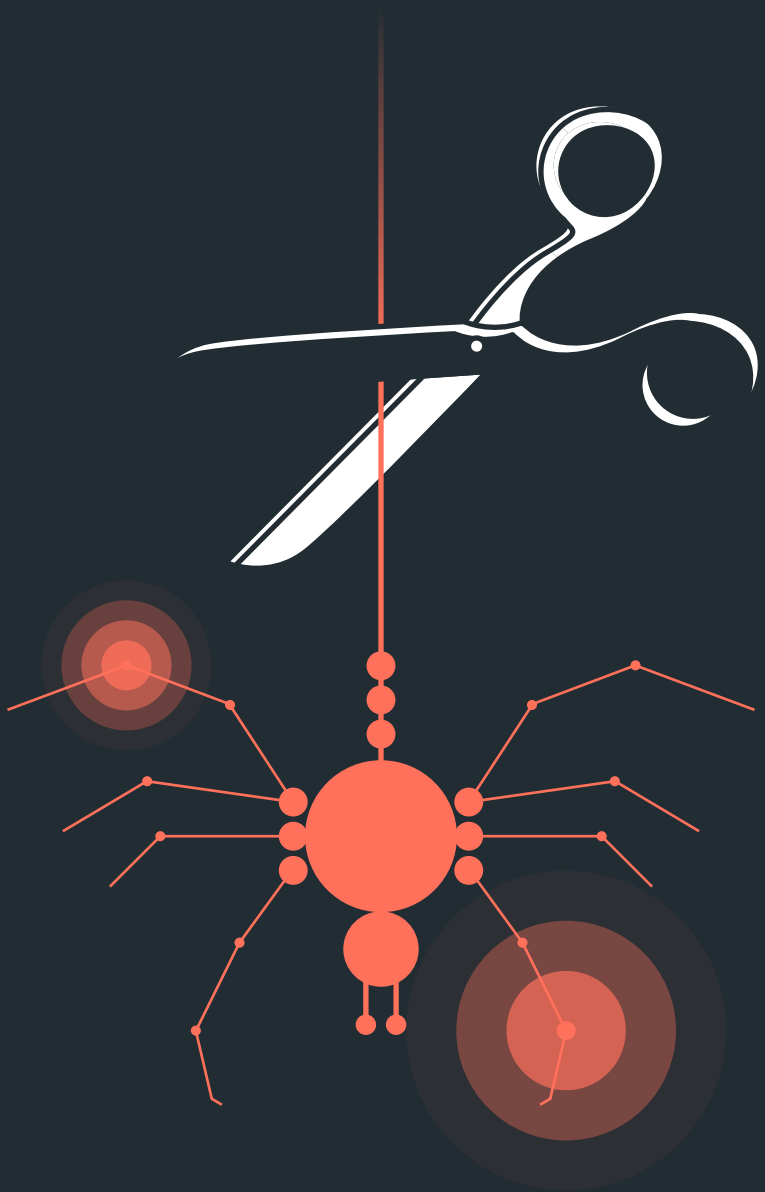
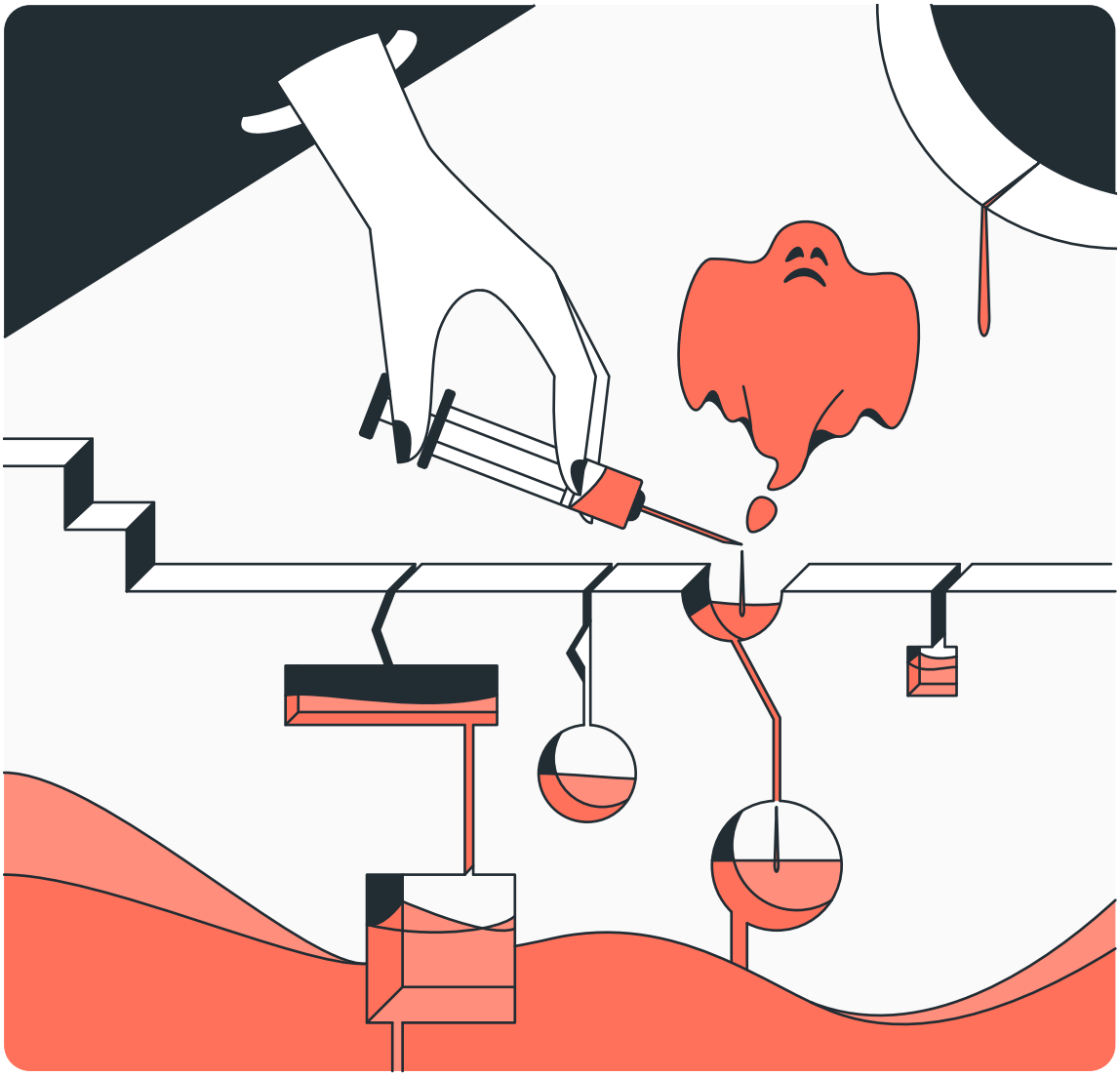
CATO
NETWORKS

Stop Playing in the Sandbox, Prevent Zero-Day Threats in Real Time



Zero-Day Threats & Polymorphic Malware

Malware has been around and evolving since the emergence of Elk Cloner in 1982 and Brain in 1986. Fast forward to today and malware is a booming industry with many variants ranging from viruses to ransomware, spyware and more. Motivation for creating and distributing malware is often financial but can also be combined with political or ideological factors. Countries such as Russia, North Korea, Iran and China have participated as nation-state actors to further their national interests or generate profit.

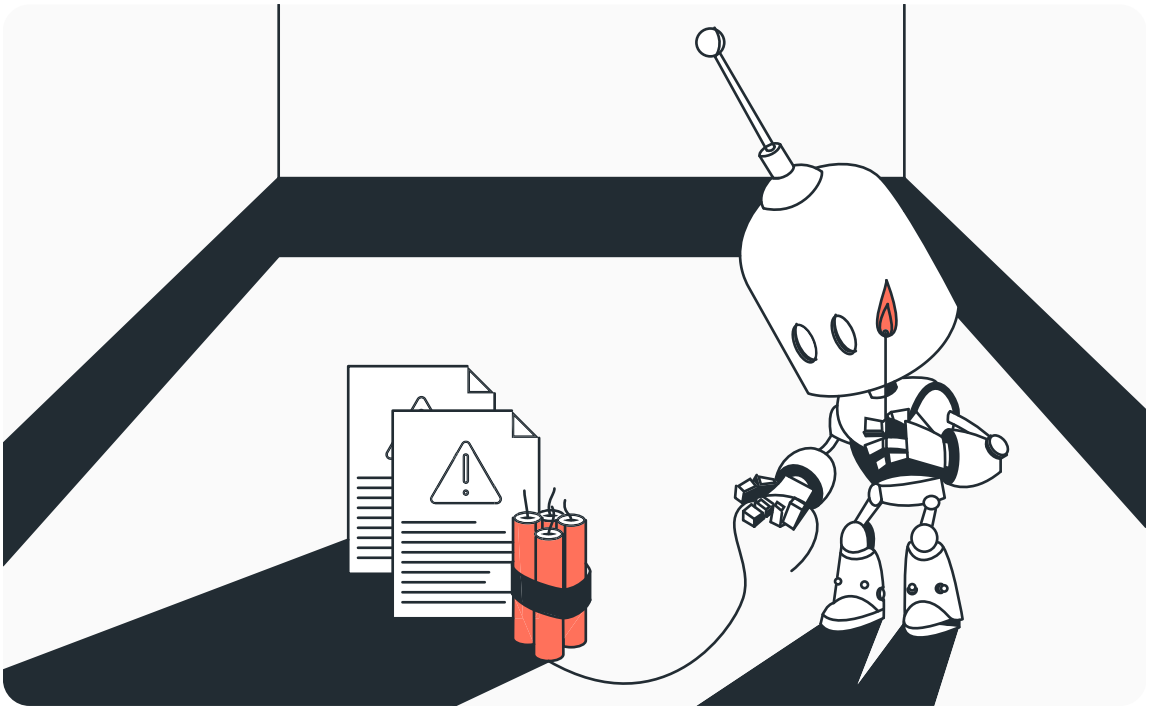


Zero-day vulnerabilities are unidentified vulnerabilities for which no patch exists. These vulnerabilities are typically exploited by threat actors to inject malware. Polymorphic malware morphs or changes in numerous ways to avoid detection. Because of the nature of both types of threats, they can be difficult or impossible to detect with traditional anti-malware engines. See the appendix for examples of well-known Zero-day attacks.

Prevention with Sandboxing

As malware has evolved and become more widespread, so have the tools used by information security teams to provide protection and prevention. Picking up where IPS and anti-malware leaves off, the most common zero-day and polymorphic malware detection tool is the sandbox. Sandboxing refers to the utilization of isolated environments to execute unknown or suspicious files to determine if its content is malicious or benign.

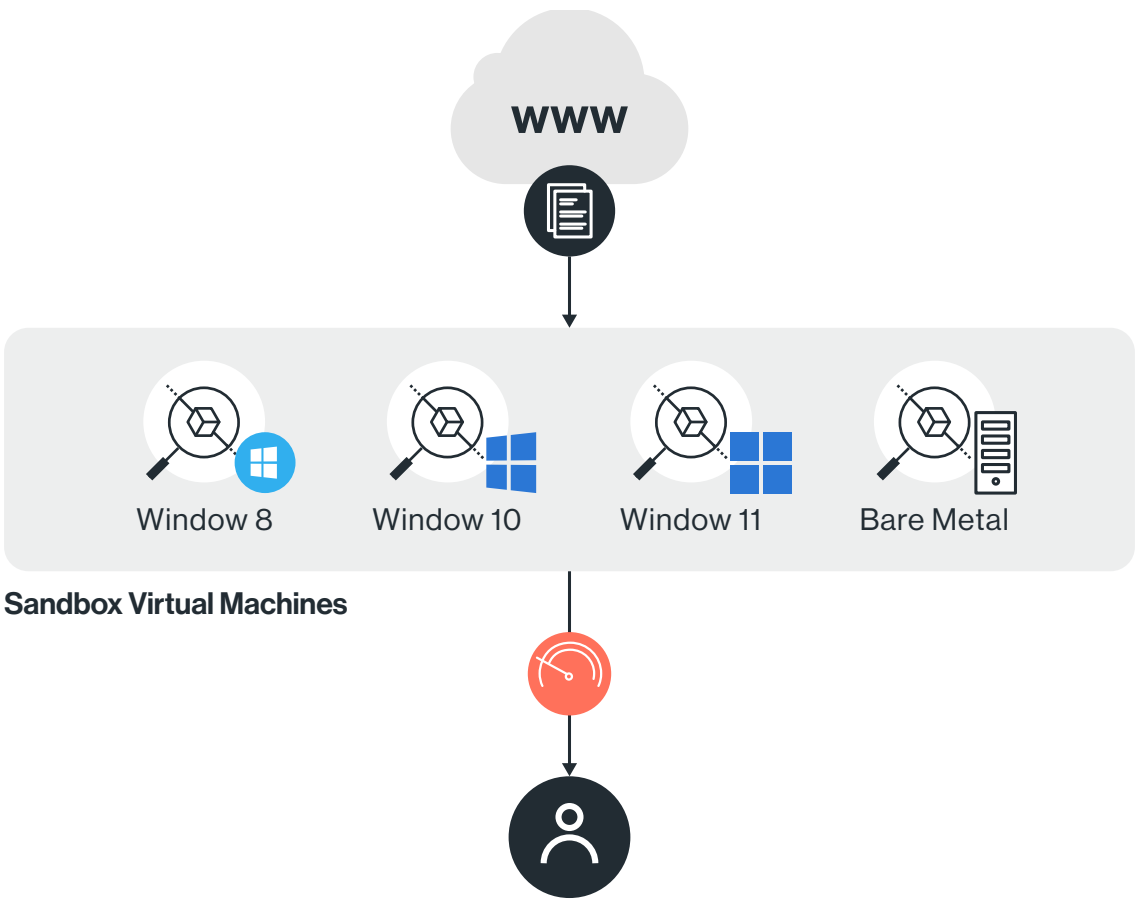
Sandboxing for malware detection is typically accomplished via the utilization of virtual machines to detonate supported file types under various conditions. These conditions can include time of day, simulated time-lapse, operating system version, installed patches and programs, etc. Some sandbox implementations may also include bare metal analysis, detonation of files in a racked and stacked hardware environment to detect malware designed to evade virtual machine analysis.



Multiple information security vendors offer standalone or integrated sandboxing products in a variety of form factors including software bundles, virtual machines, on-premise appliances, and cloud sandboxing as a service. Implementations should typically focus on matching the sandbox tests with devices that exist in their environment: simulating appropriate operating system types, applications, and patch levels while limiting cost and complexity. This can be a daunting task, especially when you factor in BYOD, WFH, and shadow IT. Are you confident that you know exactly what software and hardware is in use in your environment? Sandboxing deployments are only effective if they consistently reflect your production environment.

Inline Sandboxing: Operational Flow

The best method for utilizing sandboxing is to have the sandbox inline with quarantine enabled. This means that when a user attempts to download an untrusted file, they will be temporarily blocked while the file is analyzed by the automated sandbox product. This analysis typically takes as long as 5 to 10 minutes. Once completed, the user may be notified via email and will have to manually redownload the file if it is benign.

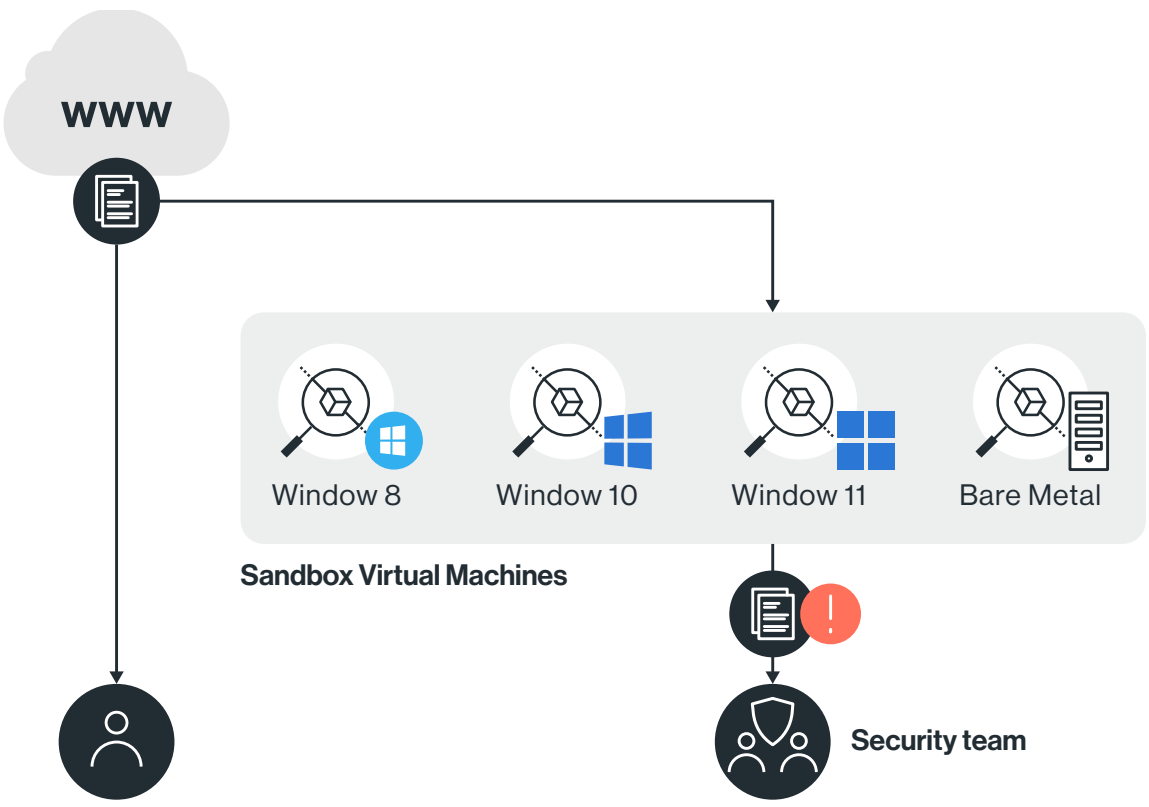


Inline Sandbox with Quarantine

Now, from the IT and security practitioner point of view 5 to 10 minutes is not a very long time to delay an end user from downloading a file, but the user might disagree. Imagine stepping into an important meeting and trying to download a file. Sandboxing will take 5-10 minutes, assuming there is no false positive.

The other option: Allow and Sandbox

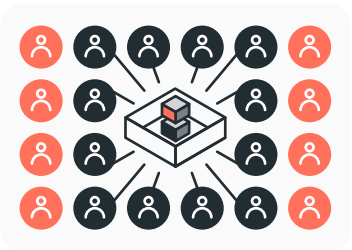
Clearly this comes with its own set of challenges as this method creates the possibility that users will download malicious files and the impact will have to be remediated. Some organizations may not have the resources to remediate quickly in these cases, and with cloud synchronized file storage, malware can spread very quickly. Neither option is ideal, and organizations are forced once again to choose between user experience and practical security.



Inline Sandbox without Quarantine

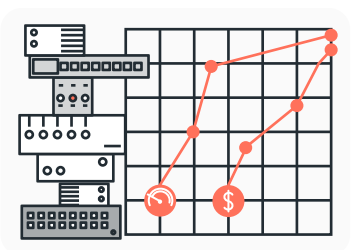
Other Limitations

The previously described dilemma aside, sandboxing is full of other limitations that make it challenging to use for real-world prevention.



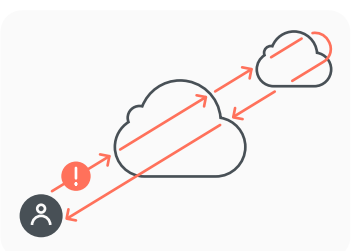
On-premise sandbox deployments

are subject to the same challenge as other physical appliances, which is lack of scalability.



Sandbox Appliances

require a balancing act of cost and effectiveness. Ideally all unknown files should be evaluated for malware, but appliances leave gaps.



Cloud-delivered sandbox services

may rely on third-party providers or non-converged platforms that service chain, sending your files to engines hosted in the public cloud for analysis. Increasing latency and processing time while decreasing overall security effectiveness.

Either way, these factors can reduce the effectiveness of sandboxes for stopping malware, adding to the multitude of variables that are already at play.

The real question, however, is for all the effort required, how effective are sandboxes at stopping malware? The truth is, not very. While utilizing a sandbox is better than having no zero-day protection, sandboxes create complexity and erode the user experience while delivering limited security value. NGAM can't catch malicious URLs inside an executable file either. Beyond this, attackers are often well funded and have access to purchase the same tools being used to protect organizations. Modern malware is designed to be evasive, using constantly improving techniques for obfuscation.

Here are some examples:

User-behavior evasion

This means the malware has checks that are designed to detect the presence of a real user interacting with the system including mouse movements, scrolling, document closing, etc.

VM evasion

VM evasion is used to detect indicators that the malware is running within a virtualized or sandboxed environment, looking for things like bios information, IP address, username, hardware specifications, etc.

Time evasion

Time evasion is exactly how it sounds, techniques that relate to timing. This can include things like detection of time acceleration by the sandbox or triggering malware actions only on a specific date and time in the future.

While sandbox products are also continually improving in response, you should know that more sophisticated sandboxes require more power, hardware, and resources. This increases the cost of your deployment or service while still forcing you to make compromises. Overall, sandboxing isn't an effective means of protecting end users from threats, however, sandboxing does have its place. Sandboxes are an extremely valuable tool for SOCs to detonate and analyze malware as part of your security practice. However, for automated protection of your users and systems there is a better way.

Next-Generation Anti-Malware (NGAM)

With any problem, you should start with a focus on the desired results, not the methods. In the context of this paper, the desired result is to consistently stop zero-day and polymorphic attacks. Hence, when evaluating a security platform, you shouldn't immediately ask for sandboxing. Instead, you should focus on the protection that is provided along with the impact on user-experience, administrative overhead, and cost.

At Cato Networks, we built the world's first Secure Access Service Edge (SASE) platform from the ground up, converging networking, security, and remote access into a cohesive, cloud-native platform. Early on, we determined that automated sandboxing did not meet the needs of modern organizations and opted to build NGAM into our platform instead. The "why" is simple: NGAM is built into our Cato Single Pass Cloud Engines (SPACE). Thus, allowing us to provide superior zero-day and polymorphic threat protection inline and at wire speed. This means that users can be protected in real-time, and organizations no longer have to compromise security like they do with legacy sandbox products.

How does NGAM work?

First, we should discuss the relationship between Cato's Anti-Malware and NGAM engines. Cato's anti-malware engine runs first and will block known threats using signatures automatically. Cato's signature database is updated hourly to ensure the best protection from known threats for our customers. The NGAM covers the last two layers (Tools & TTPs) and runs on traffic that was allowed by the anti-malware engine, all of this is happening very quickly within Cato's single pass cloud engines with no noticeable delay to the end-user.

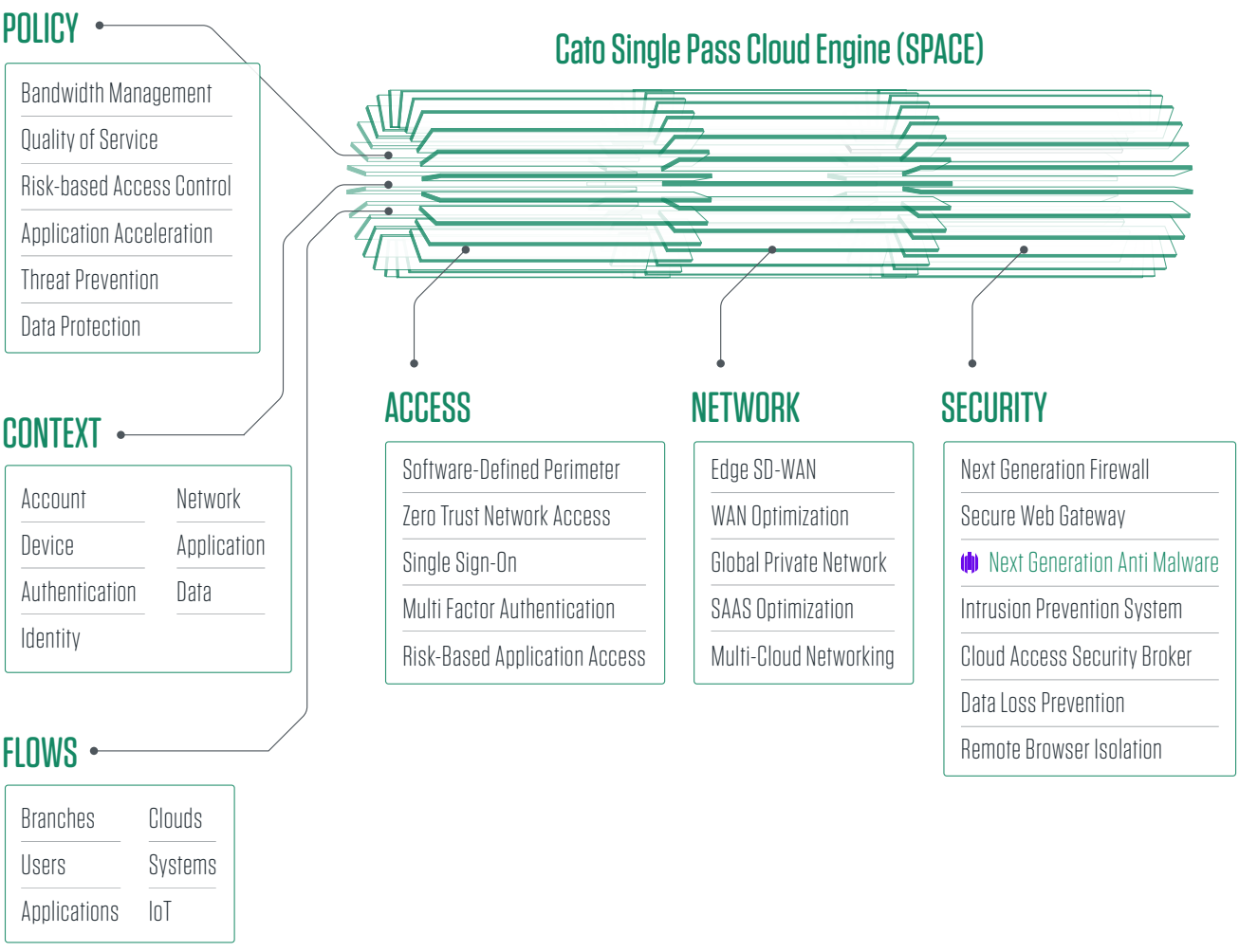
For our NGAM engine we have partnered with SentinelOne to utilize their powerful machine learning capabilities to detect malware based on the structure attributes of the file. Things like true file type, file entropy, Portable Executable (PE) headers, strings, imports, and exports and so on are all possible indicators of a malicious file. Cato's NGAM engine performs static analysis on files in the same manner as a highly trained malware analyst, only in fractions of a second instead of hours or days.

The NGAM engine maps connections and relations between thousands of data points, scores them, and then makes a verdict if a file is malicious or benign.

An advanced mathematical model was trained on millions of good and bad samples for real-time effectiveness in detecting zero-day and polymorphic threats.

New versions of malware are discovered constantly, and the model is continuously trained on an ever-growing data set. One of its efficacy metrics is how often the model needs to be updated, which is currently every 3-6 months. This indicates how powerful this malware identification method is, and that low (or zero) effort is required to benefit from it.

NGAM is just one of many ways Cato is delivering comprehensive protection to customer organizations through our global SASE platform.



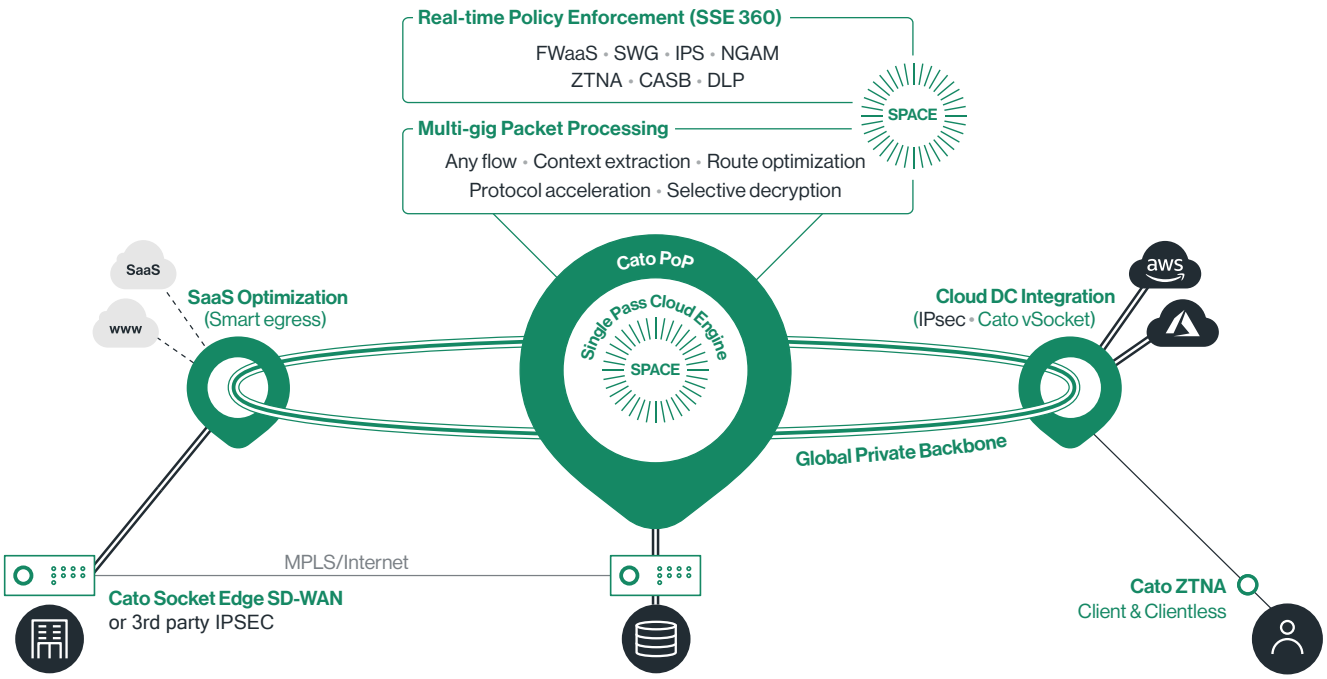
Protection for Today and Tomorrow

The threat landscape is constantly evolving as bad actors discover and exploit new vulnerabilities. Sandboxing was developed as a method to combat zero-day and polymorphic threats, but like most legacy solutions, security is achieved at the expense of user experience. Next-generation anti-malware, as part of a comprehensive global SASE solution, is the best way to provide complete, always up-to-date protection to all users and locations. Because SASE is delivered as a cloud-native service, the platform has the agility to provide complete access and protection now and into the future.

About Cato Networks

Cato provides the world's most robust single-vendor SASE platform, converging Cato SD-WAN and a cloud-native security service edge, Cato SSE 360, into a global cloud service. Cato SASE Cloud optimizes and secures application access for all users and locations everywhere. Using Cato, customers easily replace costly and rigid legacy MPLS with modern network architecture based on SD-WAN, secure and optimize a hybrid workforce working from anywhere, and enable seamless cloud migration.

Cato SASE Cloud with SSE 360



Cato SASE Cloud

- SSE 360
- Secure Remote Access
- Edge SD-WAN
- Global Private Backbone
- Multi-cloud / Hybrid-cloud
- SaaS Optimization
- Cato Management Application

Use Cases

- MPLS Migration to SD-WAN
- Secure Remote Access
- Secure Branch Internet Access
- Optimized Global Connectivity
- Secure Hybrid-cloud and Multi-cloud
- Work From Home

Cato. Ready for Whatever's Next.

SASE, SSE, ZTNA, SD-WAN: Your journey, your way.