Artificial Intelligence and Machine Learning in the Security Operation Center

OVERVIEW

MAY 2020

paloalto® NETWORKS

Introduction

Many organizations task one or more people with defending the organization against security breaches while identifying, investigating, and mitigating cybersecurity threats. This group of people is often referred to as the *Security Operation Center* (SOC), and in many organizations, there is a physical space where these people work. It doesn't matter if the SOC is one person in a cubical or a team in a dedicated space; those tasked with protecting the organization face many challenges. The SOC is responsible for securing an ever-expanding environment. Too many low-fidelity alerts can hinder timely identification of important security events. And a cybersecurity skills gap hampers the SOC's ability to hire and retain staff.

Organizations' IT environments are expanding, and as a consequence, so are the demands on the SOC. Application development and IT operations teams are accelerating the delivery of new applications in order to drive business growth by adopting cloud and container technologies, big data analytics, and automation and orchestration. Monolithic applications are long gone. Each new application and system can involve dozens of devices and live across multiple environments, in both the public cloud and on-premises data centers. Because more users are mobile, the organization must also make its critical business applications more accessible to mobile devices.

In many organizations, the number of devices that have access to business-critical applications has expanded along with the number of applications. Most organizations must contend with at least two sanctioned devices for every employee. Because software-as-a-service (SaaS) applications are available to anyone with an internet connection, SaaS application adoption can increase the number of devices with access to an organization's applications and data, unless organizations implement controls that prevent personal devices' access to those applications.

Because data and computing reside in so many places, and because there are so many devices that have access, monitoring all of the possible attack vectors and identifying important security events is challenging and getting harder every day. Based on the number of sources of relevant data alone, it is apparent that it is impractical to review log files manually. The traditional approach to easing this challenge is to rely on a system to correlate input of dozens of different security products, which each monitor a specific attack vector, to alert the SOC when something outside the bounds happens. Most often, the SOC writes these correlation rules after an incident takes place so that if it happens again, the system with alert them. However, correlation rules often miss important events and provide a high number of false positives.

Correlation rules can miss important events because they rely on a specific set of inputs to trigger. If the SOC defines the rules too narrow, then the system won't alert if the next event differs in any way. With the number of applications, systems, and environments in an organization, the likelihood that an attack looks exactly the same from one event to the next is unlikely. However, defining broad correlation rules doesn't help the SOC, either. Broad correlation rules are a significant source of false positives. False positives generate alerts that either take the SOC away from finding real problems or cause so many alerts that the SOC would never be able to work through them all. The alerts just become noise through which the SOC must sift in order to find the real problems. Either way, analysts are missing attacks in the deluge of data, or they are finding the attacks too late.

Leveraging AI and ML in the SOC

To find important security events without generating low-value alerts that require analyst time, attention, and manual remediation, the SOC must leverage artificial intelligence (AI) and machine learning (ML). *Artificial intelligence* (AI) is a broad term that refers to algorithms, models, and a field of scientific study. *Machine learning* is the concept of training a system to perform narrowly focused tasks without using explicit instructions. Instead of instructions, ML relies on detecting patterns and inferring conclusions. It focuses on a specific need. AI and ML can identify important security events in an organization, with high fidelity, by stitching together data from multiple sources while reducing the time and experience required in the SOC.

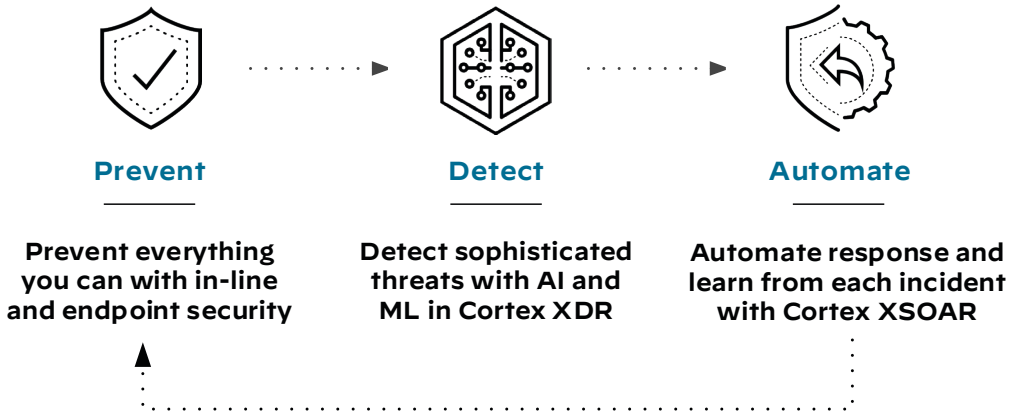
At Palo Alto Networks®, AI is a critical piece of our history of solutions that prevent successful cyber-attacks. The in-line and endpoint security components of the Security Operating Platform®, spread out across the enterprise and cloud, provide data to AI services that within minutes can detect new malware and identify malicious domains. The components also provide the point at which policy enforcement, based on the results of the AI services, prevent successful cyber-attacks.

In the SOC, the Palo Alto Networks approach is three-pronged:

- First, you prevent all of the threats you can. You achieve this with Palo Alto Networks in-line and endpoint security.
- Next, you need to rapidly detect and investigate the threats you can't prevent. You achieve this with Cortex™ XDR, which identifies, with high fidelity, the security incidents the SOC must investigate and to which they must respond.
- Then you automate future responses. Using Cortex XSOAR security orchestration allows you to execute automatable playbooks for accelerated incident response. *Playbooks* are a graphical workflow that allows you to standardize and scale a response process.

Cortex is Palo Alto Network's AI-based continuous-security platform, which continually evolves to help the SOC stop the most sophisticated threats. Cortex includes Cortex XDR, Cortex XDR Agent, and Cortex XSOAR. Cortex stitches all of your managed endpoints, network, and cloud events together. Because these various stacks witness only part of the buildup of an attack, Cortex uses AI to learn which events in one system drive attention to another system. Using AI and ML, Cortex also identifies unknown and highly evasive threats targeting your network.

Figure 1 Palo Alto Networks SOC approach



Preventing threats in the network and on the endpoint reduces the number of alerts the SOC must handle. In-line security (including the next-generation firewall, the VM-Series firewall, and Prisma™ Access) prevents:

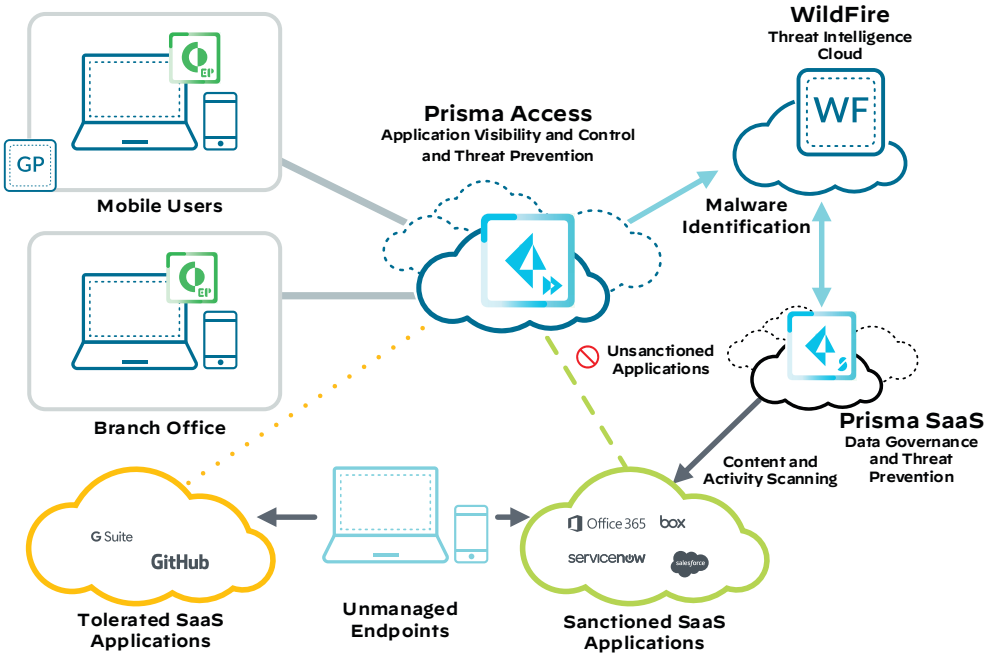
- Users from accessing risky applications and sites.
- Known malware and exploits across all applications.
- The proliferation of malware and attacks throughout an organization (when you use a Zero Trust security model).
- Credential phishing and data exfiltration.

Endpoint protection and response with Cortex XDR Agent (Formerly *Traps™*) on Windows, macOS, and Linux endpoints provides behavior-based protection in order to detect and respond to sophisticated attacks. Cortex XDR Agent thwarts malware infections by blocking the exploits used by attackers to compromise endpoints and install malware. Cortex XDR Agent protects hosts by identifying exploit techniques—not just exploit signatures—so it can stop zero-day threats.

Cortex XDR Agent identifies sequences of behavior unique to malware and ransomware with its powerful behavioral threat protection engine. Cortex XDR Agent integrates with Palo Alto Networks WildFire®, our malware prevention service, to analyze suspicious files in the cloud and coordinate protection across all Palo Alto Networks security products.

The right log data is critical for the SOC (whether or not they are using AI and ML) to identify and respond to attacks. Palo Alto Networks in-line security products and endpoint protection products can send log data to the Cortex Data Lake, which provides a central source of organizational log data to the SOC.

Figure 2 Prevention in the platform



Prevention cannot stop all threats to an organization. For example, a threat can enter the organization outside the visibility of in-line protection, or the threat might be specifically targeted to the organization and hasn't been seen before.

To detect advanced, targeted, and insider attacks, in addition to non- malicious risky behavior, Cortex XDR uses AI and ML to learn from the data in the Cortex Data Late in order to gain insight on activity across the environment. Cortex XDR establishes a baseline of normal behavior and then alerts on high-fi-delity anomalies. For example, Cortex XDR uses ML models to detect whether or not an administrative connection to a server is expected.

You need some background information in order to understand how Cortex XDR uses AI and ML.

Training data is used to teach an ML model how to categorize other data according to a set of desired outputs or labels. Training data needs to be balanced to not over-represent any desired outcomes. Training data comes in two types:

- **Labeled data**—Labeled data is used to train supervised-learning models. *Labeling* refers to data that has one or more attribute values about that data pre-classified so that a machine-learning algorithm knows how to sort the data. In cybersecurity, labels might identify things such as "benign" or "suspicious" files, URLs, or IP addresses. In healthcare, labels might identify "healthy" or "sick" patients.
- **Unlabeled data**—This label is used in unsupervised machine learning (UML) output. Unlabeled data is representative of what the ML model needs to classify.

Three types of machine learning are:

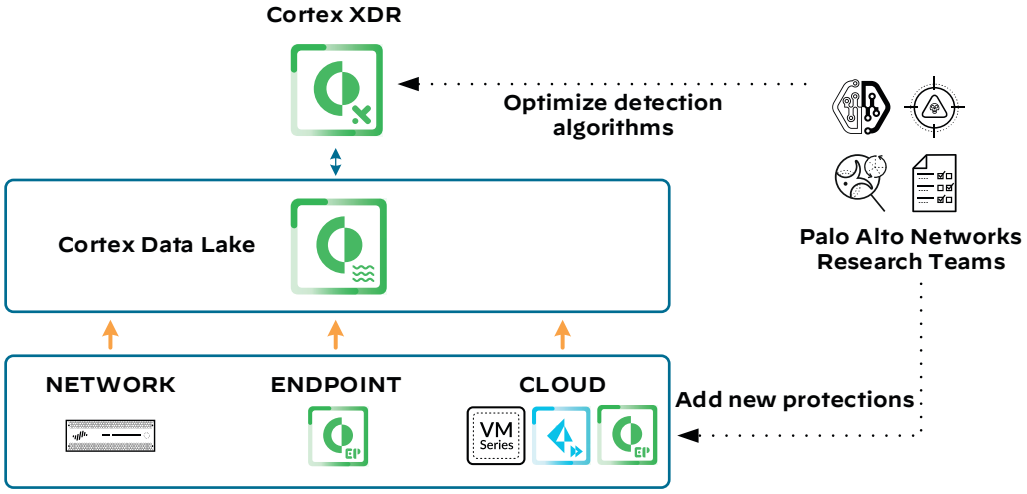
- **Supervised machine learning**—This paradigm uses labeled training data (categorized input and output examples) to derive mathematical models so that the system can classify unknown data. For example, training data might include benign files and files infected with malware so that the system can learn to distinguish between the attributes of benign and malevolent files. In cybersecurity, supervised learning requires the careful curation and constant updating of data as attackers develop new techniques. Supervised machine learning (SML) performs well when there are a finite number of data categories.
- **Unsupervised machine learning**—UML addresses the problem where the number of data categories cannot be determined in advance. There are many more use cases for UML in cybersecurity than for SML. UML builds a mathematical model from data sets that contain only categorized inputs, with no defined outputs. These models find patterns or structures in the input data, derive their output categorizations, and identify outliers or anomalies in the data. Further, UML reduces vast amounts of data down to recognizable patterns. Typically, UML identifies anomalies within the data set.
- **Semi-supervised machine learning**—Semi-supervised machine learning falls between unsupervised learning and supervised learning. Semi-supervised machine learning makes use of both unlabeled and labeled training data. In some cases, a small amount of labeled data can improve the model's accuracy when compared to UML.

For Cortex XDR, Palo Alto Networks researchers create the ML *models,* or the algorithmic representation of what a machine-learning system has learned from the training data, that detect anomalous activities. An ML model is essentially a more refined equation for the algorithm, one in which critical parameters used by the algorithms have been determined. In the example of a compromised network user, the parameters may include trusted or untrusted applications, ports being used, time of day of use, etc.

Cortex XDR uses pre-defined AI and ML models to identify anomalous activities. Many of the Cortex models are based on semi-supervised ML and require a lot of data to function. *Sensors* are responsible for gathering or generating the event records that the machine learning models process.

The Palo Alto Networks in-line and endpoint security components that provide prevention also function as the sensors for the AI and ML in Cortex XDR. Cortex Data Lake and XDR also support ingesting relevant log information from third-party devices such as Check Point firewalls.

Figure 3 ML models



For AI and ML to operate effectively, Cortex must have access to the relevant data from the next-generation and VM-Series firewalls, Prisma Access, and Cortex XDR Agent. Cortex XDR uses the data in order to identify both in-progress and latent attacks and know what controls will prevent successful attacks.

Cortex XDR AI and ML models require detailed data. The amount of data a typical organization collects in their security information and event management (SIEM) system is approximately 100 times less than what ML requires in order to be effective. Although the next-generation firewalls logs contain detailed information, including application and user identification, enhanced application logging allows the firewall to collect data that increases Cortex XDR's visibility into network activity. Examples of the types of data that enhanced application logs gather (that regular logs do not) include records of DNS queries, the HTTP header User Agent field that specifies the web browser or tool used to access a URL, and information about DHCP automatic IP address assignment. With DHCP information, for example, Cortex XDR can alert on unusual activity based on hostname instead of IP address.

Cortex XDR Agent also provides enhanced logging for Cortex XDR. By default, when a security event occurs on an endpoint, Cortex XDR Agent collects data about the endpoint. When used with Cortex XDR, Cortex XDR Agent can also continuously monitor endpoint activity. The specific endpoint data that Cortex XDR Agent collects when you enable enhanced logging varies by the platform type but includes information on files, processes, and network connections.

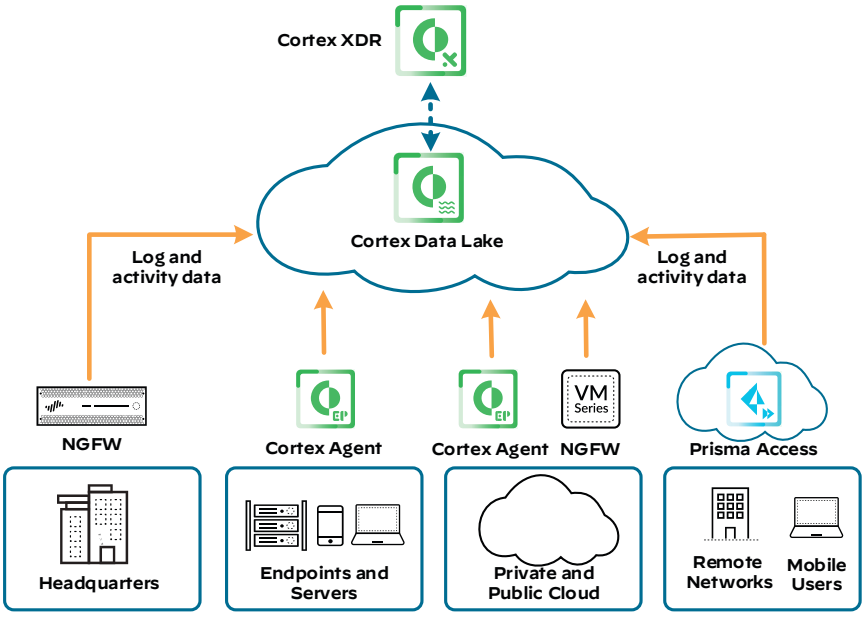
Because the sensors are part of the platform, we can collect additional security details in our logs with simple software updates. These features ensure the platform evolves with the requirements of the AI and ML model and frees our customers from having to deploy new sensors, enforcement points, and security data lakes each time they want to adopt new cybersecurity or UML technology.

The volume of data to effectively process AI and ML is so large that the data cannot be stored in a conventional file system. A *data lake* is a system that stores enterprise system data in its raw and untransformed or nearly untransformed state. A data lake stores raw data from which new transforms are created for consumption by another system or analytics engine. These transforms can calculate new fields or trends, correlate with other data sets, and broaden or narrow the data view to fit a specific purpose and timeframe. The point is that we can't always predict what data will be useful, so we want to retain any potentially valuable information.

Cortex Data Lake stores the relevant data for Cortex XDR. Cortex Data Lake makes the sensor data most useful for AI/ML because:

- It resides in one place but provides visibility across an entire organization—the cloud, the endpoint, the data center, and the network.
- It is normalized, no matter the source or type. Normalizing the data ensures that ML processing is as efficient as possible.
- It leverages the cloud for limitless storage, compute, and global locations to satisfy diverse data residency requirements and enable rapid, on-demand deployments of new security devices.

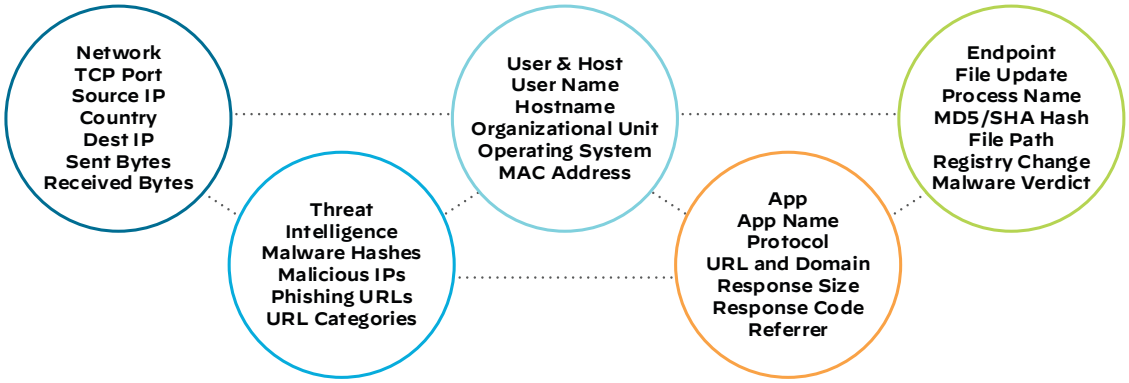
Figure 4 Cortex Data Lake



These requirements ensure that data stored in Cortex Data Lake can support AI-based innovation in Cortex XDR. The Cortex Data Lake houses threat information, including malware verdicts, IOCs, known bad sites, and threat actor playbooks.

To provide meaningful insights to both detection and investigation activities in the SOC, Cortex XDR stitches the different data contexts together into one data set. For example, if Cortex XDR detects malicious network activity in data from the firewall, Cortex XDR can correlate that activity with endpoint logs to observe the impact of the activity and identify the cause of the behavior. Log stitching streamlines detection and reduces response time by eliminating the need for manual analysis across firewalls and endpoints.

Figure 5 Log stitching

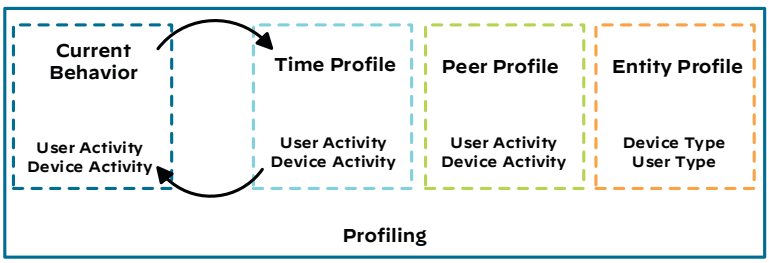


To ensure the SOC only sees the most relevant and high fidelity alerts, Cortex XDR uses AI and ML to detect and identify anomalies. Cortex XDR uses supervised, unsupervised, and semi-supervised machine learning models to generate organization-specific insights.

Cortex XDR uses the network traffic and endpoint activity data stored in Cortex Data Lake to understand the baseline of normal behavior so that it can raise alerts when abnormal activity occurs. There isn't one baseline behavior for the whole organization; there are thousands of baselines that Cortex XDR is continually updating.

To accurately baseline a device, Cortex XDR first profiles the type of device and the type of user on the device. After Cortex XDR creates the profile, it creates a history of the behavior of that device and devices that have similar profiles. This collective view of the device's behavior and its peers' behavior is the baseline against which new activity is compared, and establishing it requires time. This learning period exists to give Cortex XDR enough data to establish the baseline, which in turn helps to avoid false positives.

Figure 6 Intelligent profiling



To keep false positives low, identifying devices with similar profiles is important. Without the peer groups, ML in Cortex XDR might compare a regular user with an IT administrator or a server with a mobile device. In those cases, anomalies would happen, but they wouldn't necessarily be behavior that should warrant attention in the SOC.

The predefined ML models used in Cortex XDR are called *detectors*. Each detector is responsible for raising an alert when it detects abnormal behavior. Because organizations and their environments continuously evolve, the detectors use a sliding window of time to update the "current" behavior and where old behavior falls off.

To raise alerts, as firewalls and endpoints send logs to Cortex Data Lake, the ML in Cortex XDR analyses the behavior as soon as it arrives. Each detector compares the behavior to the expected baseline and reports an alert when it determines there is an anomaly. Because Cortex XDR has stitched the log data, it can display the alert-triggering sequence of activity, or *causality chain*, across the organization. Causality chains help the SOC quickly identify the root cause of every alert. Cortex XDR also identifies a complete forensic timeline of events to help the SOC determine the scope and damage of an attack and provide an immediate response.

To make investigations easier on the SOC, Cortex XDR determines the most relevant artifacts in each alert and aggregates the alerts into *incidents* corresponding to a single event. The artifacts the Cortex XDR provides includes items like the command-line argument of a process, hash values, path, start-time and end-time of execution, user name, and all activities done by a given process (network connection, registry changes, loaded modules, file activity, etc.).

When an investigation finds something that needs remediation, Cortex XDR allows you to perform immediate response actions. To stop future attacks, you can pro-actively define indicators of compromise (IOC) and behavioral indicators of compromise (BIOCs) to detect and respond to malicious activity.

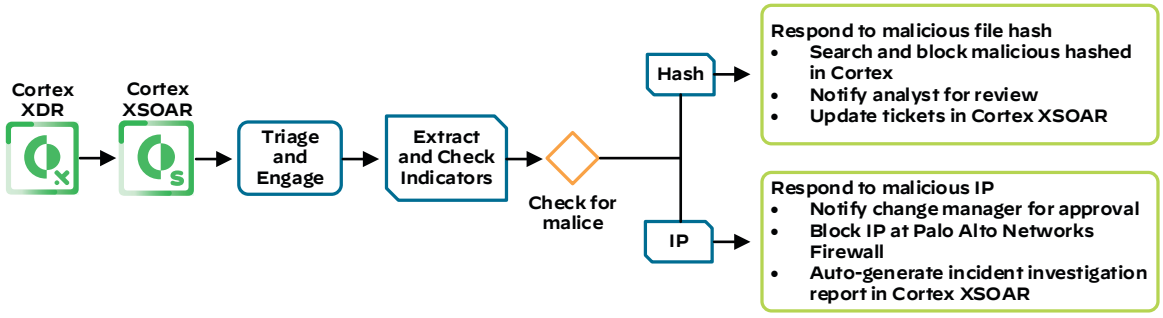
Cortex XDR scans data in the Cortex Data Lake for BIOCs and IOCs. When you configure new indicators, Cortex XDR matches them against all the previously collected data stored in the Cortex Data Lake, as well as all new incoming data. This means that, for example, when you create a new BIOC, Cortex XDR scans all events that exist in Cortex Data Lake for historical matches, alongside searching for matches in the incoming event stream.

As SOCs grow and mature, the team spends most of their day fighting fires and can't devote enough time to set standard response processes or spot patterns that reduce rework. This results in response quality being dependent on individual analysts, which can lead to inconsistent effectiveness.

Cortex XSOAR is a security orchestration, automation, and response (SOAR) platform that combines orchestration, incident management, and interactive investigation in order to serve security teams across the incident life cycle. The Cortex XSOAR orchestration engine coordinates and automates tasks across hundreds of products, resulting in increased coordination between existing security investments. Cortex XSOAR helps SOC teams to reduce mean time to response, create consistent incident management processes, and increase team productivity.

Leveraging the threat-detection capabilities of Cortex XDR, along with the security orchestration and automation of Cortex XSOAR, helps teams unify and automate response processes. Cortex XSOAR can ingest information from hundreds of sources, including Cortex XDR, and trigger playbooks that coordinate a response.

Figure 7 Cortex XSOAR and Cortex XDR



Cortex XSOAR integrations allow the playbooks to automate tasks across products, while also allowing for human oversight and interaction. Playbooks can be automated, manual, or a mixture of both, depending on user requirements. Cortex XSOAR playbooks triggered off Cortex XDR data help minimize screen switching, manual reconciliation of data, and repetitive work for security teams.

Cortex XSOAR uses machine learning to provide incident handling guidance based on past actions and historical information. Before Cortex XSOAR assigns an incident owner, its machine-learning studies the details of all past incidents in the system, including incident types. Cortex XSOAR then cross-references this data with existing analyst load to suggest the top three analysts that are best-placed to own the incident.

Cortex XSOAR facilitates real-time investigation through the War Room, a shared space where SOC team members can collaborate, remotely execute actions, and have all their actions documented at one source. The War Room allows team members to quickly pivot and run unique commands relevant to incidents in their network from a common window. All participating team members have full task-level visibility of the process and are able to run and document commands from the same window eliminating the need for gathering information from multiple sources for documentation.

RELATED DOCUMENTATION

Prevention, Detection, and Response for Security Operations Reference Architecture Guide

This guide provides solutions for prevention, detection, investigation, and response to help security operations prevent threats and efficiently manage alerts. This guide can be found on the Reference Architecture site.

https://www.paloaltonetworks.com/referencearchitectures

To learn more about Palo Alto Network's solutions for the SOC, including Cortex XDR and Cortex XSOAR, see https://www.paloaltonetworks.com/cortex

You can use the feedback form to send comments about this guide.

HEADQUARTERS

Palo Alto Networks
3000 Tannery Way
Santa Clara, CA 95054, USA
http://www.paloaltonetworks.com

Phone: +1 (408) 753-4000
Sales: +1 (866) 320-4788
Fax: +1 (408) 753-4001
info@paloaltonetworks.com

B-000108P-20b