VECTRA®
SECURITY THAT THINKS

WHITE PAPER

The data science behind Vectra AI threat detection models

TABLE OF CONTENTS

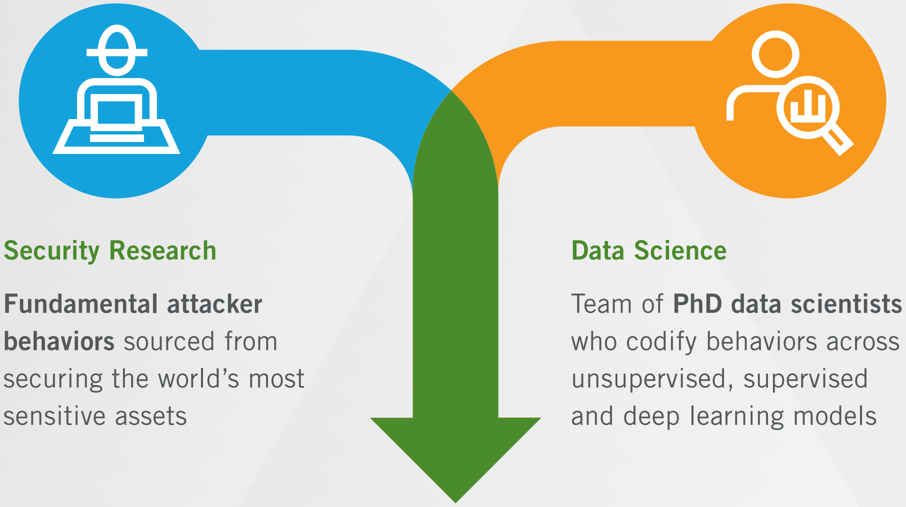
Vectra® protects business by detecting and stopping cyberattacks.

As a leader in network detection and response (NDR), Vectra® AI protects your data, systems and infrastructure. Vectra AI enables your SOC team to quickly discover and respond to would-be attackers —before they act.

Vectra AI rapidly identifies suspicious behavior and activity on your extended network, whether on-premises or in the cloud. Vectra will find it, flag it, and alert security personnel so they can respond immediately.

Vectra AI is *Security that thinks*®. It uses artificial intelligence to improve detection and response over time, eliminating false positives so you can focus on real threats.

Vectra® delivers a continuous cycle of threat intelligence and learning based on cutting-edge research, global learning models, and local learning models.



Security Research

Fundamental attacker behaviors sourced from securing the world's most sensitive assets

Data Science

Team of **PhD data scientists** who codify behaviors across unsupervised, supervised and deep learning models

Security Analyst in Software

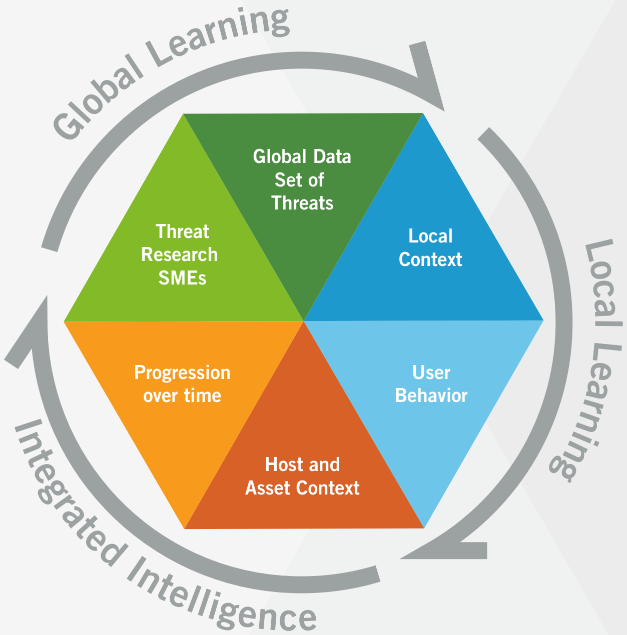
85% of the MITRE ATT&CK framework
Security **enrichments** (e.g. privilege)
Automate Tier-1 activities: **34X workload reduction**

Introduction

The Cognito® automated threat detection and response platform from Vectra® blends human expertise with a broad set of data science and machine learning techniques. This model delivers a continuous cycle of threat intelligence and learning based on cutting-edge research, global learning models, and local learning models.

With Cognito, these different perspectives combine to provide an ongoing, complete and integrated view that reveals complex multistage attacks as they unfold inside your network.

These unique stages of intelligence are essential to the detection of modern threats. This white paper explains how each critical stage contributes to the overall detection model, shows examples of specific detection techniques, and describes the various threats these detection techniques can find..



Global Learning

Identifying the fundamental traits that threats share in common

Techniques:

- Supervised machine learning, heuristics

Example:

- Random forest

Local Learning

Identify what is normal and abnormal in the local network

Techniques:

- Unsupervised machine learning, anomaly detection

Example:

- K-means clustering

Integrated Intelligence

Connect events to reveal the larger attack narrative

Techniques:

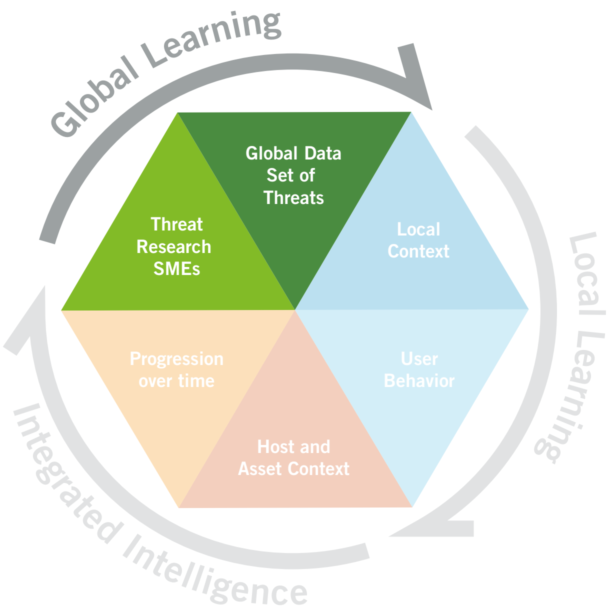
- Event correlation and host scoring

Example:

- Bayesian networks

Global learning

The Cognito AI model begins by detecting and analyzing as many threats as possible to identify the characteristics they have in common. It requires a large-scale analysis of many types of malicious traffic and content, and the expertise to find specific characteristics that truly matter.



Global Learning

Identifying the fundamental traits that threats share in common

Techniques:

- Supervised machine learning, heuristics

Example:

- Random forest

The human element

Although data science is at the heart of the Cognito AI model, the process begins with human subject-matter experts. The Vectra Threat Labs™ is a full-time group of cybersecurity experts and threat researchers who continually analyze malware, attack tools, techniques, and procedures to identify new and shifting trends in the threat landscape.

In addition to analyzing metadata provided by customers, the Vectra Threat Labs analyzes attacks collected in the wild and shared via research channels. Lab members use this real-world evidence to select the characteristics and behaviors of an attack that the data science models will be trained to detect.

Supervised machine learning

By understanding the role of human subject-matter experts, it’s possible to dive into the data science. In this global learning phase, the goal is to analyze very large volumes of malicious and attack traffic and distill it down to the key characteristics that make malicious traffic unique. This job is well-suited for supervised machine learning models.

Supervised, in this case, simply means that Vectra data scientists can dramatically improve a detection model by feeding it known bad traffic and known good traffic. Conversely, unsupervised machine learning models must augment their intelligence in a specific customer environment with no direct oversight by a data scientist.

For example, a supervised machine learning model can be designed to identify the unique behaviors of remote access tools (RATs.) By analyzing large numbers of RATs, a supervised machine learning model can learn how traffic from these tools differs from normal traffic.

This intelligence enables the Cognito AI model to reliably detect new, custom and otherwise unknown RATs in real time and without using signatures.

As mentioned earlier, supervised machine learning models are vital to any overall detection strategy. They don't have to baseline or learn what's normal in a real-world customer environment.

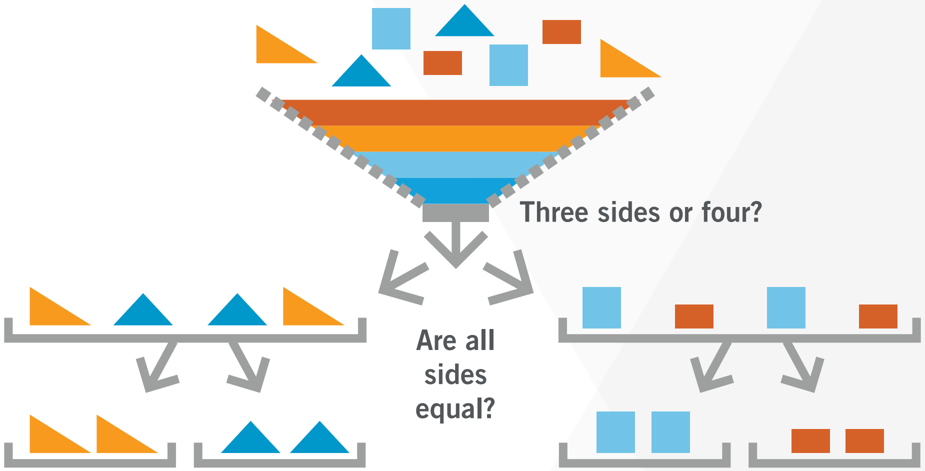
Supervised machine learning models also tend to be more directly correlated with malicious behavior compared to models that identify irregularities and anomalies. While an anomaly might be malicious or benign, an unknown outsider who is controlling a device inside your network represents a more serious risk.

Example: Random forest

One of the most common methods of supervised machine learning is the random forest. The Cognito AI model leverages random forests in a variety of ways that can be extremely useful in detecting active threats as well as revealing subtle traits that they have in common.

Understanding a random forest requires a basic understanding of decision trees. At a high level, a decision tree can use unique characteristics to classify or sort data into homogenous groups.

> Instead of relying on one tree, a random forest averages the results from a massive number of trees where each one tests a unique combination of data and characteristics.



A basic decision tree

For example, the image shown is a simplified decision tree that classifies different shapes. The shapes can be separated based on whether they have three or four sides. Those groups can be further separated based on whether the sides of the shape are of equal length. After this simple set of decisions, all the shapes are sorted into uniform groups.

The decision tree isn't quite so simple with cybersecurity. There are a wide variety of threats to analyze and they can be sorted using unlimited characteristics. A decision tree that includes all the ever-changing variants of malware traffic and their associated characteristics would be a formidable problem.

However, a random forest can help overcome this challenge. As its name implies, a random forest is a collection of many randomly generated decision trees. Instead of relying on one tree, a random forest averages the results from a massive number of trees where each one tests a unique combination of data and characteristics.

Instead of trying to build a detection model that knows how to identify one specific sample, a random forest puts an unknown sample through a battery of tests to determine whether it should be classified as good or bad.

Since the learning model is supervised, data scientists can measure how the model performs and further improve it by feeding it known good and known bad traffic. The model can be augmented continuously to accurately classify malicious traffic based on a variety of characteristics.

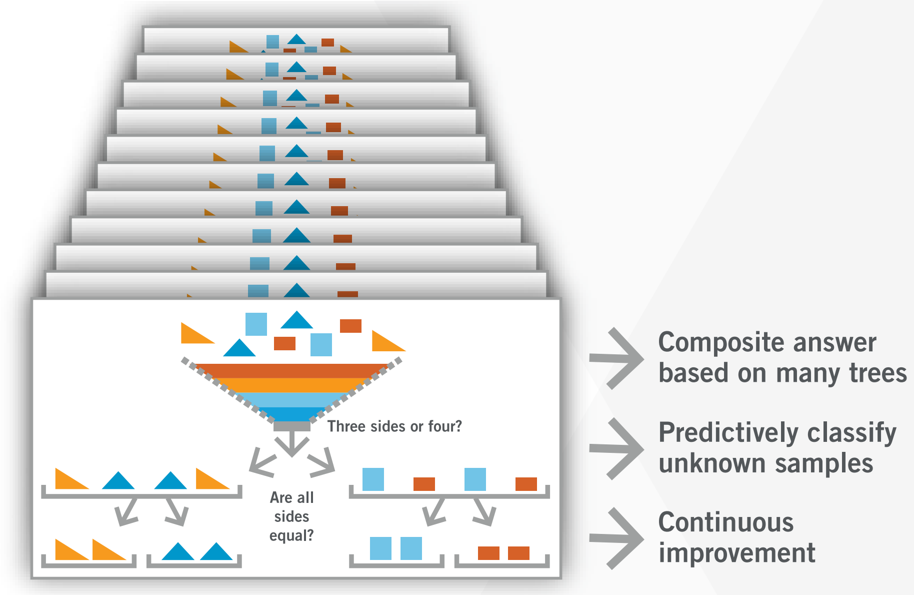
This approach offers some very important benefits. Random forest decision trees allow data scientists to analyze massive numbers of samples and find the traits they have in common.

The model continuously thinks and learns to determine which approaches and sets of characteristics give the best results.

As behaviors change in the wild, the model naturally adapts. This helps to uncover new features or traits that are predictive of malicious traffic that wouldn't be obvious in manual, human- driven investigations.

Most importantly, the model continuously builds a base of intelligence that classifies unknown traits, as shown below. Instead of finding the answers to the test, a random forest collects what has been learned from aggregate, large-scale analysis and applies that knowledge to a completely new detection – all in real time.

> A random forest collects what has been learned from aggregate, large-scale analysis and applies that knowledge to a completely new detection – all in real time.



Composite answer based on many trees

Predictively classify unknown samples

Continuous improvement

Example: Suspicious HTTP

A random forest is just one of the passel of tools at the disposal of data scientists and researchers in the Vectra Threat Labs. Many other detections incorporate multiple styles of analysis.

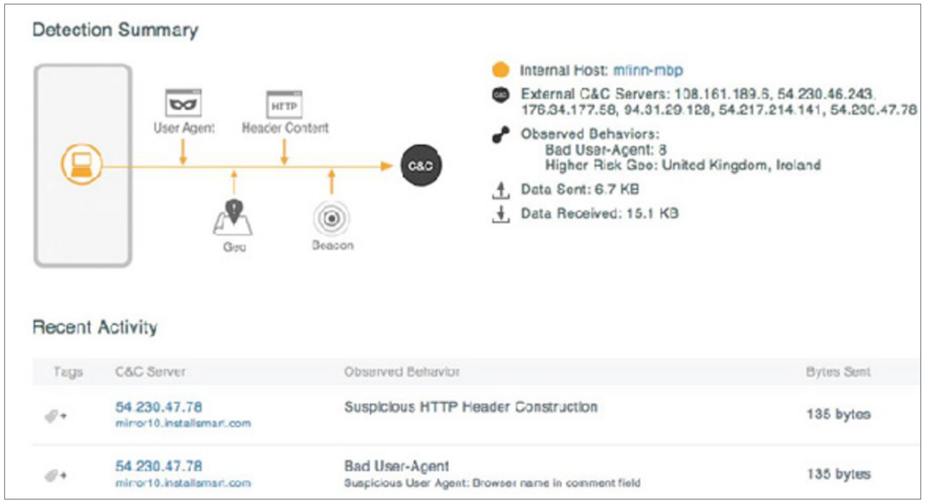
The suspicious HTTP detection is a good example. This detection reveals Web-based command-and-control traffic, but it solves the problem in several different ways. One way uses a random forest analysis of traits in HTTP headers to identify unique patterns of command-and-control behavior that don't exist in benign traffic.

By analyzing a wide range of command-and-control traffic, data scientists in the Vectra Threat Labs can focus on traits that are common across many types of malware.

Instead of trying to keep up with attackers as they change domains and IP addresses, the Cognito AI model quickly detects command-and-control traffic without using signatures.

Instead of trying to keep up with attackers as they change domains and IP addresses, the Cognito AI model quickly detects command-and-control traffic without using signatures.

This proven model works in real time and is accurate across the broad spectrum of malware, identifying botnets, banking malware, and even mobile malware equally well.



Detection Summary

Internal Host: mfinn-mbp

External C&C Servers: 108.161.189.6, 54.230.46.243, 176.34.177.58, 94.31.29.128, 54.217.214.141, 54.230.47.78

Observed Behaviors:
Bad User-Agent: 8
Higher Risk Geo: United Kingdom, Ireland

Data Sent: 6.7 KB

Data Received: 15.1 KB

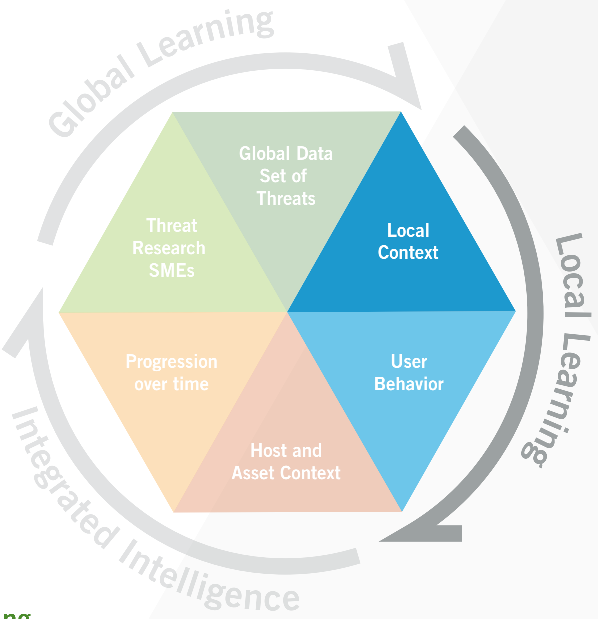
Recent Activity

Tags	C&C Server	Observed Behavior	Bytes Sent
+	54.230.47.78 mirror10.installsmart.com	Suspicious HTTP Header Construction	135 bytes
+	54.230.47.78 mirror10.installsmart.com	Bad User-Agent Suspicious User Agent: Browser name in comment field	135 bytes

Local learning

While global learning is critical, some things can only be learned based on local experiences. For instance, no amount of global analysis can tell you when employees behave differently than they have in the past.

The Cognito AI model employs a variety of different local learning techniques and approaches to instantly detect these types of anomalous behaviors.



Local Learning

Identify what is normal and abnormal in the local network

Techniques:

- Unsupervised machine learning, anomaly detection

Example:

- K-means clustering

Establishing local context

Vectra local learning models are logical complements to global AI. Instead of analyzing bad traffic to learn what threats have in common, local learning understands what makes the local network environment unique and identifies abnormalities when they occur.

But not every anomaly is an indicator of an attack. Employees are not automatons, and will behave unpredictably in the normal course of doing their jobs. As a result, Vectra looks for indicators of important phases of an attack or attack techniques, instead of concentrating on finding and reporting anomalies.

This includes looking for signs that an attacker is exploring the network, evaluating hosts for attack and using stolen credentials. These examples require different types of local context.

For example, a network host that tries to connect to unused IP addresses might indicate that it is unfamiliar with the local environment and is performing attack reconnaissance. Detecting these behaviors requires a long-term memory of the local network environment, including IP addresses that were used over time.

Furthermore, some malicious actions occur over multiple connections or steps inside a network. This might have nothing to do with established norms and instead may simply require an understanding of local context over time.

For instance, an attacker may steal data from an internal server and move it to another machine for staging. The progressive staging of data can be observed but it requires both memory and intelligence to build context and recognize that similar amounts of data are being staged and exfiltrated at different times.

Unsupervised machine learning

Local learning techniques are very different from global learning. The biggest difference is that local learning models can’t rely on the direct oversight of a data scientist. Unsupervised machine learning models fix this by collecting knowledge of local norms and then detecting deviations from those norms.

One way to learn these norms is to use clustering. There are many different clustering techniques and approaches, but the basic concept is to find natural groupings of data. For example, consider the simple set of data points below.

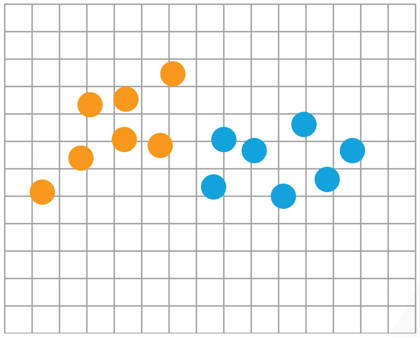


Fig1. A simple set of data points

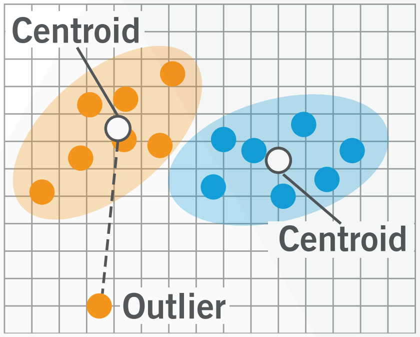


Fig2. Two groups separated by color and a distinct visual boundary.

There are two groups separated by color and a distinct visual boundary. The center of mass in each group is called the centroid. As reference points for each group, the centroid is used to evaluate new data points and determine how much they deviate from the norm. This simple visual makes it easy to identify outliers in virtually any behavior.

However, data sets are not always so clean and there can be a significant number of variables that must be tracked. The challenge is teaching the machines to properly identify clusters within the sea of data.

One solution involves using K-means clustering. The benefit of K-means clustering is that the system doesn't have to be told which group a data point belongs to – it figures that out on its own. In the previous example, K-means clustering is applied by removing the orange and blue data-point distinctions, as shown below.

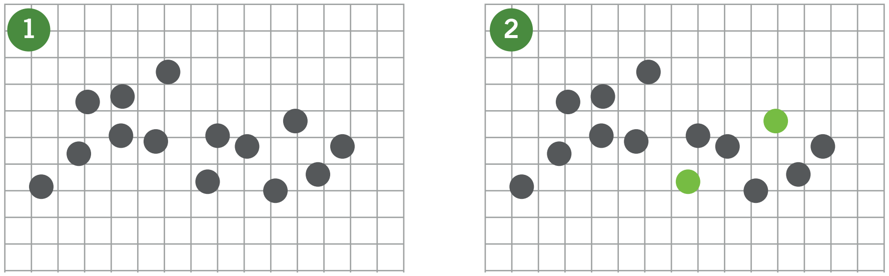


Fig3. K-means clustering removes the orange and blue distinction and two points are randomly selected

To classify the points into two groups, two points can be randomly selected, as highlighted above. By chance, both reference points are within the same cluster. The K-means clustering model will learn and fix this problem on its own.

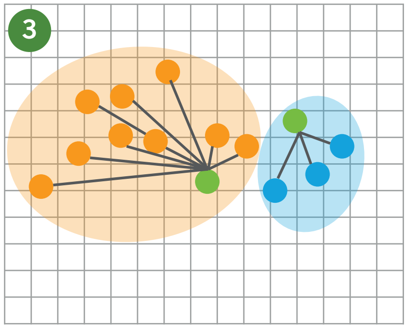


Fig4. Data points are classified based on closest reference points

Starting with these two reference points, all other data points are classified based on which of the two reference points is closest. This results in the creation of two groups.

Although the groups aren't accurate yet, they can be used to identify a new centroid.

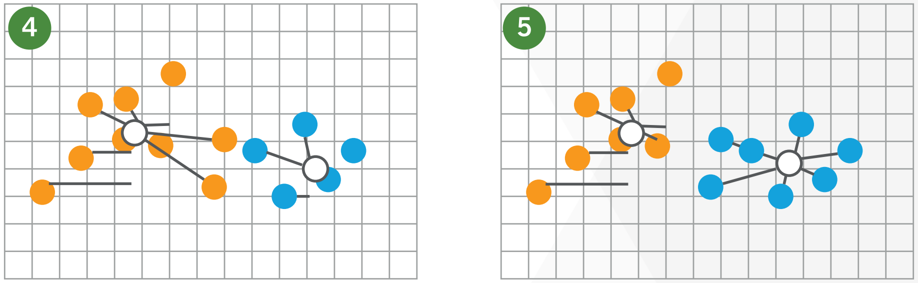


Fig5. The data points are used to identify a new center and the K-means clustering model stabilizes

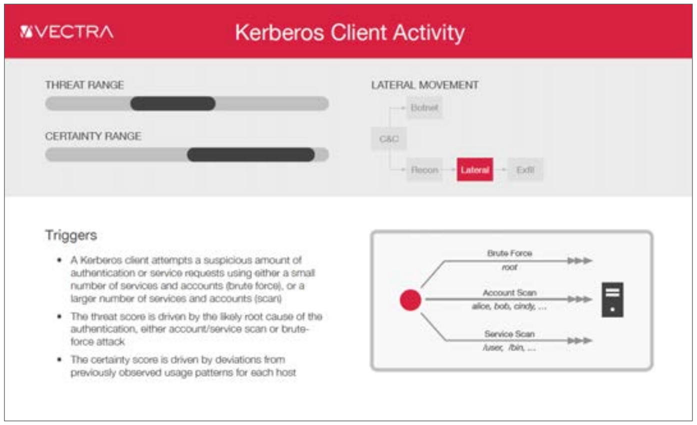
The process is repeated, measuring each data point from the new center. By calculating the new center and remeasuring, some data points that were originally in the orange group on the left are moved to the blue group on the right.

After a few iterations, the K-means clustering model stabilizes and the centroids no longer move. At this stage, the data points become accurately grouped into their appropriate clusters.

Consequently, K-means clustering enables the creation of data groups based on virtually any action or behavioral characteristic in the network. These automatically learned clusters can then be used to identify deviations in behavior that indicate an attack. This approach easily learn norms for any observable trait in the network.

Example: Kerberos client detection

K-means clustering can be used to identify the theft of valid credentials from a compromised host. This includes pass-the-hash and golden ticket techniques that involve stealing tokens from trusted host devices or creating fake ones. Keylogging malware can also be used to steal user names and passwords.



Kerberos Client Activity

Triggers

- A Kerberos client attempts a suspicious amount of authentication or service requests using either a small number of services and accounts (brute force), or a larger number of services and accounts (scan)
- The threat score is driven by the likely root cause of the authentication, either account/service scan or brute-force attack
- The certainty score is driven by deviations from previously observed usage patterns for each host

To identify these threats, Cognito monitors Kerberos traffic to establish norms for a variety of behaviors on every host in the network. This may include user accounts on each host as well as the services they typically request.

When a user credential is compromised, a trusted user account can be spotted being used on new hosts. Likewise, the same user account may begin requesting a variety of new services.

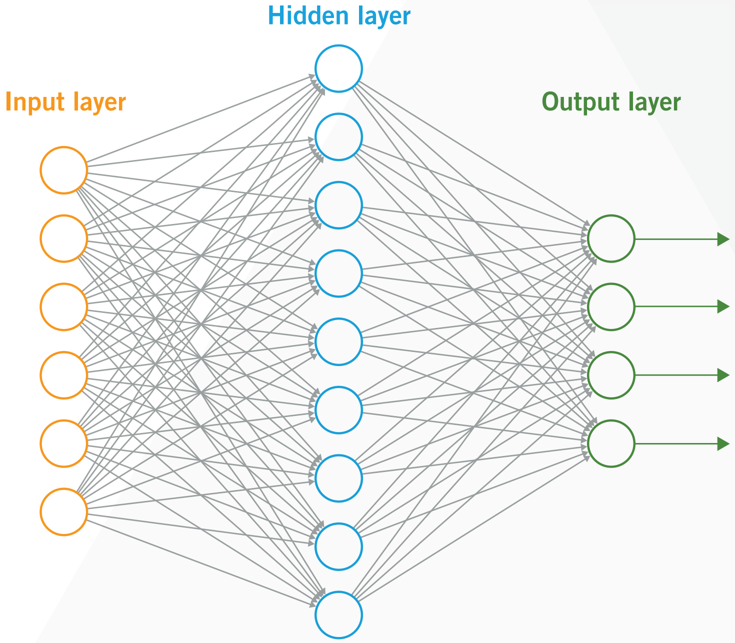
With clustering and unsupervised machine learning, Cognito reveals in real time when credentials are stolen. Further analysis can reveal what type of malicious behavior is occurring, if the attacker is trying to enumerate a user account or services, and whether the attacker has gained access to privileged areas of the network.

Deep learning

Deep learning is a branch of machine learning techniques that has led to the rapid advancement of natural language processing, translation, image recognition, and other fields. It is rooted in mathematics that were explored in the 1940s but its potential has only recently been tapped due to advances in modern computing power.

Based on neural networks

Inspired by the biological structure and function of neurons in the brain, deep learning techniques rely on large, interconnected networks of artificial neurons. These neurons are organized into layers, with individual neurons connected to one another by a set of weights that adapt in response to newly arriving inputs.



Although there are a wide range of network architectures, the depth of a network generally refers to the number of layers in the model and the number of processing-time steps in the case of recurrent networks (i.e., models can be deep in both space and time).

For a variety of reasons, the greater the depth of the network, the more difficult a model is to train, although much of the rapid development in the field is due to techniques and mathematical analysis of how these difficulties can be overcome.

More traditional machine learning methods like the decision tree, random forest and K-means clustering require data scientists to carefully and manually select features on the input data to the model to derive meaningful results. In the context of looking for attacker behaviors in network traffic, there are large lists of potentially relevant features.

For example, when looking at remote procedure calls within the SMB protocol, some of the relevant features are the SMB UUID, named pipe and account, but many other less-relevant features are present. Selecting features for doing image recognition of handwriting may be more difficult.

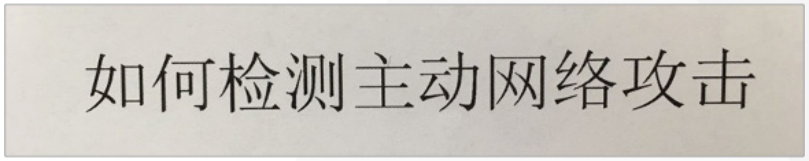
However, neural networks can learn relevant features from a data set and build increasingly complex representations of these features as information flows into higher network layers. These representations are learned rather than predetermined by data scientists, making them powerful for various classes of problems.

In a supervised setting, a large set of well-labeled data is used as input to the network. At the same time, an algorithmic training process adjusts the model's weights so it will be less likely to make a prediction error if it ever sees similar inputs in the future. This makes the input feature selection less burdensome because part of the feature extraction is baked into the development process.

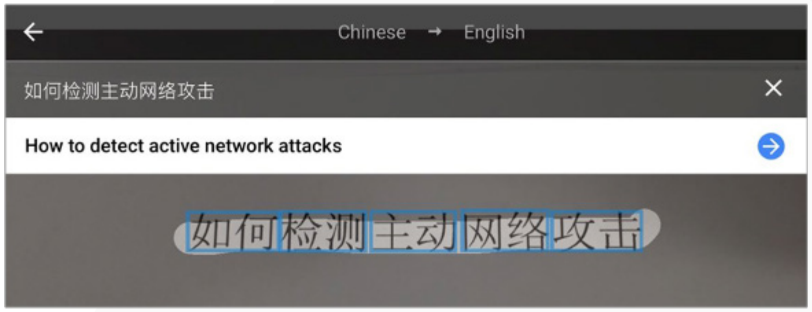
Example: How deep learning is used today

Deep learning is making an impact in language translation, autonomous driving, cybersecurity and many other fields. Google Translate, the popular multilingual machine translation service, provides an excellent example of deep learning in action.

An iPhone camera and the Google Translate app were used to capture an image of the text below in simplified Chinese. Google Translate, which leverages deep learning, aptly recognizes the simplified Chinese characters and accurately translates them into English.



如何检测主动网络攻击



如何检测主动网络攻击

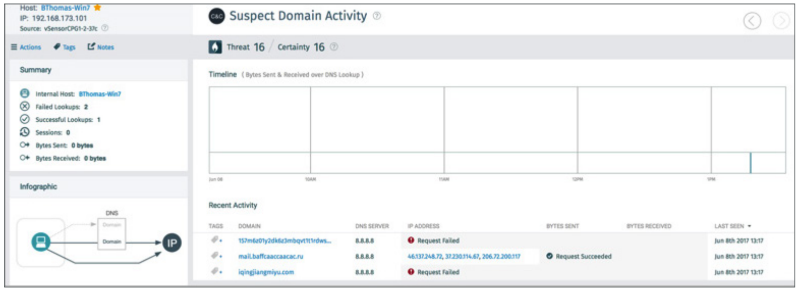
How to detect active network attacks

Example: How Vectra uses deep learning

In the case of suspect domain activity, Vectra uses deep learning to detect algorithmically-generated domains that cyberattackers set up as the front-end of their command-and-control infrastructure.

Freshly registered domains consisting of random characters give attackers an effective way to obfuscate their command-and-control infrastructure. Malware will then attempt to find these freshly registered random domains when phoning home.

By parsing DNS request traffic, Cognito can inspect and classify a given domain name whether it looks like one that has been algorithmically generated or not. To do this, Cognito utilizes recurrent neural networks that are extremely effective at learning and predicting what will come next in a sequence of things.



Host: BThomas-Win7 ★
IP: 192.168.173.101
Source: vSensorCPG1-2-37c

Summary

Internal Host: BThomas-Win7
Failed Lookups: 2
Successful Lookups: 1
Sessions: 0
Bytes Sent: 0 bytes
Bytes Received: 0 bytes

Suspect Domain Activity

Threat 16 / Certainty 16

Timeline (Bytes Sent & Received over DNS Lookup)

Recent Activity

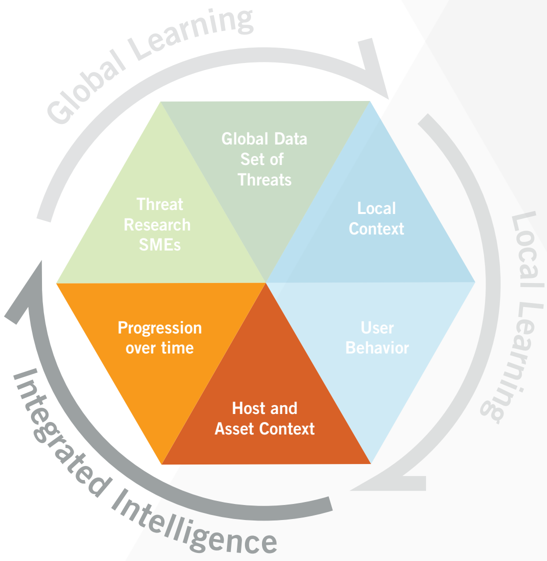
TAGS	DOMAIN	DNS SERVER	IP ADDRESS	BYTES SENT	BYTES RECEIVED	LAST SEEN ▾
+	157m6z01y2dk6z3mbqvt1t1rdws...	8.8.8.8	❗ Request Failed			Jun 8th 2017 13:17
+	mail.baffcaaccaacac.ru	8.8.8.8	46.137.248.72, 37.230.114.67, 206.72.200.117	✓ Request Succeeded		Jun 8th 2017 13:17
+	iqingjiangmiyu.com	8.8.8.8	❗ Request Failed			Jun 8th 2017 13:17

These sequences might be words in a sentence or sentences in a paragraph, or in the case of domain name analysis, the letters in a string. When the actual sequence of letters begins to diverge widely from the predicted and expected next letters, the probability increases that the domain was algorithmically generated.

The memory-like aspect of the neural network – imparted by its many nodes, layers and recurrent connections – is what enables it to predict what letter is likely to come next based on what it has seen, ultimately giving Vectra the ability to classify a domain as suspect or not.

Today's cyberattacks are complex, multistage operations that evolve over time and encompass a variety of techniques and strategies that help attackers move deeper into the network.

As a result, it is critically important for threat detection models to have the intelligence to assimilate all of the available information to identify the larger attack, not just the component events.



Integrated Intelligence

Connect events to reveal the larger attack narrative

Techniques:

- Event correlation and host scoring

Example:

- Bayesian networks

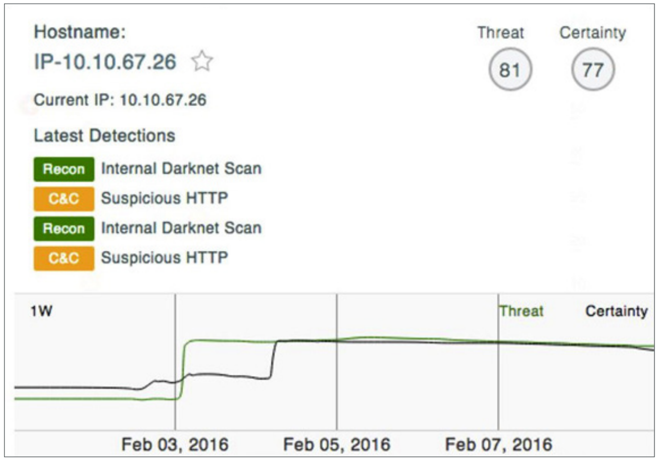
Tracking the attack progression over time

To detect patient, slow-moving attacks, detection models must have long-term memory to retain context over time.

The Cognito AI model acquires this vital context by tracking events over time and through every strategic phase of the cyberattack kill chain.

All detection events are correlated to specific hosts that show signs of threat behaviors. In turn, all context is assimilated into an up-to-the-moment score of the overall risk to the organization.

The term low-and-slow is often used to describe modern cyberattacks. It's a sobering reminder that time is a fundamental component of today's highly sophisticated cyberthreats.



Hostname:
IP-10.10.67.26

Threat: 81
Certainty: 77

Current IP: 10.10.67.26

Latest Detections

Recon	Internal Darknet Scan
C&C	Suspicious HTTP
Recon	Internal Darknet Scan
C&C	Suspicious HTTP

To detect patient, slow-moving attacks, detection models must have long-term memory to retain context over time.

Cognito automatically scores every detection and host in terms of the threat severity and certainty and tracks each one over time. Threat and certainty scores are visible so security teams can instantly see the progression of analysis over time.

The image to the left shows the threat detection details of a specific host and the progression of its threat and certainty scores over time.

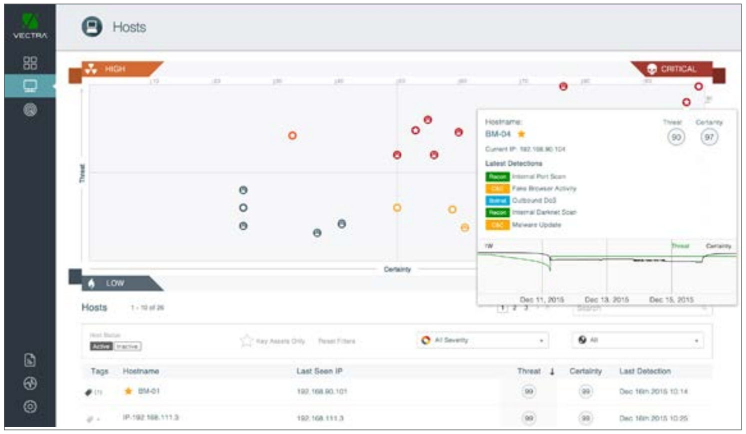
In addition to tracking events over time, Cognito prioritizes the events that pose the greatest risk to an organization. This risk is determined in a variety of ways, but special attention is paid to:

1. Events that are of strategic value to an attacker
2. Devices that exhibit behaviors that cover multiple phases of the cyber attack kill chain
3. Events that may jeopardize key assets inside the network

For example, a botnet that sends spam from your network is potentially damaging to your reputation, but the risk is considerably lower than an attack that spreads malware to hosts in your network.

It is especially important to know when an attack progresses from one phase of the kill chain to another. An attack that advances from the internal reconnaissance phase to the lateral movement phase is more significant than the sum of its parts, and the Vectra scoring model accounts for this significance.

In addition, not all hosts in the network have equal importance. For instance, servers with sensitive data or a CFO's laptop can be considered vital assets. Threat events related to these assets should be prioritized to understand the risk in context.



The Cognito AI model considers all these aspects and offers prioritized threat scores that understand the strategic nature of an attack and the context of hosts under attack. The result is a clear network view that puts high-risk hosts in the top right of the user interface, as shown above.

Example: Bayesian networks

Bayesian networks – powerful tools that are used for everything from diagnosing diseases to predicting financial markets – represent a technique that enables threat and certainty scoring for compromised hosts.

At their core, Bayesian networks, or Bayes nets, show the probabilistic relationships between events or a of series events. A probabilistic relationship can be as simple as the relationship between cloudy weather and rain or having a cough and coming down with an illness.

The value of Bayesian networks is in their ability to consider complex systems with many independent variables and understand the statistical linkages. For example, a doctor's patient may have symptoms that point to a variety of underlying illnesses.

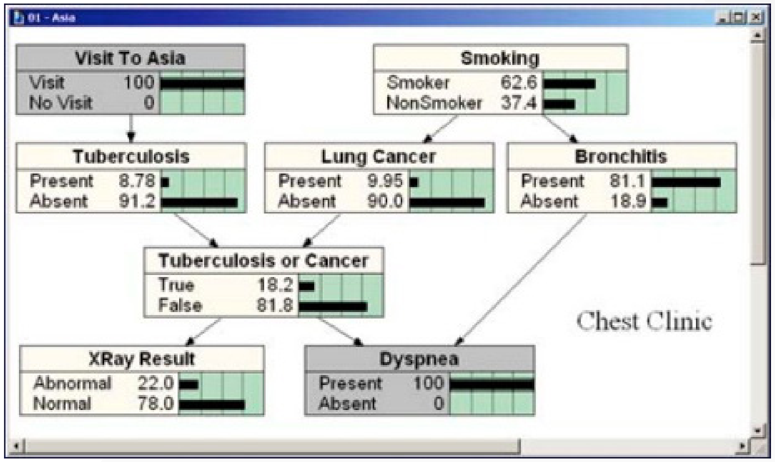
Bayesian networks can take a variety of data such as the prevalence of each illness, the patient's symptoms, environmental factors, and medical history to reveal the probabilities of specific diseases.

The graphic below shows how a Bayesian network can help make smarter decisions about diagnosing a patient with shortness of breath. Some information is known to the physician and some is not.

In the example, the physician knows that the patient has dyspnea, or shortness of breath, and has visited the Asian continent. The physician also knows the prevalence of each possible disease population.

But most important, Bayesian networks understand statistical linkages, such as the link between smoking and lung cancer. Before an X-ray is taken, the Bayesian network example shows statistics on the most probable diagnosis – bronchitis.

As more data becomes available, such as when an X-ray is performed, the results will automatically percolate through the Bayesian network to update all the various probabilities.



Source: Norsys Software Corp.

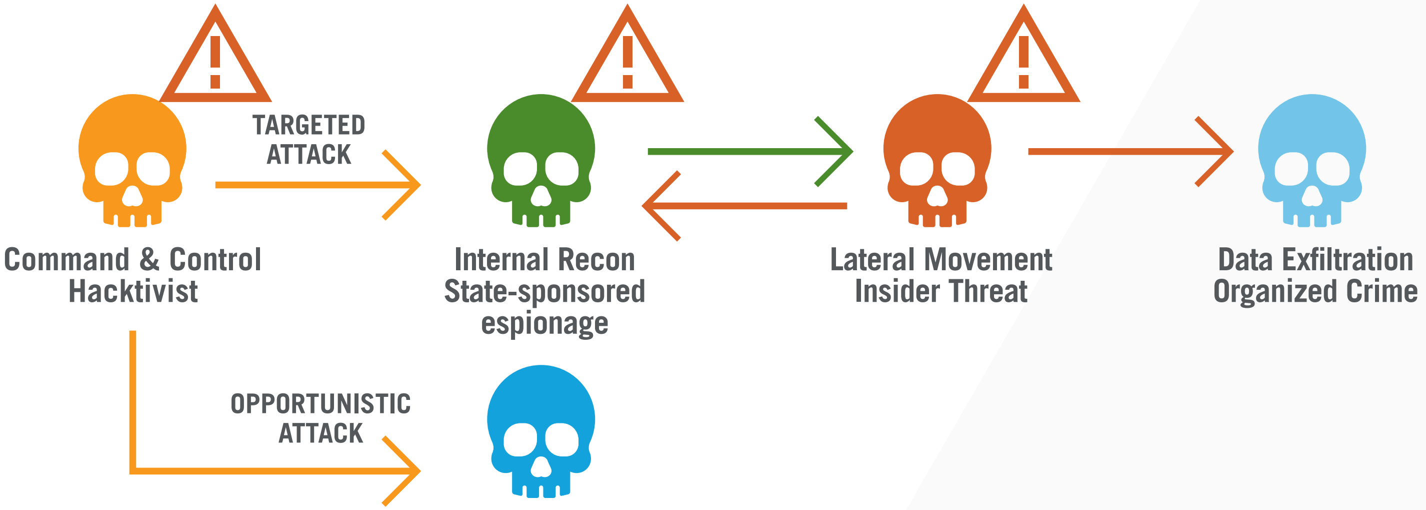
The Cognito AI model uses a similar technique when generating detection scores and host scores. Detections like suspicious HTTP leverage many different techniques and use an internal correlation engine to assimilate all the information.

On a larger scale, Cognito uses Bayesian networks to understand the linkages between multiple detections in order to identify and prioritize a targeted attack. Below, the cyberattack kill-chain diagram shows how some events are more indicative of a targeted attack than others.

For example, botnet monetization may indicate the presence of crimeware but not a targeted attack. However, internal reconnaissance and lateral movement behaviors are strongly linked to targeted attacks.

In Bayesian network terms, spam emails have a lower linkage to targeted attacks when compared to a host that exhibits staged transfers of data or data smuggling behaviors.

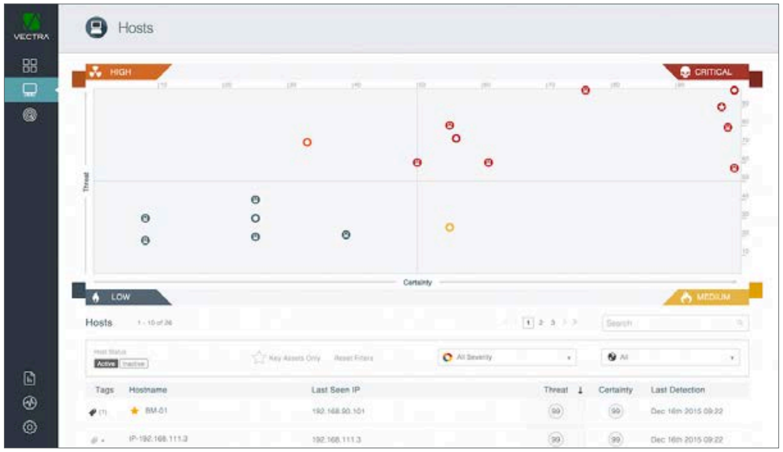
If a host exhibits command-and-control communication, internal reconnaissance and lateral movement behaviors, the risk of a targeted attack becomes more likely.



The Cognito AI model also assigns particular importance to behaviors that progress across strategic phases of the kill chain. If a host exhibits command-and-control communication, internal reconnaissance and lateral movement behaviors, the risk of a targeted attack becomes more likely.

While the use of Bayesian networks is not quite as simple as some of these examples, it offers a framework for understanding how Cognito identifies the linkages between a wide range of events, hosts and detection methods to automatically elevate coordinated attacks.

Although the underlying calculations are quite complicated, these combined methods enable Cognito to prioritize attack behaviors and hosts that represent the most serious threats with the highest certainty in the network, as shown below.



Hosts 1 - 10 of 26

1 2 3

Search

Host Status: Active Inactive

Key Assets Only   Reset Filters

All Severity

All

Tags	Hostname	Last Seen IP	Threat	Certainty	Last Detection
(1)	★ BM-01	192.168.90.101	99	99	Dec 16th 2015 09:22
	IP-192.168.111.3	192.168.111.3	99	99	Dec 16th 2015 09:22

For more information please contact a service representative at sales-inquiries@vectra.ai.

Leveraging packet-level behavioral analysis, this new model uniquely combines human expertise, global learning and local learning in one simple, integrated solution that greatly improves security operations.

Conclusion

The data science behind the Cognito AI model represents an unprecedented innovation in detection methodologies.

Leveraging packet-level behavioral analysis, this new model uniquely combines human expertise, global learning and local learning in one simple, integrated solution that greatly improves security operations.

Detecting the plethora of new, highly sophisticated cyberattacks continues to be a challenging prospect. But Cognito remains steadfast in its commitment to the development of new tools and methodologies that detect all manner of threats well into the future.

Email info@vectra.ai   vectra.ai