WHITE PAPER

Detect threats faster with machine learning

How Plixer uses machine learning for security

Plixer

What is machine learning & why is it important?

Imagine trying to make sense of the millions of data points created by each network device interaction. With the size of today's typical enterprise network, the sheer number of people and hours needed to analyze a single day of interactions only emphasizes the impracticability of the task.

Sophisticated attacks are taking advantage of our inability to manually make sense of network behavior data. Cybercriminals can bypass other security controls and then mask their behavior on the network by moving slowly and in ways that appear normal to the human eye. By appearing normal, attackers can compromise more areas of your network, exfiltrate more data, and cause more damage.

Luckily, you do not have to rely on the human eye to spot abnormalities or a spreadsheet to process all your network data. You can stop attackers by having a machine learning engine process and analyze the network behavior data.

An ML engine can ingest all your network flow data and see the underlying behavior patterns to make sense of the data and refine its understanding over time to provide actionable insights. In short, the ML engine adapts to the unique behaviors on your network to provide alerts about potential issues. This allows ML to provide intelligent threat detection of even the most sophisticated attacks that try to mimic normal device activity.

An ML engine then applies additional AI logic to dynamically eliminate alarms that can be explained and provide relief to under-resourced teams. These advanced capabilities elevate true versus false positives and provide the contextual information needed to resolve problems quickly. With ML you can more easily identify issues, investigate root-cause, and respond before your business can be disrupted.

Types of machine learning

While most people have a general sense of ML and may even employ tools that use ML, they may be unsure how to tell the sophisticated ML applications from the simpler applications. While there is not really a battle of good versus bad ML, some security solutions rely on basic ML, while others use more sophisticated applications of AI to detect, investigate, and respond to threats.

In general, all ML is used to continuously better the performance of a desired function. For example, the desired function of an ML engine could be to trigger network alarms (for either security or performance monitoring) about abnormal device behavior. Or the desired function could be to forecast future network behavior based on current patterns.

There are three main ways that ML can learn and increase detection accuracy:

- **Supervised learning**: the process of training an ML on specific datasets to achieve a desired result—i.e., identify characteristics of malware
- **Unsupervised learning**: the process of ingesting new datasets and allowing the ML to make its own connections
- **Deep learning**: A progression of supervised and unsupervised learning to create an artificial neural network that can learn and make intelligent decisions on its own

Overview of how Plixer's NDR platform uses machine learning

The Plixer NDR platform leverages ML to help security teams in a few important ways. The ML engine ingests network flow data to establish a visualization of expected behavior on the network (who talks to whom, which applications are being used and in what ways, who's present on the network at which times, etc.). After about a week of observing the network, the Plixer ML engine has learned to distinguish normal from abnormal traffic behavior. To improve the dynamic model of what's considered normal, we also account for granular configurations, such as subnet activity, custom sensitivity thresholds and seasonality behaviors— to account for different behaviors on workdays, nights, and weekends. Customers can also create custom ML definitions to fit their specific needs and further refine alarm accuracy. The ML models are then regenerated every 24 hours to maintain accuracy.

By processing network flow data, the Plixer NDR platform provides intelligent threat detection. Rather than hunting for indications of a hack after a breach has occurred, the ML engine detects the tactics, techniques, and procedures a bad actor must take to compromise the network in real-time. Because it continuously establishes a baseline for what normal traffic looks like, the ML engine can

quickly detect abnormal behaviors like data accumulation, data exfiltration, brute force, tunneling, worm detection, and lateral movements. Once detected, the Plixer NDR platform maps these to the MITRE ATT&CK framework. With automated forensics, your team can quickly dig into the detection to follow the metadata trail of the attack and gauge the severity of the threat.

In addition, we've tuned the ML engine to help detect common malware classifications. We trained our ML engine to spot suspicious behaviors by various malware families based on the classification of commonly observed network traffic behaviors. Introducing malware behavioral detections to the ML engine allowed it to more readily find the behavior of devices trying to compromise the network.

The goal of ML is to shorten attacker dwell time. Powerful and intelligent threat detection allows you to remediate compromises on your network before they become disruptive and costly.

A technical dive into Plixer's ML engine

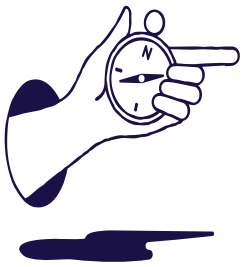
Plixer uses a combination of supervised, unsupervised, and deep learning to power our ML engine. For actionable network intelligence, it's important to have ML process flow data from your network, as this provides the most accurate and far-reaching dataset for analyzing network behavior. Each network is too unique to be analyzed from a generic dataset, and processing only a portion through packet data can leave you blind to areas of your network.

How Plixer uses supervised learning

For threat detection, an ML engine can add great value if it has been trained to recognize common characteristics of malware. By quickly spotting malware, you can greatly reduce mean time to resolution (MTTR) and the risk of a significant data breach. Plixer helps you achieve faster malware detection by training our ML engine through supervised learning.

In general, supervised learning is best used for predicting answers to classification and regression questions. Both classification and regression perform predictive modeling, but the output of classification is a label, while the output of regression is a quantity. Classification could be used to answer questions like: what genre of music is this?; while regression would be used to answer questions like: how much does this house cost?

Plixer uses supervised learning to perform traffic classifications and detect different types of malware, such as banking trojans, command and control clients, coin mining traffic, and more. We do this by collecting a wide variety of malware families and inject each malware strain in an isolated lab one by one. This lab is meant to simulate typical enterprise traffic levels. Once the malware is present, we then collect the traffic, generate thousands of features based on the traffic, and create models using gradient boosted trees that will detect similar traffic behaviors. This is how Plixer's ML engine is trained to detect malware.



Once our ML engine is on your network, it will continuously ingest and analyze traffic. If it detects similar traffic behavior as seen by any one of the malware strains tested in our lab, it will trigger a malware alarm. You can then investigate the location and spread of malware and respond quickly.

How Plixer uses unsupervised learning

The devices on your network generate thousands of data points every day, each one telling a story about the behavior on your network. And while attackers may be able to hide from the human eye, they cannot hide from the network. By continuously ingesting and analyzing network behavior data, you can detect stealthy attacks.

In general, unsupervised learning is used for finding unknown patterns in large data sets. It is called "unsupervised" because traffic patterns don't need to be identified and labeled in advance (usually done manually by a human). This type of machine learning is ideal for processing large amounts of computer-generated data.

Plixer uses unsupervised learning to determine a dynamic threshold for normal network behavior on your specific network. When a threat, such as ransomware, is present on the network, the traffic pattern will be different than what is normally seen, and an alert will be triggered. This is what we call anomalous behavior.

For anomaly detection, we use a K-means clustering algorithm. This algorithm groups the behaviors into common clusters—a dynamic view of normal traffic. When new data comes in, the new data is compared to the common behaviors present in the K-means

clustering algorithm. If the new data shows enough deviation from the common behaviors, an anomaly alert is generated. Plixer then performs automated forensics to determine if the behavior matches activities like lateral movement, data exfiltration, data accumulation, brute force attacks, tunneling of data, and more.

Plixer's ML builds models of network traffic behavior every day to keep our anomaly detections as accurate as possible. Additionally, as mentioned above, we've added seasonality considerations by having the ML engine build three different models for weekday office hours, weekday office nights, and weekends. The ML models use time context to adapt as your network changes over nights/weekends for optimal accuracy.

How Plixer uses deep learning

As threats become more sophisticated, they will harness more methods that help them blend in. A way to fight this is to create a neural network that learns from the patterns generated by the behavioral data moving across your network. In essence, you need to create an analysis that is too sophisticated to hide from.

Deep learning is a newer application of machine learning. In general, deep learning is best used for processing large amounts of abstract data to be patterned using multiple processing layers

Plixer uses deep learning to perform link prediction between devices on your network. Link prediction provides another means of anomaly detection, but instead of monitoring a single behavior on a device, link prediction monitors the way a device interacts with all other devices. If a device interaction deviates from what the deep learning neural network is used to seeing, an anomaly is triggered.

The process of mapping the anomaly to an activity is followed as described above. But an additional step is taken. The device exhibiting uncommon behavior is added to an "endpoint monitoring" protocol. This allows you to closely monitor devices acting unusual. In some cases, you may want to take immediate action, but in other cases you may just want to keep tabs on the device.

Summary

Using ML to secure and strengthen your network gives your organization a competitive edge. The benefits of sophisticated ML are unmatched. When using a tool that harnesses a combination of supervised, unsupervised, and deep learning, your network security and performance efforts can be greatly augmented for more efficient and effective work.

In short, ML ingests network flow data to determine device and application behavior. With a strong, dynamic baseline, you can be alerted to unusual or suspicious behavior. With ML, you can reduce false positives and shorten the dwell time for threat detection, investigation, and response.

About Plixer

Plixer provides a single platform for network security and monitoring, delivering the insight and analytics needed to manage the immense opportunities and risks of a digital business. As a leader in the Network Detection and Response & the Network Performance Monitoring and Diagnostics markets, Plixer provides the comprehensive intelligence and visibility needed to analyze, evaluate, and visualize the millions of conversations that cross networks every second.