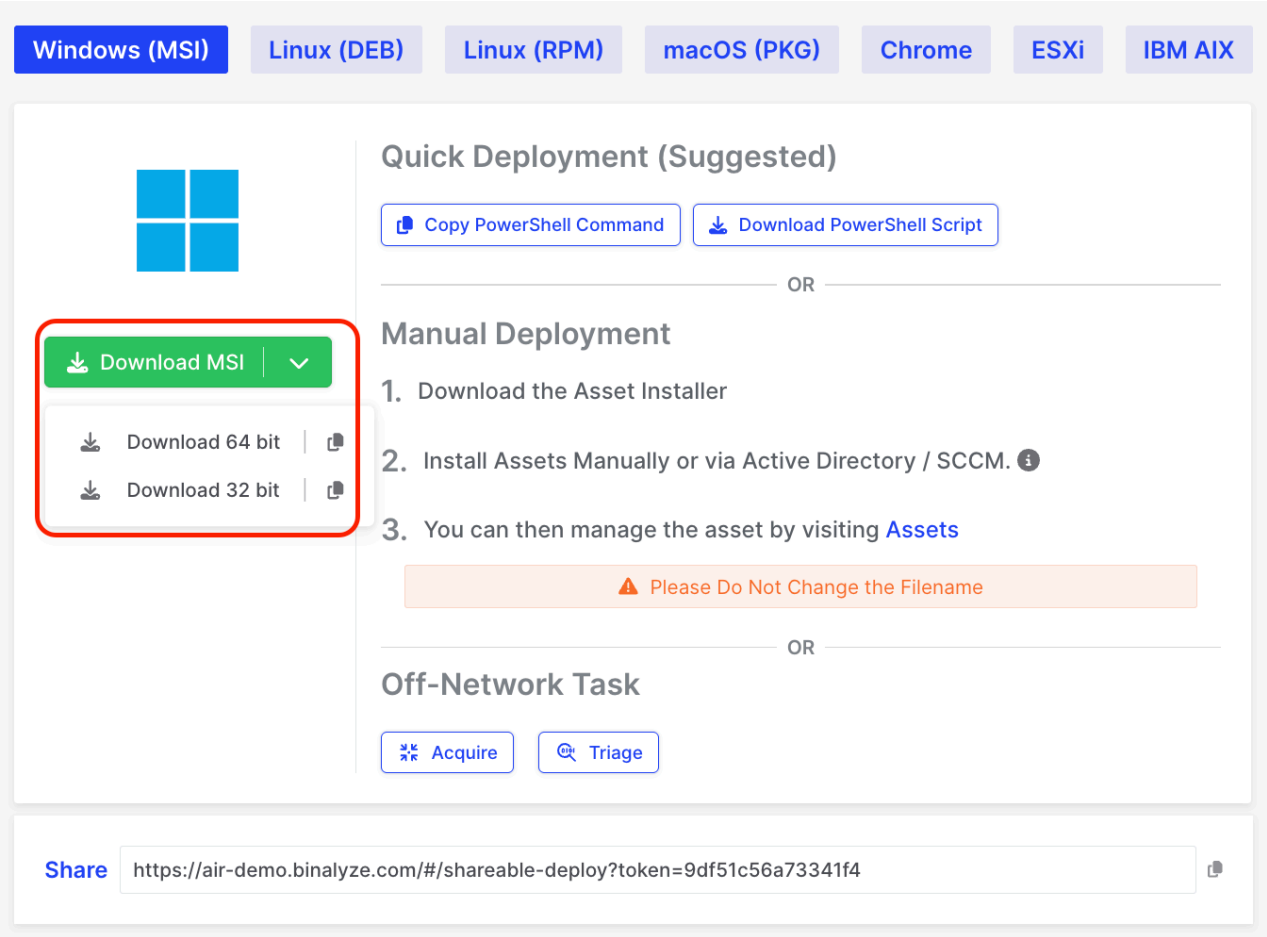
For a silent installation you can use the following command:

```
msiexec /i AIR.responder_2.26.4_air-
demo.ACME.com_176_9df51c56XXXXXXXX_.msi /qn /norestart
```

⚠ **These commands are specific to your console address and Organization.**

Windows Responder MSI Download:

- The MSI for the Windows Responder can be downloaded directly from the page, as depicted in the screenshot below:



Responder Deployment: MSI Download

Shareable Deployment Link for Windows/Linux/macOS:

- All three operating systems support the Shareable deployment link available in the console. This method is often the most straightforward—simply share the link with your client, allowing them to download and install the Responder. An example link is shown below:

```
https://air-demo.ACME.com/#/shareable-deploy?token=d297145dXXXXXXXX
```

macOS and Linux Deployments:

- Unlike Windows, macOS and Linux do not utilize PowerShell commands or scripts. Instead, they can employ **CURL or WGET** commands. Alternatively, you can use the Shareable deployment page link mentioned above.

Example of CURL deployment command:

```
sudo curl -kfsSL "https://air-
demo.ACME.com/api/endpoints/download/176/deploy/darwin?deployment-
token=9df51c56XXXXXXXX" | sudo sh
```

Example of WGET deployment command:

```
sudo wget --no-check-certificate -O- "https://air-
demo.ACME.com/api/endpoints/download/176/deploy/darwin?deployment-
token=9df51c56XXXXXXXX" | sudo sh
```

⚠ **These commands are specific to your console address and Organization.**

Granting Full Disk Access for Responder on macOS

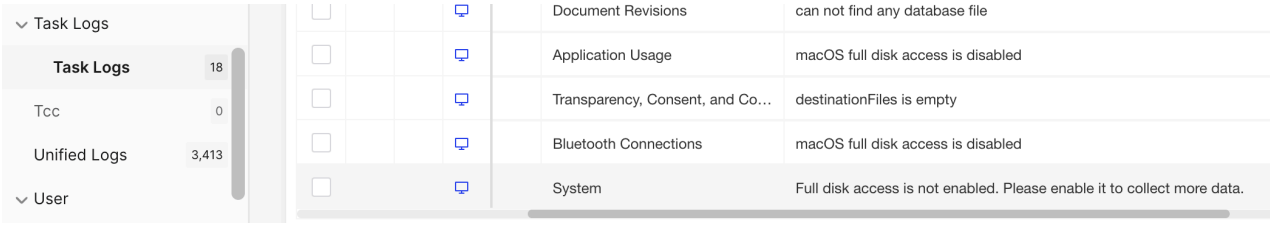
For macOS, the user/administrator must allow Full Disk Access (FDA) to the XDR Forensics Responder for it to have full access to the disk for collections.

Open "System Settings → Privacy & Security → Full Disk Access"

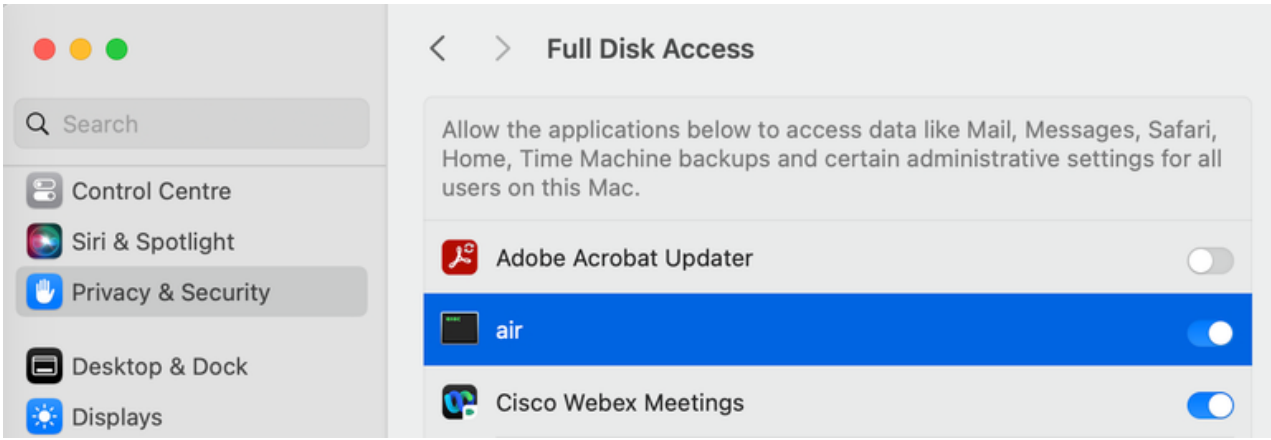
Toggle the switch 'on' to enable Full Disk Access for the XDR Forensics Responder.

After installing a responder on macOS, users will need to grant Full Disk Access permission. To guide users through this process, a pop-up will appear after installation stating: "Allow XDR Forensics to access files on your disk. Open System Settings > Security & Privacy > Full Disk Access to grant permission to "XDR Forensics".

If Full Disk Access permission is not granted when starting any Acquisition, this will be shown in the Acquisition logs:

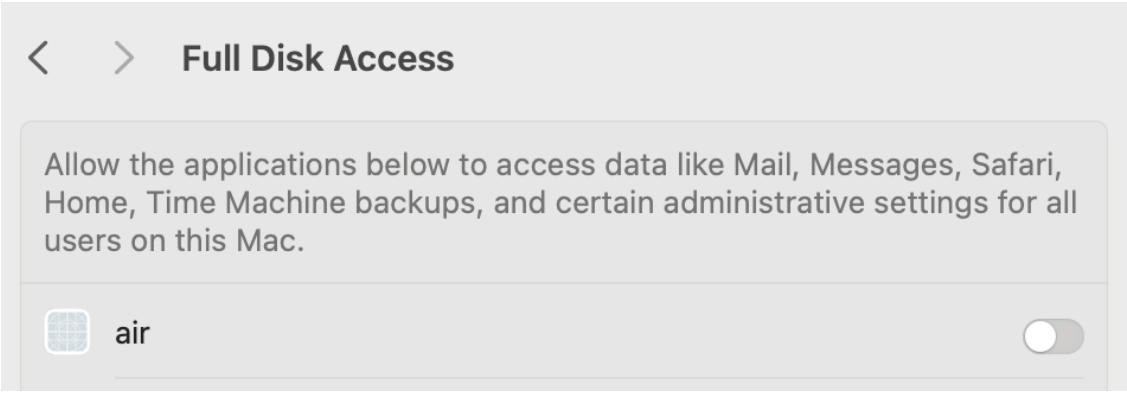


Responder Deployment: Full Disk Access permission is not enabled



Responder Deployment: Full Disk Access toggled on

After toggling on the FDA on this page, select the **/opt/cisco/forensics/air** file in the file manager that opens. Once this is done, our responder will appear in the list under the name **'air',** ready for the user to toggle 'on'.



Responder Deployment: Allow access to data.

Why is there no logo next to XDR Forensics on the Full Disk Access page in macOS?

The XDR Forensics responder operates as an executable binary running as a service rather than a traditional macOS application. This approach ensures consistency across platforms like Linux and macOS.

Since XDR Forensics is not packaged as a macOS app, it does not include a `.plist` file, which typically contains the application icon metadata. Consequently, it cannot display a logo on the Full Disk Access page.

This design choice does not affect the functionality or performance of XDR Forensics.

Problem with MDM Installation

While the pop-up effectively guides users in manually installed scenarios, it presents challenges for enterprise environments where macOS devices are managed via Mobile Device Management (MDM). MDM allows remote application installation and security policy enforcement, including granting Full Disk Access.

Customers prefer silent installations for MDM-deployed responders, as permissions are already set through security policies, eliminating the need for pop-ups. However, our current setup cannot distinguish between user-initiated and MDM-initiated installations, resulting in the pop-up appearing in all cases.

We are actively working on a solution to address this issue for seamless enterprise deployments.

Updating the XDR Forensics responder is discussed on this page.

Responder Hardware Requirements

Responder - Supported Operating Systems

Operating Systems that are supported by the responder

The XDR Forensics responder can be installed on Microsoft Windows, Linux, and Apple macOS operating systems. All supported operating systems and associated versions are listed below.

Responder - MS Windows supported systems

- Windows 7 SP1 (with latest updates)
- Windows 8
- Windows 8.1
- Windows 10
- Windows 11
- Windows Server 2008 R2 (with latest updates)
- Windows Server 2012
- Windows Server 2012 R2
- Windows Server 2016
- Windows Server 2019
- Windows Server 2022
- Windows Server 2025

Responder - Apple macOS supported systems

- macOS 10.15
- macOS 11.0
- macOS 12.0
- macOS 13.0
- macOS 14.0
- macOS 15.0

Responder - Linux (DEB/RPM) supported systems

- Centos 7
- Centos 8
- Centos 9
- Fedora 21
- Fedora 22
- Fedora 24
- Fedora 26
- Fedora 34
- Fedora 36
- Amazon Linux 1 Latest
- Amazon Linux 2 Latest
- Redhat 7
- Redhat 8
- Redhat 9
- Pardus 17
- Pardus 21
- Rockylinux 9
- Rockylinux 8
- Debian 7
- Debian 8
- Debian 10
- Debian 11
- Debian 12
- Ubuntu 12.04
- Ubuntu 14.04
- Ubuntu 16.04
- Ubuntu 18.04
- Ubuntu 20.04
- Ubuntu 22.10
- Ubuntu 23.04

- Ubuntu 24.04
- Boss Linux 7
- Boss Linux 8
- Boss Linux 9
- Boss Linux 10

All Linux distros can run on 32/64 bit and ARM64 architectures.

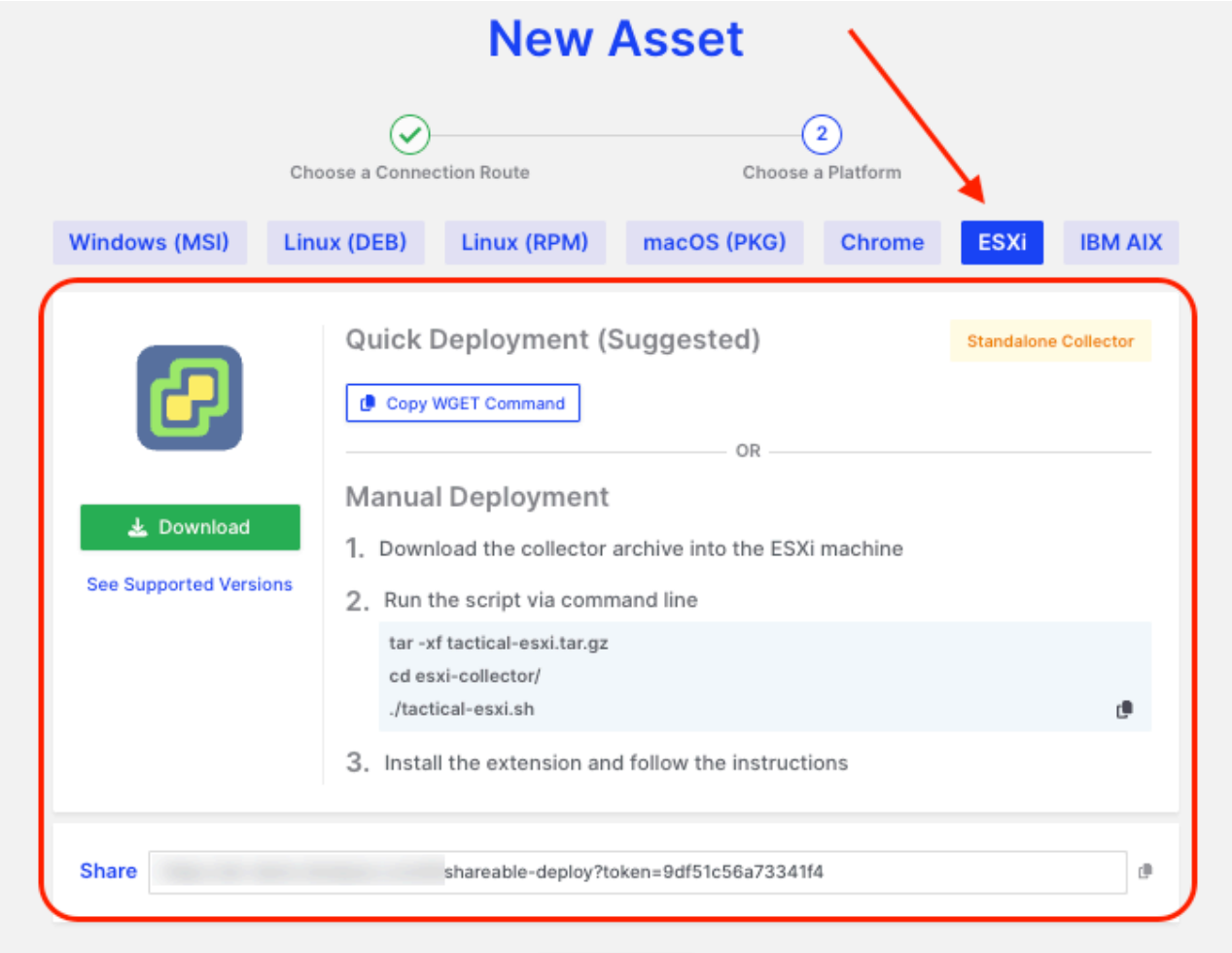
ESXi Standalone Collector

The XDR Forensics standalone collector currently provides support for execution on ESXi 6.5+ systems.

VMware ESXi is a type of hypervisor, which is software that creates and runs virtual machines (VMs). It is a part of VMware's vSphere product suite and is used for enterprise-level virtualization. ESXi is popular due to its stability, performance, and extensive feature set for managing and running virtual machines.

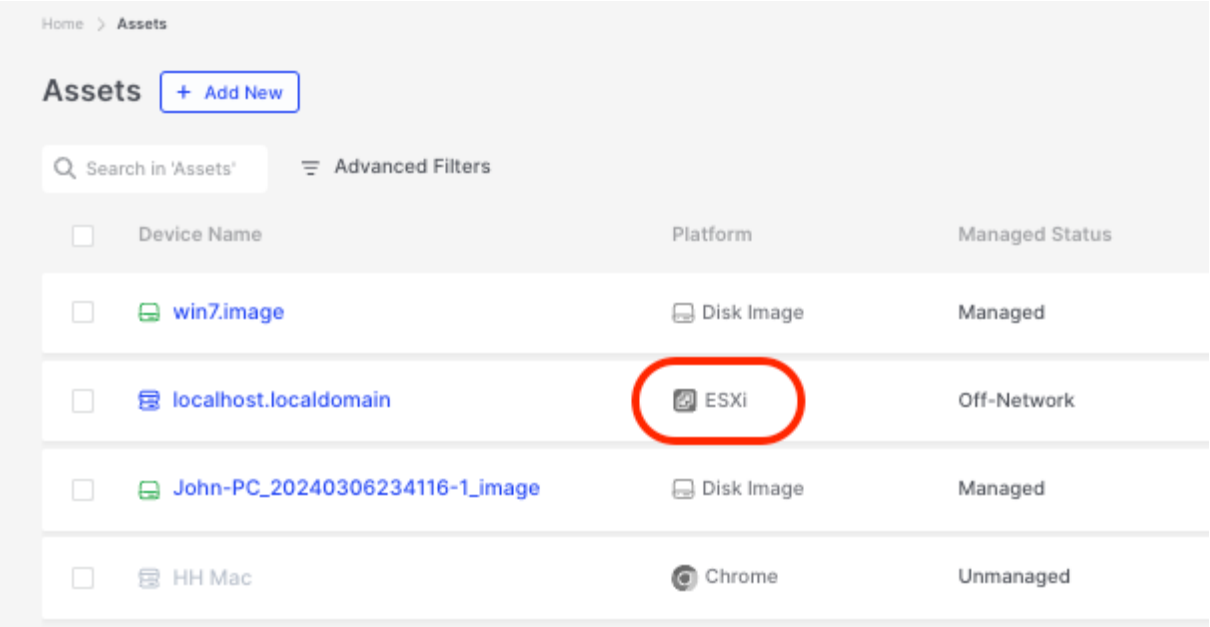
XDR Forensics offers a robust approach for evidence collection from ESXi platforms. DRONE is not currently supported for ESXi systems. This is achieved through a standalone ESXi collector, available for download on the Assets page of your XDR Forensics console:

Assets>Add New>Deploy New>Direct connection to XDR Forensics Console >ESXi



ESXi Standalone Collector: New Asset

After running Responder using your chosen method, the collected evidence should be converted into a PPC file. This PPC file can then be imported into the XDR Forensics Console. Once imported, the asset will be displayed alongside all other assets in XDR Forensics, ensuring seamless integration and visibility within the platform.



ESXi Standalone Collector: ESXi platform is shown on the XDR Forensics Asset page

For the conversion to PPC, you'll need an **off-network Responder** binary specific to your operating system on which you want to carry out the conversion.

Here's an example for Microsoft:

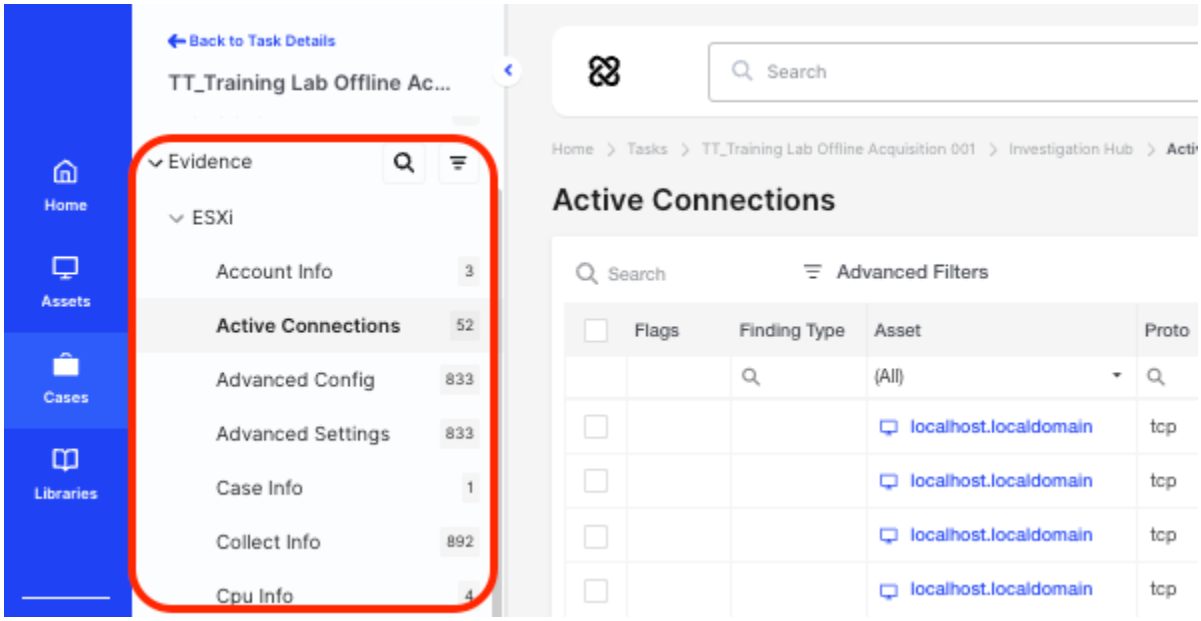
1. **Download the Off-Network Responder Package:**
   - If you are not sure where to get the binary, visit the following link for an explanation: Off-Network Responder Package ↗.
2. **Extract the Package:**
   - Extract the contents of the downloaded Off-Network Responder zip file.
3. **Prepare Your Evidence:**
   - Copy your ESXi evidence file into the same extracted folder.
4. **Run the Command:**
   - Execute the following command, replacing `your_ESXi_evidence_name` with the actual name of your ESXi evidence file:

```
offnetwork_windows_amd64 esxi --input 220240621113447-EsxiDATA.tar.gz
```

Following these steps will create a new folder containing a `Case.ppc` file. Please import this `Case.ppc` file into the XDR Forensics Console.

This process will ensure that your ESXi evidence is accurately processed and seamlessly integrated into the XDR Forensics platform.

After ingestion into XDR Forensics the ESXi evidence is parsed and pesented in the Investigation Hub in the normal way:



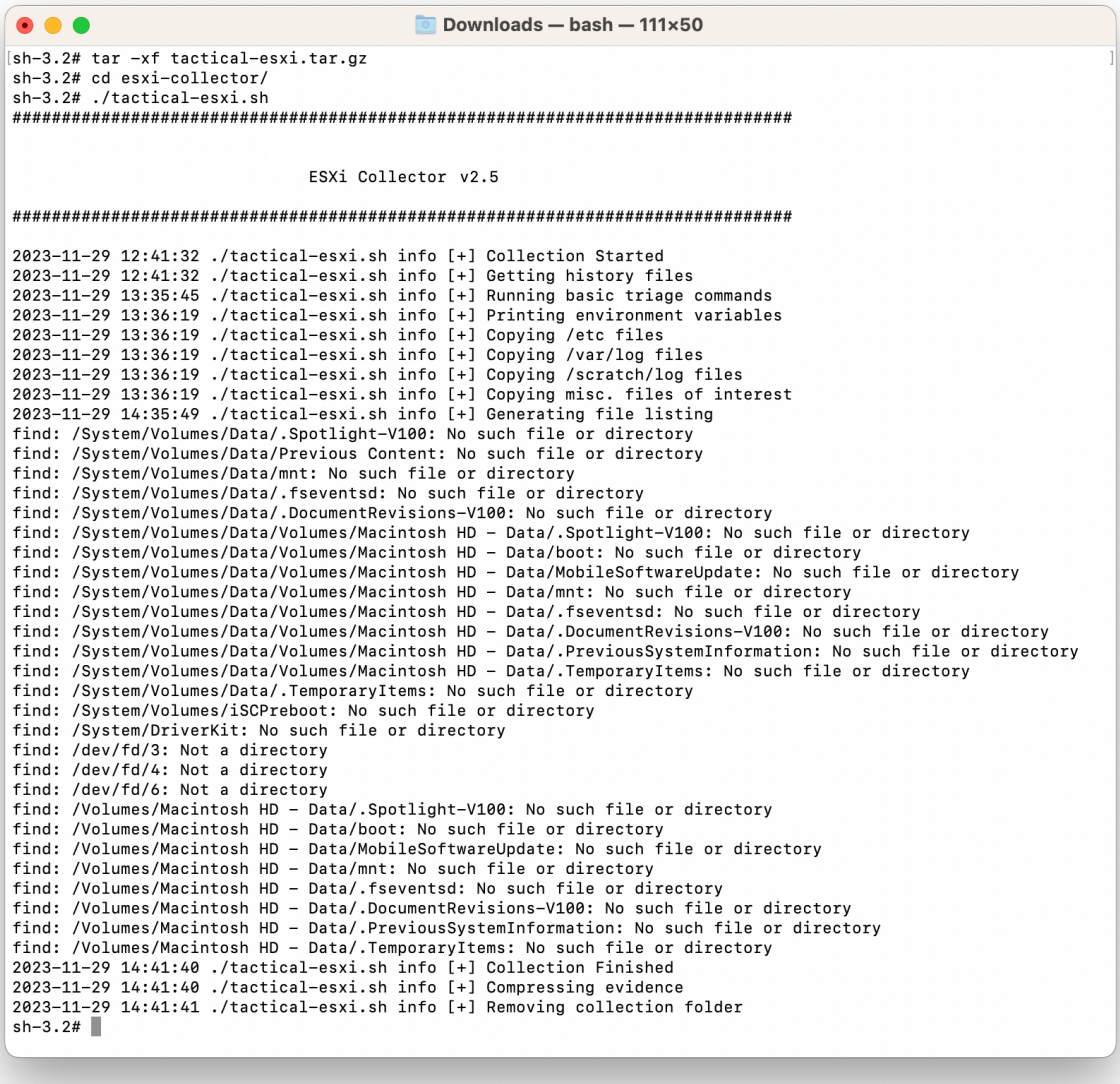
ESXi Standalone Collector: ESXi evidence in the Investigation Hub

However, you can if required decompress the tar.gz file to independently access and examine the evidence. Typically, the evidence will include the following: :

- **System Info:** Basic system information about the ESXi machine.
- **Bash History:** Command history executed on the Bash shell.
- **Collect Bash Files:** Gathering files associated with the Bash shell.
- **Environment Variables:** Variables defined in the system environment.
- **Collect /etc Files:** Gather files under the /etc directory.
- **Log Files:** Collecting various log files.
- **SSH Config:** Retrieves the configuration settings related to the SSH (Secure Shell) protocol.
- **SSH Authorized Keys:** Collects information about authorized SSH keys, which are used for secure authentication.
- **SSH Known Hosts:** Gathers details about known hosts in the context of SSH.
- **File System Enumeration:** Involves enumerating and collecting information about the file system on the ESXi machine.

A full list of ESXi collected items is shown here

Having run the binary the progress will be displayed in the user's terminal/shell:



ESXi Standalone Collector: ESXi collection example

Full list of ESXi collected items

File Collectors:

ID	Collector Name	Collected Files
1	History Files	.ash_history, .bash_history, .sh_history, .tsch_history, .psql_history, .sqlite_history, .mysql_history, .vsql_history, .lesshst, .viminfo
2	Files of Interest	.bashrc, .bash_logout, .bash_login, .bash_profile .mkshrc, .pam_environment, .profile, .zshrc, authorized_keys, known_hosts, ssh_config
3	Cronjob Files	/etc/cron.hourly, /etc/cron.daily, /etc/cron.weekly, /etc/cron.monthly, /etc/cron.d
4	Cronjob Related Files	*If any executable file is found in crontabs, it is collected.
5	/etc Collector	All files under /etc is collected
6	Log Files	All files under /var/log and /scratch/log is collected
7	Spool Files	All files under /var/spool is collected

Triage Collectors

ID	Collector Name
1	Process Snapshot Detailed
2	Process Snapshot Verbose
3	Open Files
4	User Info
5	Disk Usage
6	Disk Usage By User
7	Disk Usage Human Readable
8	System Hostname
9	VMware Version
10	System Info
11	Shell Aliases
12	Environment Variables
13	ESX Advanced Configuration
14	ESX FCoE Configuration
15	ESX FCoE Networking
16	ESX IPSec Configuration
17	ESX IPsec Policy
18	ESX Module List
19	ESX Module Query
20	ESX Multipathing Info
21	ESX NAS Configuration
22	ESX Network Interface Cards
23	ESX Routing Table
24	ESX Network Routes

25	ESX IPv6 Routing Table
26	ESX IPv6 Network Routes
27	ESX SCSI Devices List
28	ESX VMKnic List
29	ESX Volume List
30	ESX VSwitch List
31	ESX Configuration Info
32	List all of the CPUs on this host.
33	List usb devices and their passthrough status.
34	List the boot device order, if available, for this host.
35	Display the current hardware clock time.
36	Get information about memory.
37	List all of the PCI devices on this host.
38	Get information about the platform.
39	Information about the status of trusted boot. (TPM, DRTM status).
40	List active TCP/IP connections.
41	List configured IPv4 routes.
42	List configured IPv6 routes.
43	List ARP table entries.
44	List the VMkernel network interfaces currently known to the system.
45	List configured Security Associations.
46	List configured Security Policys.

47	Print a list of the DNS server currently configured on the system in the order in which they will be used.
48	List the rulesets in firewall.
49	List the Physical NICs currently installed and loaded on the system.
50	List the virtual switches current on the ESXi host.
51	Hostname
52	Get Open Network Files
53	Get Unix Socket Files
54	Get the network configuration.
55	Get the DNS configuration.
56	Get the IP forwarding table.
57	Gets information about virtual NICs.
58	Displays information about virtual switches.
59	Lists the installed VIB packages.
60	Gets the host acceptance level. This controls what VIBs will be allowed on a host.
61	Display the installed image profile.
62	List the VMkernel UserWorld processes currently on the host.
63	Collect the list open files.
64	Report a snapshot of the current processes including used time, verbose, session ID and process group, state and type.
65	List the NAS volumes currently known to the ESX host.

66	List the NFS v4.1 volumes currently known to the ESX host.
67	List the volumes available to the host. This includes VMFS, NAS, VFAT and UFS partitions.
68	Display the mapping of logical volumes with physical disks.
69	List the VMkernel modules that the system knows about.
70	List the enforcement level for each domain.
71	Get FIPS140 mode of ssh.
72	Get FIPS140 mode of rhttpproxy.
73	List the advanced options available from the VMkernel.
74	List VMkernel kernel settings.
75	Display the date and time when this system was first installed. Value will not change on subsequent updates.
76	Show the current global syslog configuration values.
77	Show the currently configured sub-loggers.
78	Display WBEM Agent configuration.
79	List local user accounts.
80	Display the current system clock parameters.
81	List permissions defined on the host.
82	Display the product name, version and build information.
83	List networking information for the VM's that have active ports.

84	List the virtual machines on this system. This command currently will only list running VMs on the system.
85	Get the list of virtual machines on the host.
86	List Summary status from the vm.
87	Configuration object for the vm.
88	Virtual devices for the vm.
89	Datastores for all virtual machines.
90	List of networks for all virtual machines.
91	List registered VMs.

Other Collectors:

ID	Collector Name	Description
1	File Listing	All files in the system is enumerated with following infos; File Name,File Type,Size (bytes),Access Rights,User ID,User Name,Group ID,Group Name,Number of Hard Links,Mount Point,Inode Number,Birth Time,Last Access Time,Modification Time,Change Time
2	Executable Hashes	All files' MD5 hashes that has executable permission in the system is collected

Responder - Chrome supported systems

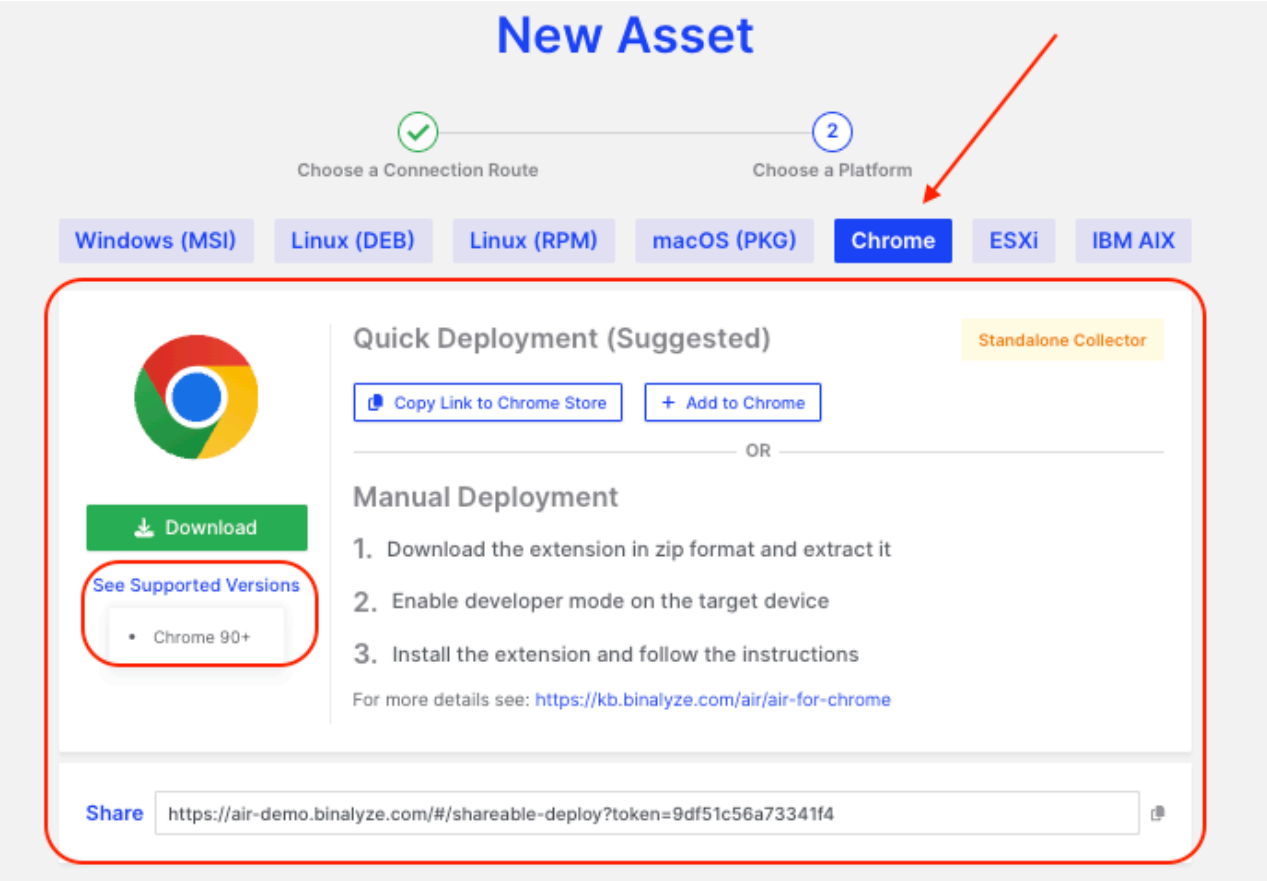
Chrome 90+

XDR Forensics For Chrome

1- Click evidence collection for Chrome

The XDR Forensics responder standalone collector currently provides support for execution on Chrome v90+ operating systems.

XDR Forensics For Chrome is the evidence collector extension for Chrome and ChromeOS. XDR Forensics For Chrome extension allows investigators and analysts to capture forensically sound data with a single click at machine speed. All data is collected into a well-organized HTML report that is accompanied by individual CSV files. Investigators and analysts can use XDR Forensics For Chrome Extension to collect forensically sound data from Google Chrome and ChromeOS.

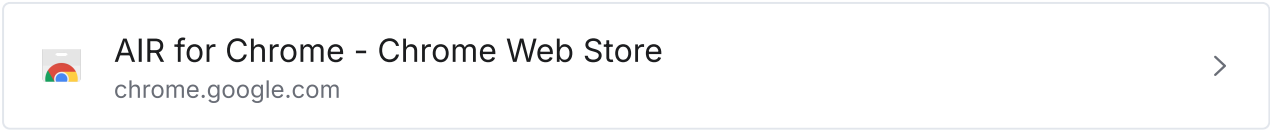


XDR Forensics For Chrome: New Asset

XDR Forensics For Chrome is the fastest and easiest way of capturing forensically sound data from Google Chrome browsers. The forensically sound data collected by XDR Forensics For Chrome are listed below.

- Browser History
- Bookmarks
- Cookies
- Downloads
- Extensions
- Platform Keys
- Privacy Settings
- Proxy Settings
- Sessions
- Storage
- Top Sites
- Windows & Tabs

Add the extension to your Chrome.



Responder for Golden Images

Golden Image is for customers who want to use the same Operating System Images to start new machines. As we use the computer name/hostname of the machine/asset as a unique identifier for the machine/asset, customers cannot use the same image in which XDR Forensics responder is already installed without the newly introduced golden image support.

It basically cleans some configuration options set during registration and then disables and stops the XDR Forensics responder service before the image of the operating system is taken. To do this, we use `--prepare-golden-image` flag that is explained below. This must be called before the imaging process takes place.

After the image is prepared, the user must use `--init-golden-image` flag, which is explained below, before the image is used to create a new instance.

`--prepare-golden-image`

The user must use this flag before creating a golden image.

Windows:

```
"C:\Program Files\Cisco\Forensics\AIR\AIR.exe" configure --prepare-golden-image
```

Linux/macOS:

```
/opt/cisco/forensics/air/air configure --prepare-golden-image
```

This flag does the following:

- Stops the service.
- Disables the service.
- Cleans the **RegisteredTo**, **SecurityToken**, and **EndpointID** fields in the **config.yml**.
- Uninstalls the watchdog (if tamper detection was enabled)

`--init-golden-image`

This flag activates the responder again after the golden image is up and after the hostname is changed.

Windows:

```
"C:\Program Files\Cisco\Forensics\AIR\AIR.exe" configure --init-golden-image --deployment-token 769aca0ff45a433a --console-address <tenantname>.cisco-<region>.binalyze.io --organization-id 0
```

Linux/macOS:

```
/opt/cisco/forensics/air/air configure --init-golden-image --deployment-token 769aca0ff45a433a --console-address <tenantname>.cisco-<region>.binalyze.io --organization-id 0
```

Note: The use of `--deployment-token` is required. Because the deployment token is clean after the registration of the XDR Forensics responder. The use of `--console-address` and `--organization-id` is optional. They are used to overwrite the console address and organization ID, which are already set in the configuration file at the first installation before the image was taken.

This flag does the following:

- Updates the **DeploymentToken**, **ConsoleAddress,** and **OrganizationID** values entered as a command in the **config.yml**.
- Starts the service.
- Enables the service.
- Watchdog is installed automatically after registration if it is enabled by XDR Forensics Console.

Troubleshooting

Exit code other than 0 (zero) means an error occurred while executing commands. The terminal will print the error messages, and the log file will contain the error messages.

If something goes wrong, the first option is to re-run the same command.

If a re-run of the command doesn't succeed, the user should perform the same steps manually.

Responder and Active Directory OUs

This page summarizes the capabilities and current limitations of Responder for Organization Units (OUs) within an Active Directory (AD) environment.

Key Points:

1. **Current Capability:**
   - Once Active Directory integration is complete, the XDR Forensics will display the domain on the Assets page.
   - Users can filter assets by clicking on their Organization Unit on the Assets page. Further filtering for "Managed Status in Managed" will show assets where the Responder is installed.
2. **Limitations and Requests:**
   - As of now, XDR Forensics does not support querying or installing Responders directly at specific OU levels (e.g., SecurityTesting.XDR Forensics.local) beyond the root AD level (e.g., XDR Forensics.local).
   - A feature request has been submitted to allow integration directly at the OU level to enhance targeted management within the domain structure.
3. **Installation Note:**
   - The XDR Forensics Responder will report on systems where it is installed. It does not automatically install on systems within an AD environment where it is not already installed.

Integrating XDR Forensics with Active Directory: Permissions Information

When integrating XDR Forensics with Active Directory, it is important to note that the account used for this integration **does not** require Domain Admin permissions. The integration primarily involves LDAP searches for reading directory information. Therefore, having Domain Users permission is sufficient for LDAP integration with XDR Forensics. This ensures that the necessary operations can be performed securely without granting excessive privileges.

Conclusion: Efforts to extend XDR Forensics's integration capabilities to specific OUs are ongoing, following feedback and feature requests. This enhancement aims to provide more granular control and efficiency in managing cybersecurity operations across different organizational units.

Responder Exception Rules

Why Create Exception Rules for XDR Forensics in EDR/AV Systems

XDR Forensics operates by collecting and analyzing extensive forensic data from assets. This process involves the execution of binaries, the creation of temporary files, and access to sensitive directories. Without proper allow-listing or exception rules in your Endpoint Detection and Response (EDR) or Antivirus (AV) systems, these activities may be flagged as suspicious or malicious, potentially leading to disrupted investigations and incomplete forensic acquisitions.

Allow-listing XDR Forensics components ensure that XDR Forensics can perform its essential tasks without interference, enabling fast, efficient investigations. By setting up proper exception rules in your security systems, you help maintain the integrity of the incident response process while avoiding false positives that could delay critical operations.

How to Configure EDR and AV Exceptions for XDR Forensics

For optimal performance, configure your EDR and AV systems to exclude specific XDR Forensics folders and binary files. The paths and binaries to exclude vary by operating system:

Windows

Folders to Exclude:

- `C:\Program Files\Cisco\Forensics\AIR\`
- `C:\ProgramData.air`

Binaries to Exclude:

- `C:\Program Files\Cisco\Forensics\AIR\\AIR.exe`
- `C:\Program Files\Cisco\Forensics\AIR\\DRONE.exe`
- `C:\Program Files\Cisco\Forensics\AIR\\Tactical.exe`
  `%ProgramData%.air\WATCHDOG.exe`
- `C:\Program Files\Cisco\Forensics\AIR\\utils\curl.exe`
- `C:\Program Files\Cisco\Forensics\AIR\\utils\osqueryi.exe`

Linux

Folders to Exclude:

- `/opt/cisco/forensics/air/air`
- `/usr/share/.air/`

Binaries to Exclude:

- `/opt/cisco/forensics/air/air`
- `/opt/cisco/forensics/air/drone`
- `/opt/cisco/forensics/air/tactical`
- `/opt/cisco/forensics/air/osqueryi`
- `/opt/cisco/forensics/air/curl`
- `/usr/share/.air/watchdog`

macOS

Folders to Exclude:

- `/opt/cisco/forensics/air/`
- `/usr/local/share/.air/`

Binaries to Exclude:

- `/opt/cisco/forensics/air/air`
- `/opt/cisco/forensics/air/drone`
- `/opt/cisco/forensics/air/tactical`
- `/opt/cisco/forensics/air/utils/osqueryi`
- `/opt/cisco/forensics/air/utils/curl`
- `/usr/share/.air/watchdog`

XDR Forensics Watchdog Folder

XDR Forensics Watchdog Folder:

`C:\ProgramData\.air\` or `%ProgramData%\.air\*`

The **XDR Forensics Watchdog Folder** (`C:\ProgramData\.air\` or `%ProgramData%\.air\`) is a critical directory used by the **XDR Forensics responder** for storing internal data required to maintain and monitor the health and proper functioning of the XDR Forensics responder agent. This folder contains temporary files, logs, and configuration data that help the **Watchdog** component of the XDR Forensics platform ensure that the responder agent is running correctly and automatically restarts the agent if any issues arise.

Purpose of the Watchdog Folder

1. **Health Monitoring**:
   The Watchdog monitors the responder agent's status. If the agent stops unexpectedly or malfunctions, the Watchdog uses this folder to store diagnostic data and trigger the necessary actions (e.g., restarting the agent).
2. **Temporary Storage**:
   The folder stores temporary files used by the XDR Forensics responder during its forensic and investigative processes. These may include logs, process monitoring data, or execution-related files.
3. **Configuration Data**:
   The directory can also house configuration and state files that help the agent track its operational state, ensuring that it maintains continuity of processes even in the event of interruptions.

Exception Configuration

When configuring **EDR (Endpoint Detection and Response)** or **AV (Antivirus)** software, it is essential to exclude this folder from being scanned or interfered with. Failure to do so may cause unnecessary alerts or interruptions to the operations of the XDR Forensics responder, potentially halting the forensic collection process or causing data collection to fail.

Folder Path Variations

- **Absolute Path**:
  `C:\ProgramData\.air\*`
  This is the standard path used by the XDR Forensics Watchdog on Windows systems.
- **Environment Variable Path**:
  `%ProgramData%\.air\*`
  This variation uses the `%ProgramData%` environment variable, which points to the `C:\ProgramData\` folder. It's a more dynamic way of referencing the same location in different system configurations.

Importance of Allow-Listing This Folder

For XDR Forensics to function seamlessly, especially during critical incident response tasks, excluding this folder from AV/EDR scans or interference is vital. The **Watchdog** service ensures that the responder remains operational and recovers automatically if disrupted.

To ensure uninterrupted operation, follow these allow-listing rules in your security setup:

- **Windows AV/EDR Systems**: Allow-list the folder `C:\ProgramData\.air\*`
- **Linux/macOS Equivalents**: Similar watchdog components may exist in those environments within paths like `/usr/share/.air/` or `/opt/cisco/forensics/air/` (adjust based on OS).

By allowing the Watchdog folder, you ensure XDR Forensics remains resilient and responsive, even in the event of unexpected issues.

FDA via Jamf and Apple's PPPC utility

Jamf is a software company that supplies one of the most well-known and popular **Mobile Device Management (MDM)** software solutions used to manage Apple devices. Using Jamf, and following the steps below, you can **silently grant full disk access to XDR Forensics responder's remotely.**

Full Disk Access (FDA) on macOS can be activated by importing a **Privacy Preferences Policy Control (PPPC)** config file instead of manually providing permission options via the Jamf UI.

Why is FDA required?

XDR Forensics (and all other platforms) will only achieve complete macOS data acquisitions if FDA is enabled. Typically some of the artifacts that will give partial or no results if FDA is not active include:

- App Usage
- Bluetooth Connections
- Document Revisions
- Downloads
- DS_Store
- Notification Info
- TCC

A **PPPC config file** in macOS manages permissions for apps to access sensitive data and system features like Full Disk Access, camera, and microphone. It's used by organizations to pre-configure these permissions, often through MDM, ensuring necessary apps run without user prompts. These files are in `.mobileconfig` (XML) format and help balance security with convenience by automating privacy settings for applications.

Steps to follow:

1. Download and open the Jamf PPPC Utility: https://github.com/jamf/PPPC-Utility/releases/tag/1.5.0 ↗
2. From a MacBook where XDR Forensics is already installed, go to the path /opt/cisco/forensics/air, drag the "air" binary to PPPC Utility, and you will be able to see identifier details
3. In properties - "Full Disk Access" → Choose "Allow"
4. Bottom right, Click "Save", and provide a Payload Name, for example, "XDR Forensics"
5. Save AIR.mobileconfig.

Now you can Import the saved config file into Jamf - Configuration Profiles.

Identifier and Identifier Type for importing the config created using PPPC utility to achieve FDA:

Verification of Full Disk Access:

- An entry is created in /Library/Application Support/com.apple.TCC/TCC.db for all the applications that were assigned FDA (Manual Install)
- For remote deployments, an entry is created in /Library/Application Support/com.apple.TCC/MDMOverrides.plist
- For practical verification, users should try to collect KnowledgeC evidence. Successful collection confirms that the responder has Full Disk Access.
- Reference: https://docs.sophos.com/central/customer/help/en-us/PeopleAndDevices/ProtectDevices/EndpointProtection/MacCheckSecurityPermissions/index.html#terminal ↗

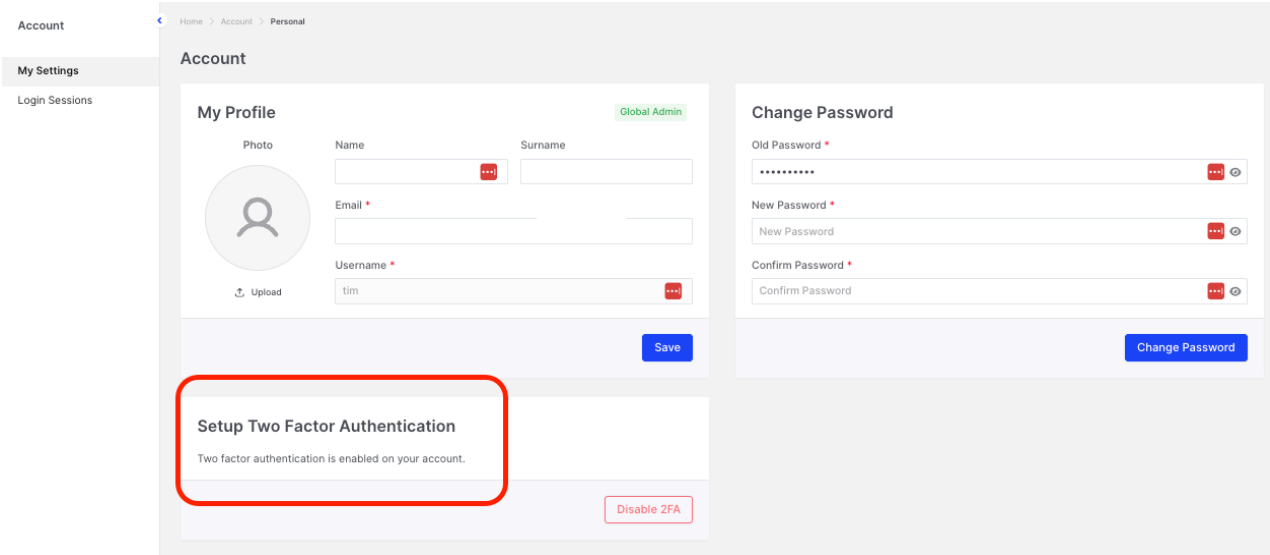
Security

Two-factor authentication (2FA)

In XDR Forensics (Settings > Security > Authentication), two-factor authentication (2FA) is a security feature designed to enhance user account protection by requiring two forms of verification when logging in. This adds an additional layer of security beyond the traditional username and password combination, significantly reducing the risk of unauthorized access.

Some key points about 2FA in XDR Forensics:

- **LDAP User Compatibility with 2FA:** XDR Forensics supports two-factor authentication (2FA) for LDAP users. You can easily configure 2FA directly from the account settings within XDR Forensics, making the setup process straightforward and efficient for centralized user management systems.
- **Administrators can enforce two-factor authentication (2FA) for all users.** This uniform security policy enhances overall security by requiring all users to authenticate with an additional method, such as a one-time password (OTP) sent to a mobile device or generated by an authenticator app.
- **Individual 2FA Setup and Reset:** Users can enable two-factor authentication (2FA) independently in 'Account > Setup Two Factor Authentication'. Global Admin and users with the "user update" privilege can reset 2FA.



Two-factor authentication: Set Up

- **Enhanced Security Posture**: By enabling 2FA, XDR Forensics significantly reduces the risk of unauthorized access, even in the event of compromised credentials. This is a critical step in safeguarding sensitive investigation data and maintaining the confidentiality and integrity of your forensic and cybersecurity operations.
- **User-Friendly Configuration**: The integration of 2FA in XDR Forensics is designed to be user-friendly, making it easy for administrators to enable and enforce 2FA without complex configuration steps.
- If you have activated the XDR Forensics SSO feature, this will override 2FA.

Troubleshooting 2FA Issues: Time Synchronization

If you are experiencing issues with Two-Factor Authentication (2FA) in XDR Forensics, it may often be due to time synchronization problems on your system. Ensuring your system's time is correctly synchronized with an NTP (Network Time Protocol) server is crucial for the proper functioning of 2FA.

Steps to Check Time Synchronization:

1. **Run the `timedatectl` Command:** Open a terminal and execute the following command:

   ```
   timedatectl
   ```

2. **Verify the Output:** After running the command, check the output for the following two lines:
   - **System clock synchronized:** `yes`
   - **NTP service:** `active`

   Here's an example of what the correct output should look like:

3. **What to Do if the Time is Not Accurate:** If your system clock is not synchronized or the NTP service is not active, this could be the root cause of your 2FA issues. To resolve this, you may need to synchronize your system's time using NTP.

How to Synchronize Your System Time:

1. **Enable NTP Synchronization:** You can synchronize your system's time by running:

   ```
   sudo timedatectl set-ntp true
   ```

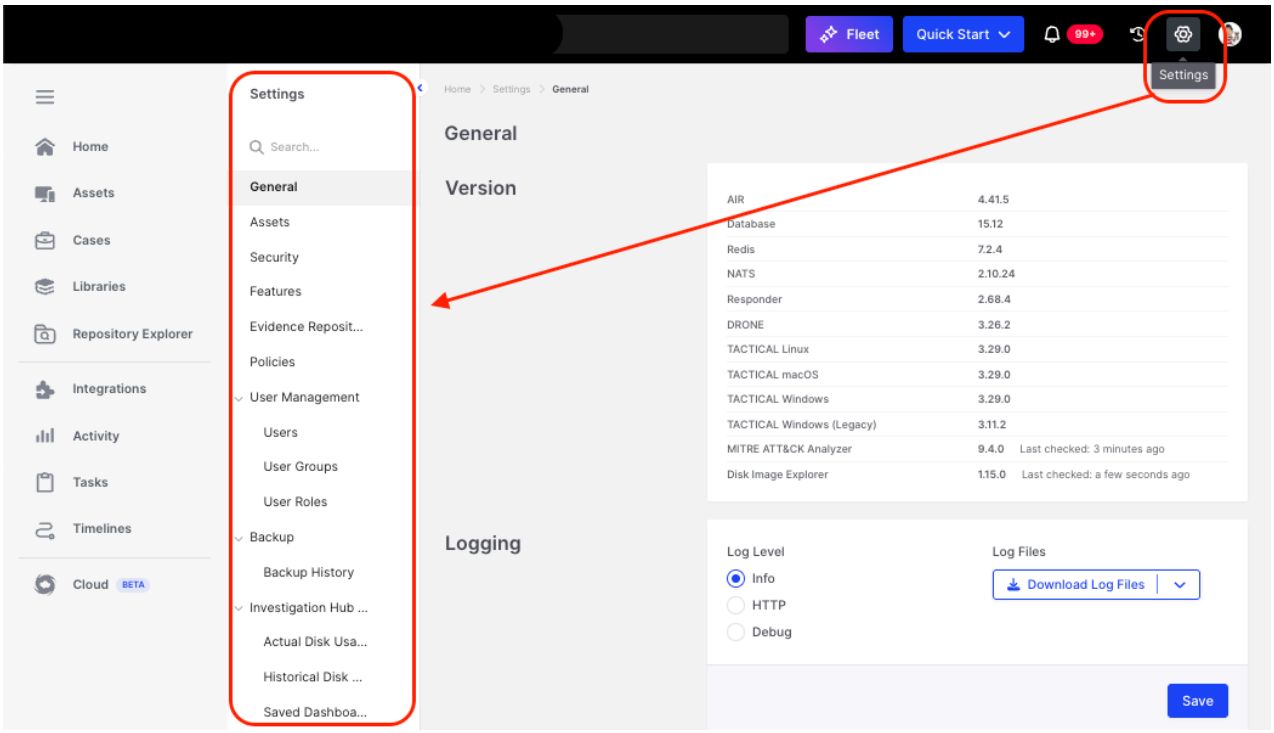
2. **Re-check the Time Status:** After enabling NTP, re-run the `timedatectl` command to ensure that the system clock is now synchronized and the NTP service is active.

By ensuring your system's time is accurate and synchronized, you can help prevent potential issues with 2FA in XDR Forensics. If the issue persists even after correcting the time, please contact our support team for further assistance.

Settings

- Console Settings
- Organization Settings
- Account Settings

Console Settings



Console Settings: Categories in XDR Forensics's Secondary Menu

All console settings can be accessed via the gear icon in the top-right corner of the header bar. This page provides a complete guide to configuring and managing settings across XDR Forensics, including:

- **General Settings**: Platform-wide configurations.
- **Assets**: Managing asset inventories.
- **Security**: Setting up security features.
- **Features**: Customizing XDR Forensics's core functionalities.
- **Evidence Repositories**: Configuring storage for collected evidence.
- **Policies**: Defining evidence collection rules.
- **User Management:**
  - Users
  - Groups
  - UserRoles
- **Backup and Backup History**: Managing backups and retention schedules.
- **Investigation Hub Disk Usage: E**mpowers users to manage **Investigation Hub data storage** effectively.
- **Danger Zone.**

Each section ensures optimal setup for your XDR Forensics environment.

General

Version Information

This section provides details on the versions of various components of the XDR Forensics platform, helping administrators ensure that all parts of the system are up to date.

1. **XDR Forensics**: The main application version (e.g., 4.41.1). This represents the core platform's release and includes the latest features and security updates.
2. **DB (Database)**: The version of the database used by XDR Forensics (e.g., 6.0.7), which stores all data related to the platform’s tasks and configuration settings.
3. **Responder**: The version of the XDR Forensics responder (e.g., 2.50.5) installed on assets for data acquisition and remote interaction.
4. **DRONE**: The version of the DRONE analysis engine (e.g., 3.11.0), which processes collected evidence to deliver findings and insights on this and some live artifacts through automated analyzers.
5. **TACTICAL**: These versions indicate the status of various responders for different operating systems, including Linux, macOS, Windows, and the legacy version for older Windows systems. For example, the latest responders are at version 3.12.1, ensuring compatibility with the latest operating system environments.
6. **MITRE ATT&CK Analyzer**: This version (e.g., 7.0.0) refers to the built-in mapping against the MITRE ATT&CK framework, which helps identify adversary tactics, techniques, and procedures during investigations.
7. **Disk Image Explorer**: This component (e.g., version 1.0.0) provides functionality for exploring disk and volume images acquired during investigations.

License

This section provides details about the current licensing status of the XDR Forensics installation.

1. **License Key**: Displays the license key currently in use (e.g., TEST-LICENSE).
2. **Valid Until**: The expiration date of the license (e.g., 2025.09.29), indicating the duration for which the platform is licensed.
3. **Max Client**: The maximum number of assets (clients) that can be managed under this license (e.g., 1,000,000 assets).
4. **In Use**: The number of assets currently being monitored by XDR Forensics (e.g., 447,908 assets).
5. **Remaining**: The number of asset slots still available (e.g., 552,092 assets). This helps ensure scalability and license compliance.

Assets

Responder Updates

Manage updates for the XDR Forensics responders installed on assets.

- This feature enables or disables **automatic updates** for responders. If enabled, the responders will automatically update to the latest version when a new release is available. This ensures that responders are always running the most current version, complete with all the latest features and security patches.
- **Deployment Tokens**: These tokens are used to securely install and register responders on new assets, ensuring the responders communicate correctly with the XDR Forensics Console upon installation.

Backward Compatibility for XDR Forensics and Responder Updates

Clarifying Backward Compatibility since XDR Forensics v4.29+

Overview
In XDR Forensics v4.29, we introduced a major improvement: **decoupling XDR Forensics console updates from Responder updates**. This gives teams greater flexibility when deploying XDR Forensics updates, especially in large-scale environments.

What This Means (and What It Doesn't)

- **Starting with XDR Forensics v4.29**, the XDR Forensics console can be updated independently of Responder updates.
- All XDR Forensics versions (4.29 and onward) will maintain **backward compatibility** with Responders **that are also on version 4.29 or newer**.
- **Responders running versions older than 4.29 (e.g., 2.54.3)** are **not compatible** with certain key features such as:
  - Evidence acquisition
  - Triage
  - interACT

Users with older Responder versions will see messages like:

> *"The asset's XDR Forensics Responder must be updated to accept tasks."*

To summarize:
Backward compatibility was introduced with XDR Forensics version 4.29 and **onwards**. If your Responders are still on versions earlier than 4.29, they **must be upgraded at least once** to benefit from this compatibility model going forward.

Tamper Detection

Enable alerts for tampering attempts on responders.

- When **Tamper Detection** is enabled, the responder will actively monitor its own operation for any interference or attempts to disable it.
- **Functionality**: If there is an attempt to modify or interfere with the responder (e.g., by disabling it or altering its files), the responder will notify the XDR Forensics Console, ensuring that any malicious attempts are flagged immediately.
- This feature is critical for ensuring the integrity and continuous operation of responders in high-security environments.

Uninstallation Password

Prevent unauthorized uninstallation of responders by requiring a password.

- When this feature is enabled, users must enter a **protection password** to uninstall the responder from an asset. This prevents unauthorized personnel from removing the responder, which could otherwise leave the asset vulnerable or unmonitored.
- **Uninstallation Method**: The uninstallation process will be restricted to shell commands, meaning it can't be removed via a simple GUI or file system manipulation, adding an extra layer of security.

Active Directory (AD) Integration

Synchronize assets from Active Directory with XDR Forensics.

- This feature allows XDR Forensics to integrate with your **Active Directory (AD)** environment. You can specify the **AD server** (e.g., 10.0.0.1) and the **domain** (e.g., company.local) to automatically synchronize information about computers and users from AD into XDR Forensics.
- **LDAP Synchronization**: By manually starting the **LDAP synchronization**, you can query Active Directory for specific objects such as computers, ensuring that XDR Forensics can discover and manage assets from your organization's AD.
- The **Query For Computers** field (e.g., `(&(objectCategory=computer))`) uses an LDAP filter to query and sync only computer objects from the directory.
- **Authentication**: You will need to provide an **AD username and password** to authenticate and pull information from the directory.

Security

Authentication

Configure user authentication security settings.

You can **enforce Two-Factor Authentication (2FA)** for all users, adding an extra layer of security by requiring a second form of verification (e.g., a mobile app code) when logging in. (SSO will override this option)

- This setting enhances overall security by ensuring that only authenticated and verified users can access the system.

Single Sign-On (SSO)

Enable and configure Single Sign-On (SSO) for XDR Forensics.

- SSO allows users to log in to XDR Forensics using their organization's existing identity provider (e.g., Azure AD, Okta) without needing separate credentials. This simplifies the login process and enhances security by centralizing authentication management.
- **Tenant ID** and **Client ID**: These are provided by the SSO identity provider (e.g., Azure, Okta) and uniquely identify the organization's SSO configuration.
- **Client Secret**: A secure key used for authenticating the connection between XDR Forensics and the SSO provider (shown as encrypted in the system).
- **Callback URL**: This is the URL where users are redirected after successful authentication via SSO (e.g., `https://<tenantname>.cisco-<region>.binalyze.io /api/auth/sso/azure/callback`) . It ensures that users are logged into the XDR Forensics platform after authenticating through the identity provider.
- **Entry Point** and **Issuer**: These fields are also part of the SSO configuration, ensuring that XDR Forensics communicates correctly with the identity provider.
- **Certificate**: Uploading a certificate from the identity provider is necessary for secure communication between XDR Forensics and the SSO service.

SSO improves user management and security by centralizing login credentials with your existing identity provider, simplifying the user experience while ensuring strong authentication practices.

Features

Enable interACT

This feature enables or disables the **interACT** functionality in XDR Forensics.

- **interACT** allows users to remotely open a shell session to interact with assets. Users can execute commands and scripts based on their assigned privileges.
- **Security Requirement**: To use interACT, users must have enhanced security in place—either **Two-Factor Authentication (2FA)** or **Single Sign-On (SSO)**. This ensures secure access to sensitive systems, limiting unauthorized use.
- Read more about interACT here: interACT ↗

Two-Factor Authentication (2FA) for isolated XDR Forensics installations

To enhance security, **XDR Forensics interACT** requires Two-Factor Authentication (2FA) using **Time-Based One-Time Passwords (TOTP)**. You can set up offline 2FA solutions such as **Google Authenticator** or **Microsoft Authenticator**, making it suitable for use in isolated networks.

Why is 2FA Mandatory in interACT?

1. **Preventing Unauthorized Access**
   interACT provides direct access to systems, making security a top priority. Relying solely on a password increases the risk of unauthorized individuals gaining control. 2FA significantly reduces this risk by adding an extra layer of authentication.
2. **Securing Critical Command Execution**
   interACT allows users to execute commands directly on a system. Without a strong authentication mechanism, a malicious actor could exploit access to perform harmful operations. 2FA ensures that only authorized users can issue commands, maintaining system integrity and security.

By enforcing 2FA, interACT safeguards against unauthorized access and potential misuse, ensuring a secure and controlled environment for forensic investigations.

Resolve Responder Public IP

This feature allows XDR Forensics to capture and associate the public IP of an asset.

- When enabled, the **XDR Forensics Console** parses HTTP request headers to extract the `X-Forwarded-For` header provided by proxies. This header reveals the **public IP address** of the responder (asset), even if it's behind a proxy or firewall.
- **Visibility**: If the feature is enabled, XDR Forensics will display the `X-Forwarded-For` IP address instead of the communication IP (the one directly visible to XDR Forensics). This provides more accurate forensic visibility of an asset's location and origin.

Case Selection

Enforce mandatory case selection when starting tasks.

- This feature requires users to associate every task they run in XDR Forensics with a specific **case**.
- **Benefit**: It enforces structured workflows, ensuring that all investigations are organized and traceable to a particular case, which is critical for auditing and maintaining clarity in incident response efforts.

RFC3161 Timestamping

Provides cryptographic proof of when data was acquired and its integrity.

- RFC3161 timestamping ensures that the data collected during acquisition has a digital signature, proving that the data existed at a specific time and has not been altered since.
- When enabled, every new acquisition task will include a signature file with metadata, adding legal and forensic robustness to your investigation process.

Chain of Custody

Protect evidence integrity by registering it on the blockchain via LOCARD, which is a blockchain-based system for secure evidence handling in digital forensics. It has seen some adoption in Europe but remains underutilized in the U.S. due to regulatory and infrastructure challenges, leading to slower adoption and less frequent use.

- This feature integrates with **LOCARD**, a blockchain-based platform for evidence integrity. When enabled, the chain of custody for digital evidence is secured by submitting evidence metadata to the blockchain, ensuring it hasn't been tampered with.
- **LOCARD Credentials**: To use this, you'll need to provide the **Organization**, **Host**, **Username**, and **Password** for your LOCARD account.

SMTP (Email Configuration)

Set up email notifications, such as password-reset emails.

- Specifying an **SMTP server** allows XDR Forensics to send out automated emails, particularly for password resets. This is useful for self-service password recovery.
- You must configure the SMTP server address, port, sender email, username, and password. For example, using `mail.smtp2go.com` as the server.

Syslog / SIEM Integration

Enable integration with Syslog servers or SIEM systems.

- This feature allows XDR Forensics to send event logs to a centralized **Syslog** or **SIEM (Security Information and Event Management)** system for enhanced log monitoring and analysis.
- You will need to configure the **protocol** (TCP/UDP), **server address**, and **port** to send logs from XDR Forensics to your preferred log management system.

Banner Message

Display a custom banner message across all XDR Forensics Console pages.

- This feature allows you to set a **banner message** that will appear on all pages of the XDR Forensics Console. This is useful for displaying system notices, warnings, or other important information to all users.

Policies

Enforce task options and preferences across assets.

Policies allow administrators to define global task preferences and restrictions for assets in the organization.

- **Customizability**: Policies can be tailored for different subsets of assets using filters, and a user must have the "Override Policy" privilege to modify the default organizational policies.

Auto Asset Tagging

Automate tagging of assets when they are added to XDR Forensics.

- When this feature is enabled, XDR Forensics automatically applies **asset tags** based on predefined rules as soon as a responder is installed on an asset.
- **Flexibility**: Even if this feature is disabled, users can still run the **Auto Asset Tagging** task manually on assets.

Enable Frank.AI

Activate AI-powered assistance for investigations.

Frank.AI is an **AI-driven assistant** integrated into XDR Forensics. It helps guide users through investigations, providing suggestions and assistance to streamline the forensic analysis process. Frank.AI acts as a copilot for investigators, improving efficiency by leveraging AI to answer analysts' questions.

Evidence Repositories

XDR Forensics allows you to set up various **Evidence Repositories** for storing and managing collected data securely. The supported repository types are:

1. **SMB**: Ideal for sharing files across network devices.
2. **SFTP**: Utilizes SSH for encrypted data transfer.
3. **FTPS**: Combines FTP with SSL/TLS for secure transfers.
4. **Amazon S3**: Provides scalable cloud-based storage, perfect for large-scale investigations.

Key Features:

- **Global or Organization-Level Setup**: Repositories can be defined at both global and organizational levels, providing flexibility in evidence management across multiple XDR Forensics instances or within a single organization.
- **Secure Data Management**: Protocols like SFTP and FTPS ensure that data transfers are encrypted, safeguarding sensitive information during uploads and downloads.
- **Automatic and Manual Uploads**: Evidence can be automatically uploaded to repositories based on configured tasks, or users can manually upload files as needed.
- **Task Management**: Repositories support task scheduling for evidence uploads, ensuring a smooth workflow for collecting, storing, and analyzing evidence.
- **Connection Settings**: When configuring repositories, users must provide essential connection details such as credentials, encryption options, and repository paths. For cloud-based storage like Amazon S3, you also need to configure bucket settings.

This setup ensures secure, scalable, and efficient management of evidence within XDR Forensics, accommodating various infrastructure needs.

Policies

Policies serve to define how evidence is collected and managed, providing fine-grained control over resources and processes.

Policies in XDR Forensics provide central configuration management and support global configurations that can be overridden at the Organisation level when required.

This overriding is **only possible** when the user has the “Override Policy” privilege allocated to their role.

Key Components:

1. **Name & Organization**: Policies must have a unique name and be assigned to a specific organization.
2. **Evidence Storage**: Configures where evidence is stored—either locally (default paths: `CiscoForensics\` on Windows, `/opt/cisco/forensics/` on Linux/macOS) or in defined repositories like SMB or SFTP.
3. **Resource Limits**: Controls CPU usage, bandwidth, and disk space during collection to prevent resource overuse. You can specify CPU limits (e.g., 100%) and restrict bandwidth and disk space.
4. **Compression & Encryption**: Enables optional compression and encryption of the collected evidence, with a password for added security.
5. **Scan Scope**: You can opt to restrict scans to local drives only, excluding network and external drives.
6. **Isolation Settings**: Policies can include an IP/Port and ‘process allow’ lists for isolation tasks, which ensures that specific communication channels remain open during an asset’s isolation.

Use Case Example:

When creating a policy for a specific investigation, you could configure it to save evidence in an AWS S3 bucket, limit the CPU to 50%, compress the evidence for efficient storage, and ensure network drives are excluded from the scan. You could also configure the policy to allow communication with critical servers even if the asset is isolated.

User Management

XDR Forensics > Settings > User Management > Users:

This section enables administrators to view existing users and their attributes. It also allows for the addition of new users to the XDR Forensics platform, where key details such as name, organization, role, and login credentials are specified during setup.

Mandatory fields are marked with an asterisk (*).

1. **Full Name**:
   - **Name**: The first name of the user being added (e.g., "John").
   - **Surname**: The last name of the user (e.g., "Doe"). These fields are crucial for identifying and managing users within the system, particularly in larger organizations.
2. **Username***:
   - The **username** is a mandatory field (indicated by the asterisk). This is the unique identifier that the user will use to log in to the XDR Forensics platform (e.g., `IH_reviewer2@cisco.com`).
   - The username is often based on the user's email address to ensure uniqueness and facilitate easy recognition.
3. **Email***:
   - The **email** is also a mandatory field. It is used for account-related communications, such as password resets, system alerts, or notifications.
   - This email should be valid and associated with the user being created to ensure they receive important platform-related information.
4. User Groups:
   - Select the User Groups to which the new user needs to be added.
5. **Organization***:
   - This field allows you to assign the new user to a specific **organization** within the XDR Forensics system.
   - If multiple organizations are managed within the XDR Forensics platform (e.g., in a multi-tenant setup), you can select the organization to which the user belongs.
   - The system can restrict users from viewing or managing other organizations, depending on their access privileges.
   - **Note**: If no organization is selected or assigned, the user may have limited or no permissions within the platform.
6. **Role***:

- The **Role** dropdown allows you to assign the user's role within the platform. Roles define the level of access and permissions the user will have. Common roles could include:
  - **Administrator**: Full access to manage the platform, users, and assets.
  - **Investigator**: Access to forensic and incident investigation features.
  - **Viewer**: Read-only access to view data and reports.
- This field is crucial for setting user permissions and ensuring that they can only perform actions aligned with their responsibilities.

7. **Password***:
   - This is where you set the **password** for the user’s account. The password should meet the organization's security requirements (e.g., complexity, length).
   - A secure password is crucial to prevent unauthorized access to the platform.
8. **Confirm Password***:
   - This field is used to **confirm** the password entered above. Ensuring that the passwords match helps avoid login issues caused by incorrect entries.

User Groups

User Groups are a powerful feature in XDR Forensics designed to streamline user access management across organizations, especially in large environments or those using Single Sign-On (SSO).

What Are User Groups?

User Groups allow you to manage users collectively instead of individually. This simplifies role assignments and organization-level access across XDR Forensics. Whether you're handling internal teams or integrating with Azure AD or Okta, User Groups provide scalable, centralized control.

Why Use User Groups?

Managing user access across multiple organizations can be time-consuming. With User Groups, you can:

- Group users by function, department, or region
- Assign them to organizations and roles in bulk
- Sync groups directly from SSO providers like **Azure AD** or **Okta**

This reduces manual overhead and ensures consistency in access management.

Accessing the User Groups Page

To access this feature:

- Navigate to **Settings** → **User Groups**

> *Note: Only Global Admins can access and manage User Groups.*

Creating a New User Group

1. Click **Create Group**
2. Enter a **Group Name** (e.g., "HR Team")
3. Optionally, provide a description for context

Adding Users to a Group

After creating a group:

1. Select users from your existing XDR Forensics user list
2. Users in the group automatically inherit all organization and role assignments associated with the group

> *You can always view and manage group membership from the main User Groups table.*

Syncing with SSO (Optional)

If you're using Azure AD or Okta:

- Toggle **"Sync with SSO"** when creating or editing a group
- Once enabled, group membership is pulled from your identity provider
- Manual editing of group members in XDR Forensics will be disabled

> *A tooltip will indicate that synced groups are read-only within XDR Forensics.*

Assigning a User Group to an Organization

1. Go to **Settings → Organizations**
2. Create or edit an organization
3. In **Step 2** of the wizard, assign the desired User Group(s)

All users in the selected group will be granted access to the organization with their predefined roles.

Managing Group Membership

- In the **User Groups** table, click the **user count** to view members

> *For SSO-synced groups, membership cannot be modified in XDR Forensics.*

Safeguards When Removing Users

Users added via a group **cannot be removed individually** from an organization.
To revoke access:

- Remove the user from the **User Group**

> A tooltip will appear if removal is attempted at the individual level.

Viewing Group Membership from the Users Page

The **Users** list now includes a **Groups** column to show which user groups a person belongs to. This offers helpful context when reviewing or auditing user access.

Notes and Limitations

- Only **Global Admins** can create and edit User Groups
- **Group duplication** is not yet available, but it is planned
- When a user is assigned to an organization both individually and via a group, the **group assignment takes precedence**
  - To fully remove access, remove the user from the group

Summary

User Groups simplify access control and scale with your environment. Benefits include:

- Centralized user and role management
- SSO directory sync support
- Consistent access inheritance across organizations

Whether you're managing 10 users or 10,000, User Groups provide a more efficient and secure way to manage access in XDR Forensics.

User Roles

In XDR Forensics, the **Global Admin** has full control over managing **109 specific privileges**, allowing the creation of highly customized user roles. This granular access control ensures that each user or group has permissions tailored to their specific needs, such as handling evidence acquisition, interACT sessions, or audit log management.

A useful feature within this setup is the **tooltips** provided alongside each privilege. These tooltips highlight any **dependencies** that may exist between privileges, helping administrators configure roles accurately without unintentionally restricting necessary functions.

For example, an admin could create a role that enables a user to access interACT for remote evidence collection while restricting access to audit logs or system-wide settings. The tooltips ensure that admins are aware of any required privileges to avoid misconfigurations.

This approach provides both flexibility and clarity, empowering admins to manage user roles effectively.

Backup

XDR Forensics > Settings > Backup:

The **XDR Forensics Backup** feature allows users to back up system data securely and flexibly through the **UI** or **Command Line Interface (CLI)**. Backups can be stored locally, on **SFTP**, or in **Amazon S3**, and encrypted using **AES256** with a password.

Backups can be performed **immediately** or scheduled at intervals of **every 4 hours, daily, weekly, or monthly**. Users can set the number of backups to retain and the scheduled start time. CLI backup options are available, with detailed instructions in the **Knowledge Base**.

XDR Forensics > Settings > Backup History:

This page displays the Backup History avaiable to the XDR Forensics console.

Investigation Hub Disk Usage

This feature empowers users to manage **Investigation Hub data storage** effectively and is fully explained here: Investigation Hub – Data Usage Statistics Dashboard.

Danger Zone

Killswitch: Cancel All Running Tasks

The **Killswitch** is a powerful administrative control available under **Settings > Danger Zone** in the XDR Forensics Console. It allows Global Admins to immediately cancel *all currently running cancellable tasks* across the entire platform, without needing to select individual assets or tasks.

Safety Confirmation

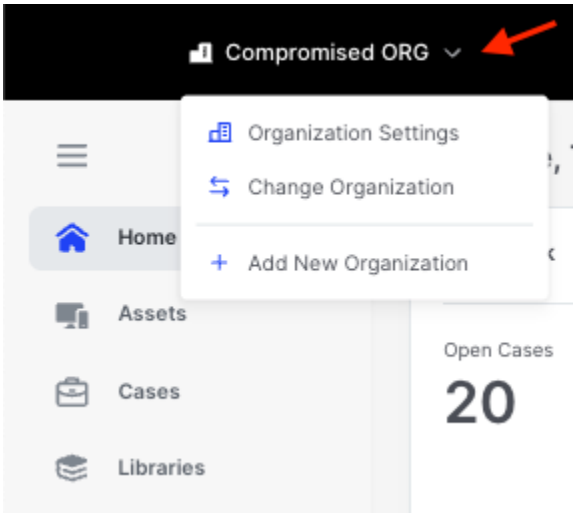
To safeguard against accidental activation, a case-sensitive confirmation is required. You must, before the operation can proceed. Type the exact phrase: `Cancel All Tasks`

> **Note:** This action cannot be reversed.
>
> Only tasks that are in a cancellable state will be affected.

Use this feature carefully, particularly during large-scale investigations or scheduled operations.

Organization Settings

Organizational Controls



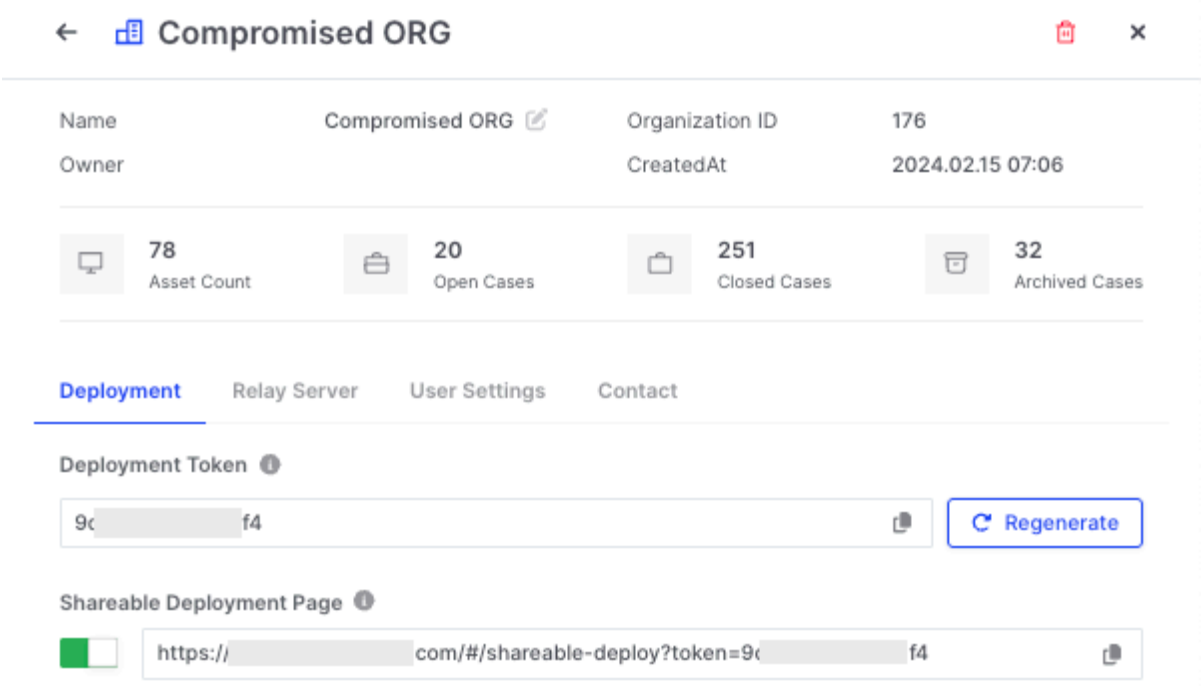
Organization Settings: Access vis the top bar

From the top bar, the drop-down allows the user to select:

- **Organization Settings**
- **Switch Organizations**
- **Add new Organizations**

> Each responder is associated with a single Organization, ensuring integrity and clarity in investigations.

Organization Overview



Organization Settings: Overview

When viewing an Organization, you'll see key details like:

- **Name**, **ID**, **Owner**
- **Created At** timestamp
- **Asset count**, and case stats (Open, Closed, Archived)

Configuration Tabs

Each Organization has four main configuration areas:

1. Deployment

- Displays the Organization's **deployment token**
- Includes a **Shareable Deployment Page** for fast responder distribution
- The link can be regenerated or disabled from this view

2. Relay Server

- Manage Relay Servers used to support off-network response
- Includes search and "New Relay" functions

3. User Settings

- View and manage users, roles, and groups
- Use the **Assign Users** button to add new team members

4. Contact

- Optional form fields for storing the Organization's contact name, title, and phone number

Use Cases

Organizations help:

- MSSPs manage multiple customers
- Enterprises segment data by business unit, region, or compliance zone
- Security teams isolate cloud, on-prem, and off-network assets

Access & Policy Enforcement

- Policies can be global or overridden at the Organization level
- Shareable deployment links and responder access are organization-scoped
- User roles support granular privileges

Summary

Organization Settings are central to structuring your AIR environment securely and efficiently. They govern deployment, user access, case management, and responder scope—all from a single, intuitive interface.

Account Settings

The **Account Settings** section in XDR Forensics allows users to manage personal preferences and session details. This section is accessible via the **user icon** located in the **top-right corner** of the XDR Forensics interface toolbar.

What You Can Do in Account Settings

- **View Account Details**: Includes your registered name, surname, email address, and username.
- **Change Password**: Locally authenticated users can update their password directly from here.
- **Sign Out**: Securely log out of your XDR Forensics session.
- **Customize Theme Preferences**: Switch between Light and Dark Mode or set the UI to follow system time-based settings.

Dark Mode

The XDR Forensics interface supports a **Dark Mode** UI theme, providing a modern, eye-friendly visual style that is ideal for use in low-light conditions or during extended periods of use. This feature:

- Enhances usability and reduces eye strain.
- Can be toggled manually between **Light** and **Dark** modes.
- Includes an **Auto** mode that adapts the UI based on your machine's **time zone** and **daylight saving hours**, ensuring a seamless experience.

Dark Mode aligns with current UI standards and user expectations, supporting a comfortable and productive working environment.

User Profile Photos

Users can upload their own profile images. These will be displayed across the XDR Forensics console UI in user-facing areas, such as the user menu, case history, and activity logs. Supported for both regular and SSO-authenticated users, although not synced from the identity provider.

Updating

Updating the XDR Forensics console

Console Updating - SaaS

XDR Forensics offers flexible options for updating your SaaS tenant, ensuring you always have access to the latest features and improvements.

Update Options

Customers can choose from the following methods to update their XDR Forensics SaaS tenant via; **Settings>Update:**

1. **Manual Update**: Initiate the update by clicking the "Update" button within the XDR Forensics Console user interface.
2. **Automatic Updates**: Enable the auto-update feature in settings to allow XDR Forensics to update automatically when a new version is released.
3. **Scheduled Updates**: Configure updates to occur at a preferred time, minimizing disruption to your operations.

> **SaaS - Automatic Backup Before Updates**
>
> XDR Forensics automatically generates a backup before every SaaS console update. This eliminates the need for manual backups during the update process. If an update fails, the system can roll back to the previous working state using this backup. This capability ensures resilience and simplifies update management in our SaaS environment.

Features

Fast, remote, and scalable across the corporate network

- Acquisition
- Auto Tagging & Tags
- Triage
- interACT
- Compare
- Timeline
- DRONE
- Investigation Hub
- Repository Explorer
- Evidence Repositories
- File Explorer
- Asset Isolation

Policies

Off-Network Responder

Responder Proxy Support

Console Audit Logs

Asset Isolation

Isolating assets during an investigation

Asset Isolation works by terminating all connections of an endpoint and not allowing any new connections.

When an asset is isolated, you can still perform tasks such as Acquisition, Triage, interACT and Time-lining.

How it works

This feature uses a Kernel Mode Driver for performing the isolation and does not depend on Windows Firewall.

The isolation task is persistent. Even if you reboot an isolated machine from the XDR Forensics Console, the asset will still be isolated after the reboot until you un-isolate it from the Asset Details page.

Acquisition

Data acquisition is the collection of forensically sound data from any computer system (disk, external storage, memory, etc.). This data generally varies based on the operating system installed on the computer or server. Acquired data often needs to be parsed, stored, and presented in a human-readable format for further analysis and investigation.

Data acquisition is the primary activity of most digital investigations. Before data acquisition, the investigators generally identify the data they'll need. Since data or evidence is an essential element of any investigation, investigators tend to take as much as they can in the first instance to avoid, if possible, a second acquisition. Therefore, the power of the digital investigation and DFIR solution is often proportional to the acquisition capability and the features associated with it.

XDR Forensics provides easy-to-deploy and fast data acquisition capabilities with a wide range of operating systems supported for the collection of 600+ forensically sound data types. XDR Forensics provides remote data acquisition for on-premise, cloud, and off-network devices. Thus, investigators can remotely investigate multiple devices at speed and scale.

XDR Forensics supports a growing number of operating systems, including Windows, Linux, macOS, ChromeOS, ESXi, and IBM AIX.

The results of XDR Forensics's Acquisition and Triage processes can be further analyzed using DRONE's automated Post Acquisition Analyzers. DRONE's findings, along with all collected artifacts, are then presented within the **Investigation Hub**.

The XDR Forensics responder needs to be deployed first to acquire data. All data acquisition is performed according to the Data Acquisition Profile created before the acquisition is started.

Data acquisition is classified into three categories: Evidence, Artifacts, and Network Capture. Additionally, investigators have the flexibility to create custom content profiles, allowing them to collect specific files or data from designated locations.

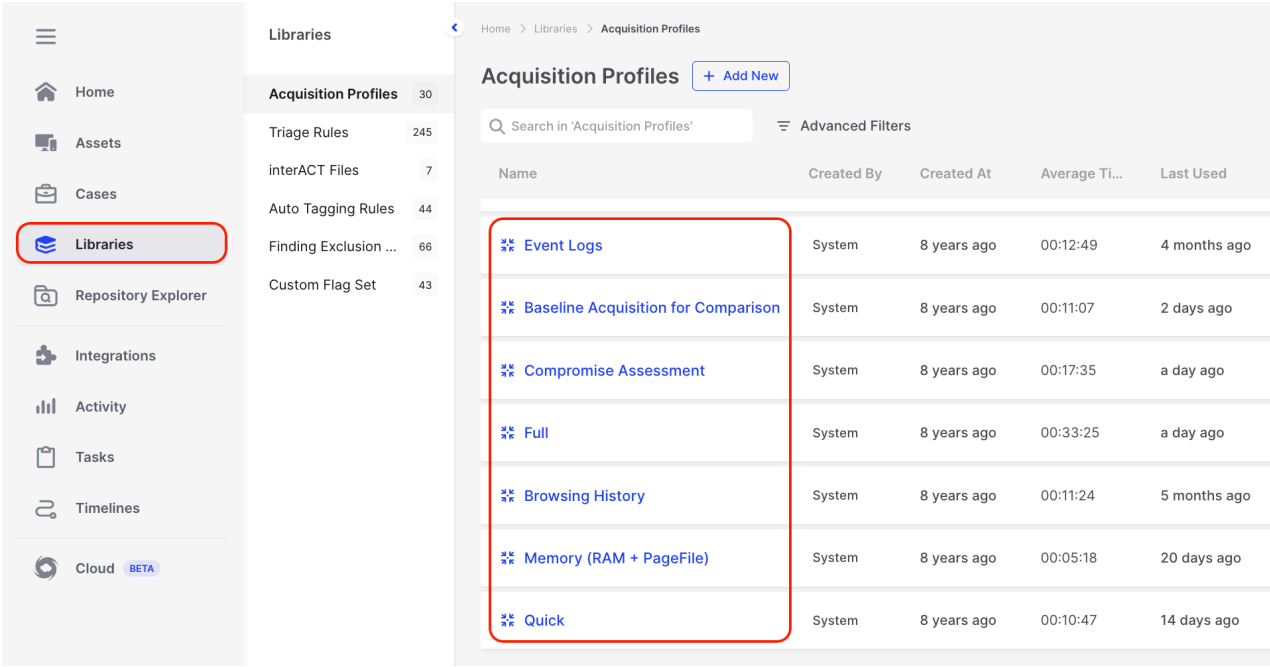
At Cisco, we recognize the critical importance of maintaining a strong 'chain of custody' when it comes to the collection and handling of evidence. That's why we employ SHA-256 hashing in combination with RFC3161 digital timestamp certificates. This approach serves to safeguard data content and offers assurance regarding the precise timestamp of the content's creation, as well as a guarantee that it has remained unaltered.

Read more about RFC3161 and the secure way XDR Forensics maintains a strong chain of custody here: **Protect Your Chain Of Custody With Content Hashing And Timestamping** ↗

Acquisition Profiles

Acquisition profiles in XDR Forensics define the specific types of data to be collected during an acquisition task. These profiles enable you to customize and streamline data collection to meet the unique requirements of your investigation. Saved within the XDR Forensics Libraries, acquisition profiles can be easily shared, reused, or edited for further refinement, ensuring efficiency and consistency across investigations.

Using ‘Out-of-the-Box’ Acquisition Profiles



Acquisition Profiles: Built-in Profies

As shown above, XDR Forensics comes with several predefined acquisition profiles that you can use immediately, for example:

1. **Quick**: Designed for fast data acquisition with essential evidence types.
2. **Full**: Collects a comprehensive and rich set of data from the assets.
3. **Compromise Assessment**: Focuses on indicators of compromise and suspicious activity, defined by the XDR Forensics threat hunting team.

These ‘out-of-the-box’ profiles are ideal for common scenarios and provide an ideal quick start for your investigations.

Creating your own Acquisition Profiles

To create your own custom acquisition profile, follow these steps:

(1) Navigate to Acquisition Profiles:

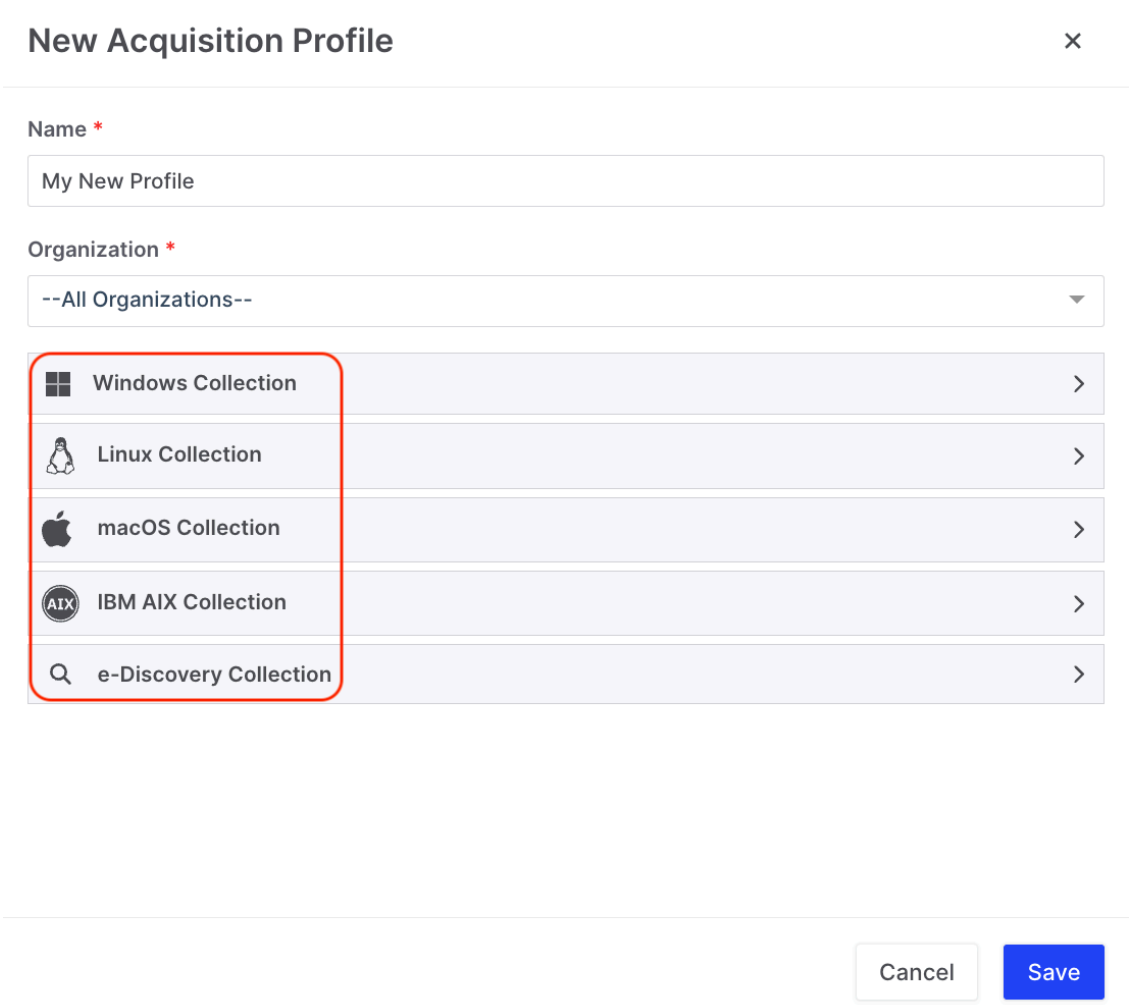
- Go to the "Libraries > Acquisition Profiles" section from the main dashboard.

(2) Create a New Profile:

- Click on the "+ New Profile" Action Button.
- Provide your new profile with a name that will help you identify its purpose later.

(3) Select the Operating System(s) for your new profile:

- Windows
- Linux
- macOS
- IBM AIX
- Or a cross-platform eDiscovery collection



Acquisition Profiles: Supported platforms and eDiscovery option

(4) Select Evidence Types:

- XDR Forensics supports an ever-growing number of evidence types for collection and presentation in the Investigation Hub. To build your profile, choose the data you want to collect from the extensive options grouped under the following five tabs:
  - **Evidence List**
    - System artifacts (e.g., registry hives, event logs)
  - **Artifact List**
    - Application artifacts such as server Logs, RMM, AV tools, etc
  - **Event Log Records**
    - XDR Forensics allows users to collect and present event logs or define specific channels for log collection. (product-platform/features/drone/analyzers/windows-analyzers/windows-event-records-and-how-they-are-handled/))
  - **Custom Content Profiles**
    - Select bespoke file locations for collection.

Custom Content Profile Path Tips

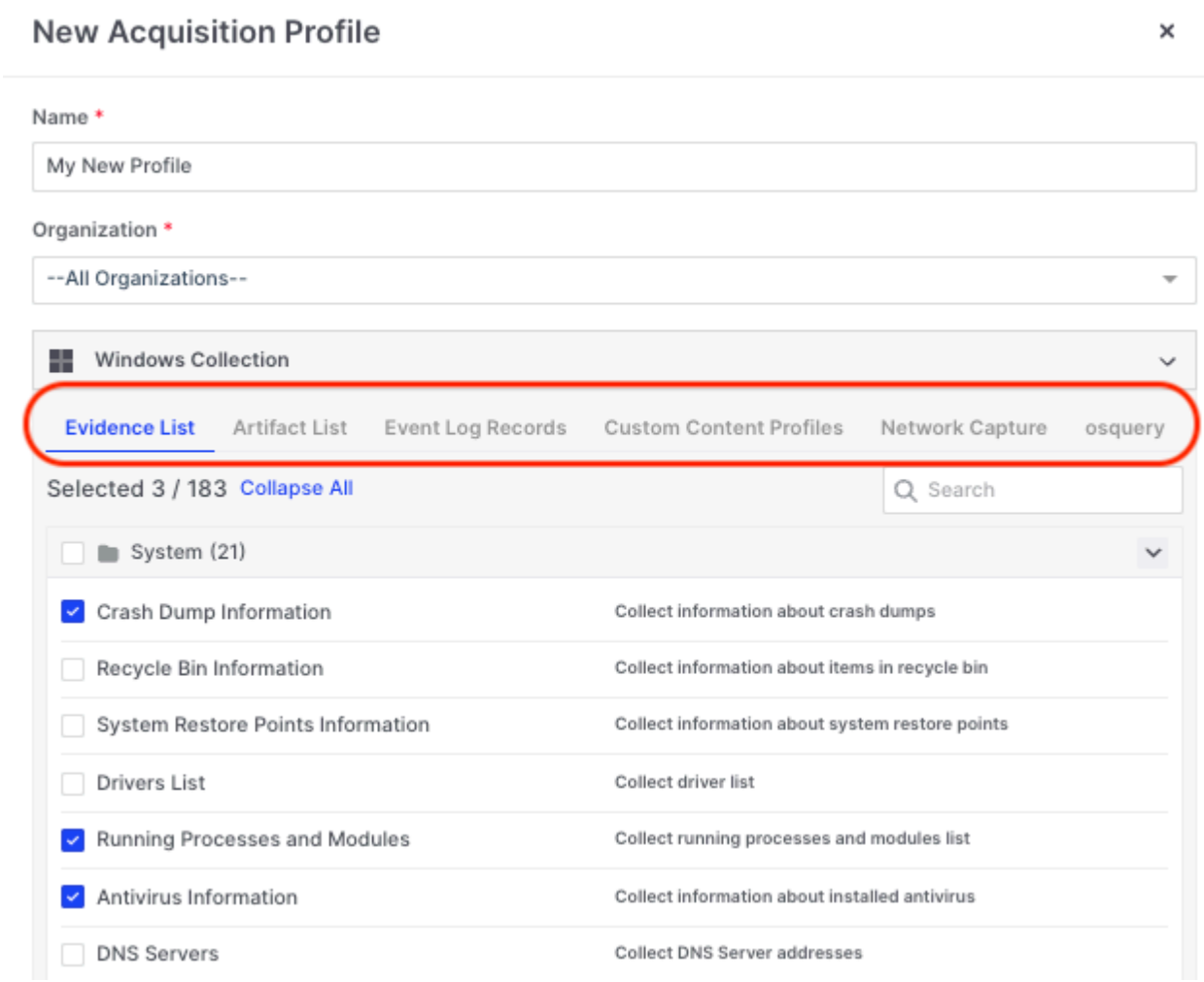
- If using **"Files and Folders Recursively"**, no need to end paths with `*`.
- You can omit the drive letter — start from the folder name (e.g., `Program Files\Common Files`).
- Use `**` for recursive matching:
  - `C:\Users\**\.txt` collects all `.txt` files under that path.
  - `**\malware.exe` (or `**/malware.exe` on Unix) finds all `malware.exe` files system-wide.
- Windows paths can use `/` or `\` — both are supported.

- **Network Capture**

  Network Flow captures TCP/UDP connections and stores them as a CSV.

  PCAP will capture IP packets and save them as a PCAP file.

  The duration of the Network Capture is determined by the user.
  - **osquery**
    - Use osquery language to capture data.



Acquisition Profiles: Choose items for collection from the 6 tabbed groupings

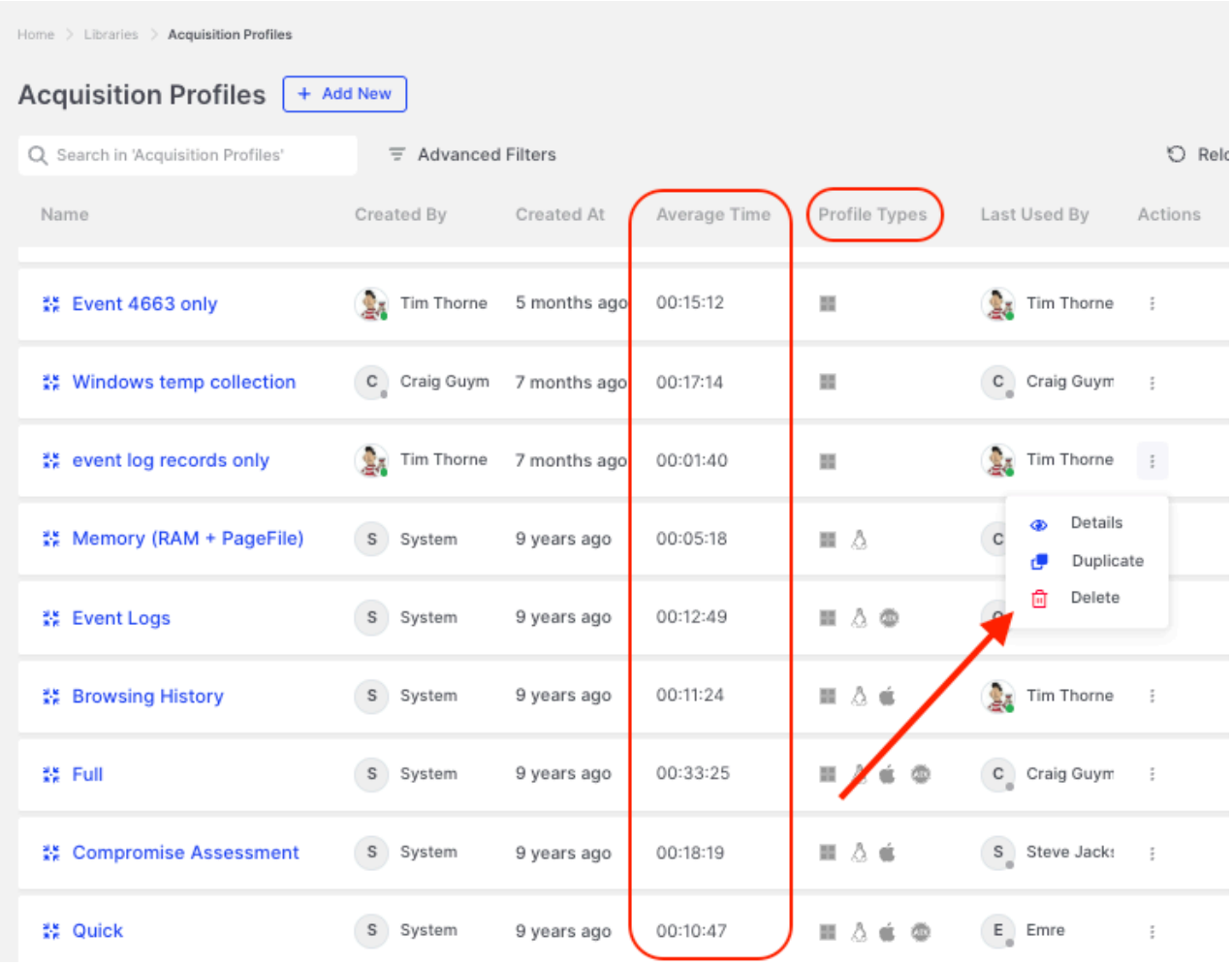
- **(5) Save the Profile**:
  - Once you have configured all the necessary settings, click "Save" to create your custom acquisition profile.

Managing Acquisition Profiles

- **Edit Profiles**: You can edit existing profiles by selecting the profile and making necessary changes.
- **Delete Profiles**: Remove profiles that are no longer needed to keep your list organized.
- **Duplicate Profiles**: Create a copy of an existing profile to use as a template for a new one.
- **User Privileges** for acquisition profiles can be managed via ‘Settings > Roles’

Best Practices

- **Check Profiles**: Ensure your acquisition profiles are up-to-date with the latest evidence types and investigation requirements.
- **Test Profiles**: Test new profiles in a controlled environment to ensure they collect the intended data.
- **Average Time Taken:** In the Acquisition Profiles table, you can see the 'Average Time' taken by each profile. This can be useful when considering the performance and efficiency of individual profiles.
- **The Profile Types** column provides a quick visual reference to indicate which operating systems—Windows, Linux, or macOS—are supported by each acquisition profile.



Acquisition Profiles: Average time to run profiles and action button to duplicate the profile.

By using acquisition profiles in XDR Forensics, you can efficiently gather relevant data for your investigations, saving time and ensuring comprehensive evidence collection.

Chain Of Custody

How XDR Forensics Protects your Chain of Custody with content hashing and RFC3161 Time-stamping

At XDR Forensics, we use **SHA-256** to hash all of the files collected by XDR Forensics, and then we take this to the **next level**. We achieve this by further hashing our .ppc collection file and sending that value to a Trusted Timestamp Server to generate a certificate.

This not only proves that the report and all of the data associated with it exist exactly as it did on acquisition, but it did so at the date and time notarized by a Trusted Timestamp Authority (TSA) certificate.

Thanks to RFC3161, you can not only prove that the data content is 100% intact, but also that the date and time of collection are guaranteed.

Trust in RFC 3161

Request for Comments (RFC) is a system that has been adopted as the official documentation of Internet specifications, communications protocols, procedures, and events. Used initially to record unofficial notes related to the ARPANET project in 1969, the system is now considered a standard-setting body for the Internet and its connected systems.

A published RFC will undergo a review and revision process, overseen by several groups, including the Internet Engineering Task Force (IETF), a large, open international community of network designers, operators, vendors, and researchers. As part of their collective role, they review the evolution of everything related to the development of internet architecture and the smooth operation of the internet. A list of RFC 3161-compliant Time Stamping Authorities (TSAs) can be found here. ↗ When choosing TSAs, users may want to consider whether their implementation of RFC 3161 has been qualified by organizations such as eIDAS (Electronic Identification and Trust Services).

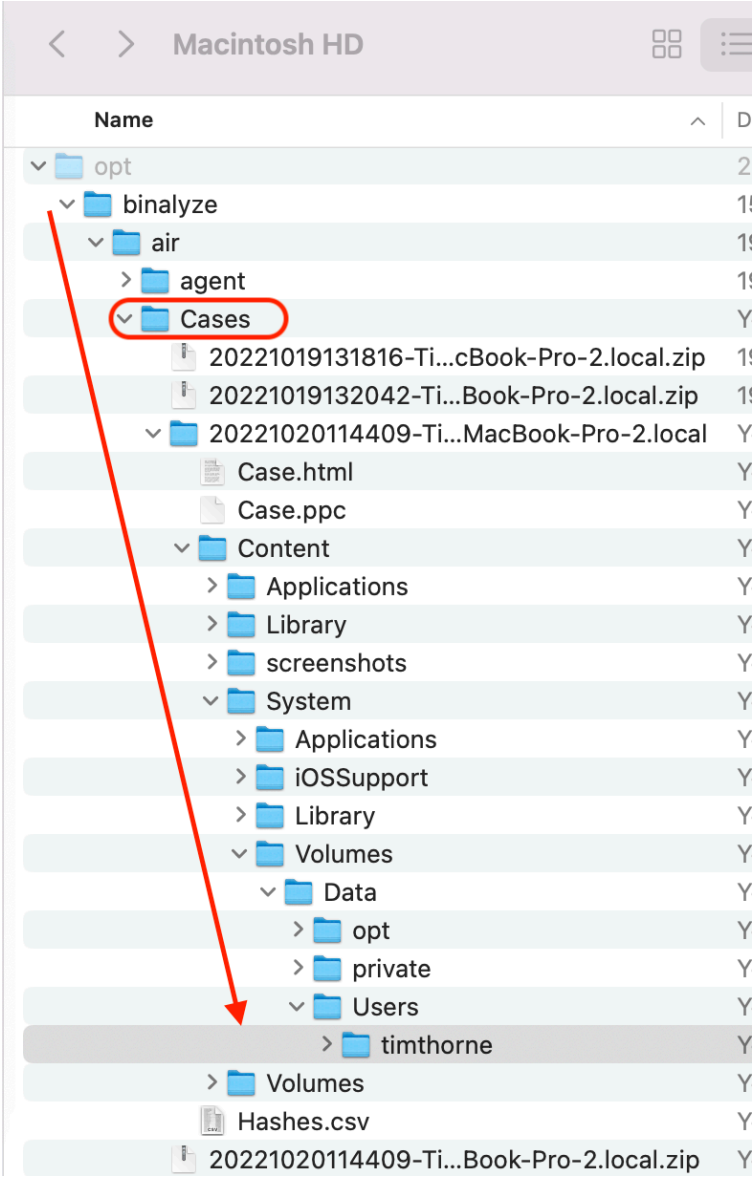
RFC3161 defines how trusted timestamping leverages public-key cryptography and the Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP) sets the required protocols for standardization.

One way to use a TSA allows a requestor to take the hash they've generated for the total of their collected data set, send that hash to the TSA, and receive in return a Timestamp Request Token (TST). This TST can be saved and, at any later time, used to verify both the content of the collection and the date and time at which the collection took place.

The RFC 3161 capability is not unique and is available from a whole range of independent third parties. This is important as any in-house time-stamping process could be open to challenge or criticism due to its lack of independence or verified accuracy.

How does this work in XDR Forensics?

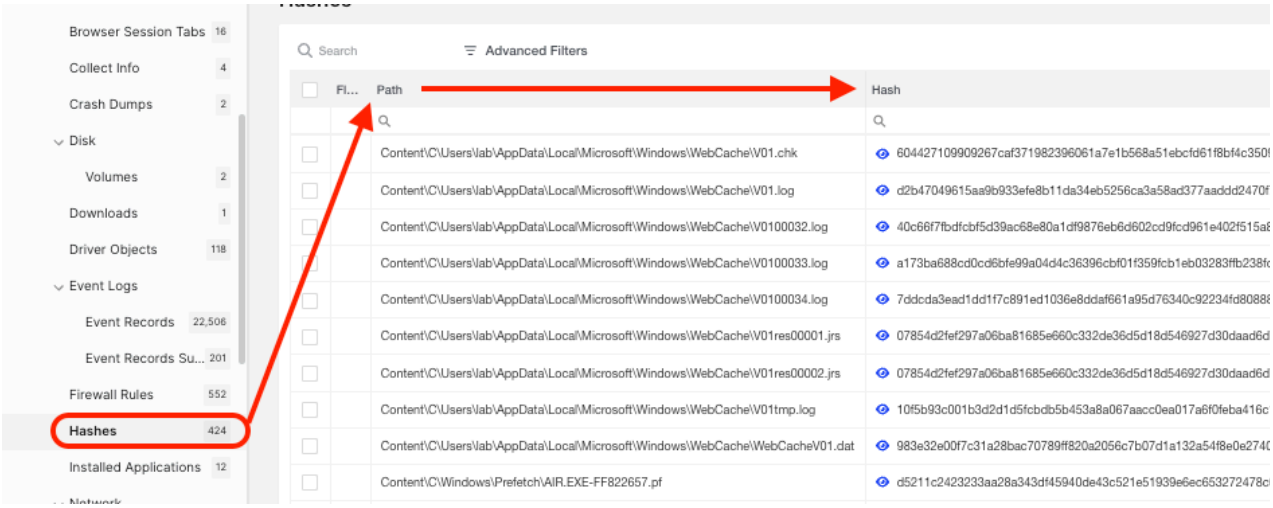
In the XDR Forensics platform, when you send a collection task to an asset's responder, the responder will build the collection on the asset in a directory named 'Cases'. This collection is in a .zip file, with a filename that starts with the date and time of the collection. If you expand the .zip file, you'll note that the collected data has been added while maintaining the directory tree structure. This is good news if you want or need to further investigate the collection in other forensic solutions.



Chain of Custody: Collection directory structure

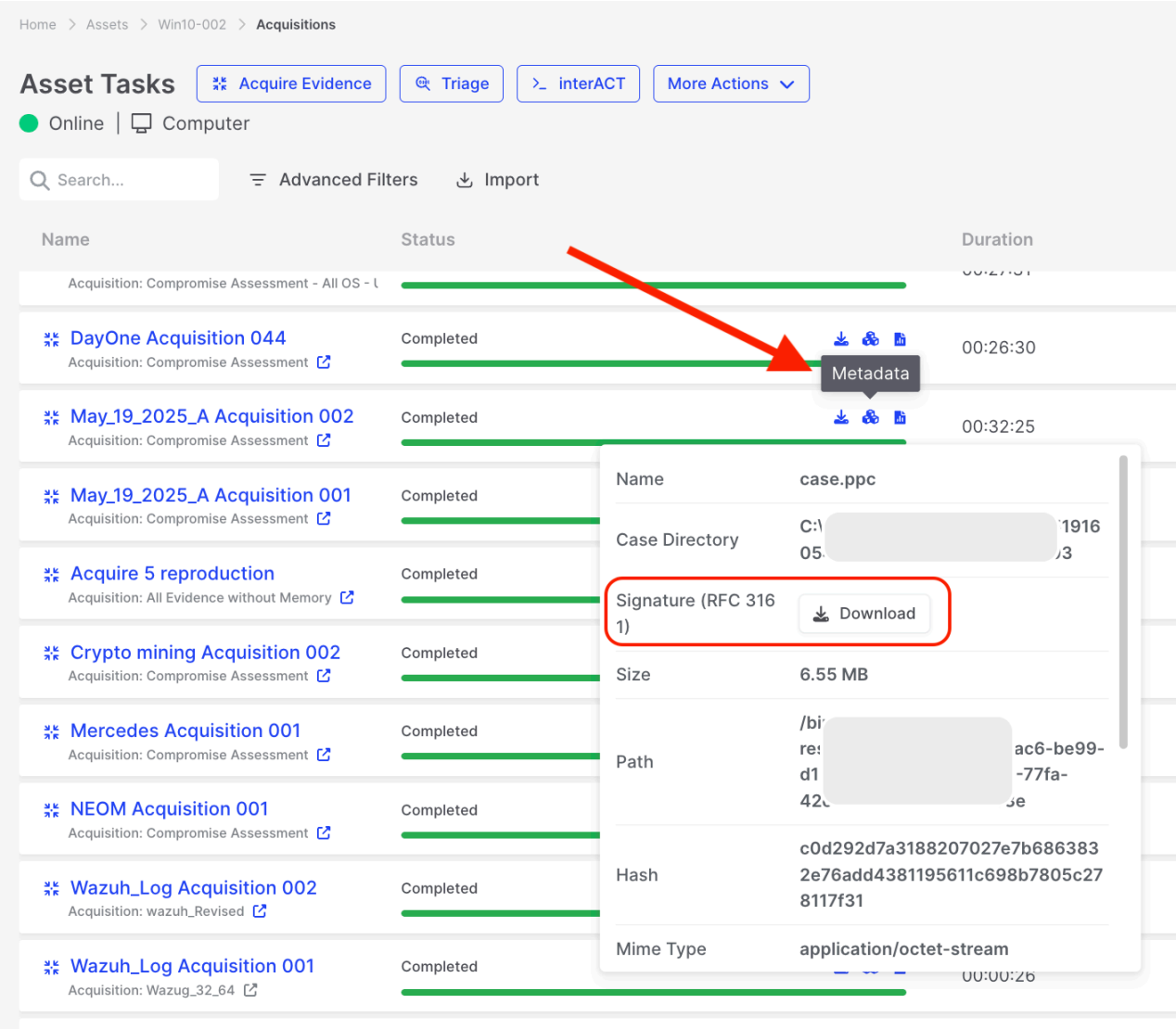
At the root of the collection shown above, you can see the Case.ppc file. This is another .zip container, and if you expand this, you can inspect the contents.

The hash values for the collected files are available in the Investigation Hub, from where they can be exported as a .csv file:



Chain of Custody: Hashes of acquired files are located in the Investigation Hub

With XDR Forensics, RFC 3161 timestamping is enabled by default. This means the hash value of your collection. ppc file is sent to the TSA, and their TST response is automatically saved as metadata for that collection in the XDR Forensics console. You can download and verify the TST from here at any time you or others need to.



Chain of Custody: Metadata button reveals RFC 3161 download

You can also disable the RFC3161 Timestamping functionality at any time via the XDR Forensics Settings > Features page.

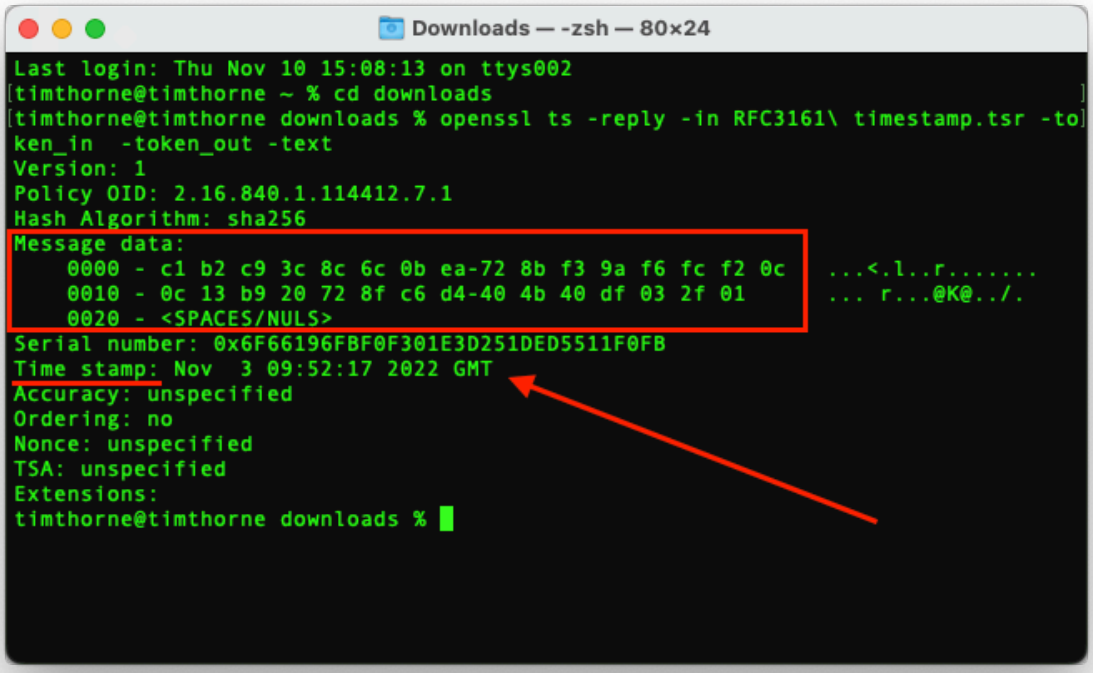
How to verify the .ppc via the RFC 3161 Timestamp Token

To verify the .ppc via RFC 3161, the first step is to download the TST from the metadata button in the XDR Forensics asset details > Task tab (as shown in the screenshot above, labeled "Metadata button reveals RFC 3161 download").

In the example below we have changed the name of the TST to ‘RFC3161 timestamp.tsr’ and saved it to the downloads folder.

We can then open a shell session and change the directory to the downloads folder.

To see the information in the TST Run: `openssl ts -reply -in RFC3161\ timestamp.tsr -token_in -token_out -text` and in the output, you'll see the hash of your .ppc and the Timestamp



Chain of Custody: openssl used to view ppc hash and timestamp

To verify this TST, we now need to download the root certificate from a TSA: https://cacerts.XXXX.com/XXXXtAssuredIDRootCA.crt.pem ↗.

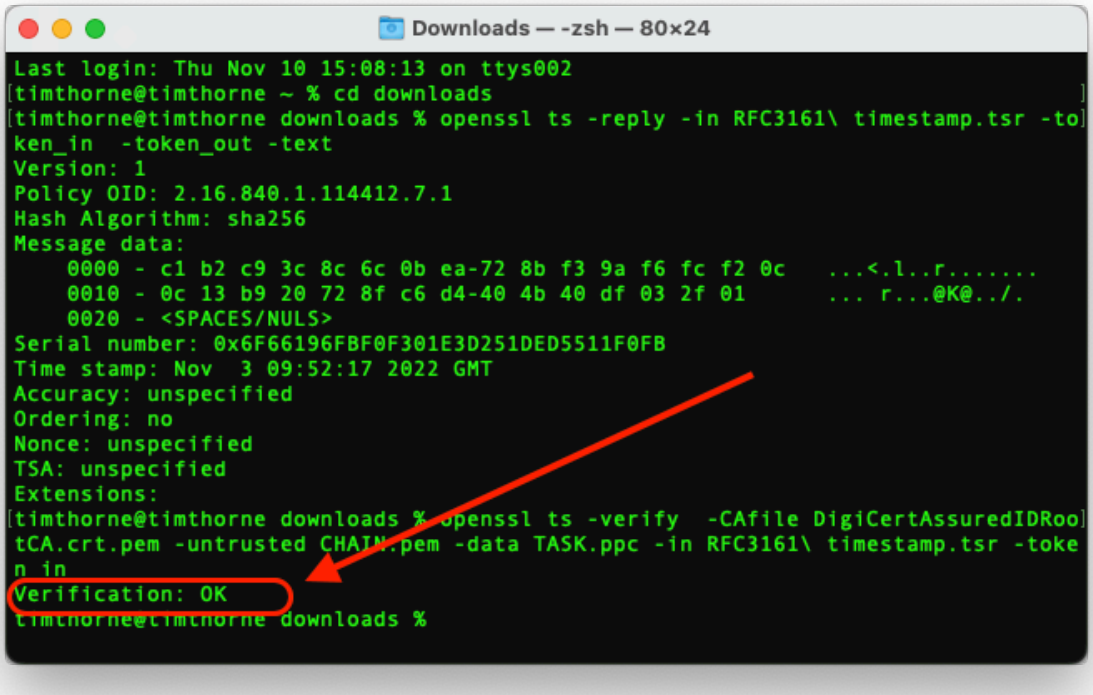
We will also need the following TSA certificates from the TSA server to build a 'chain certificate'. In this case, I took the content of each .cer file, in the order shown, and concatenated them into one file that I named 'CHAIN.pem'.

TSACertificate.cer ↗

XXXXTrustedG4RSA4096SHA256TimeStampingCA.cer ↗

XXXXTrustedRootG4.cer ↗

With all these files remaining in the same directory, I then ran the following command to verify the TST: `openssl ts -verify -CAfile XXXXAssuredIDRootCA.crt.pem -untrusted CHAIN.pem -data TASK.ppc -in RFC3161\ timestamp.tsr -token_in`

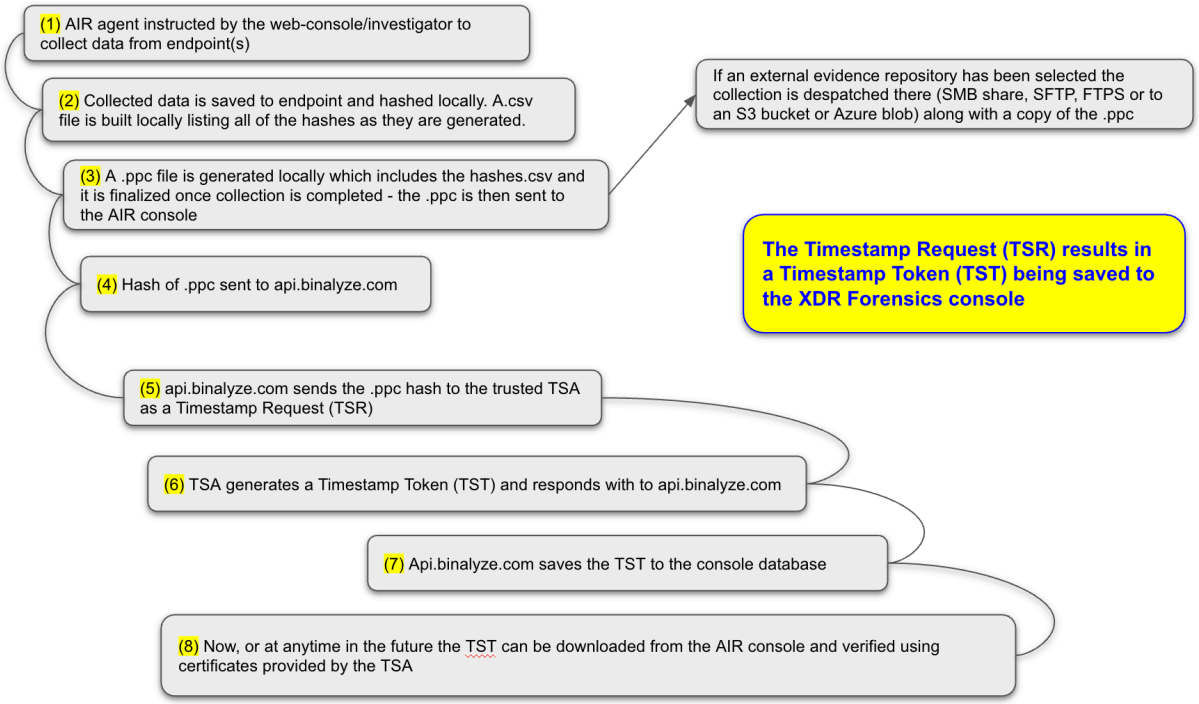


Chain of Custody: openssl with certificate chain used to verify TST

This simple verification ‘ok’ message confirms that the TST is correct, indicating that my data is sound and that it existed at the date and time shown by the timestamp

Conclusion - Robust best practice

Thanks to the RFC 3161 and SHA-256 hashing features of XDR Forensics, it’s now possible to prove that not only is your data content 100% intact but that it existed at a particular moment in time. So we can now be sure that we know exactly what was collected and when it was collected. In short, RFC 3161 provides immutable timestamping for an effective chain of custody to maintain forensic integrity.



Chain of Custody: Process flow to receive a TST in XDR Forensics

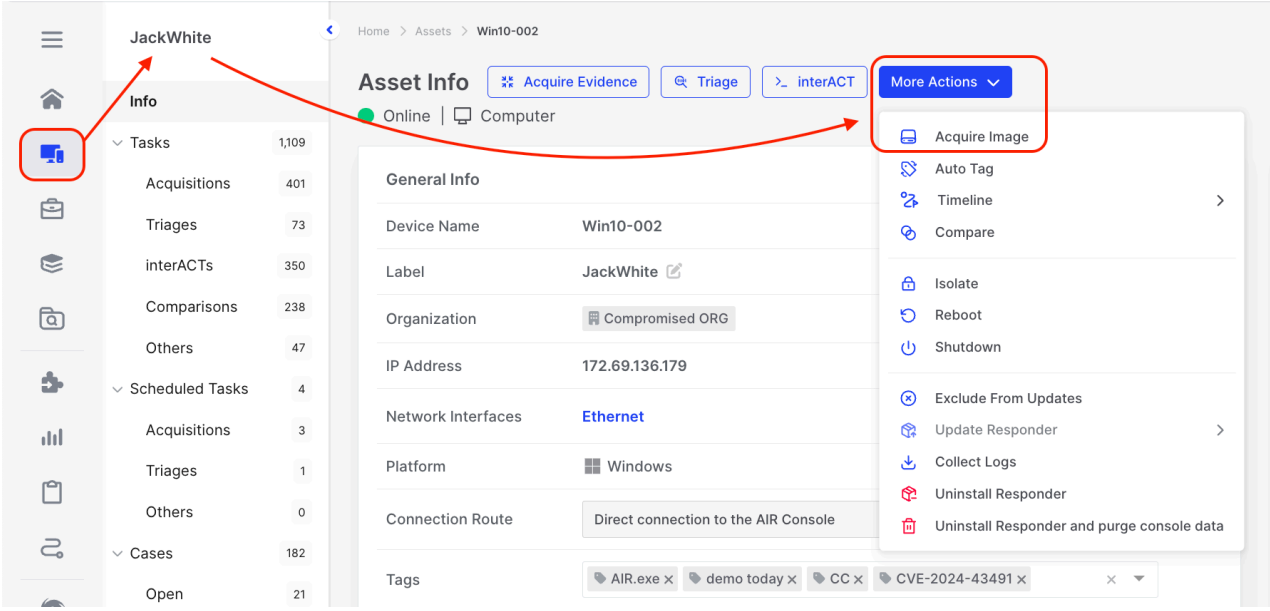
Disk and Volume Imaging

The acquisition of physical disk images and volume images can be done via an **Acquire Image Task** in the UI or by using commands in an **interACT session**.

In addition to NTFS and FAT, XDR Forensics also supports the logical imaging of ext4 and ext3 volumes, as well as physical disk imaging, which is possible from all operating systems supported by the XDR Forensics Responder.

> When performing forensic disk imaging on Mac devices with T2 or later chips, obtaining a physical disk image of APFS volumes is often ineffective. This is because the data on these disks is encrypted, and decryption is exclusively managed by the chip that originally encrypted the data. Consequently, decryption can only occur during the acquisition process using that specific chip.
>
> For most investigative purposes, a logical collection of files using XDR Forensics acquisition profiles typically provides sufficient information. This method, supported by XDR Forensics, enables investigators to access and analyze the file system and its contents efficiently, thereby bypassing the complexities associated with Apple Silicon APFS-encrypted physical disk images.

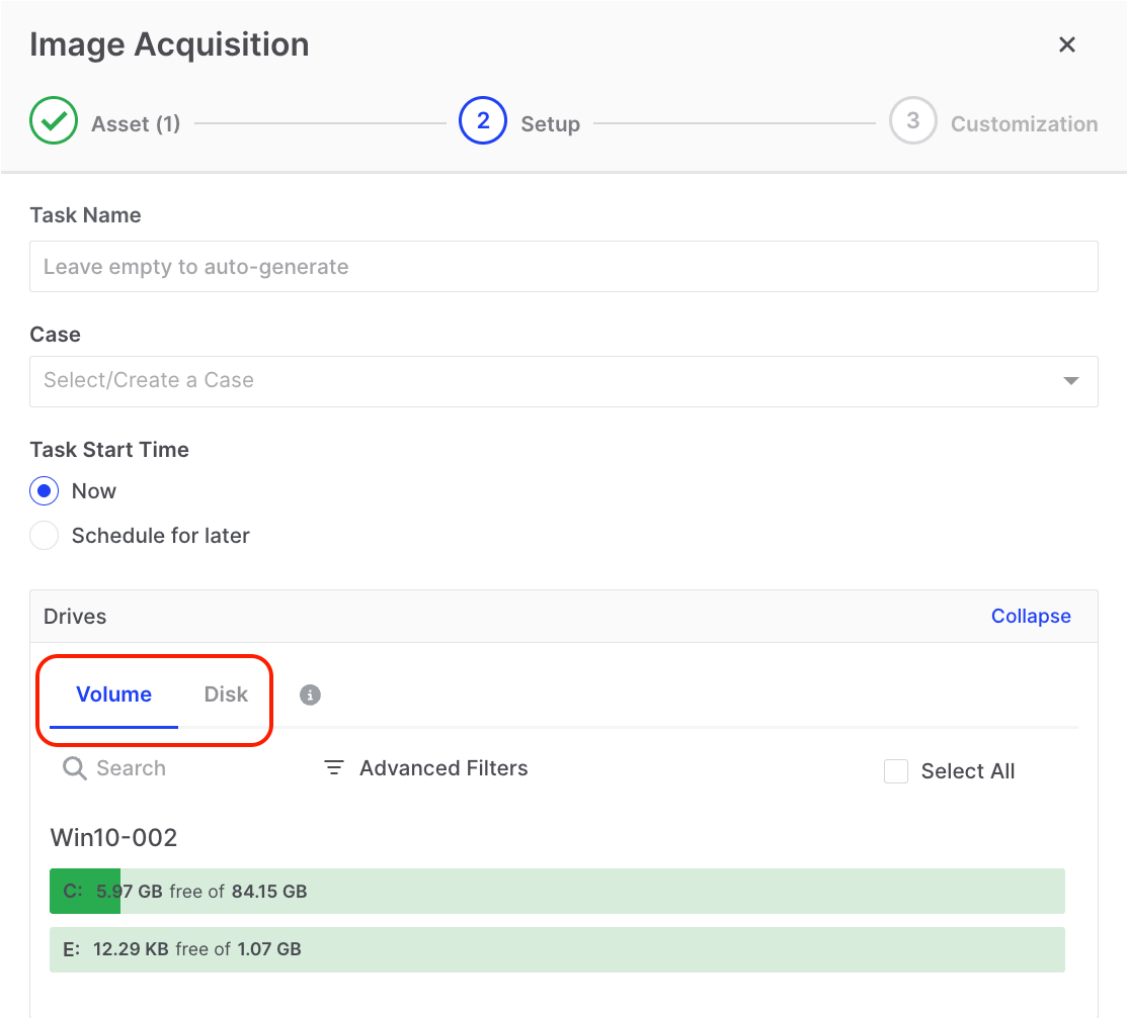
In the XDR Forensics UI you select Assets from the primary menu and then in the Asset Info window when you select the Asset Actions button a drop-down menu appears listing the actions that can be applied to that individual asset. **Acquire Image** is one such option:



Disk and Volume Imaging: Acquire an image from a single asset

The Acquire Image wizard will now walk you through the steps needed to take a forensic image from the asset:

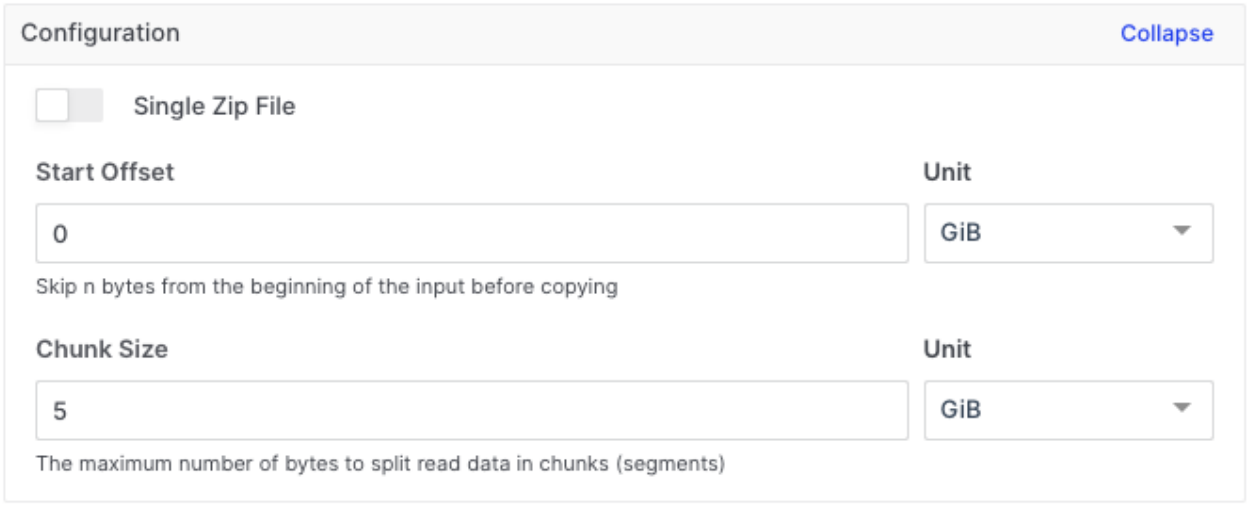
1. Choose a Task Name.
2. Select or create a case to which the image should be associated.
3. Choose **either** the Volume or Disk tab (note that the size is displayed, so you can ensure the Repository has enough free space to hold the collected image).
4. If there is more than one disk or volume, you can select the desired one by searching, filtering, or manually selecting it.



Disk and Volume Imaging: The Volume and Disk tabs

Having chosen what is to be imaged, you can now configure/set up the image file:

1. Select an Evidence Repository to which the image file can be saved.
2. Select your image format, RAW (dd) or EWF2 (Ex01), which is currently supported.
3. For RAW (dd) **only**, a toggle switch provides users with the option to enable or disable the consolidation of physical disk or volume image files into a single zip file, eliminating the need to split them into separate chunks.
4. For RAW (dd), if the 'single zip file' option is not toggled on, users will have the option to choose the size of image file chunks. If you want to use XDR Forensics's File Explorer to browse the image file, the image must be supplied to XDR Forensics from an SMB, SFTP, an Amazon S3 bucket, or Azure Blob Storage shared location, where it needs to be saved as **a single contiguous RAW file or an EWF file which can be segmented.** (product-platform/features/file-explorer/))
5. Users can also choose to skip a configurable number of bytes before starting the imaging process.



Disk and Volume Imaging: Configuration setting for the image files

In the 'Resource Limits' section, as with other XDR Forensics taskings, you can set limits on the network bandwidth used during the image acquisition process. Meanwhile, the 'Compression and Encryption' section provides options for conserving storage space and enhancing the security of the gathered evidence.

The output of your imaging task will be located in the evidence repository you selected when building the task in the wizard. The metadata associated with the acquisition will also be found there and this is explained here: Understanding errors documented in the metadata.yml file ↗

The acquired image can be investigated using the XDR Forensics File Explorer.

Imaging with interACT

interACT has an imaging command with several options/switches to allow users to read a disk or volume and write its contents out as a .dd file. As seen on the previous page, this can also be done from the XDR Forensics UI but remains here in interACT for those who prefer to image from the command line.

interACT Imaging Options

```
   --list, -l                          list available disk/volume
devices (default: false)
   --input value, -i value             read disk device from given
input
   --output value, -o value            write output file(s) to a
repository or directory. Use 'repository' to upload to evidence
repository set at session start
   --max-chunk-size value, -c value    maximum chunk size in bytes
or use suffix K,M,G. Use 0 to use dynamically calculated chunk size
(default: "512M")
   --chunk-retry-count value, -r value  number of retries for chunk
creation attempts when output is repository. If output is a
directory, this flag is ignored (default: 1024)
   --start-offset value, -s value      start offset in bytes or
use suffix K,M,G (default: "0")
   --max-chunk-count value, -n value   maximum number of chunks to
create. 0 value dynamically calculates the number of chunks
(default: 0)
   --block-size value, -b value        input block size in bytes
or use suffix K,M,G. This must be multiple of logical sector size.
Bigger values will result in faster reads but will skip the same
amount of data on each read if read error occurs. 1M is a good
value for most cases. (default: "1M")
   --file-prefix value, -p value       prefix for output file
name(s)
   --no-proxy                          bypass proxy if enabled
while transferring file to a repository (default: false)
   --bandwidth-limit value             maximum bandwidth limit in
bytes or use suffix K,M,G (default: "0")
   --help, -h                          show help
```

Here is an example of an imaging command in its simplest form:

```
image -i E: -o OutputFolder2
```

In this command, the `-i` flag is used to specify the input source for the `image` command.

Here's a breakdown of the command:

- `image`: This is the name of a command or script that is used to create an image (a copy) of a disk or volume.
- `-i E:`: This flag specifies the input source for the image creation process. In this case, `E:` represents a disk or volume identifier on the system. It indicates that the image creation process should target the contents of the disk or volume associated with the drive letter `E:`.
- `-o OutputFolder2`: This flag specifies the output destination for the image file. The image file generated by the command will be stored in the `OutputFolder2` directory.

Imaging output and the metadata.yml file

To inspect the results of the command shown above, `image -i E: -o OutputFolder2,` we can navigate to the folder using interACT and list the contents as shown below:

```
air@JackWhite:C:\Users $ cd OutputFolder2
C:\Users\OutputFolder2
air@JackWhite:C:\Users\OutputFolder2 $ ls
C:\Users\OutputFolder2:
ModTime                        Size  Name
2024-03-20T09:29:30          670 kB  image.001.zip
2024-03-20T09:29:34          522 kB  image.002.zip
2024-03-20T09:29:34           541 B  metadata.yml
```

Imaging with interACT: Output and associated metadata file

In this case, we see that there are two image chunks; image.001.zip and image.002.zip, along with a file named metadata.yml. This file exists in your output folder even when you use the XDR Forensics UI to image a disk or volume.

This metadata file can be read in the shell with the 'cat' command. It provides information about your image including the source, imaging start and end times, size, and hash values:

```
air@JackWhite:C:\Users\OutputFolder2 $ cat metadata.yml
Hostname: Win10-002
Source: '\\.\E:'
Target: C:\Users\OutputFolder2
StartTime: 2024-03-20T09:29:27.1737139-07:00
BlockSize: 1048576
StartOffset: 0
Duration: 7.6099484s
ChunkSizeInBytes: 536870912
BytesRead: 1073737728
ReadDuration: 451.5243ms
SeekDuration: 0s
BytesWritten: 1073737728
NumberOfChunks: 2
WriteDuration: 7.1373425s
Compression: true
Encryption: false
Hash:
    MD5: fca9b2842db5decdf894327adf4a1ed9
    SHA1: ad6a937e97fa73e64d6d0fdabb0a357ca01c9df4
    SHA256: eeb5961f8f83ae3d70495831da307d429c6ffe881364c5396da76408a8ad8224
air@JackWhite:C:\Users\OutputFolder2 $
```

Imaging with interACT: Contents of metadata.yml

Understanding errors documented in the metadata.yml file

From time to time all imaging tools will have issues with areas of the the disk that can not be read. In such cases, XDR Forensics will report errors in the metadata.yml file and they will be recorded as shown below:

```
metadata:
    Hostname: Win10-002
    Source: '\\.\E:'
    Target: C:\Users\OutputFolder2
    StartTime: 2024-03-19T19:58:21.2390559-07:00
    BlockSize: 1048576
    StartOffset: 0
    Duration: 7.734205s
    ChunkSizeInBytes: 536870912
    BytesRead: 1073737728
    ReadDuration: 466.9625ms
    SeekDuration: 0s
    BytesWritten: 1073737728
    NumberOfChunks: 2
    WriteDuration: 7.249291s
    Compression: true
    Encryption: false
    Hash:
        MD5: fca9b2842db5decdf894327adf4a1ed9
        SHA1: ad6a937e97fa73e64d6d0fdabb0a357ca01c9df4
        SHA256:
eeb5961f8f83ae3d70495831da307d429c6ffe881364c5396da76408a8ad8224
ReadErrorTable:
    Errors:
        0: error-1
        1: error-2
    Regions:
        - Offset: 0
          Size: 1048576
          RefError: 0
        - Offset: 2097152
          Size: 1048576
          RefError: 1
```

This imaging metadata report outlines the process and outcome of an imaging operation carried out in XDR Forensics. The report provides details about the operation, including the source, target, data transfer metrics, and errors encountered. Let's break down the key parts and interpret the errors mentioned in the report:

Basic Operation Details

- **Hostname**: Win10-002 indicates the machine name where the operation was performed.
- **Source**: '\.\E:' shows that imaging was done from a device mounted at E: (likely a disk drive).
- **Target**: C:\Users\OutputFolder2 is where the imaged data was written.
- **StartTime**: The operation started on March 19, 2024, at 19:58:21 local time.
- **Duration**: It took approximately 7.73 seconds to complete.
- **Compression**: Enabled, indicating the data was compressed during the imaging process.
- **Encryption**: Not used during this imaging operation.

Data Transfer Metrics

- **BytesRead and BytesWritten**: Both are 1,073,737,728 bytes, indicating that a bit over 1 GB of data was read from the source and written to the target.
- **NumberOfChunks**: 2 chunks of data were processed, aligning with the bytes read/written and chunk size.
- **ChunkSizeInBytes**: Each chunk was 536,870,912 bytes, about 512 MB, which fits the total data size indicating two chunks were necessary.
- **ReadDuration and WriteDuration**: Reading took under half a second, whereas writing took the majority of the operation time (about 7.25 seconds).

Errors and Their Implications

The `ReadErrorTable` section is particularly noteworthy as it outlines issues encountered during the read operation:

- **Errors Listed**: Two errors, `error-1` and `error-2`, were encountered during the imaging process.
- **Regions Affected**:
  - The first error occurred at the very beginning of the read operation (Offset: 0), affecting 1,048,576 bytes (1 MB).
  - The second error occurred after skipping the next 1 MB chunk (notably absent from the errors), affecting the third 1 MB segment of data (Offset: 2,097,152).

Interpreting the Errors

- The presence of read errors in specific regions suggests issues with the source device at those locations. This could be due to bad sectors, physical damage, or corruption within the disk's storage.
- The operation continued despite these errors, which is common in forensic imaging processes where the goal is to recover as much data as possible, even in the presence of damaged or inaccessible areas.
- The absence of errors for the second 1 MB segment (from 1,048,576 to 2,097,152 bytes) indicates that not all regions of the source had issues, highlighting the localized nature of the problems.

Scheduling Tasks

Proactive DFIR and automated threat analysis by scheduling XDR Forensics Evidence Collections

Scheduling tasks in XDR Forensics not only enables the automation of acquisitions and DRONE analysis, but it transforms XDR Forensics into a proactive DFIR platform. By setting up scheduled tasks for regular collections and automated DRONE analysis, XDR Forensics can proactively identify issues that might well go unnoticed by other security systems.

Instead of waiting for alerts from external sources, XDR Forensics takes the initiative to regularly collect and analyze assets according to a predefined schedule and acquisition profile. This proactive approach allows organizations to stay ahead of potential threats and vulnerabilities by detecting issues early on, even before they manifest as security incidents.

By integrating task scheduling into security operations, organizations can enhance their defense strategies and strengthen their overall security posture. Additionally, it ensures that the 'best evidence' is automatically acquired and forensically preserved, facilitating further investigation when needed.

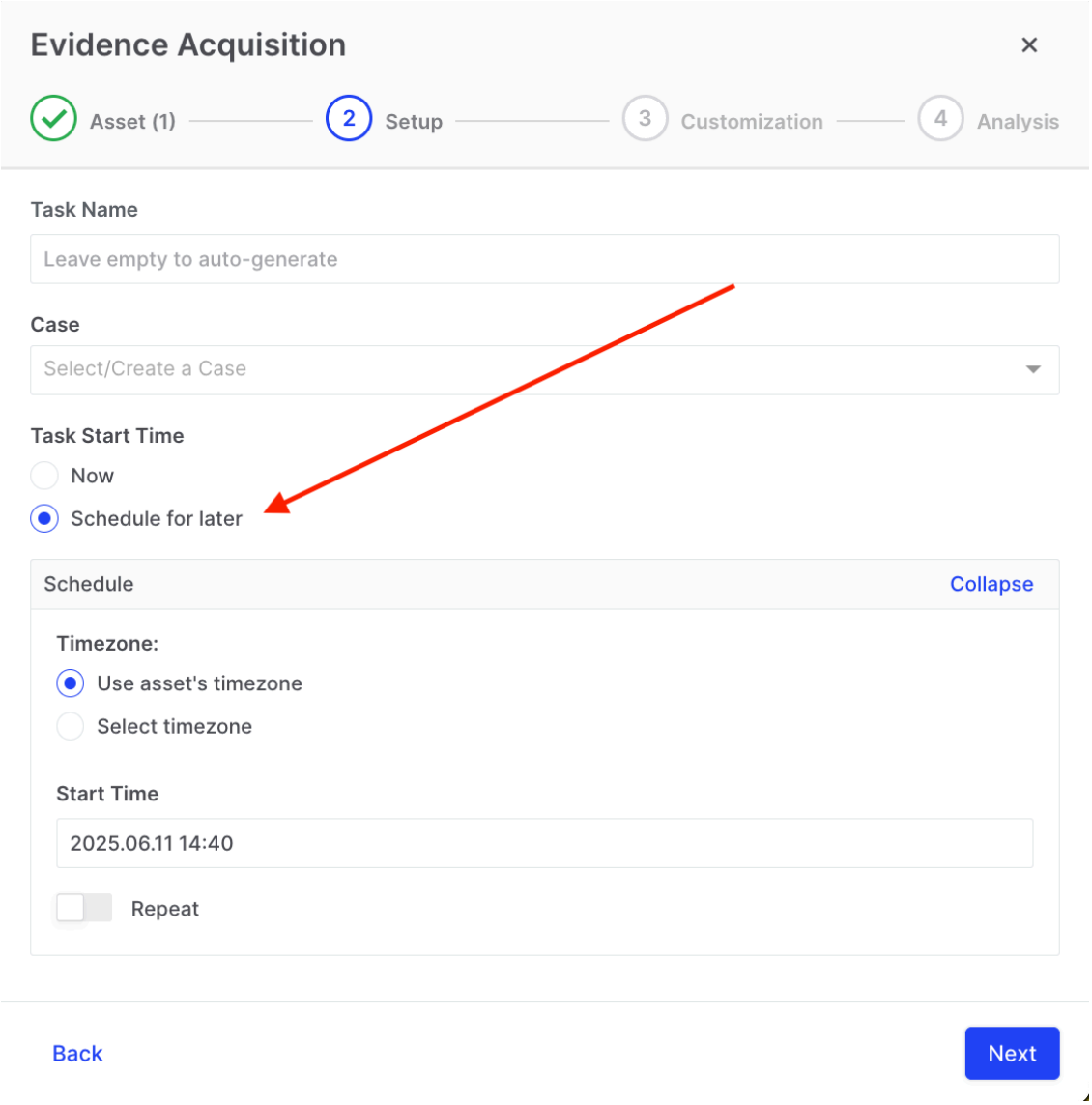
Investigators simply use the tasking wizard to schedule tasks for the following activities:

- Evidence collections.
- Triage/Threat Hunting.
- Disk and Volume Imaging.
- Auto Asset Tagging.

To set up a scheduled task, please follow the steps below:

Step 1 - Assets

1. Choose the Asset(s) on which you wish to schedule tasks - the Bulk Action Bar will be available if you choose more than one asset.
2. Create a Task Name if required.
3. Allocate the task to a case.
4. Select 'Schedule for later'

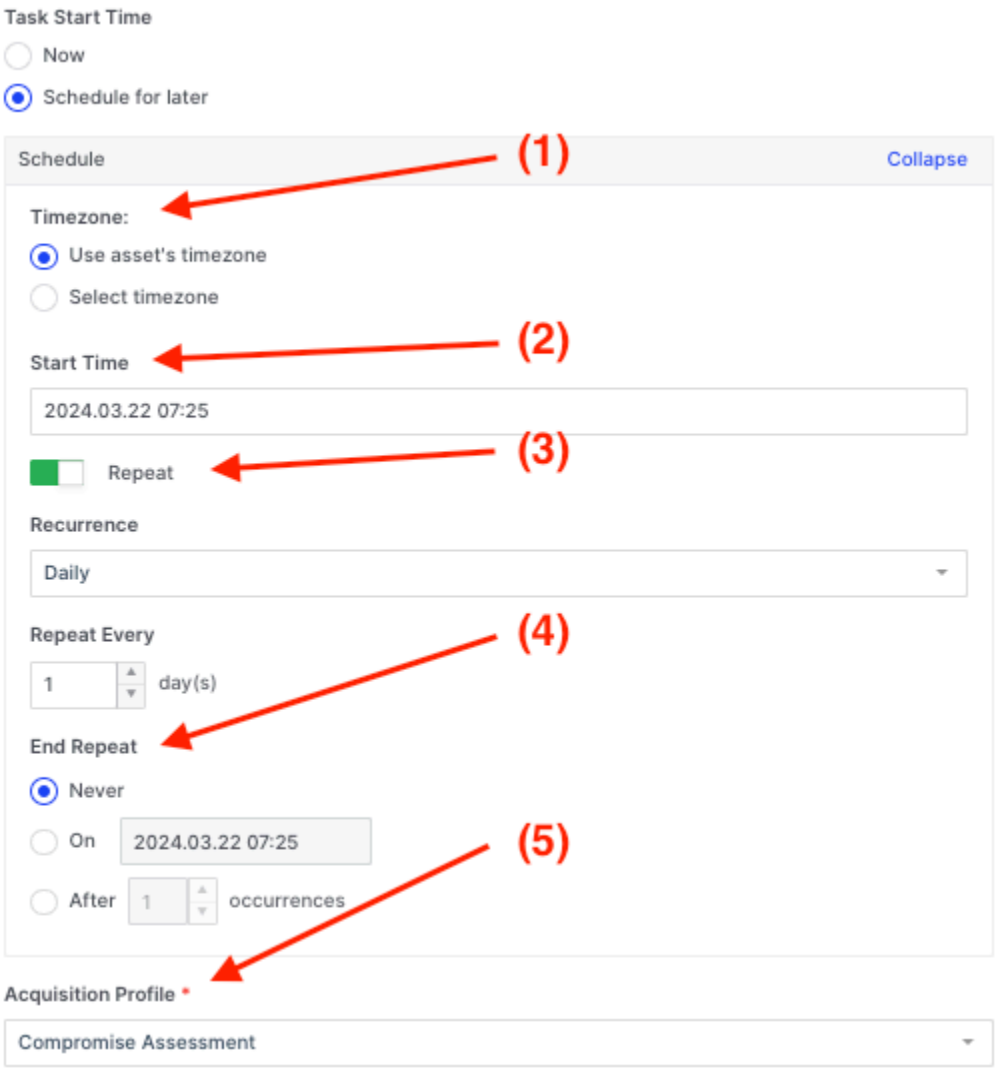


Scheduling Tasks: Schedule for later

Step 2 - Setup

1. Select the time zone that determines when the task is executed. You have the flexibility to execute the task in the local time zone of each selected asset or simultaneously across all assets by choosing a single time zone.
2. Select a start date and time for the task.
3. If required, toggle the Repeat switch on and set the cadence or recurrence rate.
4. Select when or whether you want to end the schedule.
5. Lastly, choose the acquisition profile you wish to apply to the scheduled task.

> **Note:** Scheduled tasks cannot be repeated within a "Case". This is because there is no destination for the task results when a Case is closed, leaving them without a location to be sent to.



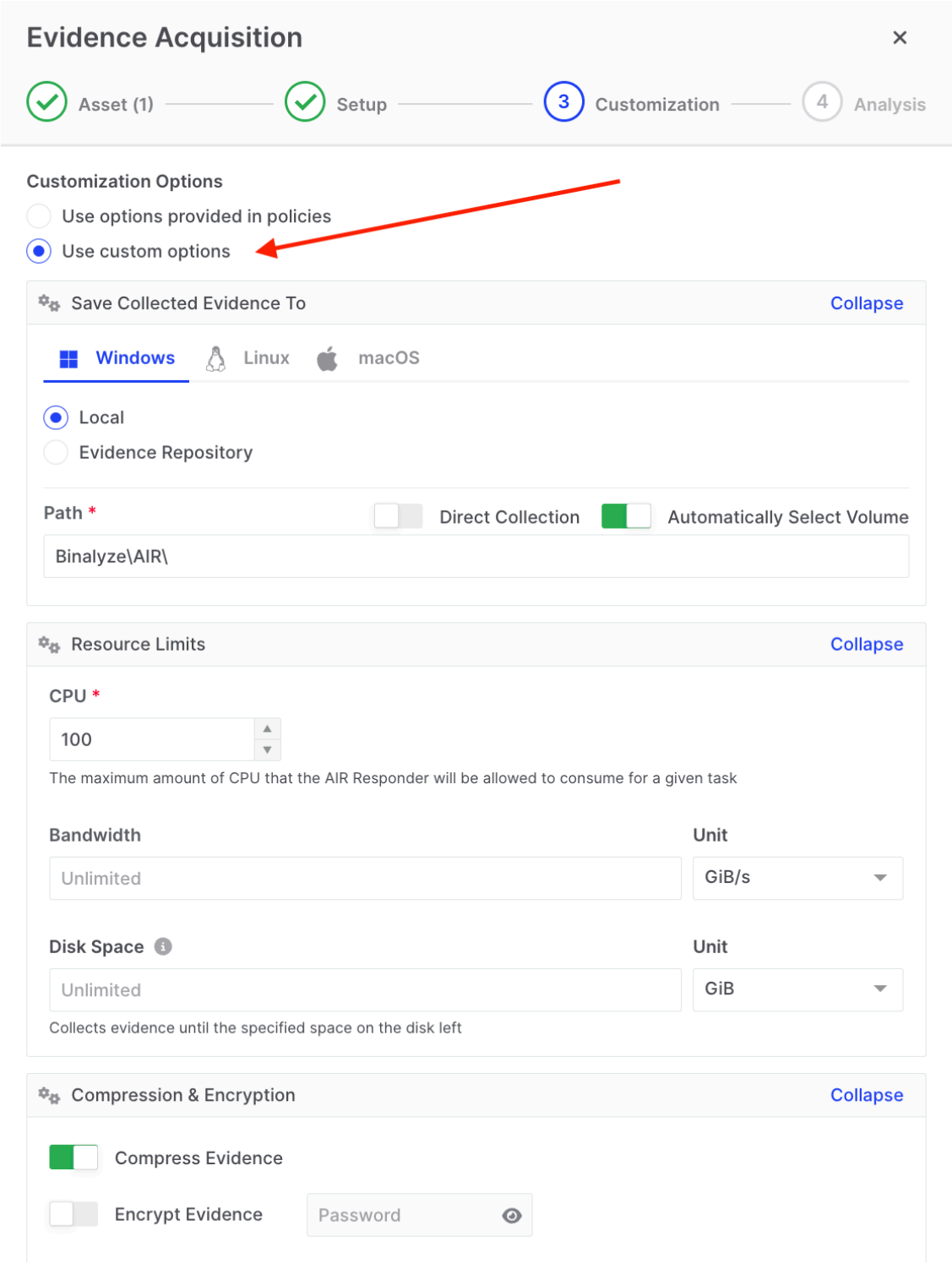
Scheduling Tasks: Setting up the scheduled task

User Privileges for Task Scheduling:

- XDR Forensics administrators can now restrict users from scheduling tasks or editing existing ones:
  - **Schedule Task**: Enables users to "Schedule for later." Without this privilege, this option is disabled, and a tooltip explains the restriction.
  - **Update Scheduled Task**: Allows users to edit scheduled tasks. If this privilege is not granted, the "Edit" button is disabled, accompanied by an explanatory tooltip.

Step 3 - Customization

- This step allows you to use the policies already set as an organizational policy or, if you have the necessary privileges, make changes to:
- Where the collected evidence is to be saved.
- Apply a Resource limit to the task assignment to reduce potential impacts on the asset.
- Enable compression and encryption to be applied to the collected evidence.

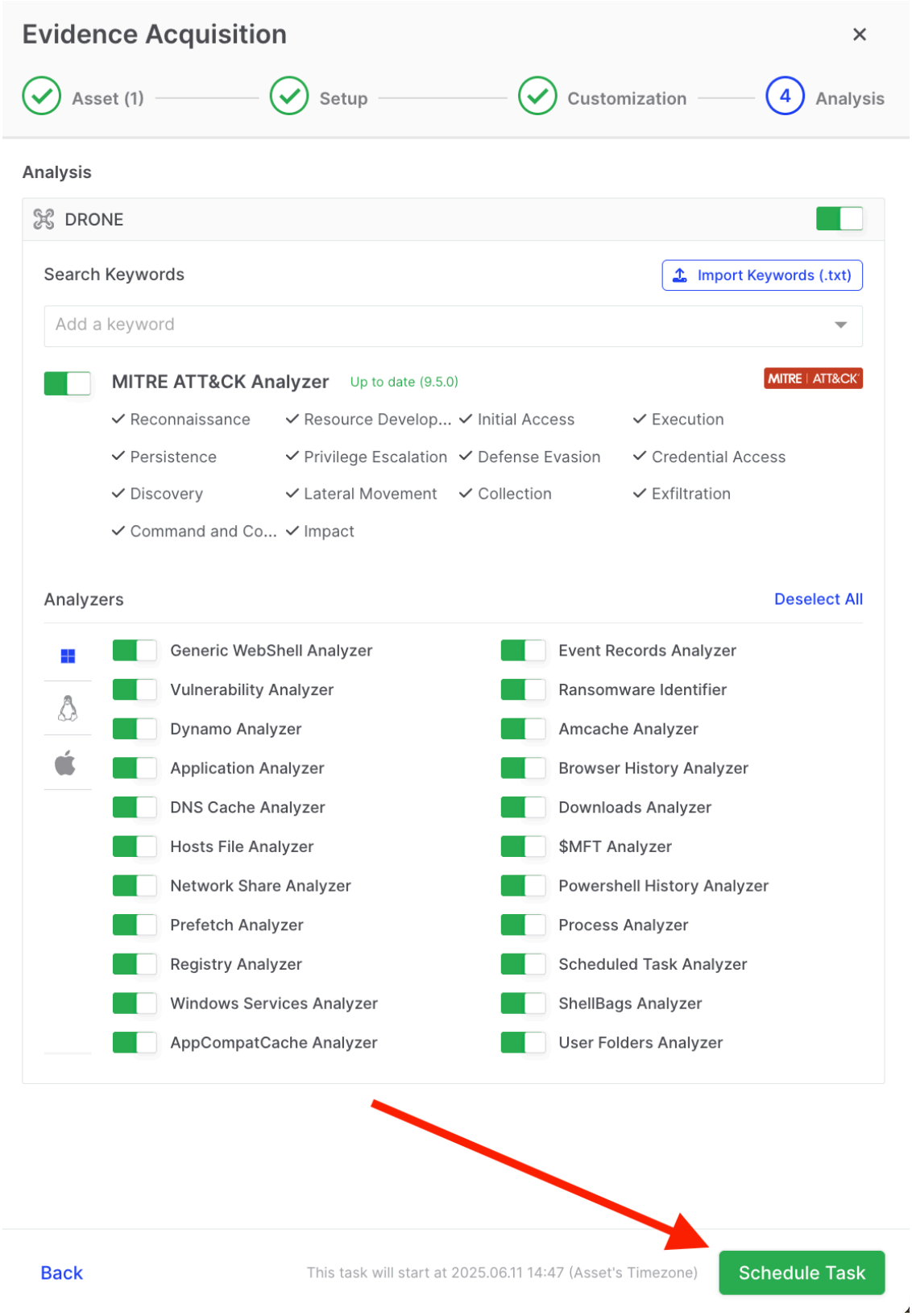


Scheduling Tasks: Customization of options

Step 4 - Analysis

In this final step, you can enable or disable the DRONE and MITRE ATT&CK Analyzers.

We highly recommend keeping both analyzers active as they have minimal impact on resources. The MITRE ATT&CK analyzer runs on live assets and, when combined with other analyzers, facilitates immediate identification of potentially compromised assets. This allows for efficient prioritization of investigative efforts.



Scheduling Tasks: Toggle options for DRONE

The results of all tasks, along with their associated reports, are accessible via the Investigation Hub.

Links to these reports within the Investigation Hub can be found in the following three locations:

1. Tasks, via the main menu.
2. The page for the individual Asset > Acquisitions.
3. And finally, Case Acquisitions, if the task is sent to a Case.

Tasks scheduled through the console will execute as planned. However, if an asset is offline at the scheduled time, it will automatically receive and carry out the task upon its next connection.

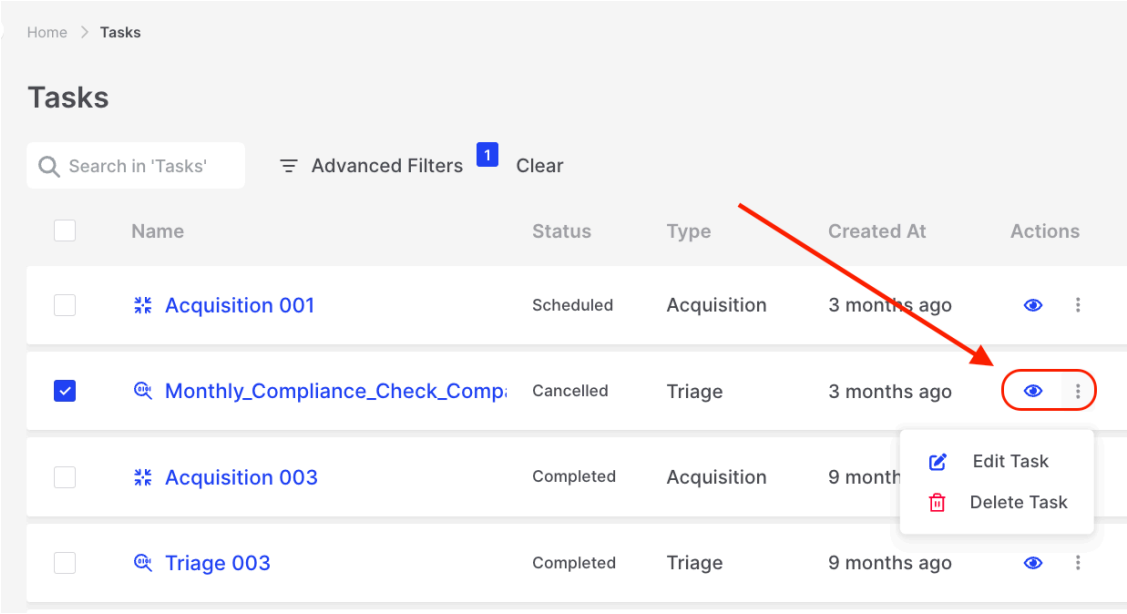
How to edit existing Scheduled Tasks

Managing scheduled tasks has never been easier. The Edit Scheduled Task feature allows you to modify existing scheduled tasks, eliminating the need to cancel and reconfigure them.

You can easily add or remove assets without needing to restart the task, saving valuable time and enhancing workflow efficiency. After selecting the assets, you can then update the task setup, customize options, and manage follow-up actions, streamlining the task management process for a smoother and faster workflow.

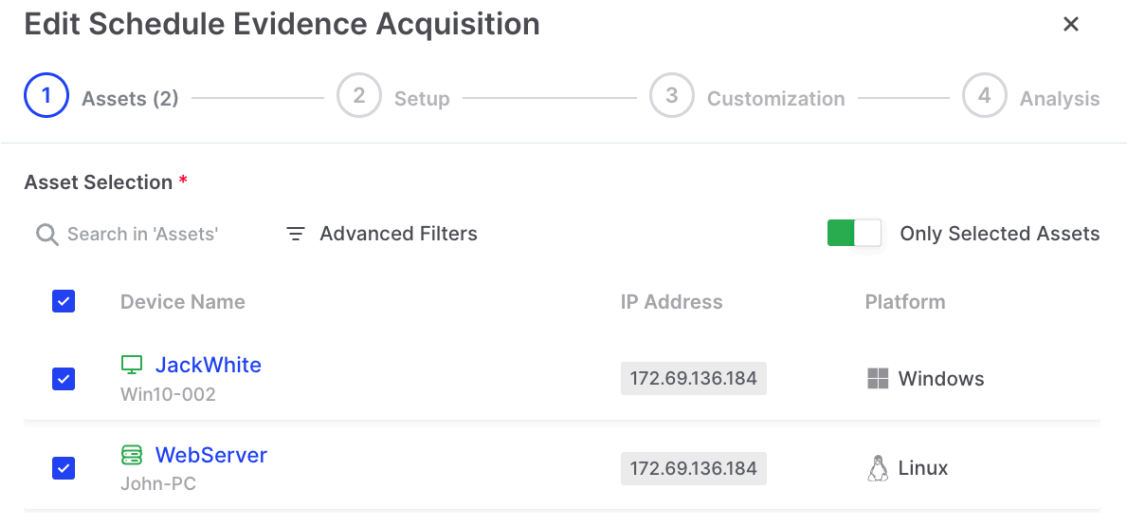
To modify a scheduled task, go to your Task listings page and filter by Status > Scheduled to display only the scheduled tasks:

From the filtered results, selecting the ‘eye’ icon presents you with the Task Details and the ellipse with Edit or Delete Task options:



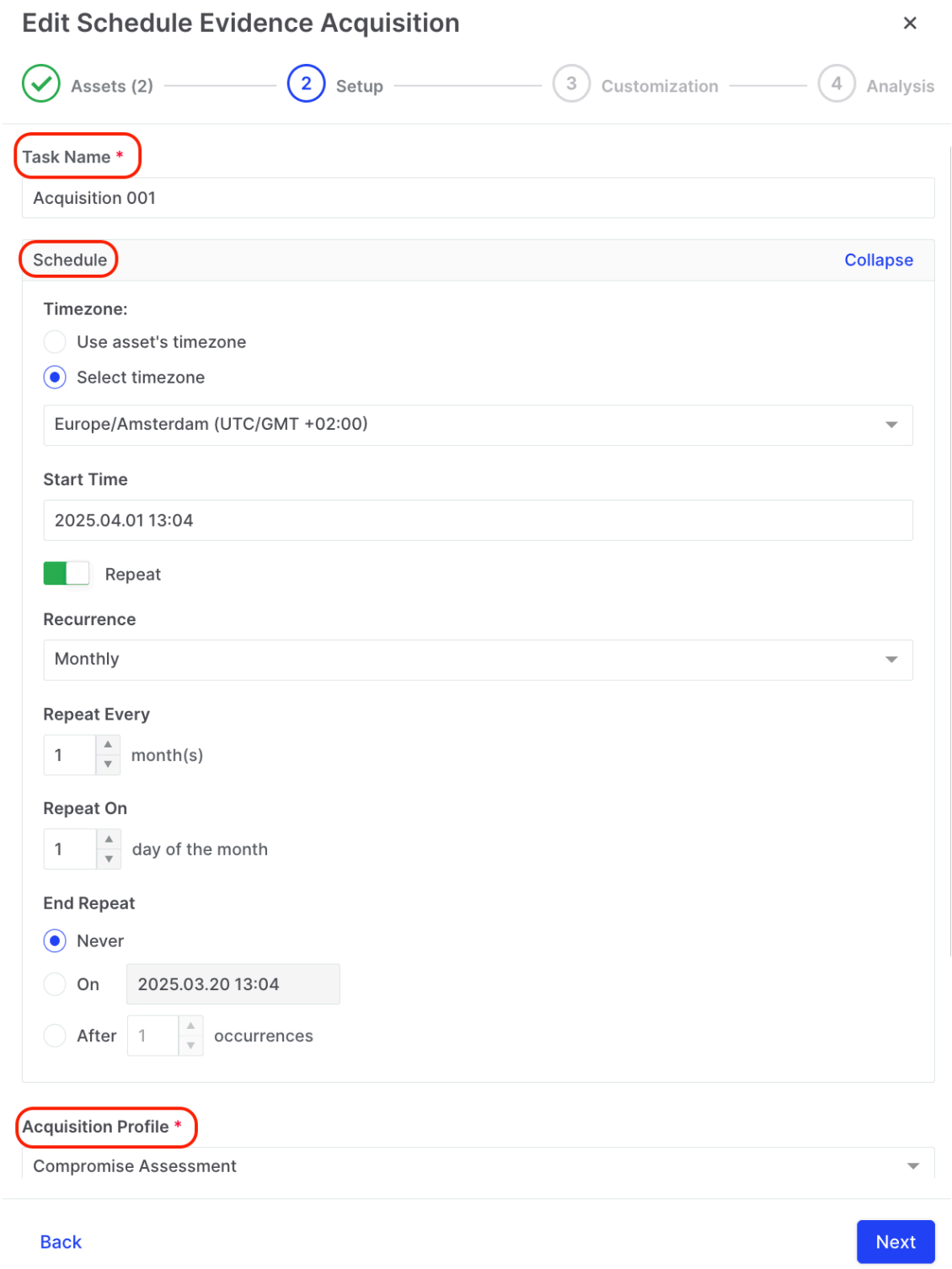
Scheduling Tasks: Edit or Delete Task options

The Edit Scheduled Task Wizard will now open, allowing the user to toggle off the "Only Selected Assets" switch (as shown below). This now reveals all other available assets that can be added to the scheduled task:



Scheduling Tasks: Select the assets requiring an edit to scheduled task

Step 2, the Setup, allows you to edit the Task Name, the Schedule, and even the acquisition profile to be used:



Scheduling Tasks: Edits

Steps 3 and 4, Customization and Analysis, are fully configurable, allowing you to edit the scheduled task as needed.

Supported Evidence

These pages categorize the supported evidence and artifacts by OS, indicating whether each item is parsed and presented in the Investigation Hub and/or if the associated file is collected.

- Windows Collections
- macOS Collections
- Linux Collections
- IBM AIX Collections

The table below provides a count of the currently supported evidence and artefact items

Collection Type:	File Count
Windows artifact	119
Windows evidence	191
macOS artifact	27
macOS evidence	175
Linux artifact	25
Linux evidence	135
AIX artifact	7
AIX evidence	19
Grand Total	**698**

IBM AIX Collections

XDR Forensics supports the following IBM AIX Evidence and Artifacts

IBM AIX Evidence List

1	System	Cron Jobs	Collect cron jobs
2	System	ULimit Information	Collect ulimit information
3	Disk	Mounts	Collect mounts
4	File System	File System Enumeration	Dump file and folder information.
5	Processes	Processes	Collect process list
6	Users	User Groups	Collect user group list
7	Users	Users	Collect user list
8	SSH	SSH Known Hosts	Collect SSH known hosts
9	SSH	SSH Authorized Keys	Collect SSH authorized keys
10	SSH	SSH Configs	Collect SSH configurations
11	SSH	SSHD Configs	Collect SSHD configurations
12	Network	Hosts	Collect hosts
13	Network	DNS Resolvers	Collect DNS resolvers
14	Other Evidence	YUM Sources	Collect YUM sources
15	Other Evidence	YUM History	Collect YUM history
16	Other Evidence	SUID Binaries	Collect SUID binaries
17	Other Evidence	Shell History	Collect shell history

18	Other Evidence	System Artifacts	Collect system artifacts (Files of collected evidence. For example: /etc/passwd file)
19	Other Evidence	Log Files	Collect log files under /var/log/

IBM AIX Artifact List

1	Server	MySQL Logs	Collect MySQL Logs
2	Server	SSH Server Logs	Collect SSH Server Logs
3	Server	DHCP Server Logs	Collect DHCP Server Logs
4	System	System Logs	Collect System Logs
5	System	Auth Logs	Collect Auth Logs
6	System	Boot Logs	Collect Boot Logs
7	System	Mail Logs	Collect Mail Logs

Linux Collections

XDR Forensics supports the following Linux Evidence and Artifacts

Linux Evidence List

1	System	System Controls	Collect system controls
2	System	Cron Jobs	Collect cron jobs
3	System	AppArmor Profiles	Collect AppArmor profiles
4	System	ULimit Information	Collect ulimit information
5	System	Kernel Modules	Collect kernel modules
6	System	Lock Files	Collect lock files
7	System	Systemctl Services	Collect Systemctl Running Services
8	Disk	Block Devices	Collect block devices
9	Disk	Fstab	Collect fstab configuration
10	Disk	Mounts	Collect mounts
11	Disk	NFS Exports	Collect NFS exports
12	File System	File System Enumeration	Dump file and folder information.
13	Processes	Processes	Collect process list
14	Processes	Process Open Files	Collect process open files information
15	Memory	Shared Memory	Collect shared memory
16	Memory	Memory Map	Collect memory map
17	Memory	Swaps	Collect swap info
18	Memory	RAM Image	Create an image of RAM

19	Browser	Default Browser	Collect Default Browser
20	Browser	Chrome Cookies	Collect Chrome Cookies
21	Browser	Chromium Cookies	Collect Chromium Cookies
22	Browser	Edge Cookies	Collect Edge Cookies
23	Browser	Opera Cookies	Collect Opera Cookies
24	Browser	Vivaldi Cookies	Collect Vivaldi Cookies
25	Browser	Brave Cookies	Collect Brave Cookies
26	Browser	Chrome Bookmarks	Collect Chrome Bookmarks
27	Browser	Chromium Bookmarks	Collect Chromium Bookmarks
28	Browser	Edge Bookmarks	Collect Edge Bookmarks
29	Browser	Opera Bookmarks	Collect Opera Bookmarks
30	Browser	Vivaldi Bookmarks	Collect Vivaldi Bookmarks
31	Browser	Brave Bookmarks	Collect Brave Bookmarks
32	Browser	Chrome User Profiles	Collect Chrome User Profiles
33	Browser	Chromium User Profiles	Collect Chromium User Profiles
34	Browser	Edge User Profiles	Collect Edge User Profiles

35	Browser	Opera User Profiles	Collect Opera User Profiles
36	Browser	Vivaldi User Profiles	Collect Vivaldi User Profiles
37	Browser	Brave User Profiles	Collect Brave User Profiles
38	Browser	Chrome Extensions	Collect Chrome Extensions
39	Browser	Firefox Extensions	Collect Firefox Extensions (Addons)
40	Browser	Chrome Local Storage	Collect Chrome Local Storage
41	Browser	Chromium Local Storage	Collect Chromium Local Storage
42	Browser	Edge Local Storage	Collect Edge Local Storage
43	Browser	Opera Local Storage	Collect Opera Local Storage
44	Browser	Vivaldi Local Storage	Collect Vivaldi Local Storage
45	Browser	Brave Local Storage	Collect Brave Local Storage
46	Browser	Dump Chrome Indexed DB	Dump Chrome Indexed DB
47	Browser	Dump Chromium Indexed DB	Dump Chromium Indexed DB
48	Browser	Dump Edge Indexed DB	Dump Edge Indexed DB
49	Browser	Dump Opera Indexed DB	Dump Opera Indexed DB
50	Browser	Dump Vivaldi Indexed DB	Dump Vivaldi Indexed DB

51	Browser	Dump Brave Indexed DB	Dump Brave Indexed DB
52	Browser	Chrome Web Storage	Collect Chrome Web Storage
53	Browser	Chromium Web Storage	Collect Chromium Web Storage
54	Browser	Edge Web Storage	Collect Edge Web Storage
55	Browser	Opera Web Storage	Collect Opera Web Storage
56	Browser	Vivaldi Web Storage	Collect Vivaldi Web Storage
57	Browser	Brave Web Storage	Collect Brave Web Storage
58	Browser	Chrome Form History	Collect Chrome Form History
59	Browser	Chromium Form History	Collect Chromium Form History
60	Browser	Edge Form History	Collect Edge Form History
61	Browser	Opera Form History	Collect Opera Form History
62	Browser	Vivaldi Form History	Collect Vivaldi Form History
63	Browser	Brave Form History	Collect Brave Form History
64	Browser	Chrome Thumbnails	Collect Chrome Thumbnails
65	Browser	Chromium Thumbnails	Collect Chromium Thumbnails
66	Browser	Edge Thumbnails	Collect Edge Thumbnails

67	Browser	Opera Thumbnails	Collect Opera Thumbnails
68	Browser	Vivaldi Thumbnails	Collect Vivaldi Thumbnails
69	Browser	Brave Thumbnails	Collect Brave Thumbnails
70	Browser	Chrome Favicons	Collect Chrome Favicons
71	Browser	Chromium Favicons	Collect Chromium Favicons
72	Browser	Edge Favicons	Collect Edge Favicons
73	Browser	Opera Favicons	Collect Opera Favicons
74	Browser	Vivaldi Favicons	Collect Vivaldi Favicons
75	Browser	Brave Favicons	Collect Brave Favicons
76	Browser	Chrome Login Data	Collect Chrome Login Data
77	Browser	Chromium Login Data	Collect Chromium Login Data
78	Browser	Edge Login Data	Collect Edge Login Data
79	Browser	Opera Login Data	Collect Opera Login Data
80	Browser	Vivaldi Login Data	Collect Vivaldi Login Data
81	Browser	Brave Login Data	Collect Brave Login Data
82	Browser	Chrome Sessions	Collect Chrome Sessions

83	Browser	Chromium Sessions	Collect Chromium Sessions
84	Browser	Brave Sessions	Collect Brave Sessions
85	Browser	Edge Sessions	Collect Edge Sessions
86	Browser	Opera Sessions	Collect Opera Sessions
87	Browser	Vivaldi Sessions	Collect Vivaldi Sessions
88	Browser	Chrome Browsing History	Collect visited URLs from Google Chrome
89	Browser	Firefox Browsing History	Collect visited URLs from Mozilla Firefox
90	Browser	Chromium Browsing History	Collect visited URLs from Chromium
91	Browser	Edge Browsing History	Collect visited URLs from Edge
92	Browser	Opera Browsing History	Collect Visited URLs from Opera
93	Browser	Vivaldi Browsing History	Collect visited URLs from Vivaldi
94	Browser	Brave Browsing History	Collect visited URLs from Brave
95	Browser	Chrome Downloads	Collect Chrome Downloads
96	Browser	Chromium Downloads	Collect Chromium Downloads
97	Browser	Firefox Downloads	Collect Firefox Downloads
98	Browser	Brave Downloads	Collect Brave Downloads

99	Browser	Edge Downloads	Collect Edge Downloads
100	Browser	Opera Downloads	Collect Opera Downloads
101	Browser	Vivaldi Downloads	Collect Vivaldi Downloads
102	Browser	Firefox Cookies	Collect Firefox Cookies
103	Users	User Groups	Collect user group list
104	Users	Users	Collect user list
105	Users	Last Access	Collect last access records
106	Users	Logged Users	Collect logged user list
107	Users	Shadow	Collect shadow content
108	Users	Sudoers	Collect sudoers
109	Users	Failed Login Attempts	Collect fail login attempts
110	SSH	SSH Known Hosts	Collect SSH known hosts
111	SSH	SSH Authorized Keys	Collect SSH authorized keys
112	SSH	SSH Configs	Collect SSH configurations
113	SSH	SSHD Configs	Collect SSHD configurations
114	Network	Hosts	Collect hosts
115	Network	ICMP Table	Collect ICMP table
116	Network	IP Routes	Collect IP routes

117	Network	IP Tables	Collect IP tables
118	Network	Raw Table	Collect Raw table
119	Network	Network Interfaces	Collect network interfaces
120	Network	TCP Table	Collect TCP table
121	Network	UDPLite Table	Collect UDPLite table
122	Network	UDP Table	Collect UDP table
123	Network	Unix Sockets	Collect unix sockets
124	Network	ARP Table	Collect ARP table
125	Network	DNS Resolvers	Collect DNS resolvers
126	Other Evidence	APT Sources	Collect APT sources
127	Other Evidence	APT History	Collect APT history
128	Other Evidence	DEB Packages	Collect Debian packages
129	Other Evidence	YUM Sources	Collect YUM sources
130	Other Evidence	SELinux Configs	Collect SELinux configurations
131	Other Evidence	SELinux Settings	Collect SELinux settings
132	Other Evidence	SUID Binaries	Collect SUID binaries
133	Other Evidence	Shell History	Collect shell history
134	Other Evidence	System Artifacts	Collect system artifacts (Files of collected evidence. For example: /etc/passwd file)

135	Other Evidence	Log Files	Collect log files under /var/log/

Linux Artifact List

1	Server	Apache Logs	Collect Apache Logs
2	Server	NGINX Logs	Collect NGINX Logs
3	Server	MongoDB Logs	Collect MongoDB Logs
4	Server	MySQL Logs	Collect MySQL Logs
5	Server	PostgreSQL Logs	Collect PostgreSQL Logs
6	Server	SSH Server Logs	Collect SSH Server Logs
7	Server	DHCP Server Logs	Collect DHCP Server Logs
8	System	System Logs	Collect System Logs
9	System	Messages	Collect Messages Logs
10	System	Auth Logs	Collect Auth Logs
11	System	Secure	Collect Secure Logs
12	System	Boot Logs	Collect Boot Logs
13	System	Kernel Logs	Collect Kernel Logs
14	System	Mail Logs	Collect Mail Logs
15	Docker	Docker Changes	Collect Docker Changes.
16	Docker	Docker Containers	Collect Docker Containers.
17	Docker	Docker Image History	Collect Docker Image History.
18	Docker	Docker Images	Collect Docker Images.
19	Docker	Docker Info	Collect Docker Info.

20	Docker	Docker Networks	Collect Docker Networks.
21	Docker	Docker Processes	Collect Docker Processes.
22	Docker	Docker Volumes	Collect Docker Volumes.
23	Docker	Docker Container Logs	Collect Docker Container Logs
24	Docker	Docker Logs	Collect Docker Logs on Filesystem
25	Communication	AnyDesk Logs	Collect AnyDesk Logs

macOS Collections

XDR Forensics supports the following macOS Evidence and Artifacts

macOS Evidence List

1	Server	Apache Logs	Collect Apache Logs
2	Server	NGINX Logs	Collect NGINX Logs
3	Server	MongoDB Logs	Collect MongoDB Logs
4	Server	MySQL Logs	Collect MySQL Logs
5	Server	PostgreSQL Logs	Collect PostgreSQL Logs
6	System	System Logs	Collect System Logs
7	System	Install Logs	Collect Install Logs
8	System	Wifi Logs	Collect Wifi Logs
9	System	KnowledgeC	Collect KnowledgeC Database
10	Docker	Docker Changes	Collect Docker Changes
11	Docker	Docker Containers	Collect Docker Containers
12	Docker	Docker Image History	Collect Docker Image History
13	Docker	Docker Images	Collect Docker Images
14	Docker	Docker Info	Collect Docker Info
15	Docker	Docker Networks	Collect Docker Networks
16	Docker	Docker Processes	Collect Docker Processes
17	Docker	Docker Volumes	Collect Docker Volumes
18	Docker	Docker Container Logs	Collect Docker Container Logs

19	Docker	Docker Logs	Collect Docker Logs on Filesystem
20	Communication	AnyDesk Logs	Collect AnyDesk Logs
21	Communication	Teamviewer Logs	Collect Teamviewer Logs
22	Communication	Discord Desktop Cache	Collect Discord Desktop Cache
23	Communication	Splashtop Mac Logs	Collect Splashtop Mac Application Logs
24	Utilities Artifacts	Parallels Logs	Collect Parallels Logs
25	Utilities Artifacts	Homebrew Logs	Collect Homebrew Logs
26	Antivirus Logs	Sophos Events Database	Collect Sophos Events Database
27	Antivirus Logs	Sophos Logs	Collect Sophos Logs
1	Processes	Auto Loaded Processes	Collect info on autoloaded processes
2	Processes	Processes	Collect Processes
3	Browser	Default Browser	Collect Default Browser
4	Browser	Chrome Cookies	Collect Chrome Cookies
5	Browser	Edge Cookies	Collect Edge Cookies
6	Browser	Opera Cookies	Collect Opera Cookies
7	Browser	Vivaldi Cookies	Collect Vivaldi Cookies

8	Browser	Arc Cookies	Collect Arc Cookies
9	Browser	Brave Cookies	Collect Brave Cookies
10	Browser	QQ Cookies	Collect QQ Cookies
11	Browser	Chrome Bookmarks	Collect Chrome Bookmarks
12	Browser	Edge Bookmarks	Collect Edge Bookmarks
13	Browser	Opera Bookmarks	Collect Opera Bookmarks
14	Browser	Vivaldi Bookmarks	Collect Vivaldi Bookmarks
15	Browser	Arc Bookmarks	Collect Arc Bookmarks
16	Browser	Brave Bookmarks	Collect Brave Bookmarks
17	Browser	QQ Bookmarks	Collect QQ Bookmarks
18	Browser	Chrome User Profiles	Collect Chrome User Profiles
19	Browser	Edge User Profiles	Collect Edge User Profiles
20	Browser	Opera User Profiles	Collect Opera User Profiles
21	Browser	Vivaldi User Profiles	Collect Vivaldi User Profiles
22	Browser	Arc User Profiles	Collect Arc User Profiles
23	Browser	Brave User Profiles	Collect Brave User Profiles
24	Browser	QQ User Profiles	Collect QQ User Profiles

25	Browser	Chrome Extensions	Collect Chrome Extensions
26	Browser	Edge Extensions	Collect Edge Extensions
27	Browser	Opera Extensions	Collect Opera Extensions
28	Browser	Firefox Extensions	Collect Firefox Extensions (Addons)
29	Browser	Chrome Local Storage	Collect Chrome Local Storage
30	Browser	Edge Local Storage	Collect Edge Local Storage
31	Browser	Opera Local Storage	Collect Opera Local Storage
32	Browser	Vivaldi Local Storage	Collect Vivaldi Local Storage
33	Browser	Arc Local Storage	Collect Arc Local Storage
34	Browser	Brave Local Storage	Collect Brave Local Storage
35	Browser	QQ Local Storage	Collect QQ Local Storage
36	Browser	Dump Chrome Indexed DB	Dump Chrome Indexed DB
37	Browser	Dump Edge Indexed DB	Dump Edge Indexed DB
38	Browser	Dump Opera Indexed DB	Dump Opera Indexed DB
39	Browser	Dump Vivaldi Indexed DB	Dump Vivaldi Indexed DB
40	Browser	Dump Arc Indexed DB	Dump Arc Indexed DB

41	Browser	Dump Brave Indexed DB	Dump Brave Indexed DB
42	Browser	Dump QQ Indexed DB	Dump QQ Indexed DB
43	Browser	Chrome Web Storage	Collect Chrome Web Storage
44	Browser	Edge Web Storage	Collect Edge Web Storage
45	Browser	Opera Web Storage	Collect Opera Web Storage
46	Browser	Vivaldi Web Storage	Collect Vivaldi Web Storage
47	Browser	Arc Web Storage	Collect Arc Web Storage
48	Browser	Brave Web Storage	Collect Brave Web Storage
49	Browser	QQ Web Storage	Collect QQ Web Storage
50	Browser	Chrome Form History	Collect Chrome Form History
51	Browser	Edge Form History	Collect Edge Form History
52	Browser	Opera Form History	Collect Opera Form History
53	Browser	Vivaldi Form History	Collect Vivaldi Form History
54	Browser	Arc Form History	Collect Arc Form History
55	Browser	Brave Form History	Collect Brave Form History
56	Browser	QQ Form History	Collect QQ Form History

57	Browser	Chrome Thumbnails	Collect Chrome Thumbnails
58	Browser	Edge Thumbnails	Collect Edge Thumbnails
59	Browser	Opera Thumbnails	Collect Opera Thumbnails
60	Browser	Vivaldi Thumbnails	Collect Vivaldi Thumbnails
61	Browser	Arc Thumbnails	Collect Arc Thumbnails
62	Browser	Brave Thumbnails	Collect Brave Thumbnails
63	Browser	QQ Thumbnails	Collect QQ Thumbnails
64	Browser	Chrome Favicons	Collect Chrome Favicons
65	Browser	Edge Favicons	Collect Edge Favicons
66	Browser	Opera Favicons	Collect Opera Favicons
67	Browser	Vivaldi Favicons	Collect Vivaldi Favicons
68	Browser	Arc Favicons	Collect Arc Favicons
69	Browser	Brave Favicons	Collect Brave Favicons
70	Browser	QQ Favicons	Collect QQ Favicons
71	Browser	Chrome Login Data	Collect Chrome Login Data
72	Browser	Edge Login Data	Collect Edge Login Data

73	Browser	Opera Login Data	Collect Opera Login Data
74	Browser	Vivaldi Login Data	Collect Vivaldi Login Data
75	Browser	Arc Login Data	Collect Arc Login Data
76	Browser	Brave Login Data	Collect Brave Login Data
77	Browser	QQ Login Data	Collect QQ Login Data
78	Browser	Chrome Sessions	Collect Chrome Sessions
79	Browser	Edge Sessions	Collect Edge Sessions
80	Browser	Opera Sessions	Collect Opera Sessions
81	Browser	Vivaldi Sessions	Collect Vivaldi Sessions
82	Browser	Arc Sessions	Collect Arc Sessions
83	Browser	Brave Sessions	Collect Brave Sessions
84	Browser	QQ Sessions	Collect QQ Sessions
85	Browser	Chrome Browsing History	Collect visited URLs from Google Chrome
86	Browser	Edge Browsing History	Collect visited URLs from Microsoft Edge
87	Browser	Firefox Browsing History	Collect visited URLs from Mozilla Firefox

88	Browser	Opera Browsing History	Collect visited URLs from Opera
89	Browser	Safari Browsing History	Collect visited URLs from Safari
90	Browser	Vivaldi Browsing History	Collect visited URLs from Vivaldi
91	Browser	Waterfox Browsing History	Collect visited URLs from Waterfox
92	Browser	Brave Browsing History	Collect visited URLs from Brave
93	Browser	Arc Browsing History	Collect visited URLs from Arc
94	Browser	QQ Browsing History	Collect Visited URLs from QQ
95	Browser	Chrome Downloads	Collect Chrome Downloads
96	Browser	Safari Downloads	Collect Safari Downloads
97	Browser	Firefox Downloads	Collect Firefox Downloads
98	Browser	Edge Downloads	Collect Edge Downloads
99	Browser	Opera Downloads	Collect Opera Downloads
100	Browser	Vivaldi Downloads	Collect Vivaldi Downloads
101	Browser	Arc Downloads	Collect Arc Downloads
102	Browser	Brave Downloads	Collect Brave Downloads
103	Browser	Waterfox Downloads	Collect Waterfox Downloads

104	Browser	QQ Downloads	Collect QQ Downloads
105	Browser	Firefox Cookies	Collect Firefox Cookies
106	System	Crashes	Collect Crashes
107	System	Gatekeeper	Collect Gatekeeper details
108	System	Gatekeeper Approved Apps	Collect Gatekeeper apps allowed to run
109	System	Installed Applications	Collect info on installed apps
110	System	Kernel Extensions Info	Collect kernel extensions info
111	System	Launchd Overrides	Collect override keys for LaunchDaemons and Agents
112	System	Package Install History	Collect Package Install History
113	System	System Extension Info	Collect system extension info
114	System	System Integrity Protection Status	Collect SIP status
115	System	Print Jobs	Collect print job info
116	System	Printer Info	Collect printer info
117	System	Transparency, Consent, and Control (TCC)	Collect Transparency, Consent, and Control Information
118	System	Quarantine Events	Collect Quarantine Events Database
119	System	Sudo Last Run	Collect Sudo Last Run

120	System	iMessage	Collect iMessages
121	System	Dock Items	Collect Dock Items
122	System	Document Revisions	Collect Document Revisions
123	System	Apple System Logs (ASL)	Collect Apple System Logs (ASL)
124	System	Apple Audit Logs	Collect Apple Audit Logs
125	System	Shared File List	Collect Shared File List (SFL) items
126	System	Shell History	Collect Shell History
127	System	Downloaded Files Information	Collect information about downloaded files
128	System	Cron Jobs	Collect Cron Jobs
129	System	Quick Look Cache	Collect Quick Look Cache
130	System	Event Taps	Collect Event Taps
131	System	Re-Opened Apps	Collect Re-Opened Apps
132	System	Most Recently Used (MRU)	Collect Most Recently Used (MRU) items
133	System	Login Items	Collect Login Items
134	System	Collect File System (FS) Events	Collect File System Events
135	System	Parse File System (FS) Events	Parse File System Events
136	Disk	Block Devices	Collect Block Devices

137	Disk	Disk Encryption	Collect Disk Encryption status
138	File System	File System Enumeration	Dump file and folder information.
139	File System	.DS_Store Files	Collect information about .DS_Store files.
140	File System	Trash Files	Collect trashed files of users.
141	Configurations	ETC Hosts	Collect ETC Hosts
142	Configurations	ETC Protocols	Collect ETC Protocols
143	Configurations	ETC Services	Collect ETC Services
144	Network	Listening Ports	Collect Listening Ports
145	Network	IP Routes	Collect IP Routes
146	Network	Network Interfaces	Collect Network Interfaces
147	Network	DNS Resolvers	Collect DNS Resolvers
148	Network	DHCP Settings	Collect DHCP (Dynamic Host Configuration Protocol) Settings
149	Network	Wireless Network Connections	Collect Wireless Network Connections
150	Users	User Groups	Collect User Groups
151	Users	Users	Collect Users
152	Users	Logged Users	Collect Logged Users

153	KnowledgeC	Application Usage	Collect Application Usage
154	KnowledgeC	Bluetooth Connections	Collect Bluetooth Connections
155	KnowledgeC	Notification Info	Collect Notification Info
156	Unified Logs	Logind	Filter user login events
157	Unified Logs	Tccd	Filter tccd events
158	Unified Logs	Sshd	Filter ssh activity events
159	Unified Logs	Command Line Activity	Filter command line activity run with elevated privileges
160	Unified Logs	Kernel Extensions	Filter kernel extension events
161	Unified Logs	Screensharing	Filter screen sharing events
162	Unified Logs	Keychain	Filter keychain unlock events
163	Unified Logs	Session Creation and Destruction	Filter sessions creation and destruction events
164	Unified Logs	XProtect Remediation	Filter detecting and blocking malicious software events
165	Unified Logs	Failed Sudo	Filter failed sudo events
166	Unified Logs	Manuel Configuration Profile Install	Filter MDM Clients Events
167	Unified Logs	Network Usage	Filter Network Usage Logs

168	Persistence	Mail Rules	Collect Mail Rules that contain AppleScript
169	Persistence	Login Hooks	Collect Login Hooks
170	Persistence	Logout Hooks	Collect Logout Hooks
171	Persistence	Emond Clients	Collect Emond Clients
172	SSH	SSH Authorized Keys	Collect SSH authorized keys
173	SSH	SSH Configs	Collect SSH configurations
174	SSH	SSH Known Hosts	Collect SSH known hosts
175	SSH	SSHD Configs	Collect SSHD configurations

macOS Artifact List

1	Server	Apache Logs	Collect Apache Logs
2	Server	NGINX Logs	Collect NGINX Logs
3	Server	MongoDB Logs	Collect MongoDB Logs
4	Server	MySQL Logs	Collect MySQL Logs
5	Server	PostgreSQL Logs	Collect PostgreSQL Logs
6	System	System Logs	Collect System Logs
7	System	Install Logs	Collect Install Logs
8	System	Wifi Logs	Collect Wifi Logs
9	System	KnowledgeC	Collect KnowledgeC Database
10	Docker	Docker Changes	Collect Docker Changes
11	Docker	Docker Containers	Collect Docker Containers
12	Docker	Docker Image History	Collect Docker Image History
13	Docker	Docker Images	Collect Docker Images
14	Docker	Docker Info	Collect Docker Info
15	Docker	Docker Networks	Collect Docker Networks
16	Docker	Docker Processes	Collect Docker Processes
17	Docker	Docker Volumes	Collect Docker Volumes
18	Docker	Docker Container Logs	Collect Docker Container Logs

19	Docker	Docker Logs	Collect Docker Logs on Filesystem
20	Communication	AnyDesk Logs	Collect AnyDesk Logs
21	Communication	Teamviewer Logs	Collect Teamviewer Logs
22	Communication	Discord Desktop Cache	Collect Discord Desktop Cache
23	Communication	Splashtop Mac Logs	Collect Splashtop Mac Application Logs
24	Utilities Artifacts	Parallels Logs	Collect Parallels Logs
25	Utilities Artifacts	Homebrew Logs	Collect Homebrew Logs
26	Antivirus Logs	Sophos Events Database	Collect Sophos Events Database
27	Antivirus Logs	Sophos Logs	Collect Sophos Logs

Windows Collections

XDR Forensics supports the following Windows Evidence and Artifacts

Task Creation

Introduction to Task Creation in XDR Forensics

Tasks in XDR Forensics are operations assigned to assets via the XDR Forensics console, either manually or automatically through triggers. Each task can comprise multiple 'tasking assignments,' where a single task on one asset is a 'tasking assignment,' but the term 'task' can also describe the same tasking assignment across many assets. These tasks facilitate various operational needs and can be categorized into three types:

Types of Tasks

1. **Manual Tasks:**
   - These are assigned manually by users directly through the XDR Forensics console.
2. **Scheduled Tasks:**
   - Created by users to commence at a future time. Scheduled tasks can be one-time events or recurring at daily, weekly, or monthly intervals.
3. **Triggered Tasks:**
   - Automatically assigned to assets in response to trigger requests from integrated SIEM, SOAR, or EDR solutions.

Tasks enhance operational efficiency by allowing flexible and automated responses to various cybersecurity scenarios, ensuring that your assets are continually monitored and managed effectively. For more detailed information, please visit our Knowledge Base ↗ and refer to the XDR Forensics release notes.

Activities that Generate Tasking Assignments

In XDR Forensics, tasking assignments are generated by various activities that target asset operations, including:

1. **Data Acquisition:**
   - Initiating the collection of digital evidence from an asset. This can be a comprehensive acquisition or targeted to specific evidence types.
2. **Triage:**
   - Running predefined or custom rules (YARA, Sigma, osquery) to identify suspicious activities or indicators of compromise on the assets.
3. **Timeline (Investigation):**
   - Creating and analyzing timelines to understand the sequence of events on an asset for forensic investigation.
4. **interACT Sessions:**
   - Establishing a secure remote shell session to manually investigate and interact with the asset in real-time.
5. **Baseline Acquisition and Comparison:**
   - Running comparisons to detect deviations from a predefined baseline state of the asset and acquiring baseline data.
6. **Disk/Volume Imaging:**
   - Capturing the complete state of disks or volumes for comprehensive forensic analysis.
7. **Auto Tagging:**
   - Automatically tagging assets based on predefined criteria for easier management and identification.
8. **Calculating Hash:**
   - Generating hash values for files to ensure data integrity and assist in identifying duplicate or tampered files.
9. **Offline Acquisition and Offline Triage:**
   - Performing data acquisition and triage on assets that are not connected to the network.

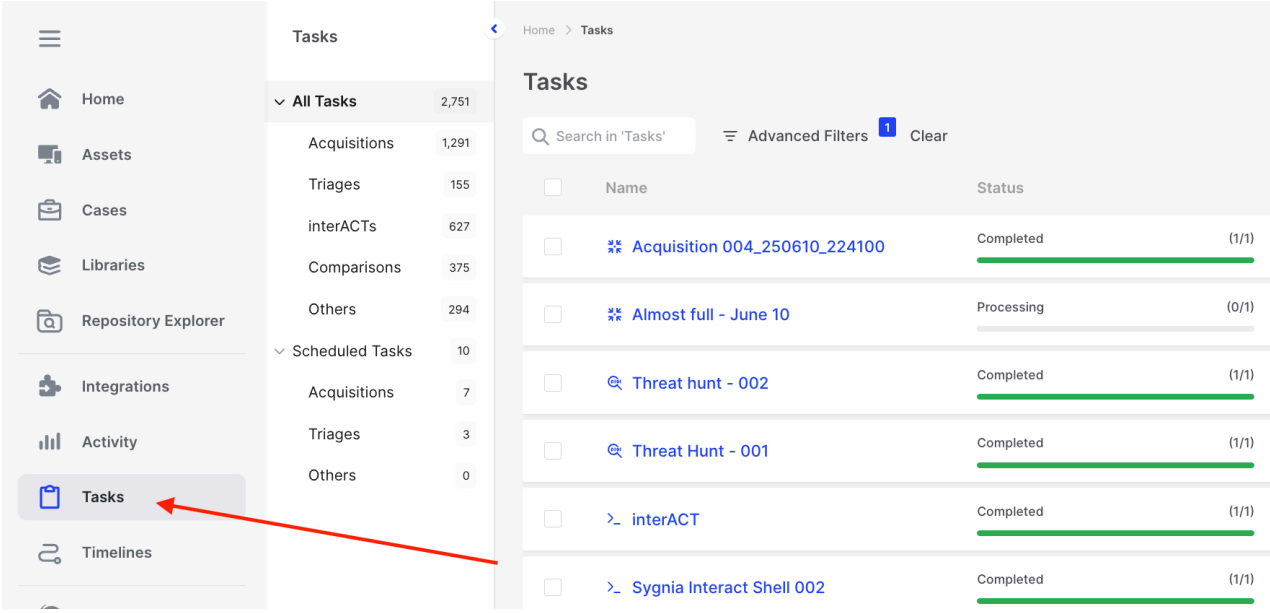
Administrative Tasks

Some more 'administrative' activities also generate tasking assignments and these include:

- **Shutdown, Reboot, and Uninstall:**
  - Remotely managing the power state and software configuration of assets.
- **Isolation:**
  - Isolating an asset from the network to prevent further compromise.
- **Responder Deployment:**
  - Deploying response tools to the asset for immediate action.
- **Purge Local Data and Retry Upload:**
  - Managing data on the asset, including purging local data and retrying data uploads.
- **Migration and Version Update:**
  - Migrating data between systems and updating software versions.
- **Log Retrieval:**
  - Collecting logs for further analysis and troubleshooting.

By understanding and utilizing these task types, users can streamline their incident response and investigation workflows, improving overall security posture and response times.

Tasks are saved at the organization level and can be reviewed comprehensively by navigating to the main XDR Forensics menu > Tasks:



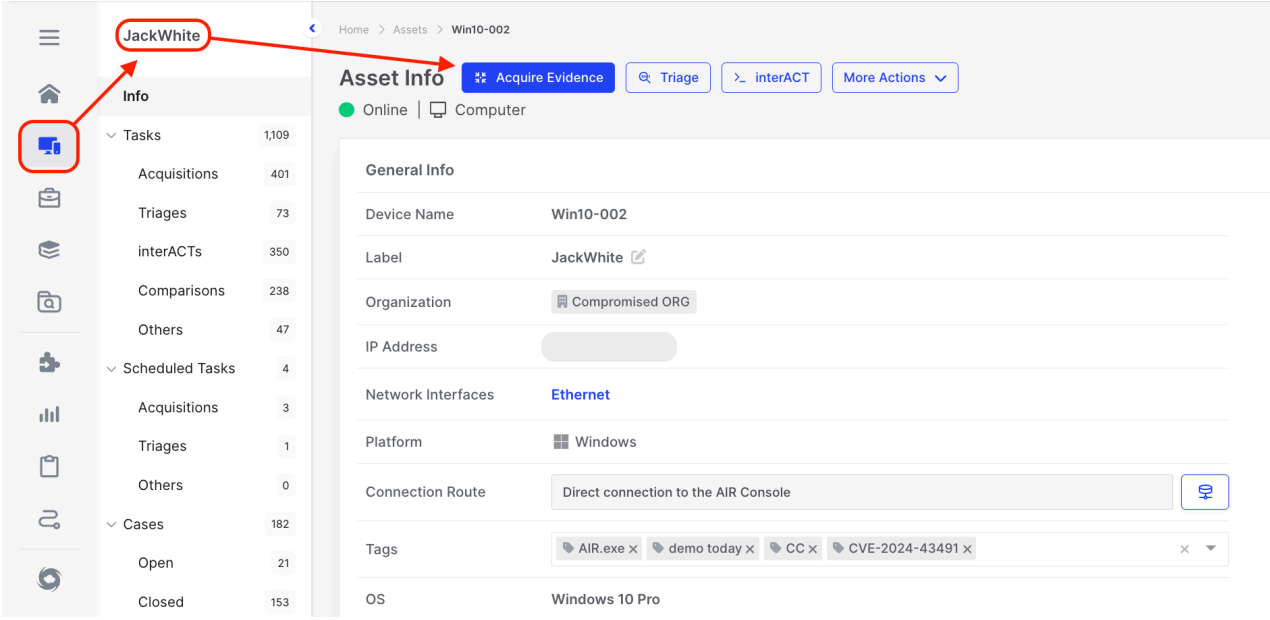
Task Creation: Where to review all of an Organizations's Tasks

Individual tasking assignments (one task on one asset) for an asset can also be reviewed by visiting the specific asset. From the secondary menu, you can select "All Tasks" or utilize the filtered tasks view to focus on specific task types.

Let's now take a look at how to create a Task in XDR Forensics

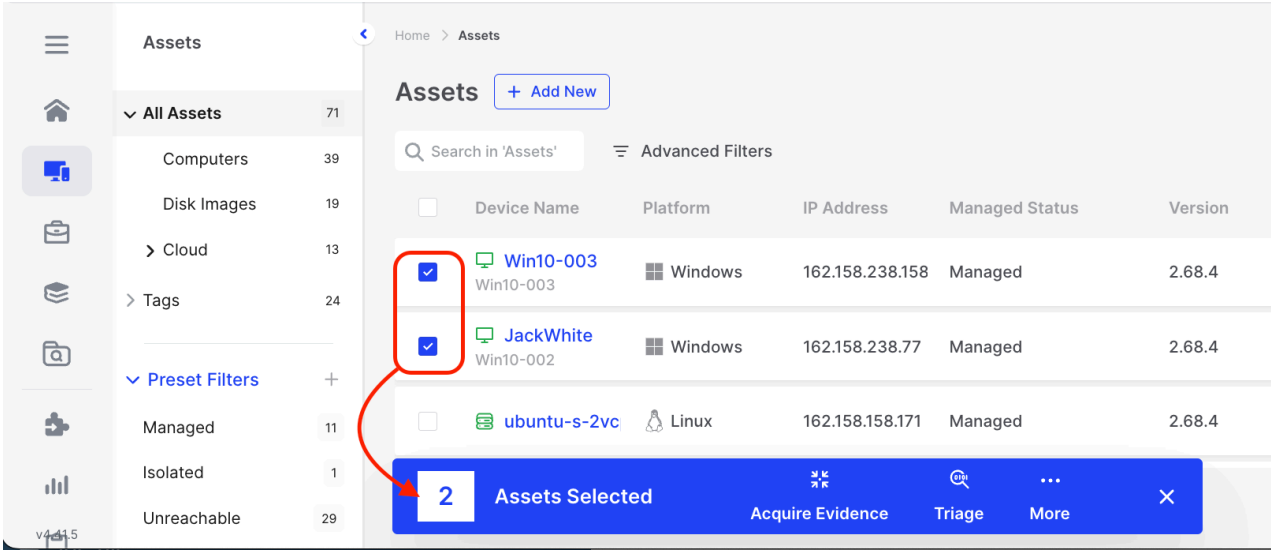
Step 1 - Assets

Select the Asset(s) on which you wish to execute tasks - In the example below we will **Acquire Evidence** from an asset named JackWhite:



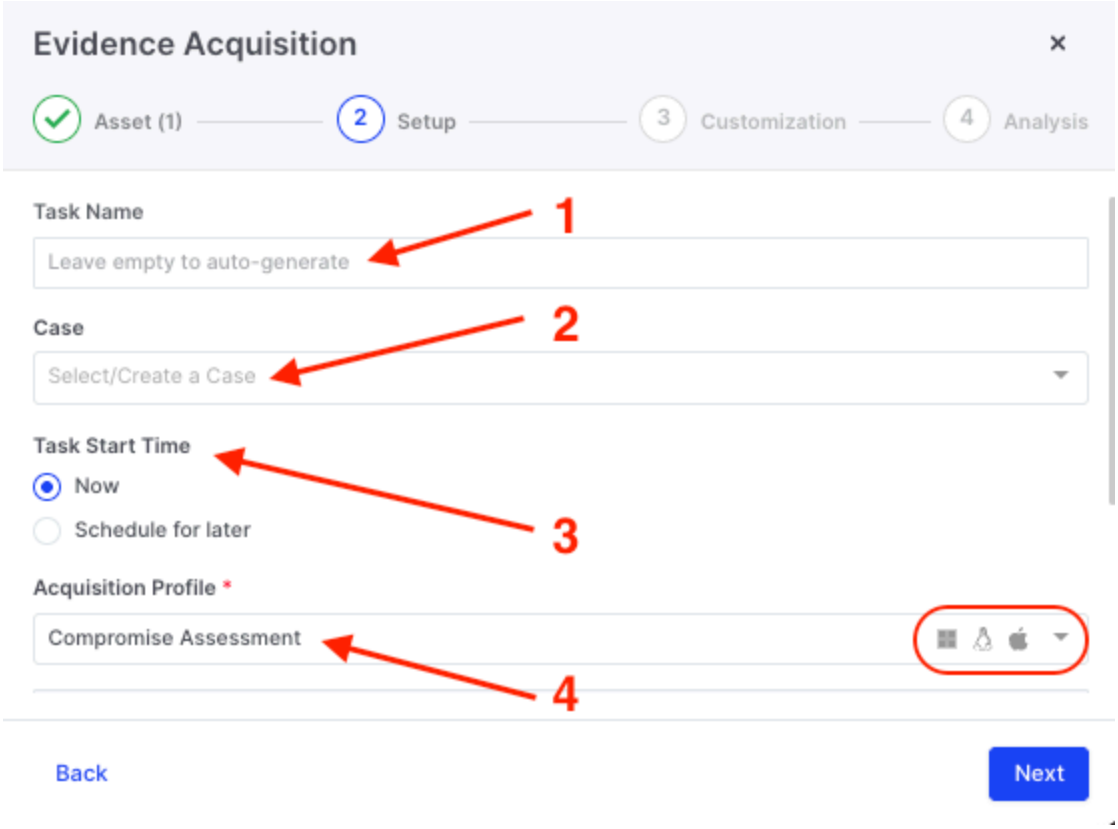
Task Creation: The Acquire Evidence Action Button selected for the asset 'JackWhite"

The Bulk Action Bar will be available if you choose more than one asset:



Task Creation: The Bulk Action Bar indicating 2 assets selected

Step 2 - Setup



Task Creation: Step 2 the setup

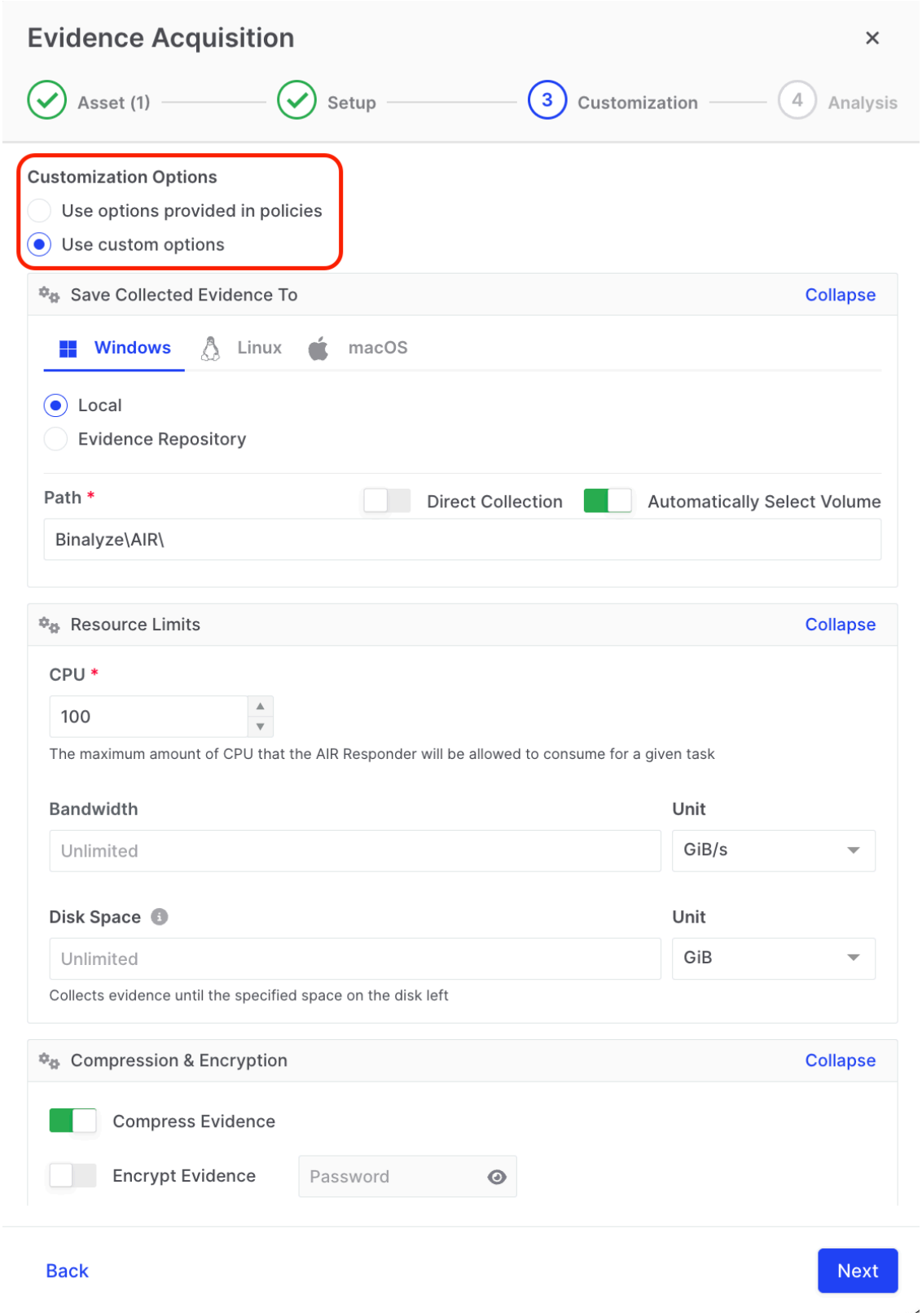
1. **Optionally specify a Task Name.** If left blank, one will be automatically generated based on the task type, sequence number, and date/time.
2. **Allocate the task to a Case. This is important if you need to build a case for an ongoing investigation and you plan to investigate this asset further or other assets as part of the same investigation.** All investigation activity can be recorded within Cases. A Case can be thought of as a container into which activity for a particular investigation can be grouped, making Incident Response management and investigations easier, especially as the Case will also be presented in the Investigation Hub ↗.
3. Select Now as a **Task Start Time to execute the task immediately** (product-platform/features/scheduling-tasks/))
4. Choose an **Acquisition Profile** (e.g., Compromise Assessment, Full Acquisition, etc). We offer many 'out of the box' profiles, but you can also create and save your own as needed.

Acquisition Profile Platform Visibility
As shown in the bottom-right of the screenshot above, AIR now displays the supported operating systems—Windows, Linux, or macOS—within each acquisition profile. This helps ensure analysts select valid profiles for the target asset.

Step 3 - Customization

This step allows you to apply the existing organizational policy or, if you have the appropriate privileges, customize the following settings:

- **Specify where the collected evidence should be saved.** You can choose a local path on the asset—configurable per operating system—or direct the collection to an external evidence repository.
- **Optionally apply resource limits**—including CPU, bandwidth, and disk space—to the task assignment to minimize potential impact on the asset.
- Enable compression and encryption to be applied to the collected evidence.

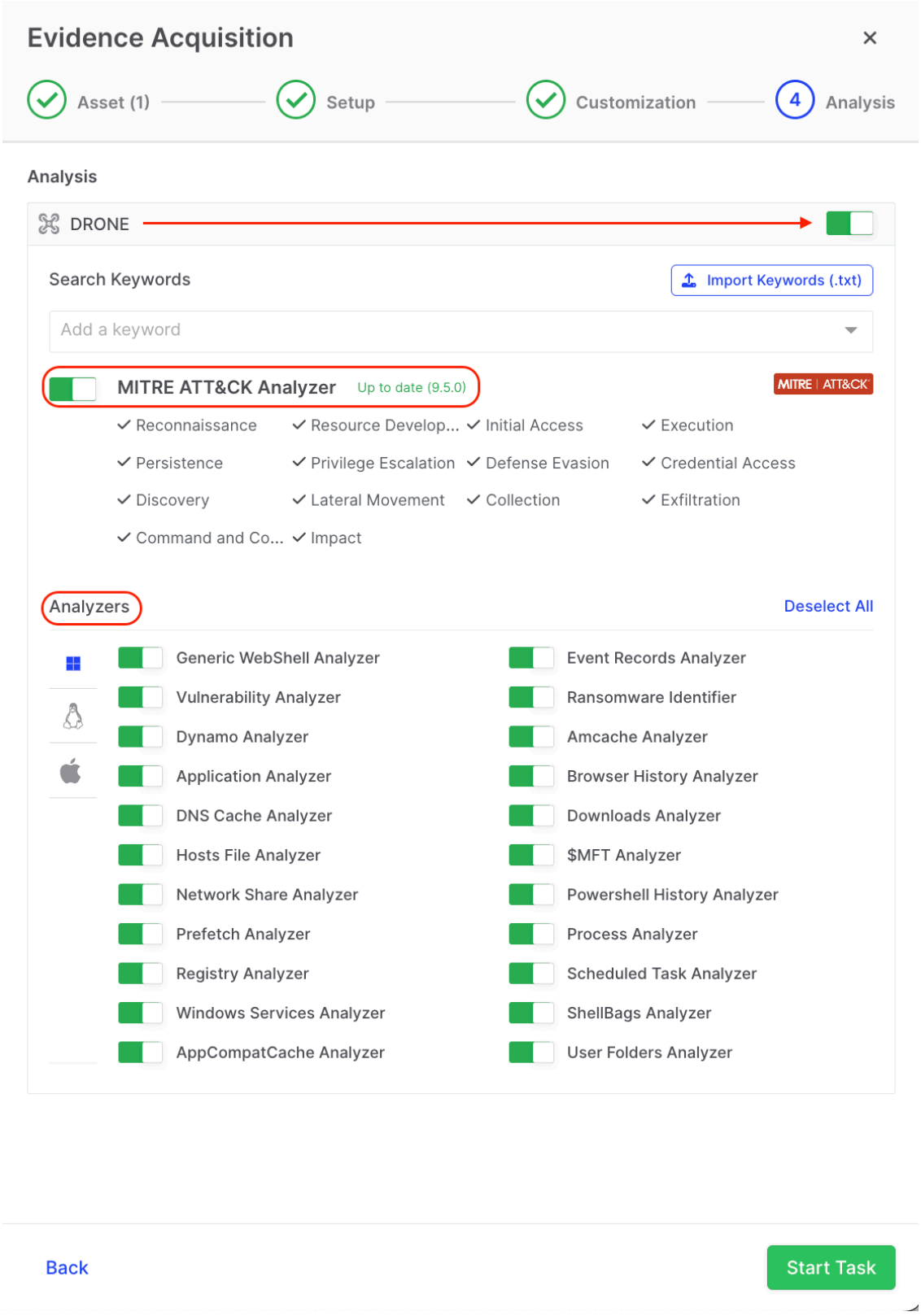


Task Creation: Customization of options

Step 4 - Analysis

In this final step, you can choose to enable or disable the automated DRONE and MITRE ATT&CK analyzers.

We strongly recommend keeping both enabled—they have minimal impact on system resources and provide prioritized threat intelligence to help accelerate your investigations. The MITRE ATT&CK analyzer runs directly on live assets and, when combined with other analyzers, enables immediate identification of potentially compromised systems. This streamlines your workflow by helping you focus on the most critical areas first.



Task Creation: Toggle options for DRONE, MITRE ATT&CK and XDR Forensics's analyzers

Keyword Searches

Step 4, 'Analysis', also provides the option to add individual keywords or upload keyword list files.
This enables DRONE to perform targeted searches, helping investigators conduct more focused and efficient analysis within their evidence collections.

Keyword Lists Features:

- No character limit for keyword lists, but a **1 MB file size limit** applies.
- Each keyword must be on a new line for proper search functionality.
- Keyword searches are limited to data within the **Case.db** (excluding CSV files).
- Keyword searches are supported by regex, read more here: **Regex in XDR Forensics** ↗
- This search functionality extends to **event log data** collected by Sigma analyzers, including:
  - **Windows**: Event Record Analyzer
  - **Linux**: Syslog Analyzer
  - **macOS**: Audit Event Analyzer

> Keyword searches are limited to **data within the Case.db** and do not include content within any CSV files prepared by XDR Forensics (eg; $mft, $UsnJrnl). Additionally, the search will not cover the contents of files on or collected from the asset.

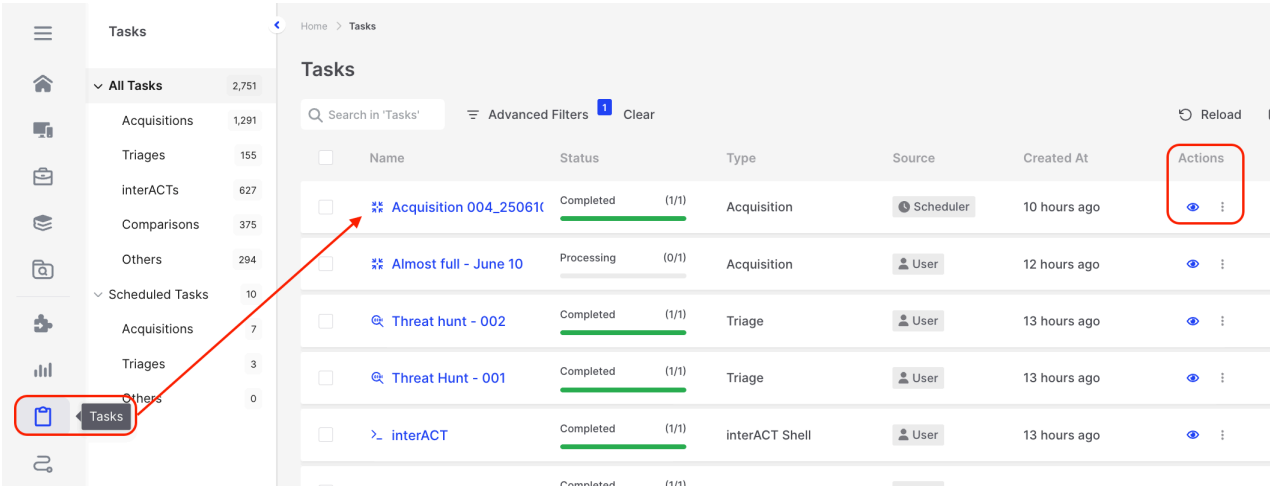
This feature offers investigators greater flexibility and precision in their searches, significantly enhancing the DRONE module's capabilities. Regex support will be added soon in an upcoming release.

The results of all tasks—and their associated reports—are accessible via the Investigation Hub.

Links to these reports within the Investigation Hub can be found in the following three locations::

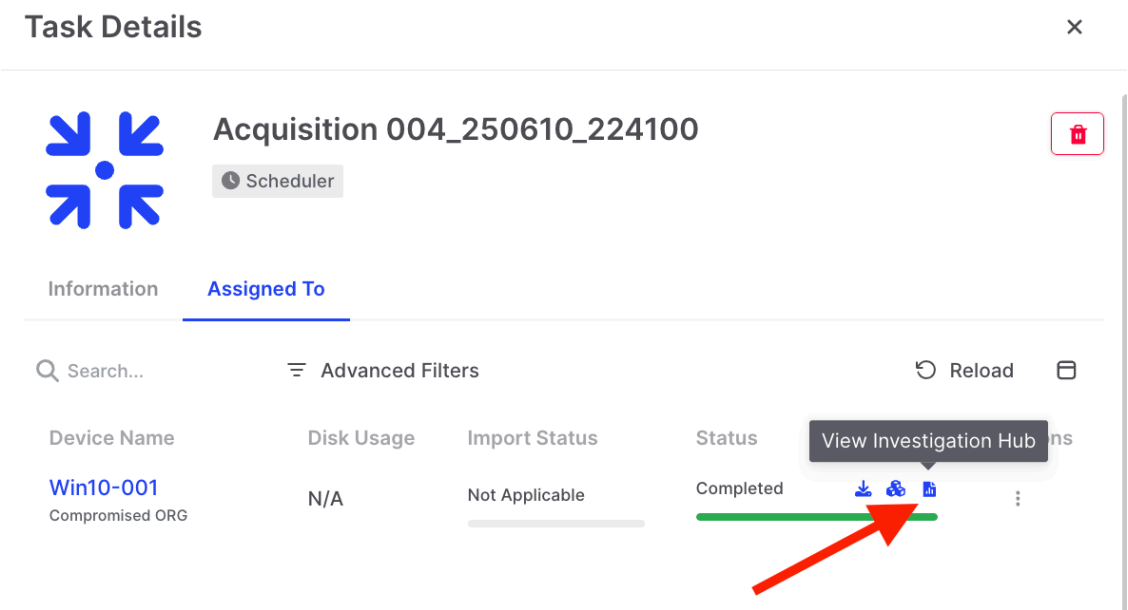
1. Tasks, via the main menu.
2. The page for the individual Asset > Acquisitions.
3. And finally, Case Acquisitions, if the task is sent to a Case.

1) Accessing the Tasks via the main menu:



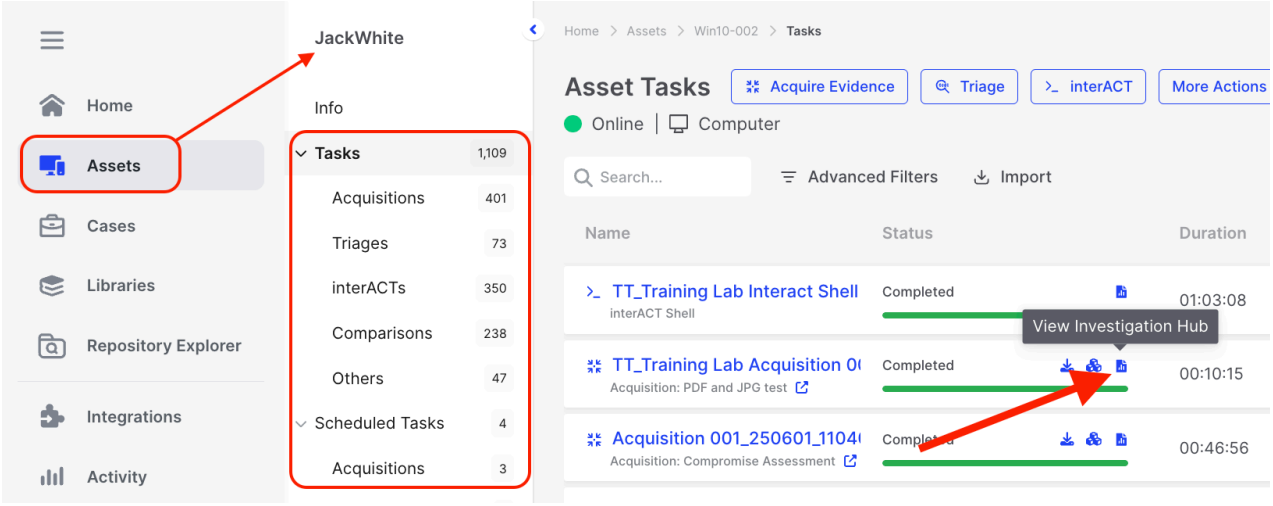
Task Creation: Accessing the Task report via Tasks in the main menu

Clicking the 'eye' icon in the Actions column opens the Task Details view, where you can access the data associated with the acquisition task via the link to the Investigation Hub.



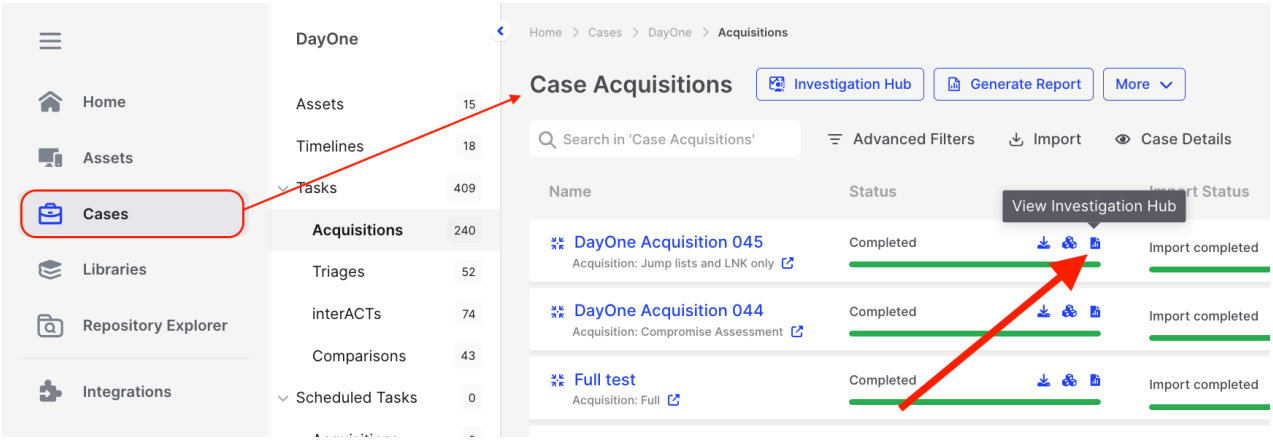
Task Creation: Accessing the Task report via Task Details

2) Below we see a report being accessed from the Assets menu:



Task Creation: Accessing the Task report via the Assets menu

3) And finally a report from the Case Acquisition page but only if you have sent it here:



Task Creation: Accessing the Task report via the Cases menu

Asset Management with Persistent Saved Filters

Managing large-scale asset inventories enables users to save and quickly apply custom filters, facilitating efficient asset management.

Persistent Saved Filters enable users to create and store custom asset filters, making it easier to locate and manage assets without having to reapply filter conditions in each session.

Persistent Saved Filters operate at the organizational level for asset management, meaning any filters you create for asset management will persist only within the organization where they were defined. If you switch to a different organization, those filters will not be visible. This feature is currently limited to asset management and does not apply to other sections of the platform.

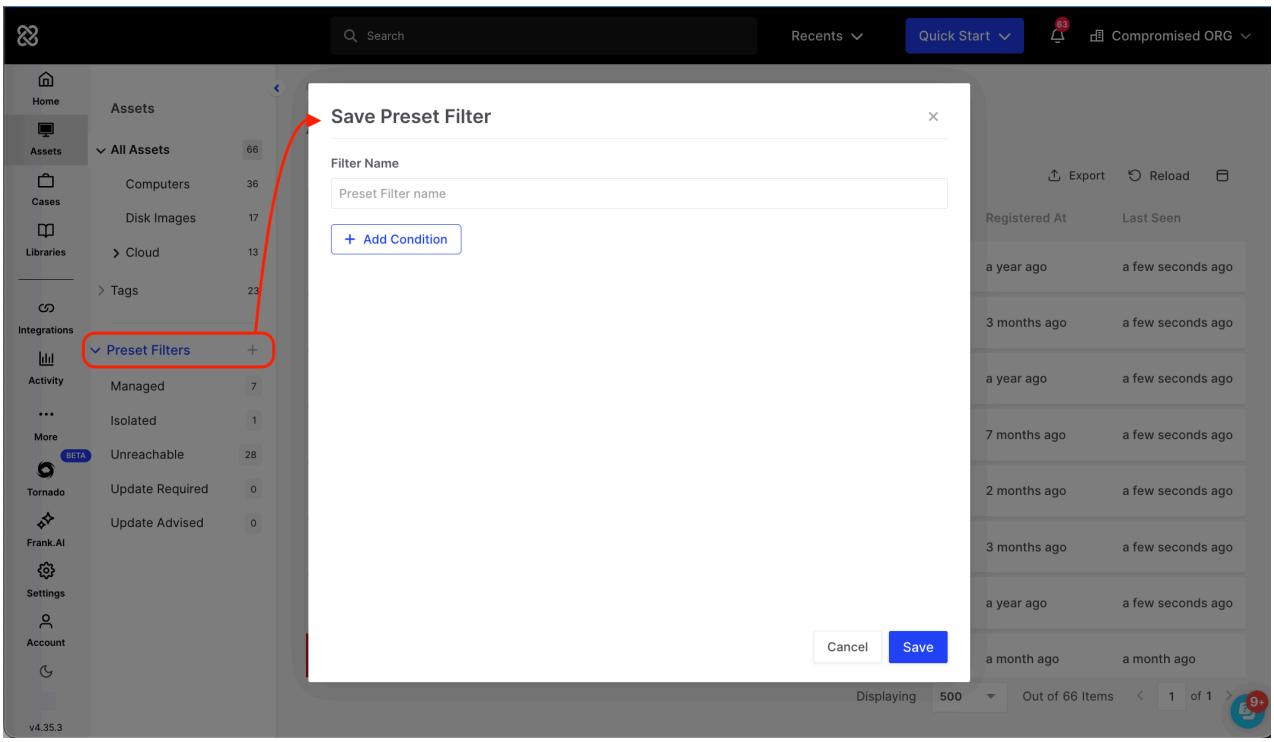
Key Benefits

- Users can save custom filters for frequently used asset searches.
- Saved filters allow for quick application, streamlining asset management.
- Bulk actions and asset monitoring can be performed without re-entering filter conditions.
- Quick actions on saved filters enable faster asset selection.

How to Use Persistent Saved Filters

Step 1: Apply Filters to Your Asset List

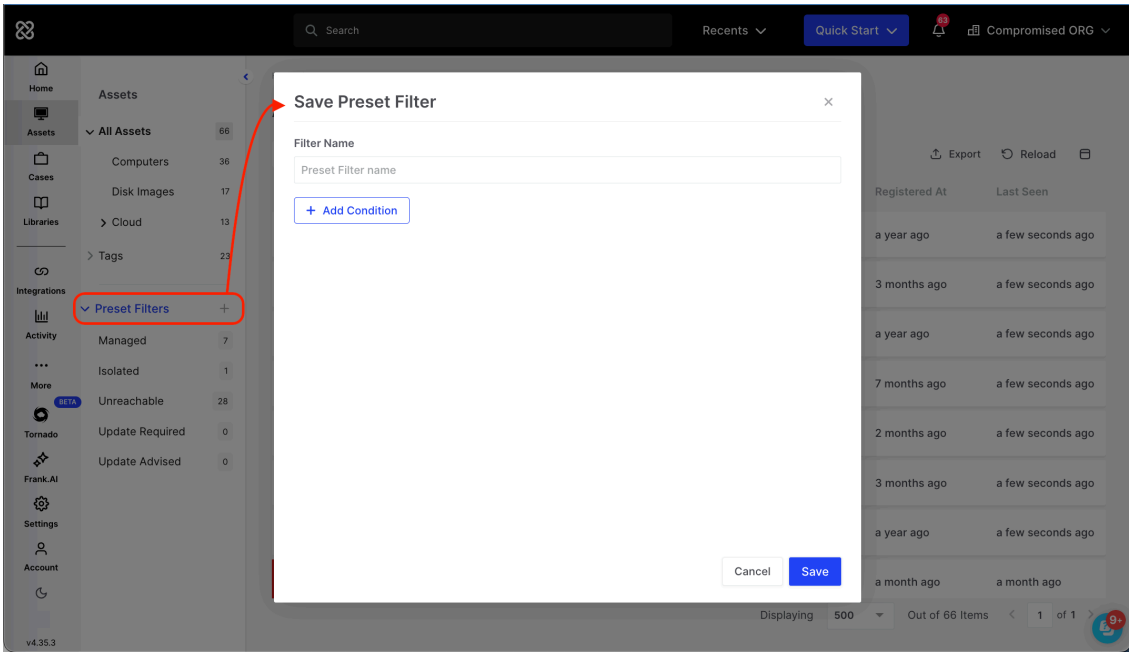
Navigate to the secondary menu on the **Assets** page and click the **'+'** icon to create a **Preset Filter**. The **Save Preset Filter** window will open, allowing you to configure a custom **Asset Filter** that remains persistent for your user account.



Asset Management with Persistent Saved Filters: Save Preset Filter

Step 2: Configure a custom Asset Filter

Use the filtering options to build a filter to refine the asset list based on criteria such as status, tags, or isolation state.



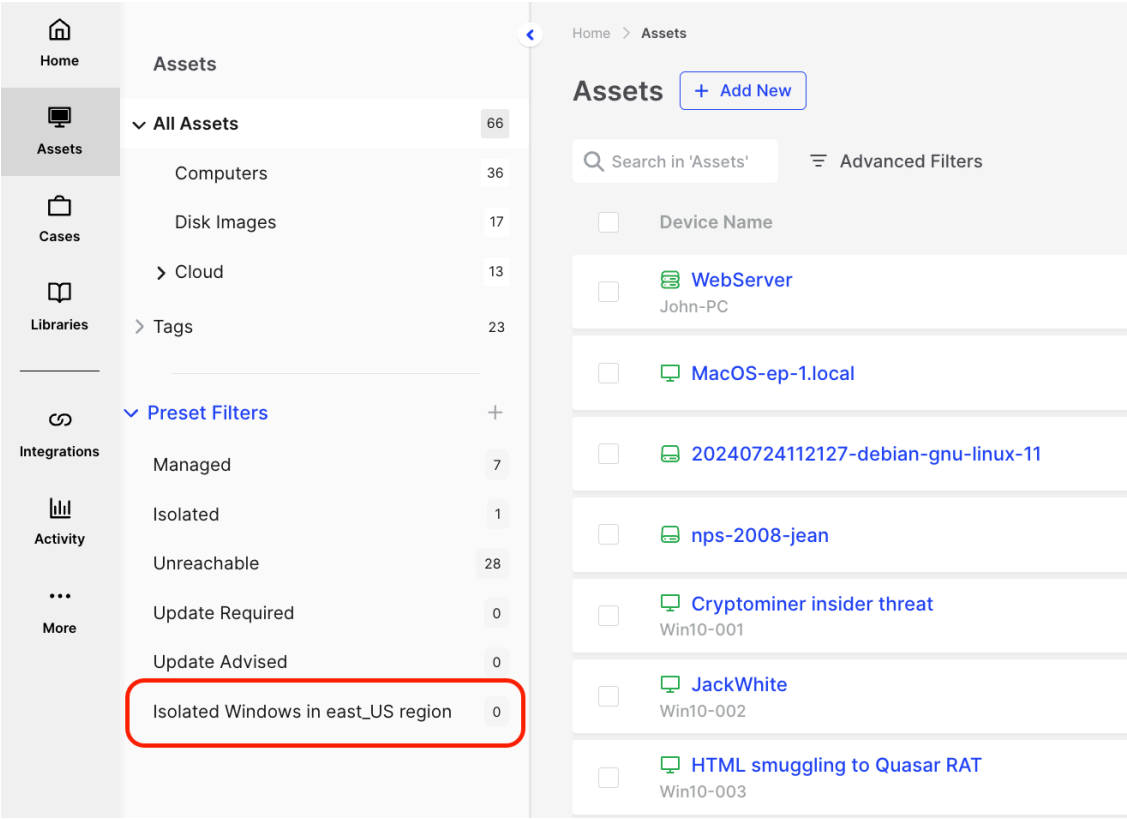
Asset Management with Persistent Saved Filters: Save Preset Filter

Step 3: Save a Custom Filter

After applying the desired filter conditions, enter a name for the filter and click **"Save,"** to confirm your selection.

Step 3: Access and Apply Saved Filters

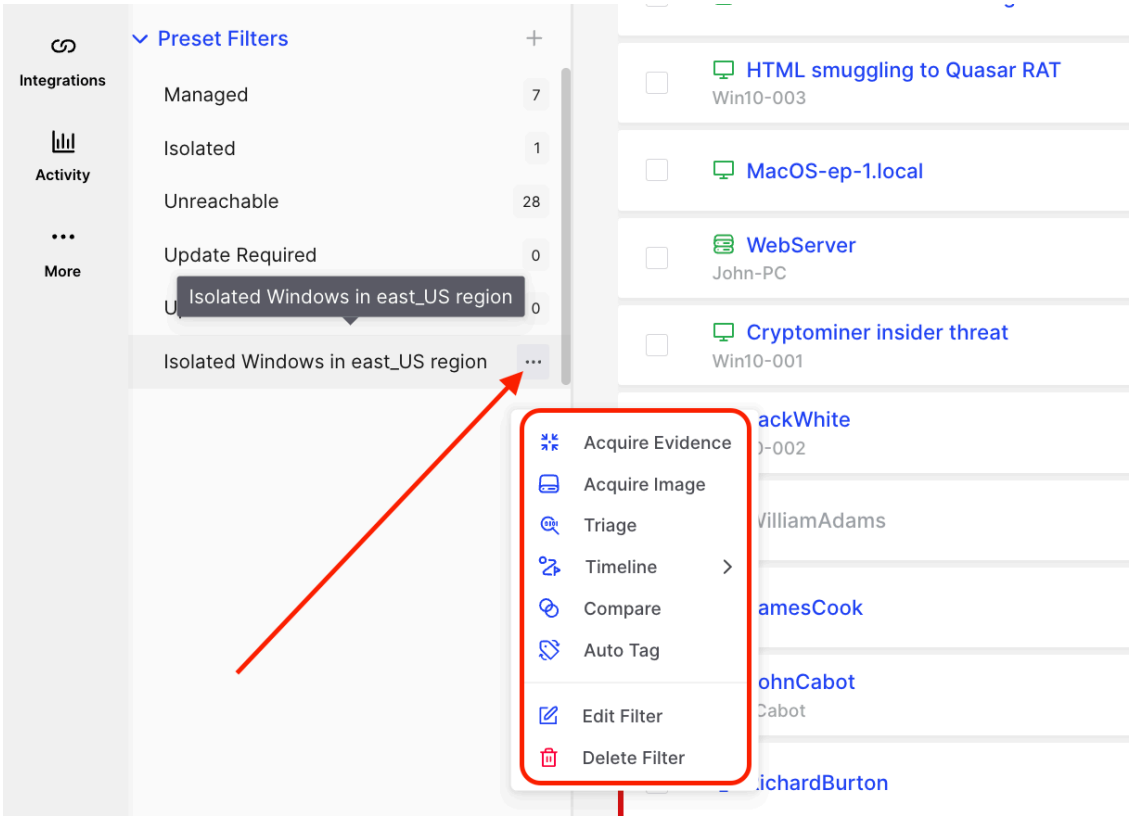
Select the newly saved filter from the Preset Filters drop down list in the secondary menu to instantly apply it.



Asset Management with Persistent Saved Filters: Apply Saved Preset Filter

Step 4: Manage Saved Filters

Users can edit or delete their saved filters anytime by clicking the three-dot menu next to the preset filter name. This menu also provides access to **Quick Actions**, enabling bulk operations directly from saved filters.



Asset Management with Persistent Saved Filters: Manage Saved Preset Filters

Important Notes

Saved filters are per-user, meaning each user can only see and manage their own filters.
Preset filters remain unchanged; predefined system filters such as Managed Assets and Isolated Assets are still available.
Filters persist across sessions, ensuring users do not need to reapply them after logging out.

Regex in DRONE:

When performing keyword searches in DRONE, you can leverage regular expressions (POSIX regex) for more flexible and advanced search capabilities. Here's how it works:

1. Regex Search Format:

To use regex, your keyword must be enclosed between `/ /` slashes. For example:

- `/\d+/` – This will search for one or more digits.
- `/[a-z]+/i` – This will search for one or more lowercase letters and is **case-insensitive** (thanks to the `i` flag).

You can also include optional flags at the end of the regex to modify its behavior:

- **g**: Global (search all occurrences).
- **m**: Multiline (match across multiple lines).
- **i**: Case-insensitive (ignore letter case).
- **s**: Dot matches newline (dot `.` will match any character, including newlines).

Example:
`/[A-Z]+\d+/i` – This will match sequences like "ABC123" or "abc123" regardless of case.

2. Wildcard Search:

If your search contains wildcard patterns like `*?` (indicating lazy quantifiers), it will be treated as a **wildcard search** instead of a regex search.

- For example, `abc*?` will match "abc", "abcd", "abcxyz", etc.

3. String Contains Search:

If your input doesn’t match the regex format and doesn’t contain wildcard symbols, DRONE will perform a **case-insensitive "string contains" search**.

- For example, searching for `example` will return results containing "example", "Example", or "EXAMPLE".

Task Cancellation and Deletion

Viewing and Managing Tasks

All tasks within XDR Forensics can be centrally managed at the organizational level from a dedicated **Tasks** page. This view is accessible from the **Main Menu** by selecting **Tasks**.

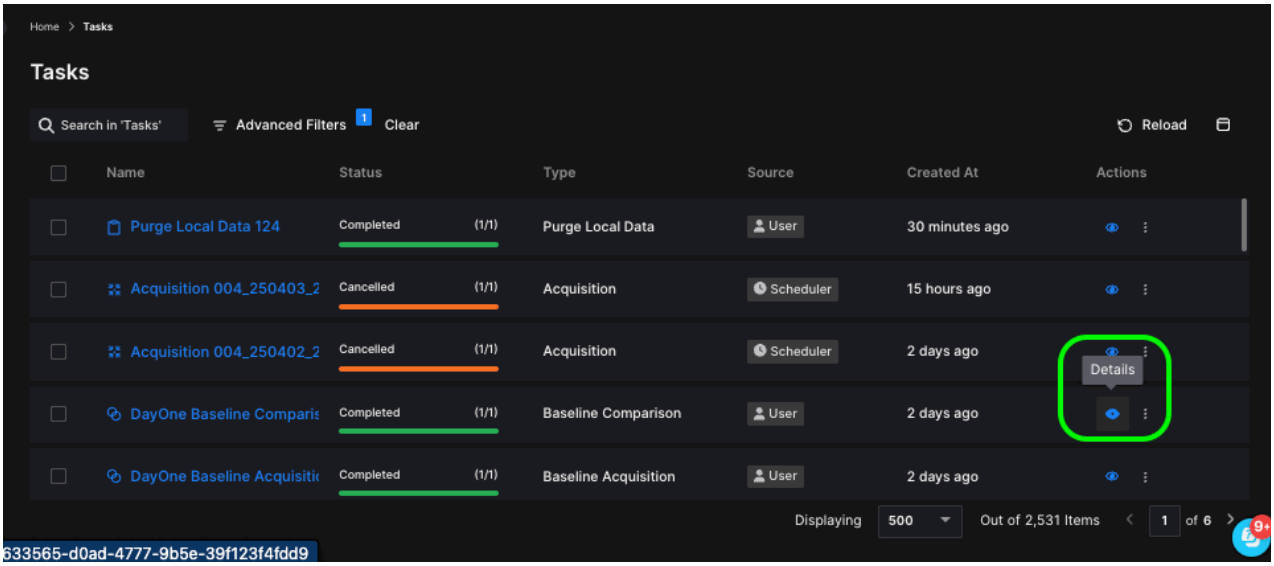
From this interface, users can:

- View a **comprehensive list of all tasks** associated with the currently selected organization.
- Filter and monitor tasks across their lifecycle, including those that are:
  - **Completed**
  - **In progress**
  - **Cancelled**
  - **Failed**

This consolidated view is essential for maintaining operational oversight and ensuring that investigation and response workflows are being executed as expected.

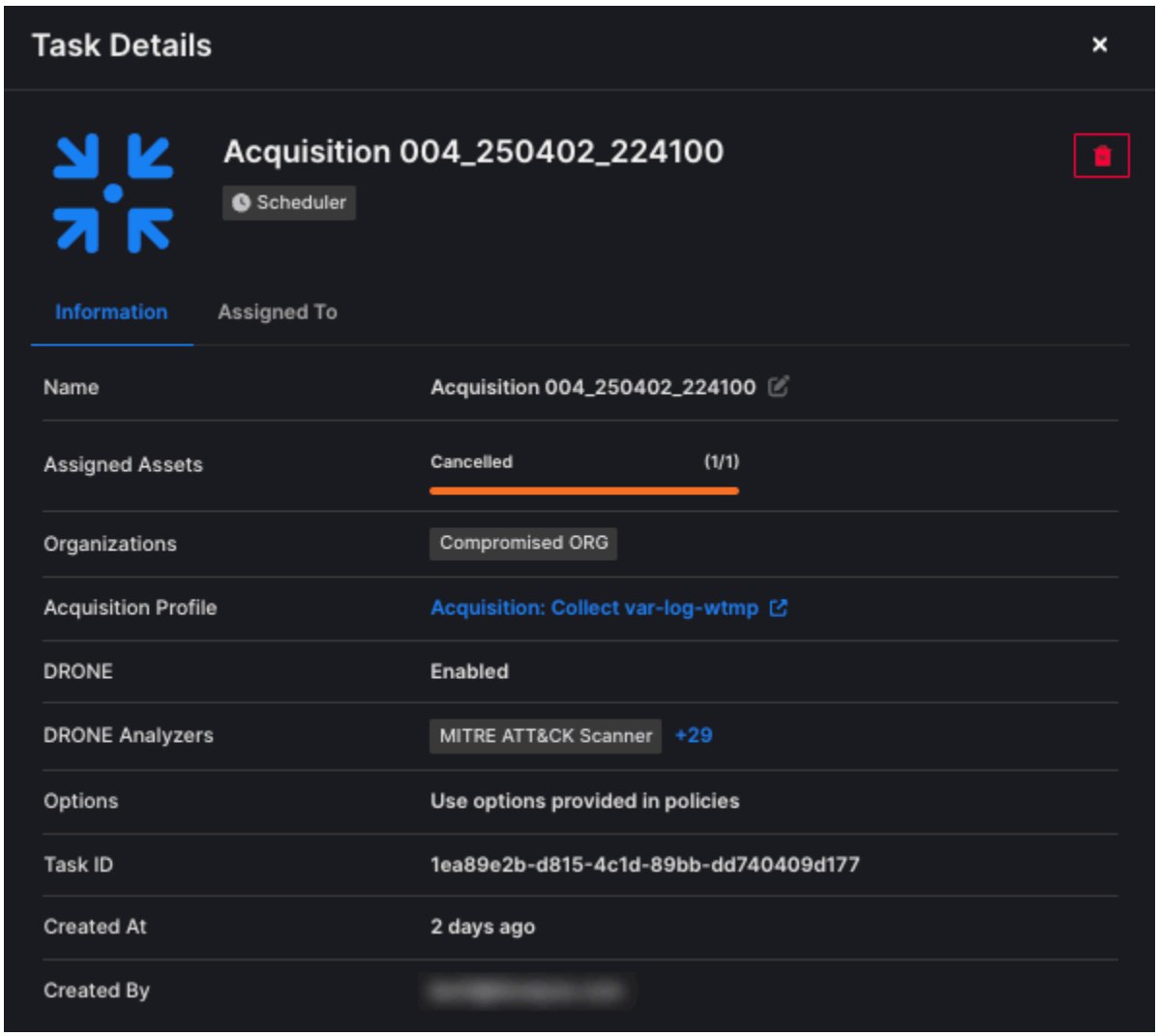
For each task, detailed information such as task type (e.g., acquisition, triage, timeline generation), status, and associated endpoints is available, allowing for granular tracking and auditability.

The blue **'eye'** icon at the end of each task entry allows users to view detailed information about the task.



Task Cancellation and Deletion: Details icon

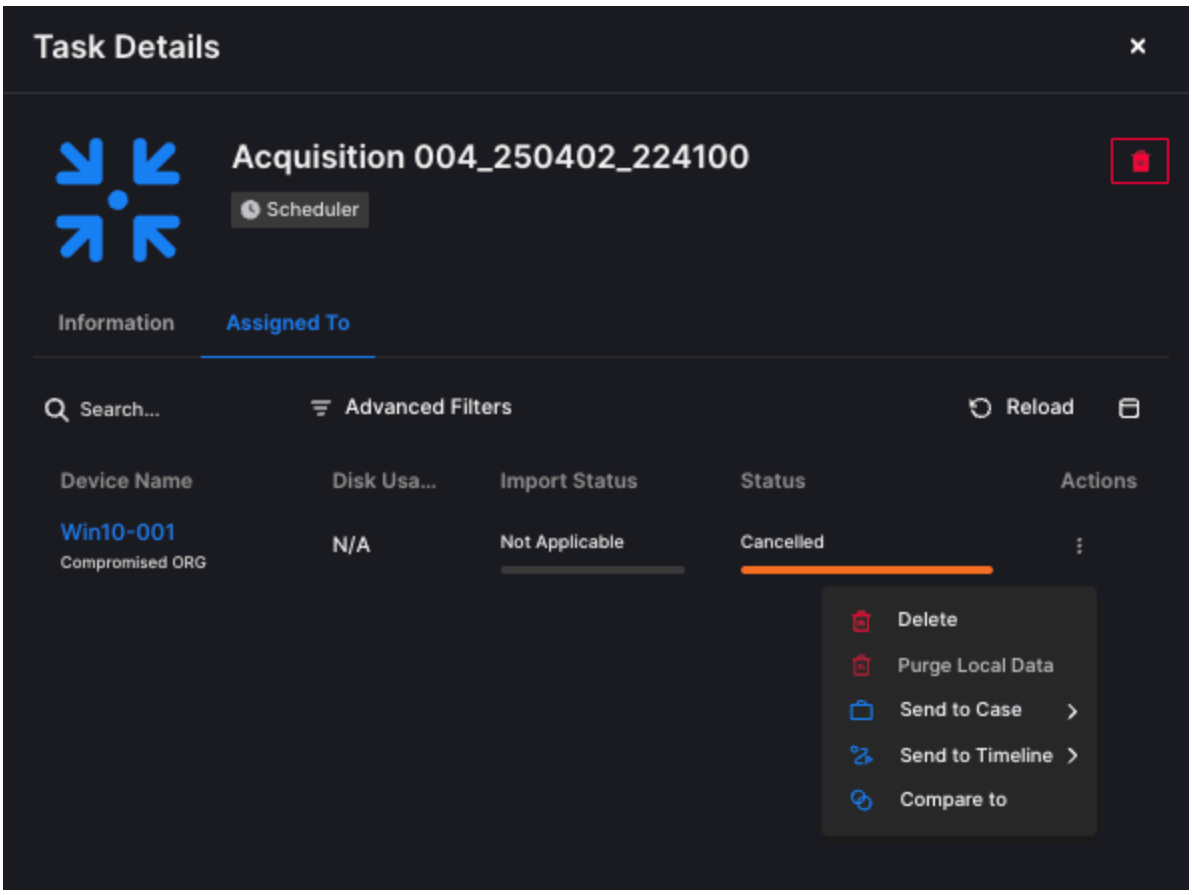
In the **Task Details** window, there are two tabs: **Information** and **Assigned To**. The **Information** tab displays details such as the creator of the task, the policy applied, the DRONE analyzers used, and the selected acquisition profile.



Task Cancellation and Deletion: Task Details

The **Assigned To** tab provides information about the assets to which the task has been assigned. For each asset, an **Actions** menu is available, enabling investigators to:

- **Delete the task**
- **Purge local data** from the endpoint
- **Add the data to a case**
  - Note: Data can be added to multiple cases simultaneously
- **Send the data to a timeline** for correlation and analysis
- **Use the collected data to perform a Compare task** against other acquisitions



Task Cancellation and Deletion: Options avaiable under Actions

Bulk Task Cancellation and Deletion

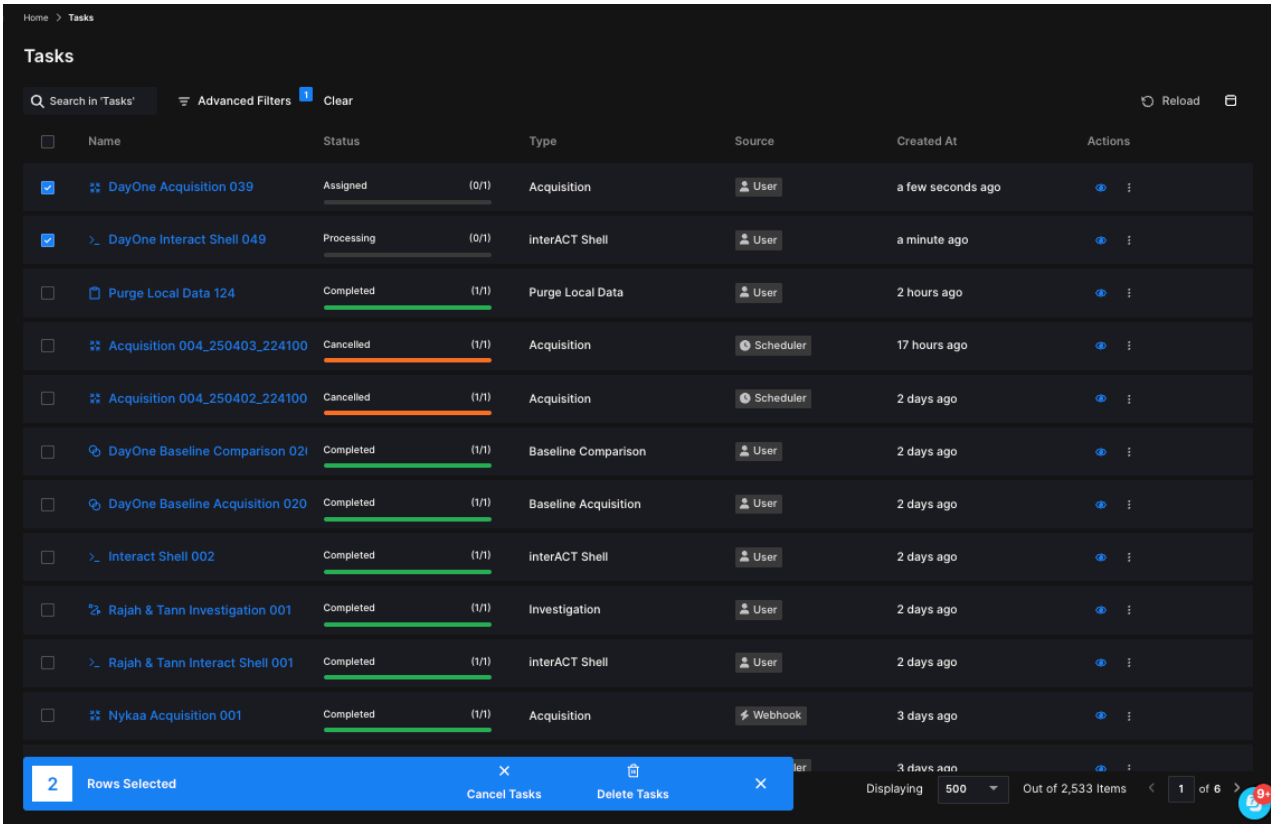
Managing large volumes of tasks is now faster and more efficient with the introduction of **Bulk Task Cancellation** and **Bulk Task Deletion** capabilities.

Bulk Task Cancellation

Users can now cancel one or more tasks directly from the **Tasks** view using the familiar **bulk action toolbar**. This is particularly useful for streamlining task management across environments with high automation or frequent testing.

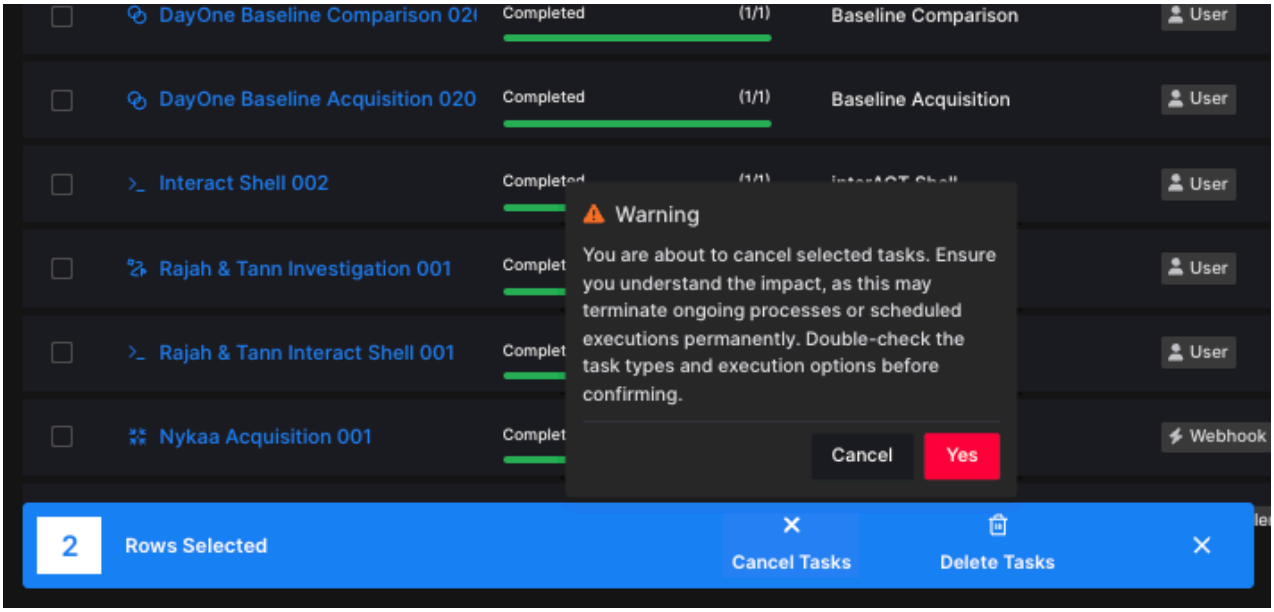
Key features include:

- Full support for **advanced filtering**, allowing users to quickly identify tasks based on:
  - Task type
  - Task name
  - Source
  - Status
- As seen below, once tasks are selected, the bulk action bar automatically appears, displaying a task count (this count reflects the number of tasks selected, not the number of affected assets).



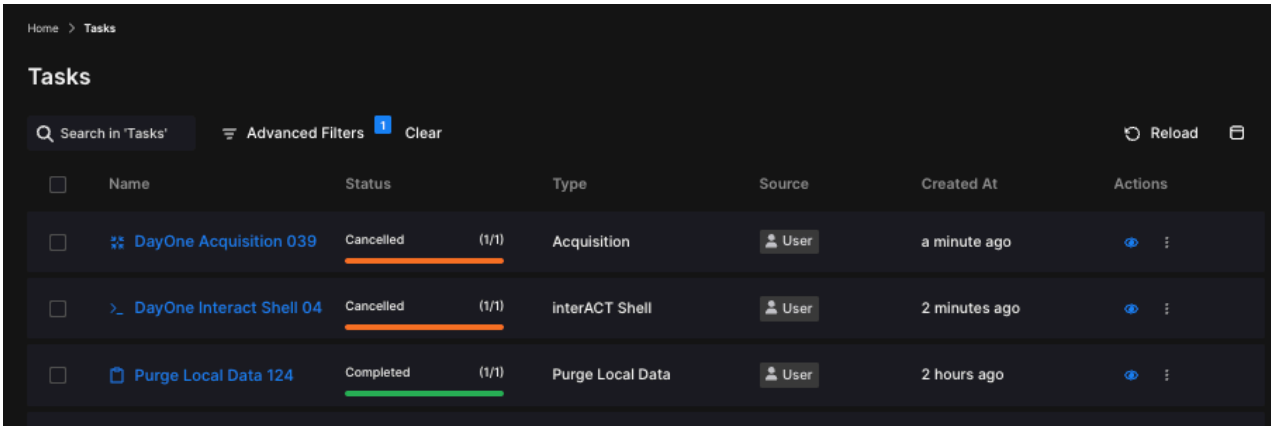
Task Cancellation and Deletion: Bulk actions

Selecting **Cancel Task** will cancel both the **Assigned** and **Processing** tasks listed above as soon as the user selects 'Yes' in the warning message that appears:



Task Cancellation and Deletion: Warning message

Both tasks will then be marked with the status **“Cancelled.”**

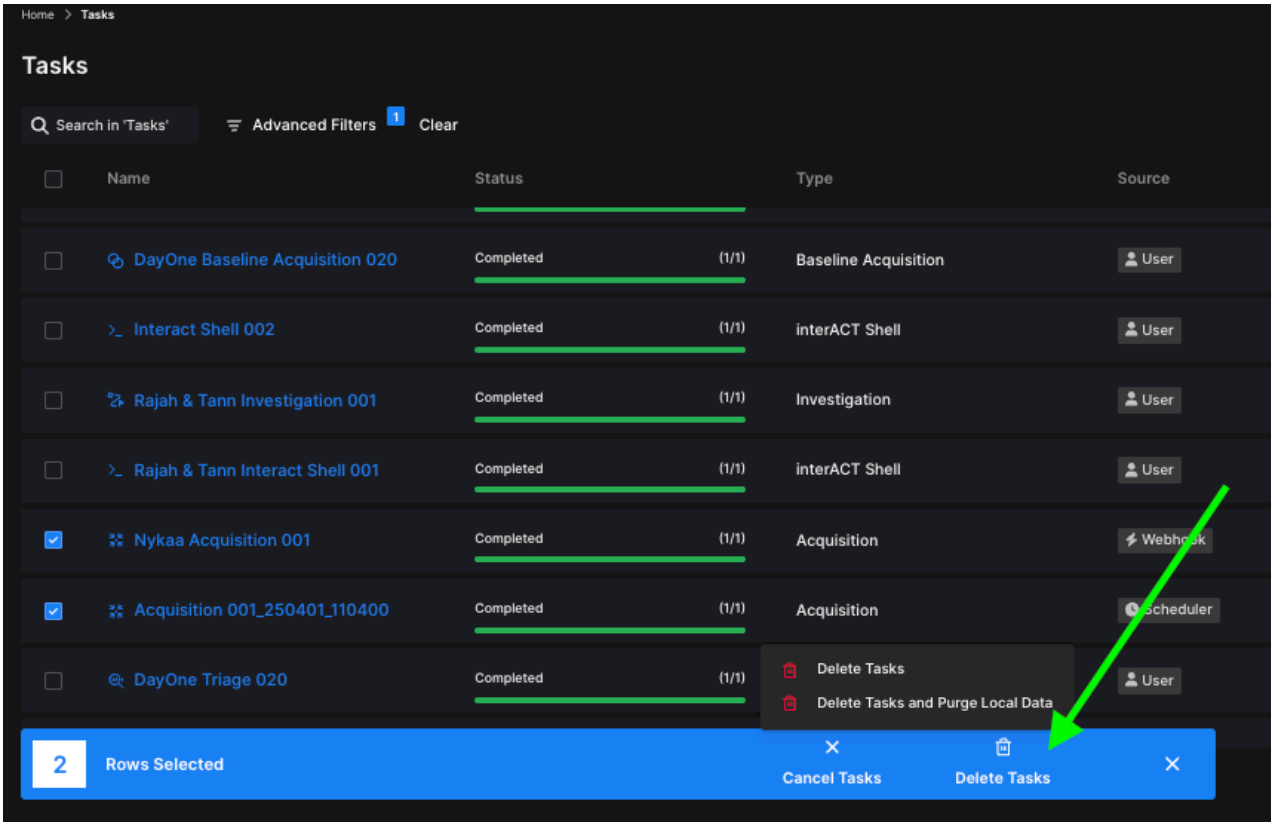


Task Cancellation and Deletion: Task cancelled

Bulk Task Deletion

In addition to cancellation, users can now delete multiple tasks in one action. When selecting the **Delete Tasks** option from the bulk action bar, two deletion modes are available:

- **Delete Task Only**
  Removes the task from the console, but leaves any associated data intact on the endpoint.
- **Delete and Purge Local Data**
  Deletes the task and removes all associated local data from the target asset(s).



Task Cancellation and Deletion: Bulk operations

Confirmation Warning for Large-Scale Tasking

To safeguard against unintentional bulk operations, XDR Forensics now includes an automatic confirmation prompt when tasking operations affect **70% or more of your assets**.

How It Works

Whenever a task—such as **isolation**, **acquisition**, or **triage**—is configured to target a large portion of your asset inventory, XDR Forensics will **pause execution** and display a confirmation dialog. This gives you a chance to review and confirm the action before it proceeds.

This safeguard also applies to **API-based tasking**, providing an additional layer of protection for automated workflows.

Benefits

- **Accidental Prevention**
  Prevents disruptive actions, such as isolating thousands of endpoints due to misconfiguration or scripting errors.
- **Operational Awareness**
  Reinforces deliberate decision-making before initiating potentially high-impact operations.
- **Consistent Safety**
  Applies equally to both console-initiated and API-triggered tasks.

> This feature ensures safer operation at scale, giving you greater confidence when managing thousands of assets in XDR Forensics.

Auto Tagging & Tags

How to automatically tag your assets based on simple conditions.

Overview

Conducting cybersecurity investigations and digital forensics at scale requires a well-structured classification of your assets.

Understanding the number and types of assets, such as web servers, domain controllers, or application servers, significantly reduces response time. This enables you to focus on specific groups of devices within your network, ultimately enhancing situational awareness during an investigation.

How it works

Auto Tagging is a feature of XDR Forensics that lets you automatically tag assets based on conditions such as:

- Existence of a file or directory
- Existence of a running process
- Hostnames, IP addresses, and Subnets
- Custom osquery conditions

Additionally, you can seamlessly combine conditions using AND/OR logic alongside environment variables for greater flexibility.

This feature can be enabled or disabled from the Auto Asset Tagging section in `Settings>Features>Auto Asset Tagging.`

Once enabled, any newly added asset will automatically be assigned a task to query the Auto Tagging conditions. Based on the results, XDR Forensics will apply the appropriate Tag Name to the asset.

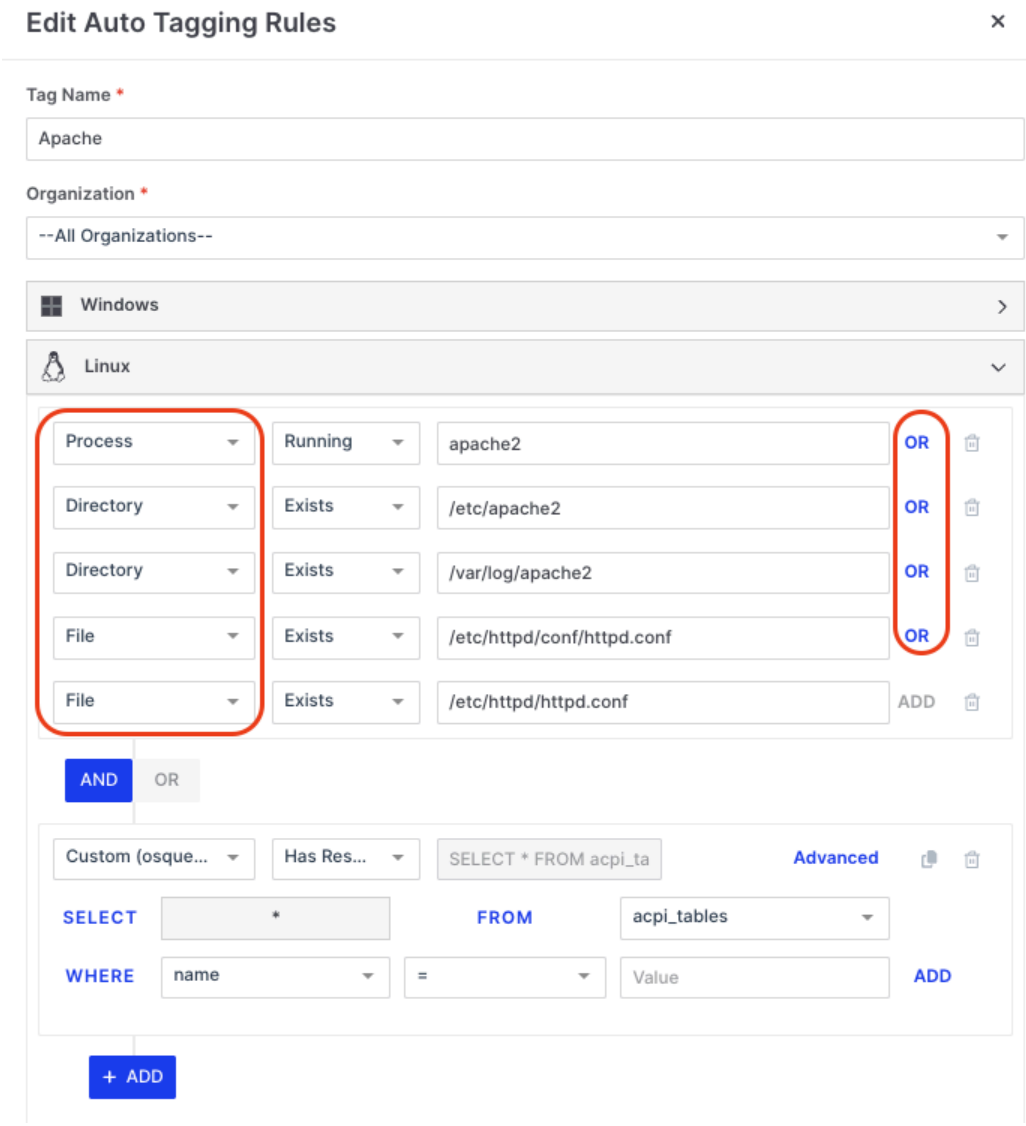
If you need to re-run tagging on all assets, you can do so by clicking the "Run Now" button on the Auto Tagging page. Alternatively, you can run the tagging process for individual assets from the Asset page or select multiple assets and execute the task using the Bulk Action feature.

Auto Tagging can be saved in XDR Forensics Libraries, specifically for individual organizations or universally across all organizations. This capability enables users to create and apply incident-specific Auto Tags selectively, thereby avoiding unnecessary use or exposure of a rule outside the intended organizational context.

There are a number of 'out-of-the-box' supported Auto Tags such as those listed below, but as we now know, you can also create custom tags whenever you need them:

- Apache
- Redis
- Mysql
- Rabbitmq
- Docker
- Kubernetes
- Domain Controller
- IIS Web Server
- Web Server
- Mail Server
- MSSQL Server

When we examine the Auto Tag conditions set for tagging an Apache Server, we can see that the XDR Forensics Responder will evaluate five conditions, all of which are independent of each other, as the OR switch is active. So, if any one of these conditions exists, the Apache Tag will be applied to the asset:

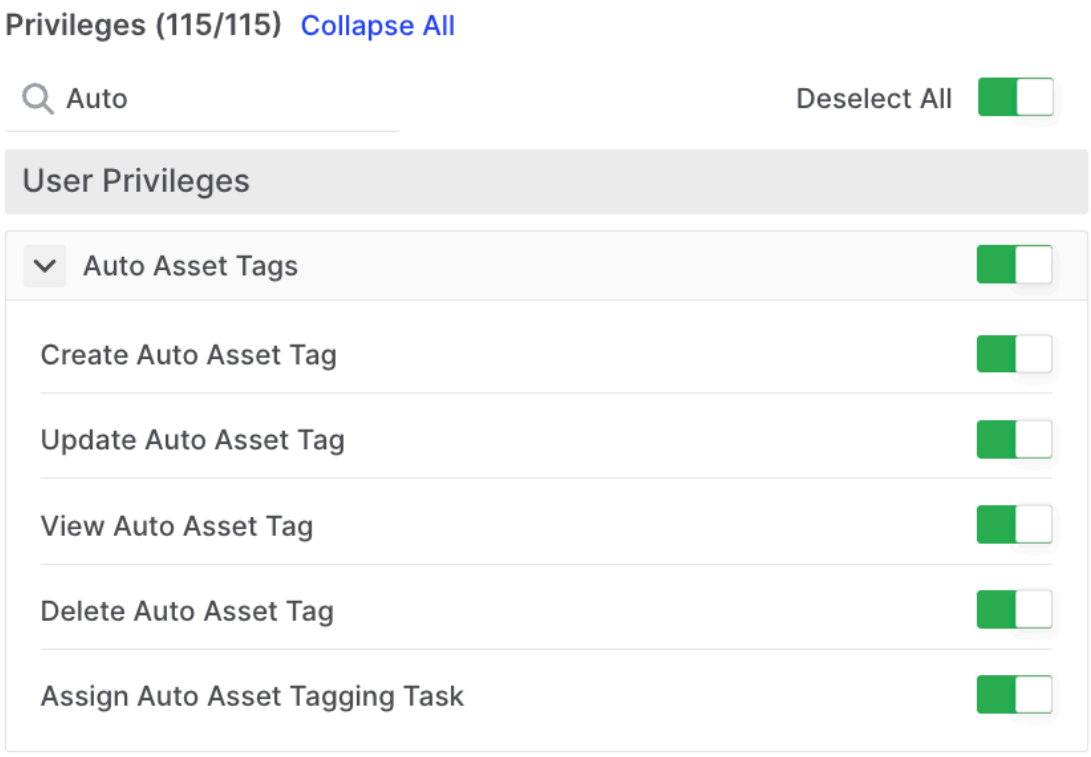


Auto Tagging & Tags: Conditions

It is possible for a user to create, edit, and delete the parameters shown below, but only if they have permission to do so:

Parameter	Matching Criteria	Value
Process	Running / Not Running	Process name or wildcard
File	Exists / Not Exists	File name or wildcard
Directory	Exists / Not Exists	Directory name or wildcard
Hostname	Is / Contains / Start With / End With	Hostname or wildcard
IP Address	Is / Contains / Start With / End With	IP Address or wildcard
Subnet	Is / Contains / Start With / End With	Subnet or wildcard
osquery	osquery format supported	osquery format supported

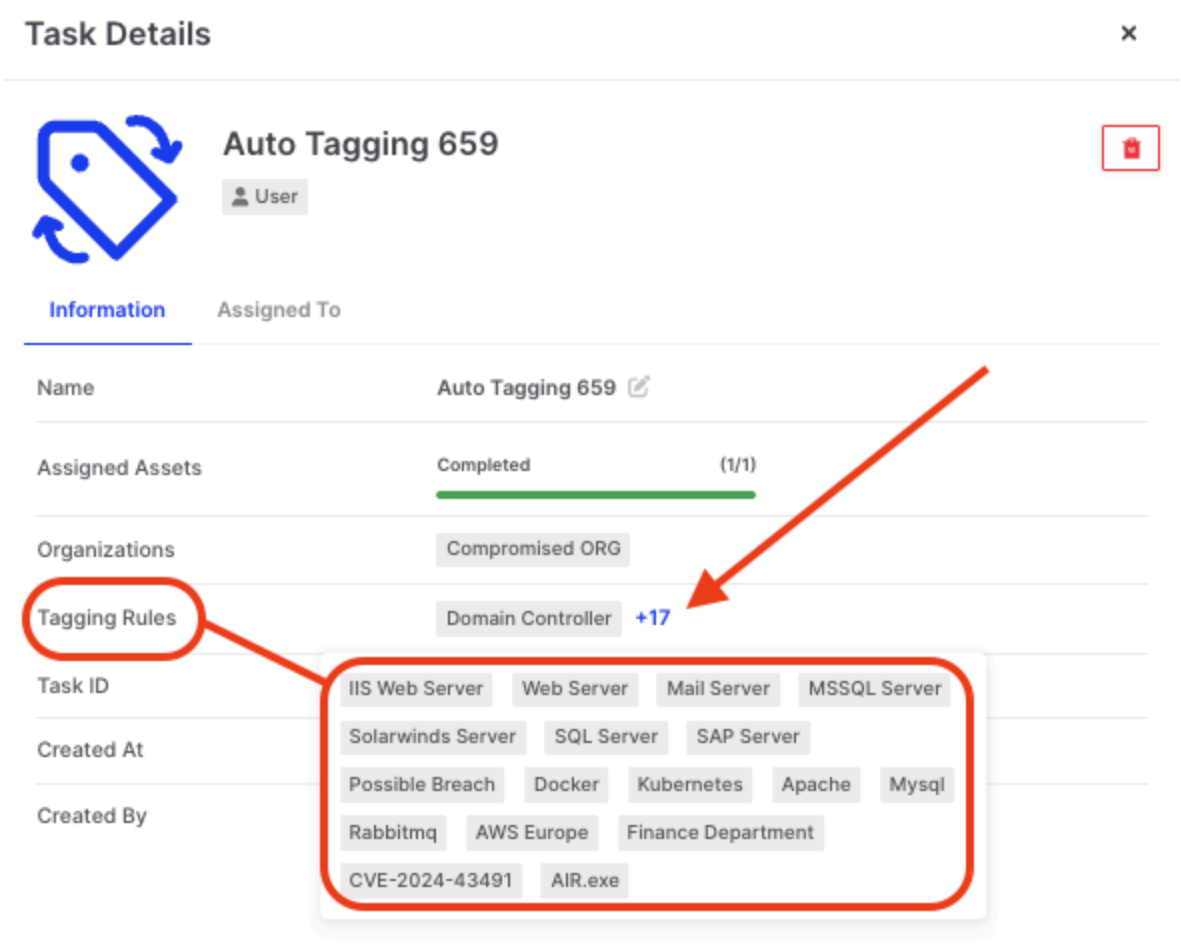
XDR Forensics has very granular permission control over Users and Roles, and within Roles, there are currently over 114 individually configurable privileges. Six of these allow Global Administrators to determine what users can do within the Auto Asset Tagging feature:



Auto Tagging & Tags: User Privileges

Read more about how XDR Forensics uses **Auto Tagging** to speed up your investigations here:
The Power of Auto Asset Tagging in DFIR ↗

Any Auto Tags used in a Tasking Assignment are displayed under the Information tab in the Task Details window. In the example below, we can see that the Tagging Rule for Domain Controller has been run along with 17 others that are related by clicking on the '**+17**' link:



Auto Tagging & Tags: AAT rules run in a task shown in Task Details

Tags

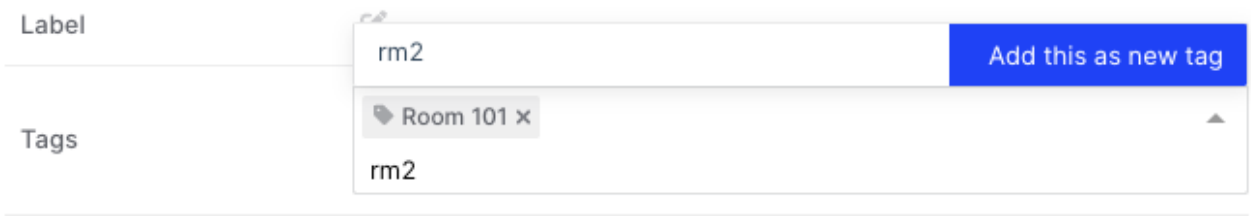
When you need to perform actions on multiple assets, the **Tagging** feature makes management easier by allowing you to group assets based on assigned tags. This helps streamline bulk operations and improves visibility across your environment.

Manual tagging is explained below. **Auto (Asset) Tagging**, as introduced on the previous page, automatically tags assets immediately after the XDR Forensics Responder is deployed. Auto Tagging can also be re-run at any time to reflect changes or updates.

To create Tags for your assets:

1. Navigate to “Assets” in the Main Menu and select the assets to which you want to assign tags.

2. Click in the dialogue box and search the drop-down list for an existing Tag or simply type a new Tag name and then click "Add this as a new tag".



Tags: Adding 'rm2' as a new Tag

3. From the same dialogue box, you can delete Tags by selecting the cross.

> *Note: You can add multiple tags to a single asset.*

4. Now you can filter all of your assets by their tags, making it easy to view specific groups of assets and apply relevant acquisition or triage tasks just to them.

Compare

Baseline analysis with Compare

The Compare feature enables proactive forensics through baseline analysis, allowing investigators to focus on forensic evidence from the earliest stages of an investigation. Using a patent-pending approach, it identifies and highlights forensic artifacts—added, modified, or deleted—between asset snapshots.

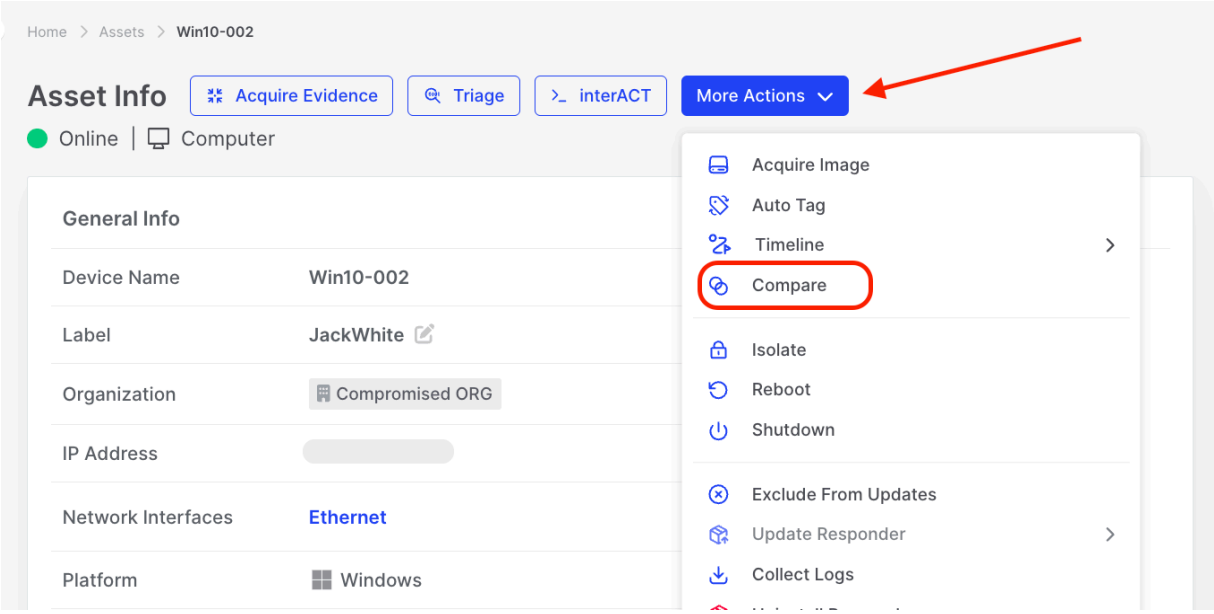
This analysis, completed in just 5 seconds, enhances security by addressing vulnerabilities before they can be exploited, without disrupting ongoing operations. The Compare feature supports both standard and offline acquisitions, providing detailed metadata to help investigators comprehensively understand potential security risks.
Compare analysis is performed directly on the Console, eliminating the need for direct access to assets.

The scope of baseline analysis is strategically based on areas commonly abused by attackers. These include:

macOS	Linux	Windows
AutoLoadedProcs	System	System
ChromeExtensions	CronJobs	NetworkAdapters
DiskEncryptions	DNSResolvers	Hosts
ETCHosts	Hosts	AutorunsServices
ETCProtocols	IPRoutes	AutorunsRegistry
ETCServices	IPTables	AutorunsScheduledTasks
GatekeeperApps	KernelModules	AutorunsStartupFolder
InstalledApps	Mounts	InstalledApplications
KextInfo	NetworkInterfaces	Drivers
LaunchdOverrides	SystemArtifacts	FirewallRules
SipStatus	Users	
SysExtInfo		

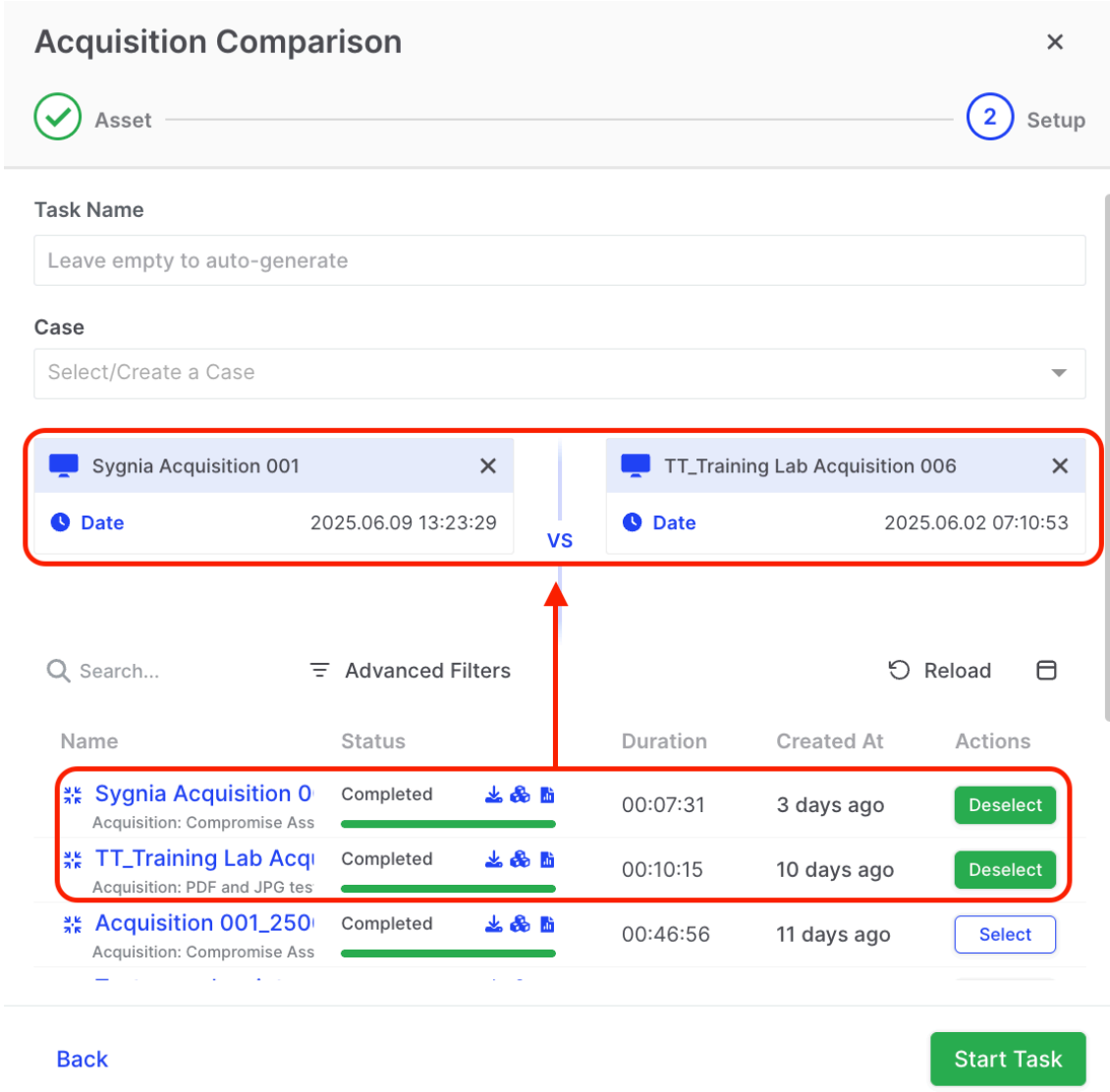
How to Use Baseline Analysis with Compare

1. **Select the Asset for analysis.**
2. **Initiate Compare Task:**
   - Navigate to 'Compare' under Asset Actions.



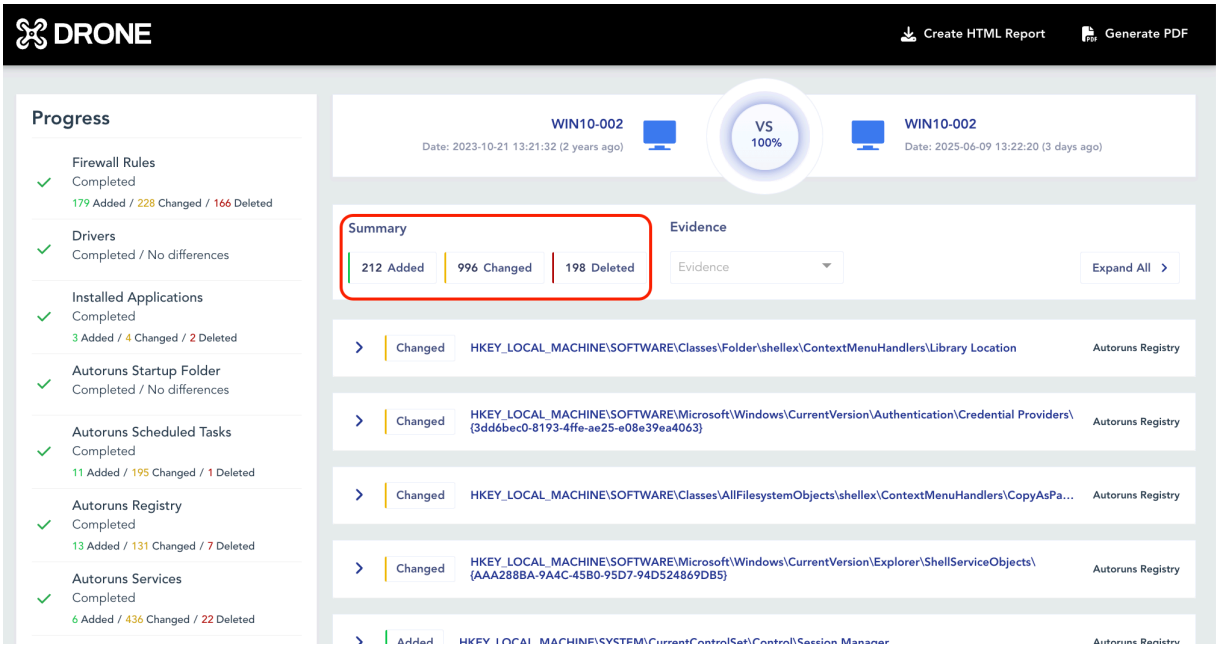
Compare: Initiating Compare Analysis under Asset Actions

3. **Specify the Acquisitions to Compare**
   - Specify two acquisitions to compare or create a new baseline by clicking "Acquire Baseline".



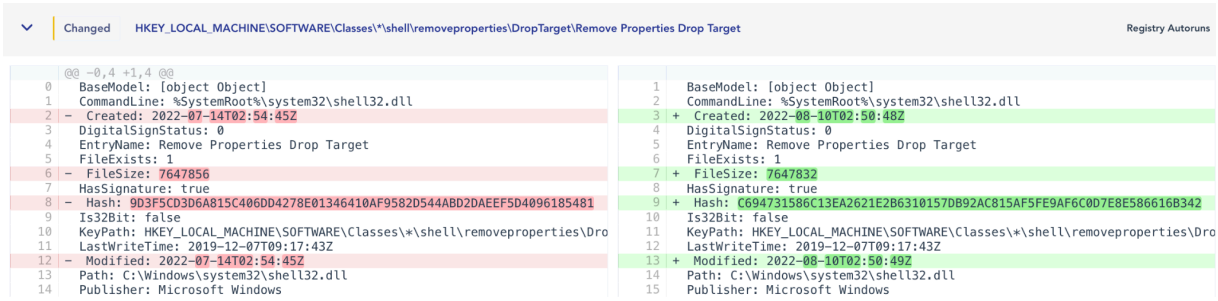
Compare: Selecting Acquisitions to Compare

4. **Review the Results**
    - Once the task is complete, review the results DRONE report.



Compare: Reviewing Results

- Explore the fine-grained details on the property level.



Compare: Explore items Added, Changed, Deleted

5. **Integrate with Investigations:**
    - Leverage the insights gained from the comparison analysis to compare and inform ongoing digital investigations.

By incorporating Baseline Analysis with Compare into the DFIR process, you empower your team with a proactive and efficient approach to identifying and mitigating potential security risks. This feature is a valuable asset in maintaining a robust cybersecurity posture.

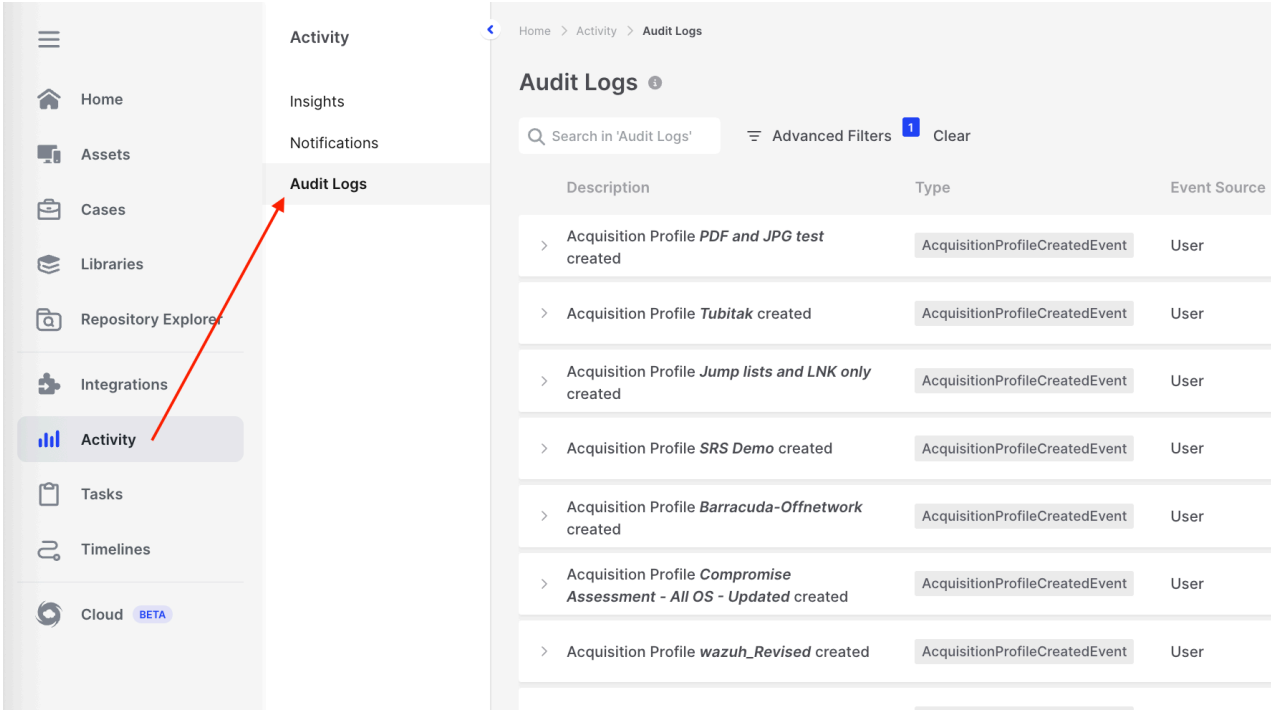
Console Audit Logs

Overview

The Audit Log feature is designed to provide detailed and comprehensive logging information about all actions performed within the platform. This ensures a solid' chain of custody', transparency, and accountability, and it also supports investigations by maintaining a clear record of all activities.

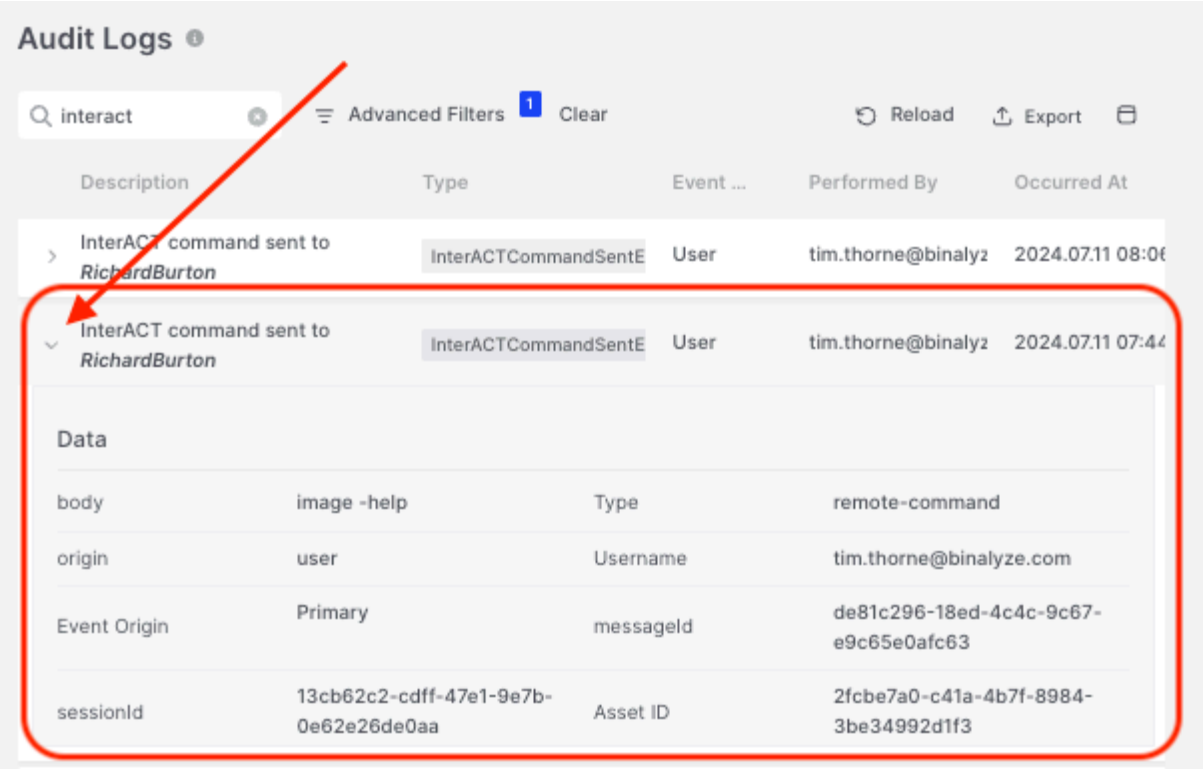
The Audit Logs can be searched, filtered, and exported at the Organization level.

Audit Logs are accessed via the Activity button in the main menu and then from Audit Logs in the secondary menu:



Console Audit Logs: Activity > Audit Logs

Each log entry can have additional properties to view, such as what individual interACT command was run and by whom:



Console Audit Logs: Expand log entry

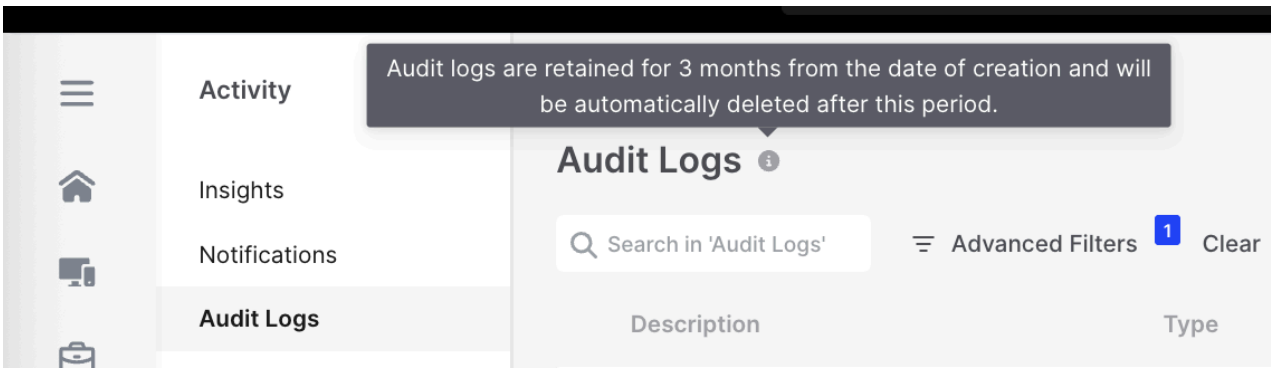
Important to know:

The audit log retention policy is designed to optimize platform performance and data management. Users are encouraged to regularly back up their audit logs to prevent data loss.

Audit Log Retention

- **Exporting Logs:** As of version 4.19, when exporting audit logs from the console user interface, only logs from the last three months will be included in the export.
- This ensures that users will always have access to the most recent audit logs in the UI while maintaining the 'historical logs'.

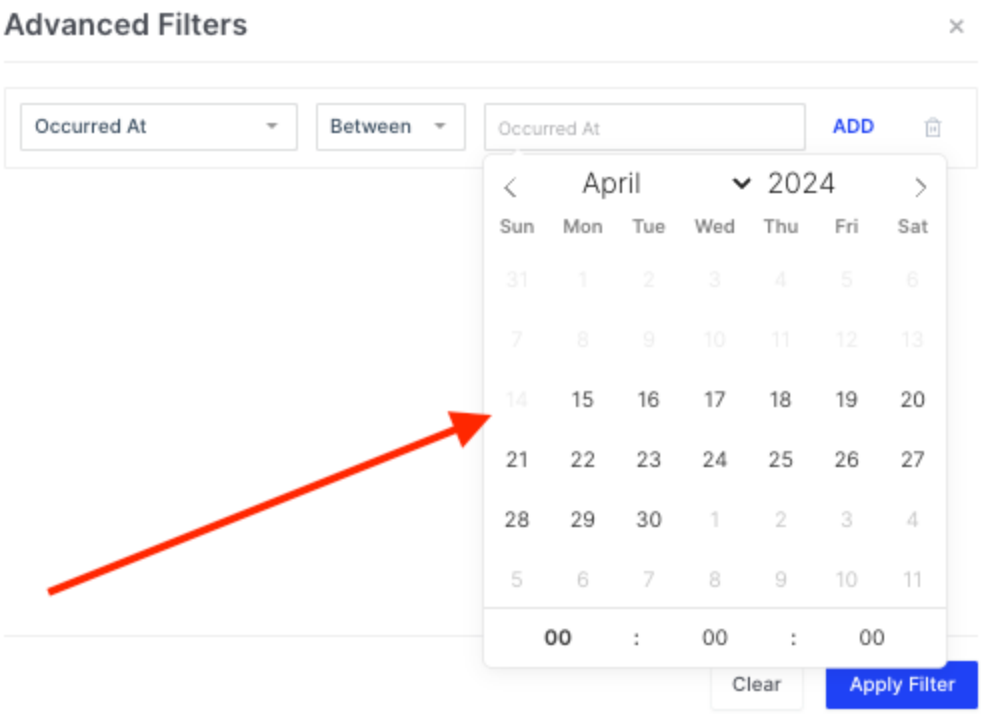
Information about the retention policy is displayed on the Audit logs page in the Tool Tip as shown below:



Console Audit Logs: Warning tooltip

This message says, "Audit logs are retained for 3 months from the date of creation and will be automatically deleted after this period."

The Audit Log advanced filter date and time selection columns are limited to the last 3 months:



Console Audit Logs: Advanced filter

Audit Log Export

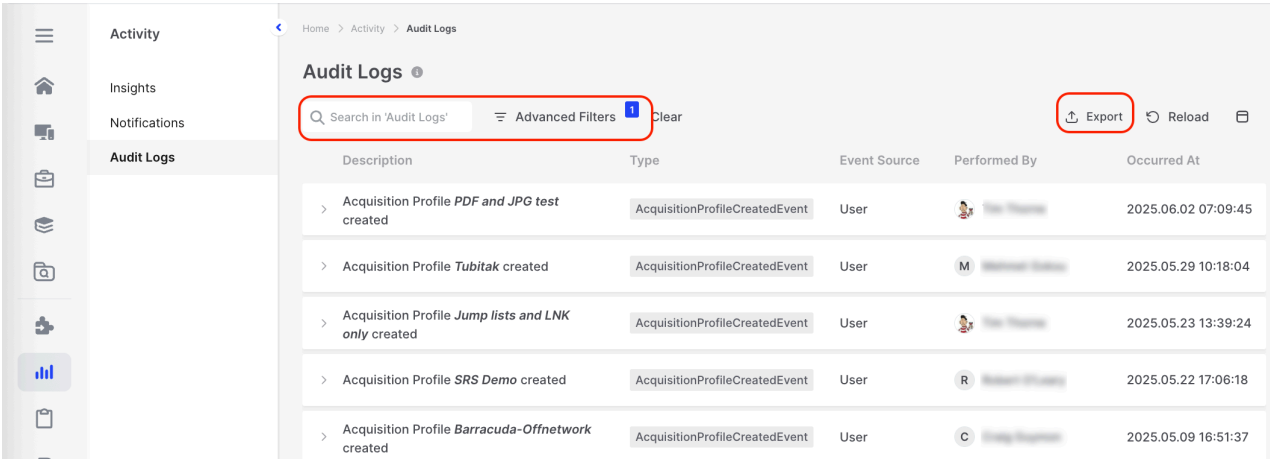
There are three options to export your Audit Logs:

- **Export Logs:** Export the logs directly from the console's Audit Log page.
- **Send Logs to Syslog Server:** Utilize the feature in settings to send logs to your Syslog Server.
- **Use the API:** Refer to our API documentation to retrieve Audit Logs.

See below:

At the **Organization level,** users can search and filter audit logs within the Console and easily export them in CSV format.

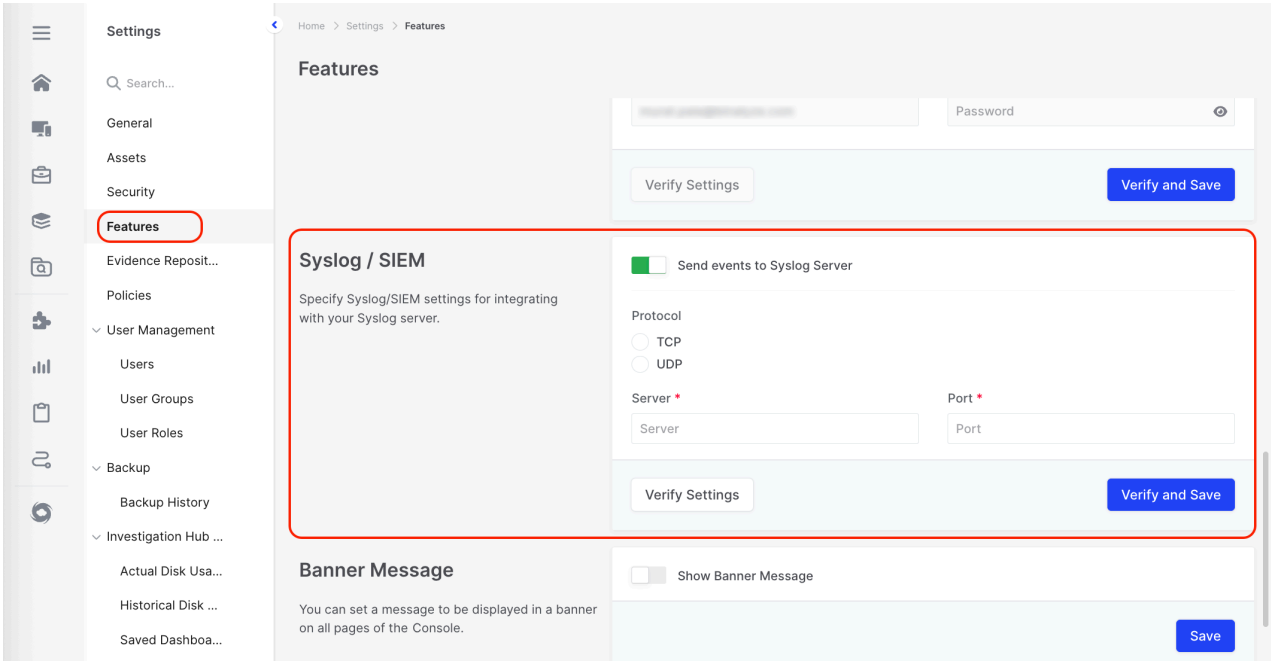
To do this, use the Export button located at the top right corner of the user interface. This action will generate a ZIP file containing the filtered audit logs in CSV format, making it convenient for further analysis and record-keeping:



Console Audit Logs: Search, Advanced Filter, and Export options

Syslog/SIEM Integration

All logs are forwarded as Common Event Format (CEF) messages when integrated with Syslog/SIEM which is found at Settings > Features > Syslog/SIEM, all logs are forwarded as Common Event Format (CEF) messages.



Console Audit Logs: Integrated with Syslog/SIEM

DRONE

Analyzers

Cross Platform Analyzers

Browser History Analyzer

The Browser History Analyzer inspects browser histories for entries that might be indicative of suspicious activity.

The term "browser history" refers to the log of web pages a user has visited in their web browser over a certain time, including URLs, page titles, and typically the date and time of each visit.

In the context of malware and cybersecurity, browser history can provide insights in several ways:

1. Indicators of Compromise or Infection:
    - Malicious URLs: The presence of known harmful URLs in the history could suggest exposure to malware or phishing.
    - Redirection Chains: Unusual sequences of redirects may indicate adware or similar malware.
    - Search Queries: Searches like "how to remove xyz malware" can signal potential infection or cybersecurity concerns.
2. Malware Propagation:
    - Browser Hijackers: Changes in browser settings reflected in history, like altered homepages or search engines, can suggest hijacker malware.
    - Drive-by Downloads: Visits to sites that exploit browser vulnerabilities to download malware can be identified.
3. Exfiltration & Espionage:
    - Spyware & Information Stealers: Malware that captures browser history to glean personal interests, habits, or for targeted attacks.
4. Evasion and Anti-Forensics:
    - Clearing Browser History: Automatic deletion of history by malware to cover its tracks.
    - Selective Deletion: Targeted removal of specific entries related to malicious activities by sophisticated malware.
5. Manipulation for Fraud:
    - Click Fraud: Repeated visits to ad-heavy pages could suggest click fraud.
    - Login Page Imitation: Frequent access to fake login pages might indicate phishing attempts.

Dynamo Analyzer

Dynamo Analyzer will parse the database in the .ppc file generated as the result of a Windows, Linux, or macOS collection tasking assignment assignment and highlight suspicious entries.

XDR Forensics's existing YARA integration can scan the filesystem and memory of assets but it cannot on its own parse the Window's registry database, scheduled tasks, DNS cache, WMI, firewall rules and other persistence methods or configuration databases - all areas of systems often abused by malicious actors.

Imagine having installed an outdated and vulnerable version of popular software, Dynamo will be able to warn you about it

Or perhaps a crypto miner domain in DNS cache records, or a scheduled task executing suspicious extension in the TEMP folder, and so on and so on.

Fileless Malware Techniques

Consider a scenario where malware stores its payload in a base64 encoded format within a registry key. It then uses a scheduled task to run and inject the payload directly into memory. Dynamo is designed to detect such fileless attack techniques, thereby bringing a considerable uplift to scanning capabilities within XDR Forensics.

Conclusion

Dynamo Analyzer is not just another tool, it's a comprehensive solution designed to identify and alert you about various forms of suspicious activities. By extending its capabilities beyond what traditional tools like YARA offer, Dynamo provides a more robust and nuanced approach to XDR Forensics and cybersecurity in general.

Generic WebShell Analyzer

The WebShell Analyzer is designed to detect suspicious Web Shells, which are scripts allowing remote server access and control. Typically deployed as web-based interfaces, webshells act like a shell environment and are developed in various scripting languages like PHP, ASP, JSP, Python, and Perl, depending on server compatibility.

Webshells are a significant threat due to several reasons:

1. Remote Access and Control: They grant attackers a robust platform to remotely execute commands, manage files, and access databases through a browser.
2. Stealth: Webshells often mimic legitimate server files in name and appearance, and may contain obfuscated code to evade detection.
3. Versatility: They can be utilized for various malicious purposes, including data theft, server defacement, or as a base for broader attacks.
4. Authentication Bypass: Webshells enable direct server system access, circumventing standard authentication.
5. Network Pivot Point: Attackers can use a compromised server as a base to infiltrate and exploit other network systems.
6. Persistent Access: Webshells can provide ongoing access, even after the initial vulnerability is patched, unless detected and removed.

Webshells are typically uploaded via:

- Exploiting vulnerabilities in web applications, like file upload flaws.
- Utilizing weak or default administrative credentials for file uploads.
- Gaining FTP or SSH access to directly upload the webshell.

MITRE ATT&CK Analyzer

MITRE ATT&CK is a globally accessible knowledge base of adversary tactics and techniques, based on real-world observations.

The ATT&CK knowledge base serves as the foundation for developing specific threat models and methodologies. These threat models and methodologies hold significance across various sectors, including private industry and government organizations, and they form a cornerstone for communities dedicated to cybersecurity products and services.

ATT&CK is an open platform, and its integration into XDR Forensics delivers additional benefits by utilizing up-to-the-minute YARA rules for detecting potential IoCs (Indicators of Compromise) or TTPs (Tactics, Techniques, and Procedures).

DRONE's MITRE ATT&CK implementation uses YARA scanning across various folders on the asset and across the running processes. These scans are carried out with rules that the Cisco XDR Forensics threat-hunting DFIR team has crafted, and XDR Forensics will check for updated rules every couple of hours. New rules will be pushed automatically to the XDR Forensics installation.

The DFIR team also defines scan locations. Here are a few examples;

- Recycle bin folders
- User folders and sub-directories
- Temp directories
- Program Files directories
- System32 directory

> The MITRE ATT&CK scanner runs on the actual asset and is not concerned with the associated triage or acquisition tasking. Nor is it scanning the collected data or case report.

Read more about how XDR Forensics uses our MITRE ATT&CK integration to deliver insights here; **Focus investigations with MITRE ATT&CK insights ↗**

MITRE ATT&CK Analyzer changelog

This page tracks the updates/changes to XDR Forensics's MITRE ATT&CK Analyzer

10.6.2 (26/09/25)

MITRE ATT&CK Analyzer / YARA

- Added detection for binaries designed to dump credentials using WerfaultSecure, enhancing credential access threat identification.
- Added detection for binaries designed to create processes as Protected Process Light (PPL), improving defense evasion monitoring.
- Enhanced detection of potentially obfuscated VBScript code commonly used for evasion purposes.
- Other smaller improvements including rule updates, quality of life and false positive fixes.

Sigma

- DRONE has been updated with the most recent Sigma rule updates from SigmaHQ and Hayabusa repositories.

10.6.1 (25/09/25)

MITRE ATT&CK Analyzer / YARA

- Added improved detection for Quasar RAT by expanding signature strings focused on keylogger functionality and reverse proxy capabilities, enhancing detection reliability for this malware family.
- Other smaller improvements including rule updates, quality of life and false positive fixes.

Dynamo Analyzer

- Improved identification of hacker and remote monitoring management (RMM) tool names by adding matched field tracking for more accurate reporting in amcache, applications, browser history, and downloads modules.
- Refined reporting to utilize matched fields where available, enhancing clarity of forensic data outputs across multiple embedded rules.
- Removed redundant checks for hacker commands in prefetch rules to streamline processing.
- Implemented enhanced date handling in process monitoring, adding explicit creation time tracking for more precise temporal context.
- Enhanced identification of various hacker tool names commonly found in forensic evidence.

Sigma

- DRONE has been updated with the most recent Sigma rule updates from SigmaHQ and Hayabusa repositories.

10.6.0 (19/09/25)

MITRE ATT&CK Analyzer / YARA

- Improved detection of base64 encoded Powershell commands.
- Updated detection for suspicious Powershell command abbreviations frequently used by malware.
- Enhanced identification of Rclone, a tool used for syncing files with cloud storage services in ransomware campaigns, expanding platform coverage to Windows, Linux, and macOS.
- Other smaller improvements including rule updates, quality of life and false positive fixes.

Dynamo Analyzer

- Added SRUM Analyzer to parse and enrich SRUM data.

10.5.5 (17/09/25)

MITRE ATT&CK Analyzer / YARA

Added detection for a malicious script involved in a supply chain attack on over 40 npm packages. The malware uses TruffleHog to steal developer credentials and propagate across repositories.

10.5.4 (16/09/25)

Dynamo Analyzer

- Enhanced identification of various hacker tool names commonly found in forensic evidence by adding detection of hacker tool names in browser history analysis.
- Improved the hacker tool names database with additional entries including support for Rclone, a command-line program for managing and synchronizing files across cloud storage and local file systems.

10.5.3 (16/09/25)

MITRE ATT&CK Analyzer / YARA

- Added detection for GodRAT malware, a Gh0st RAT-based Remote Access Trojan targeting financial institutions.
- Enhanced detection for macOS Atomic (AMOS) infostealer variants, covering both x64 and ARM64 architectures with specific decoding function signatures.

10.5.2 (14/09/25)

MITRE ATT&CK Analyzer / YARA

- Added updated detection for TinyShell, an open source UNIX backdoor, improving coverage with expanded MITRE ATT&CK mapping and enhanced pattern matching.
- Enhanced detection of execution of Windows commands via LNK shortcut files, commonly used for initial access, persistence, and execution by threat actors abusing living-off-the-land binaries (LOLBINs).

10.5.1 (12/09/25)

MITRE ATT&CK Analyzer / YARA

- Added detection for Betruger backdoor malware associated with RansomHub group.
- Added detection for files with CVE-pattern filenames that may indicate proof-of-concept exploits or malicious files mimicking CVE identifiers.
- Updated detection for SoftPerfect Network Scanner, a tool utilized for conducting network scans, now supporting Windows and Darwin platforms with improved inspection of file types.

10.5.0 (10/09/25)

Dynamo Analyzer

- Enhanced identification of various hacker tool names commonly found in forensic evidence.
- Added detection for additional remote access management (RMM) tools expanding the scope of RMM tool identification.

MITRE ATT&CK Analyzer / YARA

- Improved detection for base64 encoded PowerShell variables commonly observed in obfuscated malicious scripts.
- Refined JavaScript obfuscation detection targeting the jsobfuscator tool, improving accuracy by adjusting pattern matching for obfuscated code structures.
- Applied minor updates and corrections to YARA rule filtering paths to optimize Linux path scanning for detections.
- Other smaller improvements including rule updates, quality of life and false positive fixes.

Sigma

DRONE has been updated with the most recent Sigma rule updates from SigmaHQ and Hayabusa repositories.

10.4.3 (04/09/25)

MITRE ATT&CK Analyzer / YARA

- Fixed a false positive detection for API hashing techniques used in reflective code loading for defense evasion purposes.

10.4.2 (03/09/25)

MITRE ATT&CK Analyzer / YARA

- Added detection for PromptLock ransomware, the first known AI-powered ransomware using a local OpenAI GPT model via Ollama API to dynamically generate malicious Lua scripts for filesystem enumeration, file inspection, data exfiltration, and encryption.
- Enhanced detection of Mimikatz credential access tool commands, with improved context to reduce false positives related to antivirus signatures.
- Other smaller improvements include rule updates, quality-of-life enhancements, and fixes for false positives.

10.4.0 (28/08/25)

Dynamo Analyzer

- Enhanced identification of various hacker tool names commonly found in forensic evidence.

MITRE ATT&CK Analyzer / YARA

- Added detection for APT36 (Transparent Tribe) Linux campaign targeting Indian BOSS systems using weaponized autostart files and spear-phishing to maintain covert access against Indian government entities.
- Added detection for CANONSTAGER, a side-loaded DLL launcher used to decrypt and execute payloads in memory.
- Added detection for SOGU.SEC backdoor (PlugX) involved in a PRC-nexus espionage campaign targeting diplomats in Southeast Asia, deployed by UNC6384.
- Enhanced detection of files with double extensions used for masquerading to hide true file types and trick users into execution, now covering Windows, Linux, and Darwin platforms.

Sigma

- DRONE has been updated with the most recent Sigma rule updates from SigmaHQ and Hayabusa repositories.

10.3.3 (19/08/25)

Dynamo Analyzer

- Enhanced identification of various hacker tool names commonly found in forensic evidence.

MITRE ATT&CK Analyzer / YARA

- Added detection for OptiTune remote access software, which adversaries may misuse for remote access.
- Added detection for SimpleHelp remote control software, sometimes abused by attackers for remote access.
- Enhanced detection for Splashtop Streamer service, identifying adversary use of legitimate remote access software for command and control channels.
- Added detection for Syncro remote control software, known to be occasionally misused by attackers.
- Added detection for Dark-kill rootkit, which terminates EDR processes via kernel callbacks and ZwTerminateProcess.
- Improved detection for Angry IP Scanner, a network scanning tool used for active IP discovery and port scanning.
- Broadened identification of process monitoring and sandbox evasion tools, including Process Explorer, Process Hacker, and multiple endpoint protection services.
- Other smaller improvements include rule updates, quality-of-life enhancements, and fixes for false positives.

Sigma

- DRONE has been updated with the most recent Sigma rule updates from SigmaHQ and Hayabusa repositories.

10.3.2 (16/08/25)

MITRE ATT&CK Analyzer / YARA

- Added detection for the SoundBill shellcode loader used by the UAT-7237 APT group.
- Enhanced detection of XOR-encoded strings used in Cobalt Strike Beacon DLLs to better identify this popular adversary tool.

10.3.1 (15/08/25)

MITRE ATT&CK Analyzer / YARA

- Added detection for multiple malware variants attributed to the UAT-5647 (RomCom) threat actor, including MeltingClaw, SingleCamper, and ShadyHammock.

10.3.0 (14/08/25)

MITRE ATT&CK Analyzer / YARA

- Added detection for RustyClaw, a Rust-based malware downloader.
- Added detection for SnipBot (RomCom 5.0), a RomCom malware variant with unique obfuscation and post-infection capabilities that allow command execution and downloading additional modules on victim systems.
- Added detection for Mythic C2 dynamic HTTP shellcode loader used in attacks leveraging WinRAR vulnerabilities CVE-2025-8088 and CVE-2025-6218.
- Other smaller improvements include rule updates, quality-of-life enhancements, and fixes for false positives.

Sigma

- DRONE has been updated with the most recent Sigma rule updates from SigmaHQ and Hayabusa repositories.

10.2.5 (12/08/25)

MITRE ATT&CK Analyzer / YARA

- Added detection for potential exploitation of WinRAR path traversal vulnerabilities CVE-2025-8088 and CVE-2025-6218 targeting initial access techniques.

10.2.4 (11/08/25)

MITRE ATT&CK Analyzer / YARA

- Added detection for padded LNK files designed to exploit the ZDI-CAN-25373 vulnerability targeting Windows systems.

10.2.3 (10/08/25)

MITRE ATT&CK Analyzer / YARA

- Added detection for HeartCrypt-packed files, identifying this packer-as-a-service used by Windows-based malware since early 2024.
- Other smaller improvements, including rule updates, quality of life, and false positive fixes

10.2.2 (08/08/25)

MITRE ATT&CK Analyzer / YARA

- Added detection for indicators of the AK47 toolset, including a custom command-line tool abusing a legitimate third-party signed driver to terminate processes used by APT Storm-2603, linked to ToolShell exploitations.
- Enhanced detection of VBScript code containing obfuscated content aimed at evading detection.
- Added detection for a malicious driver designed to disable Windows Defender and observed in Akira ransomware campaigns.
- Other smaller improvements, including rule updates, quality of life, and false positive fixes

Dynamo Analyzer

- Enhanced identification of various hacker tool names commonly found in forensic evidence.

Sigma

- Added new detection for PowerShell commands collecting sensitive file types across directories, a common data exfiltration technique used by threat actors.
- DRONE has been updated with the most recent Sigma rule updates from SigmaHQ and Hayabusa repositories.

10.2.1 (06/08/25)

Dynamo Analyzer

- Added detection of scheduled tasks executing at high frequency, indicating potentially suspicious activity on Windows systems.
- Enhanced identification of various hacker tool names commonly found in forensic evidence.

MITRE ATT&CK Analyzer / YARA

- Enhanced IOC detection for exploitation attempts related to SharePoint CVE-2025-53770, including web log and URI artifacts, as well as ASPX webshell and compiled ASPX webshell usage and associated MachineKey extraction binaries.
- Added detection for the AutoColor backdoor malware, characterized by advanced evasion techniques, custom encryption, and persistence mechanisms for stealthy remote access on Linux platforms.
- Added detection for BlackSuit ransomware.
- Added detection for Chaos ransomware.
- Added detection for Storm-2603 Web shell that utilizes sockets and DNS to receive and execute commands from its command and control infrastructure.

Sigma

- DRONE has been updated with the most recent Sigma rule updates from SigmaHQ and Hayabusa repositories.

10.2.0 (29/07/25)

Dynamo Analyzer

- Enhanced identification of various hacker tool names commonly found in forensic evidence.

MITRE ATT&CK Analyzer / YARA

- Improved detection of SharePoint server exploitation via CVE-2025-53770, including updated patterns for web logs, URI artifacts, and webshells.
- Added detection for Donut shellcode loader and shellcode, enabling identification of position-independent shellcode used for stealthy in-memory execution of .NET assemblies and other payloads.
- Expanded detection capabilities for Amadey malware version 5.34 through intrinsic pattern recognition and decryption algorithm identification.
- Added rules to identify indicators common across multiple ESXi ransomware variants.
- Introduced detection for Warlock ransomware across Windows and Linux platforms based on unique file and string patterns.
- New detection for ToolShell, a . NET-based webshell used for system information gathering and cryptographic key extraction on Windows systems.
- Added SharpAdidnsdump detection, a tool for dumping Active Directory credentials.
- Introduced detection of dynamic resolving of function addresses from msvcrt.dll, used as a technique for evading detection.
- Added detection for SharpHost, a tool that collects host system and network environment information.
- Introduced detection for Python DLL sideloading attempts, highlighting suspicious Python libraries that may facilitate hijacking of execution flow.
- Other smaller improvements include rule updates, quality of life enhancements, and fixes for false positives.

Sigma

- DRONE has been updated with the most recent Sigma rule updates from SigmaHQ and Hayabusa repositories.

10.1.0 (26/07/25)

Dynamo Analyzer

- Improved detection of suspicious Windows scheduled tasks by expanding monitored command line path patterns.
- Enhanced identification of various hacker tool names commonly found in forensic evidence.

MITRE ATT&CK Analyzer / YARA

- Added new detections for APT Patchwork, including its shellcode loader, remote access trojan, and shellcode runner, to identify related malicious activity.
- Extended Lazarus APT coverage with detections for RootTroy backdoor, malicious implant, trojan, and CryptoBot infostealer, alongside existing detections for RustBucket and other Lazarus malware variants.
- Introduced detection of Nim-based binaries, covering Windows, Linux, and macOS platforms, to flag potentially suspicious executables that require further analysis.
- Added detection for the RouterScan hack tool used for network service discovery and router enumeration.
- Detected LNK files executing PowerShell commands with suspicious paths and those with embedded URLs, highlighting potential initial access techniques.
- Added detection for payloads created by the commercial evasion framework SHELLTER, commonly used to deploy post-exploitation payloads and evade antivirus and EDR solutions.
- Other smaller improvements include rule updates, quality of life enhancements, and fixes for false positives.

Sigma

- DRONE has been updated with the most recent Sigma rule updates from SigmaHQ and Hayabusa repositories.

10.0.5 (22/07/25)

Sigma

- DRONE has been updated with the latest Sigma rule enhancements from the SigmaHQ and Hayabusa repositories, including detection for exploitation attempts of CVE-2025-53770.

10.0.4 (21/07/25)

MITRE ATT&CK Analyzer / YARA

- Added detections for artifacts related to SharePoint server exploitation via CVE-2025-53770, including web log entries, URIs, and ASPX webshells (both source and compiled forms).

10.0.3 (18/07/25)

Dynamo Analyzer

- Added detection for anomalous activity volume in HubSpot users by comparing recent activity to historical baselines.
- Implemented identification of suspicious first-time user activities in HubSpot within the first day of account creation, focusing on potentially destructive or high-privilege actions.
- Introduced detection for rapid successive actions in HubSpot audit logs, flagging potential automated or compromised account behavior based on action frequency and speed.
- Created a comprehensive detector for suspicious activities in HubSpot audit logs, covering user management, API key management, data export, off-hours activity, authentication failures, and configuration changes.
- Enhanced identification of various hacker tool names commonly found in forensic evidence.
- Improved Windows Osquery detection by enriching WinSCP Host Key extraction with associated user information through joined registry and user datasets.

MITRE ATT&CK Analyzer / YARA

- Added detection for RansomHub ransomware, a cross-platform threat known for aggressive encryption and ransom note deployment.
- Added detection for the Veeamp credential dumping tool, targeting SQL databases used by Veeam backup management software.
- Other smaller improvements include detection updates, quality-of-life enhancements, and fixes for false positives.

10.0.2 (15/07/25)

MITRE ATT&CK Analyzer / YARA

- Improved detection of multiple native IIS malware families, including IIS-Raid variants, RGDoor, IIStealer, ISN, IISpy, IISerpent, and other groups targeting IIS servers.
- Added identification of a malicious IIS module used in SEO poisoning attacks.
- Refined detection logic to focus on filesystem context for better accuracy and reduced false positives.
- Other smaller improvements include rule updates, quality-of-life enhancements, and fixes for false positives.

10.0.1 (15/07/25)

Dynamo Analyzer

- Added detection for cached WinSCP host keys in the Windows registry, which may indicate unauthorized use of WinSCP for remote file transfers.

10.0.0 (14/07/25)

Dynamo Analyzer

- Improved detection of scheduled tasks with suspicious extensions by adding exclusions for specific system paths.
- Updated detection logic for services to better identify unusual paths used by adversaries.
- Enhanced identification of various hacker tool names commonly found in forensic evidence.

MITRE ATT&CK Analyzer / YARA

- Added detection for a tool demonstrating multiple methods for bypassing application whitelisting on Windows systems.
- Included rules for identifying Office template injection exploits commonly used for defense evasion.
- Enhanced detection capabilities for process injection indicators using common patterns.
- Implemented identification measures for suspicious Cmdl32 usage, often leveraged as a LOLBin for stealthy command execution.
- Developed a rule to detect the use of the SOAPHound tool, which is associated with Active Directory enumeration.
- Updated CobaltStrike beacon detection to include specific DLL characteristics signs.
- Other smaller improvements include rule updates, quality-of-life enhancements, and fixes for false positives.

9.6.1 (30/06/25)

Dynamo Analyzer

- Improved detection of suspicious service paths by including all fields in query results.

9.6.0 (30/06/25)

Dynamo Analyzer

- Added comprehensive detection for Model Context Protocol (MCP) server activities on Windows endpoints, which includes configuration files, environment variables, network activity, processes, and installed programs. These detections aim to identify unauthorized access to AI assistants and potential data exfiltration capabilities.

MITRE ATT&CK Analyzer / Yara

- Implemented detection for various native IIS malware families, including IIS-Raid derivatives, RGDoor, IIStealer, ISN, IISpy, IISerpent, and others. Each detection targets specific characteristics of the respective malware family.
- Updated detection rule for samples protected by the ConfuserEx protector, enhancing identification accuracy.
- Other, more minor improvements include rule updates, quality-of-life enhancements, and fixes for false positives.

9.5.0 (04/07/25)

YARAS

- Added detection for KoiLoader/KoiStealer malware.
- Added detection for various documented hacker tools used by the threat actors in recent operations.
- Quality of life and false positive fixes.
- Many other smaller rules are in place for detecting various Initial Access IOCs.

Dynamo

- Updated the list of various hacker tools commonly found in forensic evidence.

9.2.2 (07/05/25)

YARA

- Added detection for Python-based Anubis backdoor used by FIN7, a financially motivated threat group.
- Added detection for QDoor backdoor.
- Added detection for SectopRAT, aka ArechClient2.
- Added detection for Lazarus APT BeaverTail malware and its infostealer component.
- Added detection for a new variant of Stealcstealer.
- Added detection for EarthKurma APT Dunloader backdoor.

Dynamo

- Added identification of risky Windows Registry settings such as enabled RDP, enabled vulnerable SMB, usage of unencrypted WDigest protocol, and more.

9.0.1 (02/04/25)

YARA

- Updated detection of SystemBC multiplatform proxy malware.
- Updated detection of Play ransomware variant and the tools used by this threat actor.
- other small fixes and improvements.

9.0.0 (02/04/25)

YARA

- Added detection for HUI Loader that has been used since at least 2015 by China-based threat groups, including Cinnamon Tempest and menuPass, to deploy malware on compromised hosts. (S1097)
- Added detection for ShadowPad backdoor. ShadowPad is a well-known and privately sold modular backdoor, known to only be supplied to China-aligned APT groups. (S0596)
- Added detection for SodaMaster that has been used by Chinese threat actors to download and execute payloads since at least 2020. (S0627)
- Added detection for SparowDoor backdoor linked to the Chinese FamousSparrow threat actor.
- Added detection for ABYSSWORKER rootkit driver, deployed to target and silence different EDR vendors.
- Enhanced detection of samples using compromised/revoked digital signatures.
- Added detection for various tools used for credentials stealing, brute force, network discovery, and reverse proxy techniques.
- Many other smaller fixes and improvements.

Dynamo

- Enhanced identification of the latest hacker tool names found in forensic evidence that are commonly used in attacks.

8.7.0 - 8.7.1 (05/03/25)

YARA

- Added detection for Winos command and control framework, targeting users in Taiwan in latest campaign.
- Improved detection of ransomware variants targeting the ESXi platform.
- Added detection for Sosano backdoor, targeting organizations with a distinct interest in aviation and satellite communications, along with critical transportation infrastructure.

8.6.3 (03/03/25)

YARA

- Added detection for Sagerunex backdoor attributed to Lotus Blossom APT. (G0030)

8.6.2 (03/03/25)

YARA

- Added detection for Lazarus APT attributed InvisibleFerret and BeaverTail malware variants. (G0032)
- Enchanced detection of malicious samples targetting crypto related browser extensions.

8.6.1 (03/03/25)

YARA

- Improved detection of LightSpy windows variant, developed by Chinese attributed APT41. (G0096)
- Added detection for macOS variant of malware dubbed Rustdoor, possibly linked with notorious Windows ransomware groups.

8.6.0 (26/02/25)

YARA

- Added detection for Gh0stRAT, a remote access trojan that has been used to hack into some of the most sensitive computer networks on Earth. (S0032)
- Added detection for macOS capable ransomware with exfiltration capabilities, masquerading itself as LockBit. (T1486)
- Enhanced detection of vulnerable drivers used for privilege escalation and defense evasion purposes. (T1068)

Dynamo

- Enhanced identification of forensic evidence where PowerShell executed encoded content. (T1027.010)

8.5.1 (12/02/25)

YARA

- Minor fixes.

8.5.0 (11/02/25)

YARA

- Added detection for custom backdoor attributed to Lazarus APT group dubbed Deceptive Development spreading via fake job offers. (G0032)
- Added detection for hack tools used for dumping Veeam credentials stored in MSSQL databases. (T1555)
- Added detection for ValleyRAT backdoor attributed to Silver Fox cybercrime group.
- Other small fixes and improvements.

Dynamo

- Enhanced identification of hack tools found in forensic evidence.

8.4.0 (26/01/25)

YARA

- Minor fixes.

8.3.0 (26/01/25)

YARA

- Minor fixes.

8.2.4 (30/12/24)

YARA

- Added detection for malicious extensions involved in the Cyberhaven compromise and a broader campaign targeting Chrome extensions for credential-stealing purposes.

8.2.0 (26/12/24)

YARA

- FP fixes and verdict improvements.

8.1.0 (26/12/24)

YARA

- Access logs detection improvement. We now tend to show the entire line instead of matching string of interest only.
- Added detection for BrazenBamboo APT.
- FP fixes

Dynamo

- HTML smuggling improvement and FP fixes.
- Improved description of rules.

8.0.2 (04/12/24)

YARA

- Added detection for GHOSTSPIDER backdoor, attributed to the Chinese Earth Estries APT group, primarily targeting critical industries such as telecommunications and government entities
- Added detection for Pygmy Goat, which was discovered on Sophos XG firewall devices, providing backdoor access to the device.

8.0.1 (28/11/24)

YARA

- Added detection for STEALHOOK, an exfiltration tool used by OilRig (APT34) group. (G0049)
- Added detection for tools designed to exploit CVE-2024-30088, a Windows Kernel elevation of privilege vulnerability. (T1068)

8.0.0 (28/11/24)

YARA

- Enhanced identification of Vulnerable and Malicious drivers that are weaponized by threat actors for defense evasion purposes. (T1068)
- Added detection for EDRSandblast, a hack tool designed to bypass EDR detection. (T1562)
- Enhanced detection of various tools used by threat actors for Credentials Access, Discovery, Lateral Movement, and other TTPs.

7.3.0 (28/11/24)

YARA

- Added detection for Medusa ransomware.
- Added detection for Ymir ransomware.
- Yara rules that scan Access Logs for signs of exploitation attempts are now updated to show the entire line where suspicious activity was detected.

7.3.0 (28/11/24)

YARA

- Added detection for RDP configuration files that include unusual sets of permissions such as access to audio, disks, and the clipboard. (T1219)
- Added detection for various hack tools designed to extract passwords from password stores. (T1555)

7.2.0 (28/10/24)

YARA

- Added detection for BianLian ransomware. (T1486)
- Enhanced identification of credential stealers that collect browser data. (T1005)
- Enhanced detection of Cobalt Strike. (S0154)
- Enhanced detection of memory dumpers and scripts designed to extract and decrypt Kerberos tickets. (T1558)

7.1.0 (18/10/24)

YARA

- Added detection for DragonForce ransomware binaries. (T1486)
- Added detection for Angry IP Scanner. (T1018)

7.0.0 (18/10/24)

YARA

- Added detection for Clop and MedusaLocker ransomware binaries observed in September 2024. (TA0040)
- Enhanced detection of the Defender Control hack tool often used to disable Microsoft Defender. (T1562.001)
- Added detection for HRSword, which threat actors use for defense evasion. (T1562)
- Multiple minor FP fixes and performance improvements.

6.3.1 (08/08/24)

YARA

- Added detection for Bugsleep backdoor attributed to the Iranian MuddyWater threat actor.

6.3.0 (07/08/24)

YARA

- Added detection for Java-based STRRAT and related IOCs.
- Added detection for APT group dubbed StormBamboo/Evasive Panda that compromised an internet service provider (ISP) in order to poison DNS responses for target organizations.

6.2.0 (07/08/24)

YARA

- Andariel/Lazarus IOCs update. (G0138, G0032)
- Added detection for Maui ransomware.

YARA

- Andariel/Lazarus IOCs update. (G0138, G0032)
- Added detection for Maui ransomware.

6.1.1 (07/08/24)

YARA

- Andariel IOCs update. (G0138)
- Improved detection of Metasploit implants for Linux.

6.1.0 (07/08/24)

YARA

- Added detection for IOCs attributed to North Korean Lazarus/Andrariel groups outlined in CISA report. (G0032, G0138)
- Added detection for open-source Lilith RAT. (T1219)

6.0.0 (07/08/24)

YARA

- Improved detection of Shellcode loaders.
- Added detection for the SharpSploit post-exploitation tool.
- Other minor fixes and improvements.

5.7.0 (17/07/24)

YARA

- Added detection for Pirpi backdoor attributed to the Chinese APT3 group. (G0022)
- Added detection for IOCs used in the latest attacks by the APT41 group. (G0096)
- Added detection for URL Shortcuts, taking advantage of CVE-2024-38112 vulnerability.

5.6.2 (17/07/24)

Sigma

- Improved detection of PowerShell processes using base64 obfuscation.

5.6.1 (17/07/24)

Dynamo

- Improved detection of CobaltStrike service installation.

5.6.0 (17/07/24)

YARA

- Added detection for known malicious VSCode extensions.
- Improved detection of successful ProxyShell exploitation found in server logs.
- Various quality of life and FP fixes.

5.5.2 (17/07/24)

YARA

- Improved detection of ASPX compiled DLL webshells.

5.5.1 (17/07/24)

YARA

- Added lsass exclusion for memory scanning. (internal only changelog)

5.5.0 (17/07/24)

YARA

- Added detection for malware known as DISGOMOJI, taking advantage of emojis for C2 communication.
- Added detection for the Durian backdoor attributed to Kimsyky ATP group. (G0094)
- Added detection for BadSpace backdoor.
- Various quality of life and FP fixes.

Dynamo

- Improved detection of Registry Run entries and Scheduled Tasks with base64 encoded PowerShell keyword.

5.4.0 (10/06/24)

YARA

- Added detection for exploitation attempt indicators of a critical argument injection vulnerability in PHP (CVE-2024-4577).
- Added detection for BitRAT backdoor.
- Added detection for OrcusRAT backdoor.
- Added detection for LightSpy malware targeting macOS.
- Improved identification of path traversal indicators in server logs that suggest exploitation attempts.
- Improved detection of .NET obfuscated/protected binaries.

5.3.1 (31/05/24)

YARA

- Updated list of path traversal attacks.

5.3.0 (31/05/24)

YARA

- Added detection for CrimsonRAT. (S0115)
- Improved detection of the IcedID Trojan. (S0483)
- Improved detection of ISO archives with hidden scripts and signs of DLL Side-loading technique. (T1574.002)
- Added detection for the Mythic C2 framework agent.

Dynamo/osquery

- Added identification of possible ARP poisoning/spoofing.

5.2.0 (22/05/24)

YARA

- Added detection for DiceLoader trojan attributed to FIN7. (G0046)
- Added detection for Ebury botnet. (S0377)
- Added detection for Latrodectus trojan. (T1218.011, T1055, T1053.005, T1070.004, T1059.003)
- Added detection for macOS Cuckoo and Atomic stealer. (T1059.002, T1555)
- Enhanced detection for Relective Code Loading technique. (T1620)
- Enhanced detection of Powershell based loaders. (T1059.001)
- Added detection for Kinsing miner. (S0599)
- Improved identification of vulnerable and malicious drivers used for privilege escalation. (T1068)
- Various other fixes and improvements.

Dynamo

- Enhanced detection of network discovery and PowerShell commands in forensic evidence.

5.1.2 (17/05/24)

YARA

- Improved detection of the Metasploit framework.
- Added detection of masqueraded LUA based samples. (T1036.008)
- Added detection for the GooseEgg hack tool, which is used for privilege escalation and credential access, attributed to APT28. (G0007)

5.1.1 (17/05/24)

YARA

- Added detection for Rawdoor, a backdoor attributed to Chinese APT31 group. (G0128)
- Added detection for CR4T backdoor discovered in campaign targeting government entities in the Middle East.
- Improved detection of Pupy opensource, cross-platform C2 and post-exploitation framework constantly being used by various threat actors.
- Improved detection of Linux-based webshells. (T1505.003)
- Improved detection or ZIP archives with indicators of DLL sideloading technique. (T1574.002, T1566)

5.1.0 (18/04/24)

YARA

- Added detection for Kapeka backdoor attributed to Sandworm APT44 group. (G0034)
- Improved/added detection of various malware such as DarkGate, Nitrogen, FatalRAT, WikiLoader.
- Added detection for IOCs related to GlobalProtect CVE-2024-3400.
- Improved detection of Python-based loaders. (T1059.006)
- Improved detection of various shellcode implants e.g. Metasploit-based. (T1620)
- Added detection for IOCs masqueraded as certificates. (T1036.008, T1027)
- Improved detection of obfuscated Javascript-based droppers, suspicious base64 encoded IOCs, and PowerShell-based loaders. (T1620, T1059.001, T1059.007, T1027)
- And many other smaller improvements.

5.0.2 (18/04/24)

Sigma

- Powershell detection update

5.0.1 (11/04/24)

YARA

- Vidar stealer FP fix.

5.0.0 (09/04/24)

YARA

- Added detection for Linux local privilege escalation exploit for CVE-2024-1086.
- Added detection for various APT groups related IOCs, including APT28, APT29, APT33, and APT42. (G1006, G0007, G0016, G0064, G0059)
- Added detection for Dark Crystal a.k.a DCRat.
- Added detection for Sharpire post-exploitation agent. (S0363)
- Enhanced detection of obfuscated Golang-based binaries. (T1027)
- Enhanced detection of Nim-based binaries.
- Enhanced detection of RMM tools and software. (T1219)
- Enhanced identification of misplaced binaries is often used for DLL Side-loading. (T1574.002)
- Enhanced identification of potentially misplaced script-based samples is often used for masquerading purposes. (T1036.005)
- Enhanced identification of samples abusing double extension to trick users into executing malicious files. (T1036.007)

Dynamo

- Enhanced detection of various hacker commands found in areas such as PowerShell commands, console, and console history.
- Vastly improved detection of RMM software commonly abused by malicious actors. (T1219)

4.3.1. (01/04/24)

YARA

- Added detection for backdoored binaries and indicators of compromise found in XZ Utils 5.6.0 and 5.6.1. (CVE-2024-3094)

4.3.0 (22/03/24)

YARA

- Added detection for Xdealer malware attributed to the China-nexus threat actor tracked as Earth Lusca. (G1006)
- Added detection for custom malware dubbed DinodasRAT targeting government organizations. (G1006)
- Added detection for binaries signed by a D2innovation certificate attributed to the Kimsuky APT group. (G0094)
- Added detection for Lumma information stealer (aka LummaC2 Stealer). (T1082, T1622, T1140, T1562, T1119, T1005, T1071, T1020)
- Added detection for Meduza info stealer. (T1614, T1082, T1113, T1552, T1571)
- Added detection for indicators found in compiled ASPX Web Shell DLLs. (T1505.003)
- Enhanced detection of samples having a suspicious keyword in their PDB path (e.g. Trojan, Shellcode). (TA0005)
- Enhanced detection of Remote Access Software Tools commonly used in ransomware attacks. (T1219)
- Enhanced detection of misplaced files masquerading as legitimate Windows binaries. (T1036.005)
- Enhanced detection of malicious samples and scripts obfuscated with XOR, AES and custom encoding. (T1027)
- Enhanced detection of samples abusing double extension in order to hide true file type. (T1036.007)
- Enhanced detection of LNK files executing suspicious PowerShell commands. (T1059.001, T1204.002)
- Enhanced detection of older exploits such as Zerologon, BlueKeep and more. (T1021, T1068)
- Various other rules, fixes and performance improvements.

Dynamo

- Updated Hacker Tool list with new keywords for hunting in forensic artifacts such as Applications, Cronjobs, Downloads, MFT, Prefetch, Processes, Registry, Scheduled Tasks, Services, ShellBags, Shell History, and Shimcache. (T1588.002)
- Updated detections of Remote Management Software Website domains in DNS Cache, indicating potentially unwanted usage of remote access software. (T1219)

4.2.3 (05/03/24)

YARA

- Added detection for indicators of compromise indicating exploitation attempts of two recent vulnerabilities in JetBrains TeamCity Multiple Authentication Bypass Vulnerabilities (CVE-2024-27198 and CVE-2024-27199)
- Added detection for the Linux variant of Bifrost (aka Bifrose). Bifrost is a remote access Trojan (RAT) that allows an attacker to gather sensitive information, like hostname and IP address. (T1219)
- Added detection for Xeno RAT; an intricately designed malware, crafted with advanced functionalities, conveniently accessible at no cost on GitHub. (T1059.003, T1053.005, T1622, T1497, T1055, T1071.00)
- Added detection for suspicious unsigned executables protected with Obsidium protector. (T1027.002)
- Added detection for FudModule rootkit exploiting CVE-2024-21338 kernel elevation of privilege vulnerability. (T1068)
- Enhanced detection of files found outside of their default location which is a very popular way of hiding malicious files under a known name of a legitimate Windows component. (T1036.005)
- Enhanced detection of CobaltStrike beacons. (S0154)

4.2.2 (27/02/24)

YARA

- Added detection for indicators of compromise, indicating exploitation attempts of two recent vulnerabilities in ConnectWise ScreenConnect. (CVE-2024-1709 & CVE-2024-1708)

4.2.1 (24/02/24)

YARA

- Restored %WINDIR%\Temp to depth 2 recursion for now.

4.2.0 (23/02/24)

YARA

- Restored memory scan [INTERNAL USAGE detail]
- Added detection for the latest TinyTurla IOCs (G0010)
- Improved detection of Linux Shell scripts commonly used in malicious attacks. Examples include log removal, public DNS insertion, manipulation of root SSH keys, and other post-exploitation commands.
- Enhanced detection of various hack tools mentioned in the latest malware campaigns.

4.1.0 (20/02/24)

YARA

- Added detection for emails exploiting the Microsoft Outlook CVE-2024-21413 vulnerability.
- Enhanced detection of the Silver red team framework implant. (S0633)
- Added detection for IOCs abusing the Mockbin service for malicious purposes. (T1090.004, T1102)
- Added detection for IOCs designed to capture NTLMv2 hashes. (T1187)
- Enhanced detection of binaries named after legitimate Windows executables for masquerading and defense evasion purposes. (T1036.005)
- Enhanced detection of IOCs with base64 encoded keywords such as Powershell, WScript, and many more. (T1027)
- Enhanced detection of ESXi ransomware variants. (TA0040)
- Enhanced detection for many other IOCs with references to suspicious locations and suspicious commands, such as disabling UAC, enabling RDP, and more. (T1562.001, T1059.001, T1021.001, T1112)
- Various other fixes and performance improvements.

Dynamo

- Added detection for Crypto Mining Pool Address in DNS Cache and Browser History. (T1496)
- Added detection for registry run entries executing PowerShell command to read data stored in Registry. (T1547.001, T1059.001)
- Added detection for registry run entries executing suspicious PowerShell commands. (T1547.001, T1059.001)
- Updated list of Widely Abused Top-Level Domains found in DNS Cache. (T1583.001)
- Updated Hacker Tool list with over 100 new keywords for hunting in forensic artifacts such as Applications, Cronjobs, Downloads, MFT, Prefetch, Processes, Registry, Scheduled Tasks, Services, ShellBags, Shell History, and Shimcache. (T1588.002)
- Updated detections for hunting Large File Transfer Websites in DNS Cache, which can be used for uploading sensitive/confidential data. (T1567.002)

4.0.1 (05/02/24)

YARA

- Added detection for C# and dictionary-based webshells.
- Enhanced detection of JSP webshells.
- Enhanced detection of directory traversal and XSS injection indicators found in server logs.
- Enhanced detection of ProxyShell and ProxyNotShell vulnerabilities.
- Added detection of various Linux exploits.
- An updated list of vulnerable and malicious drivers from LOL Drivers project.
- Added detection for binaries using potentially compromised AnyDesk certificate.
- Other minor fixes.

Dynamo

- Minor FP fixes.

3.5.2 (22/01/24)

YARA

- Added more detection rules for IOCs observed in the exploitation of Ivanti VPN. (CVE-2023-46805 and CVE-2024-21887)
- Added detection for IOCs related to Russian threat group COLDRIVER (also known as UNC4057, Star Blizzard, and Callisto)
  Reference: https://blog.google/threat-analysis-group/google-tag-coldriver-russian-phishing-malware/

3.5.1 (22/01/24)

YARA

- Added detection for IOCs linked to Iranian and Russian APT groups such as BlueBravo and Siamesekitten. (APT29, G1001)
- Added detection for IOCs linked to Iranian OilRig APT. (G0049)
- Improved detection of Lazarus APT-related IOCs. (G0032)
- Added detection of Outlook CVE-2023-23397 vulnerability exploitation.
- Improved identification of remote access software. (T1219)
- Improved detection of PowerShell scripts loading obfuscated content directly into memory. (T1059.001, T1620)
- Added detection for archives exploiting Baracuda ESG vulnerability CVE-2023-2868.
- Added detection for implants related to Alchimist attack framework.
- Added detection of pkexec CVE-2021-4034 vulnerability exploitation.
- Improved detection of various hacktools used for port scanning, brute force, and privilege escalation.
- Improved detection of mixed casing keywords often used as a way of obfuscation. (T1027)
- Improved detection of double file extension masquerading in archives such as ZIP or RAR. (T1036.007)
- Enhanced detection of indicators of various exploitation attempts, including Log4j, SQL Injection, XSS attacks, path traversal attacks, and more. (T1190)
- Added detection for IOCs found in the exploitation of Ivanti Connect Secure VPN. (CVE-2023-46805, CVE-2024-21887)
  and more

Dynamo

- Added detection for scheduled tasks executing Certutil. (T1053.005, S0160)

Other

- Various FP fixes and performance improvements.

Linux Analyzers

- **Generic WebShell Analyzer (wsa)**

  Scans asset for malicious webshells using YARA rules.
- **Vulnerability Analyzer (vua)**

  Identifying if your device compromised with a known vulnerability.
- **YARA Scanner (gys)**

  Scans your asset with your YARA repositories(refer to blog post here).
- **Process Analyzer (lpa)**

  Executes rules for running Processes, Process modules and Process handles.
- **CronJob Analyzer (cra)**

  Identifies suspicious entries in CronJob tasks.
- **Package Manager Analyzer (pkgmngr)**

  Identifies suspicious entries in Package Managers.
- **Shell History Analyzer (sha)**

  Identifies suspicious entries in Shell histories.

macOS Analyzers

Audit Event Analyzer >

Dynamo Analyzer >

- **YARA Scanner (gys)**

  Scans your asset with your YARA repositories(refer to blog post here).
- **Browser History Analyzer (bha)**

  Identifies URLs of interest from the browser histories.
- **CronJob Analyzer (cra)**

  Identifies suspicious entries in CronJob tasks.
- **Downloads Analyzer (dla)**

  Identifies suspicious entries in downloads.
- **Shell History Analyzer (sha)**

  Identifies suspicious entries in Shell histories.

Audit Event Analyzer

The macOS audit system records critical operational and security data. While this provides valuable insights for security monitoring, it can also be exploited by attackers to either gather sensitive information or manipulate logs to conceal unauthorized activities.

To enhance security through a 'defense-in-depth' approach, it is imperative that the files located in **/var/audit** be exclusively owned by the 'root' user and belong to the 'wheel' group, with read-only permissions. No other forms of access should be permitted. Additionally, the use of macOS Access Control Lists (ACLs) is not recommended for securing these files.

What does the Audit Event Analyzer do?

- **Keyword Matching:** XDR Forensics will search for customer-provided keywords within each event record. If a keyword is identified, the finding is set to "Matched."
- **Generic Hacker Tools Detection:** Each event record is analyzed for the presence of known hacker tools. If such a tool is detected, the finding is set according to pre-established criteria.
- **Generic Hacker Commands Analysis:** We evaluate each event record for the presence of commonly used hacker commands. If a command is detected, the finding is determined based on predefined guidelines.
- **Sigma Rules Assessment:** Each event record is scanned against a set of Sigma rules. If a rule is matched, the finding is set by predetermined standards.

This structured approach ensures a comprehensive and efficient analysis of Audit Events in macOS, facilitating timely and accurate findings.

Windows Analyzers

Dynamo Analyzer >

Browser History Analyzer >

Generic WebShell Analyzer >

- **Application Analyzer (aa)**
  Identifies potentially malicious installed applications.
- **Registry Analyzer (ara)**
  Identifies Autoruns registry records of interest.
- **Scheduled Task Analyzer (asta)**
  Scans Scheduled task entries for items of note.
- **Windows Services Analyzer (awsa)**
  Identifies potentially malicious Windows services.
- **DNS Cache Analyzer (dnsa)**
  Scans DNS Cache records to identify abused TLDs.
- **Event Records Analyzer (ela)**
  Analyzes Event Records with the Sigma rules.
- **Hosts File Analyzer (ha)**
  Identifies host file entries with potentially malicious entries
- **$MFT Analyzer (mfta)**
  Identifies MFT records of interest
- **Network Share Analyzer (nsa)**
  Identifies potentially suspicious Network shares

Prefetch Analyzer >

- **Process Analyzer (pa)**

  Scans assets for running Processes, Process modules, and Process handles of interest

- **ShellBags Analyzer (sba)**

  Identifies suspicious entries in ShellBags

- **User Folders Analyzer (ufa)**

  Identifies suspicious entries in User Folders.

- **Events of Interest (wea)**

  Analyzer for tracking events that you are interested in. This list can be customized via config (refer to blog post here) file.

- **Vulnerability Analyzer (vua)**

  Identifying if your device is compromised with a known vulnerability.

- **YARA Scanner (gys)**

  Scans your asset with your YARA repositories.

- **Ransomware Identifier (rwa)**

  Scans the asset for ransomware using YARA rules.

- **AppCompatCache Analyzer**

  Scans AppCompatCache for suspicious entries in the executable files shimmed on the system.

Prefetch Analyzer

The Prefetch Analyzer is tasked with identifying suspicious entries within Windows's Prefetch feature.

Prefetch is a Windows OS feature aimed at enhancing performance by speeding up application and system startups. It works by tracking and preloading frequently accessed data and code based on user application usage, thus reducing load times.

In Windows, each application execution may generate or update a Prefetch file (.pf), which contains information about the application, its associated files, and the sections of the application accessed during start-up.

From a forensic and malware analysis standpoint, the Prefetch folder is highly informative:

1. Execution History: Prefetch files reveal which applications have been run, shedding light on user or malware activities. The existence of a .pf file for a specific executable is indicative of its usage.
2. Timestamps: These files include timestamps of the first and last times applications were run, aiding in event correlation.
3. Execution Frequency: The number of times an application has been run is recorded, potentially indicating abnormal or repetitive patterns suggestive of malware.
4. Associated Files: Prefetch files list files and directories the application accessed at startup, aiding in identifying further malicious elements or artifacts.
5. Evasion and Anti-Forensics: Malware may try to avoid detection by altering or deleting its own or other applications' Prefetch files. Missing files or signs of tampering can signal malicious interference.

Shellbag Data Fields

Shellbags are Windows registry artifacts that track and record user interactions with the file system via Windows Explorer. These entries provide visibility into the history of folder browsing activity and are an essential source of evidence in digital forensics and incident response (DFIR) investigations.

ShellBags are often used to identify folder access patterns, deleted directories, and user behavior—even when certain data has been removed or is no longer accessible via traditional methods. XDR Forensics supports remote collection and presentation of Shellbag data as part of its broader Windows evidence acquisition capabilities.

Key Components of a Shellbag Entry

Each Shellbag entry contains several attributes. Below is a breakdown of these attributes, how they are labeled in XDR Forensics, and what they reveal during forensic analysis:

Field	Description
`key_path`	The registry path where the Shellbag entry is stored. Indicates the user or system scope.
`value`	The raw content of the Shellbag entry within the registry.
`cached_modified`	Cached timestamp when the folder was last modified. Used to infer changes.
`cached_accessed`	Cached timestamp for last access to the folder. Useful for establishing activity windows.
`cached_created`	Cached creation date of the folder, captured by the operating system.
`path`	Full resolved path of the file or folder. Reflects the actual or historical structure.
`slot_modified_time`	Timestamp when the Shellbag slot itself was last modified. Indicates registry update time.
`mft_entry`	Entry number in the NTFS Master File Table (MFT). Helps link artifacts to disk-level data.
`mft_sequence`	Sequence number for the MFT entry. Helps detect file deletion and reuse.
`modified`	Standard NTFS metadata for last modification time of the folder.
`accessed`	Last access time recorded in NTFS metadata.
`created`	Creation timestamp recorded by the file system.

Relevance to DFIR

Shellbag analysis plays a key role in:

- **User activity reconstruction**
- **Detecting folder creation and deletion**
- **Timeline correlation with other artifacts**
- **Understanding attacker movement or staging activity**

By leveraging Shellbag entries, DFIR professionals can fill gaps that might exist in other logging systems or file system data, particularly in post-breach or post-removal scenarios.

FAQ: Is the 'Cached Accessed' field a native Microsoft label?

Not quite — and this is a subtle but important distinction in forensics.

The field `Cached Accessed` is **not an official Microsoft-named field**, but rather a **derived label** commonly used by forensic tools (including XDR Forensics) to describe data extracted from binary structures within Shellbag entries.

What's actually happening:

- Shellbags store **binary shell item data** that may contain timestamps such as:
  - **Created**
  - **Modified**
  - **Accessed**
- These timestamps are embedded in **various shell item structures**, such as `FILE_ENTRY`, `FOLDER_ENTRY`, or `ZIP_CONTENTS` records — and **not explicitly named by Microsoft**.

Forensic tool interpretation:

Tools like XDR Forensics, Shellbags Explorer, and others parse these raw structures and **label the extracted timestamps** with friendly terms like:

- `Cached Created`
- `Cached Modified`
- `Cached Accessed`

The “**cached**” prefix is used because these timestamps represent snapshots, cached at the time the folder was last browsed, rather than live file system metadata.

> **Example**: If a user browses `C:\Users\Bob\Downloads`, the Shellbag entry may include a `FILE_ENTRY` structure with a timestamp that gets interpreted and labeled as "Cached Accessed", even though Microsoft never calls it that.

Summary:

- **Used by XDR Forensics**: Yes
- **Defined by Microsoft**: No (it's a parsed artifact, not a registry value)
- **Still forensically sound**: Yes — it reflects real metadata from the shell item, just given a descriptive label

Windows Event Records and how they are handled

XDR Forensics enables users to fully customize event log collections based on specific channels, event IDs, and collection parameters, making investigations more efficient and precise.

When discussing **Windows Event Logs**, it's important to understand two key concepts: **channels** and **event IDs**.

1. **Channel**: A channel in Windows Event Logs refers to a specific "log" or source of events. Windows organizes event logs into several channels, each dedicated to logging specific types of events. Examples of commonly investigated channels include:
    - **Application**: Logs events from applications running on the system.
    - **Security**: Logs security-related events, such as logon attempts or resource access.
    - **System**: Logs system-level events, including hardware failures and system services.
    - **Setup**: Log setup-related events, typically related to Windows installation and updates.
2. Channels can be essential in the context of XDR Forensics when filtering or triaging specific logs during forensic analysis. By using channels, investigators can isolate relevant events and gain more efficient insights.
3. **Event ID**: An event ID is a unique identifier for a specific type of event within a channel. Windows assigns an event ID to categorize the nature of the event. For example:
    - **Event ID 4624** in the Security channel represents a successful user login.
    - **Event ID 6006** in the System channel indicates a system shutdown.
4. Event IDs are critical for identifying and understanding specific system actions or issues. In forensic investigations, solutions like XDR Forensics use event IDs to pinpoint the exact events relevant to a security incident, helping investigators build timelines and correlate suspicious activity.

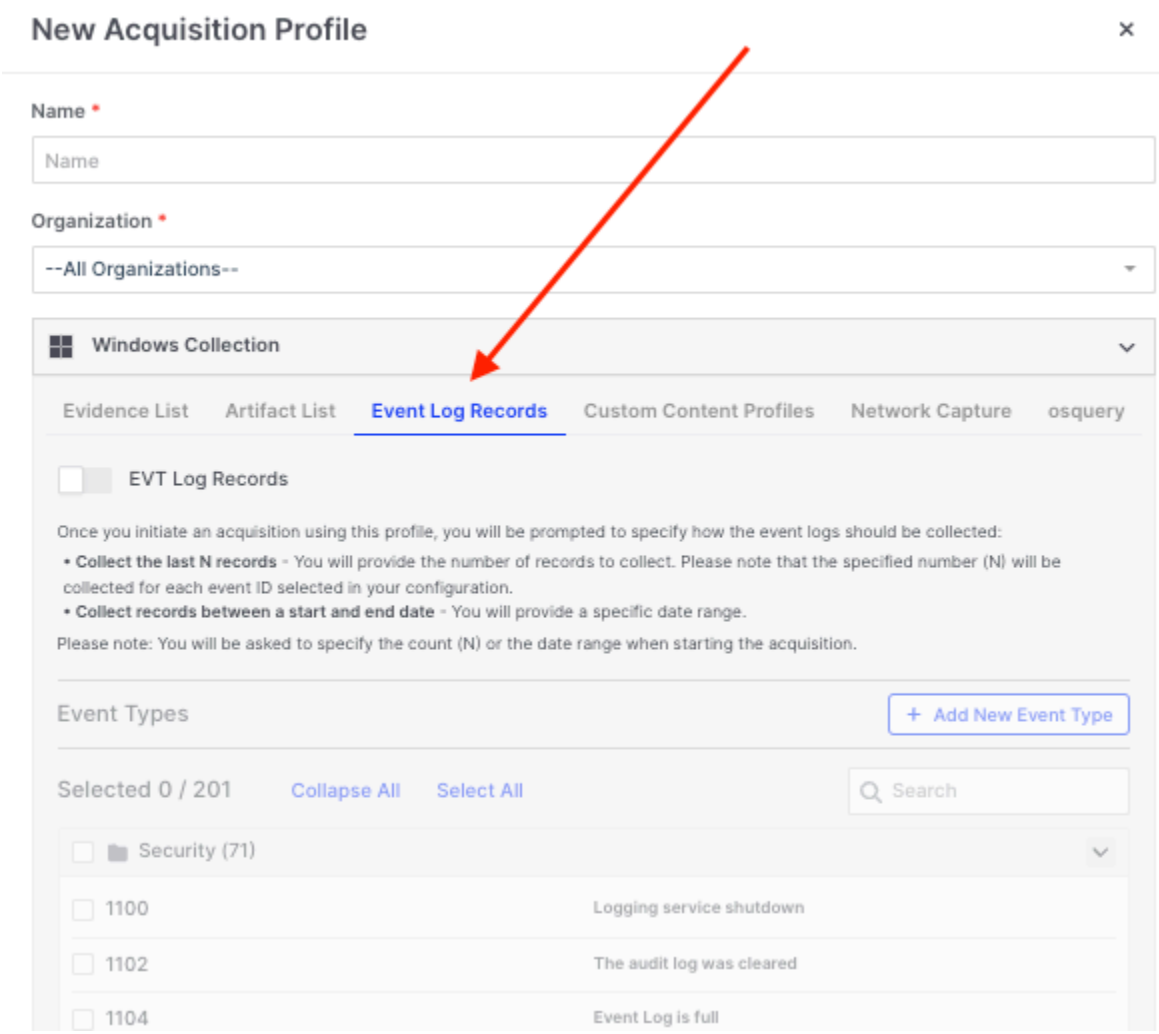
In XDR Forensics's Investigation Hub, filtering by channels and event IDs allows analysts to quickly narrow down logs and focus on the most relevant ones to an investigation.

Key Features introduced in XDR Forensics v4.23:

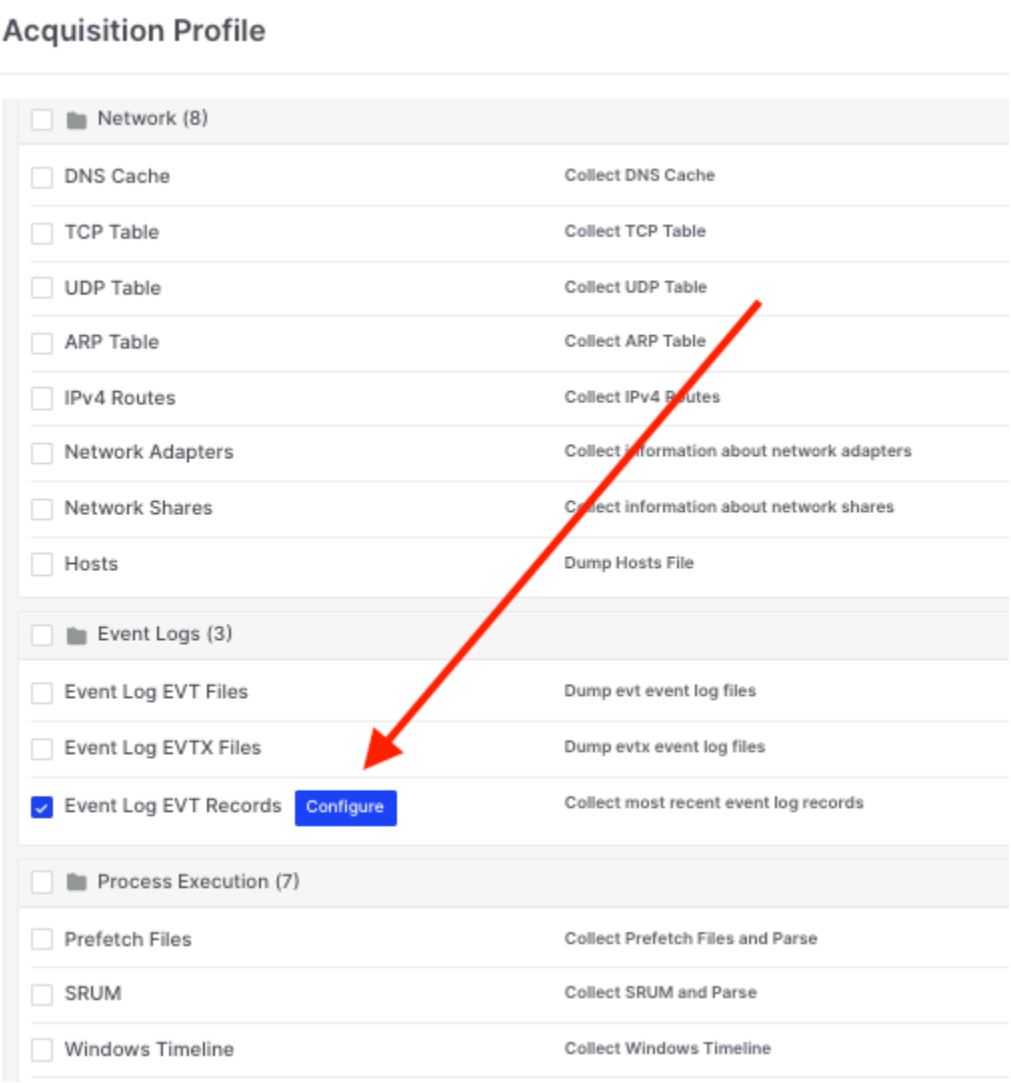
- XDR Forensics will now allow users to collect and present all event logs OR
- Define specific channels for event log collection.
- Users can select event IDs from the XDR Forensics list of over 200 of the most commonly used in DFIR or manually add custom channels and event IDs.
- With event ID selections made, an additional parameter is required:
  - Select the **number of records** to collect **OR**
  - Select **date ranges** for log collection, allowing for targeted log retrieval.

How It Works:

A new **Event Log Records** configuration tab has been added to the “New Acquisition Profile” wizard:

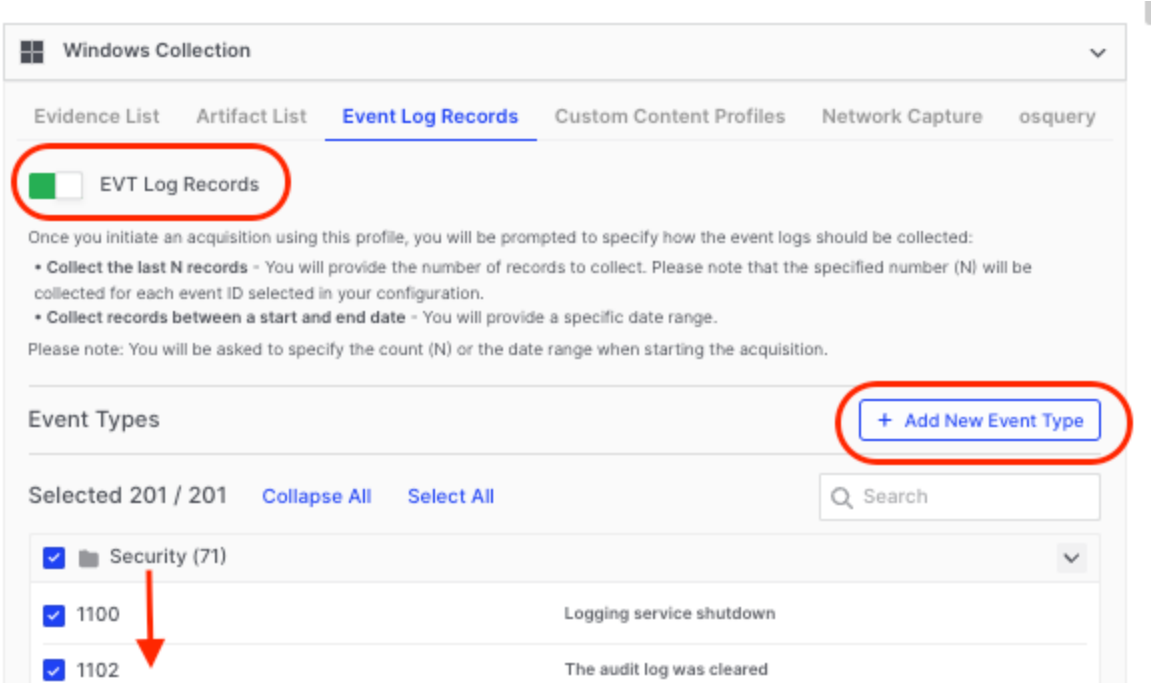


If you are already building an acquisition profile, you will eventually come to the Event Logs section, and you will not notice a configuration button which will take you to the Event Log Records tab as discussed above:

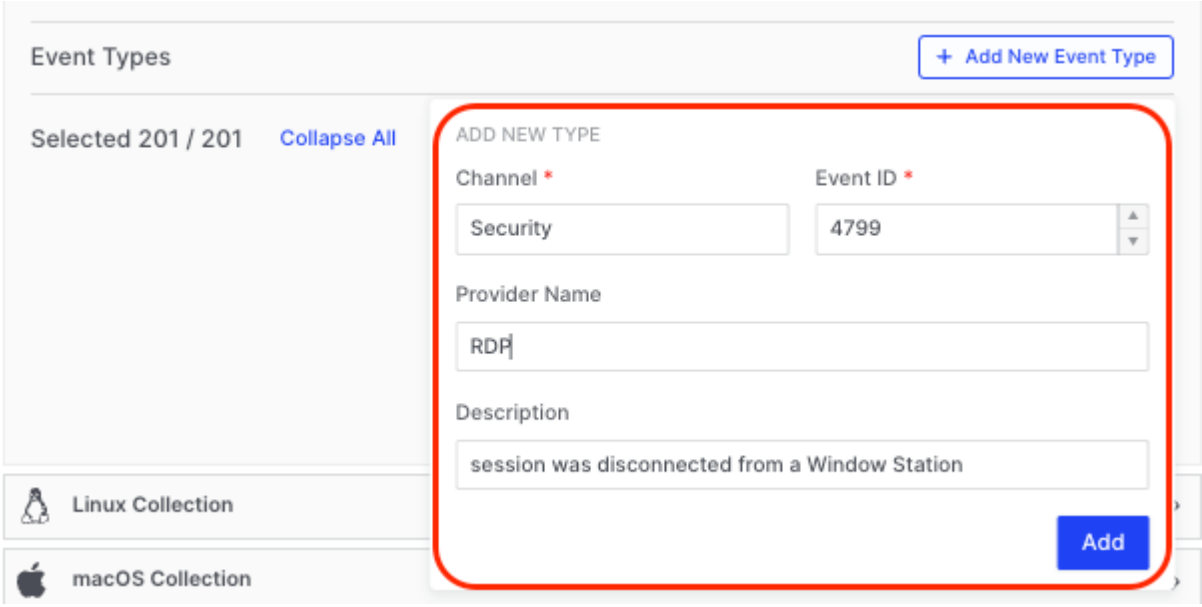


To enable this feature, switch on the **EVT Log Records by** toggling on the switch shown below and then either:

- Choose from a predefined list of 201 event IDs or
- Input custom channel and event IDs using the **Add New Event Type** option.



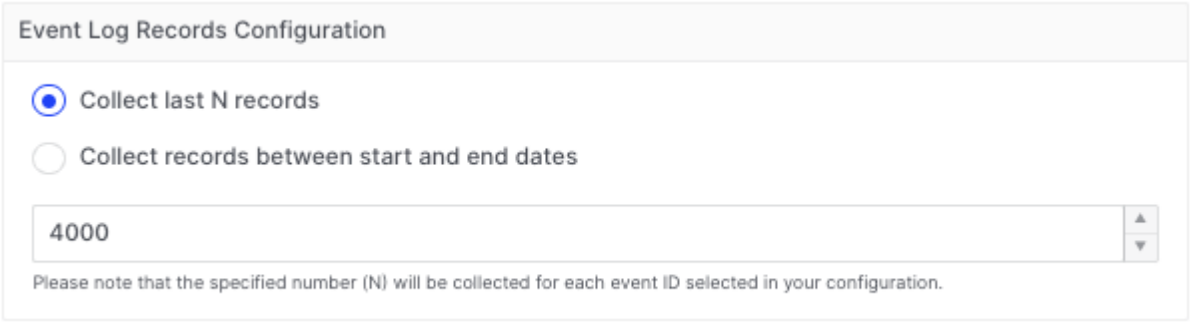
The example below demonstrates how to manually add an event ID that is not included in the 201 provided by XDR Forensics. This is typically only necessary for rare, specialized cases, as the 201 event IDs cover the most common scenarios comprehensively:



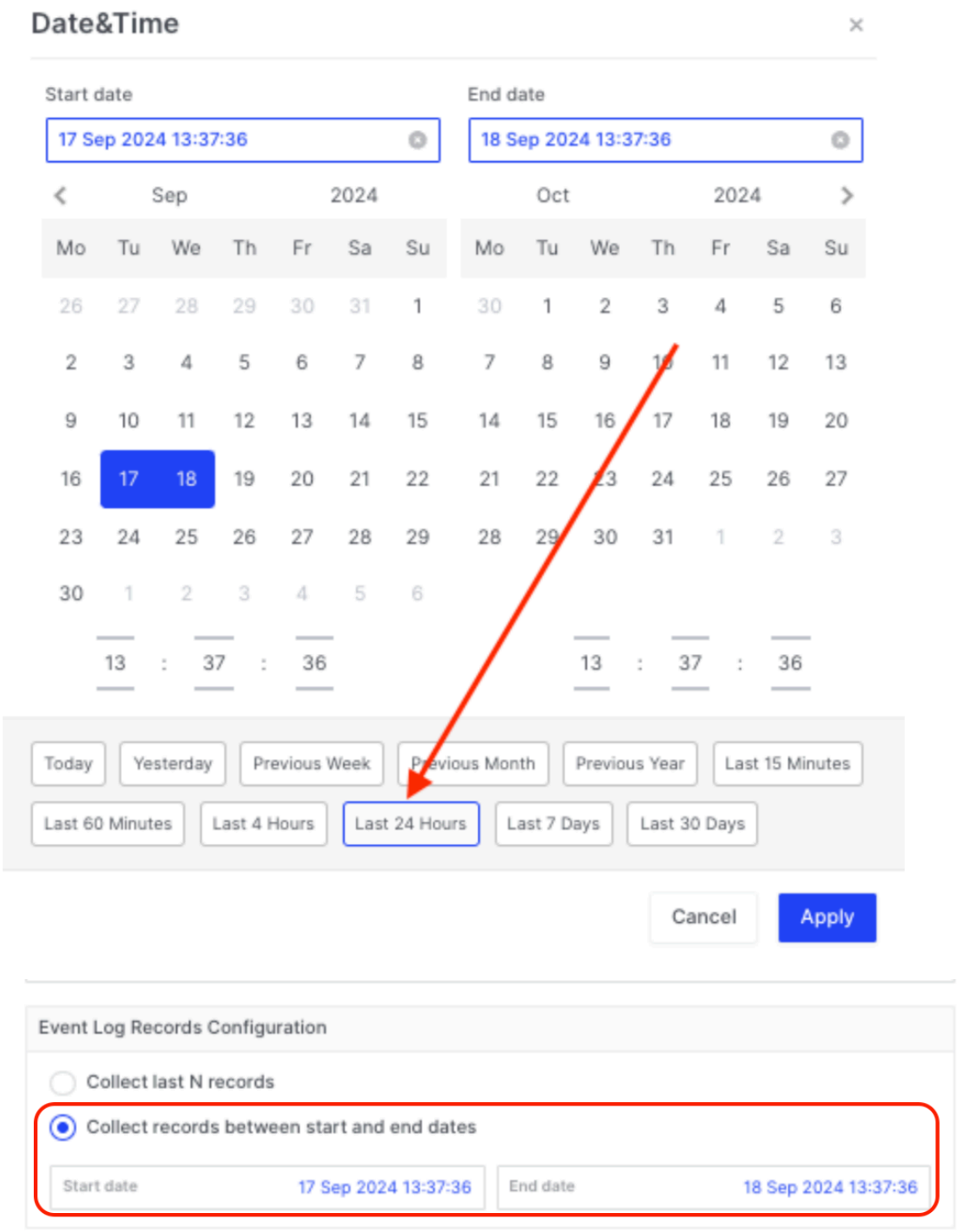
Once configured, the new Acquisition Profile will be saved and can be reused in future investigations by selecting it from the **Acquisition Profiles Library**.

When running an Acquisition Profile that includes event log records, users must select between two additional **Event Log Records Configurations** options:

1. **Collect last 'N' record**s – This gathers the **most recent** 'N' records for each event ID, in the example below, this is set to the default 4000:



2. **Collect records between start and end dates** – In the example below, we have used the date/time picker to choose the last 24 hours option, and you can see those dates/times are automatically populated in the start and end boxes:



The dates and times selected in the 'picker' align with the target asset's system date and time settings, as displayed in the browser.

These features enable highly focused and relevant data collection, drastically reducing the manual effort needed to filter out unnecessary data and ensuring that critical events are captured during investigations.

Here’s a list of what some may consider the top ten event IDs often used to support DFIR Investigations:

1. **Event ID 4624** - Successful Account Logon Tracks successful user logins, essential for identifying legitimate access.
2. **Event ID 4625** - Failed Account Logon Logs failed login attempts, which are useful for detecting brute-force attacks.
3. **Event ID 4776** - Credential Validation Indicates whether a user's credentials were successfully validated by a domain controller.
4. **Event ID 4688** - Process Creation Records every process started on the system, useful for spotting suspicious activity.
5. **Event ID 4648** - Logon Using Explicit Credentials Detects when credentials are used for network logon, helping track lateral movement.
6. **Event ID 4663** - Object Access Logs, when an object (such as a file or folder) is accessed, are crucial for monitoring sensitive data access.
7. **Event ID 4698** - Scheduled Task Creation Captures when a scheduled task is created, often used by attackers for persistence.
8. **Event ID 4719** - Audit Policy Change Tracks changes to audit policies, which may indicate an attempt to cover tracks.
9. **Event ID 1102** - Audit Log Cleared Logs when the security log is cleared, a common indicator of malicious activity.
10. **Event ID 4672** - Special Privilege Assigned Monitors when special privileges (like admin rights) are assigned, helping detect privilege escalation.

Event Records Summary vs. Event Records

When investigating digital incidents with XDR Forensics's Investigation Hub, analysts can rely on **Event Records Summary** and **Event Records** to analyze system events efficiently. While both provide critical insights, they serve distinct purposes, striking a balance between speed and detail in forensic investigations.

Key Differences Between Event Records Summary and Event Records

1. Purpose

- **Event Records Summary:** Provides an aggregated view of event types, enabling investigators to identify trends and anomalies quickly.
- **Event Records:** Stores **every individual event occurrence** with full forensic details, allowing for in-depth analysis.

2. Data Volume

- **Event Records Summary:** Contains **one row per unique event type**, significantly reducing data size for easy interpretation.
- **Event Records:** Stores **one row per event instance**, making it the most detailed source for forensic examination.

3. Level of Detail

- **Event Records Summary:** Displays **basic event information**, including counts and patterns.
- **Event Records:** Captures **detailed event data**, including timestamps, usernames, IP addresses, and other forensic markers.

4. Usage in Investigations

- **Event Records Summary:**
  - Provides a **quick overview** of event trends and frequencies.
  - Helps analysts identify abnormal activity (e.g., an unusual number of failed logins).
- **Event Records:**
  - Supports **in-depth forensic analysis by providing comprehensive** event details.
  - Enables precise querying for specific incidents (e.g., identifying the exact time, user, and IP address of failed login attempts).

Efficient Investigation Workflow with Both Event Types

Using **both** Event Records Summary and Event Records strategically can improve forensic efficiency:

1. **Start with Event Records Summary:** Identify suspicious activity patterns and event frequencies (e.g., a spike in failed logins).
2. **Drill Down into Event Records:** Once a potential issue is spotted, use Event Records to retrieve exact event details, timestamps, user actions, and relevant forensic evidence.

For example, if Event Records Summary shows an unusual **increase in failed login attempts**, analysts can pivot to **Event Records** to examine:

- The **exact timestamps** of each failed attempt.
- The **usernames** are involved.
- The **IP addresses** from which the attempts originated.

By leveraging both views effectively, forensic investigators can **prioritize threats, optimize query performance, and perform comprehensive digital forensic analysis.**

Conclusion

XDR Forensics's **Event Records Summary** and **Event Records** complement each other in forensic investigations. The summary view enables **fast pattern recognition**, while full event records provide **detailed forensic insights**. Understanding when to use each ensures a **more efficient, structured, and thorough investigation process.**

What is an Analysis Pipeline?

Brief overview of DRONES's Analysis Pipeline

Traditional security tools often rely heavily on signature-based detection—a method that struggles to keep pace with modern, fast-evolving threats. XDR Forensics addresses this challenge by incorporating **DRONE**, an advanced automated analyzer designed to rapidly evaluate collected evidence with forensic precision.

The Analysis Pipeline Approach

At the heart of DRONE is the **Analysis Pipeline**, a modular evaluation framework that scrutinizes each evidence item across multiple stages. Each stage, or pipeline, targets a specific category or characteristic, such as suspicious processes, unusual network behavior, file anomalies, or signs of persistence.

As the evidence progresses through these pipelines, DRONE applies a combination of proprietary analyzers, YARA rules, Sigma rules, and osquery-based logic to assess the forensic significance of each artifact. When an issue is detected, it is logged as a **finding**, with one of four possible **severity classifications**:

- **High** – Confirmed malicious behavior or artifacts indicating critical compromise.
- **Medium** – Indicators of suspicious or potentially unwanted behavior.
- **Low** – Anomalies or uncommon patterns that may warrant further investigation.
- **Matched** – Items flagged through either:
  - **Keyword hits**, where the evidence matches one or more pre-defined text, wildcard, or regular expression patterns, or
  - **Triage rule matches, where custom YARA, Sigma, or osquery rules are applied at the time of acquisition or through manual triage, identify relevant indicators.**

This **Matched** category is particularly powerful, as it allows analysts to surface evidence linked to threat hunting hypotheses, indicators of compromise (IOCs), or tactical queries — even when the finding does not yet have a known severity level. It ensures investigators never miss contextually important clues, even if those clues are not immediately classifiable as high or medium severity.

Findings in the Investigation Hub

All findings, including "Matched" hits, are made available in the **Investigation Hub** — XDR Forensics's unified workspace that consolidates triage results, acquisition data, and analysis verdicts from multiple assets and cases. The integration of severity-scored and keyword/triage-matched results enables faster triage, prioritization, and response, especially across large-scale or multi-asset investigations.

This methodical and automated analysis pipeline ensures **forensically sound, scalable, and efficient** evidence evaluation, helping analysts quickly home in on what matters most

What is DRONE?

Lightning Fast, 24/7 Automated Compromise Assessment Technology

DRONE is XDR Forensics's automated compromise assessment module, which delivers a truly effective **decision support system** to facilitate the rapid investigation and assessment of multiple or individual assets. With DRONE, your forensic digital evidence is automatically analyzed by DRONE's out-of-the-box, always-up-to-date **analyzers**.

DRONE significantly reduces investigation time by providing automated findings to analysts. In this way, analysts are not wasting time deliberating over what is different, strange, or unexpected in a case, as they are automatically supplied with any such findings.

DRONE will guide analysts, helping them to ‘pinpoint’ anomalies and potential evidence of an attack in the shortest possible time by tagging each finding as a 'type'—**High, Medium, Low, or Matched.**

Read more about DRONE, XDR Forensics’s built-in automated compromise assessment technology here: **Automated Compromise Assessment with DRONE** ↗

Event Subscription

Overview

The Event Subscription feature enables users to register a URL that the system will call with event details. This allows users to integrate with external systems by sending JSON payloads to the specified URL whenever subscribed events occur.

Key Features

1. **URL Registration**: Users can provide a URL that will be called with a `POST` request containing event information in a JSON payload.
2. **Secure Headers**: Every request to the registered URL includes an `Authorization: Bearer <token>` header for authentication.
3. **Event Selection**: Users can select one or multiple events they want the system to notify their URL about.
4. **Subscription Status**: Users can activate or deactivate subscriptions at any time.
5. **Audit Logs**: The system logs errors or issues when the registered URL cannot be accessed, providing insights into potential failures.

Example HTTP request

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json } "eventName": "TaskProcessingCompletedEvent", "organizationId": 1, "data" : { "taskId": "12345", "taskDetailsUrl": "/#/task-details/12345", "taskType": "Export", "assetId": "67890" } }
```

Audit Logs

The system maintains logs for monitoring calls:

- Failed attempts to access the URL (e.g., `404 Not Found`, `500 Internal Server Error`) are recorded.
- Logs are accessible in the "Audit Logs" section, providing timestamps and error details for troubleshooting.

Event List

DeploymentTokenRegeneratedEvent

This event is triggered when a deployment token is regenerated for an organization. The old token will be invalidated, and a new token will be used for deployments.

Parameters

- **organizationId**: (string) The ID of the organization.
- **organizationName**: (string) The name of the organization.
- **oldToken**: (string) The old deployment token.
- **newToken**: (string) The new deployment token.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "DeploymentTokenRegeneratedEvent", "organizationId": 1, "data": { "organizationId": "1", "organizationName": "example", "oldToken": "oldToken", "newToken": "newToken" } }
```

Supported Events:

DeploymentTokenRegeneratedEvent
CaseFileSavedEvent
TaskProcessingCompletedEvent
TaskProcessingFailedEvent
AirVersionAvailableEvent
AssetCreatedEvent
CaseArchivedEvent
CaseClosedEvent
CaseCreatedEvent
EndpointDeletedEvent
EndpointOfflineEvent
EndpointOnlineEvent
EndpointIsolationStatusUpdatedEvent
EndpointRegisteredEvent
EndpointUninstalledEvent
TasksHaveBeenTakenByEndpointEvent
DroneFileSavedEvent
OrganizationCreatedEvent
OrganizationDeletedEvent
OrganizationUpdatedEvent
RelayServerRegisteredEvent
RelayServerRemovedEvent
TaskAssignedToEndpointEvent
TaskAssignmentCancelledEvent
TaskAssignmentDeletedEvent
TaskCancelledEvent
TaskCompletedEvent
TaskFailedEvent
TaskDeletedEvent
TaskScheduledForEndpointEvent
TriageRuleMatchedEvent
TriageTaskCompletedEvent
AcquisitionTaskCompletedEvent
InterACTShellStartedEvent

CaseFileSavedEvent

This event is triggered when a case file is saved to the AIR by Off-Network or Assets.

Parameters

- **assetId**: (string) The ID of the asset
- **taskId**: (string) The ID of the task.
- **reportURL**: (string) Report URL.

HTTP request example

```
POST <your-path> HTTP/1.1
Host: <your-host>
Authorization: Bearer <your-token>
Content-Type: application/json

}
  "eventName": "TaskProcessingCompletedEvent",
  "organizationId": 1,
  "data" : {
    "taskId": "12345",
    "taskDetailsUrl": "/#/task-details/12345",
    "taskType": "Export",
    "assetId": "67890"
  }
}
```

TaskProcessingCompletedEvent

This event is triggered when a Task's processes are completed. These processes may include actions such as data collection, analysis, reporting, or other workflow steps associated with the task. Completion indicates that all required steps have been successfully executed.

Parameters

- **taskId**: (string) The ID of the task.
- **taskDetailsUrl:** (string) The URL of the task details
- **taskType**: (string) The task type of the task (Acquire evidence, Triage vs.)
- **assetId**: (string) The assetId which is assigned to the task.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "TaskProcessingCompletedEvent", "organizationId": 1, "data": { "taskId": "taskId", "taskDetailsUrl": "url", "taskType": "taskType", "assetId": "assetId", } }
```

TaskProcessingFailedEvent

This event is triggered when a Task's processes fail. This could occur due to various reasons such as resource unavailability, misconfiguration, or unexpected errors during execution. Users should review the task details and the associated error reason to identify the root cause and take corrective actions, such as retrying the task or fixing the underlying issues.

Parameters

- **taskId**: (string) The ID of the task.
- **taskDetailsUrl:** (string) The URL of the task details
- **taskType**: (string) The task type (Acquire evidence, Triage vs.)
- **assetId**: (string) The assetId which is assigned to the task.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "TaskProcessingFailedEvent", "organizationId": 1, "data": { "taskId": "taskId", "taskDetailsUrl": "url", "taskType": "taskType", "assetId": "assetId", } }
```

AirVersionAvailableEvent

This event is triggered when a new version of the AIR product is available. It informs the system or users about the newly available version and compares it with the current version in use.

Parameters

- **newVersion**: (string) The new version of the AIR that is now available.
- **currentVersion**: (string) The current version of the AIR in use.
- **releaseNotes:** (string) The notes of the release

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "AirVersionAvailableEvent", "organizationId": 1, "data": { "newVersion": "2.0.0", "currentVersion": "1.0.0", "releaseNotes": "Hi new changes applied" } }
```

AssetCreatedEvent

This event is triggered when a new asset is created in the system. It provides details about the asset, including its ID, name, type, and creation timestamp.

Parameters

- **id**: (string) The unique identifier of the asset.
- **name**: (string) The name of the asset.
- **type**: (string) The type of the asset
- **createdAt**: (string) The date when the asset was created.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "AssetCreatedEvent", "organizationId": 1, "data": { "id": "asset-id", "name": "Asset Name", "type": "Asset Type", "createdAt": "2025-01-28T12:00:00Z" } }
```

CaseArchivedEvent

This event is triggered when a case is archived in the system. It provides details about the archived case, including its ID and name.

Parameters

- **id**: (string) The unique identifier of the archived case.
- **name**: (string) The name of the archived case.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "CaseArchivedEvent", "organizationId": 1, "data": { "id": "case-id", "name": "Case Name" } }
```

CaseClosedEvent

This event is triggered when a case is closed in the system. It provides details about the closed case, including its ID, name, and the timestamp when it was closed.

Parameters

- **id**: (string) The unique identifier of the closed case.
- **name**: (string) The name of the closed case.
- **closedOn**: (string) The date when the case was closed.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "CaseClosedEvent", "organizationId": 1, "data": { "id": "case-id", "name": "Case Name", "closedOn": "2025-01-28T12:00:00Z" } }
```

CaseCreatedEvent

This event is triggered when a new case is created in the system. It provides details about the created case, including its ID, name, and the owner user of the case.

Parameters

- **id**: (string) The unique identifier of the created case.
- **name**: (string) The name of the created case.
- **ownerUser**: (string) The user who owns the created case.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "CaseCreatedEvent", "organizationId": 1, "data": { "id": "case-id", "name": "Case Name", "ownerUser": "user-name" } }
```

EndpointDeletedEvent

This event is triggered when an endpoint is deleted from the system. It provides details about the deleted endpoint, including its ID, name, label, version, and platform.

Parameters

- **id**: (string) The unique identifier of the deleted endpoint.
- **name**: (string) The name of the deleted endpoint.
- **label**: (string) The label associated with the deleted endpoint.
- **version**: (string) The version of the deleted endpoint.
- **platform**: (string) The platform of the deleted endpoint.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "EndpointDeletedEvent", "organizationId": 1, "data": { "id": "endpoint-id", "name": "Endpoint Name", "label": "Endpoint Label", "version": "1.0.0", "platform": "Platform Name" } }
```

EndpointOfflineEvent

Triggered when an endpoint goes offline.

Parameters

- **id**: (string) The unique identifier of the deleted endpoint.
- **name**: (string) The name of the deleted endpoint.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "EndpointOfflineEvent", "organizationId": 1, "data": { "id": "endpoint-id", "name": "Endpoint Name" } }
```

EndpointOnlineEvent

Triggered when an endpoint comes online.

Parameters

- **id**: (string) The unique identifier of the deleted endpoint.
- **name**: (string) The name of the deleted endpoint.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "EndpointOnlineEvent", "organizationId": 1, "data": { "id": "endpoint-id", "name": "Endpoint Name" } }
```

EndpointIsolationStatusUpdatedEvent

This event is triggered when the isolation status of an endpoint is updated. It provides details about the endpoint and the changes in its isolation status.

Parameters

- **endpointId**: (string) The unique identifier of the endpoint whose isolation status has been updated.
- **oldStatus**: (string) The previous isolation status of the endpoint.
- **newStatus**: (string) The new isolation status of the endpoint.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "EndpointIsolationStatusUpdatedEvent", "organizationId": 1, "data": { "endpointId": "endpoint-id", "oldStatus": "isolation-enabled", "newStatus": "isolation-disabled" } }
```

EndpointRegisteredEvent

This event is triggered when a new endpoint is registered in the system. It provides details about the registered endpoint, including its ID, organization ID, name, platform, and version.

Parameters

- **id**: (string) The unique identifier of the registered endpoint.
- **organizationId**: (string) The ID of the organization that the endpoint belongs to.
- **name**: (string) The name of the registered endpoint.
- **platform**: (string) The platform associated with the registered endpoint.
- **version**: (string) The version of the registered endpoint.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "EndpointRegisteredEvent", "organizationId": 1, "data": { "id": "endpoint-id", "organizationId": "org-id", "name": "Endpoint Name", "platform": "Platform Name", "version": "1.0.0" } }
```

EndpointUninstalledEvent

This event is triggered when an endpoint is uninstalled from the system. It provides details about the uninstalled endpoint, including its ID, name, platform, version, and the source of the uninstallation.

Parameters

- **id**: (string) The unique identifier of the uninstalled endpoint.
- **name**: (string) The name of the uninstalled endpoint.
- **platform**: (string) The platform associated with the uninstalled endpoint.
- **version**: (string) The version of the uninstalled endpoint.
- **uninstallSource**: (string) The source of the uninstallation, such as whether it was uninstalled via task, user, or due to migration

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "EndpointUninstalledEvent", "organizationId": 1, "data": { "id": "endpoint-id", "name": "Endpoint Name", "platform": "Platform Name", "version": "1.0.0", "uninstallSource": "user" } }
```

TasksHaveBeenTakenByEndpointEvent

This event is triggered when an endpoint has taken a set of tasks. It provides details about the endpoint and the tasks that have been assigned to it.

Parameters

- **endpointId**: (string) The unique identifier of the endpoint that has taken the tasks.
- **taskCount**: (number) The number of tasks that have been assigned to the endpoint.
- **taskNames**: (array of strings) The names of the tasks that have been assigned to the endpoint.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "TasksHaveBeenTakenByEndpointEvent", "organizationId": 1, "data": { "endpointId": "endpoint-id", "taskCount": 3, "taskNames": ["Task 1", "Task 2", "Task 3"] } }
```

DroneFileSavedEvent

This event is triggered when a file related to a drone task is saved in the system. It provides details about the saved file, including the file path, the associated task ID, and the endpoint ID.

Parameters

- **droneZipPath**: (string) The path to the saved drone file (ZIP file).
- **taskId**: (string) The ID of the task associated with the saved drone file.
- **endpointId**: (string) The ID of the endpoint associated with the saved drone file.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "DroneFileSavedEvent", "organizationId": 1, "data": { "droneZipPath": "/path/to/drone/file.zip", "taskId": "task-id", "endpointId": "endpoint-id" } }
```

OrganizationCreatedEvent

This event is triggered when a new organization is created in the system. It provides details about the newly created organization, including its ID and name.

Parameters

- **id**: (string) The unique identifier of the created organization.
- **name**: (string) The name of the created organization.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "OrganizationCreatedEvent", "organizationId": 1, "data": { "id": "organization-id", "name": "Organization Name" } }
```

OrganizationDeletedEvent

This event is triggered when an organization is deleted from the system. It provides details about the deleted organization, including its ID and name.

Parameters

- **id**: (string) The unique identifier of the deleted organization.
- **name**: (string) The name of the deleted organization.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "OrganizationDeletedEvent", "organizationId": 1, "data": { "id": "organization-id", "name": "Organization Name" } }
```

OrganizationUpdatedEvent

This event is triggered when an organization is updated in the system. It provides details about the updated organization, including its name and the fields that were updated.

Parameters

- **name**: (string) The name of the updated organization.
- **updatedFields**: (string) A comma-separated list of the fields that were updated in the organization.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "OrganizationUpdatedEvent", "organizationId": 1, "data": { "name": "Updated Organization Name", "updatedFields": "field1, field2, field3" } }
```

RelayServerRegisteredEvent

This event is triggered when a new relay server is registered in the system. It provides details about the registered relay server, including its ID, name, endpoint ID, version, and endpoint name.

Parameters

- **id**: (string) The unique identifier of the registered relay server.
- **name**: (string) The name of the registered relay server.
- **endpointId**: (string) The ID of the endpoint associated with the relay server.
- **version**: (string) The version of the relay server.
- **endpointName**: (string) The name of the endpoint associated with the relay server.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "RelayServerRegisteredEvent", "organizationId": 1, "data": { "id": "relay-server-id", "name": "Relay Server Name", "endpointId": "endpoint-id", "version": "1.0.0", "endpointName": "Endpoint Name" } }
```

RelayServerRemovedEvent

This event is triggered when a relay server is removed from the system. It provides details about the removed relay server, including its ID, name, and the associated endpoint ID.

Parameters

- **id**: (string) The unique identifier of the removed relay server.
- **name**: (string) The name of the removed relay server.
- **endpointId**: (string) The ID of the endpoint associated with the removed relay server.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "RelayServerRemovedEvent", "organizationId": 1, "data": { "id": "relay-server-id", "name": "Relay Server Name", "endpointId": "endpoint-id" } }
```

TaskAssignedToEndpointEvent

This event is triggered when a task is assigned to an endpoint. It provides details about the task assignment, including the task name, endpoint name, case name, and the associated IDs.

Parameters

- **taskName**: (string) The name of the assigned task.
- **endpointName**: (string) The name of the endpoint the task is assigned to.
- **endpointId**: (string) The unique identifier of the endpoint.
- **caseName**: (string) The name of the case associated with the task.
- **assignmentId**: (string) The unique identifier of the task assignment.
- **caseId**: (string) The unique identifier of the case.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "TaskAssignedToEndpointEvent", "organizationId": 1, "data": { "taskName": "Task Name", "endpointName": "Endpoint Name", "endpointId": "endpoint-id", "caseName": "Case Name", "assignmentId": "assignment-id", "caseId": "case-id" } }
```

TaskAssignmentCancelledEvent

This event is triggered when a task assignment to an endpoint is cancelled. It provides details about the cancelled task assignment, including the task's ID, name, type, and the endpoint it was assigned to.

Parameters

- **taskId**: (string) The unique identifier of the cancelled task.
- **taskName**: (string) The name of the cancelled task.
- **taskType**: (string) The type of the cancelled task.
- **endpointName**: (string) The name of the endpoint the task was assigned to.
- **endpointId**: (string) The unique identifier of the endpoint.
- **taskAssignmentId**: (string) The unique identifier of the cancelled task assignment.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "TaskAssignmentCancelledEvent", "organizationId": 1, "data": { "taskId": "task-id", "taskName": "Task Name", "taskType": "Task Type", "endpointName": "Endpoint Name", "endpointId": "endpoint-id", "taskAssignmentId": "task-assignment-id" } }
```

TaskAssignmentDeletedEvent

This event is triggered when a task assignment is deleted. It provides details about the deleted task assignment, including the task's name, type, and the endpoint to which it was assigned.

Parameters

- **taskName**: (string) The name of the deleted task.
- **taskType**: (string) The type of the deleted task.
- **endpointName**: (string) The name of the endpoint the task was assigned to.
- **endpointId**: (string) The unique identifier of the endpoint.
- **taskId**: (string) The unique identifier of the deleted task.
- **taskAssignmentId**: (string) The unique identifier of the deleted task assignment.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "TaskAssignmentDeletedEvent", "organizationId": 1, "data": { "taskName": "Task Name", "taskType": "Task Type", "endpointName": "Endpoint Name", "endpointId": "endpoint-id", "taskId": "task-id", "taskAssignmentId": "task-assignment-id" } }
```

TaskCancelledEvent

This event is triggered when a task is cancelled. It provides details about the cancelled task, including its ID, name, and type.

Parameters

- **taskId**: (string) The unique identifier of the cancelled task.
- **taskName**: (string) The name of the cancelled task.
- **taskType**: (string) The type of the cancelled task.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "TaskCancelledEvent", "organizationId": 1, "data": { "taskId": "task-id", "taskName": "Task Name", "taskType": "Task Type" } }
```

TaskCompletedEvent

This event is triggered when a task is completed. It provides details about the completed task, including its ID, name, type, the organization it belongs to, and statistics about the assigned and completed endpoints.

Parameters

- **id**: (string) The unique identifier of the completed task.
- **name**: (string) The name of the completed task.
- **type**: (string) The type of the completed task.
- **organizationId**: (string) The ID of the organization to which the task belongs.
- **totalAssignedEndpoints**: (number) The total number of endpoints assigned to the task.
- **totalCompletedEndpoints**: (number) The total number of endpoints that have completed the task.
- **caseIds:** (array of strings) Ids of the case's
- **createdBy**: (string) The name of the owner

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "TaskCompletedEvent", "organizationId": 1, "data": { "id": "task-id", "name": "Task Name", "type": "Task Type", "organizationId": "org-id", "totalAssignedEndpoints": 5, "totalCompletedEndpoints": 4, "caseIds": [C-2025-03] "createdBy":"John doe" } }
```

TaskFailedEvent

Triggered when a task fails (distinct from `TaskProcessingFailedEvent`, which refers to execution process failures).

Parameters:

- **taskId**: (string) The ID of the task.
- **taskName**: (string) The name of the task.
- **taskType**: (string) The type of the task.
- **caseIds**: (array of strings) The IDs of the cases associated with the task.
- **createdBy**: (string) The user who created the task.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "TaskFailedEvent", "organizationId": 1, "data": { "taskId": "task-id", "taskName": "Task Name", "taskType": "Task Type", "caseIds": ["case-id-1", "case-id-2"], "createdBy": "John Doe" } }
```

TaskDeletedEvent

This event is triggered when a task is deleted. It provides details about the deleted task, including its ID, name, and type.

Parameters

- **taskId**: (string) The unique identifier of the deleted task.
- **taskName**: (string) The name of the deleted task.
- **taskType**: (string) The type of the deleted task.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "TaskDeletedEvent", "organizationId": 1, "data": { "taskId": "task-id", "taskName": "Task Name", "taskType": "Task Type" } }
```

TaskScheduledForEndpointEvent

This event is triggered when a task is scheduled for an endpoint. It provides details about the task and the endpoint it is scheduled for, along with the associated case information.

Parameters

- **endpointName**: (string) The name of the endpoint the task is scheduled for.
- **taskName**: (string) The name of the scheduled task.
- **taskType**: (string) The type of the scheduled task.
- **endpointId**: (string) The unique identifier of the endpoint.
- **caseId**: (string) The unique identifier of the associated case.
- **caseName**: (string) The name of the associated case.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "TaskScheduledForEndpointEvent", "organizationId": 1, "data": { "endpointName": "Endpoint Name", "taskName": "Task Name", "taskType": "Task Type", "endpointId": "endpoint-id", "caseId": "case-id", "caseName": "Case Name" } }
```

TriageRuleMatchedEvent

This event is triggered when a triage rule is matched for a task on an endpoint. It provides details about the matched rule, the associated task, and the endpoint involved.

Parameters

- **endpointId**: (string) The unique identifier of the endpoint where the rule was matched.
- **endpointName**: (string) The name of the endpoint where the rule was matched.
- **taskId**: (string) The unique identifier of the associated task.
- **taskName**: (string) The name of the associated task.
- **ruleName**: (string) The name of the triage rule that was matched.
- **ruleType**: (string) The type of the triage rule that was matched.
- **details**: (object) Additional details or context about the matched rule.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "TriageRuleMatchedEvent", "organizationId": 1, "data": { "endpointId": "endpoint-id", "endpointName": "Endpoint Name", "taskId": "task-id", "taskName": "Task Name", "ruleName": "Rule Name", "ruleType": "Rule Type", "details": { "key": "value" } } }
```

TriageTaskCompletedEvent

Triggered when a triage task is completed.

Parameters

- **id**: (string) The task ID.
- **name**: (string) The task name.
- **organizationId**: (string) The organization ID.
- **totalAssignedEndpoints**: (number) Number of endpoints assigned.
- **totalCompletedEndpoints**: (number) Number of endpoints completed.
- **totalMatched**: (number) Number of matched items.
- **totalMatchedEndpoints**: (number) Number of endpoints that had matches.
- **mitreAttackEnabled**: (boolean) Whether MITRE ATT&CK detection was enabled.
- **triageRules**: (array) List of rules used.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "TriageTaskCompletedEvent", "organizationId": 1, "data": { "id": "task-id", "name": "Task Name", "organizationId": "org-id", "totalAssignedEndpoints": 5, "totalCompletedEndpoints": 4, "totalMatched": 3, "totalMatchedEndpoints": 2, "mitreAttackEnabled": true, "triageRules": [ { "id": "triage-id", "name": "Rule Name", "engine": "yara", "searchIn": "system" } ] } }
```

AcquisitionTaskCompletedEvent

Triggered when an acquisition task is completed.

Parameters

- **id**: (string) Task ID.
- **name**: (string) Task name.
- **organizationId**: (string) Organization ID.
- **totalAssignedEndpoints**: (number) Assigned assets.
- **totalCompletedEndpoints**: (number) Completed assets.
- **profileId**: (string) Profile ID.
- **profileName**: (string) Profile name.
- **droneEnabled**: (boolean) Drone used.
- **cpuLimit**: (number) CPU limit for the task.
- **compressionEnabled**: (boolean) Compression enabled.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "AcquisitionTaskCompletedEvent", "organizationId": 1, "data": { "id": "task-id", "name": "Task Name", "organizationId": "org-id", "totalAssignedEndpoints": 5, "totalCompletedEndpoints": 4, "profileId": "profile-id", "profileName": "Profile Name", "droneEnabled": true, "cpuLimit": 100, "compressionEnabled": true } }
```

InterACTShellStartedEvent

Triggered when an InterACT shell session starts.

Parameters

- **taskName**: (string) The name of the task.
- **sessionId**: (string) The session ID.
- **caseId**: (string) The ID of the related case.
- **endpointId**: (string) The ID of the endpoint.
- **endpointName**: (string) The name of the endpoint.

HTTP request example

```
POST <your-path> HTTP/1.1 Host: <your-host> Authorization: Bearer <your-token> Content-Type: application/json { "eventName": "InterACTShellStartedEvent", "organizationId": 1, "data": { "taskName": "Task Name", "sessionId": "session-id", "caseId": "case-id", "endpointId": "endpoint-id", "endpointName": "Endpoint Name" } }\
```

Evidence Repositories

Where to save your collected data

By default, XDR Forensics supports saving collected evidence locally on the asset with paths set as `CiscoForensics` for Windows, `/opt/cisco/forensics` for Linux, and `/opt/cisco/forensics` for macOS. Alternatively, users can opt to send their collections to Evidence Repositories, such as network shares, SMB, FTPS, SFTP, or to cloud storage, including AWS S3 buckets and Azure Blob Storage.

> Defining exact system resource requirements for the Evidence Repository is challenging due to variations in environments, asset counts, and the types of acquisition tasks performed. Disk space, CPU, and memory requirements can vary significantly, influenced by factors such as the size of disk images, log files generated during acquisitions, and the volume of evidence collected from each asset. As a result, **it's impractical to offer a one-size-fits-all recommendation for resource allocation**.

The term Evidence Repository describes a remote location, separate from the actual asset subject to the tasking assignment, whether it is one of the five currently supported storage options.

You can create **Evidence Repositories** in three different ways:

- From the "Evidence Repositories" page
- During Policy creation
- During the Acquisition task creation

A common query from our customers concerns the configuration of the Evidence Repository and its interaction with the XDR Forensics Console, particularly regarding evidence uploads and the required network permissions. Here's what you need to know:

Evidence Upload Process

When configuring the Evidence Repository, it's essential to understand the pathway through which evidence files are uploaded. Specifically, there might be confusion about whether these uploads occur directly from the assets to the Evidence Repository or if they go through the XDR Forensics console.

To clarify: Evidence files are uploaded directly from the assets to the Evidence Repository. This process necessitates configuring your firewall to permit traffic from the asset to the Evidence Repository on the relevant ports. For example, if using SMB for evidence transfer, you must allow access through port 445.

XDR Forensics Console Access

For the File Explorer feature within the XDR Forensics console to function correctly, the XDR Forensics console requires access to the Evidence Repository. This setup ensures that users can seamlessly browse and interact with the stored evidence directly through the XDR Forensics console interface.

Configuring Your Firewall

Given these operational details, it's necessary to adjust your firewall settings accordingly:

- Allow traffic from your assets to the Evidence Repository, particularly if you are using specific protocols, such as SMB on port 445.
- Ensure the XDR Forensics Console has access to the Evidence Repository to enable full functionality of the File Explorer feature.

Creating an evidence repository from “Evidence Repositories”

1. Navigate to the Evidence Repositories section by clicking the Settings button in the Main Menu and then select “Evidence Repositories” from the Secondary Menu.

2. Click the “+Add New” button at the top of the page.

3. From the New Evidence Repository window, provide a name to the repository and then select the relevant repository.

4. Depending on the type of evidence repository you choose, the required fields are adjusted accordingly:

SMB

- Path**:** The location that is polled for evidence. If the IP address of the repository is "172.16.1.1", and the folder name is "Share", the path will be “\\172.16. 1.1\Share” without quotes.
- Username (if required)
- Password (if required)

SFTP

- Host: Hostname or IP address of the SFTP server.
- Port: The port on which the SFTP server is listening to. The default port for SFTP is 22.
- Path: The location directory that is polled for evidence.
- Username (if required)
- Password (if required)

FTPS

- Host: Hostname or IP address of the FTPS server.
- Port: The port on which the FTPS server is listening. The default port for FTPS is 21.
- Path: The location directory that is polled for evidence.
- Username (if required)
- Password (if required)

NB: Implicit SSL/TLS is not supported

Amazon S3

- Region: Region name for the bucket that was created in.
- Bucket: Name of the bucket
- Access Key ID
- Secret Access Key

Note: IAM users must have proper rights and permissions to access the S3 bucket.

Azure Blob

- Shared Access Signature (SAS) URL &#xNAN; See https://docs.microsoft.com/en-us/azure/storage/common/storage-sas-overview ↗ for details.
- To generate a SAS URL, please refer to this link: Generating a SAS URL

Creating an evidence repository during Policy creation

1. Select the Settings button in the Main Menu and then select “Policies” from the Secondary Menu.

Click the “+Add New” button at the top of the page

2. Provide a name to the repository and then select the relevant repository type:

3. Select the relevant repository type by clicking on it.

4. Click the “Save” button.

5. The newly created repository will appear in the drop-down list. Select the relevant repository and finalize the process.

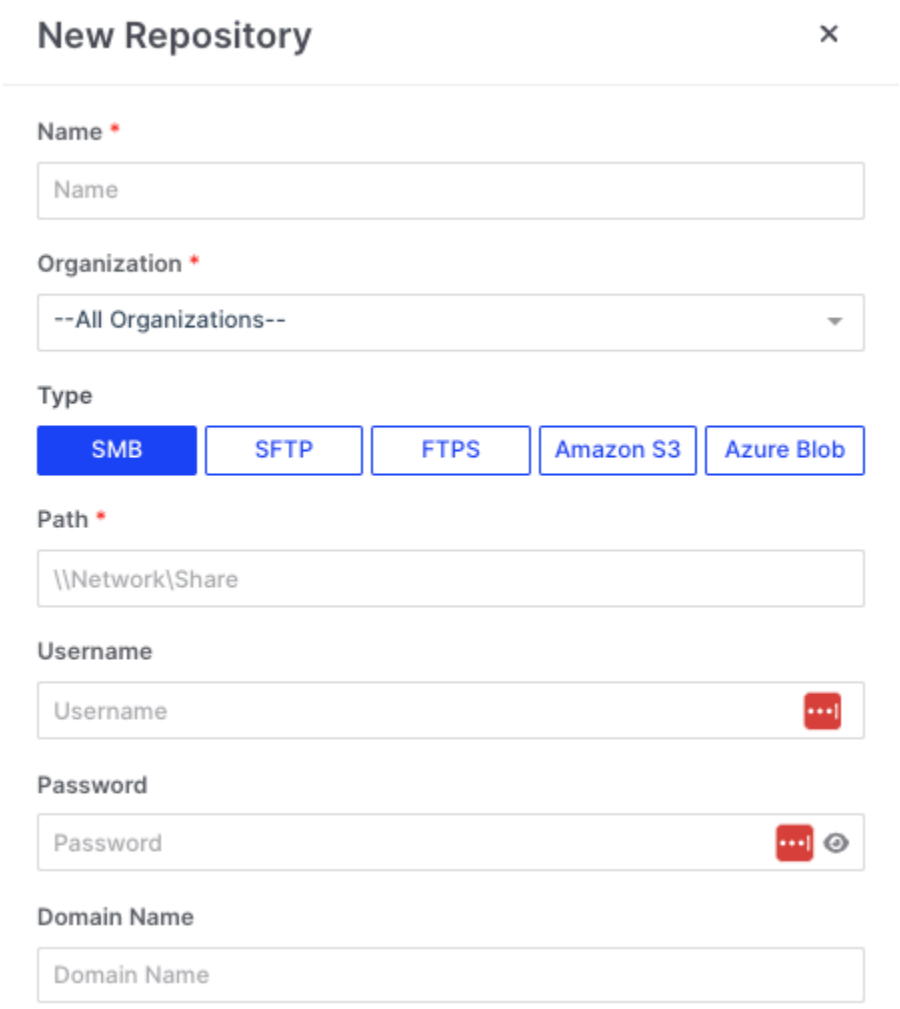
Creating an evidence repository during the acquisition task creation

1. From the “Acquire Evidence” pane, click on the **Evidence Repository** radio button under the "Save Collected Evidence To" section.

2. Click in the "Repository" box and then select “+ Add new repository”:

3. From the window 'New Repository', complete the mandatory fields and select the type of repository you wish to add. There are five options:

- SMB
- SFTP
- FTPS
- Amazon S3
- Azure Blob



Evidence Repositories: New Repository

4. The newly created repository will appear in the drop-down list. Select the repository you want for this particular acquisition and finalize your Acquisition Task via the wizard.

To improve task reliability and prevent failed uploads, a **connection check** for evidence repositories will take place when starting acquisition tasks (both scheduled and immediate). Here's how it works:

- When creating tasks like Acquisition or Acquire Image, XDR Forensics automatically checks the connection to the selected repository (SFTP, FTPS, Azure, or AWS).
- If the connection check takes longer than 10 seconds, it will be canceled, and a warning message will appear. However, task creation is not blocked—you can choose to proceed or cancel the task if the repository is inaccessible.

UI Warning Message: If the repository is inaccessible, you'll see this warning: &#xNAN; *"The following evidence repositories are currently inaccessible. If responders cannot access these repositories, they will not be able to send the collected evidence. Please note that access to these repositories is managed through the XDR Forensics server. If the responders have access, evidence transmission will proceed without issues. Do you still want to continue?"*

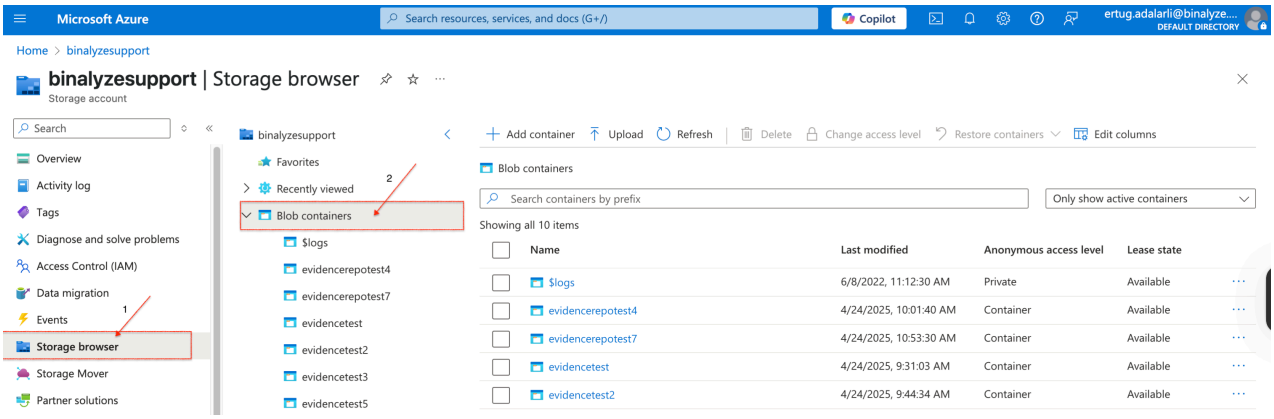
This feature ensures you're aware of potential access issues before initiating a task, helping you avoid wasted time on failed uploads. It’s important to note that this connection check occurs between the XDR Forensics console and the repository, not between the responders and the repository. While it's impractical to check every responder in large-scale tasks, a successful console check significantly reduces the likelihood of connection issues for responders.

Generating a SAS URL

Step-by-Step: Generating a SAS URL

Access Azure Blob Container

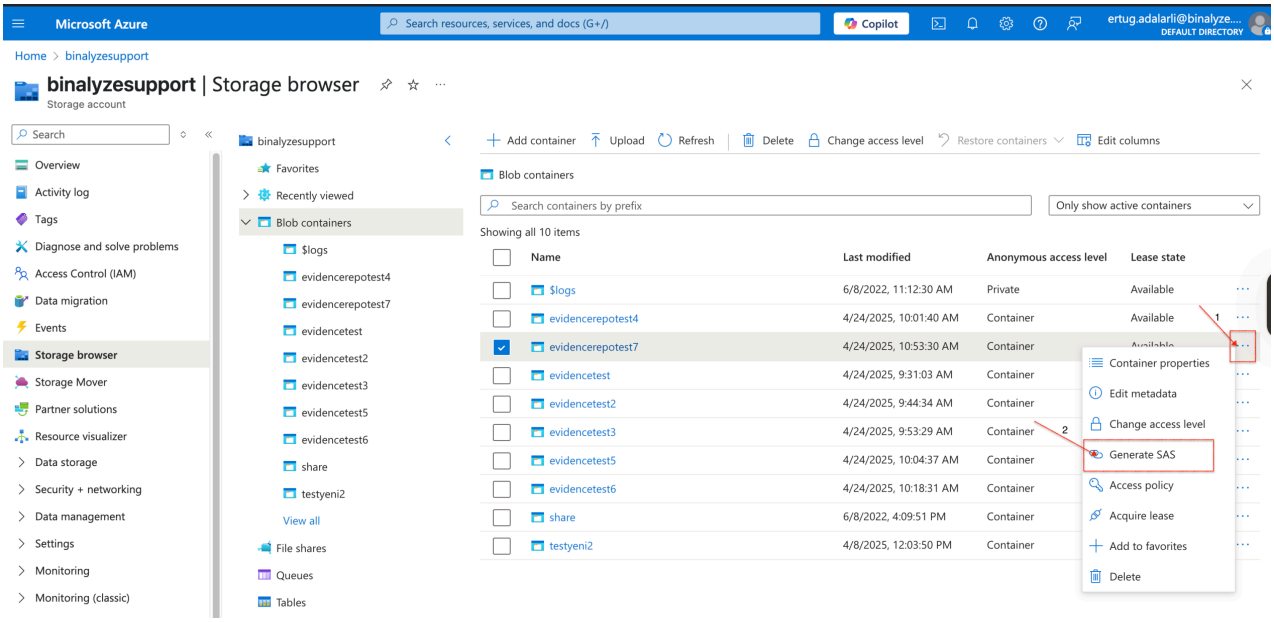
- Log in to the **Azure Portal**.
- Navigate to your storage account and follow this path:
  `Storage Browser` → `Blob Containers`



Generating a SAS URL: **Access Azure Blob Container**

Generate the SAS URL

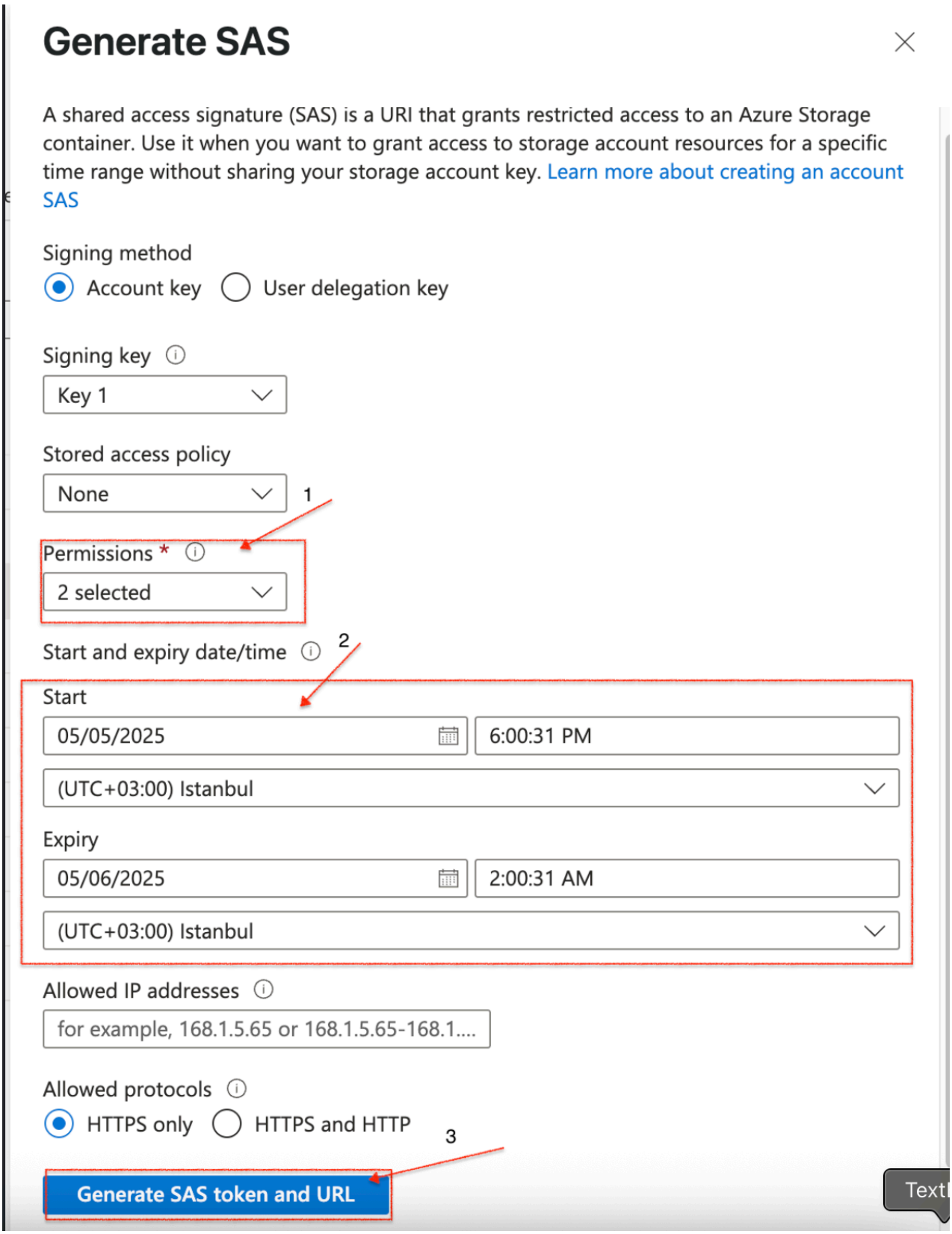
- Locate your container in the list.
- Click the **three dots (...)** next to the container name.
- Select **“Generate SAS”** from the menu.



Generating a SAS URL: Generate the SAS URL:

Configure SAS Settings

- A panel will appear on the right.
- Here you can define:
  - **Permissions** (Recommended: `Create`, `Write`, and optionally `List`)
  - **Token validation period** (start and expiry date/time)



Generating a SAS URL: Configure SAS settings

Copy the SAS URL

- After setting the permissions and time frame, click **“Generate SAS token and URL”**.
  Example: If you update the token expiration time or adjust permissions, please regenerate and update the SAS URL in AIR accordingly.

File Explorer

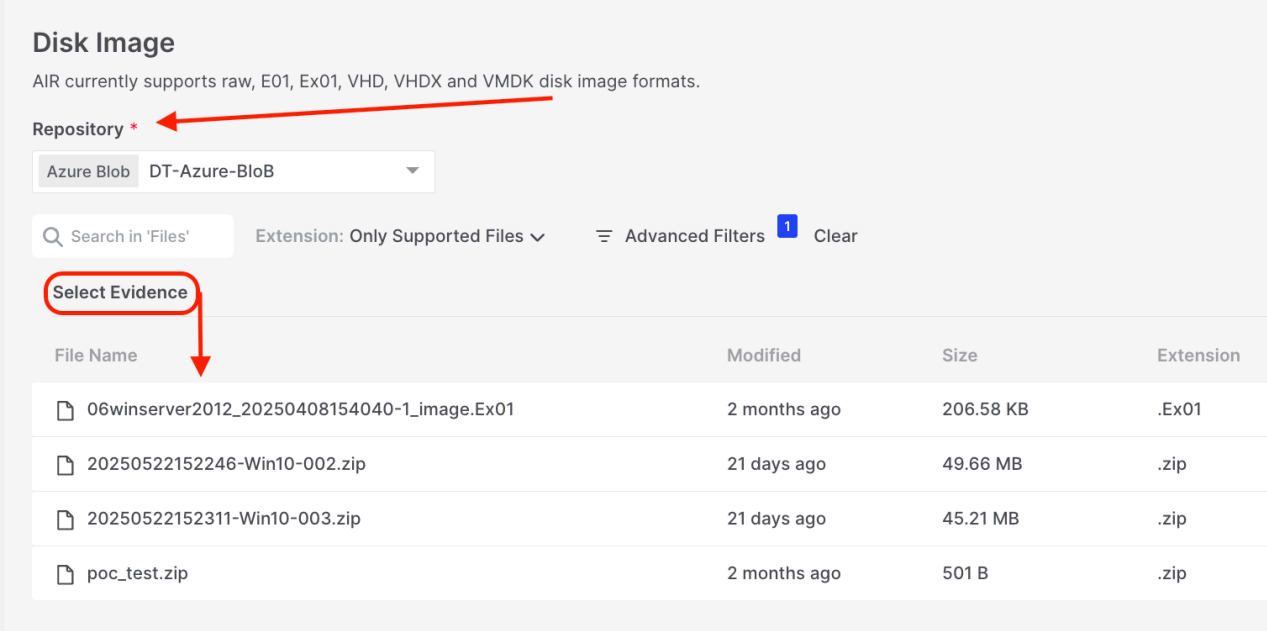
XDR Forensics can be used to explore the file systems of Windows, macOS, and Linux systems where full disk or volume images have been acquired in either the **RAW (dd), EWF (E01/Ex01), VHD/X, or VMDK formats**.

The forensic image can be added from your SMB, SFTP, Amazon S3 bucket, or Azure Blob storage to XDR Forensics as a new asset in a simple three-step process:

- 1. On the Assets page, click on the 'Add New' button and then select Disk Image:

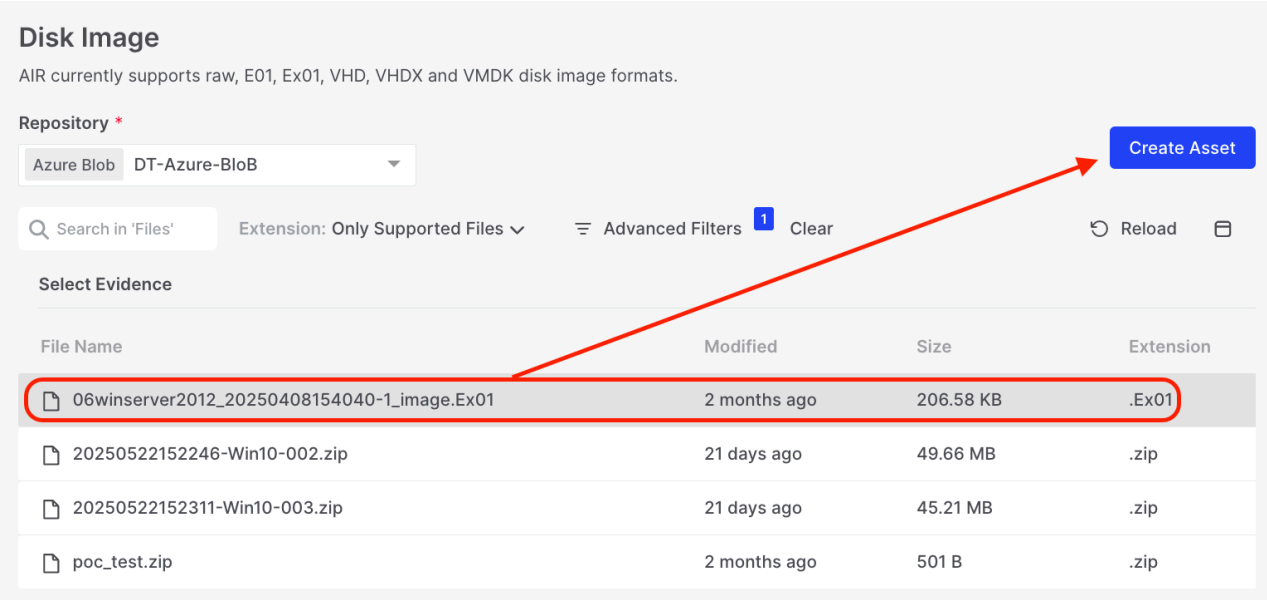
File Explorer: Add a Disk Image

- 2. Select your connected repository and then select the first segment of the RAW, EWF or VMDK file you wish to mount and explore:



File Explorer: Select the first segment of an image file from the evidence list

- 3. Select 'Create Asset':



File Explorer: Create Asset

The image must be supplied to XDR Forensics from your **SMB, SFTP, Amazon S3 bucket, or Azure Blob storage** evidence repositories; segmented files are supported.

AIR Image Files & File Explorer

Image Format	Supported locations	Supply to File Explorer as a Single File?	Supply to File Explorer as Segmented Files	Can be generated by AIR as a single file	Can be generated by AIR as a segmented file
RAW(dd)	SMB, SFTP, Amazon S3 bucket, Azure Blob Storage	Yes	Yes	Yes	Yes
E01	SMB, SFTP, Amazon S3 bucket, Azure Blob Storage	Yes	Yes	No	No
Ex01	SMB, SFTP, Amazon S3 bucket, Azure Blob Storage	Yes	Yes	Yes	No (AIR generates a single Ex01 File)
VMDK	SMB, SFTP, Amazon S3 bucket, Azure Blob Storage	Yes	Yes	No	No
VHD/VHDX	SMB, SFTP, Amazon S3 bucket, Azure Blob Storage	Yes	Yes	No	No

File Explorer: Compatibitly chart

Tool Tip for File Explorer users:

- **Ex01 and E01 Images**: These are accessible immediately in File Explorer. Using XDR Forensics to generate Ex01 files avoids the need to unzip files in the Evidence Repository.
- **DD Images**: Generated in a zip file by XDR Forensics. To access, connect to the Evidence Repository, unzip the zip file, and then mount or explore the image in File Explorer.

Next, select your new asset from the Assets table to launch the XDR Forensics File Explorer. The asset's directory structure will appear in the secondary menu (highlighted below), allowing you to browse and select individual files for inspection in Hex, Text, or Metadata views.

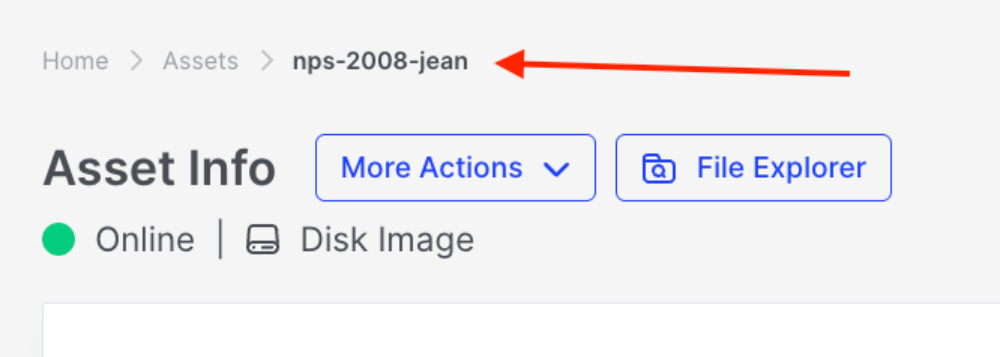
File Explorer: Directory Tree displayed in Secondary Menu

A file can be selected with a right-click to download it locally or calculate its hash values.

Advanced filters can be applied to filter the files displayed.

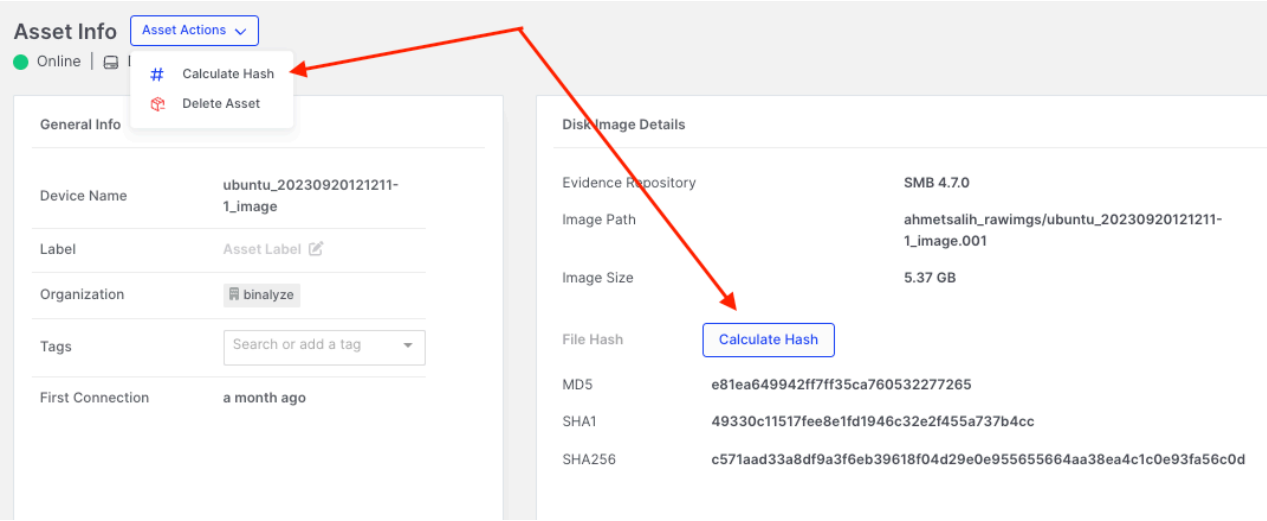
File Explorer - Calculate Hash for disk images

Navigating to the root of the Device Name in the breadcrumb path opens the Asset Info page for the mounted disk image:



File Explorer: Asset Info

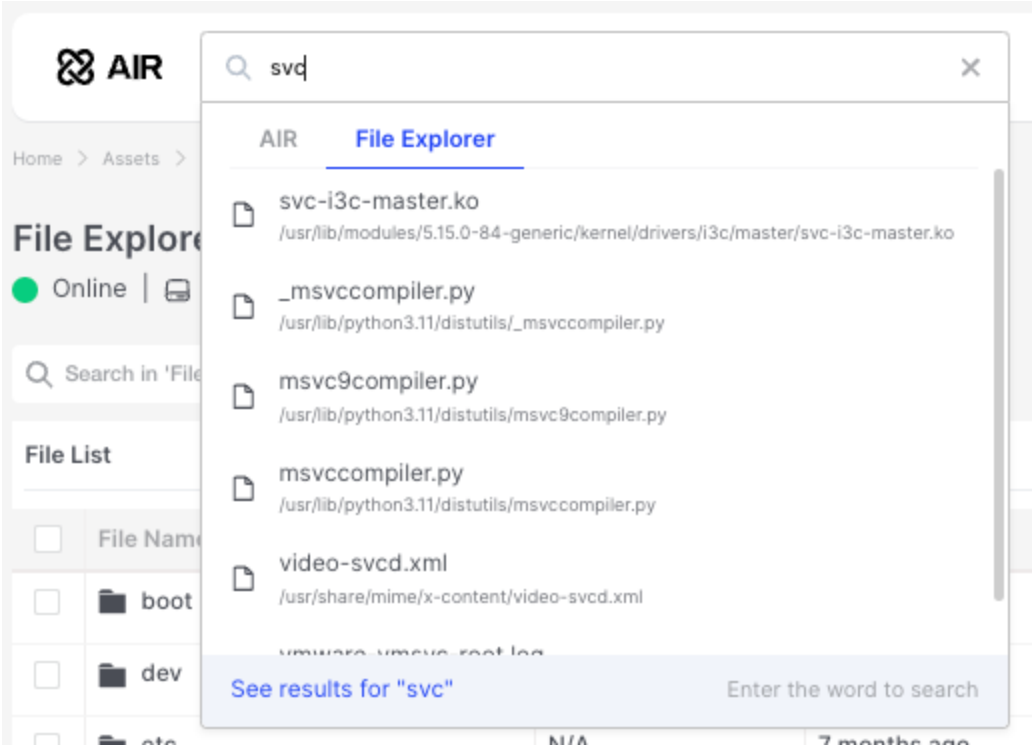
- When a disk image is added as an asset to XDR Forensics, users can now calculate the hash value of that image file either through the Asset Actions button or from the Disk Image Details window.
- MD5, SHA1, and SHA256 are all calculated simultaneously.
- This hash function can be carried out at any time.



File Explorer: Hash Calculation

File Explorer - Recursive Search

- Recursive searching is now possible in the XDR Forensics File Explorer via the Global Search box, where the File Explorer tab will display any hits found in the File Explorer.



File Explorer: Recursive Search

This is just the beginning of our File Explorer project - many more features are planned, and your feedback is most welcome.

File Explorer - FAQs

Q1. What is the XDR Forensics File Explorer?

The XDR Forensics File Explorer is a built-in, cross-platform GUI browser tool within XDR Forensics, designed for seamless navigation of full disk or volume image files. It allows users to explore directory structures and efficiently inspect individual file contents in Hex, Text, or Meta Data views.

Q2. What value does File Explorer provide?

A full disk or volume image can provide crucial evidence for digital forensic investigations. Offering investigators easy access to disk image files allows them to navigate investigations swiftly in a unified view, minimizing the need for mastering multiple tools.

Q3. What are the plans for File Explorer?

The initial release of **File Explorer (v4.7)** introduced the ability to explore raw disk or volume images. Since then, we've added support for more image file types, whether they are segmented or single files. Subsequent versions have expanded functionality, adding support for additional disk types for ingestion into **XDR Forensics** and increasing the range of supported remote storage locations from which these image files can be pulled, presented, and mounted within the **XDR Forensics File Explorer**.

Please see this page, XDR Forensics File Explorer ↗, for the list of supported disk types and evidence repositories from which they can be mounted.

Our plans include introducing a live version of File Explorer, enabling users to explore files directly on remote assets, and DRONE support.

Q4. Are there any requirements to use XDR Forensics's File Explorer?

Existing customers (installed XDR Forensics 4.7 and below) who have yet to do so will need to add the new **'air-tornado'** Docker container to support the **File Explorer** feature - please contact the support team to help do so. Customers who installed XDR Forensics v4.7 or later will already have the additional Docker container.

Q5. Why is a new Docker container necessary to support the feature?

Refer to Q15.

Q6. What does RAW, EWF, VMDK, or VHD/X mean when discussing an image file?

Whether acquired by XDR Forensics or captured by a third-party tool, a disk image can only be explored in the XDR Forensics File Explorer if it's in the RAW(dd), EWF (E01/Ex01), VMDK, VHD/X formats:

1. **RAW** format is a bit-by-bit copy of a disk or volume, preserving all data without interpretation or modification. XDR Forensics supports the generation of single or segmented RAW image files.
2. **EWF2** is also commonly used in digital forensics to store and compress digital evidence but this format supports additional metadata. XDR Forensics supports the generation of single EWF2/Ex01 files but **NOT** segmented EWF2/Ex01 image files.
3. **VMDK** (Virtual Machine Disk) is a file format used to store virtual disk images for VMware virtual machines. It enables the encapsulation of an entire hard drive, including its filesystem and data, into a single file or set of files, facilitating easy management, migration, and backup of virtual environments.
4. **VHD (Virtual Hard Disk):** VHD is an older virtual disk format used by Microsoft solutions like Hyper-V. It supports up to 2TB and is ideal for compatibility with legacy systems.

   VHDX (Virtual Hard Disk Extended): VHDX is the newer format, supporting disk sizes up to 64TB with improved performance and data protection. Use it for modern systems or larger disks.

NB re VMDK: We support reading RAW (Flat), COWD, and VMDK extent file formats.

For VMDK, we support the following disk types:

- monolithicSparse
- twoGbMaxExtentSparse
- monolithicFlat
- twoGbMaxExtentFlat
- streamOptimized

However, the "vmfs" and "thin" disk types are not supported.

We can also open and read snapshots, but the actual VMDK files are required as well.

Q7. How does it work?

Clicking the “New Asset” button on the Asset page now offers the option to add “Disk Image”. Users can navigate to an **SMB, SFTP, Amazon S3, or Azure Blob storage** shared location, select the disk image, and add it as a new asset. Once created, users can browse the directory structure in the asset detail screen.

Q8. Where can the image file be saved so XDR Forensics can access it?

Currently, the image must be supplied to XDR Forensics from an **SMB, SFTP, S3 or Azure Blob storage** shared location, saved as:

- a single or segmented RAW file, **OR**
- a single or segmented E01/Ex01 file **OR**
- a single or segmented VMDK file **OR**
- a single or segmented VHD/VHDX file.

Q9. What are some important details about the displayed information?

The hash of the image file isn't calculated automatically to save time during asset creation. You can calculate the image hash on the Asset Info page as soon as it is created.

Various views for selected files include hex view, text view, and metadata view.

Q10. How long does it take to create an asset from an image file?

Testing has shown that a 107GB full disk image is available for exploration in XDR Forensics's File Explorer within 20 to 30 seconds after hitting the “Create Asset” button.

Q11. Can I bookmark items/files in File Explorer?

Not yet, but it's in the pipeline.

Q12. What actions are users able to take from within File Explorer?

Users can calculate the hash of individual or multiple files using the bulk action bar, and download individual files.

Q13. What other enhancements are planned?

In the future, users may have the ability to right-click a file for an option to 'hunt' for that item or other assets attached to the XDR Forensics console. Plans also include allowing users to run triage and acquisition tasks on the disk image and add the collections/results to the Investigation Hub.

Q14. Can deleted files be carved from File Explorer?

We present the logical file system, so we can only access existing files and folders; carving unallocated space is not an option currently.

Q15. Are we pulling raw data into the console or just parsed data?

In the XDR Forensics console, the latest Docker container, named 'air.tornado', acts as a proxy for commands from the front end. When an action is requested, such as opening a folder, XDR Forensics sends the command to the container, which reads the folder data from the disk and provides it to the front end on demand. Recent updates include indexing the entire image for search capabilities without downloading the entire image to the XDR Forensics server. The image remains in the evidence repository, so a revoked token from the evidence repository will result in a lost connection.

interACT

Purpose-Built, Multi-Asset, Cross Platform Remote Shell for DFIR

interACT Commands

interACT has been built specifically for DFIR capability in $ Product. The full list of current commands can be listed by typing ‘help’ at the command prompt, and is below the following important 'hint':

> The interACT command-line parser utilizes Unix-like command-line parsing methods due to the libraries used and the absence of Windows-specific libraries. Because of that, **a Windows user will have to write a del command like this**:
>
> - del C:/xyz/abc.txt # use forward slashes
> - del 'C:\xyz\abc.txt' # within single quotes
>
> The following is currently invalid and likely to remain so in the future due to Windows' non-standard approach to command-line parsing and character escaping.
>
> - del C:\xyz\abc.txt # Invalid
> - del "C:\xyz\abc.txt" # Invalid

cat: To display the content of a file.

cd: To change the current working directory.

curl: To make HTTP requests.

del, delete, or **rm**: For deleting a file or folder.

dir or ls: Will list the files and folders in a chosen directory.

exec or **execute**: The exec or execute action allows you to execute a process on the asset without a shell and capture its output via stdout and stderr.

find: Will search for a file or directory.

get: To pull a file from the asset down to the console

hash: Will display the hash value of a file.

head: To get the first 10 lines of a file displayed.

help: Will display any help messages and switches that you can apply to commands available to you at your current position.

hex: Will display the hex-encoded output of the first 100 bytes of a file.

image: To read a disk or volume and write its contents out as a .dd file. This can also be done from the UI, but remains here in interACT for those who prefer to image from the command line.

kill: Is the command to terminate a process.

mkdir: Will make or create a directory

osquery: Gives the user access to osquery queries within the interACT session

pslist: Will display the running process list.

put: Allows the user to push a file from the library to the asset.

pwd: Displays the present working directory.

volumes: Will list the mounted volumes.

yara: Scan files or processes with YARA rules.

zip: This command compresses or decompresses a file or folder.

Flags

From XDR Forensics v4.5, users can speed up workflows by using the following flags for files they want to download using the ‘get’ command in interACT:

- Compression: '-zip'
- Password protection: '-zip-password'
- File name change: '-name'

BEWARE!

Using `zip -p` on machines monitored by EDR can trigger alerts due to its association with suspicious activities like encryption or data exfiltration.

EDRs often flag or **block such commands**, log passwords exposed in plaintext, and create compliance challenges.

XDR Forensics's InterACT offers a secure alternative for file handling and remote actions, eliminating the need for these risky commands. To ensure smooth operations, XDR Forensics users should work with their security teams to get XDR Forensics executables whitelisted in their EDR. This prevents unnecessary alerts and guarantees uninterrupted, secure workflows during investigations.

PowerShell commands in interACT

Introduction

In Digital Forensics and Incident Response (DFIR), **PowerShell** has become a powerful tool for investigators and analysts. Sometimes overlooked is its compatibility with AIR's interACT, which provides a true cross-platform remote shell for Windows, Linux, and macOS. This KB article aims to shed light on how users can leverage PowerShell within interACT to execute cmdlets and perform a variety of operations.

Why is this Important?

Many DFIR investigators rely on PowerShell (and Python) as their primary scripting and remediation tools. However, newcomers to AIR may assume that interACT is exclusively tailored for Linux, which is not the case. interACT is a versatile platform, and specific commands are available to users of both Windows and UNIX-like operating systems.

Executing PowerShell in interACT

PowerShell can be executed in interACT through several methods. Here, we'll explore three basic ways to run PowerShell commands:

Using the 'exec' Command

The 'exec' or 'execute' command in interACT serves as a gateway to run PowerShell commands. This versatile command enables DFIR practitioners to integrate PowerShell into their workflows seamlessly. Below are examples of how to use 'exec' with PowerShell:

```
exec powershell.exe whoami
```

This command executes a simple PowerShell 'whoami' cmdlet, displaying the currently logged-in user.

```
exec powershell.exe Get-ScheduledTask
```

In this instance, 'exec' invokes the 'Get-ScheduledTask' cmdlet, providing insights into scheduled tasks on the system.

```
exec powershell.exe Remove-Item -path C:\\Temp\\example.txt
```

The 'exec' command facilitates the removal of a file ('example.txt') using the 'Remove-Item' cmdlet from a specified path.

⚠ When running exec commands in InterACT, please note that commands requiring additional user input (e.g., Get-CimInstance prompting for a ClassName) may not display the prompt dynamically during execution. Instead, InterACT will continue running and appear to "hang" until the process times out or completes.

To avoid this, we recommend:

1. **Modify the Command:** Specify all required parameters directly in the command to prevent prompts. For example:
   &#xNAN;`exec powershell.exe Get-CimInstance -Namespace 'root\SecurityCenter2' -ClassName "YourClassName"`
2. **Test Commands Locally First:** Run the command in a native PowerShell console to ensure all required inputs are included before executing it in InterACT.

We are aware of this behavior and are continuously working to improve user experience.

Understanding the `-NonInteractive` PowerShell Flag in interACT

When using **PowerShell** commands in scripts or automated workflows, you may encounter scenarios where PowerShell expects user input. This can disrupt execution, especially in non-interactive environments such as **AIR's interACT** automation. To address this, PowerShell offers the `-NonInteractive` flag.

What is `-NonInteractive`?

The `-NonInteractive` flag is a command-line option for `powershell.exe` that instructs PowerShell to operate in non-interactive mode. When this mode is enabled, PowerShell does **not** prompt for user input and will terminate the script or command if user input is required.

This feature is particularly useful when running commands in environments where no user interaction is possible or desirable, such as during forensic investigations or automation tasks initiated via interACT.

Example Use Case in interACT

Here's an example of how the `-NonInteractive` flag can be applied within **interACT**:

```
powershell.exe -NonInteractive Get-CimInstance -Namespace 'root\SecurityCenter2'
```

Explanation:

- `powershell.exe`: The executable for running PowerShell commands.
- `-NonInteractive`: Ensures the command runs without expecting user interaction.
- `Get-CimInstance -Namespace 'root\SecurityCenter2'`: Retrieves information from the specified namespace.

This command is designed to collect system security information without risking a prompt for user input that could interrupt execution.

Using the `-NonInteractive` flag in AIR's interACT provides the following advantages:

- **Seamless Automation**: Prevents disruptions in workflows caused by unexpected prompts.
- **Increased Reliability**: Ensures consistent execution of PowerShell commands, even in headless or remote environments.
- **Enhanced Efficiency**: Minimizes delays during investigative or forensic operations.

Troubleshooting Tips

If a command using `-NonInteractive` fails:

1. Check the command syntax for errors.
2. Ensure the command does not inherently require user input.
3. Review interACT logs for additional context on the failure.

For more details, refer to the official PowerShell documentation on `about_PowerShell_exe`.

Additional Useful PowerShell Commands and Syntax

1). Here are some additional PowerShell commands that can be invaluable in cyber investigations:

```
exec powershell.exe Get-Process
```

This command retrieves information about running processes, which is crucial for understanding system activity.

2). You can query specific log entries within a shorter time frame. Here's an example to retrieve Security log events from the last 24 hours:

```
exec powershell.exe Get-WinEvent -LogName Security -MaxEvents 100 -
Oldest
```

In this command:

- `-MaxEvents 100` limits the query to the most recent 100 events, which should make the query faster.
- `-Oldest` ensures that the query starts with the oldest event, which is from the last 24 hours in this case.

You can adjust the `-MaxEvents` value to retrieve a specific number of events or omit it to get all events from the last 24 hours. This command should provide a quicker response with a smaller dataset.

3). Here's an example of a simple PowerShell command that retrieves information about the local computer's operating system:

```
exec powershell.exe Get-CimInstance -ClassName Win32_OperatingSystem
```

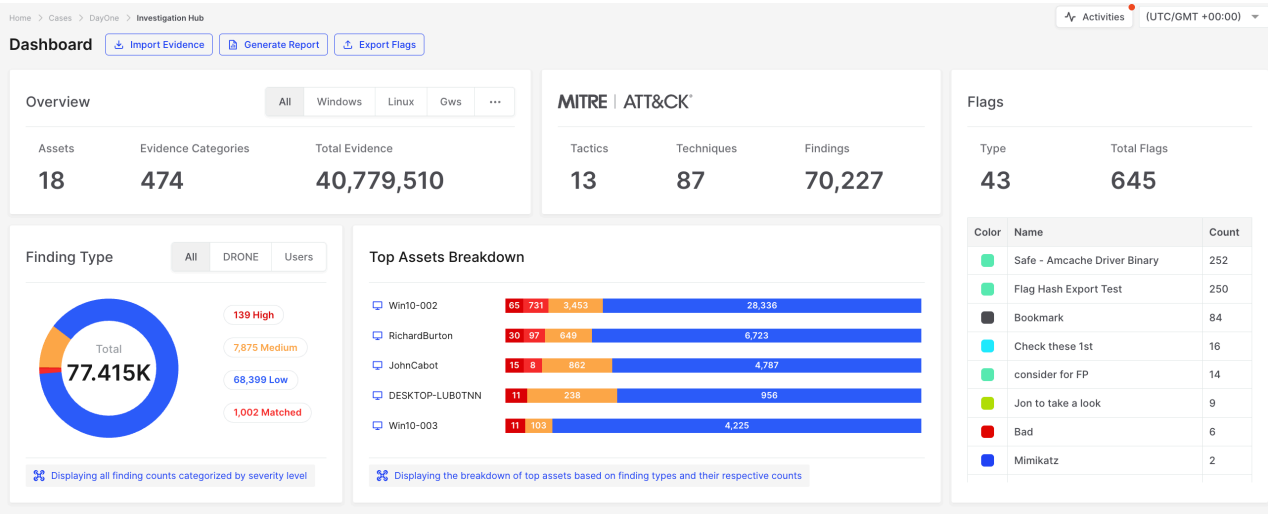
This command uses the "Get-CimInstance" cmdlet to retrieve information about the local computer's operating system. It should execute quickly and provide details about the operating system on the machine where it's run along with other information such as Build Number, Registered User, Serial Number, and Version.

Conclusion

By following these simple examples, users can harness the capabilities of PowerShell within interACT for DFIR investigations and operations. interACT's compatibility across different platforms ensures that investigators can seamlessly incorporate PowerShell into their toolkit, expanding their capabilities and efficiency in digital forensics and incident response.

Investigation Hub

Harness the power of consolidation, prioritization, and collaboration for efficient incident response investigations



Investigation Hub: The Dashboard

What is the XDR Forensics Investigation Hub?

XDR Forensics automates the rapid generation and presentation of a clear **DFIR intelligence report** directly within the Investigation Hub. This report instantly highlights DRONE's findings and consolidates all Acquisition and Triage data from multiple assets into a single view known as the **Investigation Hub**.

This central dashboard immediately **elevates your investigation**, providing analysts with a seamless experience that allows them to sort, exclude findings, filter, flag, bookmark, and easily investigate the data. The user-friendly interface streamlines the analysis process, empowering analysts to efficiently navigate and interpret the information to uncover **insights and actionable intelligence.**

The **Investigation Hub** offers a unified and well-organized view of assets, evidence, artifacts, and triage results within a case. This allows you to efficiently review and concentrate your investigation on pertinent details using filters and a powerful global search function, eliminating the need to switch between screens to piece data together manually.

The Intelligence Hub delivers Findings derived from XDR Forensics's automated **DRONE analyzers**, giving you a head start in any investigations.

With DRONE's proprietary analyzers, combined with YARA, Sigma, and osquery scanning, you can analyze assets and evidence at speed, identifying compromised machines to streamline the process of sifting through often massive datasets.

The integrated **MITRE ATT&CK** mapping provides context to discern the nature of threats, stay ahead of the attack's progression, and pinpoint areas needing further investigation.

The benefits of the Investigation Hub

1. **All in one place** - all XDR Forensics data acquisitions, results of DRONE analysis, and Triage scans of the assets related to a chosen case - are now available in one place, making the analysts and investigators work much faster and simpler.
2. **Efficiency and Speed** - analysts can **easily navigate to a specific endpoint in the Case while simultaneously leveraging information from all** their endpoints in **a high-level overview of the entire Case**. Therefore, much faster decisions can be made, such as where to start and focus investigations, but also where to divert resources when the Investigation Hub highlights new information.
3. All **multi-asset investigations** become far more efficient within the **Investigation Hub,** especially as we now allow users to **'Bring Your Own Evidence' (BYOE):**
    - Seamlessly import .csv files into the Investigation Hub using our data mapping service, accommodating all forms of structured .csv data.
    - Efficiently import and analyze .pst files, enabling the display of email data within the Investigation Hub for a more comprehensive examination.
4. The DRONE findings table can be exported from the Investigation Hub into a .csv file, enabling the integration of DRONE's analysis results into reports, SIEM, or other security tools for the development of custom alerts.

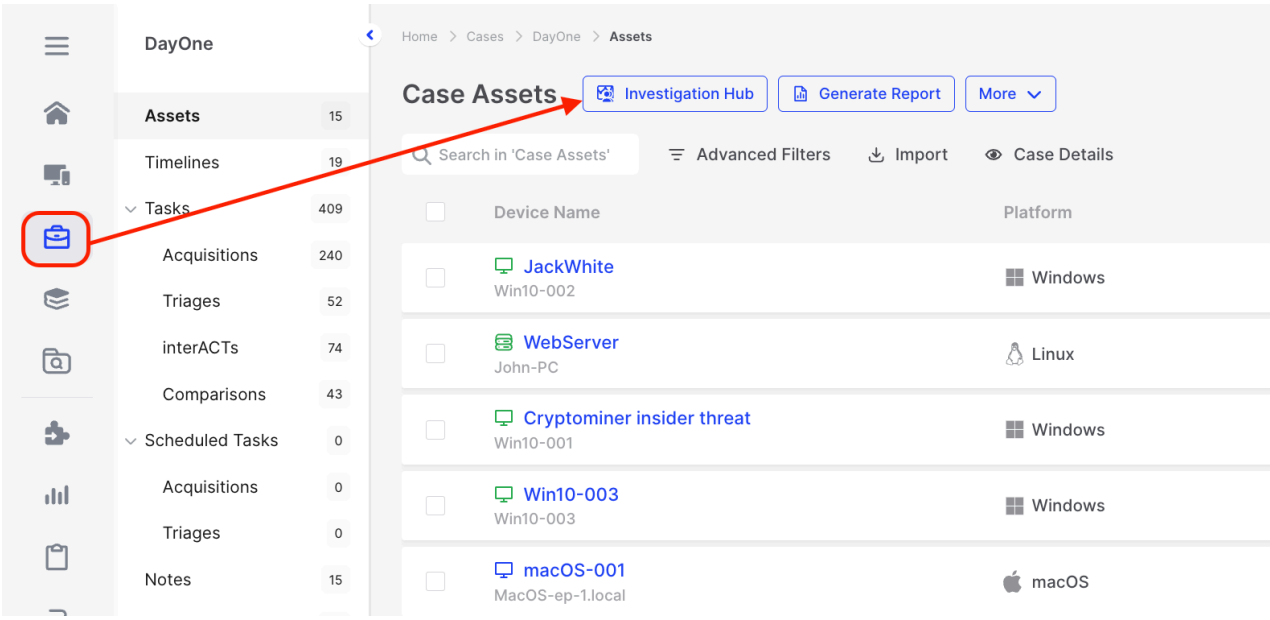
Do I have to install or update my existing Infrastructure?

The Investigation Hub is included as part of the standard XDR Forensics installation.

All of our hardware and software requirements are described here in the Setup section of the KB, no additional infrastructure updates are required.

Where to find the Investigation Hub

The **Investigation Hub** operates at the case level and is generated from the data collected for individual cases. To access it, navigate to 'Cases' in the Main menu. Once you've selected the case of interest, you can access the Investigation Hub via the action button located in the main viewing area:



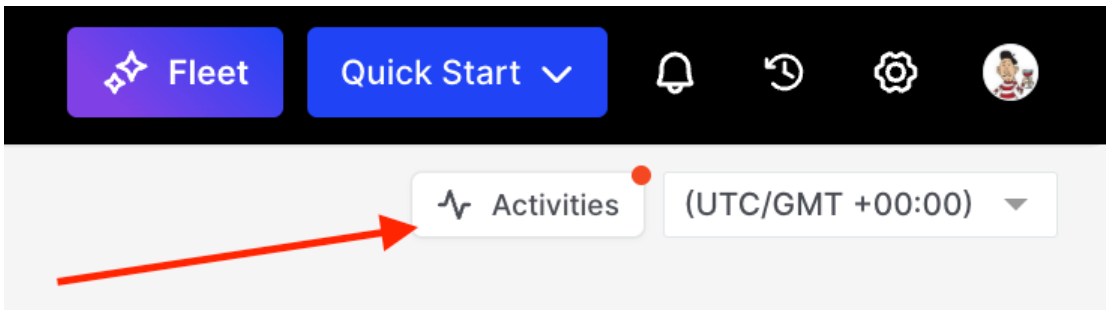
Investigation Hub: Access is via Cases

Collaboration

XDR Forensics is a highly collaborative platform that allows multiple users to access the system simultaneously. Each user's privileges can be finely adjusted based on roles assigned by the system's owner or administrator. As the fastest and most comprehensive DFIR platform globally, XDR Forensics's efficiency is further enhanced through team collaboration.

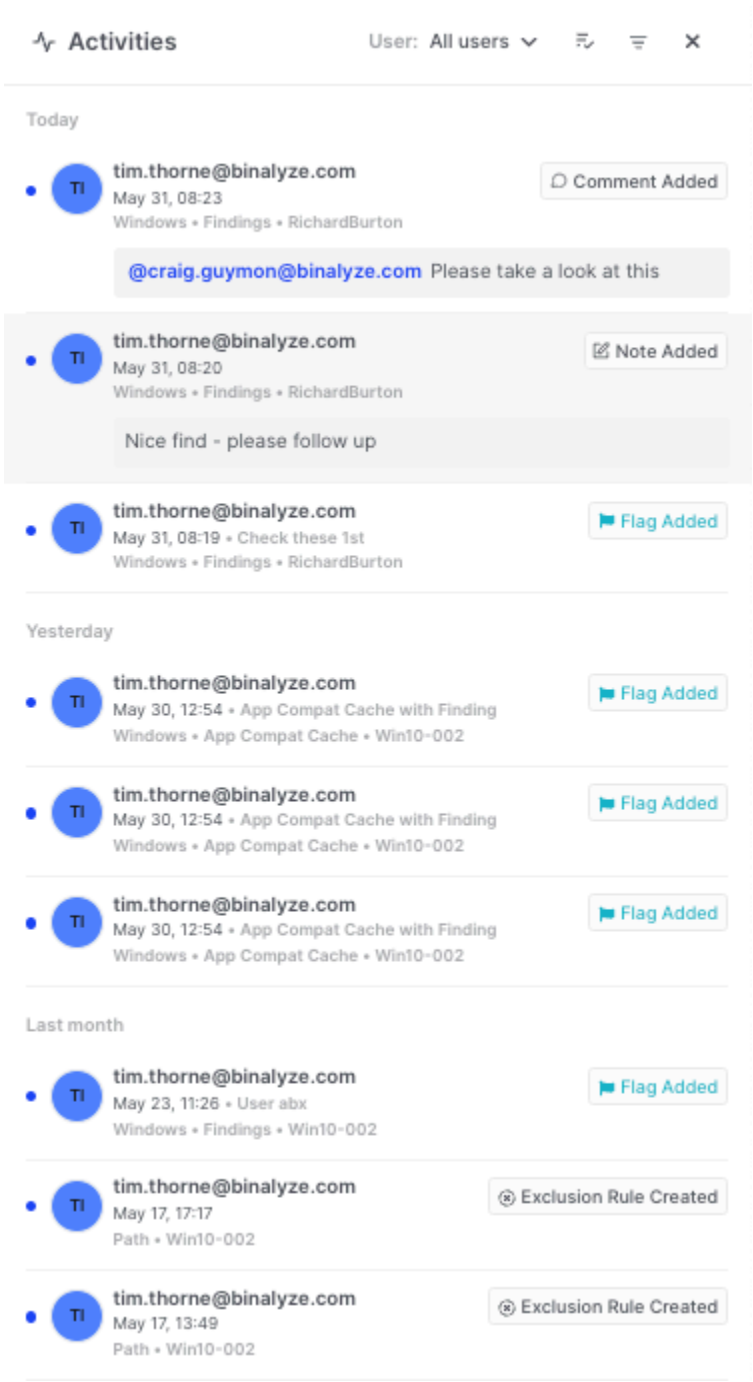
The Activity Feed

- The Activity Feed enhances team collaboration and transparency by logging actions taken by investigators. This includes creating exclusions, findings, flags, comments, and notes. Each entry includes user identification and timestamp information to ensure a comprehensive audit trail.



Investigation Hub: Activities Action Button

- All activities are labeled and linked to their corresponding individual activity by simply clicking on them. In the example below, we can see how Comment Added, Note Added, Flag Added, and Exclusion Rule Created have all been tracked as activities.

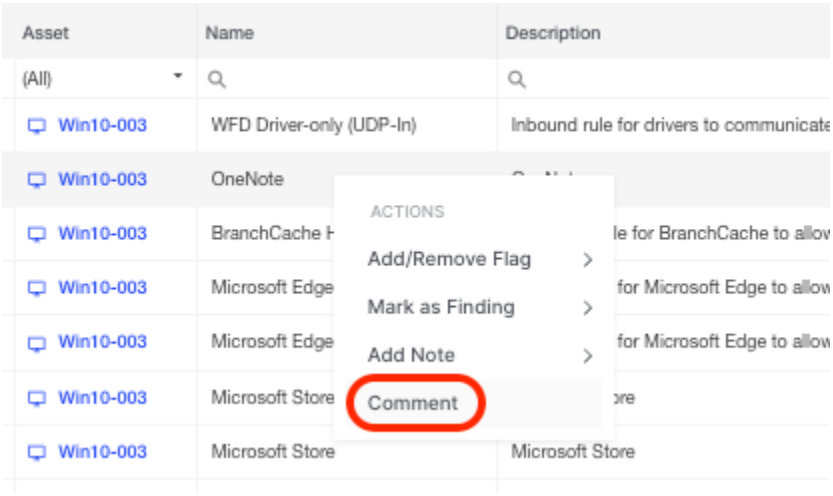


Investigation Hub: Activities

Adding Comments to the Evidence

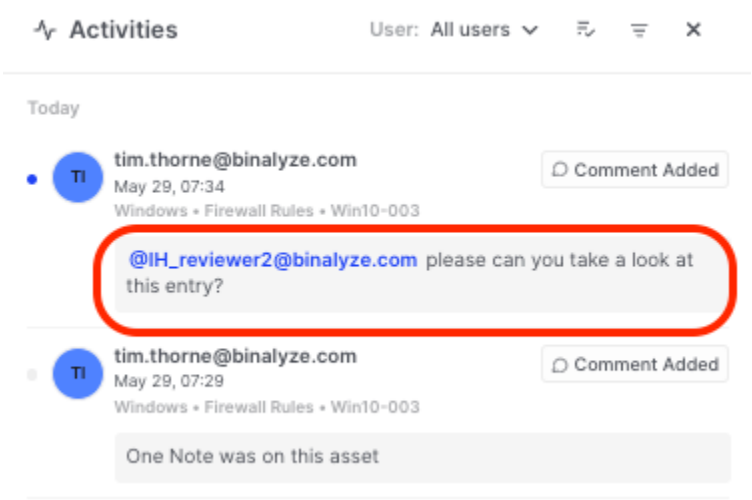
Comments enhance communication by allowing analysts to directly comment on findings and tag relevant colleagues. This ensures that all discussions are captured and documented within the activity feed, promoting effective collaboration and activity tracking.

Right-click on an item and select ‘Comment’ to attach your comment to that item:



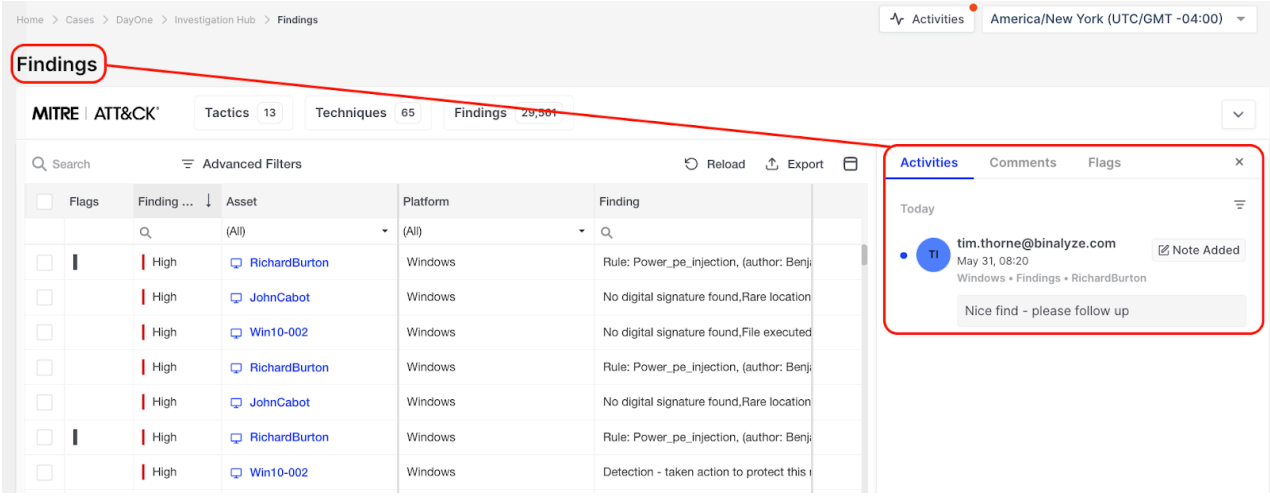
Investigation Hub: Comments

You can tag users in a comment, and they can view the item by clicking on the comment in their Activity Feeds:



Investigation Hub: Tag users

Each table will show all of the Activities, Comments, and Flags that are relevant just to that table



Investigation Hub: Activities Comments and Flags

Investigation Hub – Data Usage Statistics Dashboard

The **Investigation Hub Disk Usage Statistics Dashboard** empowers users to manage **Investigation Hub data storage** effectively by providing detailed and user-friendly insights into disk usage. This feature simplifies data analysis through visual elements like pie charts and summaries and enables users to focus on relevant data using customizable filters. Additionally, the ability to generate PDF reports makes sharing and documenting disk usage information seamless, enhancing operational efficiency and decision-making.

Why This Feature Matters

Challenges:

- **Lack of Visibility**: Users struggle to understand how disk space is utilized across cases, organizations, and evidence categories.
- **Complex Data Presentation**: Existing tools make it difficult to analyze or visualize disk usage.
- **No Reporting Tools**: There is no simple way to generate and share reports, limiting collaboration and record-keeping.

How This Dashboard Solves These Problems:

- **Clear Visualization**: Pie charts and summaries make it easy to understand disk usage at a glance.
- **Customizable Filters**: Users can filter data by organization, investigation type, platform, evidence type, and category.
- **Report Generation**: Generate PDF reports or export data as CSV files directly from the dashboard.

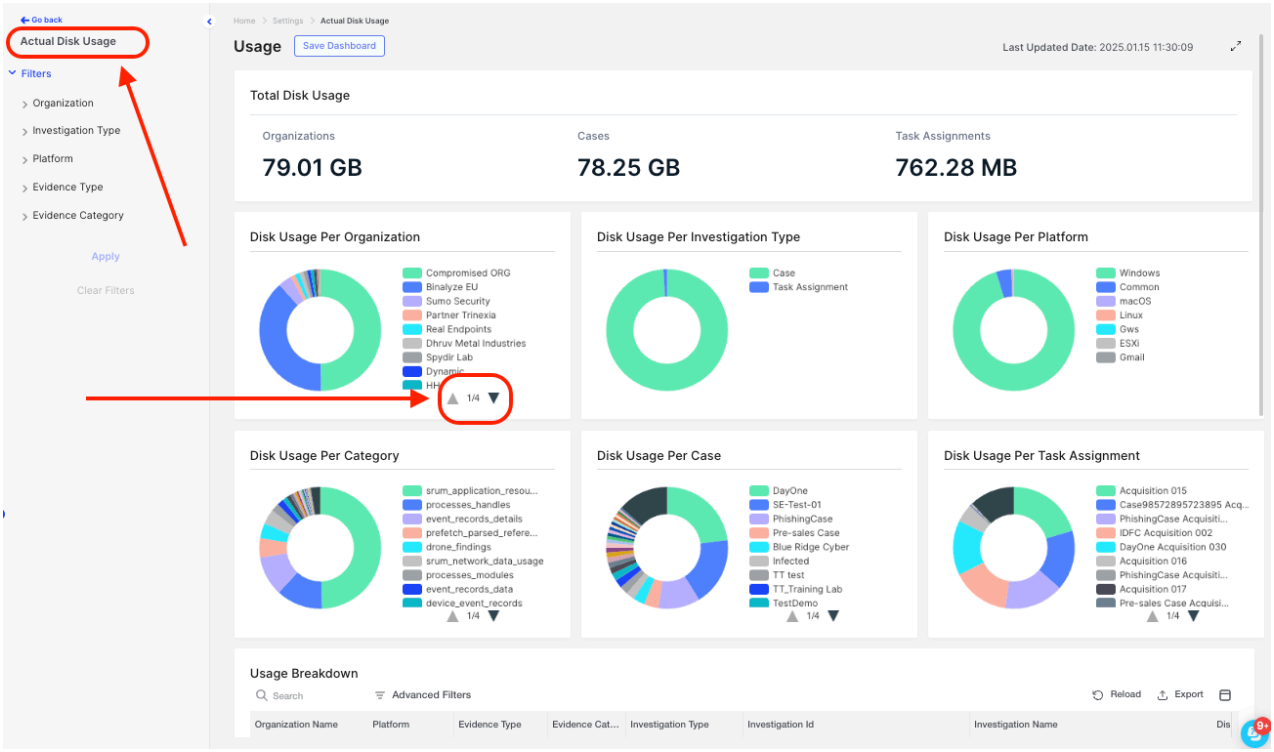
Key Features

Access Control

- This feature is accessible **exclusively to Global Admins** via the XDR Forensics console:
  - Navigate to **Settings > Investigation Hub Disk Usage**.

What's Included

- The first nine disk usage categories are displayed in charts. The remaining categories are grouped together and can be viewed by using the scrolling arrow, as shown in the screenshots of the Disk Usage Per Organization pie chart below:

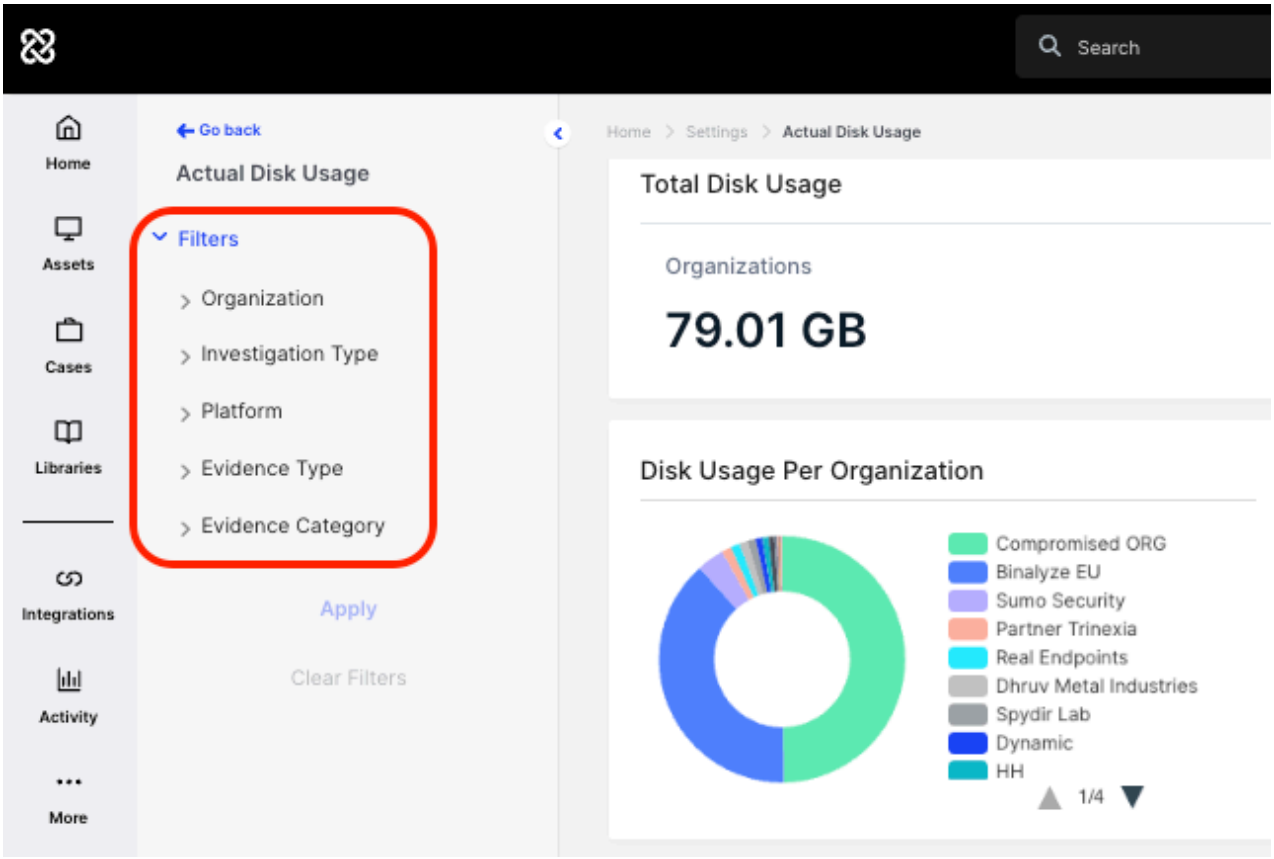


Investigation Hub – Data Usage Statistics Dashboard

- The usage summary window at the top of the page provides an overview of total disk usage, categorized by Organizations, Cases, and Task Assignments, all at a glance.
- Detailed breakdowns are shown by:
  - **Organization**
  - **Case**
  - **Platform**
  - **Evidence Type**
  - **Evidence Category**

Data Views

- **Actual Disk Usage View**:
  - Real-time insights into current disk usage.
- **Historical Insights View**:
  - Track trends and changes in disk usage over time.
  - Configure widgets to display tailored insights.
- **Filters in the secondary menu:**
  - Organization
  - Investigation Type
  - Platform
  - Evidence Type
  - Evidence Category



Investigation Hub – Data Usage Statistics Dashboard: Filters

Advanced Table View

- Sortable and exportable tables for deeper analysis.
- Filters and advanced sorting for granular insights.

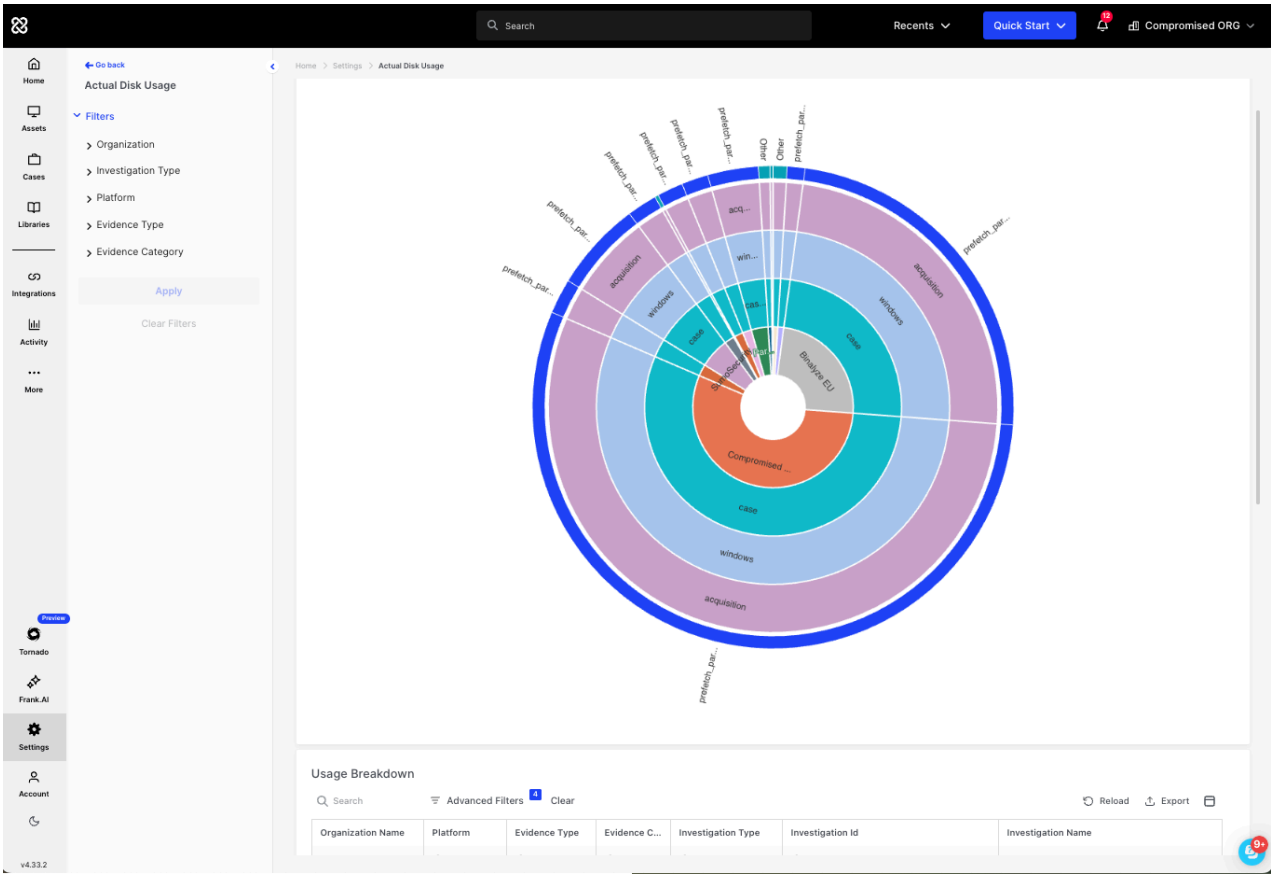
Exporting Reports

- Users can generate PDF reports for record-keeping and sharing.
- CSV exports are available for custom data analysis.

Data Usage Dashboard – Hierarchical Sunburst View

Understanding data usage in complex environments can be challenging, especially with large datasets. The **Hierarchical Sunburst View** in the Data Usage Dashboard offers an intuitive and interactive visualization that helps identify key usage patterns, trends, and anomalies at a glance.

Please see this short overview of the Hierarchical Sunburst View ↗.



Investigation Hub – Data Usage Statistics Dashboard: **Hierarchical Sunburst View**

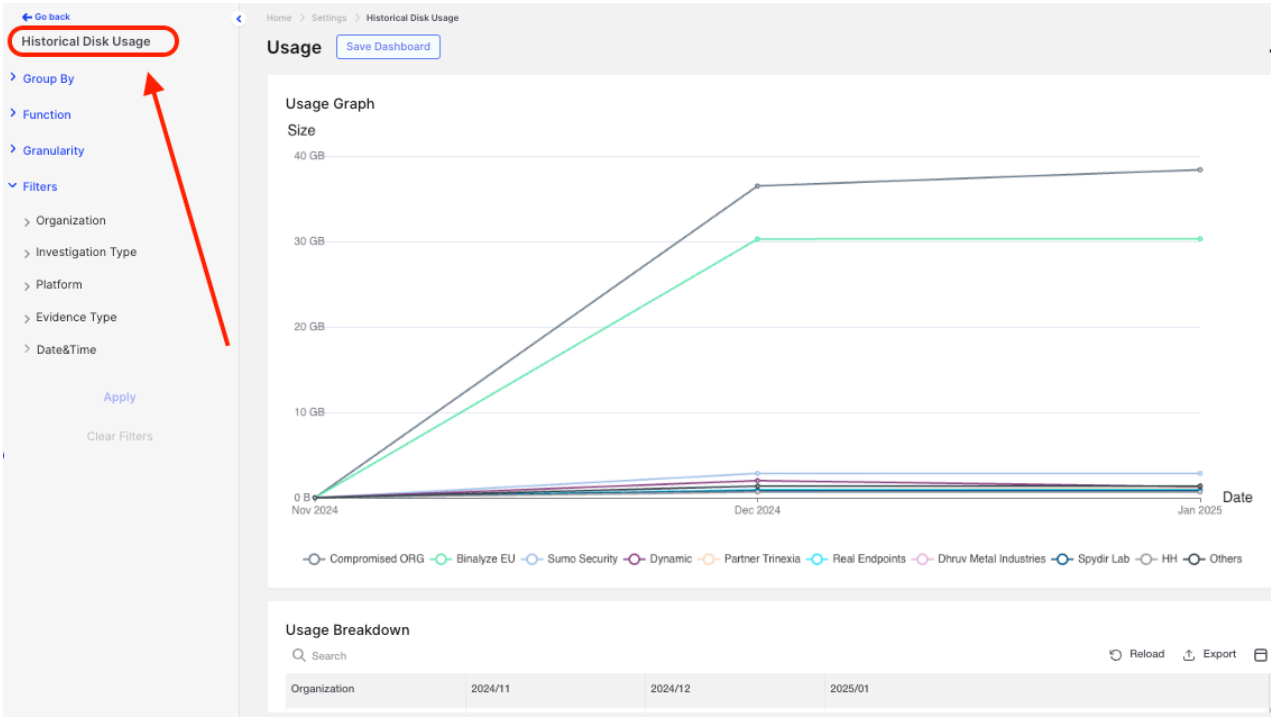
Key Benefits

- **Visual Hierarchy** – Easily navigate and understand data relationships compared to traditional tables or charts.
- **Interactive Exploration** – Click on segments to drill down into deeper data layers for more detailed insights.
- **Seamless Filtering** – The Sunburst View integrates with global filters, ensuring a consistent analysis experience.
- **Custom Dashboards** – Save customized Sunburst views for quick access and continuous monitoring.

Where to Find It?

The **Sunburst View** is available under:
Settings > Actual Disk Usage > Hierarchical View tab on the **Actual Usage** page.

Historical Insights Feature



Investigation Hub – Data Usage Statistics Dashboard: Historical Disk Usage

Overview

The historical view allows administrators and investigators to analyze disk usage trends efficiently. The dashboard includes customizable widgets and granular time-based analytics to help identify patterns and optimize storage usage.

Key Functionalities:

1. **Customizable Widgets**:
   - **Group by**: None, Organization, Investigation Type, Platform, Evidence Type, Evidence Category.
   - **Function**: Metrics such as average or maximum values.
   - **Granularity**: Hourly, daily, monthly, or yearly intervals.
   - **Filters**: Date/time, investigation type, and more.
2. **Saved Dashboards**:
   - Save and manage customized dashboards.
   - Dashboards are listed with details like name, user, save date, last update, and applied filters.
   - Prevent duplicate or empty dashboard names.
3. **Data Management**:
   - Disable unavailable dates in the date and time picker.
   - Reload or export data directly from table views.
4. **Default Settings**:
   - Reset widgets to default configurations.

Restrictions and Considerations

- **Access**: This feature is only available to Global Admins.
- **Non-Clickable Charts**: The line charts in the dashboard are for display purposes only.
- **Limited Categories**: Only the top nine disk usage categories are displayed; all others are grouped under the label “Others.”

User Guidance

1. **Accessing the Dashboard**:
   - Go to **Settings > Investigation Hub Disk Usage**.
   - Choose between **Actual Disk Usage** or **Historical Disk Usage** views.
2. **Filtering Options**:
   - Organization
   - Investigation Type
   - Platform
   - Evidence Type
   - Evidence Category
3. **Visual Representation**:
   - Pie charts for quick overviews.
   - Detailed tables for in-depth analysis.
4. **Exporting and Reporting**:
   - Generate **PDF reports** for sharing and record-keeping.
   - Export **CSV files** for custom data handling.

This feature provides administrators and investigators with powerful tools to monitor and optimize disk usage, ensuring a streamlined and efficient investigation process.

Using the Investigation Hub

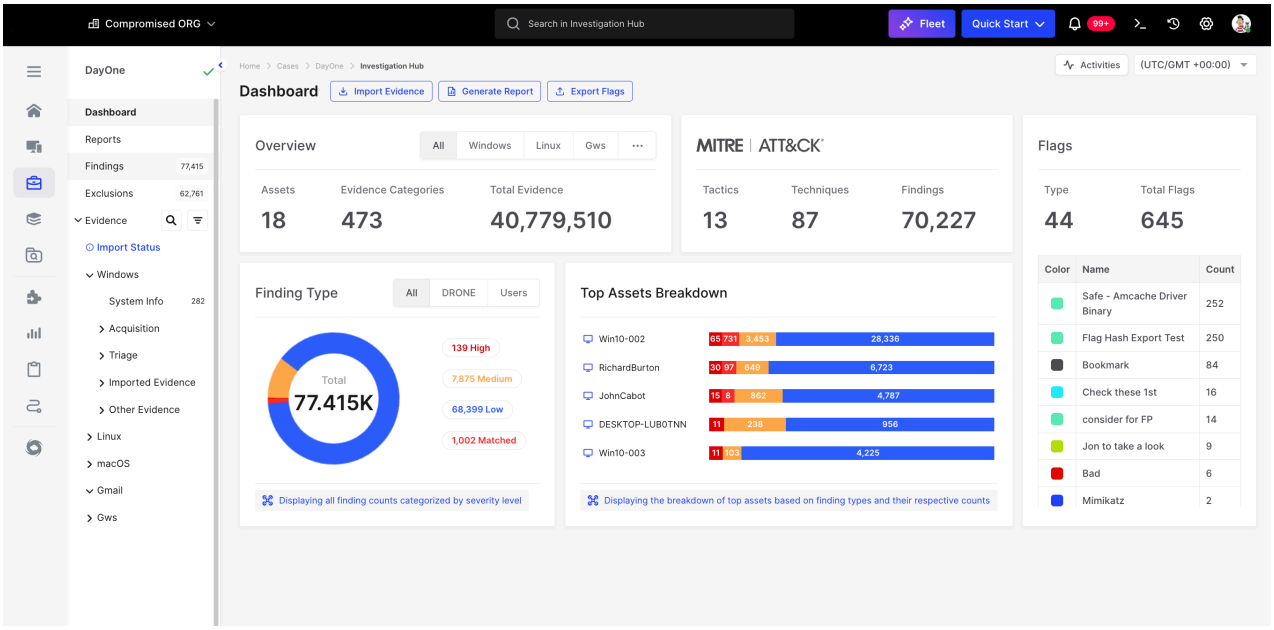
The **Investigation Hub** offers a centralized and well-organized interface to manage all case-related elements, including assets, evidence, artifacts, and triage results. It simplifies the investigative process by providing efficient filtering options and a robust global search feature, eliminating the need to switch between tools or manually piece together information from various sources.

This hub is designed to enhance the investigative process by seamlessly integrating additional data sources and context through data import capabilities. This means you can augment your analysis by importing relevant data and context, ensuring that you have access to a comprehensive and updated information set for your investigation.

The Investigation Hub is not static; it is a living and breathing space that will ingest and consolidate every report allocated to a case as the investigation progresses.

The remainder of this page delves into the various sections that comprise the displayed information. It's beneficial to become familiar with this layout to locate the specific data you're interested in easily. Given the diverse nature of investigative needs, users may employ various methods to explore this data.

Navigating the Investigation Hub



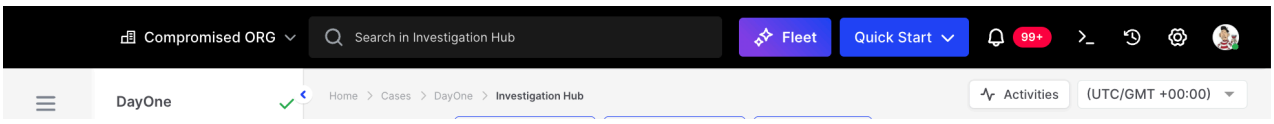
Using the Investigation Hub: The Dashboard

In this article, we will explore the various perspectives presented by the Investigation Hub and offer suggestions on how users can effectively utilize its features.

The Investigation Hub can be broken down into six sections:

1. Header
2. Secondary Menu
3. Dashboard & Widgets
4. Evidence
5. Findings & Results Table
6. MITRE ATT&CK
7. Details View
8. Automated Report Generation

1. Header



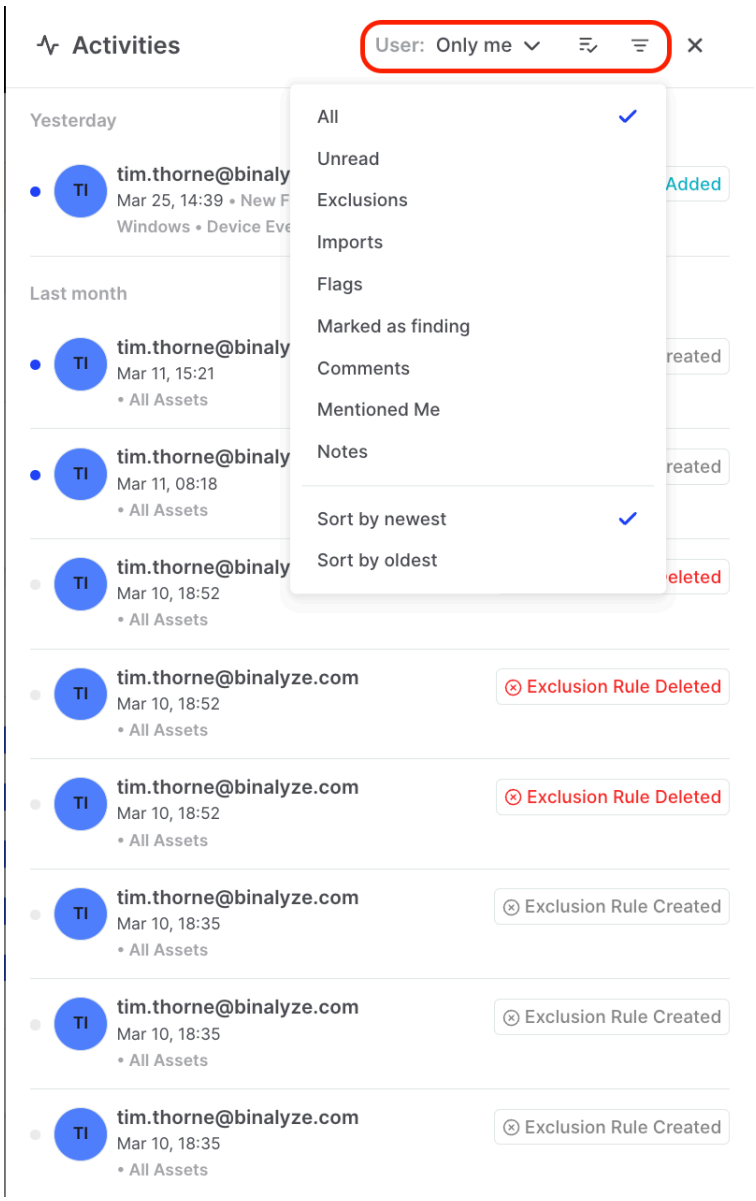
Using the Investigation Hub: Header

The Header is a persistent element across all views within the Investigation Hub.

To the right of the XDR Forensics icon and Organization Name is a **Global Search** input box. This search capability is highly versatile, enabling users to perform searches across all data within the Investigation Hub for the current case. This encompasses acquisition and triage data, as well as any imported data and findings identified by DRONE, ensuring a comprehensive and integrated approach to investigations.

The **Activity Feed** enhances team collaboration and transparency by logging actions taken by investigators. This includes creating **exclusions, findings, flags, comments, and notes**. Each entry includes user identification and timestamp information to ensure a comprehensive audit trail.

All activities are labeled and linked to their corresponding individual activity by simply clicking on them. In the example below, we can see how the user has filtered to 'Only me', showing all activity with the newest first:



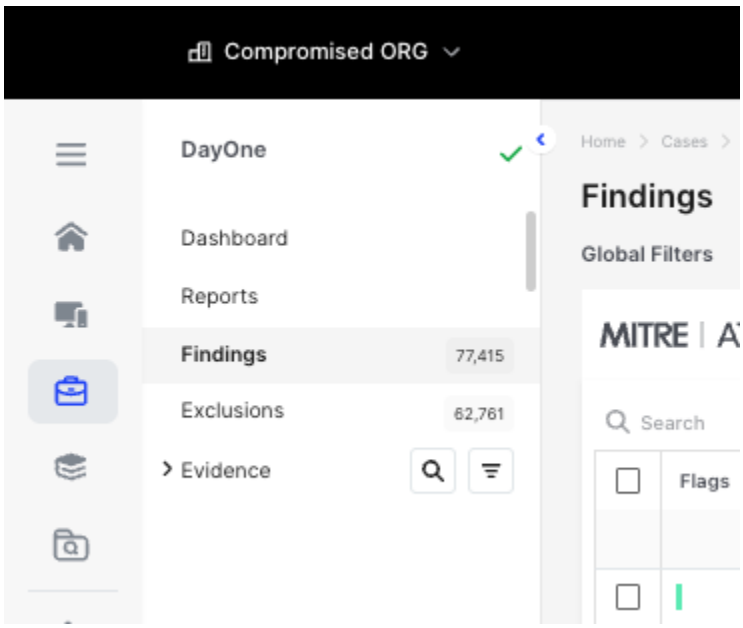
Using the Investigation Hub: Activity Filters

Notifications are accessed via the bell notifications icon, which displays a count (up to 99) of the unread messages.

To the right of the Notifications icon is the name of the **Organizational environment** in which the user is working. From this link, users can access the Organization Settings, change their organization, or add a new one.

Towards the right side of Activities, users have the option to modify **time zone settings** for all timestamps within the hub, should the need arise.

2. Secondary Menu



Using the Investigation Hub: Secondary Menu

At the top of the Secondary Menu, the name of the current case (e.g., 'DayOne' in this example) is prominently displayed. From here, users can choose the view to display in the main viewing window of the Investigation Hub, selecting from:

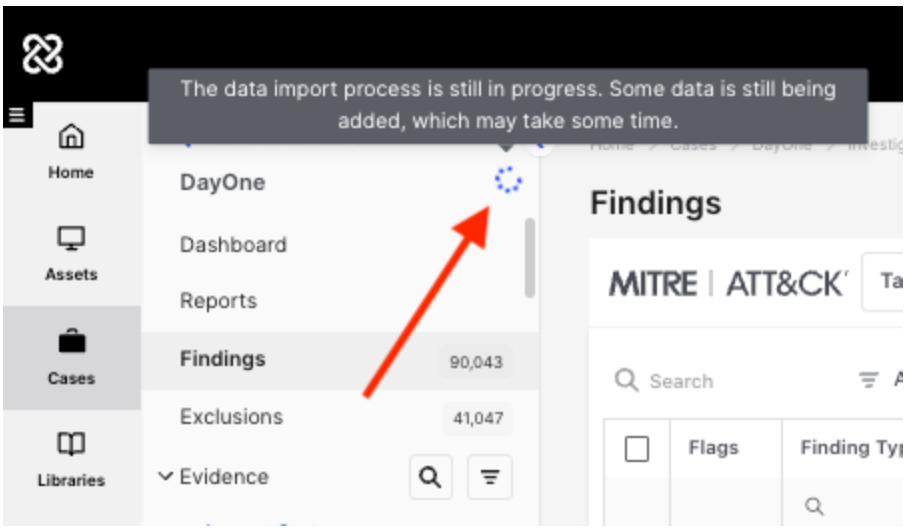
- Dashboard
- Reports
- Findings
- Exclusions
- Evidence

Live Import Progress

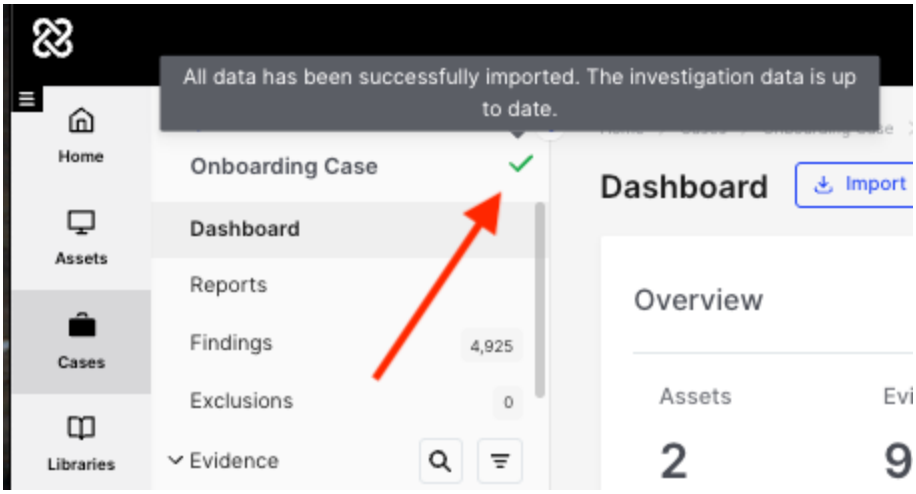
As seen below, the Secondary Menu shows 'live import progress' directly next to the case title. This provides immediate visibility into active tasks without requiring a refresh or switching views.

When hovering over the data import icon, users will now see a clear visual status:

- A green tick indicates that the import has successfully completed.
- A spinning circle of dots signifies that the import is still in progress.



Using the Investigation Hub: Data import still in progress

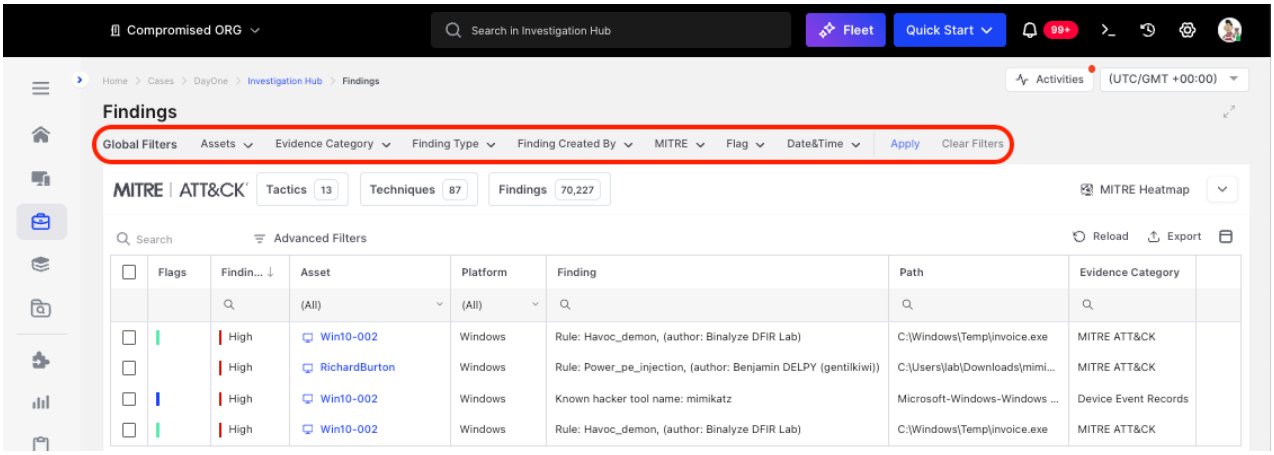


Using the Investigation Hub: Data imports are up-to-date

The bottom half of the Secondary Menu is dedicated to applying Global Filters to the current case, enhancing the ability to narrow down the displayed information. Users can filter by:

- Asset
- Finding Type
- Flag
- Dates and Times
- Creator

These filtering options, combined with the logical AND switch, enable users to customize and refine the display to show only the most relevant items based on multiple selected criteria. This structured approach helps streamline navigation and improve the efficiency of the investigative process in the Investigation Hub.



Using the Investigation Hub: Global Filters

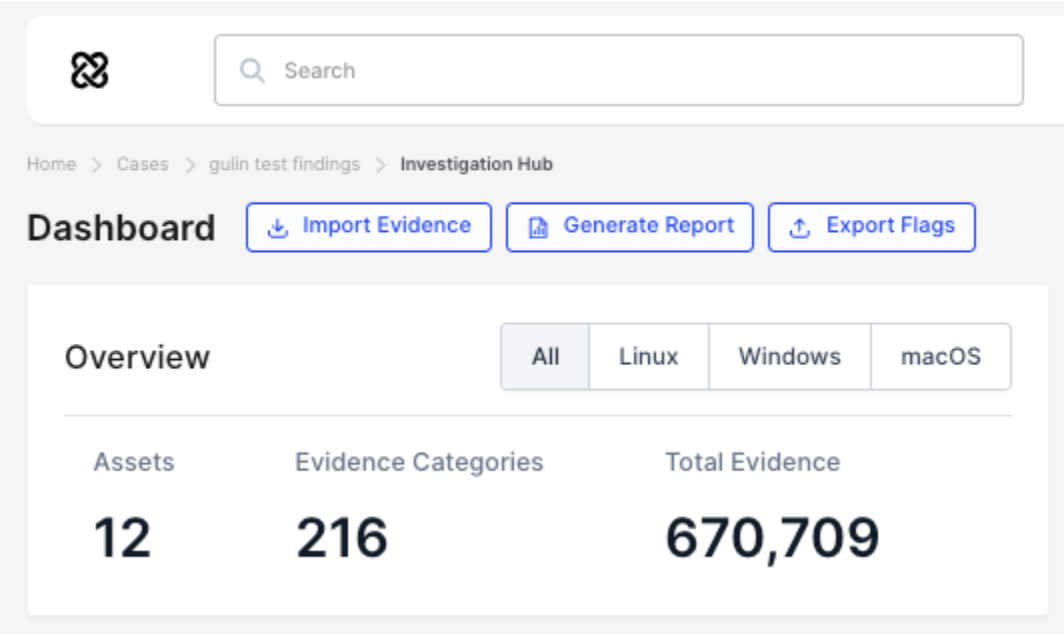
Global Filters in Investigation Hub

Global Filters are positioned at the top of the Investigation Hub, directly below the section title. This placement ensures they remain visible and readily accessible across all views.

These filters allow users to refine data in the tables below by criteria such as Assets, Evidence Category, Finding Type, MITRE mapping, Flags, and Date & Time. The *Evidence Category* filter enables more targeted investigation by isolating specific types of evidence. Filter selections persist across Investigation Hub views for seamless navigation and analysis.

3. The Dashboard and Widgets

The Dashboard provides investigators with a high-level overview of their case, highlighting key issues and investigative opportunities. It features dynamic widgets that automatically update as new evidence and artifacts are added, ensuring that investigators have the most current information at their disposal.

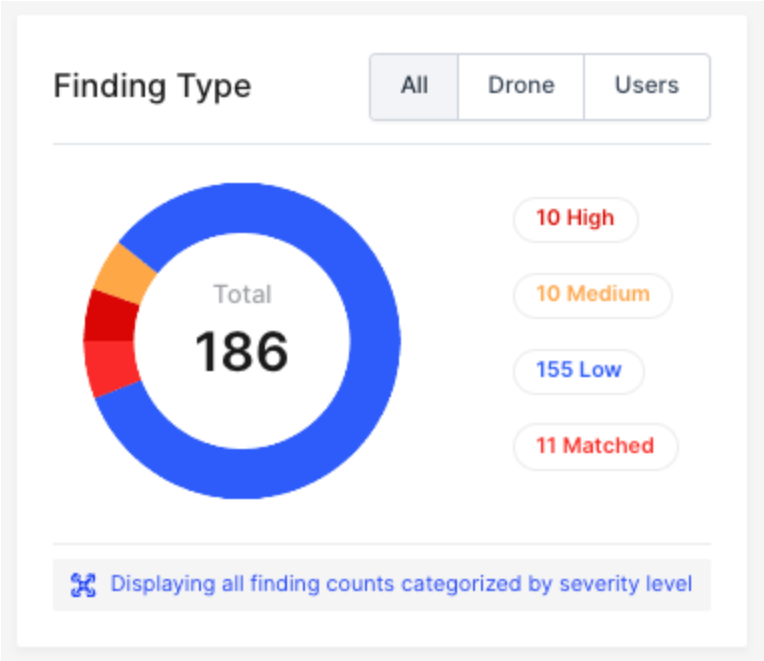


Using the Investigation Hub: Overview

The action buttons on the Dashboard page enable three key activities:

1. **Import .csv or .pst Files**: Add these files directly to your case.
2. **Generate Reports**: Access the report generation wizard.
3. **Export Flags to .csv**: Export all flagged items, including bookmarks.

Additionally, the **Overview widget** allows for filtering by operating system, giving you a quick overview of key statistics for the case.



Using the Investigation Hub: Finding Type

The **Finding Type widget** on the dashboard categorizes and filters findings as either DRONE automated, user-generated, or both. This widget simplifies the review process by presenting findings according to their severity levels:

- **Red**: Indicates High severity
- **Orange**: Denotes Medium severity
- **Blue**: Represents Low severity
- **Dark Orange**: Signifies Matched (keyword/Triage hit)

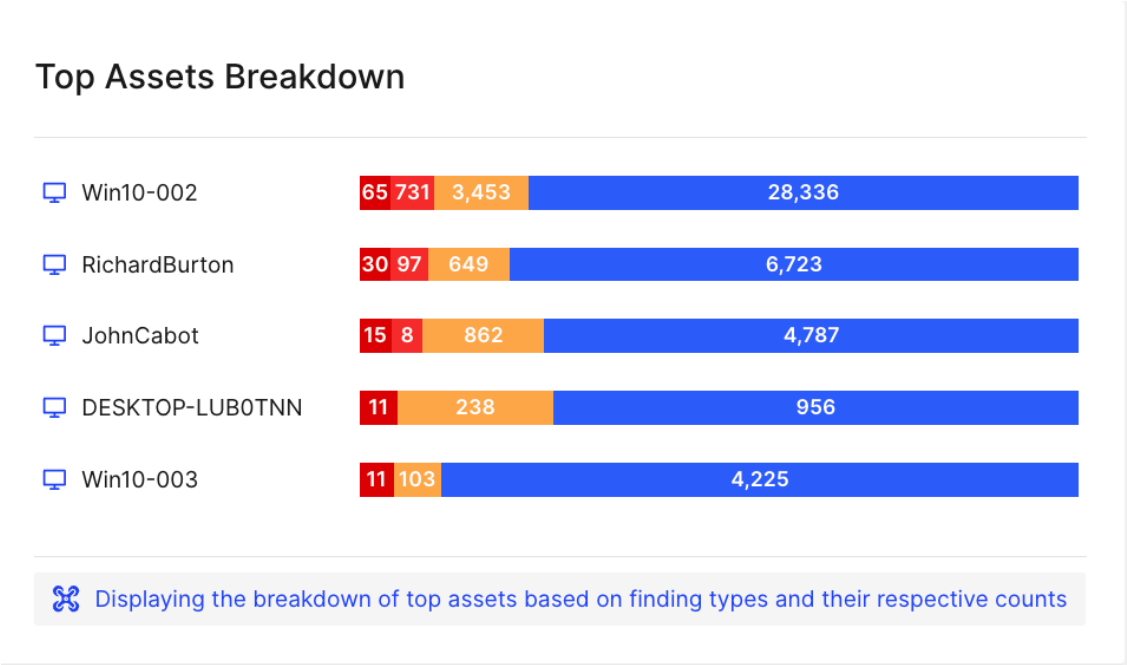
Each item within the widget is clickable, and the filtered results are displayed on the Findings page.

DRONE Finding Types

- **High:** Flags threats that pose immediate and significant risks, demanding urgent action to prevent or mitigate severe impacts. Example: An IIS process executing `cmd.exe` or `powershell.exe`, which could indicate a web shell.
- **Medium:** Targets activities that deviate from expected norms and could be indicative of potential threats, suggesting a need for deeper scrutiny. Example: A running unsigned process located in a temporary folder, or the use of known hacking tools, such as "mimikatz."
- **Low:** Identifies less critical but still unusual activities that could benefit from further investigation to clarify their nature and intent. Example: System-level processes initiated by non-privileged users, or processes operating from non-existent directories.
- **Matched:** Involves confirmed matches to predefined security rules, keywords, hashes, or patterns within the analyzed data, signaling recognized threat indicators. Example: A detected scheduled task named "MalwareTask*" that aligns with a user-defined keyword "MalwareTask*".

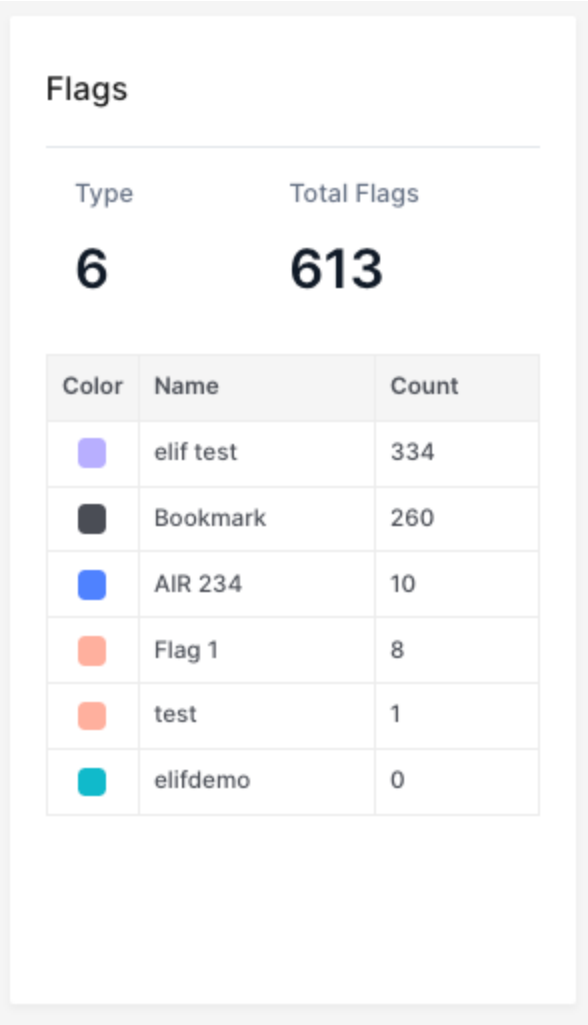
The **MITRE ATT&CK widget** provides an invaluable overview by displaying how many findings within a case have been mapped to various Tactic and Techniques from the MITRE ATT&CK framework. This framework is a globally recognized knowledge base of adversary tactics and techniques based on real-world observations. It is used extensively for threat modeling and cybersecurity defense.

By incorporating the MITRE ATT&CK widget into the dashboard, investigators can quickly identify patterns and methods used in cyber attacks, facilitating a deeper understanding of the threat landscape. This visibility enables users to align their defense strategies more accurately with the tactics and techniques that adversaries are most likely to use, improving the effectiveness of their security measures. The widget's ability to display the distribution of findings across different categories not only helps pinpoint vulnerabilities but also aids in prioritizing responses to the most critical threats based on observed patterns of attack.



Using the Investigation Hub: Top Assets Breakdown

The **Top Asset Breakdown widget** highlights the most significant assets within the current case, emphasizing those with the most severe issues. This feature provides a clear and concise view of critical vulnerabilities, helping investigators prioritize their response efforts based on the urgency and impact of the identified problems.



Using the Investigation Hub: Flags

The "Flags" widget provides a comprehensive overview of all flagged items, each accompanied by its respective count. This widget enables users to customize flags, allowing them to modify both the name and color of each flag to suit their specific needs.

The default "Bookmark" flag is pre-established and cannot be edited. Additional custom flags, once created, are stored and listed within the library for easy access.

***Note: Flags are configured at the Organization level, so they are not just case-specific. Therefore, any edits or deletions to flags will have a global effect for the Organization and are only permitted for users with Case Management privileges. This ensures consistent flagging practices across all cases, preserving the integrity of the flagging system.**

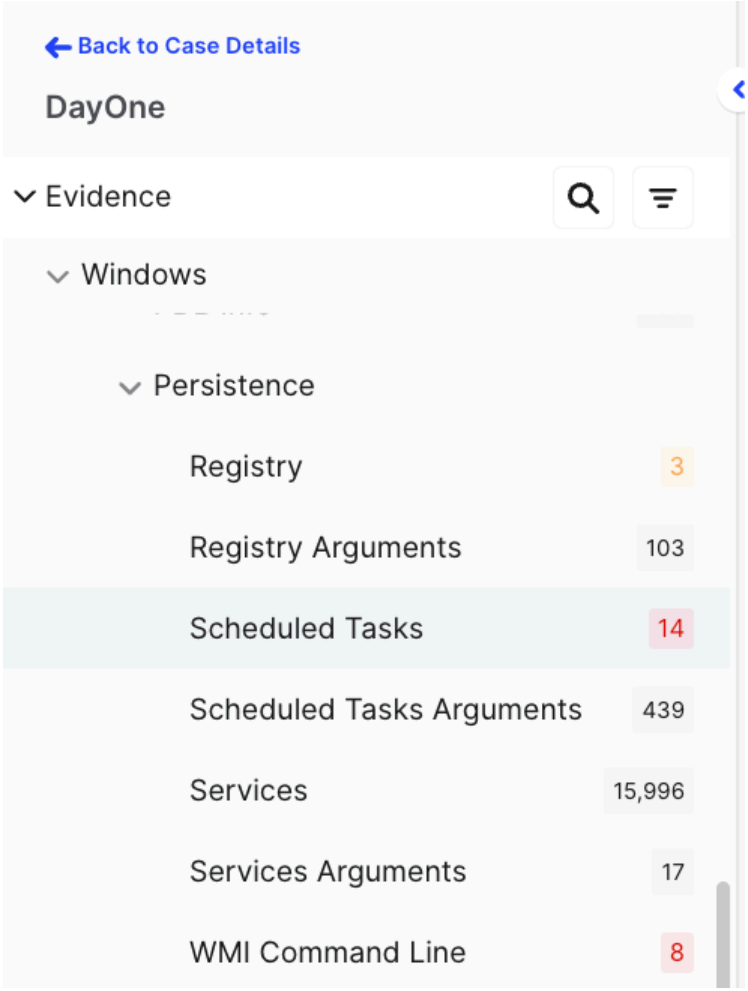
4. Evidence

Under the heading "Evidence," all evidence items collected as part of a tasking, even those with a null return, will be displayed.

A count will be displayed next to each evidence item to indicate the number of associated Findings. For example, in the screenshot below, there are 14 high-priority Findings for Persistence>Scheduled Tasks, 3 medium-priority Findings for Persistence>Registry, and other items are shown with counts but have no Findings associated. This feature helps to identify and prioritize areas of interest within the evidence quickly.

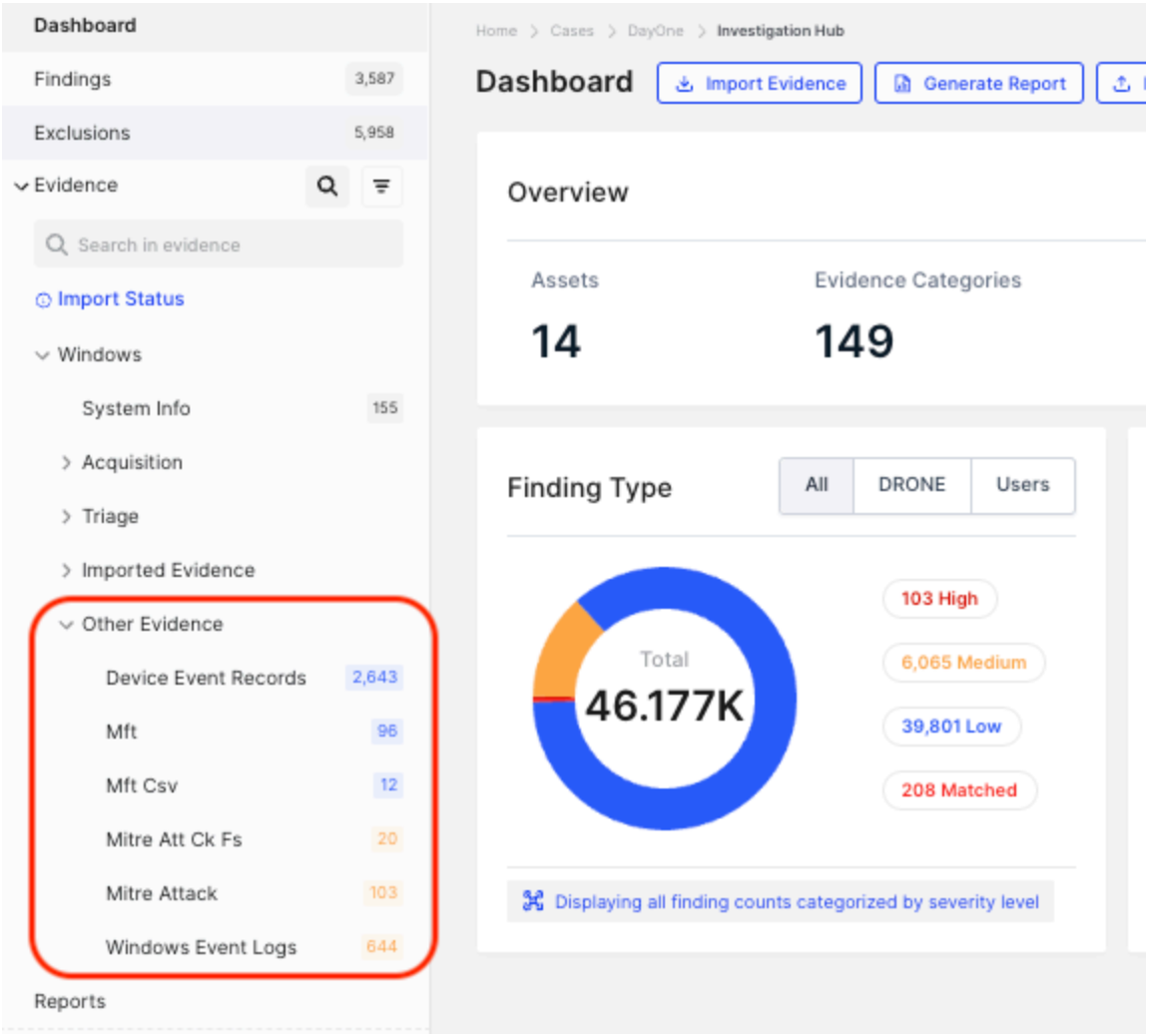
Triage results and Imported evidence items are also displayed in this Evidence section.

Within the Secondary Menu, users have the flexibility to dynamically include or exclude case assets, including imported evidence like .pst or .csv files. Assets are categorized either by the operating system or by the name of the imported evidence for ease of management. In the example above, the Windows asset called RichardBurton has been deselected so none of the evidence from this asset will be displayed in any of the Findings views.



Using the Investigation Hub: Scheduled Tasks has 14 Findings

- **Other Evidence** is a category for items that are Findings without a dedicated entry in the Investigation Hub.
    - All findings without a specific category will be grouped under '**Other Evidence**'. This ensures that every finding is allocated an evidence record within a category, allowing the total count of findings on the Investigation dashboard to accurately match the number shown in the evidence list.



Using the Investigation Hub: Other Evidence

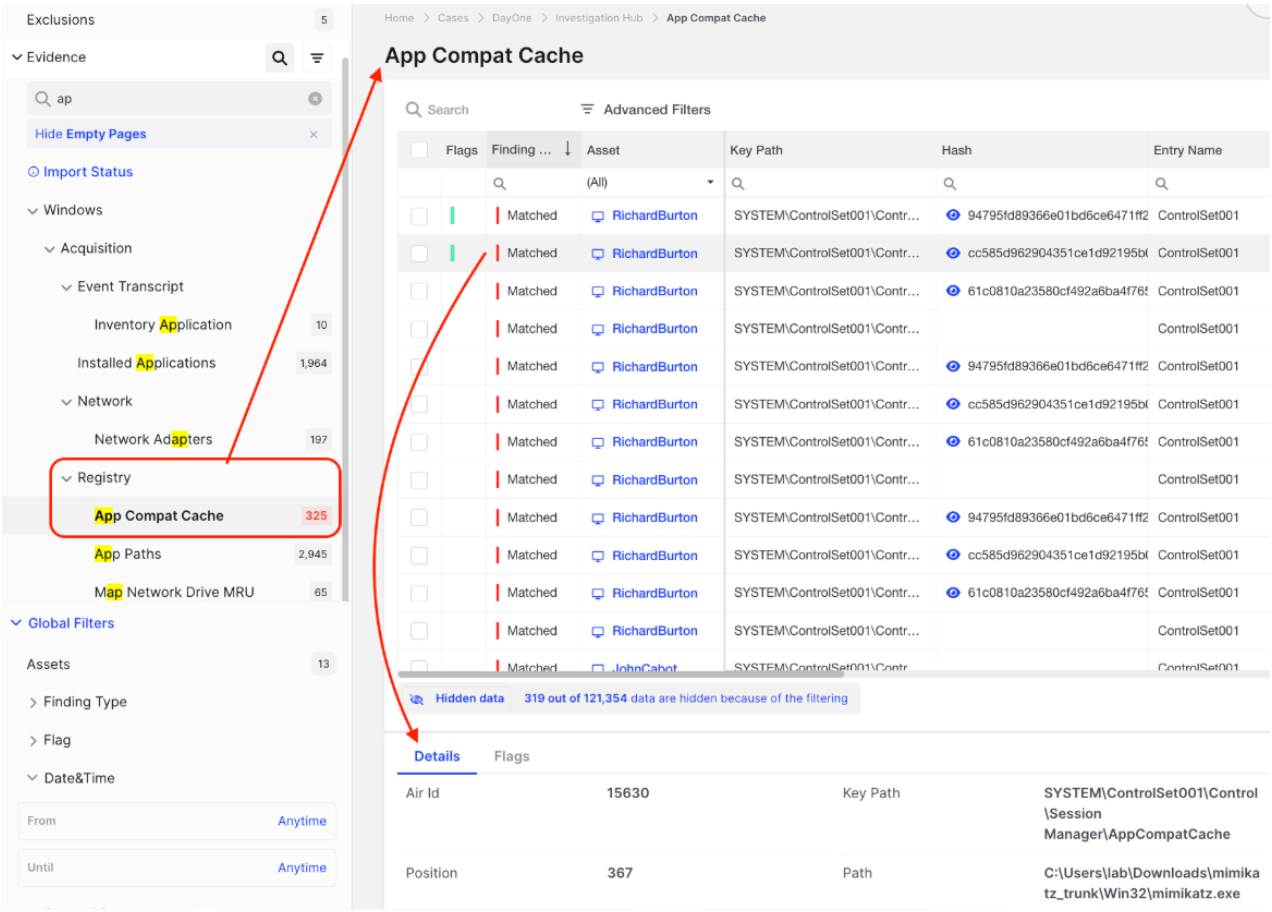
5. The Findings and Evidence Results Tables

Within the Secondary Menu, Findings are categorized and displayed alongside a count for each evidence type where applicable. The severity of each Finding is color-coded for quick identification:

- **Red**: Indicates High severity
- **Orange**: Denotes Medium severity
- **Blue**: Represents Low severity
- **Pink**: Signifies Matched (keyword hit)

This system allows for an immediate visual assessment of the evidence's priority level.

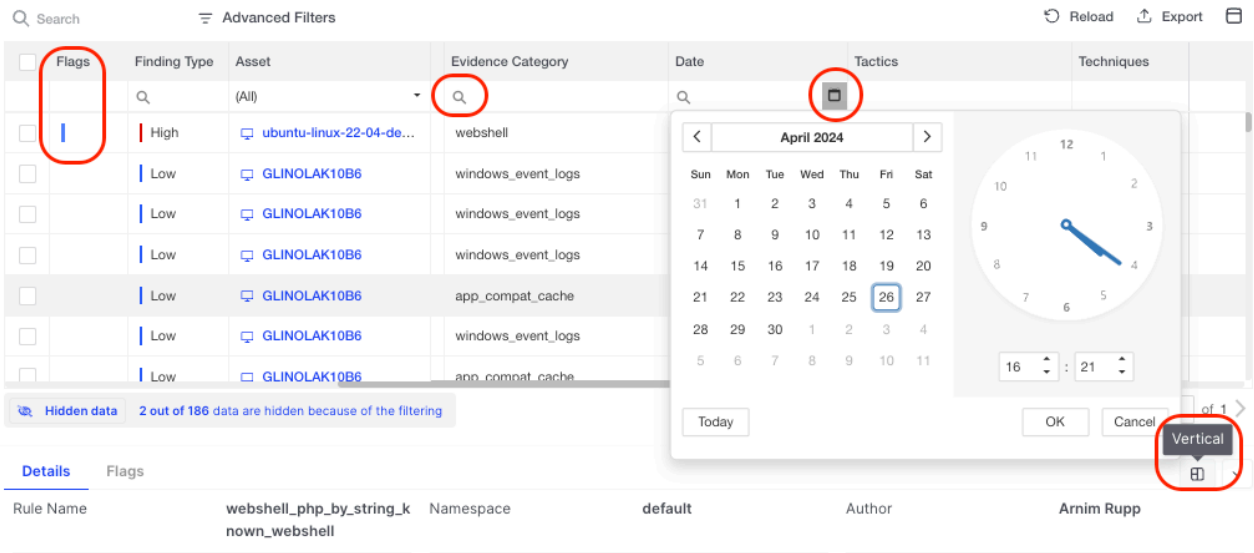
Under the Evidence subheading section, the secondary menu will display all evidence items collected as part of a task, even if a null return has occurred. In the example below, we can see that the **Hide Empty Pages** filter has been activated.



Using the Investigation Hub: Viewing App Compat Cache

In the screenshot shown above, the acquired evidence from the App Compat Cache has been highlighted for focused viewing. As a result, the main viewing area is exclusively dedicated to displaying all the evidence gathered from the App Compat Cache. In this particular example, one 'Matched' item has been singled out, and its comprehensive details are presented in the section below in the 'Details' tab.

It's essential to note that in this scenario, the table has been sorted by the findings column to order the view by the severity of the findings. Additionally, users have the flexibility to rearrange and resize columns according to their preferences. Furthermore, users can customize which columns are visible in the table using the **Column Chooser** feature, which is available on all pages with tables within the Investigation Hub.

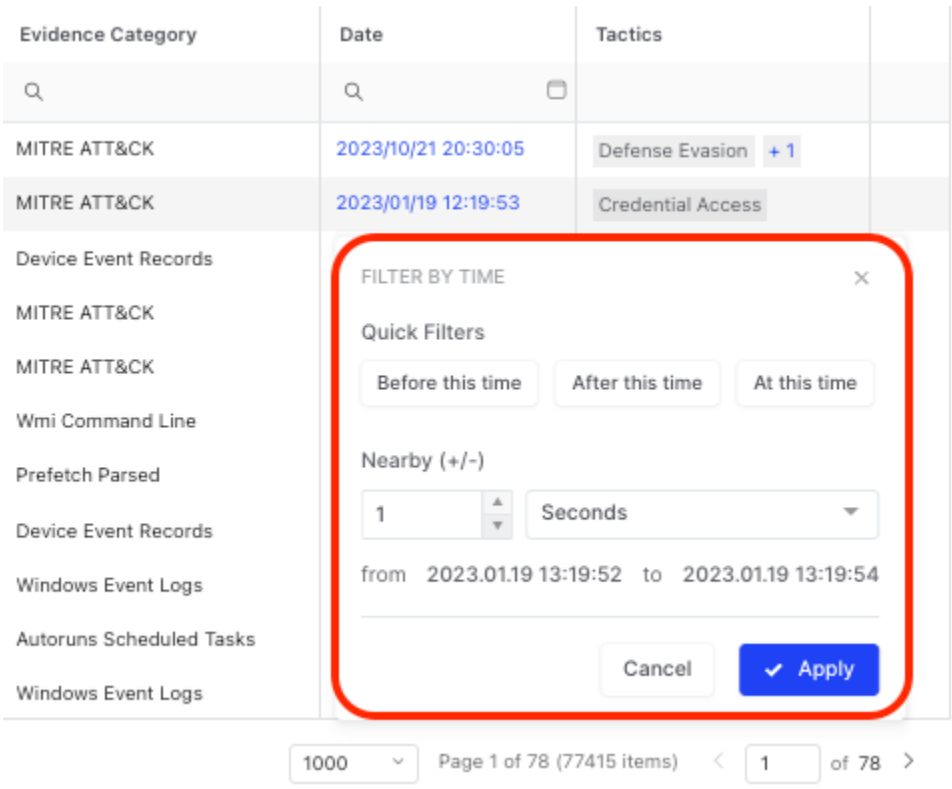


Using the Investigation Hub: Navigation

The Evidence section displays both triage results and imported evidence items, offering comprehensive insight into gathered data.

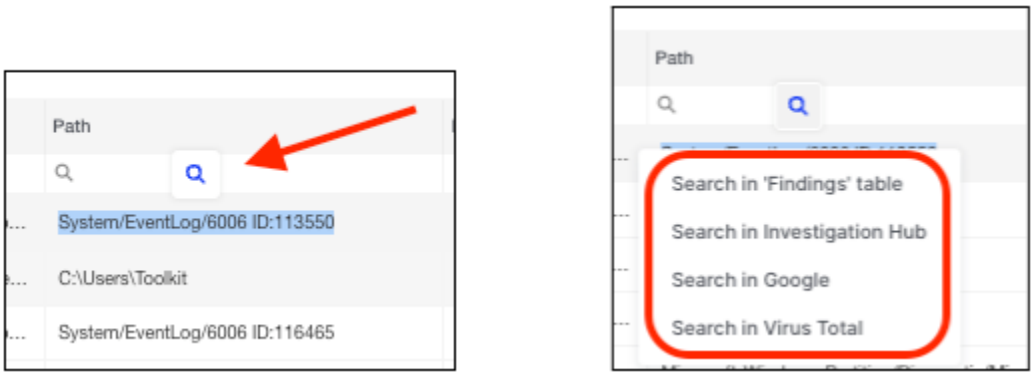
Highlighted in the above screenshot, additional table functionality enhances user interaction and data management:

- **Flag Column**: This column displays flagged items and allows filtering and sorting by flag status.
- **Magnifying Glass Icon**: Offers extensive filtering and searching capabilities with options such as Contains, Does Not Contain, Starts With, Ends With, Equals, and more, allowing precise control over the displayed data.
- **Dockable Details Pane**: Users can choose to dock this pane on the right side or at the bottom of the window. Clicking the icon toggles the Details pane's orientation from vertical to horizontal and vice versa, adapting to user preference and screen layout.
- **Filtered Items Reminder**: A helpful reminder of the currently applied filters, ensuring users are aware of the viewing context.
- **Date & Time Picker**: Allows users to narrow down evidence to specific time periods, facilitating focused analysis.
- **Relative time filtering:** In addition to absolute date and time selection, the Investigation Hub supports relative time filtering. Analysts can quickly apply dynamic ranges, such as “Last 5 minutes,” “Last 7 days,” or “Next 30 minutes,” by clicking on (blue) timestamps within tables. This flexible, context-aware filtering is especially useful for timeline analysis and reviewing live evidence.



Using the Investigation Hub: Relative time filtering

- **Created By Column:** In the Findings table, the Created By column indicates whether DRONE or a user generated the finding.
- **Highlight to search: This allows users to highlight text in the Investigation Hub tables, as shown below, right-click on the selection, select the magnifying glass that appears, and then choose to "Search in Findings table, Search in the Investigation Hub, Search in Google, or Search in Virus Total.**

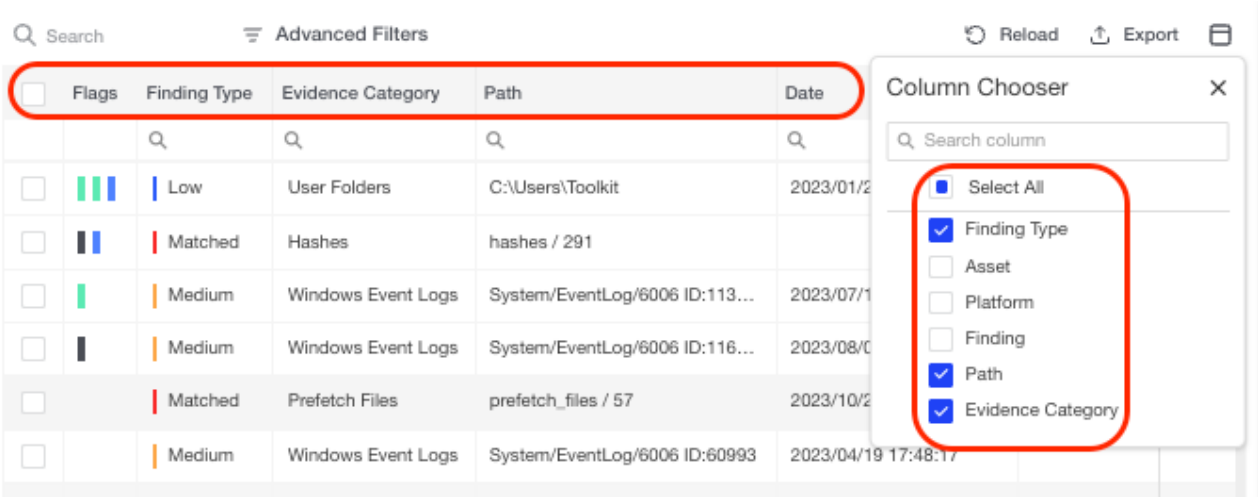


Using the Investigation Hub: Right-click on selected data in the table to be presented with search options

Fullscreen Evidence Tables: Users can view evidence tables in full-screen mode, ideal for large datasets with multiple columns and rows. This feature maximizes screen space, allowing easier navigation and analysis of complex data without the need for scrolling or resizing. To exit full-screen mode, simply click the Exit Fullscreen icon.

Using the Investigation Hub: Fullscreen Evidence Table Icon

Sticky Column Headings: The selection and position of columns will remain saved in your browser across all XDR Forensics sessions unless you clear your browser cookies. This ensures that your preferred layout and data organization remain consistent, enhancing both efficiency and the user experience.

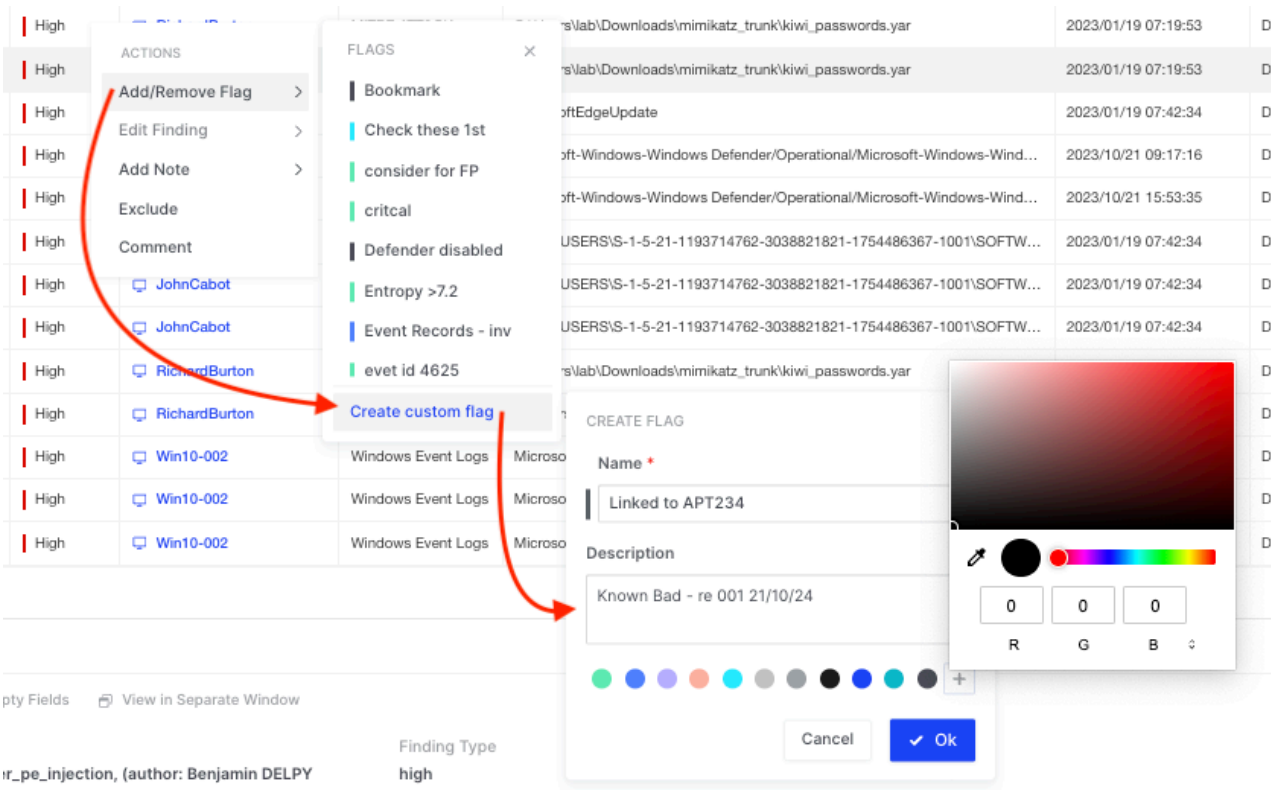


Using the Investigation Hub: **Sticky Column Headings & Column Chooser**

Flags

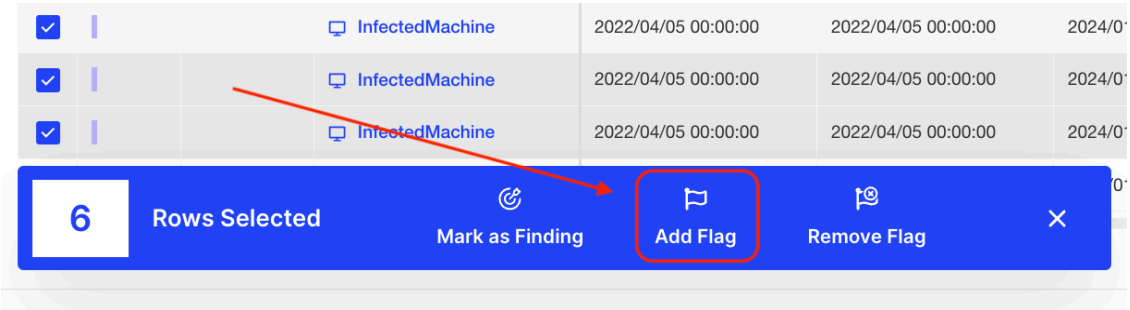
The Investigation Hub flagging feature allows users to create custom flags to mark evidence and findings they deem significant during an investigation. This flagging functionality can be used to filter by flags and facilitates collaboration with other investigators, helping to mark items for re-examination or highlighting important details for potential inclusion in reports.

Users create custom flags by right-clicking on a finding or evidence item, selecting ‘Add/Remove Flag’, and then making a name, description, and color for the flag from the 11 default options, or clicking the '+' to select your own colour from a palette:



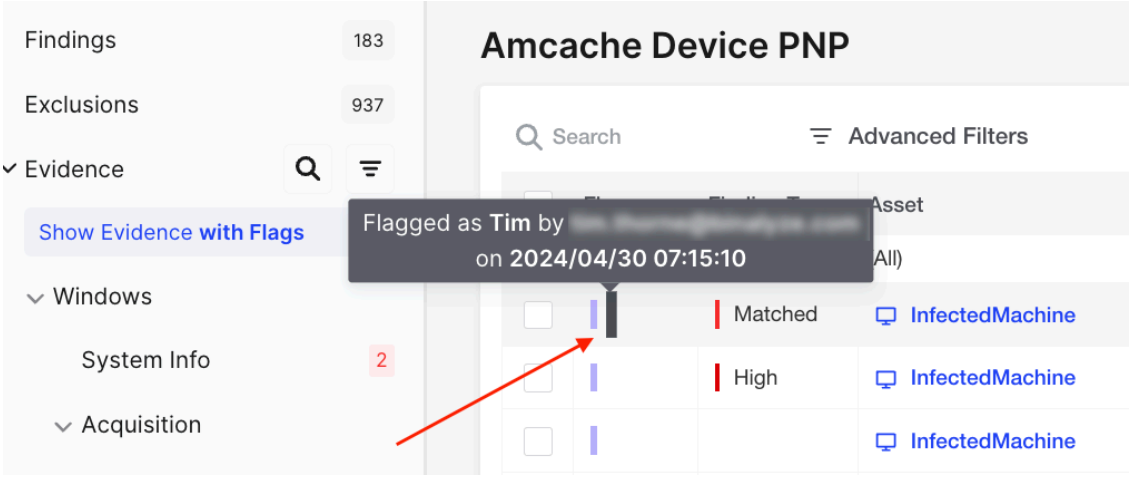
Using the Investigation Hub: Adding a flag to a Finding with a color from the palette

Multiple items can be flagged simultaneously using the Bulk Actions bar:



Using the Investigation Hub: Flagging via the Bulk Action bar

- Hovering over the flag of a flagged item in the table view will reveal:
  - The name of the flag
  - Who created the flag
  - The date & time it was created



Using the Investigation Hub: Hovering over a flag to reveal details

- Flags are saved at the Organization level in Libraries; therefore, they will be available to all cases created in the same Organization.
- Creating new flags or editing existing flags can be done in Libraries if the user has Case Management privileges.
- The **Bookmark Flag** is the only fixed/permanent flag.
- Users can use the advanced filter to include or exclude flagged items in the Investigation Hub table views, enhancing the ability to focus on prioritized or highlighted evidence.

Exclusions

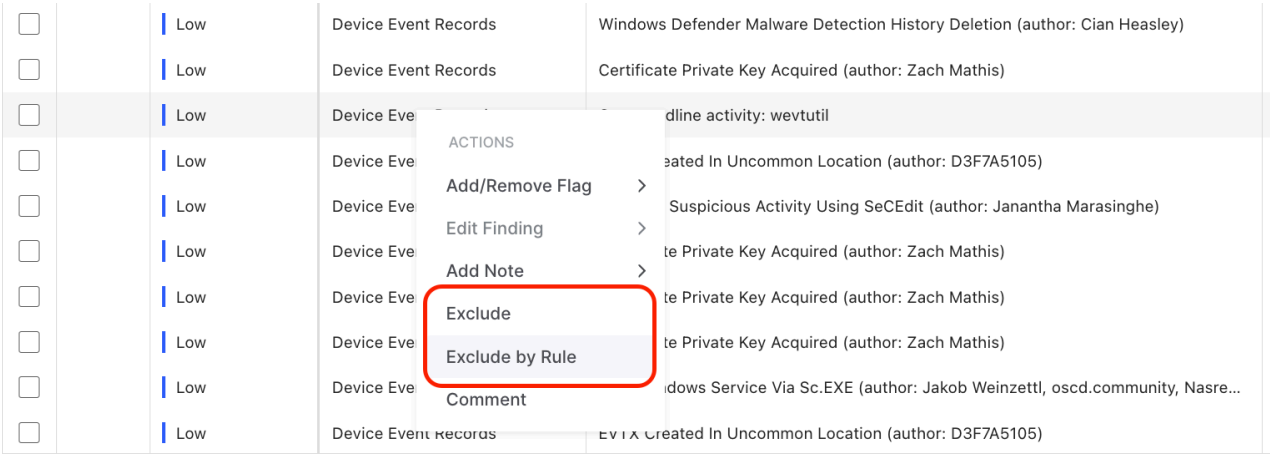
XDR Forensics's DRONE capability enhances decision support by leveraging built-in YARA, Sigma, and osquery rules to quickly identify compromised assets. By analyzing evidence, DRONE generates Findings that highlight key investigative opportunities, helping prioritize and streamline investigations efficiently.

During an investigation, users may encounter Findings that are not relevant to their case. The Investigation Hub allows them to exclude these Findings, helping investigators stay focused on pertinent information.

Exclusion Methods:

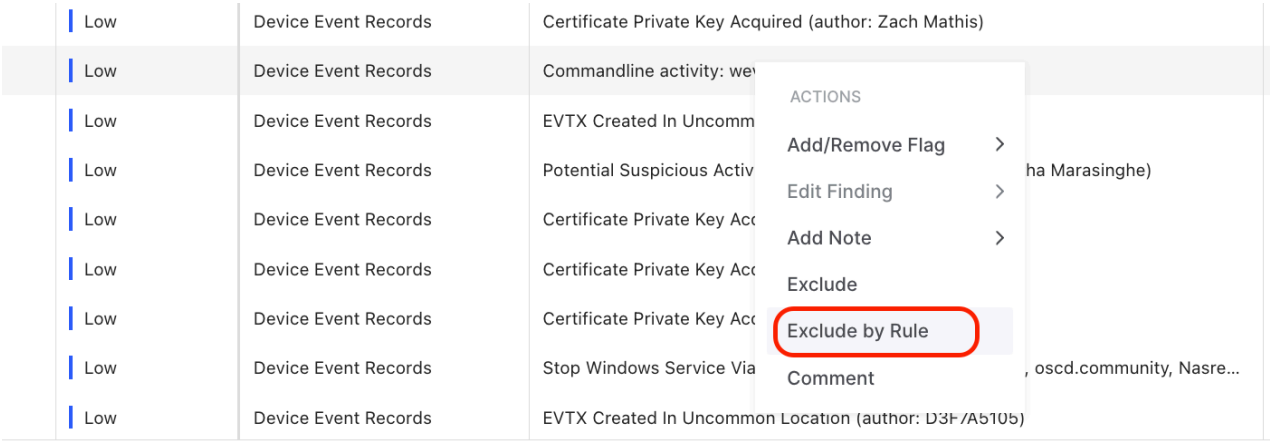
XDR Forensics offers **two exclusion methods**:

1. **Exclusion Rules** – Allows exclusion of a Finding **either** by its location/path, **OR,** regardless of its location, based on the finding itself.
2. **Individual Exclusions** – This method allows users to manually exclude any Findings **within a case**, based on their specific needs or investigation requirements, **without the need to create a rule**.



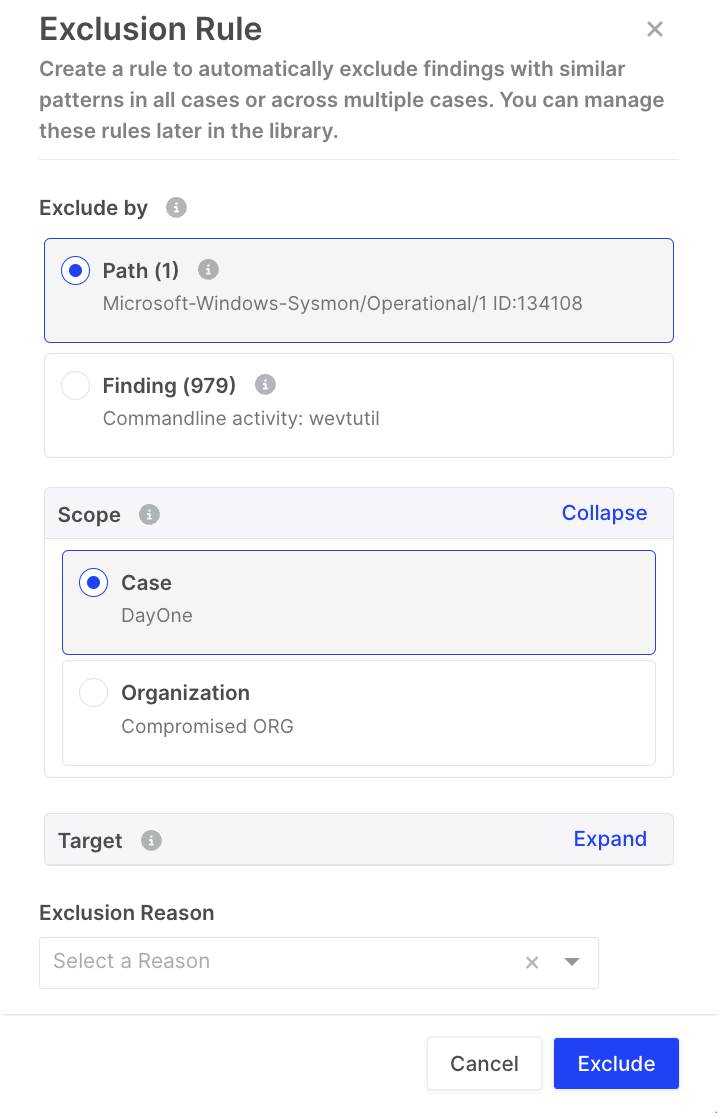
Using the Investigation Hub: Right-click on a Finding to reveal Exclusions

1. Exclude by Rule



Using the Investigation Hub: The Exclude by Rule option

- Right-click on a Finding and choose "Exclude by Rule" from the context menu.



Using the Investigation Hub: Exclusion Rule

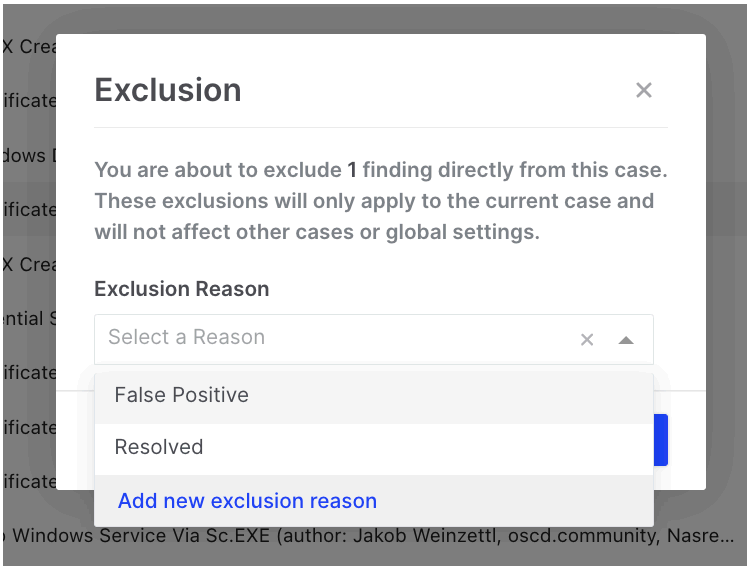
- **Exclude by:** Customize the exclusion based on the path in two ways: **On the specific path**, which excludes an item only if found in that particular location, or based **on the finding**, which excludes the item regardless of where it is found on the asset.
- **Scope: Decide whether to exclude just this case or all cases within the organization.**
- **Target:** Apply the rule to the selected asset, or all assets if applicable.
- (Exclusion by hash value is coming soon)

2. Individual Exclusions

This method allows users to manually exclude any Findings **within individual cases**, based on their specific needs or investigation requirements, without the need to create a rule.

Using the Investigation Hub: The Exclude option

Right-click on a Finding and choose "Exclude" from the context menu.

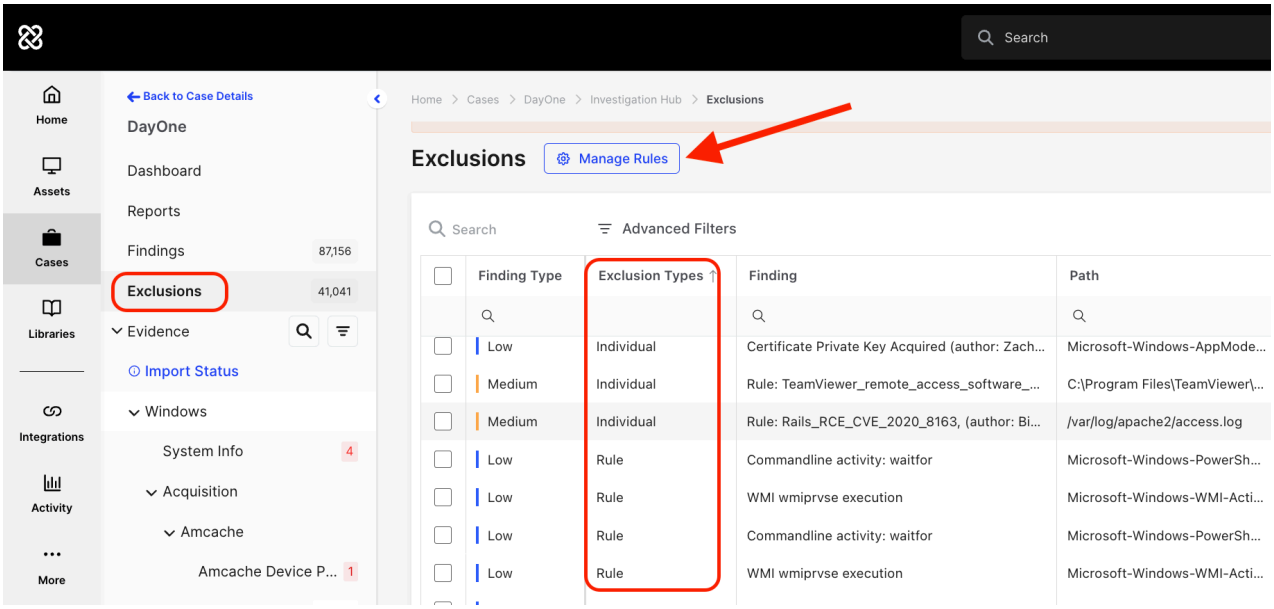


Using the Investigation Hub: Exclusion reason

Users can either select an existing **reason for exclusion** or create a new one. Any newly added reason will be saved and included in the list of available exclusion reasons for future use.

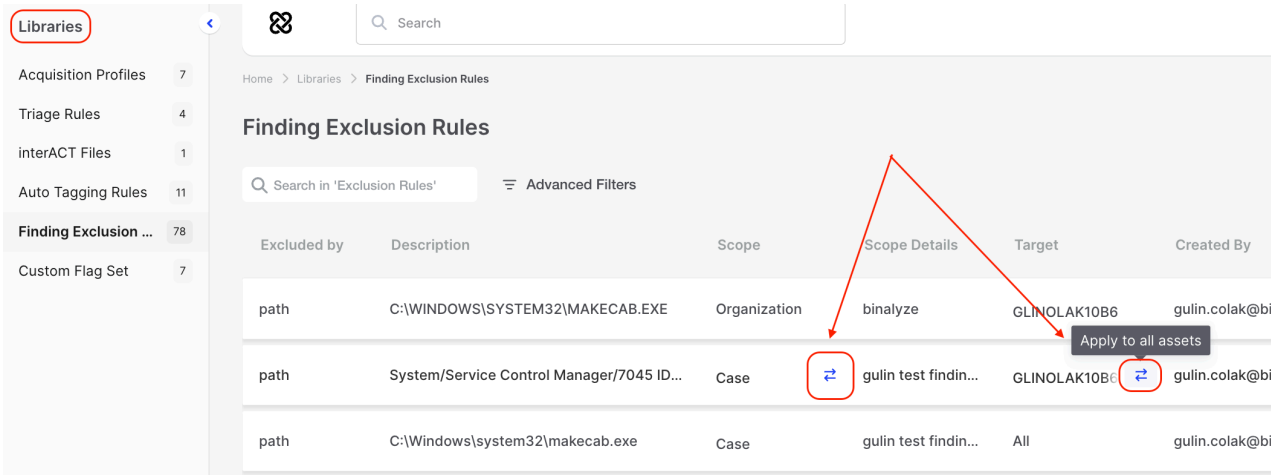
Management and Audibility:

- All excluded Findings are removed from the Findings table and added to the **Exclusions table** (directly below Findings in the Secondary Menu) as shown below. This feature enables teams to cross-validate and resolve discrepancies as needed.



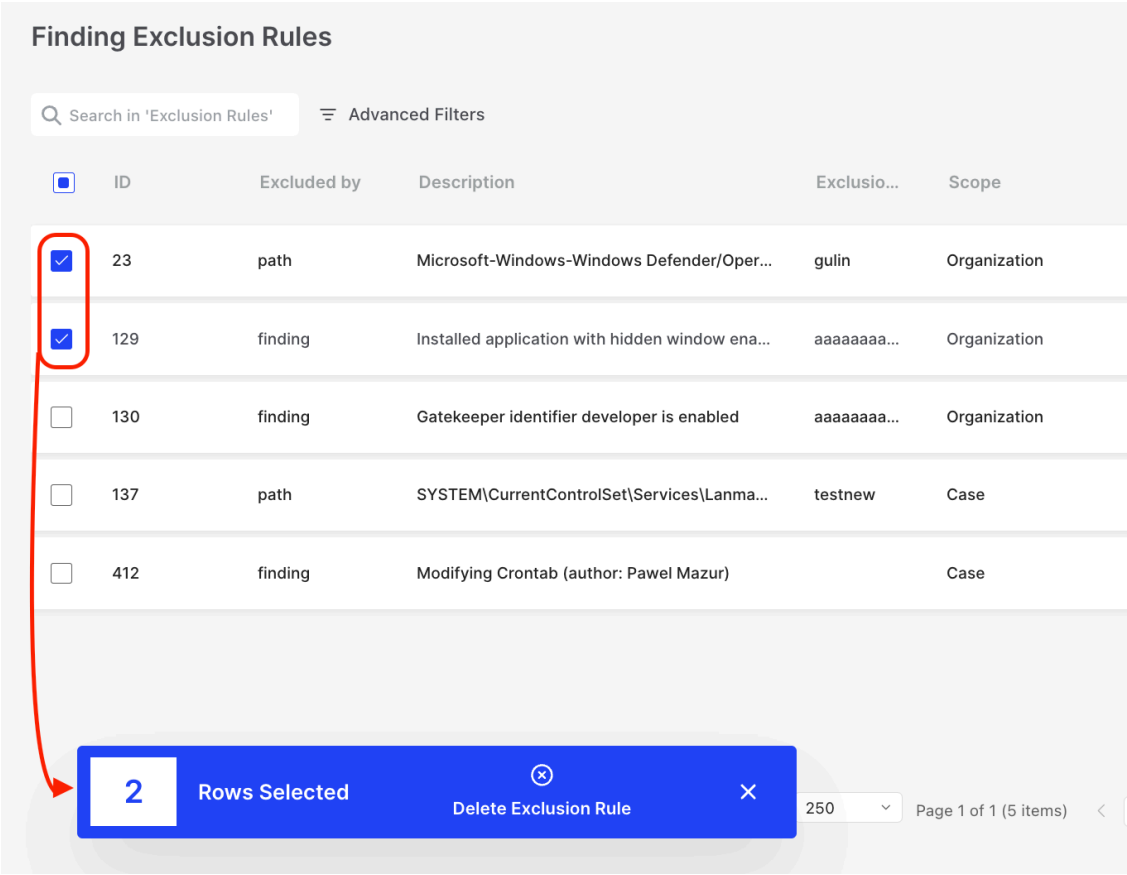
Using the Investigation Hub: Manage Rules

- At the top of the page, users can click **"Manage Rules"** to modify or delete **Exclusion Rules**. However, individual exclusions **cannot be edited**—they can only be deleted. This allows users to refine exclusion settings while maintaining control over how findings are managed.



Using the Investigation Hub: Finding Exclusion Rules

- In the **Organization Library** under "**Finding Exclusions Rules**," users can change the scope from Case to Organization, apply exclusions from one asset to all assets in the organization, or completely delete rules if they have "Case Management" privileges.
- In the **Organization Library,** users can also select Exclusion Rules to delete via the Bulk Action Bar as shown below.

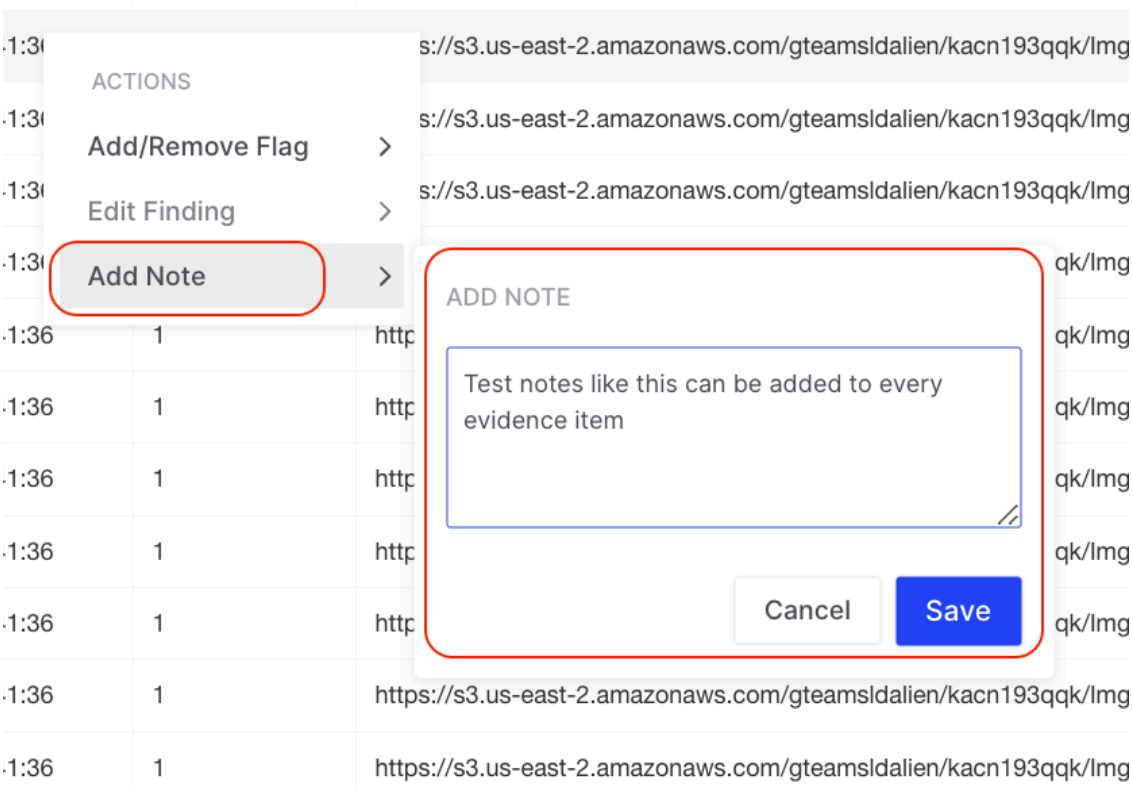


Using the Investigation Hub: Bulk Exclusions

Tooltips are provided throughout to guide users in utilizing these customizable options effectively. After an exclusion is applied, users have a brief opportunity to undo the operation or close the notification, ensuring that actions are deliberate and retrievable.

This Exclusions Capability significantly streamlines the investigative process, allowing investigators to maintain focus on essential evidence while managing irrelevant data efficiently.

Notes



Using the Investigation Hub: Notes

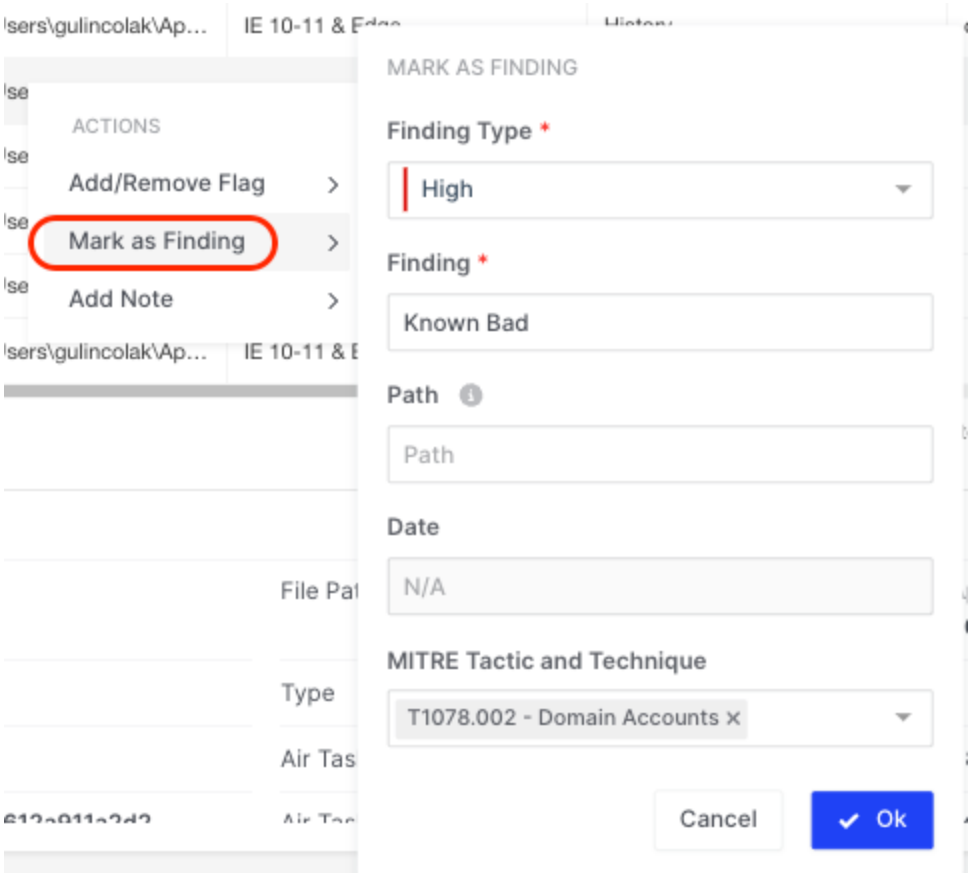
The **Notes** feature enhances collaboration by allowing users to attach notes to any evidence item or Finding, without requiring the item to be bookmarked. These notes are accessible to the entire team working on the case and are displayed in the Notes column of the table view.

This feature is handy for enhancing communication within the team, as it allows for the sharing of insights and observations directly alongside the relevant evidence. Furthermore, notes can be seamlessly integrated into reports through XDR Forensics's "Generate Report" feature, which supports the creation of customizable reports. Additionally, notes can be exported for external use.

Overall, the Notes feature in XDR Forensics 4.13 streamlines case documentation and enhances the reporting process, making it an essential tool for collaborative investigations.

User-generated Findings

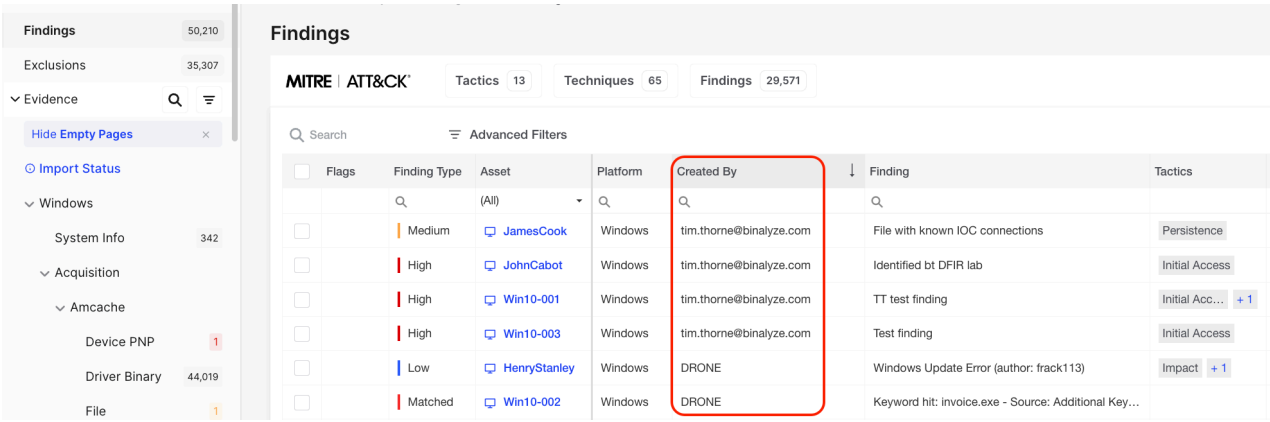
In previous versions of XDR Forensics, the decision-support system, including the automated evidence analyzers known as DRONE, helped users prioritize their investigative steps. With the release of XDR Forensics 4.13, users now have the capability to generate their own Findings, enhancing the depth of investigations by incorporating personal insights, organizational context, or specific investigative details.



Using the Investigation Hub: Mark as Finding

For instance, from the Evidence page, users can right-click on an item and select "Mark as Finding" to manually allocate a Finding to it. This process involves several steps:

- **Selecting a Finding Type**: Users choose from pre-defined types such as High, Medium, Low, or Matched.
- **Adding a Description/Label**: This provides a clear and concise description or label for the Finding, facilitating clarity and reference.
- **Detailing the Path and Associating with MITRE ATT&CK TTP**: Users can specify the file path and link the Finding to a specific tactic or technique from the MITRE ATT&CK framework.
- **Setting a Date**: Users can also include the date when the Finding was identified or marked.

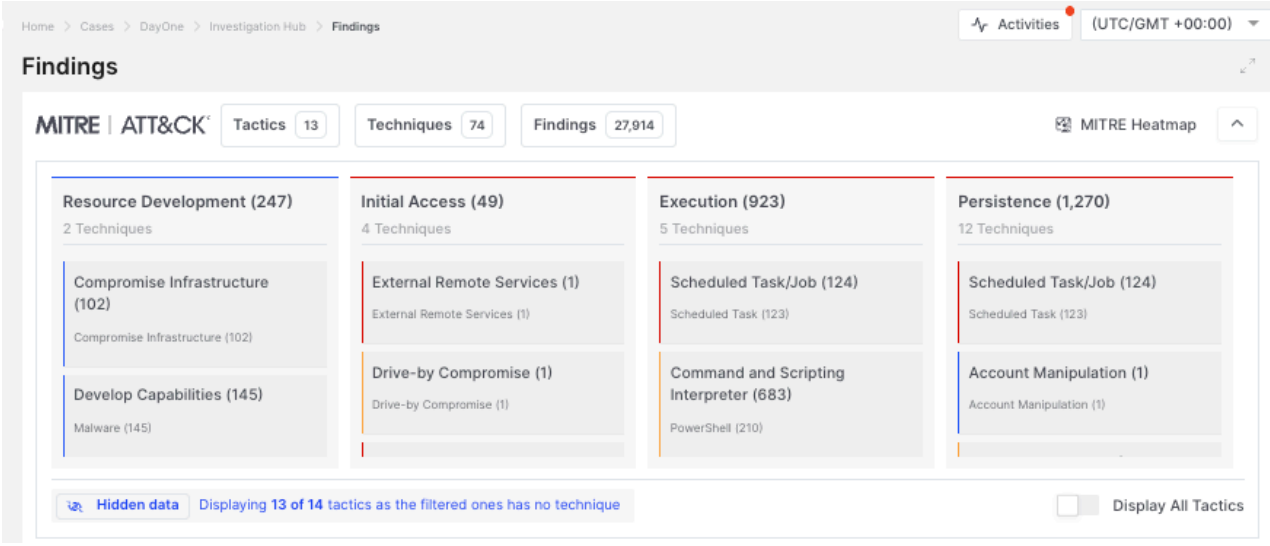


Using the Investigation Hub: Created By

Upon confirming with "OK", the Finding type is displayed in the "Finding Type" column within the Evidence page, along with a column indicating who created the Finding. These user-generated Findings are then added to the "Findings" window, where they are distinguishable by a "created by" column, which shows whether a Finding is from DRONE or user-generated, and identifies the creator. This allows for efficient tracking and searching using the column search functionality.

Additionally, these Findings are reflected in the Dashboard widget, where users can filter to view "All" findings or specifically search across DRONE or user-generated Findings using the provided tabs. This new feature in XDR Forensics 4.13 significantly empowers users to tailor their investigative processes with enriched data and personalized analysis.

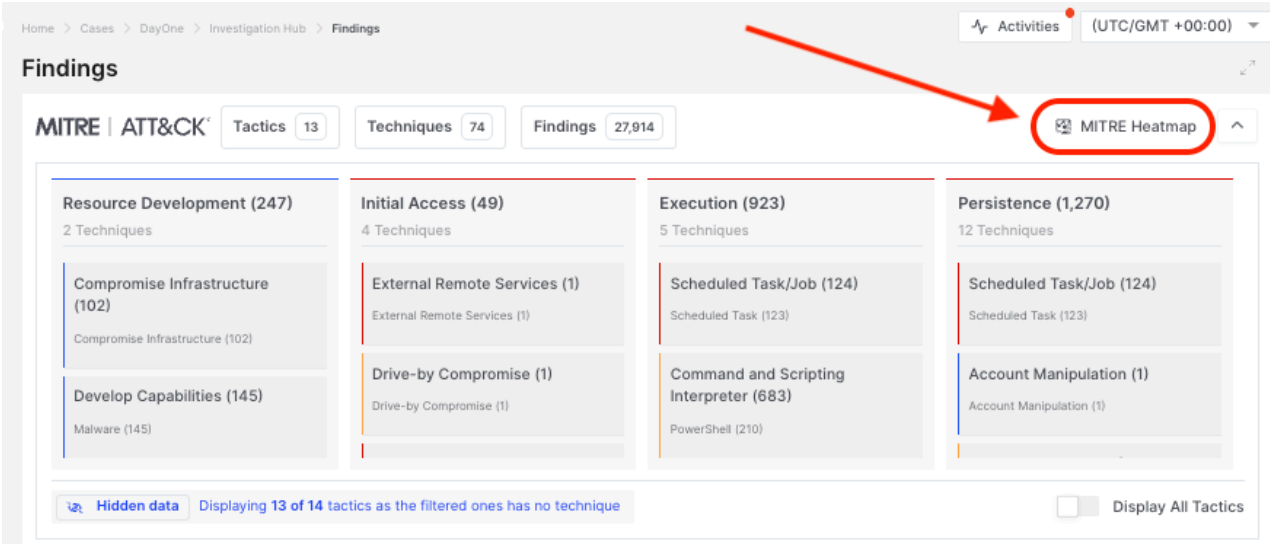
6. MITRE ATT&CK



Using the Investigation Hub: MITRE ATT&CK mapping

MITRE ATT&CK serves as a global resource for adversary tactics and techniques, guiding threat models and methodologies across industries. Integrated with XDR Forensics, it continuously maps findings to ATT&CK, enhancing detection with up-to-date YARA rules for IoCs and TTPs. DRONE's implementation scans assets and processes using crafted rules, with automated rule updates in XDR Forensics. You can read more about DRONE in this blog, '**Automated Compromise Assessment with DRONE'** ↗.

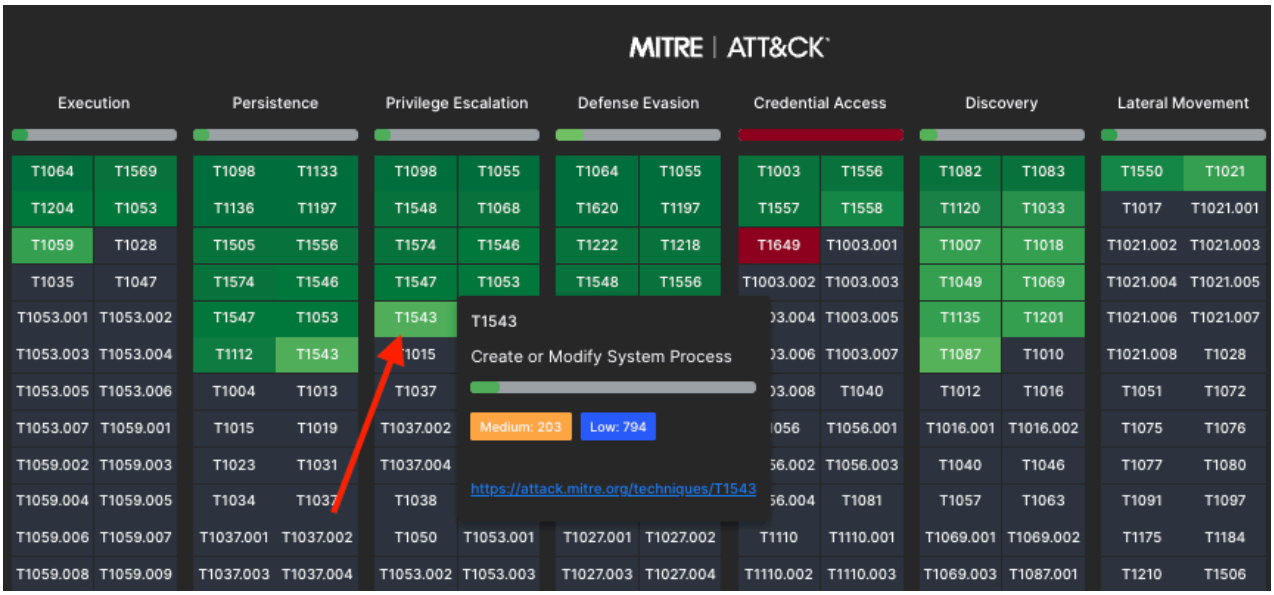
MITRE Heatmap



Using the Investigation Hub: Link to the MITRE Heatmap

The MITRE ATT&CK heatmap is an integrated visual tool within the Findings section of the Investigation Hub that maps DRONE-generated findings to MITRE ATT&CK techniques and sub-techniques. It visually indicates which techniques are involved in the current case, based on the volume of findings, enabling analysts to quickly assess the breadth and focus of an attack.

Each technique in the heatmap corresponds to a cell, and its color intensity represents the number of findings mapped to that technique. This allows analysts to see at a glance which areas of the ATT&CK framework are most heavily implicated in the investigation.

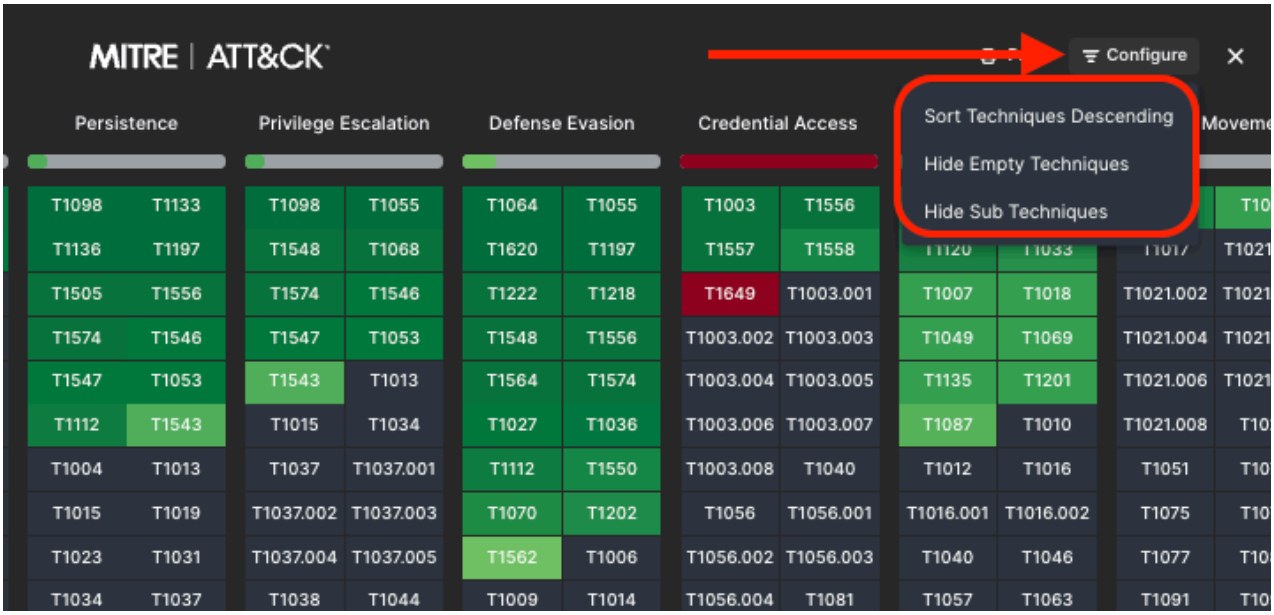


Using the Investigation Hub: The MITRE ATT&CK heatmap

Analytical Value of Heatmaps

- **Immediate Threat Context**:
  The heatmap adds actionable context by connecting raw findings with attacker behaviors defined in the MITRE ATT&CK framework. This helps analysts:
  - Understand the “how” of the attack (techniques)
  - Spot patterns across multiple assets
  - Recognize potential blind spots in detection coverage
- **Gap Identification**:
  By visualizing unused or sparsely populated techniques, SOC teams can identify **gaps in evidence collection, detection logic, or attacker behaviors that have not yet been fully uncovered**.
- **Investigation Prioritization**:
  Supports strategic triage by letting analysts focus first on high-volume techniques associated with high-severity findings.

Configurable Visualization Options



Using the Investigation Hub: Configure the Heatmap via this button

To enhance usability and relevance, the heatmap offers several customization controls:

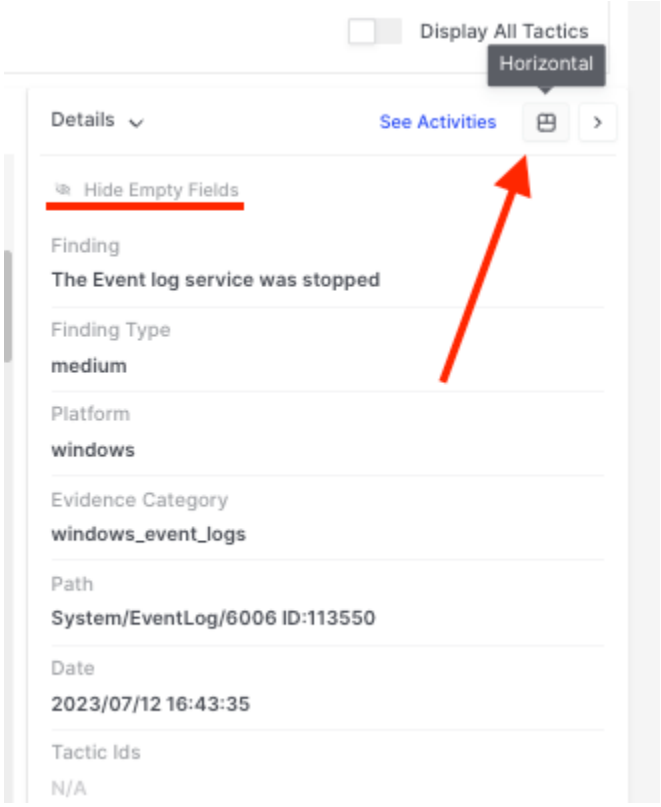
- **Sort Techniques by Volume**:
  - Techniques can be sorted in either ascending or descending order by associated findings.
  - Helps in prioritizing analysis — either by focusing on the most prevalent threats first or identifying less common, potentially stealthier techniques.
- **Toggle Empty Techniques**:
  - Analysts can **hide or display techniques with zero associated findings**, reducing visual noise and emphasizing active threat vectors.
- **Toggle Sub-techniques**:
  - Enables a granular view by revealing or hiding sub-techniques under each parent tactic.
  - Supports more precise mapping of attack behavior and facilitates deeper root cause analysis.

These options are beneficial when navigating investigations that involve high asset counts or extensive data volumes, allowing teams to zero in on critical patterns efficiently

7. Details View

To match your viewing preferences and monitor setup, the Details view for selected evidence items is both **dockable and detachable**. You can either open a separate, independent window to display the evidence details or use the icon shown below to toggle the position of the Details window.

Dockable Evidence Details Window: This feature allows you to toggle the Details window between a vertical display on the right side of the browser window and a horizontal display at the bottom. Simply click the icon to switch between these views.



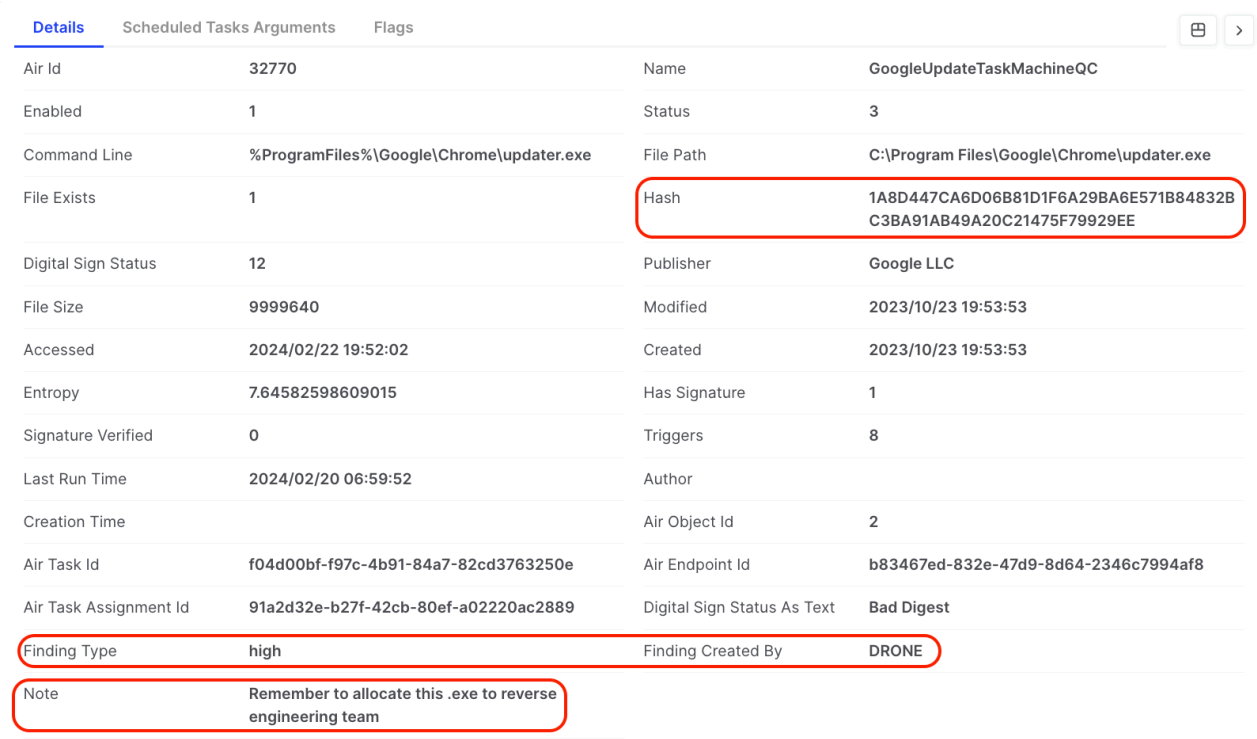
Using the Investigation Hub: Hide Empty Fields and the Dockable icon

In the Details window, there may be occasions when some fields are empty. To minimize clutter from these empty fields, you can use the '**Hide Empty Fields**' feature. This option allows you to clean up the interface by displaying only those fields that contain information.

Detachable Evidence Details Window: This feature allows users to open evidence details in a standalone window that can be resized and repositioned anywhere on the screen(s) for improved clarity. The window maintains its form even when displaying new evidence items. Clicking on a new row in the table updates the detached details view to match the newly selected item. This flexibility allows investigators to compare multiple pieces of evidence side by side, improving the overall analysis process.

Using the Investigation Hub: Detachable Details Window

Details View Example:



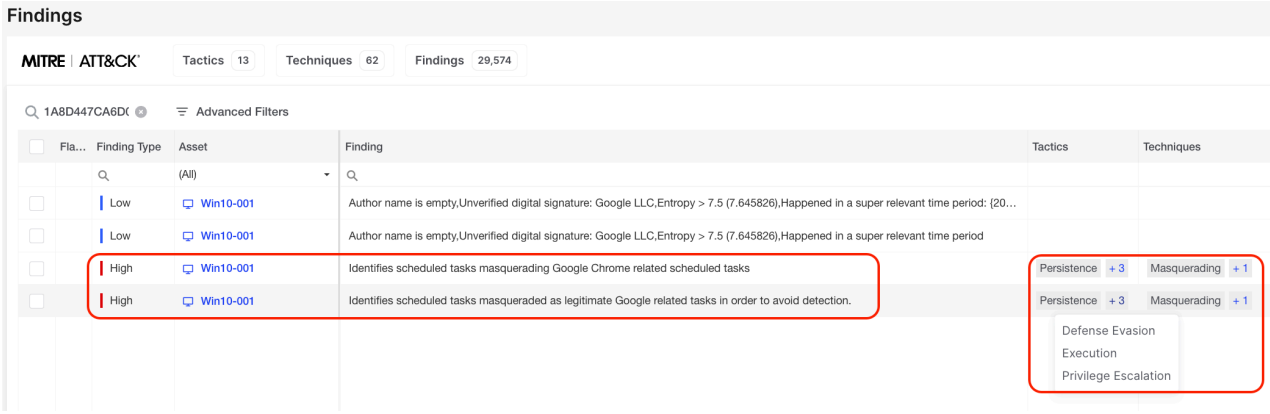
Using the Investigation Hub: Details View

The screenshot above displays an example of a typical details view of a file selected in the table view. This particular file is a Scheduled Task entry which has a DRONE Finding Type 'High'.

The term "Digital Sign Status As Text" refers to the description of the status of a digital signature in text form. When you encounter "Bad Digest" as the status, it indicates an issue with the digital signature of a file or document.

Specifically, "Bad Digest" means that the hash value calculated from the downloaded or retrieved file does not match the hash value used initially when signing the document or file. This discrepancy suggests that the file may have been altered or corrupted after it was signed. Consequently, the integrity of the file is in question, and it can no longer be trusted as authentic or unmodified from its signed state. This status is a critical indicator in digital security practices, especially when verifying the legitimacy and integrity of software downloads and updates.

Searching the hash value revealed in the details window across the Investigation Hub immediately reveals other Findings associated with this file:

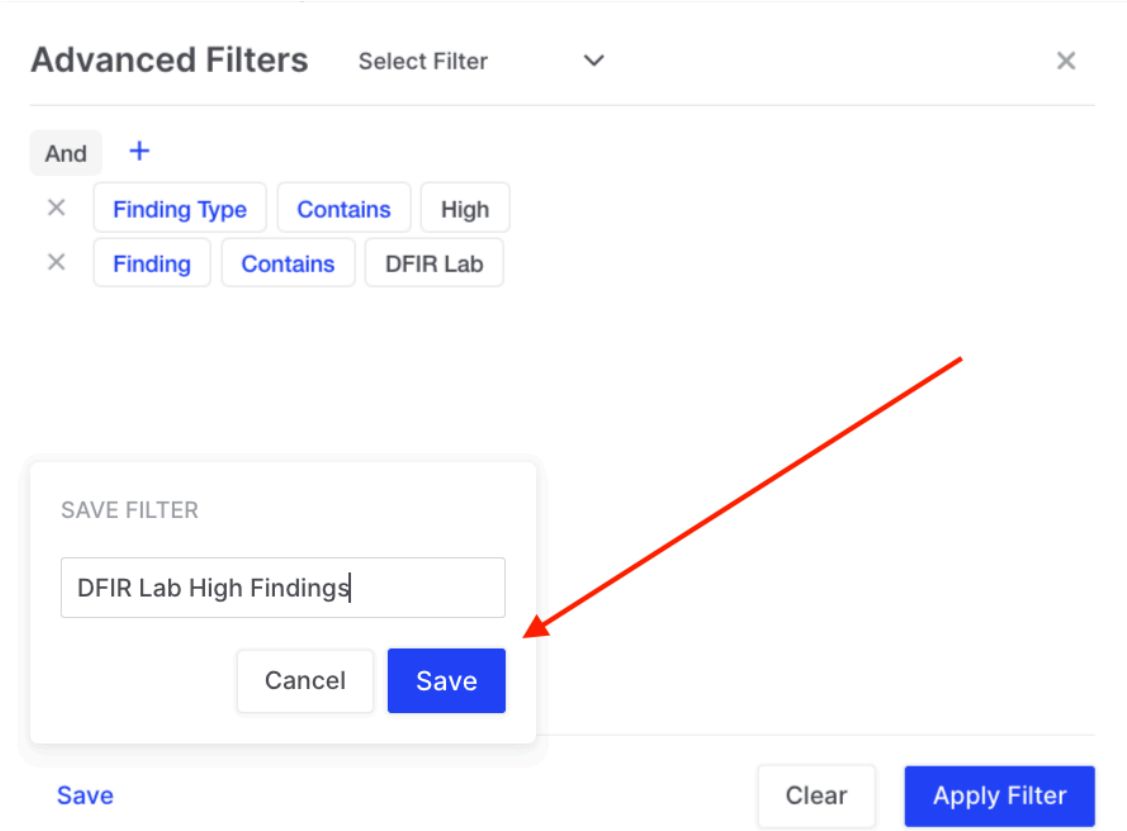


Using the Investigation Hub: Mapped Findings

Now we can see that there are two High Findings that are mapped to MITRE ATT&CK Tactics and Techniques, as shown.

8. Advanced Filters

The advanced filter save feature boosts efficiency by enabling users to save and share custom filters within a XDR Forensics organization. This functionality streamlines data analysis, promotes consistency, and enhances collaboration throughout the investigative process.

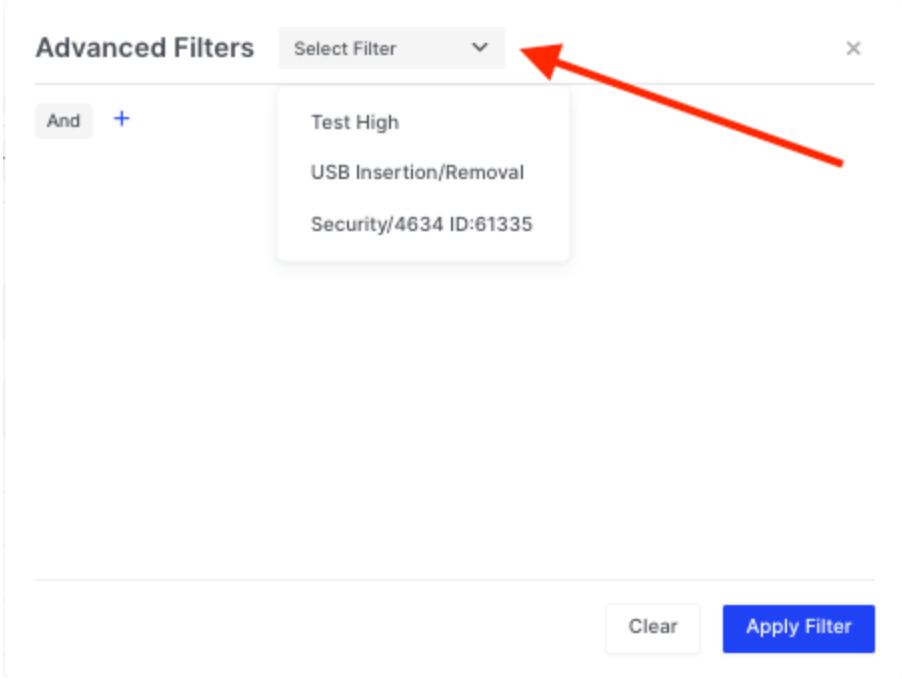


Using the Investigation Hub: Using the Advanced Filters

The Advanced Filter window remains visible as you build the filter, and you can reposition it.

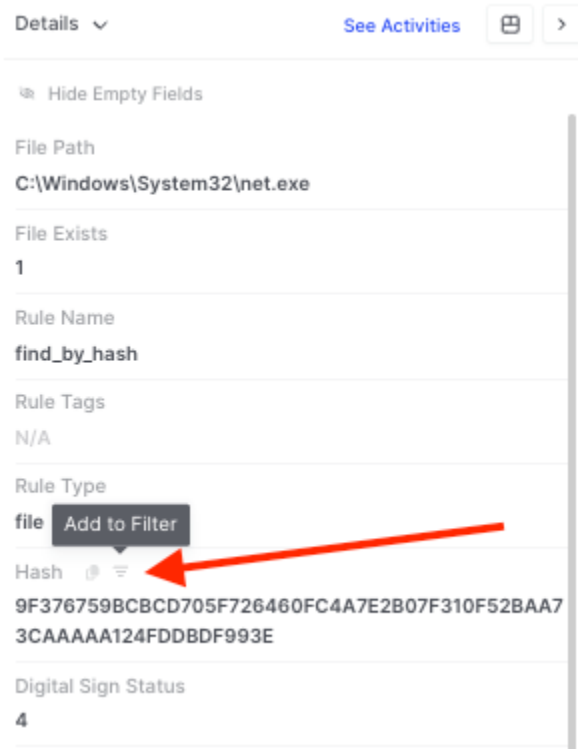
Each Advanced Filter is specific to the table in which it is built; for example, an Advanced Filter you build in Findings will not be available to you in the Browser Artifact table.

Filters can be saved and then later selected from the drop-down list.



Using the Investigation Hub: Accessing saved filters

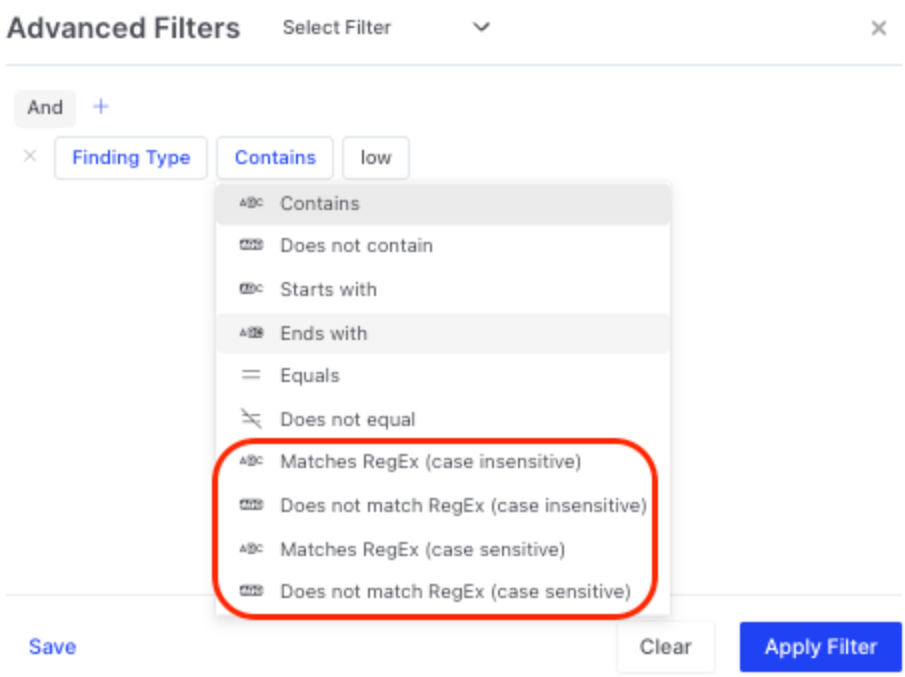
Add items to Advanced Filters directly from the Details window using the filter icon:



Using the Investigation Hub: Add items directly

Regex Operator Support in Advanced Filters

The Investigation Hub includes support for regular expression (regex) operators within the Advanced Filters section. This enables more powerful and flexible data filtering across cases, triage results, and evidence views.



Using the Investigation Hub: Regex support in Advanced Filters

How It Works: Regex filtering is available via the existing Contains filter dropdown, with the following operator options:

- Doesn't match RegEx (case sensitive)
- Matches RegEx (case sensitive)
- Doesn't match RegEx (case insensitive)
- Matches RegEx (case insensitive)

These options enable you to create highly granular search conditions, which are ideal for forensic analysts working with variable or loosely structured data inputs.

Example Use Cases

Locate executables matching a naming convention: `^cmd.*.exe$`

Identify registry keys containing GUIDs: `[A-Fa-f0-9]{8}-([A-Fa-f0-9]{4}-){3}[A-Fa-f0-9]{12}`

Example Use Cases and Syntax Explained:

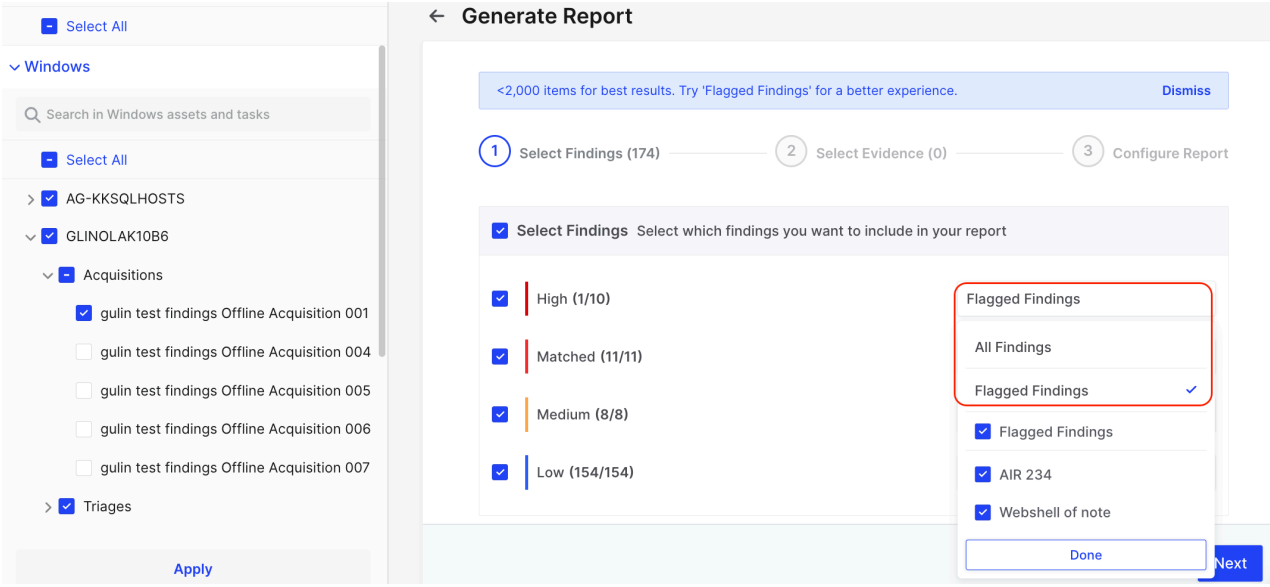
- **Locate executables matching a naming convention**
  Regex: `^cmd.*\.exe$`
  Explanation:
  - `^` anchors the match to the **start of the string**
  - `cmd` looks for the literal text `"cmd"`
  - `.*` matches **any number of any characters** (except newline)
  - `\.exe` matches the literal file extension `.exe` (Note the backslash escapes the dot)
  - `$` anchors the match to the **end of the string**
  - **Use case:** Filters for files like `cmd.exe`, `cmd123.exe`, or `cmd_tool.exe`
- **Identify registry keys containing GUIDs**
  Regex: `[A-Fa-f0-9]{8}-([A-Fa-f0-9]{4}-){3}[A-Fa-f0-9]{12}`
  Explanation:
  - `[A-Fa-f0-9]{8}` matches **8 hexadecimal characters**
  - `-` matches a literal hyphen
  - `([A-Fa-f0-9]{4}-){3}` matches **three groups** of 4 hex characters followed by a hyphen
  - `[A-Fa-f0-9]{12}` matches the final 12 hex characters
  - **Use case:** Matches standard Windows GUIDs like `f81d4fae-7dec-11d0-a765-00a0c91e6bf6`

9. Automatic Report Generation - Wizard

XDR Forensics's automated report generation feature, alongside the Compromise Assessment Report template, efficiently populates reports with relevant investigation information, offering pre-built, customizable sections tailored to different stakeholders and audiences.

Generating a Report: Users can initiate report generation from the Secondary Menu under "Reports" or directly from the Dashboard action button using XDR Forensics's Compromise Assessment report template. The report generation process has been refined to allow for greater customization and flexibility:

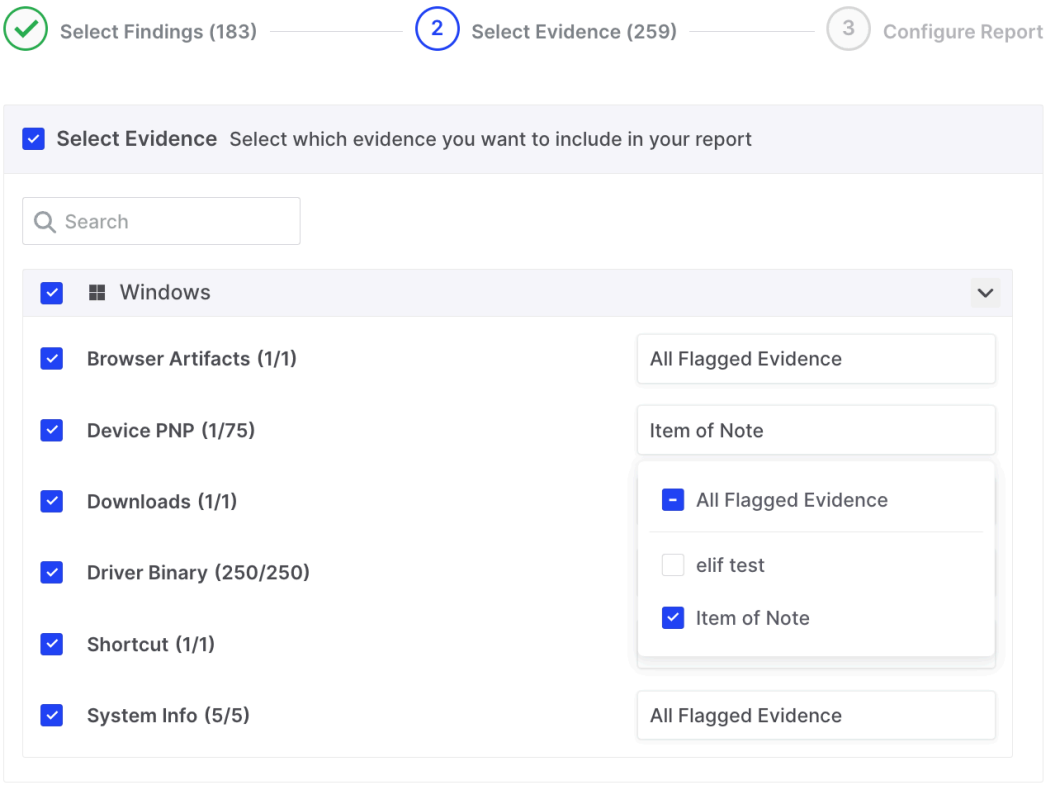
1. **Initial Setup and Customization:**



Using the Investigation Hub: Report setup

- Include all findings from the Investigation Hub, encompassing DRONE findings and user-generated findings, excluding any that have been previously filtered out.
- Filter which assets and associated tasks to include based on relevance to the report.
- Apply additional filters by severity of findings and flags to further tailor the content.

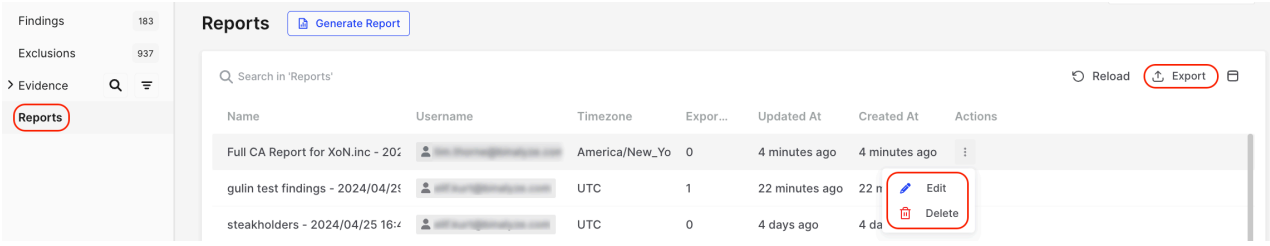
1. **Inclusion of Evidence:**



Using the Investigation Hub: Select Evidence

- Users can choose to incorporate evidence linked to flagged items, enhancing the report with crucial items of note that may not be categorized as findings.

1. **Report Customization:**
    - Add a company logo to the report, allowing service providers to brand their reports for clients.
    - Name the report and set the date/time to ensure clarity and relevance.
    - Select specific sections to include in the report based on the intended audience, ensuring the content is relevant and tailored to their needs.



Using the Investigation Hub: Reports view

Final Steps and Editing: Once the report is generated by clicking <Generate Report>, it remains fully editable within an HTML iframe editor. This flexibility allows analysts and responders to append additional analysis notes, recommendations, and other pertinent information as needed.

Post-Creation Options:

- **Save**: Saves the current HTML version of the report, allowing further modifications in the future.
- **Export PDF**: Converts the report into a PDF file for distribution or archiving.

Both HTML and PDF versions of the report can be managed, edited, generated, exported, and deleted from the "Reports" tab in the Secondary Menu of the Investigation Hub.

This wizard-driven approach not only simplifies the report generation process but also provides users with powerful tools to create detailed, customized reports that meet specific requirements, all within a few clicks.

Off-Network Responder

How to collect evidence from, or run a triage on an off-network asset and then import the results into XDR Forensics?

For assets not connected to your network, XDR Forensics enables the creation of a portable Responder package for running triages and collecting data. This package also facilitates the use of our live MITRE ATT&CK analyzer along with our post-acquisition analyzers, enhancing the depth and relevance of the gathered data.

You can transfer this package to the off-network asset using a shareable link, email, file-sharing services, or by physically taking it to the asset. Once executed on the asset, the collected data is returned to the XDR Forensics console that generated the package for comprehensive analysis and reporting via the Investigation Hub.

When the Responder is executed, it creates an evidence container file with a .zip extension on the offline asset.

> **Up to, but not including XDR Forensics v 4.27.6, all collections were automatically encrypted. The password generated by the Off-Network Responder is not predefined or known in advance; it is created programmatically. As a result, this password is not accessible until the output of your off-network task is uploaded to the XDR Forensics Console.**

With **XDR Forensics v 4.27.6+,** users can choose to encrypt the collection with a password during the off-network Responder setup process. When importing the collection zip file with the additional password back into the console, users will need to enter this password. **Biunzip** ↗, a small utility we created, is specifically helpful in this scenario, aiding in the management of multiple off-network acquisitions during console ingestion.

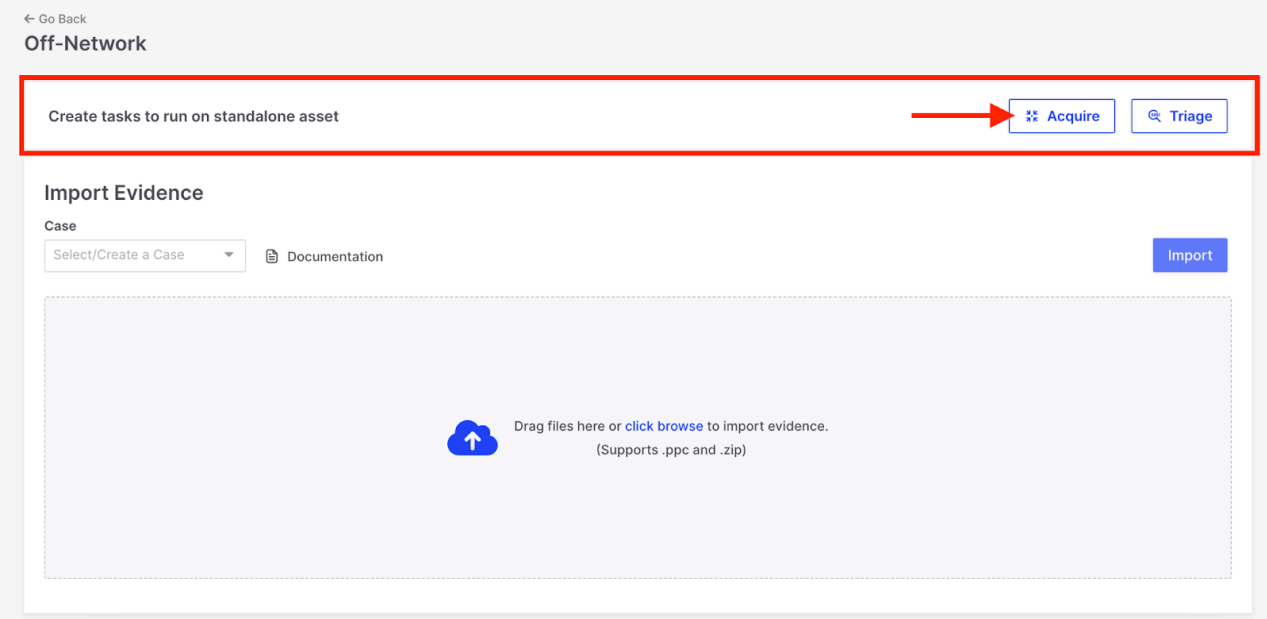
1. Task Creation

Create an Off-Network Task

1. Navigate to the "Assets" tab in the XDR Forensics console and click on the "+Add New" button.
2. Select "Off-Network" to initiate the creation of a task for devices not connected to the network.

Choose the Task Type

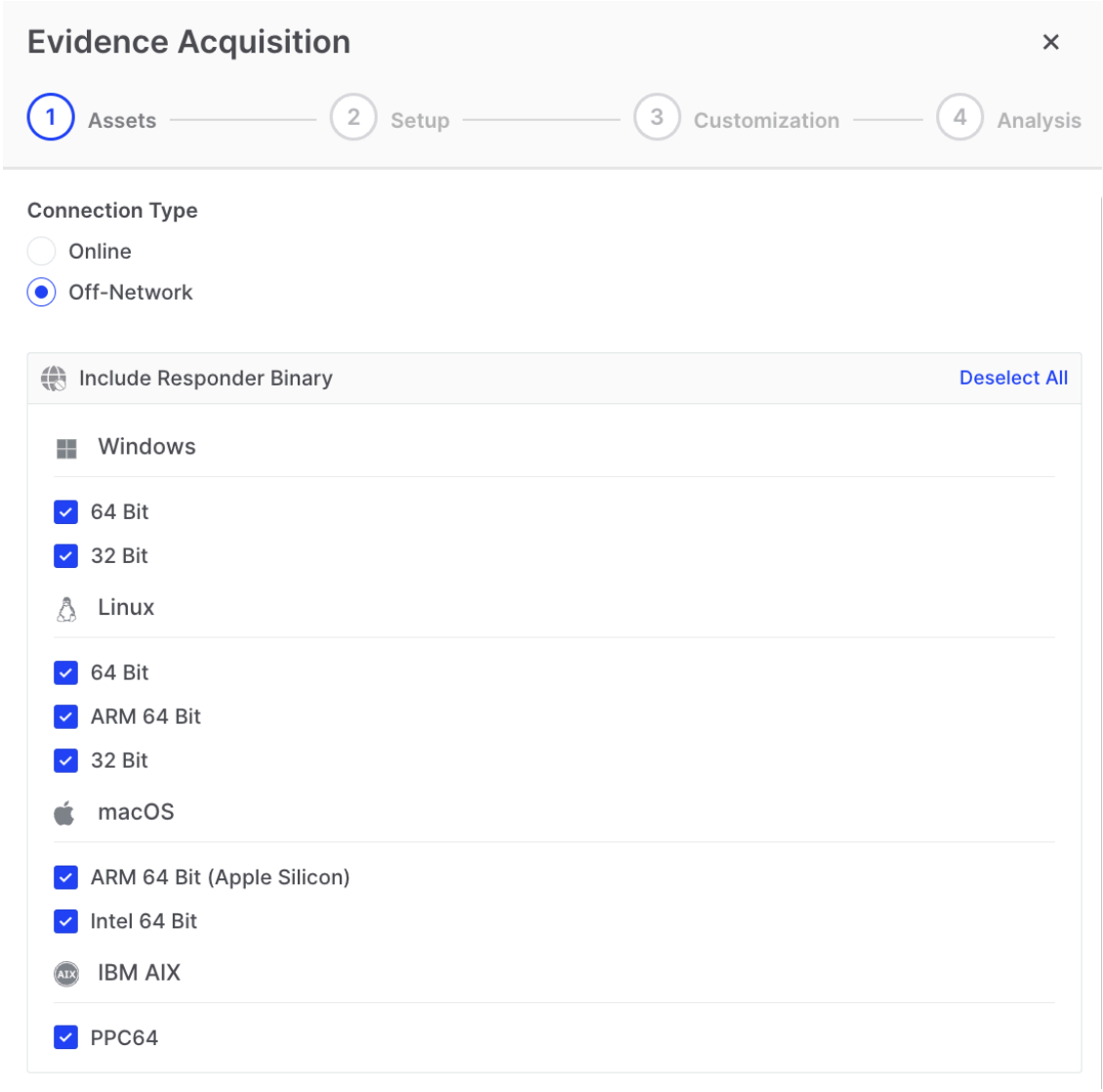
1. In the second stage, select the type of task you want to perform on the Off-Network asset. You can choose between "Acquisition" and "Triage".
2. For this example, let's proceed with the "Acquire" feature.



Off-Network Responder: Create Task

Select the Asset(s) Operating System

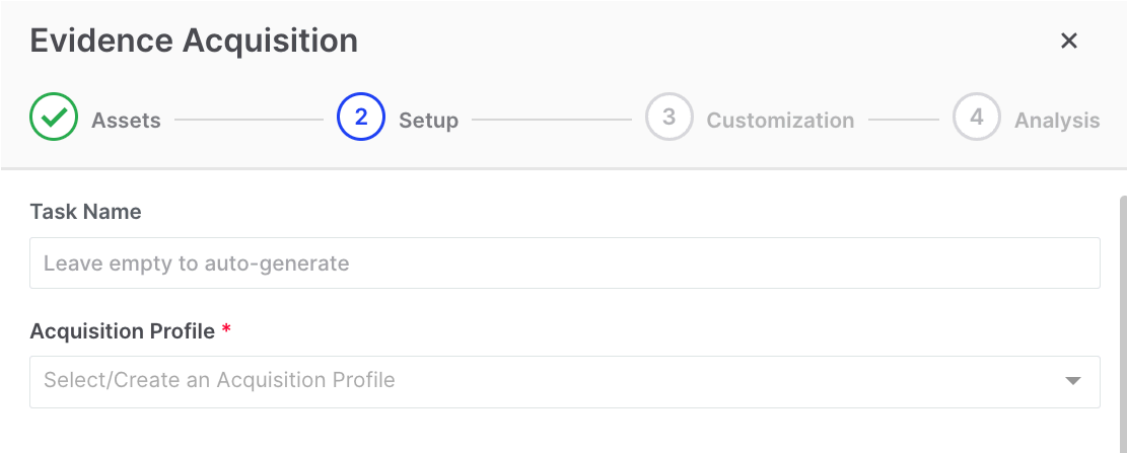
When creating an Off-Network binary, first choose the operating system on which you plan to execute the binary. If you anticipate needing to use the binary on multiple operating systems or are unsure which system will be used, consider generating a package with multiple binaries. This approach ensures that you will have a binary compatible with all the XDR Forensics-supported operating systems.



Off-Network Responder: Assets

Define the Task

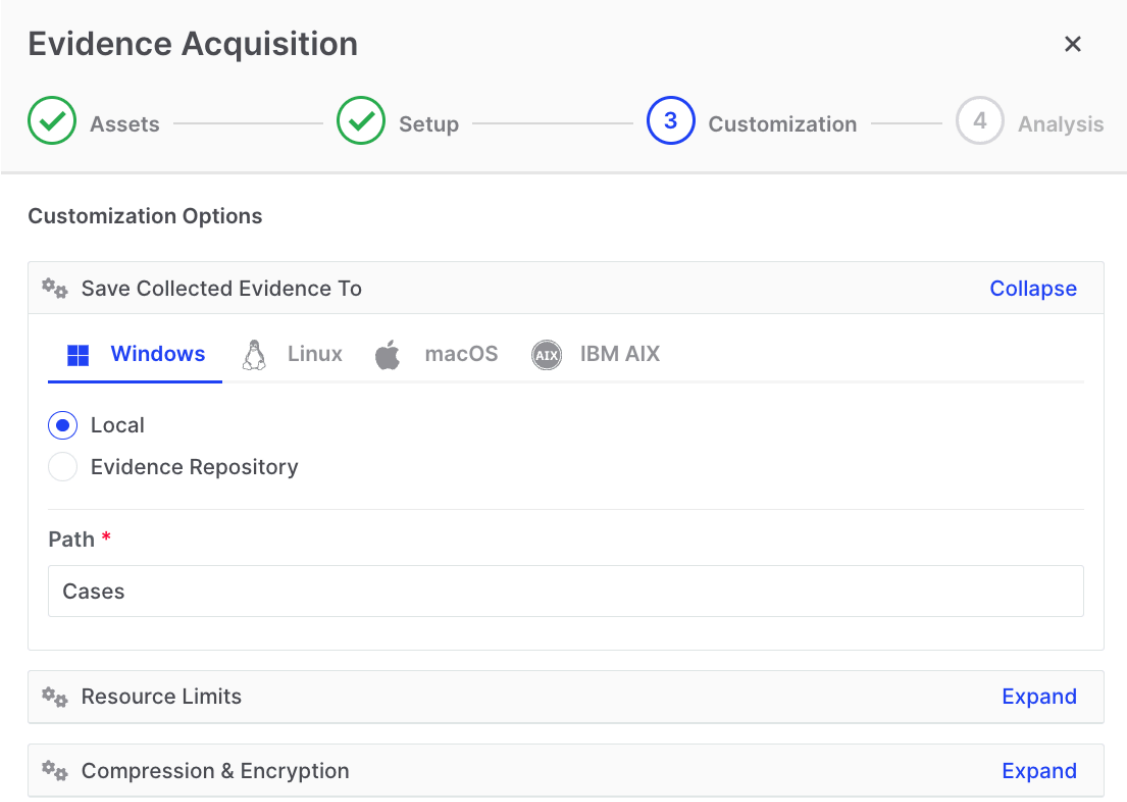
Specify the Task Name (optional) and Acquisition Profile (mandatory).



Off-Network Responder: Task Name

Customize Collection Options

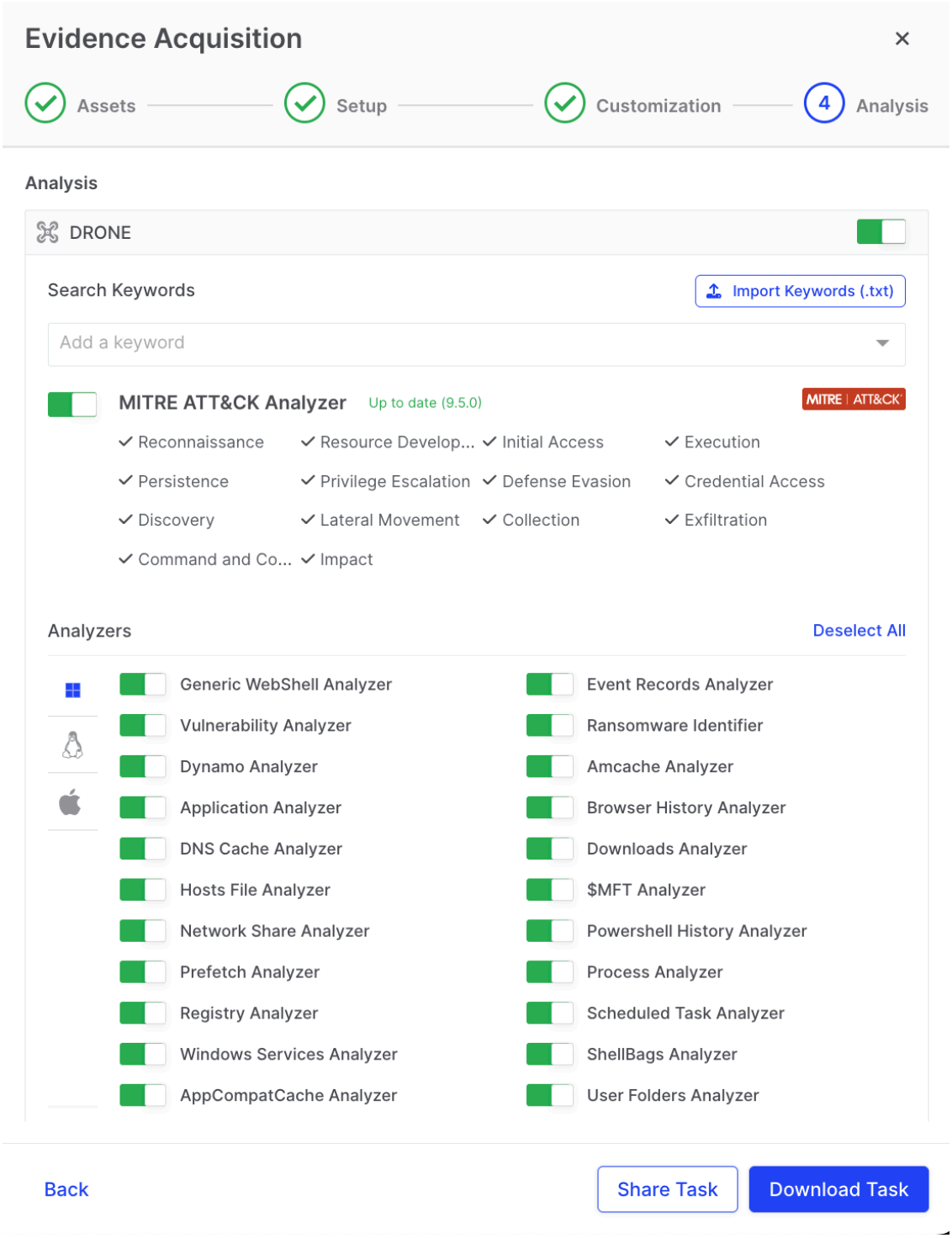
Here, you can choose to customize the task options if you need to deviate from your organizational policy settings.



Off-Network Responder: Setup

Customization - Optional DRONE Analysis

By default, the DRONE feature is enabled for off-network tasks. However, in the subsequent step, you have the option to deactivate the MITRE ATT&CK analyzer, as well as any or all of the post-acquisition analyzers, depending on your specific requirements.

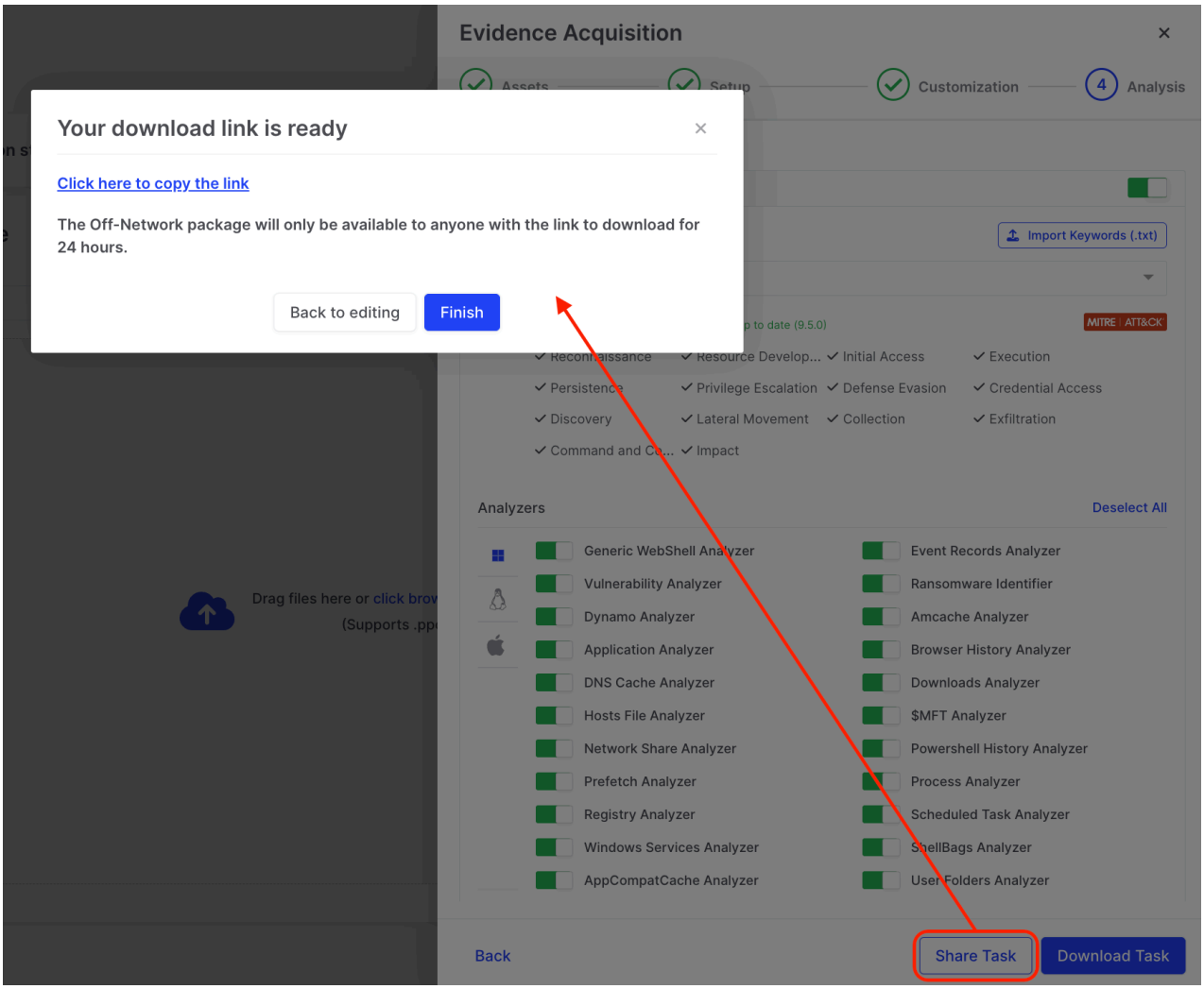


Off-Network Responder: DRONE

Download or Share the Responder Binary

You can now download or share the Responder binary you have just created.

You can access the Responder binary by either downloading it directly to your storage media or by copying the link to share the package.



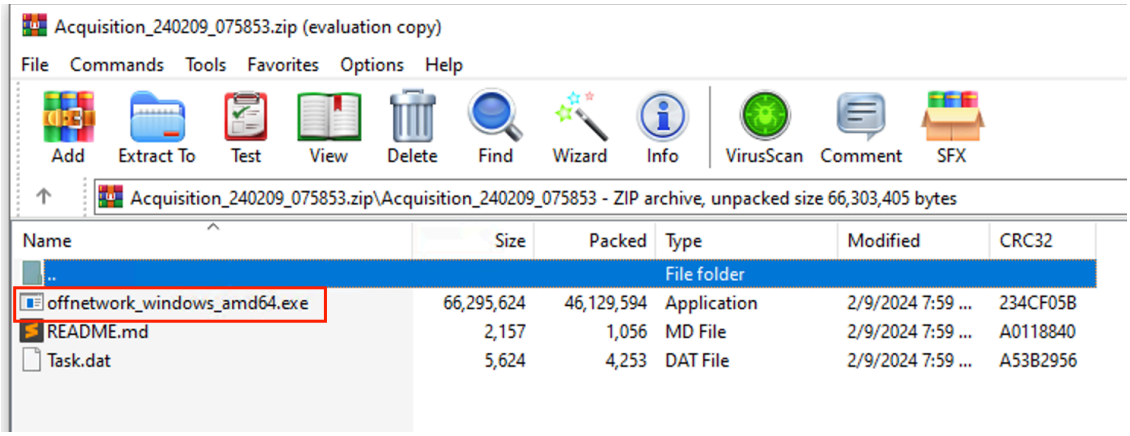
Off-Network Responder: Download Link

2. Task Execution

Execute on the Offline Asset

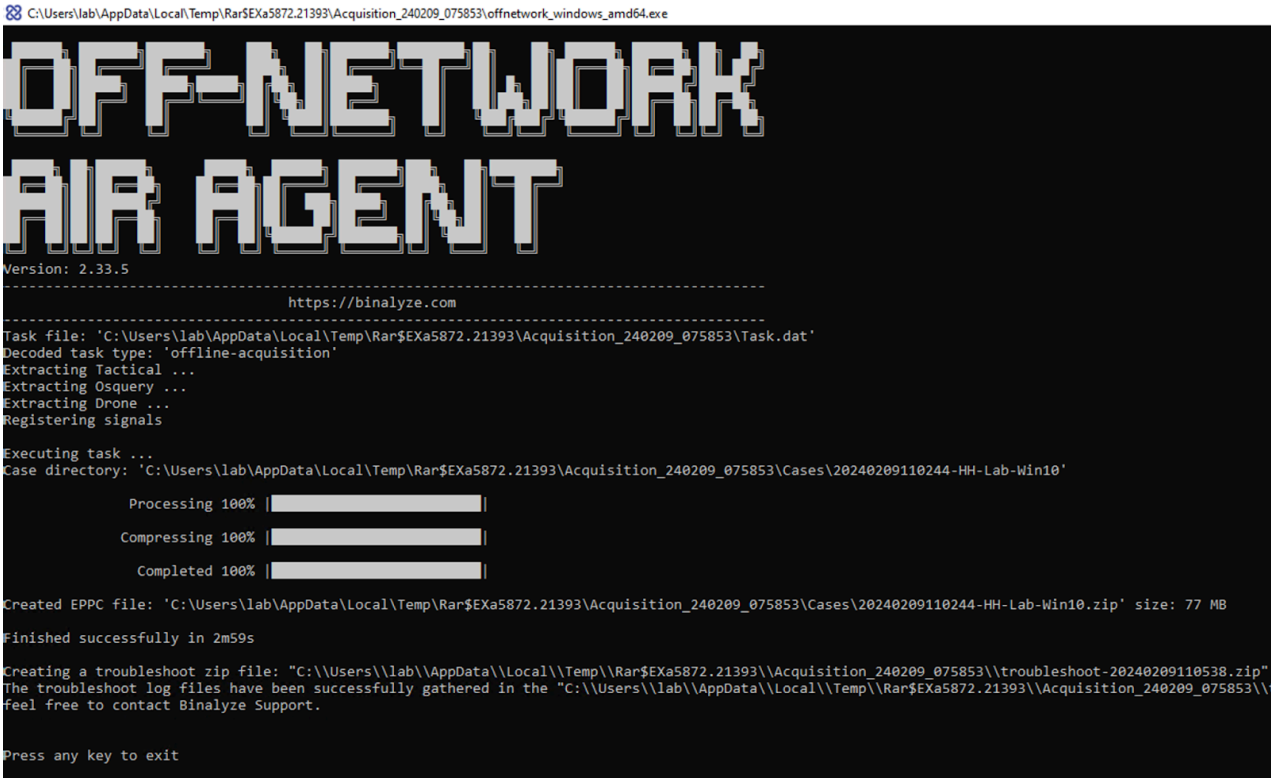
> If you don't want the collection to be saved to the directory from which you launch the binary, please refer to this page to learn how to set a custom directory for your collection or triage results.

Run the downloaded Responder binary on the relevant offline asset. In the example below, we show the downloaded executable file named 'offnetwork_windows_amd64.exe' and the UAC window where the user will need to allow permissions for the XDR Forensics Responder to run.



Off-Network Responder: The downloaded binary

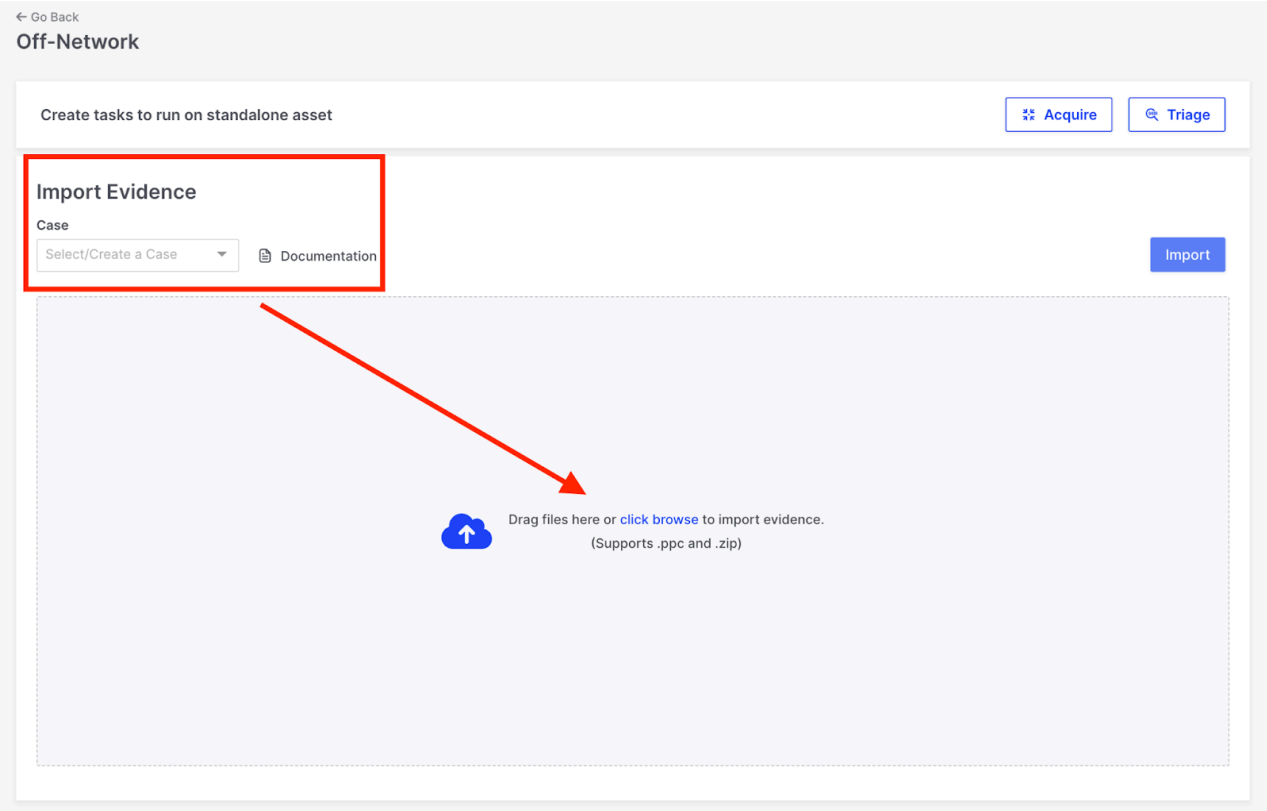
XDR Forensics will display its progress as seen below and notify the user when the activity is complete



Off-Network Responder: Progress displayed

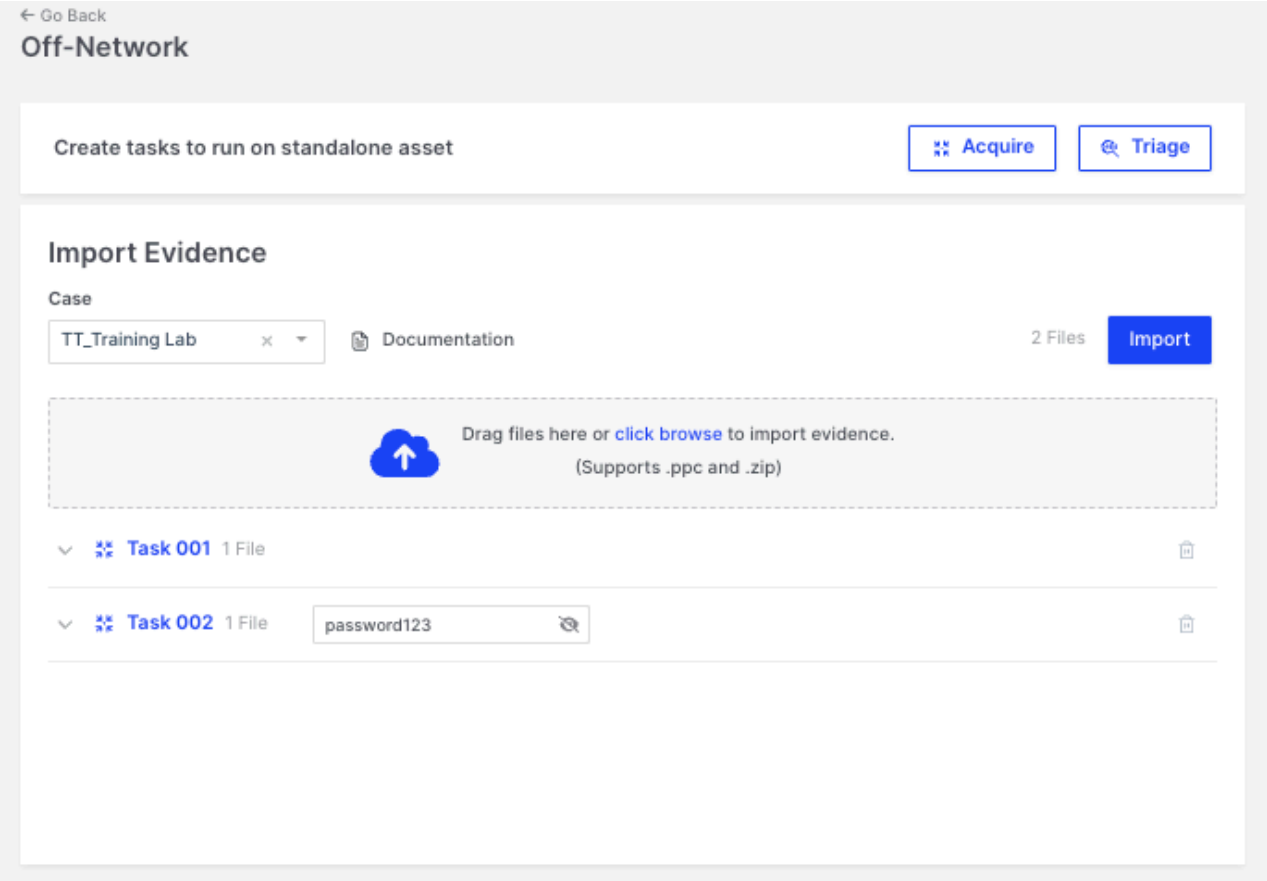
Import the Collected Data

1. After the Responder completes its task, it generates an encrypted evidence container file (.zip extension) in the directory from which it was run or the directory specified during the Responder generation process.
2. The user needs to copy or transfer the zip file so that it can be seen by and imported into the XDR Forensics console that generated it.
3. Import the .zip or .ppc file into the XDR Forensics console that created the binary.



Off-Network Responder: Import Evidence

4. If the user encrypted the collection with a password during the off-network task creation, enter the password when prompted during the import process.

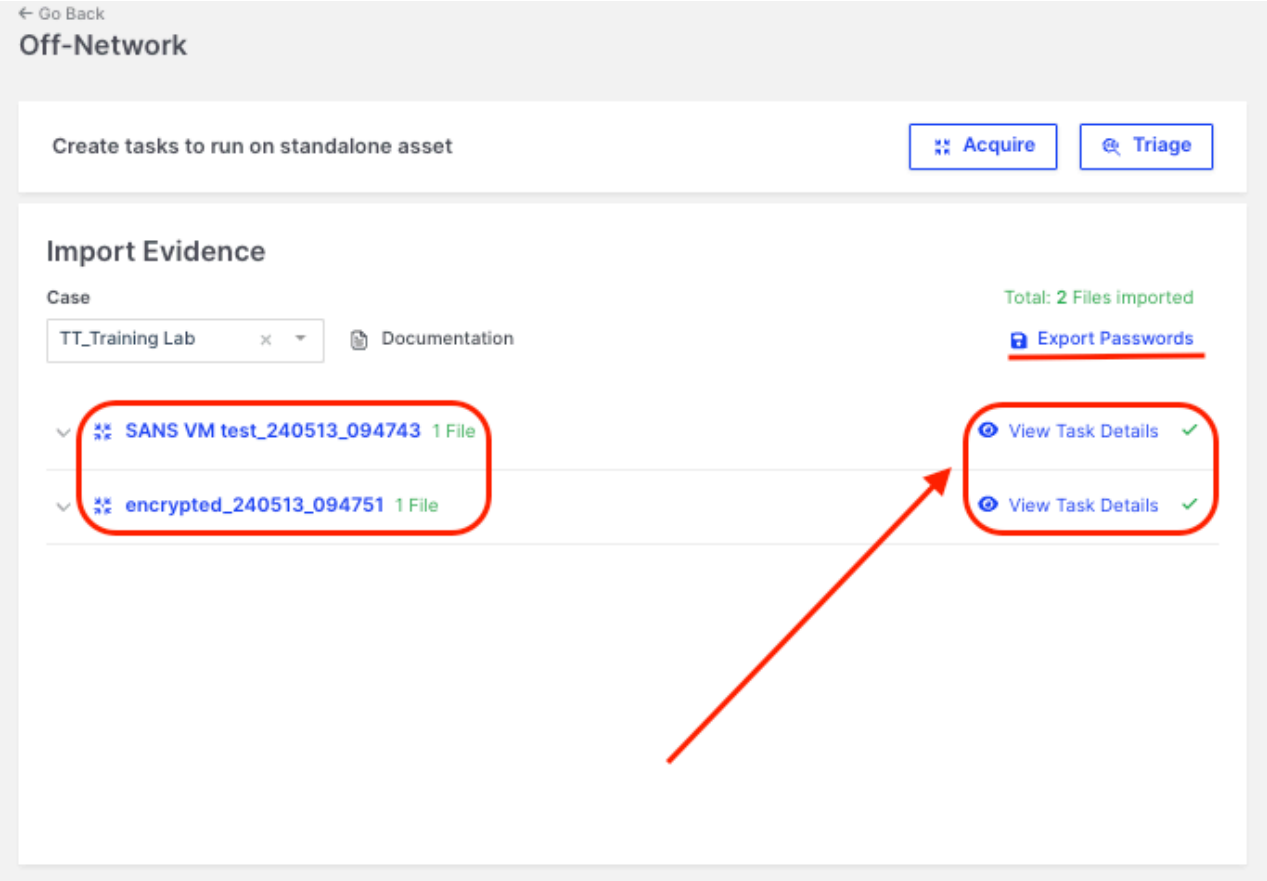


Off-Network Responder: Enter Password

In the example provided above, Task 001 is ready for import and will be automatically ingested by the console. However, for **Task 002**, the user must manually enter a password. This password is the one selected during the generation of the Responder binary as additional security to that automatically provided by XDR Forensics.

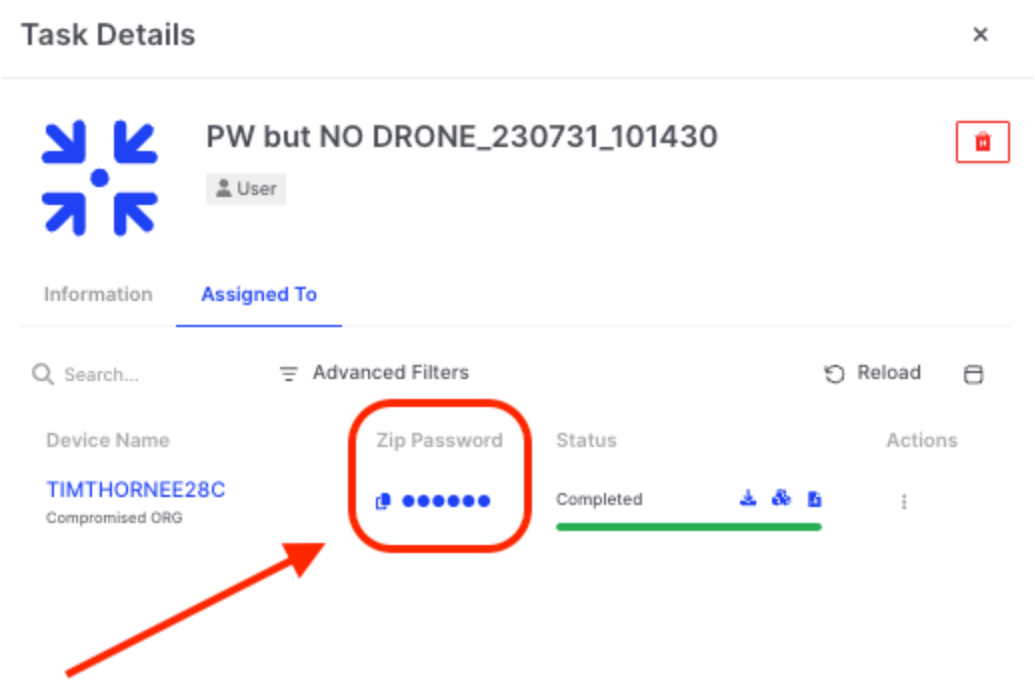
> To import collections or triage results back into the XDR Forensics console, you must use the zip file created by the off-network Responder. This file, which contains compressed evidence, can only be accessed once it's imported into the XDR Forensics console that created the off-network Responder.

After clicking the "Import" button, as shown in the example above, the user is presented with another window, as illustrated below. Here, the decrypted zip file name is revealed, allowing the user to export the passwords used to unzip the containers containing the acquired data.



Off-Network Responder: View Task Details

Clicking the "View Task Details" icon will open the Task Details window. Here, users can copy the zip password needed to decrypt the container locally.



Off-Network Responder: Password Copy

Review Report on XDR Forensics Console

Once the imported data is decrypted into the console, users review and analyze the collected data in the XDR Forensics Investigation Hub.

It is possible to import multiple .zip or .ppc files into XDR Forensics at the same time via the window shown below, while making use of our bespoke unzipping tool "**biunzip**":

biunzip >

biunzip

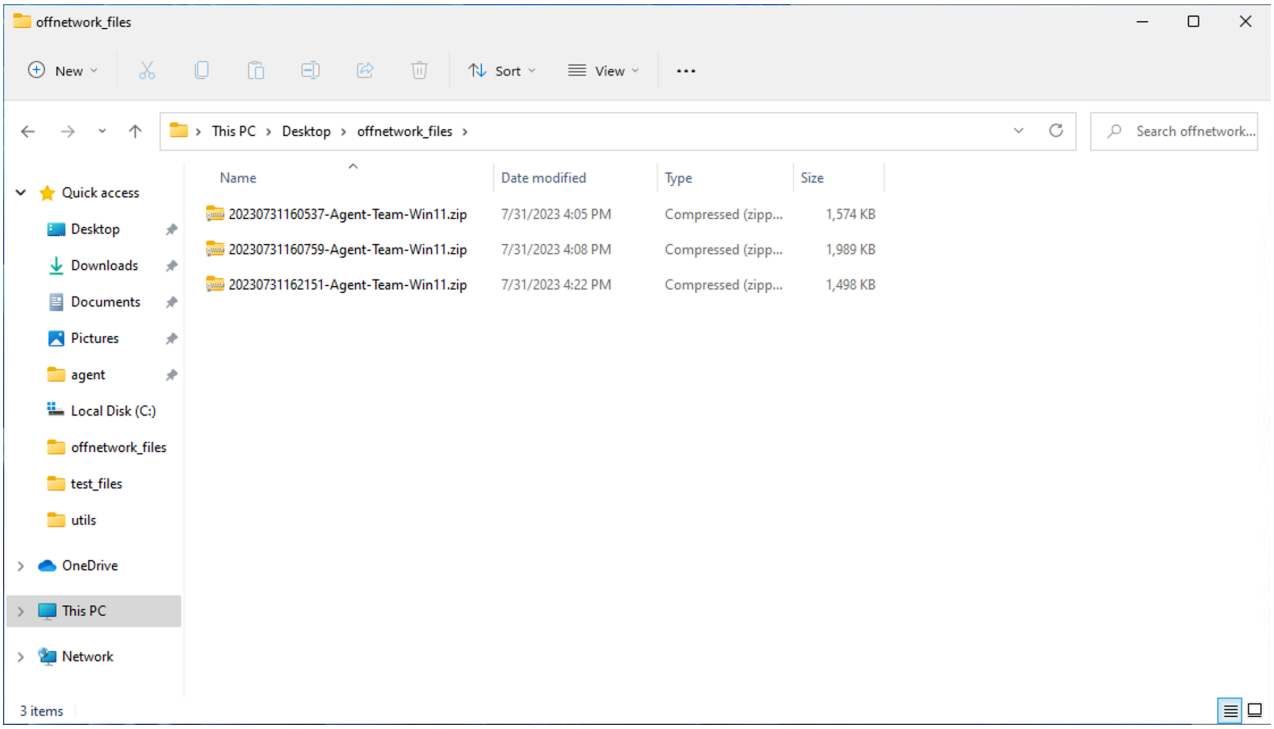
'biunzip' is a command-line tool from Cisco specifically designed to extract zip files generated by the XDR Forensics Off-Network responder.

- You can download the latest release of biunzip from the releases section on GitHub ↗.
- Biunzip will either unzip a single zip file or unzip zip files in a directory using a CSV file.
- This capability will enable running off-network investigations at scale and speed with minimal effort.

Below, we walk through the process

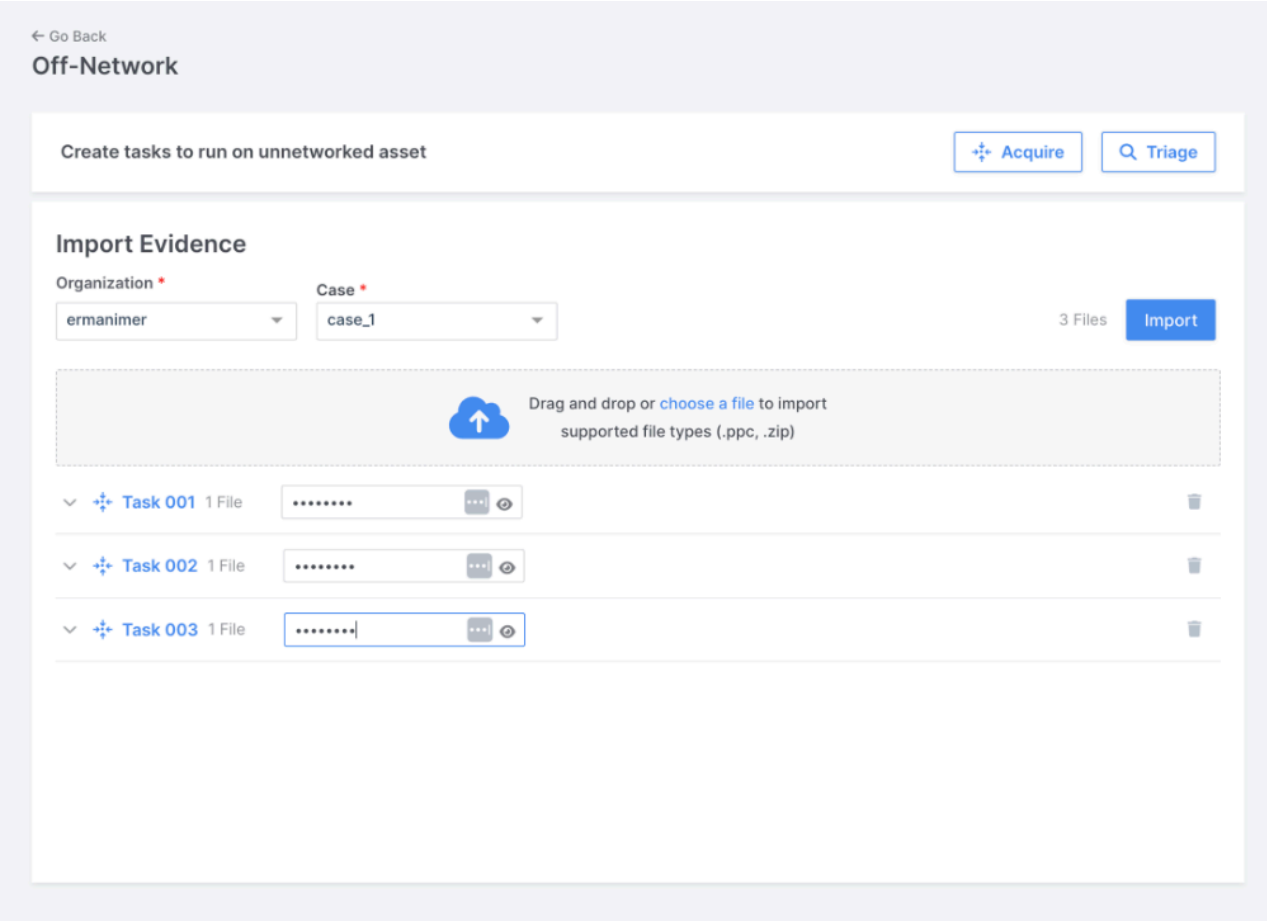
1) Download biunzip from GitHub ↗.

2) Import off-network zipped files to a machine with XDR Forensics console access and the biunzip utility:



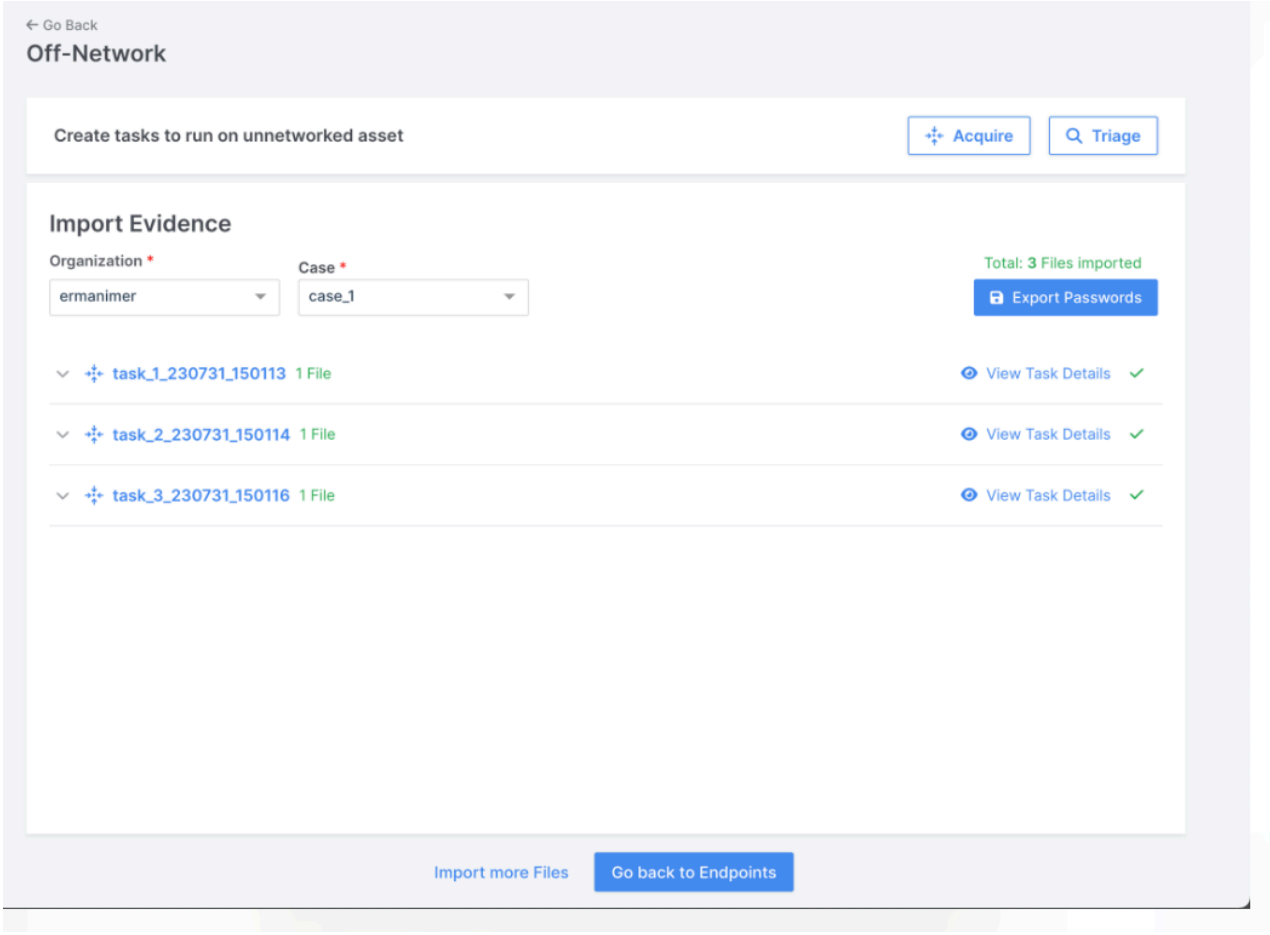
biunzip: Imported off-network collected files

3) Import off-network files into XDR Forensics (with acquisition password if the files are encrypted):



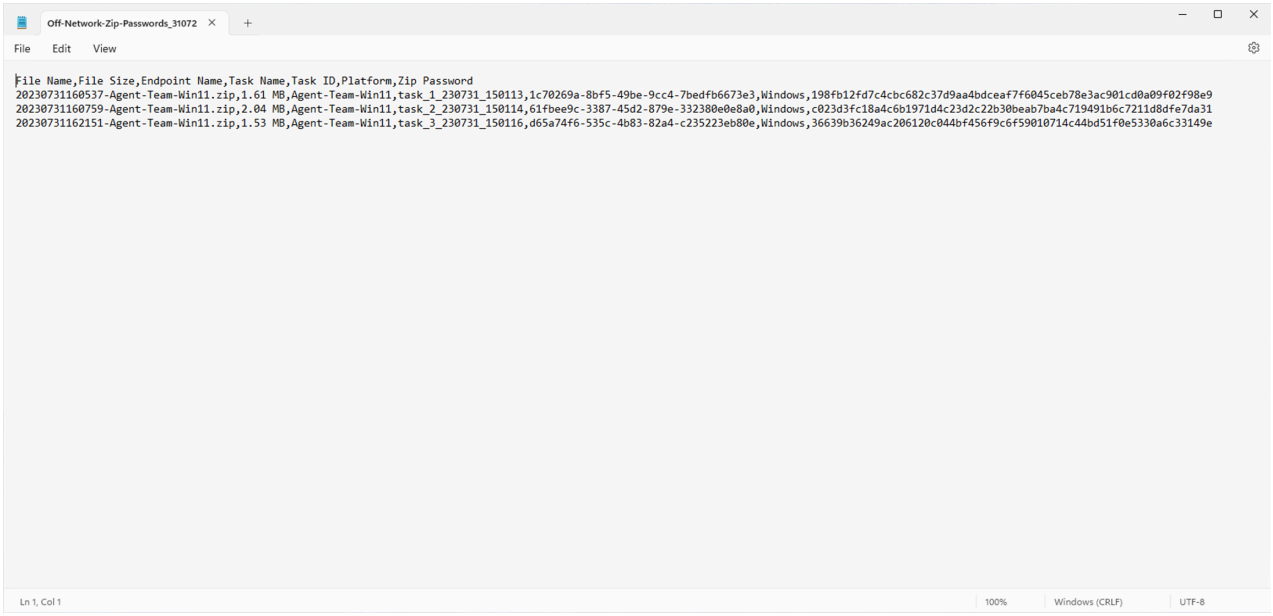
biunzip: Import Evidence

4) Export the passwords:



biunzip: Export passwords

5) View the exported passwords:

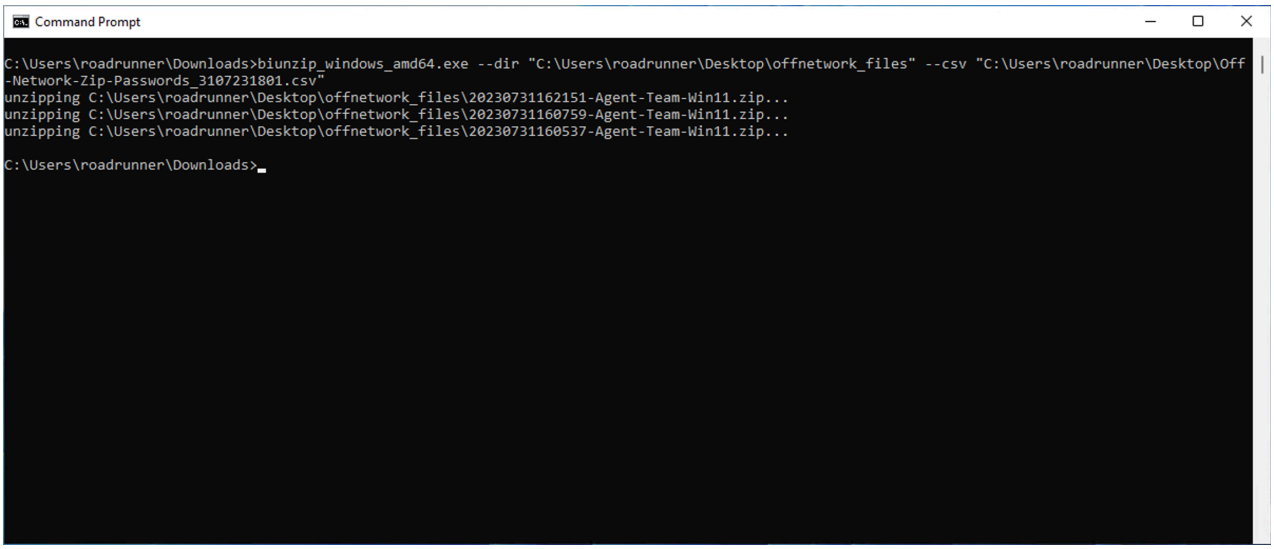


biunzip: View passwords

6) Run biunzip with the following flags and necessary flag values to unzip off-network files:

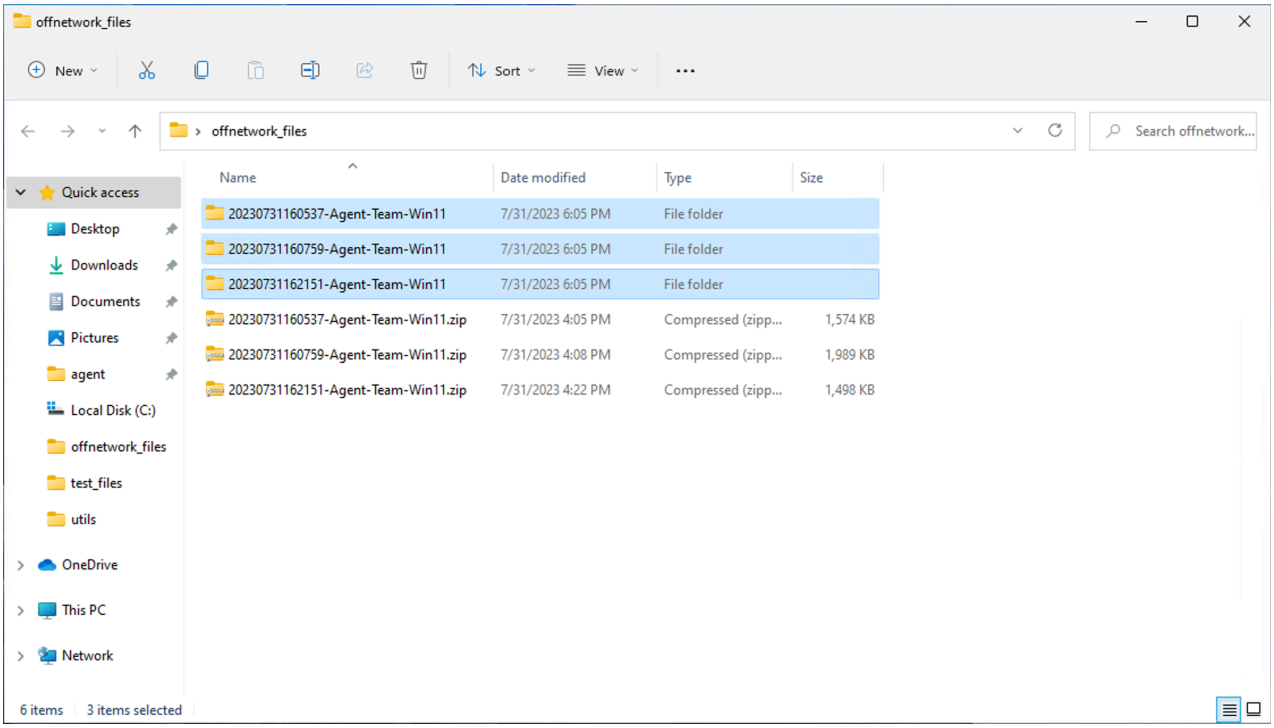
```
biunzip.exe --dir zip_dir_path --csv cvs_file_path
```

In this example, zip_dir_path points “C:\Users\roadrunner\Desktop\offnetwork_files” directory, csv_file_path points “C:\Users\roadrunner\Desktop\Off-Network-Zip-Passwords_3107231801.csv“ file.



biunzip: CLI example

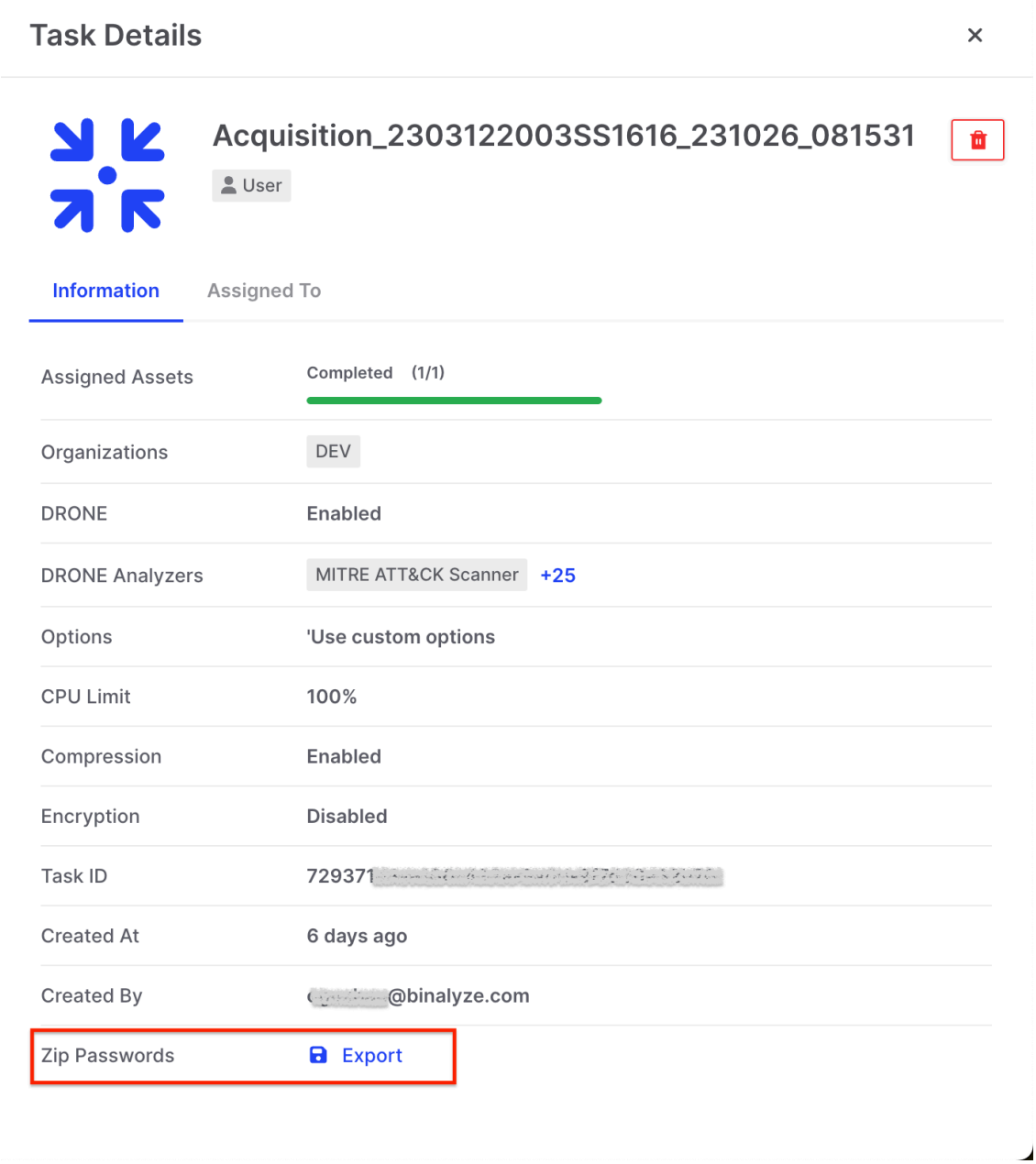
7) Here we see the unlocked zip files alongside the original locked files:



biunzip: Unlocked zip files alongside the original locked files

biunzip password file

- The biunzip tool offers enhanced access to the passwords file, streamlining the decryption of multiple Off-Network zip files.
- The Off-Network Assets passwords file will now always be available at: Asset (name) > Tasks > Details > Information Tab.



biunzip password file: Off-Network Assets passwords file location

Setting Up a Custom Case Directory

Explains how it is possible to create a custom directory for your off-network collector

1. Run as Administrator/Root
To modify settings or even to display help options, it's necessary to run the off-network program with administrator (Windows) or root (Linux/Mac) privileges. This is essential for accessing certain system directories or modifying system files

2. Using Command Line
Instead of launching the program by double-clicking, open a command terminal as an administrator and navigate to the directory where the program is located.

3. Setting the Case Directory

You can specify a custom directory for saving case files by using the `` `--case-base-dir` `` flag followed by the path to the desired folder.

For example, if you want to store cases in "`D:/Another/Folder`", the command would look like this: `offnetwork_windows_amd64.exe --case-base-dir D:/Another/Folder`

4. Viewing Available Commands
To view all available flags and commands, you can enter the following command: `offnetwork_windows_amd64.exe --help`

This will display all the configuration options available, including the `` `--case-base-dir` `` option which allows you to override the default case file directory.

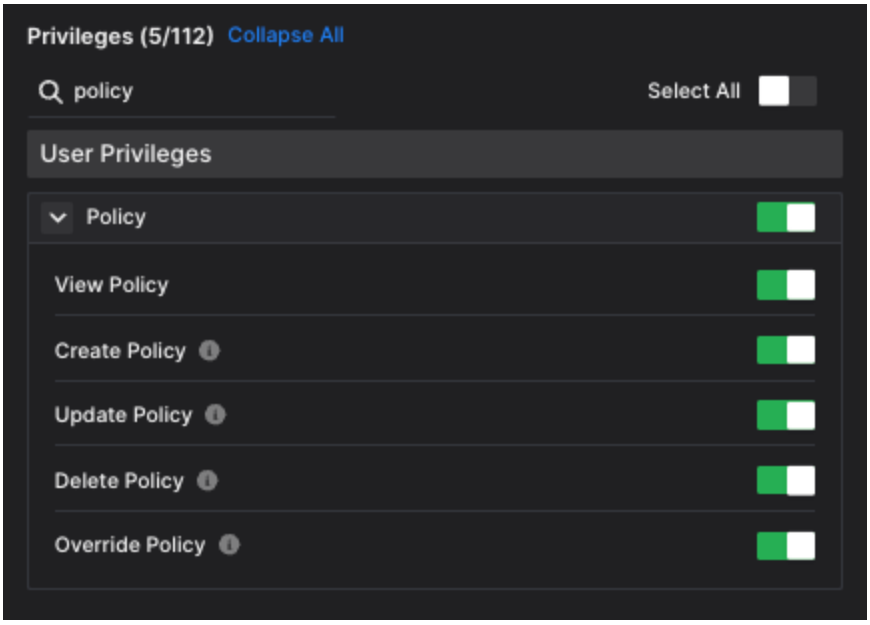
Additional Information:
&#xNAN;- The `` `--case-base-dir` `` option is supported across all operating systems that our software supports, making it flexible for various IT environments.
&#xNAN;- There is no need to create a new task file when you want to save to a different location; simply specify the desired path each time you run the program.
Please note that while the `ESXi` command is mentioned in the help output, it pertains to a new feature that might be announced later.

Policies

Policies are used in XDR Forensics to standardize critical options at the organization level for the Console, Responders, and Evidence Repositories.

XDR Forensics includes a **Default Policy** that is **read-only** and covers all configuration areas. It **cannot be edited** and acts as a fallback when no other policies are assigned or when some settings are missing. This ensures every **acquisition task** has a complete configuration, preventing gaps.

Policies in XDR Forensics help **standardize critical settings** for the **Console, Responders, and Evidence Repositories** at the organization level. While the **Default Policy** is **preconfigured and uneditable**, users with the necessary permissions can **view, create, and modify** custom policies as needed.

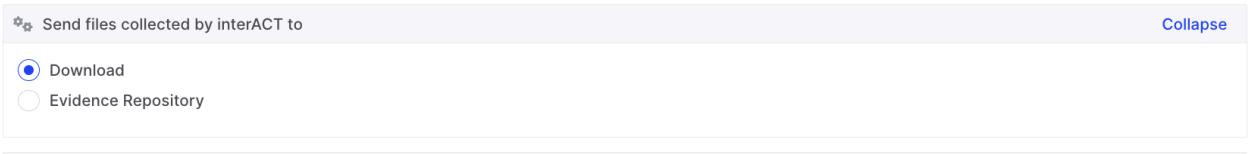


Policies: In this case the user will have access to all Policy options

To create a new policy, click the "Settings" button in the Main Menu and then select “Policies” from the Secondary Menu.

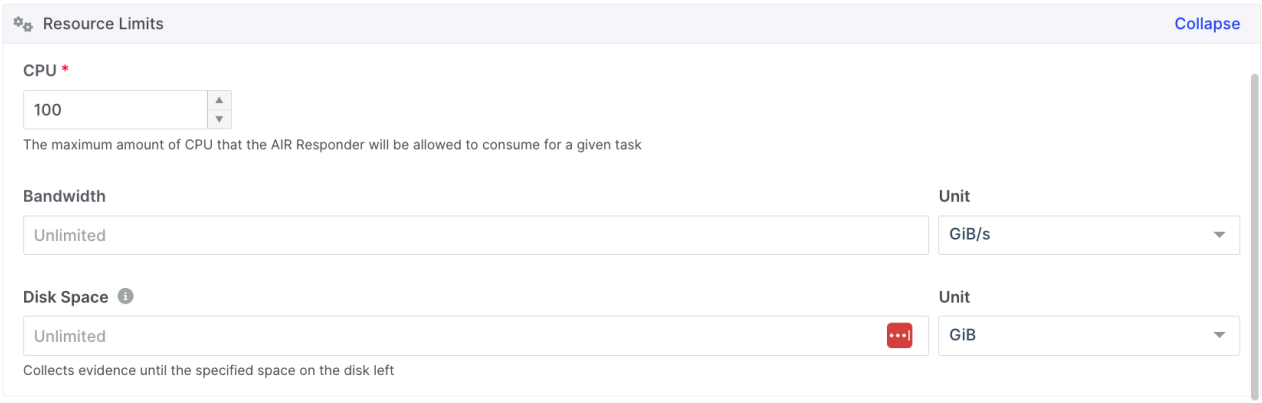
When you select the "**+Add New**" Action Button, you can create the required policies by configuring the options shown below:

1. The Policy name.
2. The Organization(s) affected by the policy.
3. The destination for collected evidence by OS platform:
    - **Local** - use the local asset.
    - **Evidence Repository** - use a remote storage location set in Evidence Repositories ↗.
    - **Path** - By default, the path to save evidence locally is: `CiscoForensics`
    - **Direct Collection** - Enable this switch to collect data while minimizing local disk space usage. During the upload process, approximately 100MB of temporary data is stored in the Cases folder, which is automatically deleted upon completion.
    - **Automatically Select Volume** - Toggle on this switch to allow XDR Forensics to select the local volume with the most available space.
4. The destination for files collected by interACT:



Policies: interACT repository settings for downloads

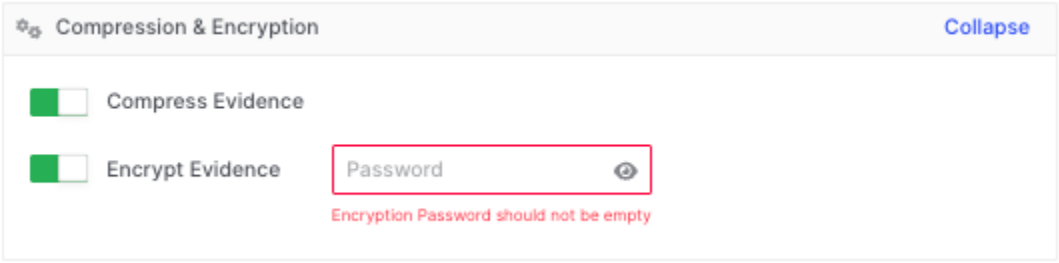
5. Asset Resource Limits utilized by XDR Forensics Task Assignments executed by Responders:



Policies: Resource Limitation Settings

- **CPU -** We recommend setting CPU usage limits to approximately 40% on active assets and endpoints to minimize disruptions. Schedule resource-intensive tasks during off-peak hours and use more lenient triage rules for in-use systems. Monitor impact and adjust as needed.
- **Bandwidth** - Bandwidth limitations primarily depend on the network and the server's constraints. To prevent accidental disruptions to mission-critical operations, users can configure a bandwidth usage limit.
- **Disk Space -** Evidence collection on the local asset will continue until the specified amount of free disk space remains.

6. **Compression and encryption settings**



Policies: **Compression and encryption settings toggled ON**

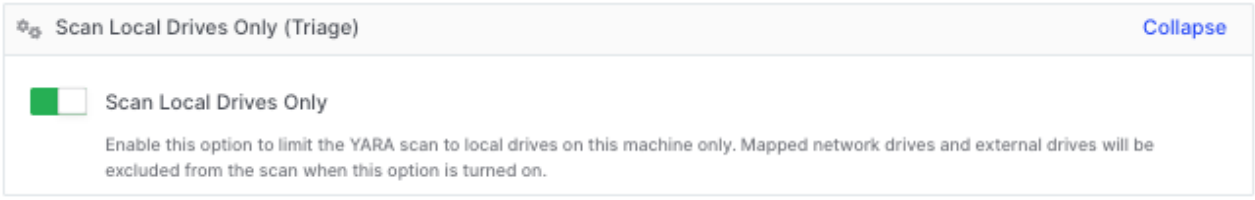
Compression is enabled by default, but users can disable it if preferred. Note that disabling compression may significantly increase the disk space required for your collections.

Encrypt Evidence toggled on will require the entry of a password, which will be needed later to access the collected data. The data will be encrypted with AES256-bit encryption and stored in a zip archive.

> We do not save the passwords you enter, and they cannot be recovered through the XDR Forensics console. Therefore, please ensure you securely store them yourself. We strongly recommend using a Password Manager application to manage and safeguard your passwords.

7. **Scan Local Drive Only (Triage)**

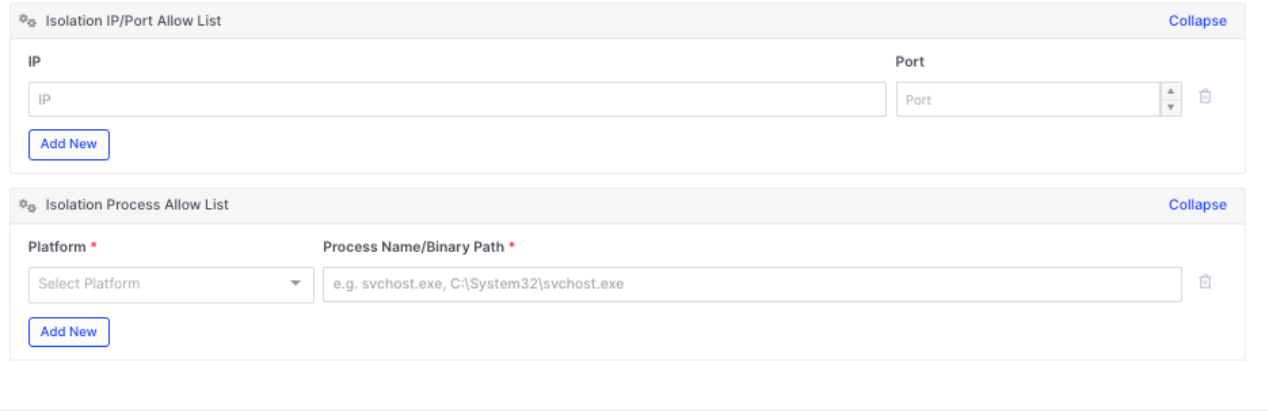
Enable this option to limit the YARA scan to the machine's local drives only, preventing it from scanning mapped network or external drives. This ensures the scan stays focused on the local system, avoiding unnecessary delays from remote locations.



Policies: **Scan Local Drive Only (Triage)**

8. **Isolation IP/Port & Process Allow Lists**

The XDR Forensics console enables the isolation of assets ↗ by terminating all existing connections to an asset and preventing any new connections. This isolation feature operates using a kernel-mode driver and does not rely on the Windows Firewall.



Policies: **Isolation IP/Port & Process Allow Lists**

In XDR Forensics Policies, the **Isolation IP/Port & Process Allow Lists** feature enables users to create custom allow lists that can be applied after an asset has been isolated by XDR Forensics. This provides more granular control over the network access and functionality of isolated assets.

Repository Explorer

Repository Explorer – Centralized Management of Evidence Repositories

The Repository Explorer centralizes access to and management of all data stored in connected external Evidence Repositories.

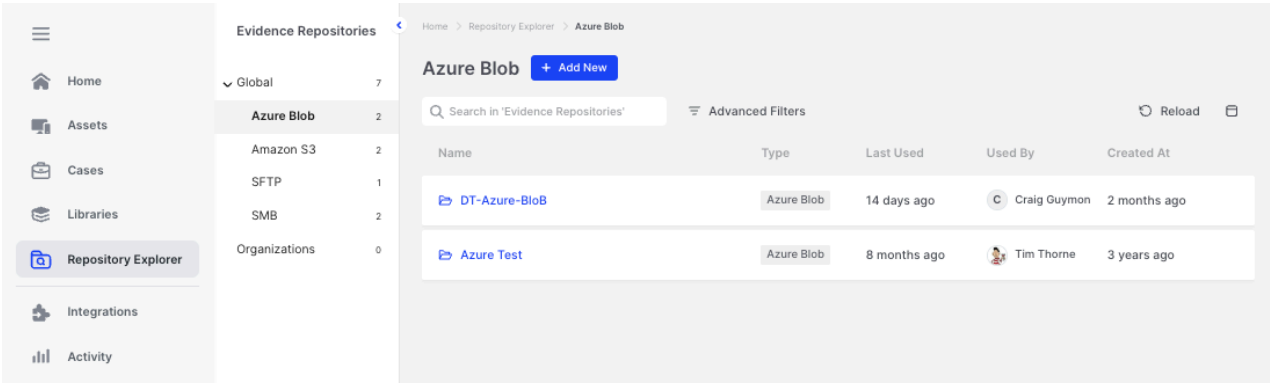
Once selected from the Main Menu, the Repository Explorer will reveal a secondary menu which lists the available repositories and their contents under Global or Organizational assignments.

Key benefits of this new feature include:

- A single interface to browse and manage repositories.
- Download individual files directly from Evidence Repositories to the user's machine.
- Directly upload files into existing repositories using the Upload File action button.
- Import any compatible file in the repository to mount as a ‘disk image asset’ using the new "Import Disk Image" action.
- Improved navigation between global and organization-specific repositories across Azure, AWS S3, SFTP, and SMB.

Using Repository Explorer

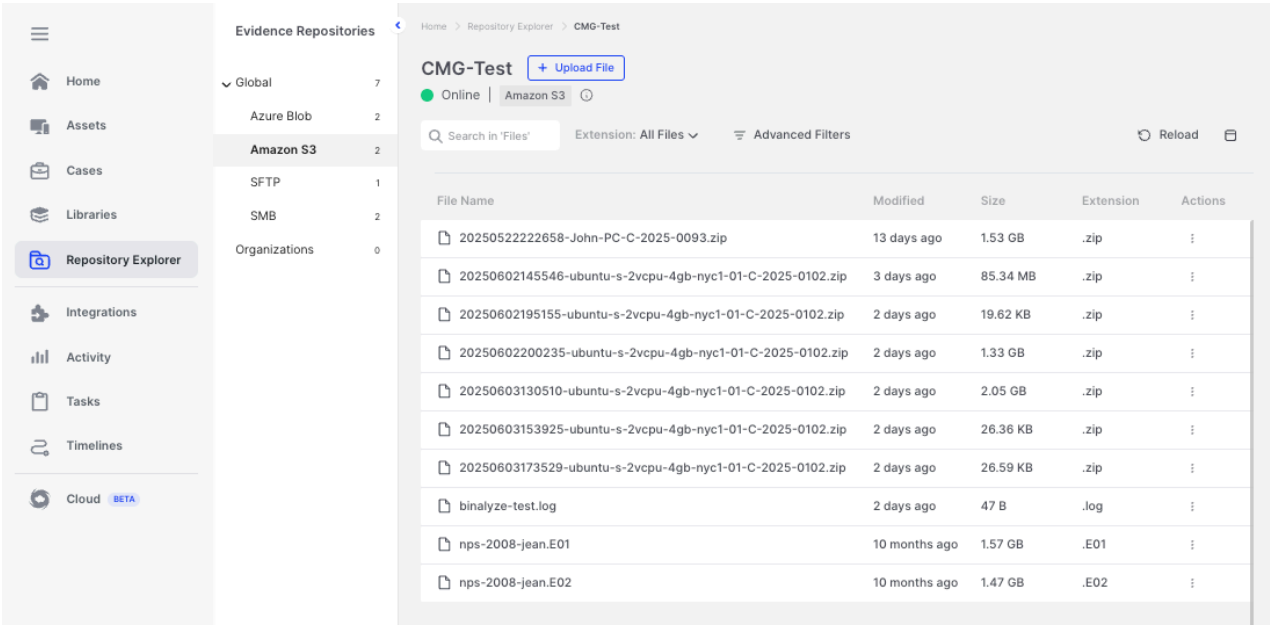
New evidence repositories can be added by selecting the Add New action button.



Repository Explorer: Azure Blob view.

Individual evidence repositories are listed in the table view and can be explored by selecting the link embedded with the repository name.

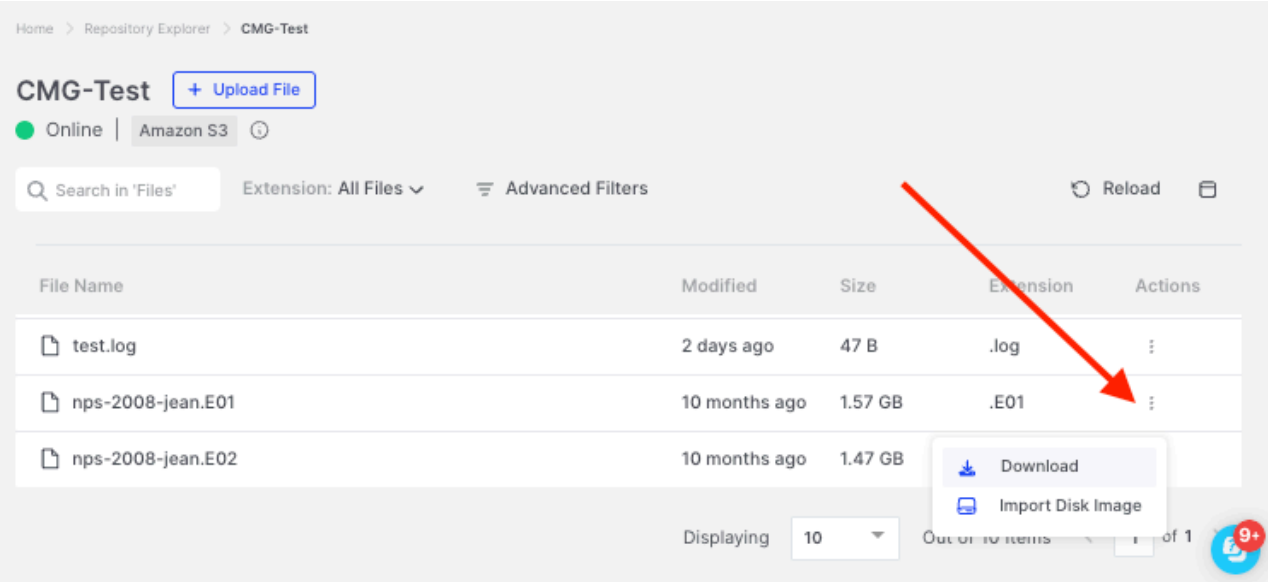
In the example below, you can see that the Amazon S3 evidence repository has been selected, and within that blob, an area named 'CMG-Test' is displaying its contents.



Repository Explorer: Amazon S3 contents

Selecting the **Action** icon next to each item in the repository reveals two options:

- **Download**: Allows the user to download the selected item to the local machine on which they are working.
- **Import Disk Image**: Enables users to import a disk image file into the File Explorer, where it can be browsed and examined in a mounted state.



Repository Explorer: Actions

Responder Proxy Support

Problem Statement

The XDR Forensics responder needs to access certain network services to work properly. If any of these connection requirements are not satisfied, the XDR Forensics responder may not work properly. XDR Forensics responders use the network connection provided by the operating system. If some kind of proxy service is used in the enterprise network, the XDR Forensics responder probably can not detect the proxy configuration hence can not connect to the required services and does not work properly.

XDR Forensics Responder Proxy Support

The updated version of XDR Forensics responders automatically detects the proxy server configuration on the asset and modifies the network connection methods to access required services. XDR Forensics responders read the proxy configuration settings where it is located according to the operating system, Windows, Linux and macOS are supported operating systems.

Minimum network connection requirements and associated definitions are listed below.

XDR Forensics Responder to XDR Forensics Console connection requirements

- TCP/IP 80, 443 HTTP/HTTPS , 4222 NATS for Real-Time Task assignments, 443 WebSocket for interACT

The XDR Forensics responder communicates with the XDR Forensics console over 80 and 443 with HTTP/HTTPS. Therefore, TCP 80,443 HTTP/HTTPs ports and protocols must be open and accessible. In order for Real-Time task assignments to work, TCP/IP 4222 port must open and accessible. Similarly, in order for interACT to work, the WebSocket protocol must be configured over HTTPS.

XDR Forensics Responder to Evidence Repository connection requirements

If the collected evidence needs to be uploaded to a remote domain, the responder must be able to access these remote domains via HTTP/HTTPs, SMB, SFTP, FTPS and Amazon, Azure and Google domains, depending on the configuration previously defined in the evidence repository. If there is no support on the proxy server during the connection phase of protocols such as SMB, SFTP, FTPS, the Direct connection method is tried.
In addition, HTTP/S Proxy connections are made by establishing a Tunnel with the HTTP Connect method. In addition to HTTP Proxies, SOCKS5 Proxy type is also supported.

Timeline

The Timeline in AIR 5.0+ has been completely redesigned and embedded directly into the Investigation Hub, providing investigators with a powerful, unified view of all timestamped evidence. This enhancement eliminates context switching, accelerates analysis, and enhances collaboration.

See It in Action ↗

> The new Timeline is only available for cases created in AIR v5 and later. Timelines generated in earlier versions cannot be migrated or upgraded to this new format.

Key Timeline Features

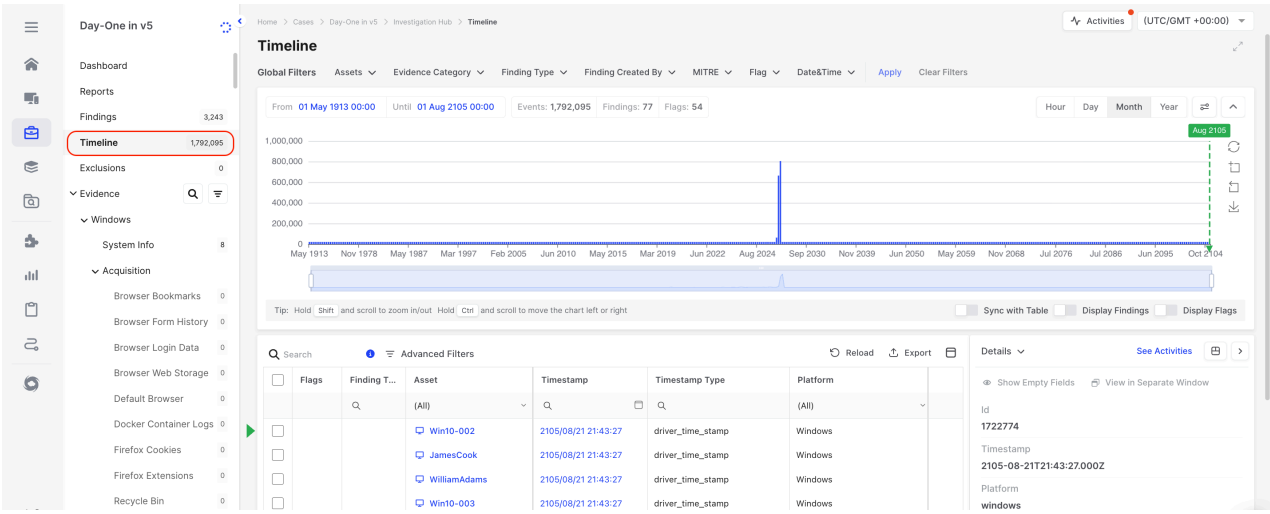
- **Automatic Event Generation**
  Every piece of evidence with a timestamp automatically generates a timeline event. There are no longer restrictions based on asset type or data source.
- **Integrated in the Investigation Hub**
  The Timeline is now a core part of the Investigation Hub, rather than a separate module. Global filters, case filters, and advanced filters work seamlessly across all views.
- **Advanced Filtering**
  Analysts can apply both global filters (date ranges, assets, evidence types) and evidence-specific filters (e.g., IP addresses, paths, user IDs). This allows precise targeting of relevant events without visual clutter.
- **Relative & Nearby Time Filtering**
  From any timestamp in evidence or findings, analysts can quickly pivot to surrounding events (e.g., “5 minutes before and after”), making causality and event sequencing clearer.
- **Interactive Timeline Bar**
  A zoomable timeline bar with density maps highlights periods of activity. Analysts can navigate quickly, change granularity (hour, day, month, year), and toggle findings/flags overlays.
- **Timeline Table Enhancements**
  A redesigned event table with infinite scrolling and expandable details lets analysts flag, annotate, or promote events to findings. All actions are synchronized across the platform.
- **Collaboration Features**
  Notes, flags, and findings applied in the Timeline are reflected across the entire Investigation Hub, supporting team-based workflows and ensuring consistent visibility.
- **Export Capability**
  Timeline tables can be exported to CSV with optional evidence metadata in JSON, enabling reporting, compliance, and cross-platform analysis.

Using the Timeline

The Timeline is embedded directly within the Investigation Hub, providing analysts with a complete, interactive view of all timestamped evidence, unrestricted by asset type or data source. This walkthrough will guide you through using the new Timeline effectively in your investigations.

Accessing the Timeline

1. Open any case in the **Investigation Hub**.
2. Select the **Timeline** tab from the secondary.
3. The Timeline view loads automatically with:
    - **Timeline bar** (overview of events, zoomable).
    - **Timeline table** (detailed events with metadata).
    - No filters, Findings, or Flags will be applied in this initial view.



Timeline: Initial view

Navigating the Timeline

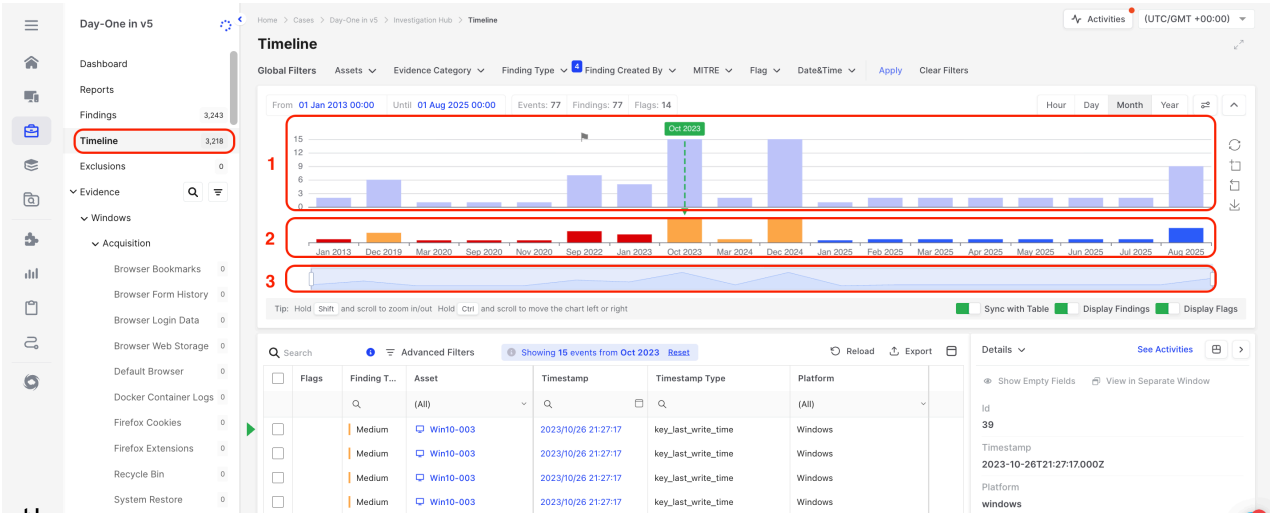
- **Zooming**: The timeline supports several zoom options for flexible navigation:
  - **Zoom buttons** – Switch between **Year, Month, Day, or Hour** views.
  - **Drag-to-zoom** – When the zoom button (located to the right of the chart) is active, drag and drop across the chart to define the exact area to focus on.
  - **Shift + scroll** – Hold **Shift** while scrolling to zoom in or out with finer control.
  - **Mini-map** – Compress or expand the visible area in the mini-map to quickly adjust the level of detail. This provides immediate visibility into data density across the timeline, helping analysts effectively explore large time spans and reduce visual fatigue.
- **Cursor sync**: Clicking on the bar updates the table; the first row always matches your selected point.

Timeline Enrichment Options

Below the timeline chart, three toggle switches allow you to customize and enrich the view:

- **Sync with Table** – Ensures the evidence table reflects only the focused time range when zooming, panning, or changing dates on the timeline bar.
- **Display Findings** – Adds a dedicated chart that highlights findings separately from other events (area 2 in screenshot below).
- **Display Flags** – Displays flag icons directly on the main timeline chart (area 1 below), indicating the location of flagged items.

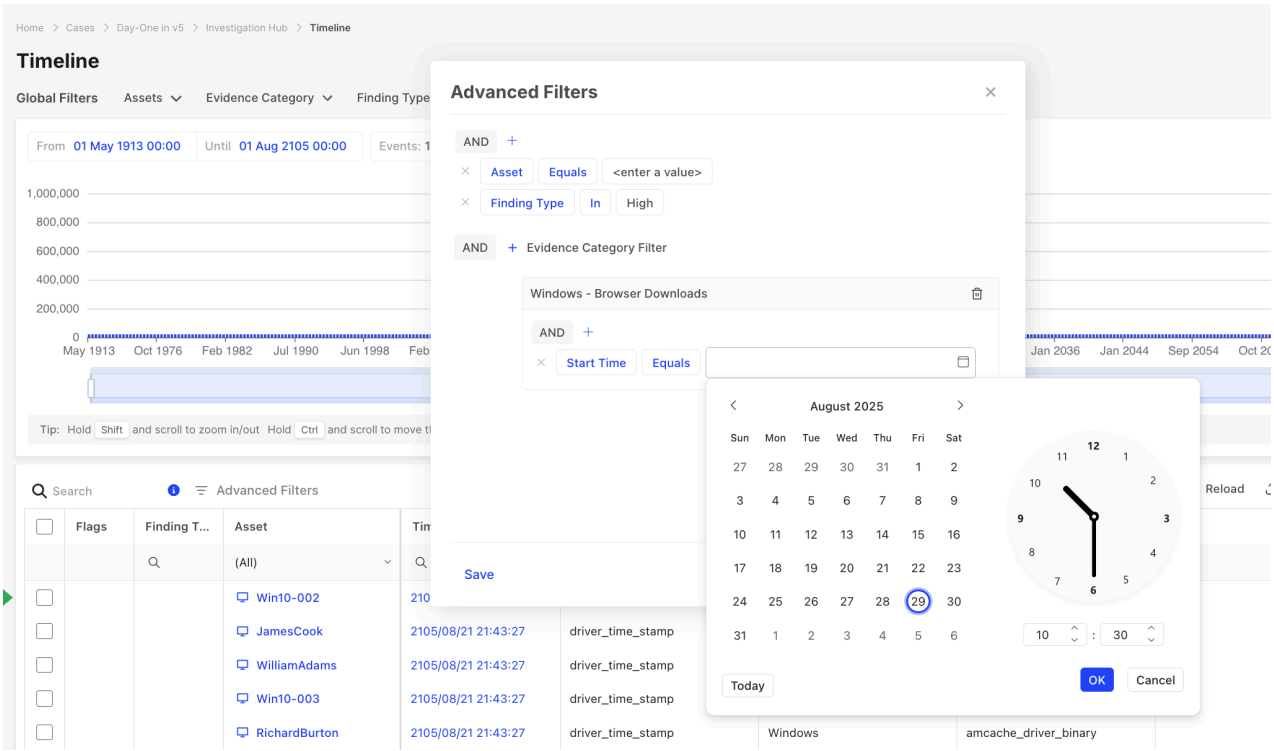
These options help analysts quickly correlate evidence, findings, and flagged events within the same timeline context.



Timeline: Enrichment options; 1-Timeline, 2-Findings, 3-Mini-map.

Applying Filters

1. **Global filters** (e.g., Assets, Evidence Category, Finding Type) apply to the entire Investigation Hub, including the Timeline.
2. **Timeline bar filters** refine the visual display via zoom functions.
3. **Advanced filters** apply at the table level, including evidence-specific fields (e.g., IP addresses, file paths, user IDs). Constructing filters, whether basic or complex, is straightforward, as shown below:



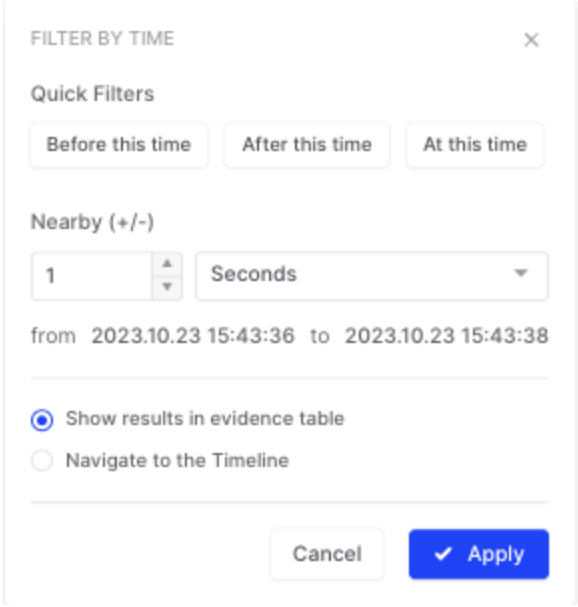
Using Relative Time Filtering

Clicking on a timestamp opens the *Filter by Time* dialog, which provides quick and precise options to refine the investigation window:

- **Quick Filters**: Instantly narrow results to events that occurred *before*, *after*, or *at* the selected timestamp.
- **Nearby (+/-)**: Define a time window around the selected timestamp (e.g., ±1 second, ±5 minutes, ±7 days). This helps capture surrounding activity that may be contextually relevant to the investigation.
- **Custom Range Preview**: The dialog automatically displays the calculated "from" and "to" time boundaries based on your selection.
- **Result Options**:
  - **Show results in evidence table** – keeps the filtered evidence within the current table view for immediate analysis.
  - **Navigate to the Timeline** – pivots the same filtered view into the Timeline for a broader chronological context.

Use Case Example:
If an analyst identifies a suspicious process execution at *15:43:36*, they can quickly pull in all related evidence from one second before to one second after. This makes it easy to uncover precursor events or immediate follow-on activity without manually building time-based filters.

This relative filtering approach complements absolute time filters, providing flexibility for both **precision pivoting** and **contextual timeline exploration**.



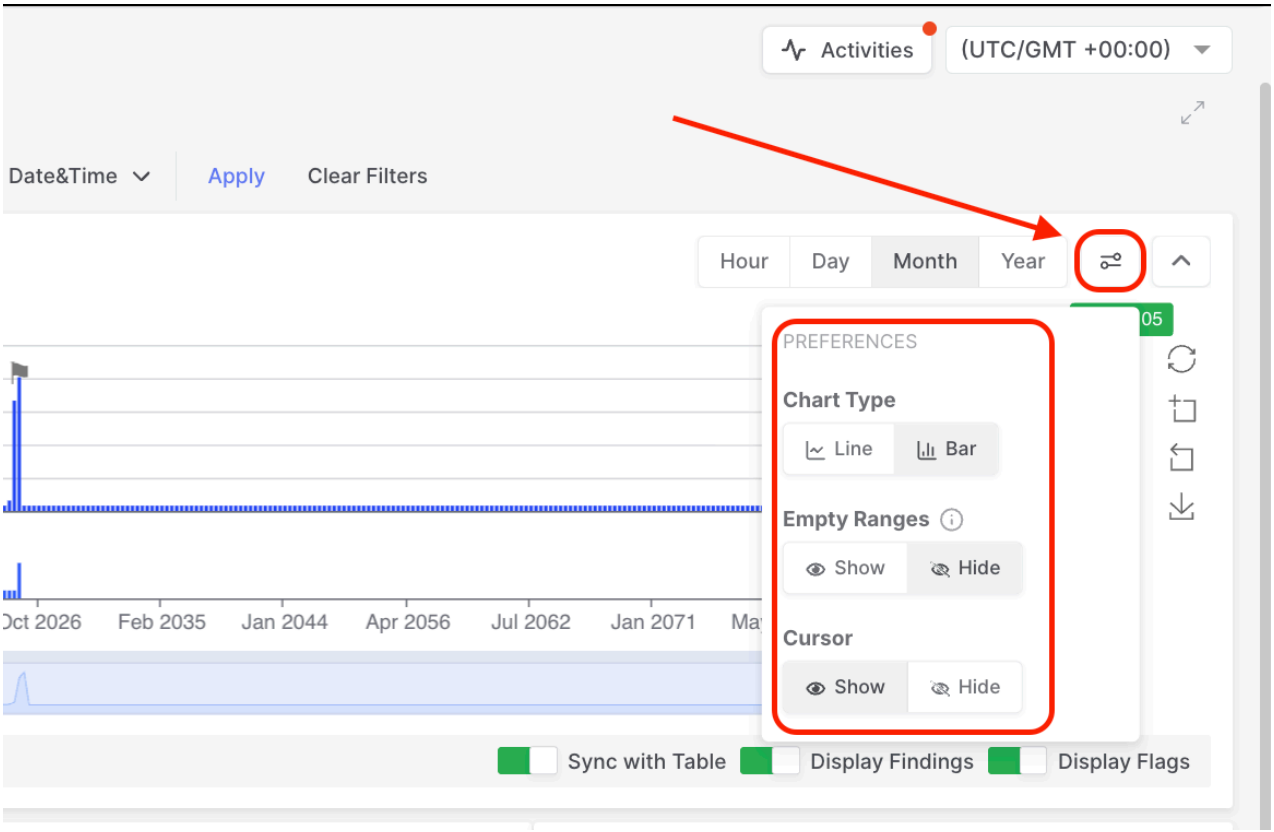
Timeline: Relative Time Filtering

Timeline search box

The **search box in the Timeline evidence tables** allows analysts to quickly refine results using keywords, exact phrases, exclusions, and logical operators. You can combine multiple terms with '`OR'`, and exclude terms with a leading minus sign (e.g., `-anonymous`), or search for exact matches using quotation marks (e.g., `"logon failure"`). This advanced search capability enables more precise filtering of evidence, allowing investigators to surface only the most relevant events within large datasets.

Timeline Preferences Panel

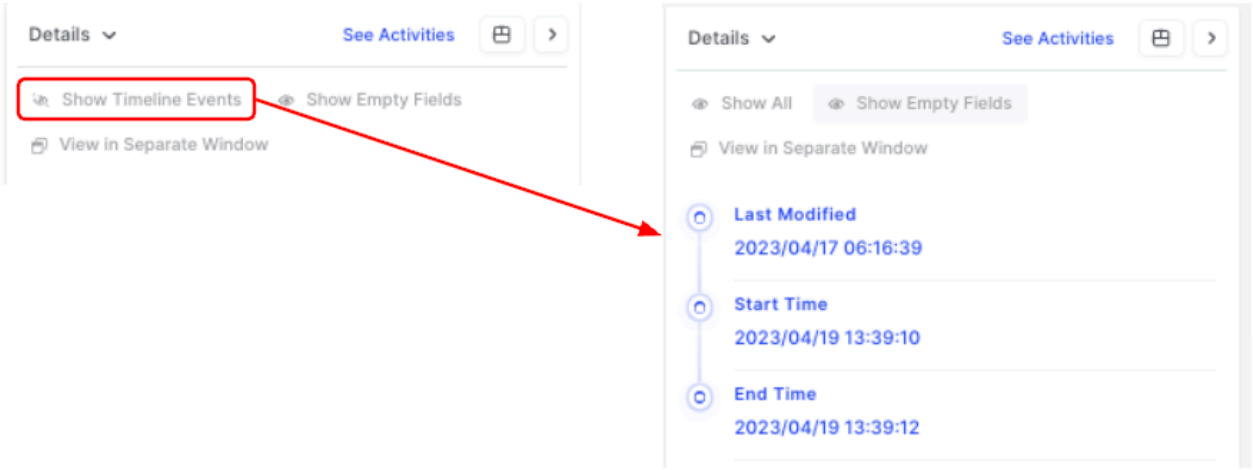
The timeline bar includes several layout and display options to help analysts tailor the view to their investigation. You can switch between **bar** and **line chart** modes, show or hide empty ranges to create a more compact timeline, and toggle the cursor for precise navigation. These preferences make it easier to adapt the visualization for complex investigations or when working with large volumes of evidence.



Timeline: Preferences Panel

Show Timeline Events

- Within the Investigation Hub, selecting an evidence item in the evidence table opens the **Details** panel. Here, the **Show Timeline Events** option reveals all timestamps associated with that item. For example, if the artifact includes a Last Modified, Start Time, and End Time, each of these values will be displayed in this view, and these additional timestamps can themselves be subjected to further relative timestamp filtering.



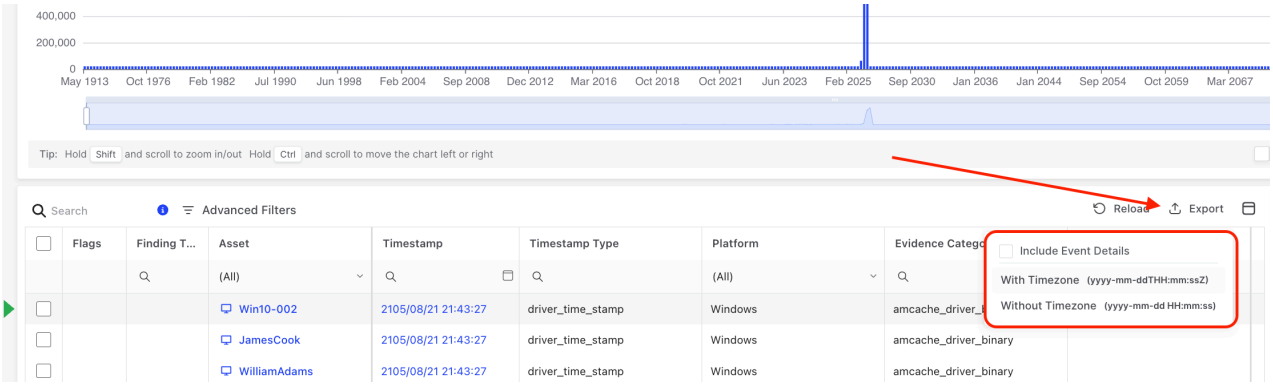
Timeline: Show Timeline Events

Flags, Findings, and Collaboration

- Use the **toggle controls** in the Timeline bar to show or hide flagged events and findings.
- Flag or annotate events directly in the table; all changes synchronize across the Investigation Hub.
- Findings promoted from Timeline events automatically appear in the Findings view for team visibility.

Exporting Timeline Data

- Select **Export** from the Timeline table.
- Choose CSV export with UTC or local time.
- Optionally include detailed evidence metadata in JSON format.
- Use exports for compliance reporting or importing into external systems.



Timeline (pre version 5.0)

> The legacy Timeline (used in AIR versions prior to 5.0) will be removed from the platform with the release of **AIR 5.1**, scheduled for mid-September 2025. The information in the following section is provided temporarily to support customers during their transition to the new, fully integrated Timeline within the Investigation Hub. This new version offers a more advanced, comprehensive, and streamlined experience for timeline analysis.

One-Click Timeline Creation for Swift Collaboration

The traditional way of creating timelines is collecting evidence, parsing it, and combining the results using CSV files. Time is a critical factor in investigations, and with the 'One-click' Timeline creation feature, investigators can initiate and collaborate on timelines with just a click. This not only expedites the process but also facilitates remote and multi-user collaboration within a single timeline.

XDR Forensics comes to the rescue to solve this problem. You can easily create timelines for multiple assets in parallel and view the results on a collaborative, web-based user interface, where you can tag/flag each piece of evidence.

Timelines can be created from a single asset and can be easily enriched using additional evidence, such as:

- Additional Assets
- CSV Files
- Milestones
- Off-Network Acquisitions

All flagged/tagged evidence is listed in the "Flagged" section, making it easy to create reports before finalizing an investigation.

Building 'Super-Timelines' on the Fly

Flexibility is at the core of effective investigations. With Timeline Analysis, investigators can add more assets at any time to an existing timeline, creating what we like to call 'super-timelines.' This dynamic approach enables the consolidation of diverse assets into a comprehensive timeline for a holistic view of the investigation.

Existing and new Timelines can be created by selecting "Timelines" from the Main Menu.

To create a new Timeline, select the "+Add New" button at the top of the page.

The New "Timeline" then gives you the option to 'Create with selected assets' or 'Create an empty timeline and add evidence later'

You can now search for and select the assets desired for the Timeline.

Having selected the assets to include in the Timeline, you now have to define the task by:

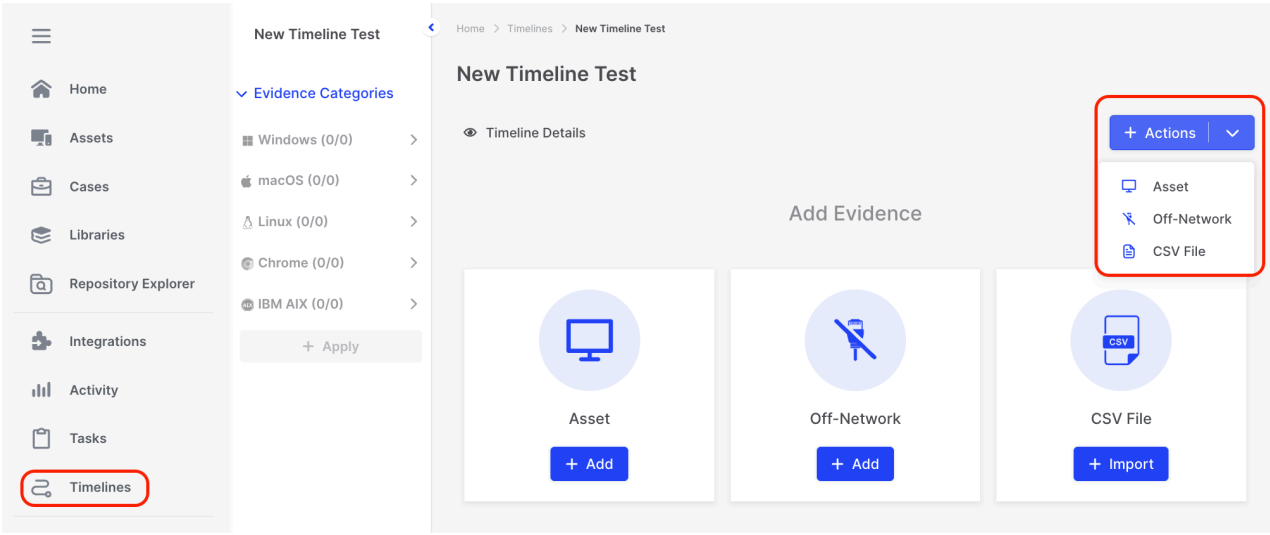
1. Giving the Timeline a name.
2. Allocating it to a Case.
3. Selecting a Timezone
4. Providing a description (Optional)

Seamless Integration of Offline Assets and CSV Datasets

Timeline Analysis goes a step further by allowing the import of offline asset acquisitions or CSV datasets into the same timeline. This ensures that investigators can amalgamate a wide range of data sources, enriching the investigative process.

XDR Forensics now presents you with three options for adding data to your new Timeline:

1. Add an asset
2. Add an off-network asset
3. Import a CSV file



Compare: Select the data type to add to the Timeline

Behind the Scenes: Trimmed-Down Evidence Acquisition

While Timeline Analysis presents a user-friendly interface, it is supported by a powerful evidence acquisition mechanism behind the scenes. This mechanism selectively includes 'timestamped evidence,' ensuring a concise and relevant timeline. By default, this includes:

- All evidence with a timestamp property
- Browsing history
- AMCache
- SRUM data

Precision Flagging for Enhanced Evidence Management

Timeline Analysis introduces the concept of multiple flags for evidence items. Investigators can flag items to highlight their significance, and all flagged items are conveniently listed in the 'Flagged Evidence' section. This section can be filtered, providing a focused view of critical evidence.

Enrich Timelines with Manually Inserted 'Milestones'

Significant events often mark investigations, and Timeline Analysis acknowledges this by allowing investigators to manually insert 'milestones'. These milestones serve as markers for noteworthy occurrences during the investigation.

In conclusion, Timeline Analysis is not just a feature; it's a comprehensive solution for investigators seeking precision, flexibility, and collaboration in their digital investigations. With 'One-click' Timeline creation, the ability to build 'super-timelines,' integration of diverse data sources, manual milestones, streamlined reporting, and precise flagging, investigators can confidently navigate the complexities of digital evidence.

Triage

Threat Hunting at speed and scale

Almost every case starts with one or more leads. If there are too many leads, the investigator may need to validate each one individually, which is a time-consuming process. Alternatively, if there are too few leads, investigators lack sufficient information to continue their case. Neither situation is good for the investigation and can impact the speed of resolution.

Triage is the process of identifying and prioritizing the evidence that will be analyzed and evaluated. However, prioritizing this evidence is not a straightforward or easy job. An Investigator needs lots of data, leads, or experience to do it well. So, they generally use known attack indicators, referred to as the IOC (Indicator of Compromise). An indicator of compromise (IoC) in computer forensics is an artifact observed on a network or in an operating system that, with high confidence, indicates a computer intrusion.

An investigator or analyst will generally scan all system data or a portion of it to identify these IOCs. When they see a match, it typically means that those systems are related to a specific attack type and need to be investigated first. Investigators use YARA, osquery, and Sigma rules for these scans. Investigators can define and scan IOCs by using XDR Forensics's built-in YARA, osquery, and Sigma template rules or editors.

The XDR Forensics DFIR Suite provides investigators with three different tools for triage, which, as stated, are YARA, osquery, and Sigma. These tools generally scan assets to find specific data using IOC (Indicator of Compromise).

> ***YARA*** *(*Yet Another Recursive Acronym) *is a tool aimed at (but not limited to) helping malware researchers identify and classify malware samples. With YARA you can create descriptions of malware families (or whatever you want to describe) based on textual or binary patterns. Each description, a.k.a. rule, consists of a set of strings and a boolean expression that determines its logic.*

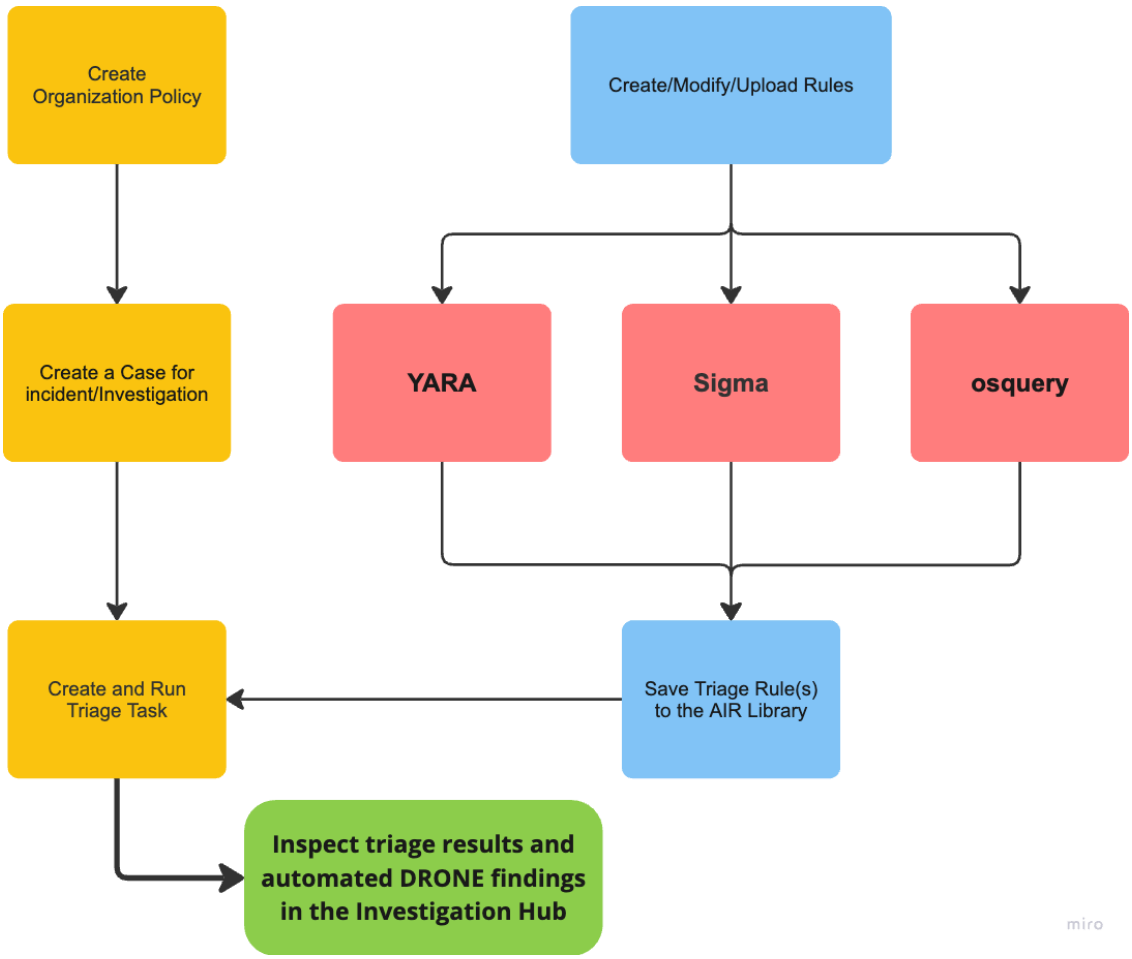
> ***Sigma*** *is a generic and open signature format that allows you to describe relevant log events in a straightforward manner. The rule format is very flexible, easy to write, and applicable to any type of log file. The primary purpose of this project is to provide a structured form in which researchers or analysts can describe their developed detection methods and make them shareable with others. Sigma is to log files what Snort is to network traffic, and YARA is to files.*

> ***osquery***, an open-source tool, employs SQL-like queries to extract intricate system data. It offers cross-platform compatibility, making it ideal for real-time system monitoring, security analysis, and compliance assessments. Security professionals often utilize it to detect vulnerabilities, monitor system changes, and ensure compliance with security standards.

XDR Forensics features a library for YARA, osquery, and Sigma rules, allowing investigators to develop, validate, and manage their rules directly within the platform using the built-in editors. These rules can be saved to **Libraries > Triage Rules** in XDR Forensics.

Investigators can efficiently conduct threat hunting and scan their assets by selecting the necessary rules from the library. The Triage process flow, depicted below, illustrates how organizational policies allow administrators to control XDR Forensics's functionality and define role-based permissions for specific activities.

Creating a case in XDR Forensics also enables users to centralize all collections, triage results, and activities related to a specific incident or investigation. This integration allows the Investigation Hub to dynamically present all information, from raw evidence to automated DRONE findings, in a unified view.



Triage: Triage creation and execution flows

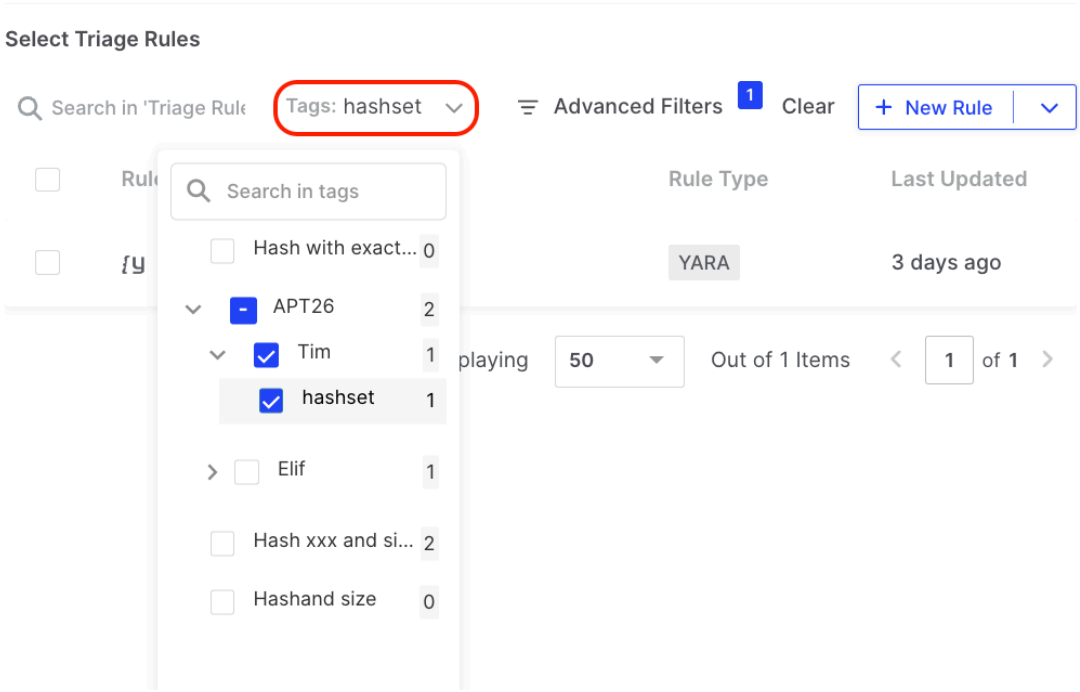
NB: Character limitation for a single triage rule

- **To prevent the browser from becoming unresponsive, we have limited the maximum character count to 350K in a single Triage Rule.**

Tagging Triage Rules

Triage rules in the XDR Forensics console can be assigned tags, which help organize the rules and filter them when required. This feature for triage rules enables more efficient management and allows for streamlined searches and improved organization within the console.

When creating or using a Triage Rule, the UI allows the user to filter existing rules by their associated Tags.



Triage: Tagging Triage Rules

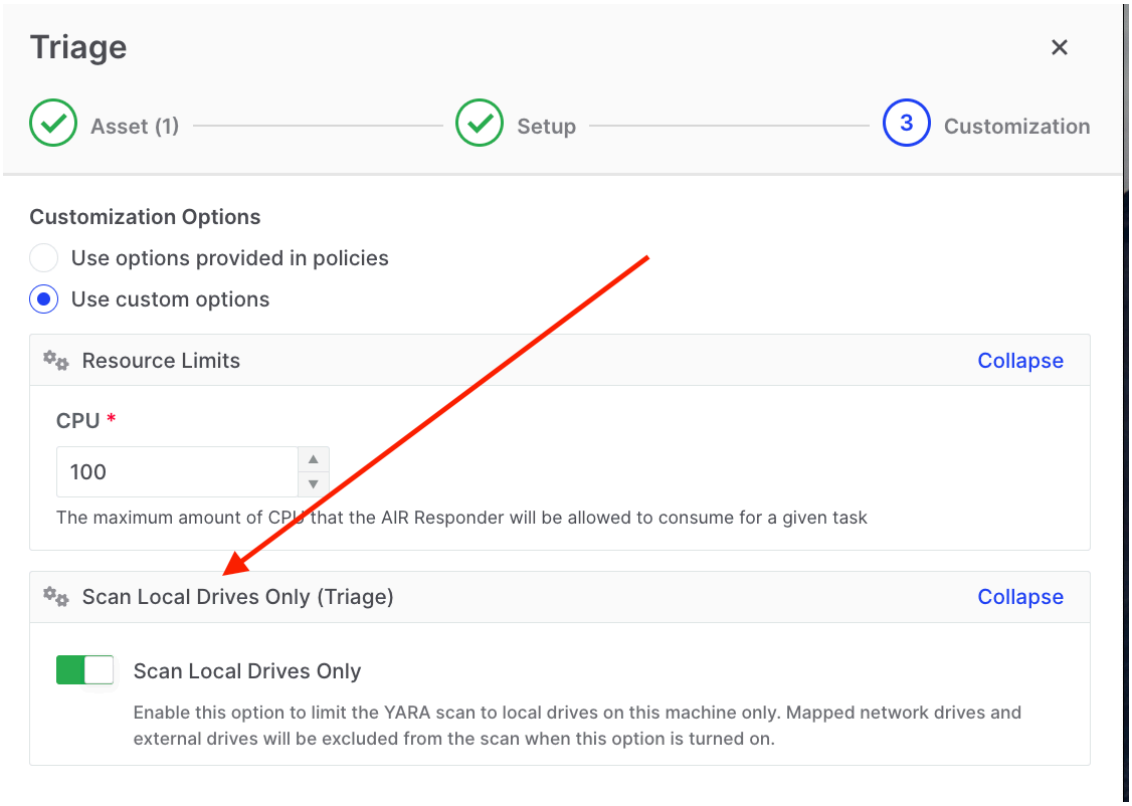
The Triage Rule Library includes Preset Filters in the secondary menu, allowing users to organize rules hierarchically, also known as '**Nested Tagging**'. By incorporating a colon in their tags, users can structure and categorize rules more efficiently. For example, the tag "APT26:Tim:hashset" helps organize related rules under a structured hierarchy, enhancing navigation and accessibility in the library.

Scan Local Drives Only for Triage Tasks

With the **Scan Local Drives Only** feature, users can improve Yara triage efficiency by focusing threat hunting and triage scans solely on local drives, excluding remote external or network drives that often introduce unnecessary data into the investigation. The attached mounted USB drives should be included as ‘local drives’.

Key Details:

- Available for all XDR Forensics-supported operating systems.
- Disabled by default, but can be toggled on or off via Settings > Policies > New Policies > Scan Local Drives Only. This setting can also be modified later.
- It can also be configured during the customization step when creating triage tasks via 'Use Custom Options'.



Triage: Custom option to scan local drives only

This feature ensures that only relevant data from local drives is collected, reducing noise and improving the speed and accuracy of investigations.

Schedule Triage Tasks

This page provides a guide of how users can schedule Triage tasks via the XDR Forensics API.

XDR Forensics Triage Scheduled Task via API Script by using crontab

- Move the script file to a directory, such as the **/opt** directory, as shown below.

  `mv air-triage-task-via-api.sh /opt/air-triage-task-via-api.sh`

- Update the console address and API Token value in the script. You must add the desired triage rule id values to the "triageRuleIds" field.

For example, there are two default rules below; you can change them.

"fireeye-red-team-tools-countermeasures", "fireeye-sunburst-countermeasures"

- Add it as a cronjob by running the command below.

  `crontab -e`

- After running the above command, add the following lines in the editor.

```
# At 00:00 on Sunday 0 0 * * 0 /opt/air-triage-task.sh
```

Triage Rule Templates

Here we provide some YARA, Sigma and osquery rule templates for users to copy and edit

- YARA Templates
- Sigma Templates
- osquery Templates

Sigma Templates

Selection of Sigma rules for use as guides or templates

- **Detection of Sysinternals Usage**

```
description: Detects the usage of Sysinternals Tools
tags:
    - attack.t1588.002
logsource:
    category: process_creation
    product: windows
detection:
    selection:
        CommandLine|contains: ' -accepteula'
    condition: selection
falsepositives:
    - Legitimate use of SysInternals tools
```

- **LSASS Dump Detection**

```
description: LSASS memory dump creation using operating systems
utilities.
tags:
    - attack.credential_access
logsource:
    category: file_event
    product: windows
detection:
    selection:
        TargetFilename|contains: 'lsass'
        TargetFilename|endswith: 'dmp'
    condition: selection
fields:
    - ComputerName
    - TargetFilename
falsepositives:
    - Admin activity
level: high
```

- **Suspicious Add Scheduled Task From User AppData Temp**

```
description: schtasks.exe create suspicious task from user
AppData\Local\Temp
tags:
    - attack.execution
    - attack.t1053.005
logsource:
    product: windows
    category: process_creation
detection:
    schtasks:
        Image|endswith: '\schtasks.exe'
    suspcommand:
        CommandLine|contains|all:
            - '/Create '
            - '\AppData\Local\Temp'
    condition: schtasks and suspcommand
falsepositives:
    - Unknown
level: high
```

- **Disable UAC Using Registry**

```
description: Detects Disable User Account Control (UAC) Using Registry
by changing its registry key
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System\EnableLUA
from 1 to 0
tags:
    - attack.privilege_escalation
    - attack.defense_evasion
    - attack.t1548.002
logsource:
    category: registry_set
    product: windows
detection:
    selection:
        EventType: SetValue
        TargetObject|contains:
SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System\EnableLUA
        Details: DWORD (0x00000000)
    condition: selection
falsepositives:
    - Unknown
level: medium
```

- **Windows Defender Service Disabled**

```
description: Detects disables the  Windows Defender service (WinDefend)
via the registry
tags:
    - attack.defense_evasion
    - attack.t1562.001
logsource:
    product: windows
    category: registry_set
detection:
    selection:
        EventType: SetValue
        TargetObject:
'HKLM\SYSTEM\CurrentControlSet\Services\WinDefend\Start'
        Details: 'DWORD (0x00000004)'
    condition: selection
falsepositives:
    - Administrator actions
level: high
```

- **PowerShell Get-Clipboard Cmdlet Via CLI**

```
description: Detects usage of the 'Get-Clipboard' cmdlet via CLI.
Adversaries may collect data stored in the clipboard from users copying
information within or between applications.
tags:
    - attack.collection
    - attack.t1115
logsource:
    category: process_creation
    product: windows
detection:
    selection:
        CommandLine|contains: 'Get-Clipboard'
    condition: selection
falsepositives:
    - Unknown
level: medium
```

- **User Account Hidden By Registry**

```
description: Detect modifications for a specific user in order to
prevent that user from appearing on the logon screen.
tags:
    - attack.defense_evasion
    - attack.t1564.002
logsource:
    product: windows
    category: registry_set
detection:
    selection:
        EventType: SetValue
        TargetObject|contains: '\SOFTWARE\Microsoft\Windows
NT\CurrentVersion\Winlogon\SpecialAccounts\Userlist\'
        TargetObject|endswith: '$'
        Details: DWORD (0x00000000)
    condition: selection
falsepositives:
    - Unknown
level: high
```

YARA Templates

Selection of YARA rules for use as guides or templates

File system only examples:

Find by Name

```
// Auto-Complete Support:
// Type modulename. followed by a CTRL + SPACE
// Yara documentation:
https://yara.readthedocs.io/en/stable/writingrules.html

rule find_by_name
{
    meta:
        description = "Find files by name."

    condition:
        file_name == "some-name.exe"
}
```

Find by Extension

```
// Auto-Complete Support:
// Type modulename. followed by a CTRL + SPACE
// Yara documentation:
https://yara.readthedocs.io/en/stable/writingrules.html

rule find_by_extension
{
    meta:
        description = "Find files by extension."

    condition:
        file_extension == "xyz"
}
```

Find by Content

```
// Auto-Complete Support:
// Type modulename. followed by a CTRL + SPACE
// Yara documentation:
https://yara.readthedocs.io/en/stable/writingrules.html

rule find_by_content
{
    meta:
        description = "Find files containing specific strings."

    strings:
        $a = "password" wide ascii nocase

    condition:
        $a
}
```

Find by Hash

```
// Auto-Complete Support:
// Type modulename. followed by a CTRL + SPACE
// Yara documentation:
https://yara.readthedocs.io/en/stable/writingrules.html

import "hash"

rule find_by_hash
{
    meta:
        description = "Find files by hash."

    condition:
        hash.sha256(0, filesize) ==
"b6800c2ca4bfec26c8b8553beee774f4ebab741b1a48adcccce79f07062977be"
}
```

Find by Size

```
// Auto-Complete Support:
// Type modulename. followed by a CTRL + SPACE
// Yara documentation:
https://yara.readthedocs.io/en/stable/writingrules.html

rule find_by_size
{
    meta:
        description = "Find files by size."

    condition:
        filesize < 1MB
}
```

Find by Size range

```
// Auto-Complete Support:
// Type modulename. followed by a CTRL + SPACE
// Yara documentation:
https://yara.readthedocs.io/en/stable/writingrules.html

rule find_by_size_range
{
    meta:
        description = "Find files in size range."

    condition:
        filesize > 100KB and filesize < 500KB
}
```

Find by Location

```
// Auto-Complete Support:
// Type modulename. followed by a CTRL + SPACE
// Yara documentation:
https://yara.readthedocs.io/en/stable/writingrules.html

rule find_by_location
{
    meta:
        description = "Find files in specific location."

    condition:
        file_path contains "Downloads" // when file path contains a
certain string
        or
        file_path == "C:\\Windows\\Temp\\svchost.exe" // for exact file
location
}
```

Find PE (portable executable) files only

```
// Auto-Complete Support:
// Type modulename. followed by a CTRL + SPACE
// Yara documentation:
https://yara.readthedocs.io/en/stable/writingrules.html

rule IsPE
{
    meta:
        description = "Identifies PE files only based on the header."

    condition:
        // MZ signature at offset 0 and ...
        uint16(0) == 0x5A4D and
        // ... PE signature at offset stored in MZ header at 0x3C
        uint32(uint32(0x3C)) == 0x00004550
}
```

Find PKZIP files only

```
// Auto-Complete Support:
// Type modulename. followed by a CTRL + SPACE
// Yara documentation:
https://yara.readthedocs.io/en/stable/writingrules.html

rule IsZIP
{
    meta:
        description = "Identifies ZIP files only based on the header."

    condition:
        uint32(0) == 0x04034B50
}
```

Find by Hash with Size filter

```
// Auto-Complete Support:
// Type modulename. followed by a CTRL + SPACE
// Yara documentation:
https://yara.readthedocs.io/en/stable/writingrules.html

// In order to make yara scan faster, it is always a good practice to
use filters.
// In this case let's say we know that sample is smaller than 1MB and we
want to search the hash.

import "hash"

rule find_by_hash
{
    meta:
        description = "Find files by hash."

    condition:
        filesize < 1MB and
        hash.sha256(0, filesize) ==
"b6800c2ca4bfec26c8b8553beee774f4ebab741b1a48adcccce79f07062977be"
}
```

Memory/process scan examples:

Find Process by Name

```
// Auto-Complete Support:
// Type modulename. followed by a CTRL + SPACE
// Yara documentation:
https://yara.readthedocs.io/en/stable/writingrules.html

rule find_process_by_name
{
    meta:
      description = "Find process by name."

    condition:
      process_name == "audiodg.exe"
}
```

Find String in Memory

```
// Auto-Complete Support:
// Type modulename. followed by a CTRL + SPACE
// Yara documentation:
https://yara.readthedocs.io/en/stable/writingrules.html

rule find_string_in_memory
{
    meta:
        description = "Find process executables containing string."

    strings :
        $a = "keylogger started" wide ascii nocase

    condition :
        $a
}
```

Find Process by Command line

```
// Auto-Complete Support:
// Type modulename. followed by a CTRL + SPACE
// Yara documentation:
https://yara.readthedocs.io/en/stable/writingrules.html

rule find_process_by_cmdline
{
    meta:
        description = "Find string in process command lines."

    condition :
        process_command_line icontains "powershell.exe" // icontains is
for case insensitive
}
```

Find Malware domain

```
// Auto-Complete Support:
// Type modulename. followed by a CTRL + SPACE
// Yara documentation:
https://yara.readthedocs.io/en/stable/writingrules.html

rule find_malware_domain
{
    meta:
        description = "Search malware domain in process memory."

    strings:
        $a = "http://malware-domain.com" wide ascii

    condition:
        $a
}
```

Find Byte pattern

```
// Auto-Complete Support:
// Type modulename. followed by a CTRL + SPACE
// Yara documentation:
https://yara.readthedocs.io/en/stable/writingrules.html

rule find_byte_pattern
{
    meta:
        description = "Search byte pattern process memory."

    strings:
        $a = { AA BB CC DD EE FF }

    condition:
        $a
}
```

Filesystem and memory scan:

Find String

```
// Auto-Complete Support:
// Type modulename. followed by a CTRL + SPACE
// Yara documentation:
https://yara.readthedocs.io/en/stable/writingrules.html

rule find_string
{
    meta:
        description = "Find containing string."

    strings :
        $a = "keylogger started" wide ascii nocase

    condition :
        $a
}
```

Find Malware domain

```
// Auto-Complete Support:
// Type modulename. followed by a CTRL + SPACE
// Yara documentation:
https://yara.readthedocs.io/en/stable/writingrules.html

rule find_malware_domain
{
    meta:
        description = "Search malware domain."

    strings:
        $a = "http://malware-domain.com" wide ascii

    condition:
        $a
}
```

Find Byte pattern

```
// Auto-Complete Support:
// Type modulename. followed by a CTRL + SPACE
// Yara documentation:
https://yara.readthedocs.io/en/stable/writingrules.html

rule find_byte_pattern
{
    meta:
        description = "Search byte pattern process memory."

    strings:
        $a = { AA BB CC DD EE FF }

    condition:
        $a
}
```

Find XOR pattern

```
// Auto-Complete Support:
// Type modulename. followed by a CTRL + SPACE
// Yara documentation:
https://yara.readthedocs.io/en/stable/writingrules.html

rule find_xor_string
{
    meta:
        description = "Search xor string pattern."

    strings:
        $xor_string = "This program cannot" xor

    condition:
        $xor_string
}
```

Find Base64 pattern

```
// Auto-Complete Support:
// Type modulename. followed by a CTRL + SPACE
// Yara documentation:
https://yara.readthedocs.io/en/stable/writingrules.html

rule find_base64_string
{
    meta:
        description = "Search Base64 encoded string pattern."

    strings:
        $mimi = "Mimikatz" ascii wide base64 base64wide

    condition:
        $mimi
}
```

osquery Templates

Selection of osquery rules for use as guides or templates

UAC_disabled

```
-- platform: windows
-- description: Controls UAC. A setting of 0 indicates that UAC is
disabled.
SELECT *
FROM registry
WHERE
path='HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Polic
ies\System\EnableLUA' AND data=0;
```

Windows Update history

```
-- platform: windows
-- description: List Windows Update history.
select title, datetime(date, 'unixepoch', 'localtime')
from windows_update_history;
```

Registry Run entries

```
-- platform: windows
-- description: List startup entries under Run keys.
select *
from registry
where key like
'HKEY_USERS\%\SOFTWARE\Microsoft\Windows\CurrentVersion\Run'
or key like
'HKEY_USERS\%\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnce'
or key like
'HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run%\%';
```

Services that start automatically

```
-- platform: windows
-- description: List installed services that start automatically
SELECT name,display_name,user_account,path
FROM services
WHERE start_type = 'AUTO_START'
      AND path NOT LIKE 'C:\Windows\system32\svchost.exe -k %';
```

Unusual Cron entries

```
-- Unexpected crontab entries
--
-- references:
--   * https://attack.mitre.org/techniques/T1053/003/ (Scheduled
Task/Job: Cron)
--   * https://github.com/chainguard-dev/osquery-defense-
kit/blob/main/detection/persistence/unexpected-cron-entries.sql
--
-- false positives:
--   * crontab entries added by the user
--
-- tags: persistent filesystem state
-- platform: posix
SELECT
  *
FROM
  crontab
WHERE
  command NOT LIKE 'root%run-parts%'
  AND command NOT LIKE '%freshclam%'
  AND command NOT LIKE '%clamscan%'
  AND command NOT LIKE '%e2scrub%'
  AND command NOT LIKE '%zfs-linux%'
  AND command NOT LIKE '%anacron start%'
  AND command NOT LIKE '%/usr/lib/php/sessionclean%'
  AND command NOT LIKE 'root command -v debian-sa1%'
```

Launched items not signed by Apple

```
-- description: Find launchd entries which purport to be by Apple, but
point to binaries that are not signed by Apple.
--
-- references:
--   * https://attack.mitre.org/techniques/T1543/004/ (Create or Modify
System Process: Launch Daemon)
--   * https://posts.specterops.io/hunting-for-bad-apples-part-1-
22ef2b44c0aa
--   * https://github.com/chainguard-dev/osquery-defense-
kit/blob/main/detection/persistence/fake-apple-launchd.sql
--
-- false positives:
--   * none have been observed
--
-- platform: darwin
-- tags: persistent launchd state
SELECT
  *
FROM
  launchd
  LEFT JOIN file ON launchd.path = file.path
  LEFT JOIN signature ON launchd.program_arguments = signature.path
WHERE
  launchd.name LIKE 'com.apple.%'
  -- Optimization, assumes SIP
  AND file.directory NOT IN (
    '/System/Library/LaunchAgents',
    '/System/Library/LaunchDaemons',
    '/Library/Apple/System/Library/LaunchDaemons',
    '/Library/Apple/System/Library/LaunchAgents'
  )
  AND launchd.run_at_load = 1
  AND signature.authority != 'Software Signing'
```

Processes running no binary on the disk

```
-- description: Find processes that are running whose binary has been
deleted from the disk.
SELECT name, path, pid FROM processes WHERE on_disk = 0;
```

Scheduled Task with Temp path reference

```
-- description: List scheduled tasks where Temp directory is contained
in Action path.
SELECT name, action FROM scheduled_tasks WHERE action LIKE '%\Temp\%';
```

List all local Users

```
-- description: List all local Users on the system.
select * from users where type = 'local';
```

List logged users

```
-- description: List logged users.
select * from logged_in_users;
```

List users with Administrative privileges

```
-- description: List all the users with Administrative privileges.
select users.uid,users.gid,users.username,users.directory from users
JOIN user_groups ON users.uid=user_groups.uid where user_groups.gid=544;
```

Check the security status of the system

```
-- description: Check the security status of the system.
select * from windows_security_center;
select * from windows_security_products;
```

List processes running from CMD (with hash value)

```
-- description: List processes running from cmd (with a hash value)
select p.name,p.path,p.pid,p.parent,h.md5,pp.path as parentpath from
processes p JOIN hash h on p.path=h.path JOIN processes pp ON
p.parent=pp.pid where pp.path like '%cmd%';
```

Troubleshooting

Understanding MSI Error Code 1618

This article explains the reasons behind MSI Error Code 1618 and provides practical steps to resolve it.

Overview

MSI Error Code 1618 is a common issue encountered during software installations or updates on Windows systems.

This error indicates that another MSI-based installation is already in progress, which blocks the current installation or update from proceeding. Although this error can occur while updating or operating the **XDR Forensics Responder**, it is not caused by the XDR Forensics Responder software itself.

Why Does This Error Occur?

The error code is related to the Microsoft Windows Installer service. It typically appears when:

- Another installation is already running in the background (e.g., a Windows update or another MSI-based installer).
- The Windows Installer service is locked by a pending installation or system process.

Since this issue originates from the Windows environment, resolving it involves managing the underlying MSI processes.

Steps to Resolve MSI Error Code 1618

1. **Check for Running Installations**

- Manually verify if there are any ongoing installations:
- Open **Task Manager** (Ctrl + Shift + Esc).
- Look for any processes related to Windows Installer or other installation programs (e.g., `msiexec.exe`).
- End these processes if they are blocking the current installation.\

2. **Check for Pending Windows Update**s

Sometimes, a pending or ongoing Windows update can cause this error. Follow these steps:

- Go to **Settings > Update & Security > Windows Update.**
- Check if there are any updates in progress or requiring a restart to complete.
- Allow the updates to finish or reboot the machine to clear them.

3. **Restart the Windows Installer Service**

Restarting the Windows Installer service can resolve locked or unresponsive installer issues:

1. Open the Run dialog box (`Win + R`).
2. Type `services.msc` and press Enter.
3. Locate the **Windows Installer** service (`msiserver`).
4. Right-click and select **Restart**.

4. Reboot the System

The simplest and most effective first step is to reboot the affected machine. Rebooting clears any pending installations or processes using the Windows Installer service.

Additional Notes

You can also refer to Microsoft's official support page for more details:

https://answers.microsoft.com/en-us/windows/forum/all/error-code-1618-when-trying-to-install-new/fb4e98b3-59d9-4cfd-916c-215502864bce ↗

Need Further Assistance?

If you continue experiencing issues or have additional questions, please contact our support team: `https://www.cisco.com/c/en/us/support/web/tsd-cisco-worldwide-contacts.html`

If you experience this error while updating or operating the **XDR Forensics Responder**, it is not caused by the XDR Forensics Responder software itself.

How to gather logs for Troubleshooting

The XDR Forensics **Console** and the XDR Forensics **Responder** generate the activities log, errors log, and warnings logs. These logs can be used to investigate and solve problems both by the users and the XDR Forensics Support Team. The log files are stored in separate files on the Console and Asset machines.

Investigators and analysts can **download** these log files either by using the **XDR Forensics Console** user interface or by **connecting to the console/asset** machines directly (find more details below).

⚠ Be sure to note that you have the option to collect logs from **two locations**:

1. **The XDR Forensics Console machine and,**
2. **XDR Forensics responders on your assets**

Collecting Responder Log Files >

Collecting Off-Network Responder Log Files >

Collecting Responder Log Files

XDR Forensics Responder Log Files

XDR Forensics Responder categorizes and stores the log files as nine separate files as listed below. All associated log records are stored in the related log file.

- TACTICAL.Log.txt
- TACTICAL.Process.Log.txt
- TACTICAL.Error.txt
- AIR.Log.txt
- AIR.Process.Log.txt
- DRONE.log
- DRONE.Process.log
- WATCHDOG.Process.Log.txt
- WATCHDOG.Log.txt

The log files that are generated by XDR Forensics responders are stored under the directory that is given below.

Windows	C:\Program Files\Cisco\Forensics\AIR
Linux	/opt/cisco/forensics/air
macOS	/opt/cisco/forensics/air

By using the command line interface

1. Log in directly or connect remotely to the asset that XDR Forensics responder is installed on by the appropriate remote device management tool
2. Browse to the directory which is mentioned above according to the associated operating system
3. Download the files or view the contents of the files with relevant tools.

By using the user interface

1. Select the Assets button on the left of the main console menu
2. Select the asset from which XDR Forensics responder logs are required
3. Select 'Logs' from the bottom of the secondary menu
4. Click on the 'Collect Logs' icon in the main Assets Logs page

This action creates a Task for collecting logs. After this log retrieval task is finished, the Task status will be changed to Completed, and it can be downloaded by clicking the icon on the right side of the green Completed bar. All available log files will be compressed as a single zip file and can be downloaded.

The Log Retrieval tasks can also be accessed in the Tasks section.

Collecting Off-Network Responder Log Files

The Off-Network XDR Forensics responder categorizes and stores log files in two locations:

1. At the root of the directory from which the XDR Forensics Off-Network responder is executed.
2. In the 'bin' directory which is also found at the root of the directory from which the XDR Forensics Off-Network responder is executed.

At the root of the directory from which the XDR Forensics Off-Network responder is executed, users will find the following log files:

- OFFNETWORK_WINDOWS_AMD64.Log.txt
- OFFNETWORK_WINDOWS_AMD64.Process.Log.txt
- troubleshoot-[TIMESTAMP].zip

In the 'bin' directory which is also found at the root of the directory from which the XDR Forensics Off-Network responderis executed, users will find the following log files:

- TACTICAL-Legacy.Log.txt
- TACTICAL.Log.txt
- TACTICAL.Process.Log.txt
- TACTICAL.Error.txt
- AIR.Log.txt
- AIR.Process.Log.txt
- DRONE.log.txt
- DRONE.Process.log.txt

NOTE: With XDR Forensics v4.4 (responder v2.30) and later, the 'troubleshoot-[TIMESTAMP].zip' will always be generated, even if there have been no errors and this file will consolidate all of the other log files shown on this page. This is to make it simple for users to collect and send log files to support if required.

FAQs

- Investigation Hub
- Responder troubleshooting
- How to gather logs for Troubleshooting
- Understanding Port Usage
- How many assets can connect to a single Console instance?
- How to download the collected evidence and artifacts?
- How do I enable SSL on Console?
- Can I use XDR Forensics with EDR/XDR Products?
- Can I integrate XDR Forensics with my SOAR/SIEM?
- Docker & Host System IP Conflict
- Monitoring Responder and UI API's
- How do I update Responders on assets?

Is there a way to move an asset from one Organization or Case to another? >

Creating exclusions/exception rules for Responder on EPP and EDR Solutions >

How to download the collected evidence and artifacts?

Introduction

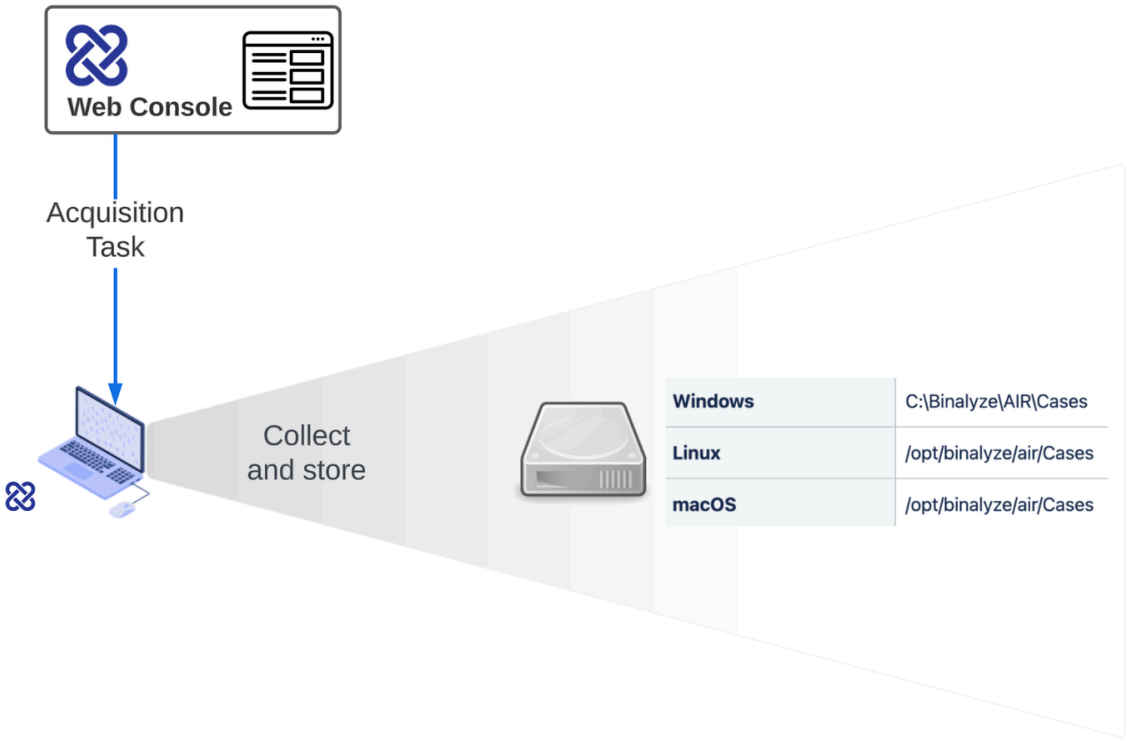
XDR Forensics provides two separate mechanisms for storing collected evidence, artifacts, and data. The first and default method is to use the local storage of the asset and the server machines' file system. The second method utilizes remote network services, which we refer to as Evidence Repositories, including SFTP, FTPS, SMB, Amazon S3 Buckets, and Azure Blob Storage.

The location of the evidence and artifacts storage is defined in the Policy section of the XDR Forensics Console. Since the Acquisition tasks are directly bound to these policies, investigators and analysts can change the type (Local or Evidence Repository) and path of the collected evidence and artifacts by using the options fields in the Policy screen.

The exact location and path of the evidence and artifacts can be viewed through the **Evidence URL** value, which is located under the **Metadata** page that is accessible via a button shown under the **Status** of the associated **Task**.

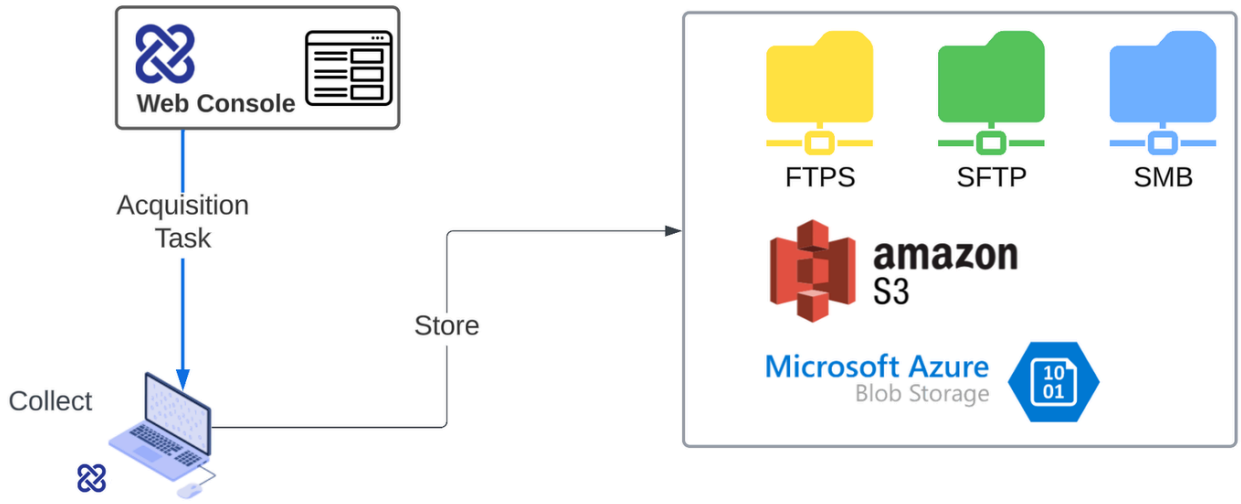
The evidence and artifacts collection and storage flow is summarized as follows.

1. An Acquisition Task is created on the XDR Forensics Console.
2. The XDR Forensics responder installed on asset machines connects to the XDR Forensics Console to retrieve the task and task details.
3. The XDR Forensics responder installed on asset machines runs the Acquire Task on the asset, collecting evidence and artifacts.
4. XDR Forensics responder stores the collected evidence and artifacts according to the configuration options defined in the Policies section.
    a. If the Local option is selected, the collected evidence and artifacts will be stored in the file system location defined in the configuration options.



How to download the collected evidence and artifacts: Fig1

    b. If the Evidence Repository option is selected, the collected evidence and artifacts will be stored in a temporary location on the local file system. Then, the asset machine directly connects to the remote network service/server and uploads the evidence and artifacts to the location defined in the configuration options. All the evidence and artifacts will be deleted from the asset. The flow is depicted in the picture below.



How to download the collected evidence and artifacts: Fig2

Download The Evidence and Artifacts from Local Storage

Two separate methods can be used to download collected evidence and artifacts, which are stored in the local file system of the asset machines. Investigators and analysts can use XDR Forensics built-in remote management tool, interACT, or their favorite remote management tool to connect and download the requested evidence and artifacts.

The default location of all collected evidence and artifacts are listed below according to the operating systems. The default location can be changed by editing the policies.

Windows	C:\Cisco\Forensics\AIR\Cases
Linux	/opt/cisco/forensics/air/Cases
macOS	/opt/cisco/forensics/air/Cases

Using interACT

1. Find the task of interest.
   The relevant task can be viewed either by navigating through the **Global Tasks Tab** or by selecting the associated assets from the asset listings and then locating the related task under the asset details page.
2. Locate the exact path of the evidence and artifacts that require downloading. Click the **Metadata** button under this **Status** on the page and view the value of the **Evidence URL.** It is also possible to directly copy the value of the Evidence URL by clicking the copy button, which is located at the beginning of the Evidence URL line.
3. Connect to the machine using interACT by clicking the interACT button, located under the asset details page.
4. Navigate to the path provided with the Evidence URL by using the **cd** command.
5. Download the associated file by using the **get** command provided by the interACT.

By Using Remote Management tools

1. Find the task of interest.
   The relevant task can be viewed either by navigating through the **Global Tasks Tab** or by selecting the associated assets from the asset listings and then locating the related task under the asset details page.
2. Locate the exact path of the evidence and artifacts that require downloading. Click the **Metadata** button under this **Status** on the page and view the value of the **Evidence URL.** It is also possible to directly copy the value of the Evidence URL by clicking the copy button, which is located at the beginning of the Evidence URL line.
3. Connect to the asset with your favorite remote management tool. This tool may vary depending on the operating system installed on the asset. The most commonly used tools include SSH, Remote Desktop Manager, and VNC, among others.
4. Navigate to the path provided with the Evidence URL by using the **cd** command.
5. Download the associated file by using the commands and activities provided by the remote management tool.

Download The Evidence and Artifacts from Evidence Repositories

1. Find the task of interest.
   The relevant task can be viewed either by navigating through the Global Tasks Tab or by selecting the associated assets from the asset listings and then locating the related task under the asset details page.
2. Download the compressed/encrypted zip file by clicking the **Evidence URL**
   The downloadable link will be created and bonded directly to the **Evidence URL.** Click the **Metadata** button under this **Status** on the page, and then click on the Evidence URL to download the zip file, which includes the chosen evidence and artifacts.

How to gather logs for Troubleshooting

The XDR Forensics **Console** and the XDR Forensics **Responder** generate the activities log, errors log, and warnings logs. These logs can be used to investigate and solve problems both by the users and the Cisco XDR Forensics Support Team. The log files are stored in separate files on the Console and Asset machines.

Investigators and analysts can **download** these log files either by using the **XDR Forensics Console** user interface or by **connecting to the console/asset** machines directly (find more details below).

System Administrators must collect both different log files. Log files are located on:

1. **The XDR Forensics Console machine and,**
2. **XDR Forensics responders on your assets**

⚠ Be sure to note that you have the option to collect logs from **two locations**:

1. **The XDR Forensics Console machine and,**
2. **XDR Forensics responders on your assets**

Collecting Responder Log Files >

Collecting Off-Network Responder Log Files >

Collecting Responder Log Files

XDR Forensics Responder Log Files

XDR Forensics Responder categorizes and stores the log files as nine separate files as listed below. All associated log records are stored in the related log file.

- TACTICAL.Log.txt
- TACTICAL.Process.Log.txt
- TACTICAL.Error.txt
- AIR.Log.txt
- AIR.Process.Log.txt
- DRONE.log
- DRONE.Process.log
- WATCHDOG.Process.Log.txt
- WATCHDOG.Log.txt

The log files that are generated by XDR Forensics responders are stored under the directory that is given below.

Windows	C:\Program Files\Cisco\Forensics\AIR
Linux	/opt/cisco/forensics/air
macOS	/opt/cisco/forensics/air

By using the command line interface

1. Log in directly or connect remotely to the asset that XDR Forensics responder is installed on by the appropriate remote device management tool
2. Browse to the directory which is mentioned above according to the associated operating system
3. Download the files or view the contents of the files with relevant tools.

By using the user interface

1. Select the Assets button on the left of the main console menu
2. Select the asset from which XDR Forensics responder logs are required
3. Select 'Logs' from the bottom of the secondary menu
4. Click on the 'Collect Logs' icon in the main Assets Logs page

This action creates a Task for collecting logs. After this log retrieval task is finished, the Task status will be changed to Completed, and it can be downloaded by clicking the icon on the right side of the green Completed bar. All available log files will be compressed as a single zip file and can be downloaded.

The Log Retrieval tasks can also be accessed in the Tasks section.

Collecting Off-Network Responder Log Files

The Cisco Off-Network XDR Forensics responder categorizes and stores log files in two locations:

1. At the root of the directory from which the XDR Forensics Off-Network responder is executed.
2. In the 'bin' directory which is also found at the root of the directory from which the XDR Forensics Off-Network responder is executed.

At the root of the directory from which the XDR Forensics Off-Network responder is executed, users will find the following log files:

- OFFNETWORK_WINDOWS_AMD64.Log.txt
- OFFNETWORK_WINDOWS_AMD64.Process.Log.txt
- troubleshoot-[TIMESTAMP].zip

In the 'bin' directory which is also found at the root of the directory from which the XDR Forensics Off-Network responders executed, users will find the following log files:

- TACTICAL-Legacy.Log.txt
- TACTICAL.Log.txt
- TACTICAL.Process.Log.txt
- TACTICAL.Error.txt
- AIR.Log.txt
- AIR.Process.Log.txt
- DRONE.log.txt
- DRONE.Process.log.txt

NOTE: With XDR Forensics v4.4 (responder v2.30) and later, the 'troubleshoot-[TIMESTAMP].zip' will always be generated, even if there have been no errors and this file will consolidate all of the other log files shown on this page. This is to make it simple for users to collect and send log files to support if required.

Responder troubleshooting

This article explains how to troubleshoot XDR Forensics responder problems.

Installation Problems

Responder installer name contains version information and IP address fields. Make sure that the MSI installer file name doesn't change and has relevant fields before installation. You can see an example below:

> AIR.Responder_2.38.7_air-demo.ACME.com_176_9df51c56a73341f4_386_.msi

Post-Installation Problems

Default installation path is "C:\Program Files\Cisco\Forensics\AIR". Check this folder to ensure it contains the following responder files:

> TACTICAL.exe
>
> TACTICAL.Log.txt

Service Problems

XDR Forensics responder uses "XDR Forensics Responder Service". Check this service from Windows Services Manager and make sure it exists and its state is "running".

Uninstall Problems

If system time and date are not set correctly or the necessary permissions are not set on %temp% and "C:\Windows\Installer" folders, you may encounter "Called RunScript when not marked in progress" and/or "Called InstallFinalize when no install in progress" errors during uninstall process.

First, check the system time/date and make sure it's correct.

Second, check the security properties of %temp% and "C:\Windows\Installer" folders and make sure that "System" and "Everyone" users have "Full control" permission over these folders.

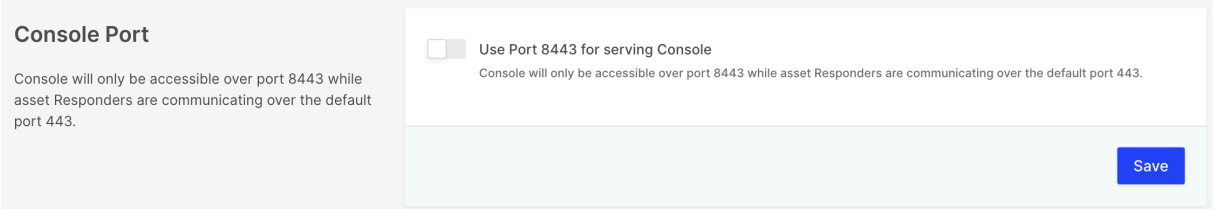
Understanding Port Usage

What ports are used by the platform?

XDR Forensics uses specific ports to manage communication between the XDR Forensics Console, responders, and other components of the system. Ensuring that the correct ports are open and configured can prevent connectivity issues and optimize performance. Below is a breakdown of the key ports used by XDR Forensics and their purposes.

Key Ports Used in XDR Forensics

- **TCP 443**: This is the default port for most XDR Forensics communications, including:
  - **XDR Forensics User → Console**: Used for users accessing the XDR Forensics Console.
  - **Asset → Console**: Used by responders on assets to communicate with the console. This is the default and preferred port, in some cases, users may wish to switch to 8443:



Understanding Port Usage: Console Port

  - **Responder Downloads**: The responder download links (on the deploy page) are accessible via port 443. Using this port ensures consistent download access, especially in environments where only port 443 is available.
- **TCP 8443**: This is an alternative port used for user access to the console, as well as other specific functions, including:
  - **XDR Forensics User → Console**: Optional port for user access to the XDR Forensics Console.
  - **Shareable Deploy Page**: The shareable page for responder deployment is available on this port.
  - **Off-Network Tasks**: The download links for tasks that run off-network are accessible on this port.
  - **Admin Portal**: The administrative portal operates on this port.
  - **REST API**: API calls to the XDR Forensics Console (`api/public/*`) are handled on this port.
  - **Azure and Okta SSO**: If you're using Azure or Okta Single Sign-On (SSO), the callback from within the browser should happen over port 8443.
- **TCP 4222**: This port is used for real-time task pushes to assets. If real-time communication is needed for task assignment, this port should be enabled.
- **TCP/UDP 389 and 636**: These ports are optional and used when Active Directory (AD) integration is enabled:
- **389**: For LDAP (Lightweight Directory Access Protocol) communication.
- **636**: For LDAPS (LDAP over SSL) communication.

- **TCP/UDP 514**: This is the optional port used when Syslog integration is enabled. Syslog helps in forwarding system logs to a centralized log server.

Recommended Usage

While port 8443 can be used manually for downloading responders and accessing the console, we **strongly recommend** using port 443 for the following reasons:

1. **Consistency**: Port 443 is universally available across most environments and networks, reducing the risk of connectivity issues.
2. **Responder Communication**: Responders may not have access to port 8443 in certain configurations, making port 443 the preferred choice for ensuring reliable responder-console communication.

Summary of Port Functions:

Port	Function
TCP 443	Default for user and responder communication with XDR Forensics Console
TCP 8443	Optional for user access, API, and other console functionalities
TCP 4222	Enables real-time task pushes to assets
TCP/UDP 389/636	LDAP and LDAPS for Active Directory integration
TCP/UDP 514	Syslog port for logging

By ensuring that these ports are correctly configured and open, you can optimize communication between the XDR Forensics Console and responders, allowing for seamless operation.

How many assets can connect to a single Console instance?

Connecting assets to a single Console Instance

XDR Forensics responders deployed to your assets are passive responders and a single basic install of the XDR Forensics Console has been tested on networks with up to 25,000 assets.

XDR Forensics is designed to deal with many more connected assets, so if you are planning to install XDR Forensics on a bigger network, please reach out to us https://www.cisco.com/c/en/us/support/web/tsd-cisco-worldwide-contacts.html.

How do I enable SSL on Console?

Enabling SSL on XDR Forensics

XDR Forensics uses HTTP as its default protocol. You can enable SSL by visiting Settings > SSL section.

Once Use SSL setting is enabled both Console and Endpoints will start using HTTPS port as their default protocol.

Enabling SSL will automatically redirect all HTTP requests to HTTPS.

Can I use XDR Forensics with EDR/XDR Products?

Using XDR Forensics with EDR/XDR Products

The level of forensic information XDR Forensics provides is the biggest differentiator that separates it from the rest of the crowd. This fact makes XDR Forensics a perfect candidate for using it side-by-side with an EDR/XDR product.

Here are some EDR/XDR use-case examples:

- Eliminating false positives by providing analysts with XDR Forensics reports,
- Investigating pre-cursors,
- Enriching an alert,
- Responding to EDR/XDR alerts automatically.

If you use an EDR/XDR or EPP software along with Cisco, check our **exclusion/exception rules** page.

Can I integrate XDR Forensics with my SOAR/SIEM?

Integrating XDR Forensics with SOAR/SIEM

XDR Forensics can be triggered by your SIEM/SOAR product without human intervention. This makes it a perfect match for responding to alerts you receive from these solutions.

Communication with SIEM products is bi-directional. So, XDR Forensics not only receives alerts/triggers from your SIEM but also reports the actions it performed back to it via Syslog Protocol.

Docker & Host System IP Conflict

Docker uses the default **172.17. 0.0/16** subnet for container networking. If your host system is in this subnet block, you will experience an IP conflict when the docker service or a container is started. To avoid IP conflict, you can change the default docker subnet by modifying the **/etc/docker/daemon.json** file:

- First, check if /etc/docker/daemon.json exists, if not, create it:

```
touch /etc/docker/daemon.json
```

- Modify daemon.json file with a preferred text editor

```
sudo nano /etc/docker/daemon.json
```

Paste the following lines:

```
{ "default-address-pools": [ {"base":" 10.10.0.0/16 ↗ ","size":24} ] }
```

- Save the file by clicking “ctrl+x” and then clicking “y” and finally pressing the “enter".
- Restart docker service by executing the following command:

```
sudo service docker restart
```

- Check the IP address of the docker0 interface:

```
ifconfig docker0
```

- The output should look like this:

docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500 **inet 10.10.0.1 netmask 255.255.0.0** broadcast 10.10.255.255 ether AA:BB:CC:DD:EE:FF txqueuelen 0 (Ethernet) RX packets 0 bytes 0 (0.0 B) RX errors 0 dropped 0 overruns 0 frame 0 TX packets 0 bytes 0 (0.0 B) TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

Monitoring Responder and UI API's

Responder API

Send a HTTP GET request to **https*:///[XDR Forensics_SERVER_ADDRESS]/api/app/check***

The app check URL doesn't require any authentication.

A successful response would be 200 OK, and the response body will be like this:

```
{
"success": true
}
```

UI API

Send a HTTP GET request to **https://[XDR Forensics_SERVER_ADDRESS]:8443/api/app/check**

The app check URL doesn't require any authentication.

A successful response would be 200 OK, and the response body will be like this:

```
{
"success": true
}
```

Note: It is highly recommended to check the response body along with the HTTP status code for both the responder *and the UI API's.*

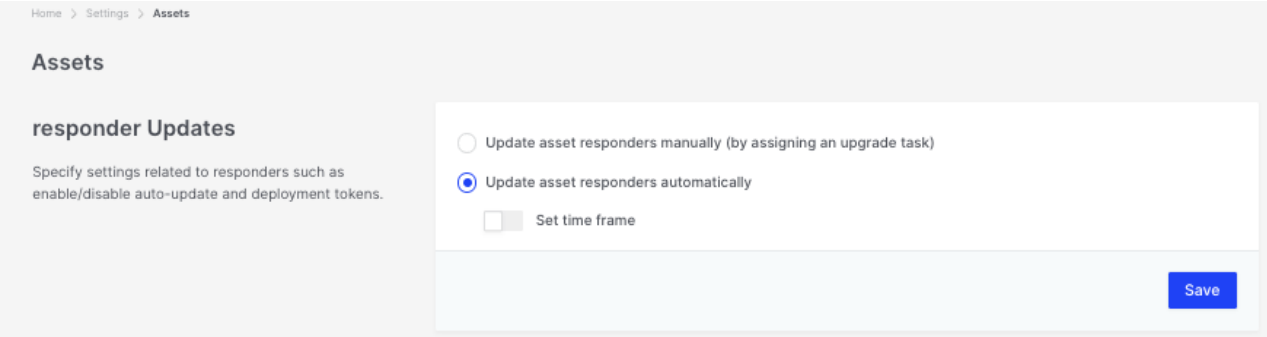
How do I update Responders on assets?

Updating XDR Forensics Responders

The XDR Forensics Responder can be updated in two ways: manually or automatically and both options can be scheduled. By default, updates are set to be manual. However, if XDR Forensics Responders on assets maintain a connection to the console from which they were deployed, they are capable of updating themselves automatically. Details on both update options are explained below.

Automatic Responder update

Once a Console update is installed, its associated Responders will automatically receive and execute an update task during their next connection to the updated console. It's important to note that this automatic update behavior is only active if the 'Update asset Responders automatically' setting is enabled, as illustrated in the screenshot below.



How do I update Responders on assets: Settings

Manual Responder update

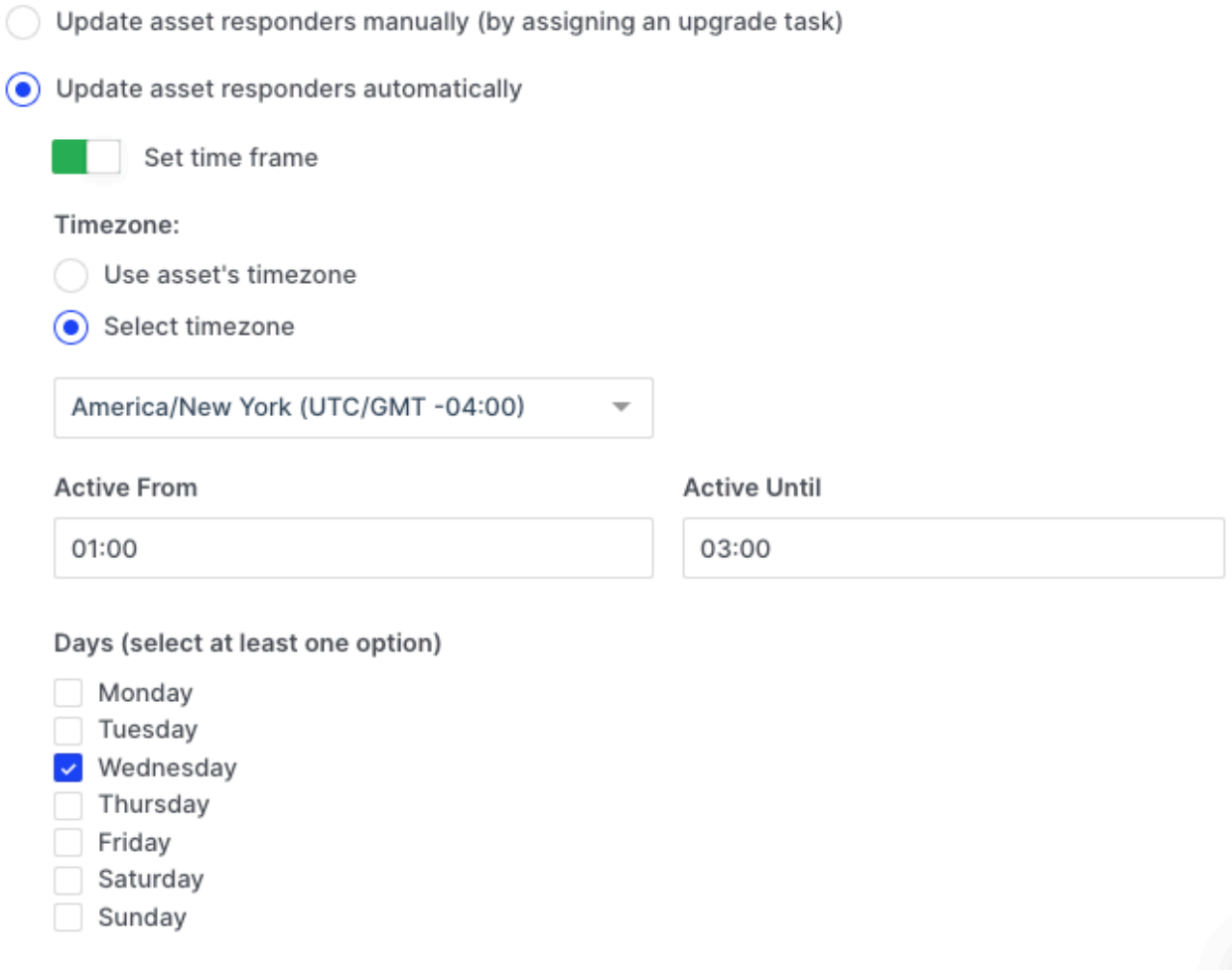
If the setting selected is, 'Update asset Responders manually (by assigning an upgrade task)' the update task can be generated from the individual Asset page:

You can also manually update Responders by selecting multiple assets. If any of the selected assets require a Responder update, the option to do so will be available to the user through the 'More' option in the Bulk Action bar.

Scheduling options for Responder updates

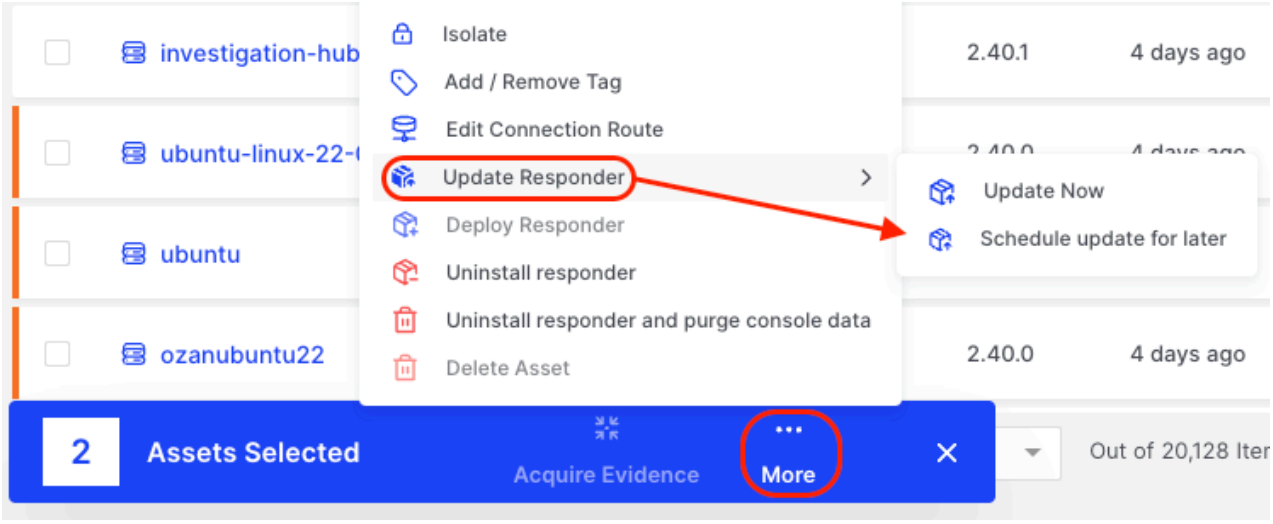
Scheduling options for Responder updates to streamline the process, ensuring that updates do not delay or disrupt ongoing investigations.

- **Scheduled Manual Updates:** Users can schedule updates for a specific time for an individual asset or a group of assets. Setting a new update time for an asset will override any previously scheduled time for that asset.
- **Scheduled Auto Updates:** Users can establish a recurring schedule to automatically check for and apply updates within a designated timeframe.
- The Assets > Settings page allows selection, timezone, times, and days of the week for the task to be executed.



How do I update Responders on assets: Schedule Responder updates

- Alternatively, assets can be scheduled for responder updates either through the individual Asset action button or by using the bulk actions bar.



How do I update Responders on assets: Schedule Responder update for later

How to reset the password of a user via the CLI?

Is there a way to move an asset from one Organization or Case to another?

Moving an asset from one Organization to another is prohibited.

Organization assignment is performed on the installation of the XDR Forensics responder process, and assets are directly bonded to the associated Organization. So there is no way to directly move an asset from one Organization to another. If you want to move an asset from one Organization, you must uninstall the responder from the asset, and then you must create a new responder deployment package and must select the Organization you want.

Moving an asset directly from one Case to another is not a valid action.

Actually, assets are not directly attached to Cases. They attach to a Case via a Task. When the Tasks of the assets are added to the Cases, the assets appear in the Case. An asset can be included in more than one Case. When you add the asset's Tasks to different Cases, the asset appears in those Cases. You can add Tasks to a Case or send a completed Task to other Cases with the "Send to case" action.

If you remove all the Tasks of the asset from the Case, the asset will be removed. You can directly remove the asset from the Case. This way, all its Tasks will be removed from the Case. Then you can add Tasks in interest to any Case you want.

Creating exclusions/exception rules for Responder on EPP and EDR Solutions

It’s common for anti-virus, EPP, and EDR (Endpoint Detection and Response) solutions to utilize exception rules to avoid unintentionally blocking important files or activities necessary for normal business operations.

These rules act as **exclusions**, allowing specific files, processes, or activities to **bypass the security software's detection or blocking mechanisms.** This is necessary in cases such as false-positive alerts triggered by (a) a **legitimate application** that may resemble malware or (b) a **critical system file** that is falsely flagged as malicious by security software.

To ensure proper functionality, the XDR Forensics responder uses **distinct executables** for different tasks, all of **which must be excluded by associated security solutions**. XDR Forensics offers **folder-level exception rules** exclusively for the XDR Forensics responder folder since different security solutions have varying exception mechanisms. See below for the operating system-specific full paths to the XDR Forensics responder folders.

Windows

Folders to Exclude:

- `C:\Program Files\Cisco\Forensics\AIR\`
- `C:\ProgramData.air`

Binaries to Exclude:

- `C:\Program Files\Cisco\Forensics\AIR\\AIR.exe`
- `C:\Program Files\Cisco\Forensics\AIR\\DRONE.exe`
- `C:\Program Files\Cisco\Forensics\AIR\\Tactical.exe`
  `%ProgramData%.air\WATCHDOG.exe`
- `C:\Program Files\Cisco\Forensics\AIR\\utils\curl.exe`
- `C:\Program Files\Cisco\Forensics\AIR\\utils\osqueryi.exe`

Linux

Folders to Exclude:

- `/opt/cisco/forensics/air/air`
- `/usr/share/.air/`

Binaries to Exclude:

- `/opt/cisco/forensics/air/air`
- `/opt/cisco/forensics/air/drone`
- `/opt/cisco/forensics/air/tactical`
- `/opt/cisco/forensics/air/osqueryi`
- `/opt/cisco/forensics/air/curl`
- `/usr/share/.air/watchdog`

macOS

Folders to Exclude:

- `/opt/cisco/forensics/air/`
- `/usr/local/share/.air/`

Binaries to Exclude:

- `/opt/cisco/forensics/air/air`
- `/opt/cisco/forensics/air/drone`
- `/opt/cisco/forensics/air/tactical`
- `/opt/cisco/forensics/air/utils/osqueryi`
- `/opt/cisco/forensics/air/utils/curl`
- `/usr/share/.air/watchdog`

Anything missing?

Help us shape the future of Enterprise Forensics.

Your ideas are important

Something missing? Tell us about it by clicking the link below!