DRAGOS

The Four Types of Threat Detection

With Case-Studies in Industrial Control Systems (ICS)

By Sergio Caltagirone and Robert M. Lee

There is a considerable amount of market confusion around the types of threat detection, how they are derived, and the uses for each. The purpose of this paper is to address those challenges by identifying the four types of threat detection and offering sample use-cases focused on industrial control system (ICS) and industrial internet of things (IIoT) environments.

Threat Detection: The Most Important Function

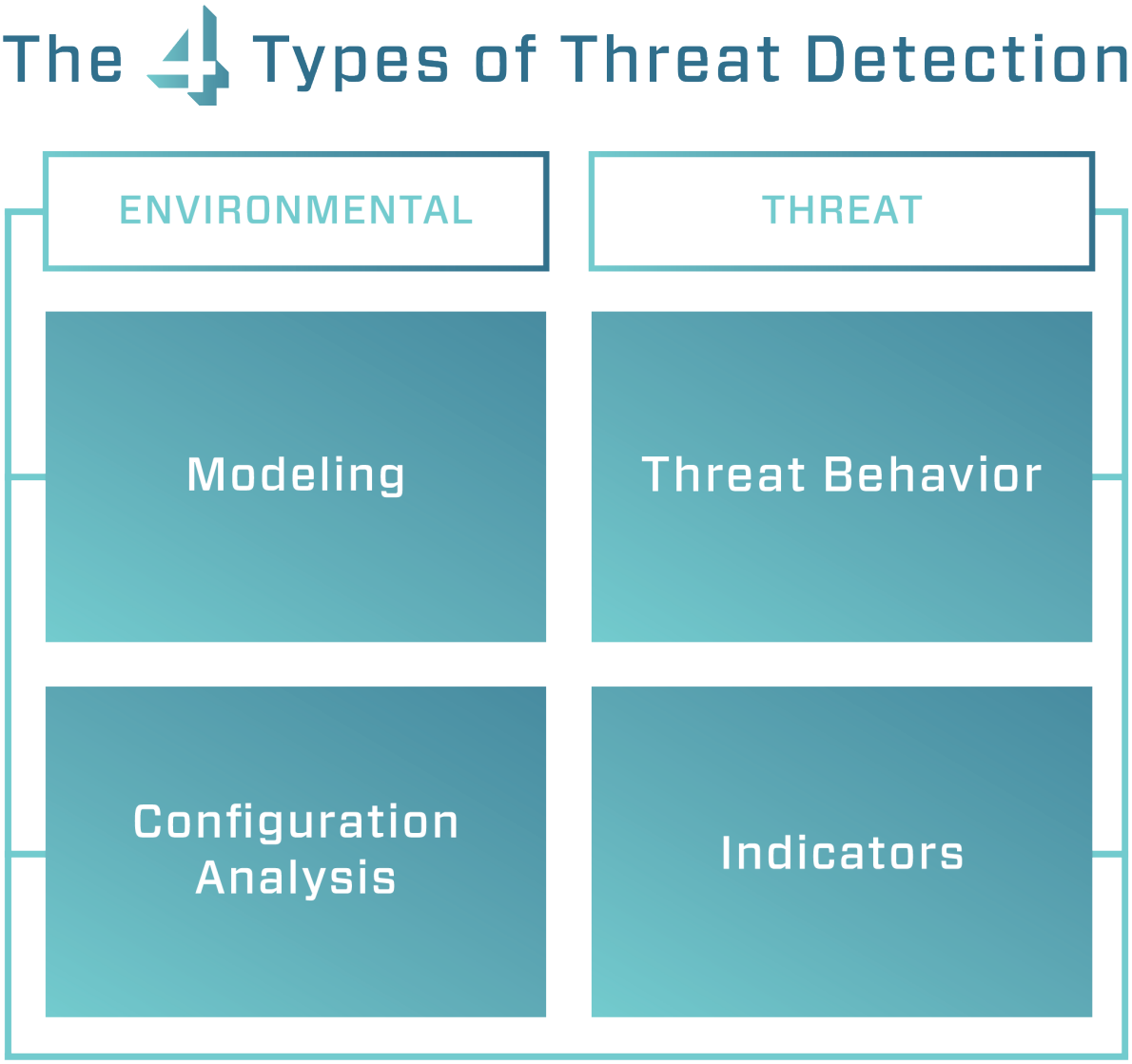
Threat detection plays an outsized role in cybersecurity as arguably the most important function in an "assume breach" world.

Threat detection comprises one of the three core cybersecurity functions, along with prevention and response. But, detection plays an outsized role as arguably the most important cybersecurity function in an "assume breach" world. Prevention is critical to reducing the noise from common threats, but sufficiently determined adversaries will always defeat prevention. Without detection, an adversary will dwell in an environment, achieving incredible freedom of movement enabling significant disruption at a time of their choosing. Good detection enables better response, and good response enables better prevention through root cause analysis.

Detection in industrial networks can help avoid significant financial impact to the organization, environmental impacts, loss of safety, or inappropriate response plans when a cyber component of the disruption is not understood. Historically, detection has been positioned in numerous ways, with a focus either on the type of threat that was being detected, like targeted threats versus cybercrime as an example; or in the tools and technologies used to facilitate the detection such as system information and event management (SIEM) rules, intrusion detection system (IDS) rules, machine learning models, and user-entity analytics. But, not all detection is equivalent or fits every scenario and application. Therefore, it's best to match the detection to the application. The following sections provide guidance for defenders on detection types and their applications so threats can be found and defeated earlier.

The Four Types of Threat Detection

Generally, all threat detection falls into four major categories: Con guration, Modeling, Indicator, and Threat Behavior. There is no best type of threat detection. Each category can support different requirements and approaches depending on the business requirement. If your goal is to d novel attacks and you are willing to spend signi cant effort, then modeling is a good approach. If your goal is to similar attacks with less effort, then threat behavior analytics[1] are a great approach.



[1] In this paper, we also refer to analytics which are the implementation of a threat detection approach replacing traditional human function/analysis with an automated process. When you hear analytic, think "threat detection by machine."

Most importantly, all of these threat detection categories become more powerful when used together to complement each other. For instance, model-based threat detection can be strengthened with expert-led con guration detection to reduce the false-positive rate problems seen with modeling. When and how you use these approaches together is the art of modern threat detection.

All threat detection types become more powerful when used together to complement each other.

The following sections will describe each threat detection approach, provide examples of their usage to threats in industrial control environments, and provide summaries of their bene challenges.

Configuration Detection

Con guration-based detection uses current knowledge of a known architecture or design to identify deviations.

Con guration-based detection identi es deviations from a known architecture or design like the known form of an internet protocol (IP) packet header or devices designed to communicate in a static pattern. Con guration can cross any domain including: network (e.g., only encrypted data over TCP 443), asset (e.g., executable es only start from a non-temporary directories), identity (e.g., users authenticate to single organizational Active Directory), cross-asset (e.g., eld devices like programmable logic controllers never communicate with each other), or any other structure. Co uration detection can easily be thought of as "newness," as it primarily alerts on new changes to the understood baseline.

Example: Configuration Detection

If assets are con gured such that a programmable logic controller (PLC) only ever legitimately uses a handful of ports and protocols such as ModbusTCP over TCP Port 502 and FTP over TCP Ports 20 and 21, then any communication from or to the device outside those ports and protocols would be suspicious to con guration-based detection. Likewise, if the PLC has speci c set points for appropriate use, knowing that n could lead to alerts outside of those set points as well.

Configuration Detection Benefits and Challenges

Con guration detection only requires one element: current knowledge of an environment's proper architecture and design. Because of this simple ingredient, it is the easiest form of detection to create in terms of experience required by the analyst. Further, con guration detection can (hypothetically, given perfect visibility and knowledge) catch all malicious actions as malicious activity must deviate from the established con guration at some point to cause a malicious event.

Unfortunately, con guration knowledge is likely to change in any environment. In ICS networks, con guration analysis can be effective especially for highly-static environments, but even the most static industrial control systems are not truly static. Maintaining detection accuracy and coverage in even moderately dynamic environments can be daunting. Additionally, because this approach detects all deviations from the co uration, there is a high false-positive rate through the alerting on legitimate changes. Poor knowledge or maintenance of the co uration can make this form of detection nearly unusable.

Con guration-based detections can be easy to generate and store and therefore are often a staple detection leveraged by security personnel for forensic examination or hunting. Single con guration alerts should rarely meet high priority thresholds but can (and should) be combined with other detections to improve their effectiveness and serve as a historical change view for forensics and response. As an example, if one of the other three detection types generate an alert, it would be useful to search for con guration changes before and after the alert generated to add a more comprehensive view of those detections.

Summary: Configuration-based Detection

Con guration-based detection identi – s deviations from a known architecture.

Example: Two d devices (e.g., PLCs) communicating with each other, counter to architecture and design expectations

Bene ts:

- With perfect visibility and coverage, it can hypothetically detect all malicious activity
- Accessible for individuals with a wide range of experience
- Easy to maintain in static environments
- Adds signi cant value to other detection types in response situations

Challenges:

- Dif cult to maintain in dynamic environments
- Limited visibility and coverage reduce effectiveness
- Assumes a knowledge of infrastructure and con guration
- False-positive prone due to likely con guration changes

Modeling

Modeling is a mathematical approach to detecting threats by de ning "normal" and measuring deviations from the de nition.

Modeling is a mathematical approach to detecting threats by de ning "normal" and measuring deviations from the de nition often over a period of time. Modeling detections base their approach on an underlying assumption that the detection engine can suf ciently distinguish illegitimate activity from legitimate activity. Modeling and con guration detections are very similar except whereas con guration is derived from experts in the operation of an environment, most modeling techniques attempt to build a picture of the environment with little-to-no expert input. However, this is dependent on the type of modeling used.

Usually, the fundamental approach of applying a model to build a pro e of the asset often includes a baseline, time, and threshold. Models may include supervised or unsupervised machine learning models, but numerous approaches exist. As an example, in IT network security the concept of user entity behavior analytics (UEBA) has gained attention for its ability to consider user actions and build models on what is normal user behavior. An interesting type of modeling in ICS is protocol behavior analytics, sometimes identi ed as behavioral anomaly detection, here protocols can be pro ed such as the state machines of the IEC104 network protocol to determine when the state machines have been violated indicating abnormal behavior. Most commonly though, organizations advertise machine learning-based approaches where the user runs a system to build a pro e of the environment and then continues to tune this device for as long as its operated.

Example: Modeling

The goal of modeling is to build pro es of the asset(s) over time and alert on anomalies that exist. As an example, Modbus TCP might be used to interact with the PLC but the model built for detection could detect that the manner in which Modbus TCP is being used is anomalous, or suspicious. It could be that the PLC is being operated by a new person, an attacker, or has a n based on abnormalities such as session lengths or the frequency of use for speci c function codes.

Modeling Benefits and Challenges

Modeling can be seen as the evolution of con guration analysis to help reduce the likelihood of false positives and also capture the actions which may not be deviations from the con guration but still represent malicious actions. There are many ways to model environments, but most strive to fully understand the modeled asset(s). As a bene t, this detection type, when done with a well-trained model, can detect unknown malicious actions because it is not considering the threat characteristics, but instead identifying changes in the environment. Modeling can also support other detection types by prioritizing anomalous activity worth inspection that may occur within the same relevant time period as other threat detection.

Additionally, this type of detection is also useful outside of threat detection and can help inform an understanding of maintenance in industrial environments by identifying anomalies that might indicate miscon ured or failing assets.

Unfortunately, modeling does have distinctchallenges as well. First, building the initial model for the environment requires signi cant investment in understanding all aspects of the systems including their communications. As changes to the environment take place, such as adding new systems or con gurations, the model will need to be re-built or re-trained. Second, modeling requires constant tuning and training. Most models require a training and tuning period before adding value, but what is not well understood is that consistent maintenance will be required. Third, because the model does need to build a pro for the environment, any malicious activity already present in the environment will likely become part of the model. In many cases of persistent threat activity, responders d that the threats have been active in an environment for hundreds of days, and in some cases, years. Finally, because the mechanism of detection is often a trained mathematical model, there is little transparency and context into why the alerts occur. Some models may note speci cs such as " rmware update detected," but understanding why the model alerted on that update and not others is not easily obtainable.

Detections based off an understanding of the environment, such as modeling and detection, do not have any context of the threat activity surrounding the alert, such as what an adversary might be doing which would otherwise help support investigations. Context is left up to defenders who must research and investigate the alert and why it exists, as well as correlate it with other activity and alerts. Modeling is the most time-consuming form of detection when done correctly, as the alert does not contain context of the threat at the time of alerting. Thus, the defender must provide all insights and context post-detection with other datasets or their own experience.

Summary: Modeling-based Detection

Modeling-based detection uses mathematical models to classify assets and activity identifying elements inconsistent with the model.

Example: Abnormal number of Write requests in Modbus TCP outside of normal, given the average over the last 30 days

Bene ts:

- Can identify novel adversary activity
- Easier to maintain in very staticenvironments
- Adds signi cant value to other detection types in response situations

Challenges:

- Difficult to maintain when environments change
- Limited visibility and coverage reduces the effectiveness
- No context of threat activity to support investigations
- Assumes analysts have in-depth knowledge of infrastructure and configuration
- False-positive prone due to likely configuration changes
- Potentially incorporates existing malicious activity into the model

Indicators

Indicators are elements of information which identify a particular state and context – there are both "good" and "bad" indicators

Indicators are simply elements of information which identify a particular state and context (i.e., what did you find and why does it matter). Within information security, there are both "good" and "bad" indicators used for different purposes. There can be indicators for legitimate files (e.g., whitelist) as much as illegitimate files (e.g., blacklist). "Indicators of Compromise" are generally the most common reference point for indicators in information security. Indicators of compromise (IOCs) represent the technical elements of malicious activity generally derived from a digital investigation which can match elements in data sets to support detection and investigation.[2] Analysts can easily derive indicators after observing threat activity. For this reason, most indicators originate from existing investigations or when performing analysis such as malware analysis. Indicators made hastily through automated methods, such as harvesting the output from malware sandboxes, have historically been ineffective and can accidentally include legitimate activity. Indicators made through the analytic process can be more effective although the effectiveness of indicators is entirely based on the ability of defenders to apply them correctly and adversary rate of change.

---

[2] The term indicator in digital forensics, threat intelligence, incident response, and security operations, encompasses so many detection types and elements it becomes too broad to generate meaningful detection strategies. The Kill Chain, as defined in the paper *Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains,* defines three "types" of indicators: atomic, computer, and behavioral. However, detection varieties and strategies have grown since then. This paper does not attempt to refine "indicator." Instead, for our purpose of discussing many types of detection strategies, this paper reverts the term indicator back to its original meaning and instead of overloading the term focuses on separating it from other detection strategies such as threat behaviors.

Example: Indicators

As an example, if an analyst identifies a specific external IP address that is accessing a field device (like a PLC), potentially through an otherwise authorized VPN, the analyst could create an indicator based off that IP address and search throughout the environment to scope what other systems are potentially being accessed inappropriately. This effort would allow the analyst to more quickly create a detection and identify malicious activity compared to the other types of detection.

Indicators Benefits and Challenges

Indicators are the quickest way of leveraging detection with threat context. When properly created, indicators identify specific activity that gives defenders the context to properly prioritize and respond to the activity observed. IOCs are the most widely used indicators, but indicators can be leveraged to indicate a wide range of activities; anything that has specific data and context can be considered an indicator.

There are two main benefits associated with indicators: knowledge enrichment and quick scoping. Knowledge enrichment seeks to take data or knowledge the defender already has and enrich it with new knowledge. As an example, a defender may understand how an intrusion is taking place in their environment along every phase of the intrusion kill chain except the command and control phase. By leveraging an externally sourced indicator for a similar intrusion with the context that the adversary is performing a specific type of DNS query for command and control, a defender may search for the same activity in their environment. Additionally, the defender may leverage indicators after other forms of detection have taken place for forensic value. This helps reduce false positives while adding context to existing information. As a scoping tool, the defender may create indicators specific to malicious activity they observe in their environment. This indicator can then be used throughout the environment to scope for assets that are likely also compromised.

Indicators are not commonly great threat detection tools in and of themselves and should mostly be used to complement other detection types.

However, indicators are not commonly great threat detection tools in and of themselves and should mostly be used to complement efforts. There are three major limitations on indicators: adversary change, upper bound capacity, and limited usefulness. First, an indicator is only as good as long as it is valid. If an adversary were to change operations nullifying an indicator, its value would be greatly limited in primary detection. This is only defined by the adversary and defenders have no control of when the indicator becomes useless Second, because there can be hundreds or more indicators per malicious activity, the sheer number of indicators over time can overwhelm processing systems. Analysts are left to try to determine which indicators are most valuable although, as stated in the first point, only the adversary determines which will have the most [illegible]t value. Lastly, indicators may not translate well across victims and are entirely reactionary thereby only identifying what is already known.

Unfortunately, indicator feeds are largely abused, and the core of most threat-sharing programs are indicators and blacklists. Indicators should not be viewed as threat intelligence, but simply the byproduct of it. Indicators should, thus, be used in concert with other efforts, but not relied upon for the primary detection mechanism.

Summary: Indicator-based Detection

Indicator-based detection searches for elements of information known about previously and are often seen in the form of Indicators of Compromise (IOCs).

Example: A spec c IP address that is accessing an internal asset

Bene ts:

- The quickest form of detection to create and deploy
- Contains speci c threat context related to the indicator
- Useful for enriching other data sources and threat detections
- Highly effective for scoping an environment post observation of the indicator

Challenges:

- The value is highly dependent on the adversary's rate of change
- Retroactive in nature given the need to observe the indicator rst
- Does not scale well between victims
- Upper limits as to how many indicators can be processed
- Unknown indicator expiry leads to inaccurate detection

Threat Behaviors

Threat behaviors codify malicious adversary tradecraft (e.g., techniques, methods) for detection, regardless of speci c indicators such as capability or infrastructure.

Threat behaviors are the of threat tradecraft, such as an adversary's methods, that represent a scalable and transposable approach to searching for malicious activity. Threat behaviors abstract away individual technical elements of indicators and instead focus on the behavior. This makes threat behaviors, often used in the form of threat behavior analytics, more capable of scaling in usage.

Example: Threat Behavior

As an example, a PLC that is remotely operated from an HMI that has been accessed inappropriately by the adversary over a VPN could be detected as a "SCADA Hijack" behavior. The threat behavior analytic alerted on the complex correlation of events of known adversary tradecraft and was not bound to speci c indicators. The threat behavior would come with an understanding of what that activity represents and context for the defender to utilize.

Threat Behaviors Benefits and Challenges

Threat behavior analytics are the prime form of scalable and transposable threat detection; they also result in context. These analytics are often formed from a chain of events that result in a complex rule set of malicious activity abstracted away from adversary data such as IP addresses. This allows defenders to use the analytical library immediately instead of requiring a model to be created and tuned or speci threat activity to be observed for indicators. The signature-like implementations are the of adversary tactics, techniques, and procedures. Thus, they are built upon knowledge of adversary methods but are not bound to tools or vulnerabilities, making them increasingly t for adversaries to avoid. In fact, tradecraft is often not even speci c to adversaries. Whereas indicators do not scale well outside of individual victims, threat analytics scale so well that they can alert to previously unknown threats that use existing tradecraft.

Threat behavior analytics reduce workload by serving up context, as well focusing the analyst's time on responding instead of understanding the alert itself.

Threat behavior analytics reduce analysts' workloads when a threat is detected by serving up context as to what is observed, thereby focusing analysts' time on responding appropriately instead of trying to understand the alert itself and why it triggered. Because this type of analytic has the context of the threat activity, they can also be paired with workflows, such as incident response playbooks, to give analysts best practices to utilize in each given scenario. Threat analytics paired with incident response playbooks are the most effective and D method of performing an investigation.

Although threat behavior analytics save analysts considerable time post-detection, they are time intensive to create pre-detection. Challengingly, threat behavior analytics cannot be procedurally generated through methods such as modeling and instead have to be speci cally crafted by analysts with knowledge of adversary tradecraft. The analytics also have two other considerations compared to environmental detections. First, although threat behavior analytics scale very well in a given industry, the more speci c the malicious activity the analytic is looking for, the less likely the analytic is going to scale well outside that industry. As an example, a threat behavior analytic to detect a SCADA hijack of an electric transmission environment would not likely be useful in the process control network (PCN) of an oil r ry, although it would scale to every electric transmission environment Second, there must be a prompt to create the analytic.

This is usually reliant upon threat intelligence, such as intrusion analysis of adversary activity, but can also be achieved through the knowledge of defenders of what a threat would do in a given environment.

Though threat behavior analytics can easily detect unknown malware and exploits, they are not effective at detecting completely novel tradecraft. Most adversaries do not leverage completely novel tradecraft, but it is a consideration that the first time in history something happens, such as a SCADA hijack at an electric transmission substation, there may not have been useful analytics to detect it. However, an adversary's intrusion is never bound to a singular event. Therefore, if the defender has good analytical coverage across the various steps an adversary can take, the threat analytics can detect non-novel tradecraft and defenders can pivot to identify the novel tradecraft in the investigation. Even in novel tradecraft scenarios, there are rarely more than one or two steps that are novel, and thus the adversary is detectable at various phases of their intrusion kill chain.[3] However, even in the most extreme of cases, after the novel tradecraft is detected, it is forever detectable even when the threat actor, tools, and vulnerabilities change.

Summary: Threat Behavior Detection

Threat behavior analytics examine activity in environments and compares single actions and aggregate actions against a set of known malicious or suspicious activities.

Example: Legitimate VPN access, followed by user account creation and file download on an engineering workstation, and finally, login from the workstation to an HMI

Benefits:

- Excellent durability against adversary change
- Easy to tune for each organization and environment
- Low false positive rates
- Immediate transparency for analysts to diagnose the alert against expected behavior
- Only requires a few analytics to detect most known malicious behavior used somewhere in an intrusion
- Integrates well with defensive playbooks and automated investigation/remediation

Challenges:

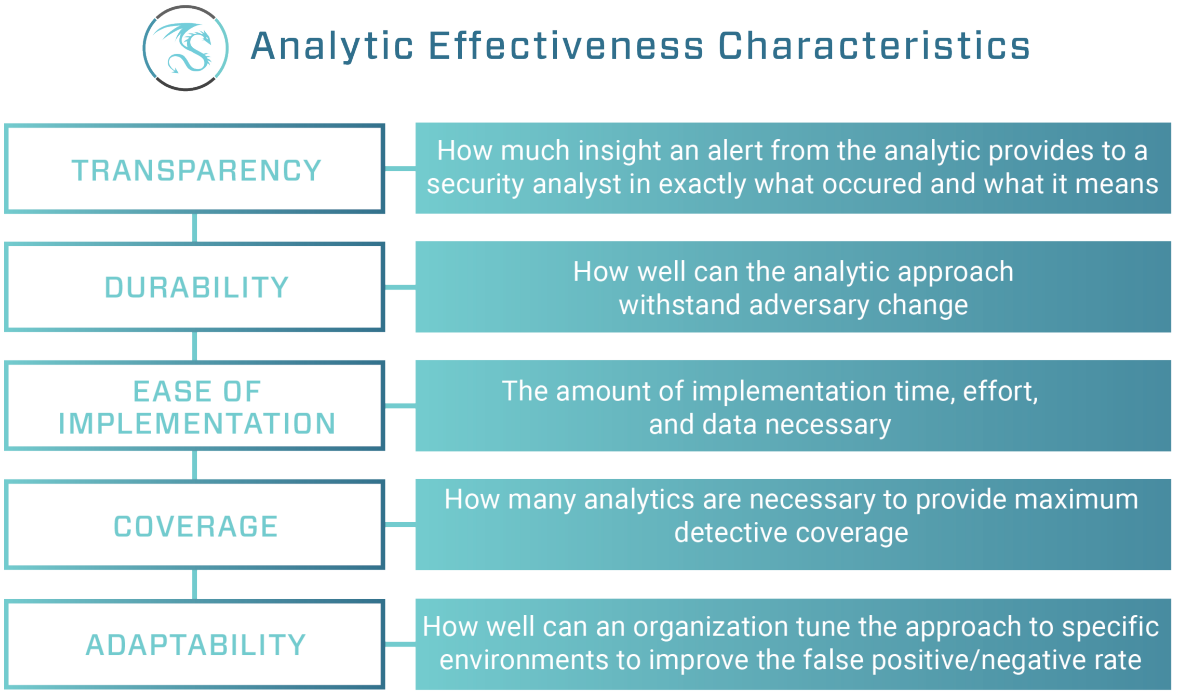
- Moderately difficult to implement
- Many analytics required to provide complete coverage
- Only detects similar threat behavior at the limit of analytic imagination
- Are not fully reusable across all industries

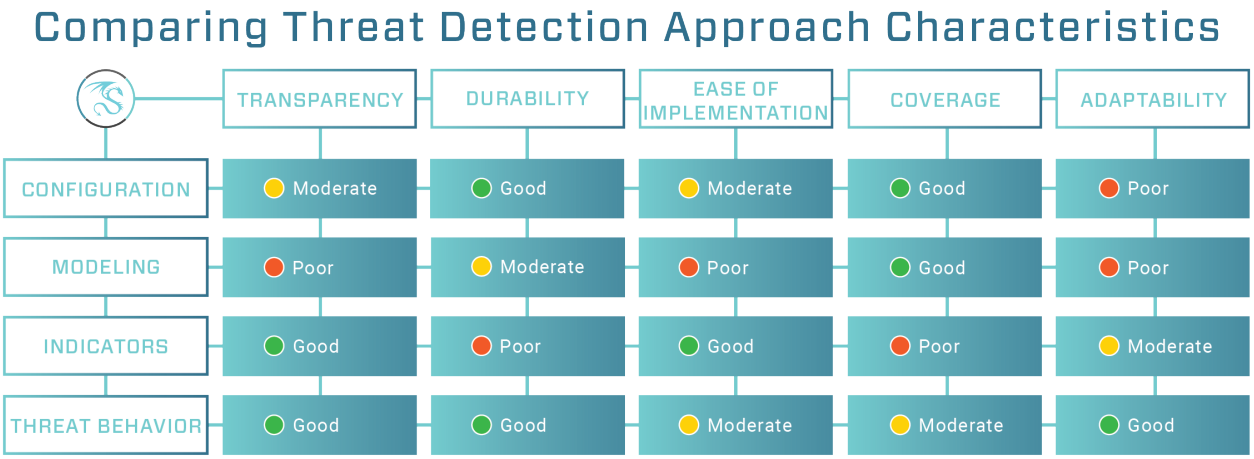
3 A common cliché is that adversaries only have to get "one thing" right and defenders have to protect everything. However, with proper analytical coverage and investigation capabilities the defenders only have to reliably detect "one thing" whereas adversaries have to do everything correctly to avoid detection. (Bejtlich, Richard. https://taosecurity.blogspot.com/2009/05/defenders-dilemma-and-intruders-dilemma.html)

Comparing Threat Detection Approach Characteristics

Different threat detection approaches provide different bene ts and costs. Therefore, it's critical for organizations to select the appropriate approach to meet their requirements. For example, it would be unwise to apply generic modeling post-compromise during an investigation when other approaches work much better with less effort.

First, we can approximate the effectiveness of an approach to the general characteristics of good threat detection.

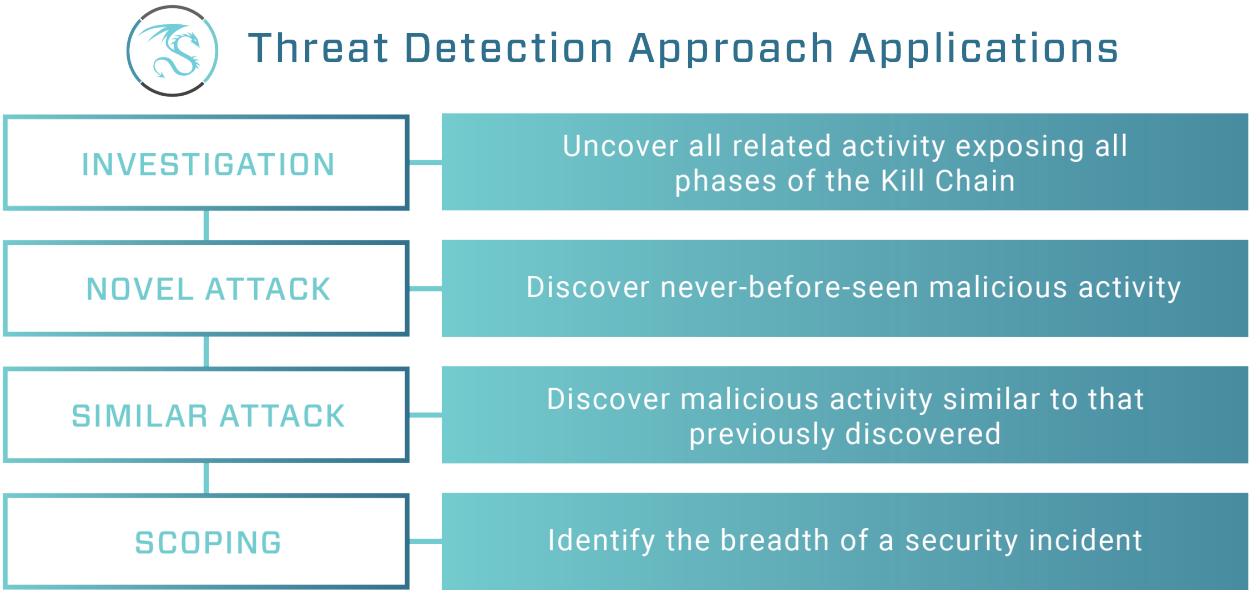


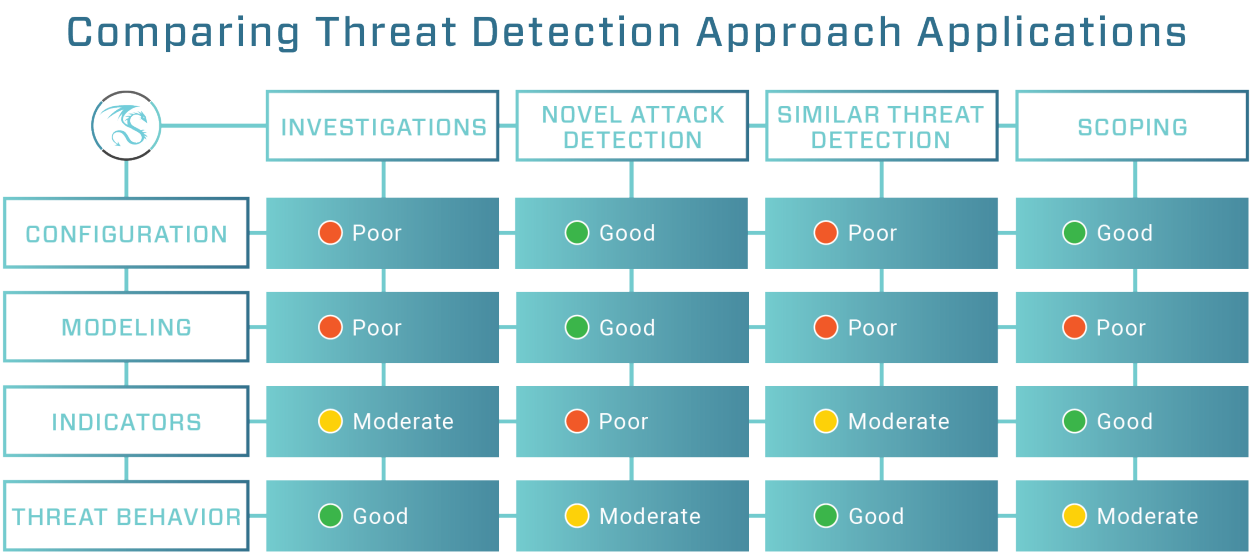


Next, we can compare the effectiveness of an approach to a application.

Example Approach Applications

Defenders ultimately want to understand the various applications of threat detection approaches. Where and when they will use these approaches in the detection, analysis, and remediation process. Again, the answer is not one-to-one. A combination of approaches will always produce the best results. But, generally different threat detection approaches align better with different applications.





Investigations (including digital forensics). Configuration-based and indicator-based detection serves as the most useful post-threat detection during investigation and forensics when organizations want to know everything that happened and why. Data generated by configuration changes serve as a lightweight and easily storable data source that can give a better view of activity happening in the environment. Well-crafted indicators leveraged across the configuration data can quickly check to see if known threats or their capabilities and infrastructure are detectable. In addition, once investigators identify the threat through modeling or threat behavior analytics, an indicator of that threat can be created and used to scope across the environment. Indicator checks against massive datasets can take time, but checking against just those datasets that also were detected through configuration changes can reduce the complexity of the problem.

Novel Attacks. Modeling is useful for identifying truly novel attacks, but modeling can also generate alerts with little-to-no context making alert prioritization difficult. Threat behavior analytics can identify previously unknown threat activity groups, infrastructure, and capabilities such as new vulnerabilities or exploits; however, threat behavior analytics cannot directly identify truly novel tradecraft. Instead, threat behavior analytics should be used to detect similar tradecraft and then pivot the investigation through configuration analysis to identify the novel tradecraft. Adversaries do not utilize novel tradecraft for every step of their intrusion, therefore modeling and threat behaviors work together to create a comprehensive strategy.

Similar Threat Detection. Identifying similar attacks to those seen before works the best with threat behavior-based detection. After an investigation, analysts document and share the threat characteristics (i.e., behaviors and techniques) identified in an intrusion or breach, allowing other defenders to discover those characteristics in their environment. Even if the intrusion is by a different adversary, but they utilize common techniques, their activity will be discovered. Because most adversaries use common behaviors during some point of every intrusion, this method works very well for detection. However, currently defenders primarily use indicator-based detection for similar threat detection but indicators are not highly effective due to their short life-span and other challenges.

Scoping. Indicators and a strong knowledge of the correct configuration is the fastest method to scope a security incident after discovery. These two detection methods don't generally get you very deep into the various threat behaviors, but they can usually find out how far across an environment a threat has reached. For example, quickly identifying all assets accessed by a subverted account (the account being the indicator) allows an analyst to identify potentially exploited hosts immediately. Threat behavior detection can still be effective when looking for the same behavior across an environment, but only moderately– modeling is even less effective for this use case.

Concluding Thoughts

There are generally four types of detection and each type has bene ts and challenges as well as effective application based on need and mission. Security-conscious organizations must create a detection strategy based on a combination of the four detection types to achieve a "detection- rst" methodology.

The four types of detection are a simple way to capture the methods which defenders detect malicious activity. Each has bene ts and challenges. With a perfect allocation of resources, teams could leverage each detection type in conjunction with each other. Unfortunately, security teams leveraging these detection types are oftenlacking resources.

Security personnel should determine which detection types work for them in different scenarios. As an example, if the purpose of the alert is to immediately block the activity, such as a new system process launching then modeling may be appropriate as the context of the threat data is not required for the action. However, if the purpose of the alert is to open an investigation and determine the surrounding events then threat analytics are a better choice because of their threat context.

Security personnel in industrial environments should considerthat even if the detections had a perfect true positive rate, the purpose of alerts in industrial environments almost always results in an investigation This is because it is critical to understand the context of the activity to determine root cause analysis and whether or not activity is simply malicious or intentional. Additionally, there are scenarios where malicious activity in an environment is an acceptable risk and can be addressed later whereas immediately blocking or stopping the activity could have adverse effects.

Vendors or complex technologies are not needed for con guration and indicator-based detection types; therefore, they should both be leveraged when possible. Modeling and behavioral analytics require dedicated personnel and technologies. All defenders should seek to take advantage of the appropriate detection type(s) for the right situation

To learn more about Dragos and our technology, services, and threat intelligence for the industrial community, please visit www.dragos.com.

DRAGOS

1745 Dorsey Rd
Hanover, Maryland 21076
info@dragos.com