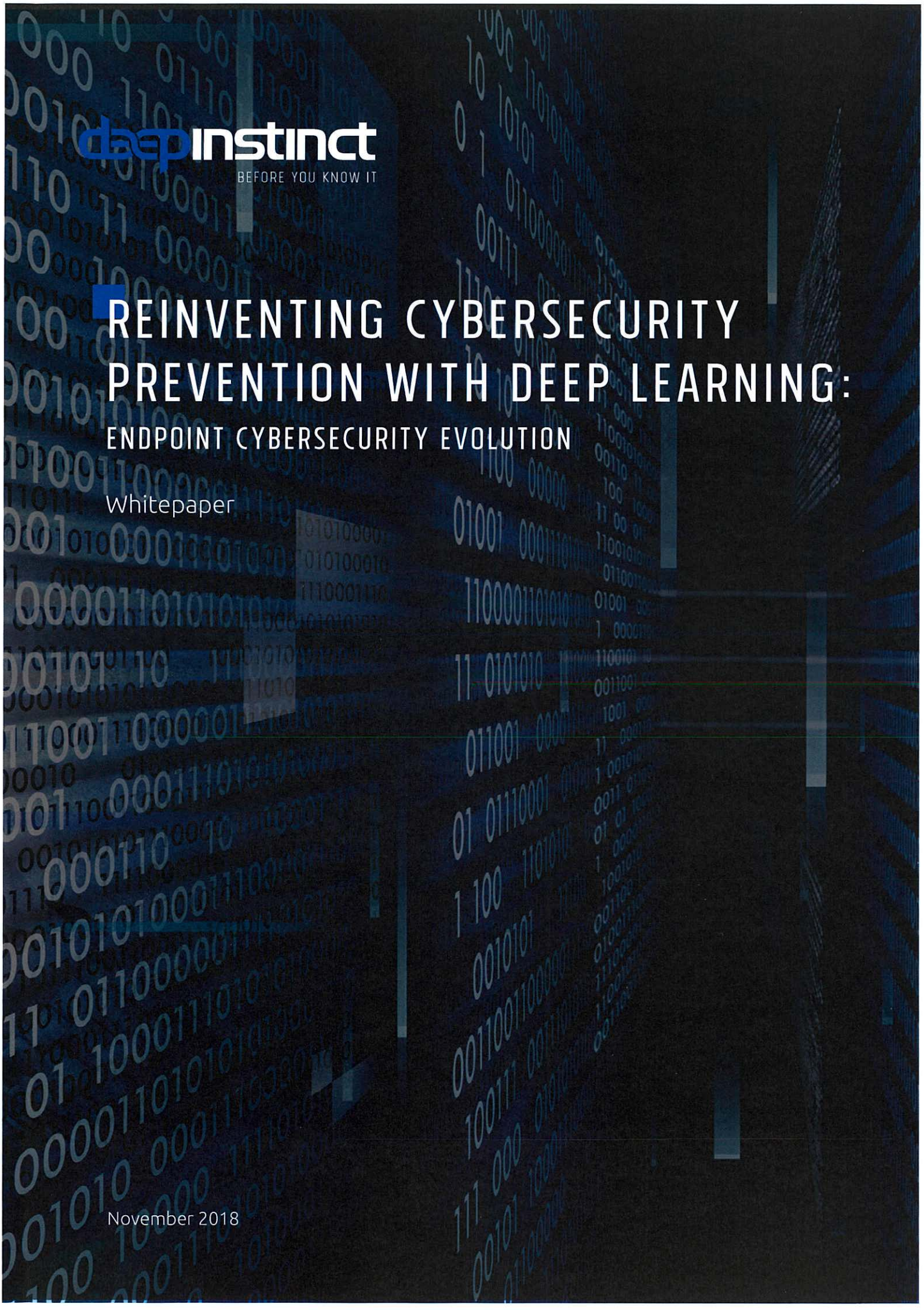


REINVENTING CYBERSECURITY PREVENTION WITH DEEP LEARNING:

ENDPOINT CYBERSECURITY EVOLUTION

Whitepaper

November 2018

TABLE OF CONTENT

INTRODUCTION 3

THE ANTIVIRUS ERA 5

THE SANDBOXING ERA 9

THE BEHAVIORAL ANALYSIS ERA 11

THE MACHINE LEARNING ERA 13

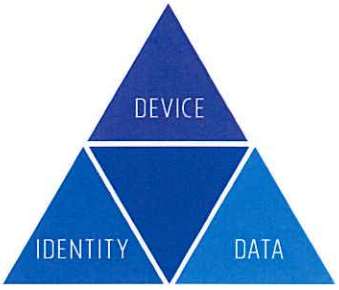
THE DETECTION & RESPONSE ERA (EDR) 15

THE DEEP LEARNING ERA 16

CONCLUSION 17

INTRODUCTION

The cybersecurity world is very diverse and includes many different segments: devices (including endpoints, mobile, IoT, network), identities and different types of data that require protection. In the device area, different types of malware and exploits exist, including ransomware, botnets, viruses, worms, spyware, and more, which exist mainly to harm or to gain money from victims; identities and data include privilege escalation attacks and lateral movement, to steal identities, to spread inside the organization or to steal sensitive information.



Each of the mentioned areas include many different solution types; for example, the endpoint area includes Antivirus, EPP, EDR; the network area includes Network Access Control, Network Firewall, Intrusion Detection/ Prevention Systems, and the identity and data areas include DLP, UEBA, Data Encryption.

Among all the areas, none have had as much diversity in the approaches as the endpoint platform. This is mostly due to the fact that endpoints contain a huge attack surface with multiple attack vectors, eventually leading to a huge cat and mouse chase between defenders and attackers with new attack vectors that make the previous security technologies less efficient.

Over time, the efficacy of protection has dipped quite dramatically, and a few concepts were changed. At the beginning, the common strategy, when the malware was known, was prevention: during the pre-execution stage, don't let the malicious activity run from first place. This approach is by far the most efficient, most secure and has the least impact on local machine resources.

70%

of successful breaches originated from the endenpoint

53%

of organizations have experienced an endpoint compromise within the last two years[1]

[1]source | https://blog.rapid7.com/2016/03/31/idc-says-70-of-successful-breaches-originate-on-the-endpoint/

Over the years, the strategy has shifted into a detection & response approach due to limited defense technologies to assist against the new threats that appeared: instead of understanding the nature of the suspected file pre-execution, it was done during the execution or after it already ran. On one hand this provides better visibility into the real nature of the attribute by looking at its real activity in real-time. But on the other hand, this led to real infections (as it was too late to mitigate the threat), which eventually led to high IT work expenses and damage control; it also required having a highly experienced and skilled team that uses the product on a day-to-day basis. Today, the approach can be shifted back to prevention due to new AI-based technologies that can defeat today's unknown attacks; some of which include machine learning, but mostly deep learning

Other than the prevention and detection era, there were also some eras in which the common technologies changed dramatically. These include the antivirus, sandboxing, ML and EDR eras.

In this whitepaper we will cover:

- The different defense technologies used over time
- How different attacks vectors influenced this evolution
- How the industry adopted different approaches over time
- Why the prevention approach has returned and why it matters

> "So much of the success of EDR-like features and investigation capabilities relies heavily on the skills and experience of the security administrators using the product day-to-day."
>
> Gartner

THE ANTIVIRUS ERA

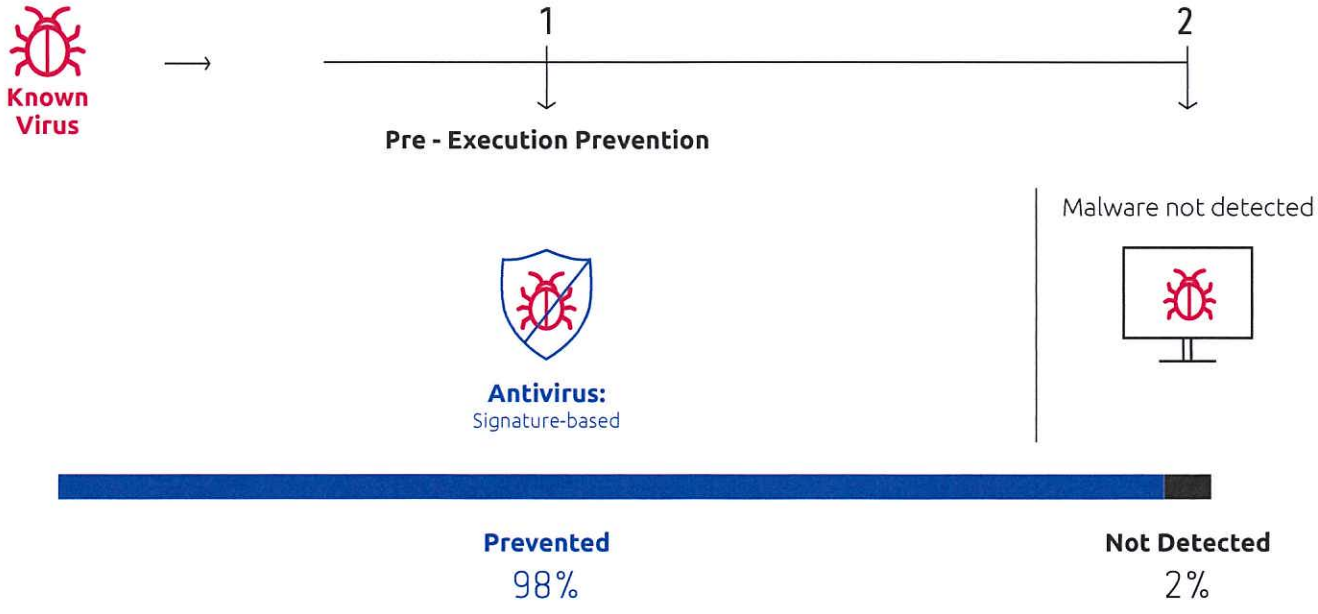
Back to the late 1980s and early 1990s, in the era of known attacks, there was such thing as simple malware. The motivation behind malware authors was quite different than today. Because the concept of threats monetization didn't exist, for the most part, data was not held in internet-accessible systems. The threat landscape was a fraction of a percentage of what it is today as most of them were viruses or worms, and their aim was to infect as many computers as possible.

One of the first Antivirus products ever developed was back in 1985 for the Atari ST system. The company that developed it dismissed the concept of Application Whitelisting (allow only legitimate files to run), and instead, opted for the blacklisting approach (block what is malicious). The reason Application Whitelisting was not considered a realistic approach was because of the many applications that can be run on an Atari system. It is quite funny that today it is still considered as a security technology since there are now thousands of more applications available.

In the early days of Antivirus, the definition file (list of signatures) was composed from a list of hashes (fingerprint) of the malicious files signed by the Antivirus solution. Rapidly, the concept of signatures was changed to a sequence of bytes, so even a small modification at the end of a file that impacted the hash was still matched to the signature. This was a pre-execution approach; if this file tried to execute, it would be prevented from doing so.

The awareness around malware was low. No evasion techniques were implemented into the malware, and so simple signatures that sign the file or the malicious code were enough to catch them all.

ANTIVIRUS ERA | 1985



Signatures → Polymorphism

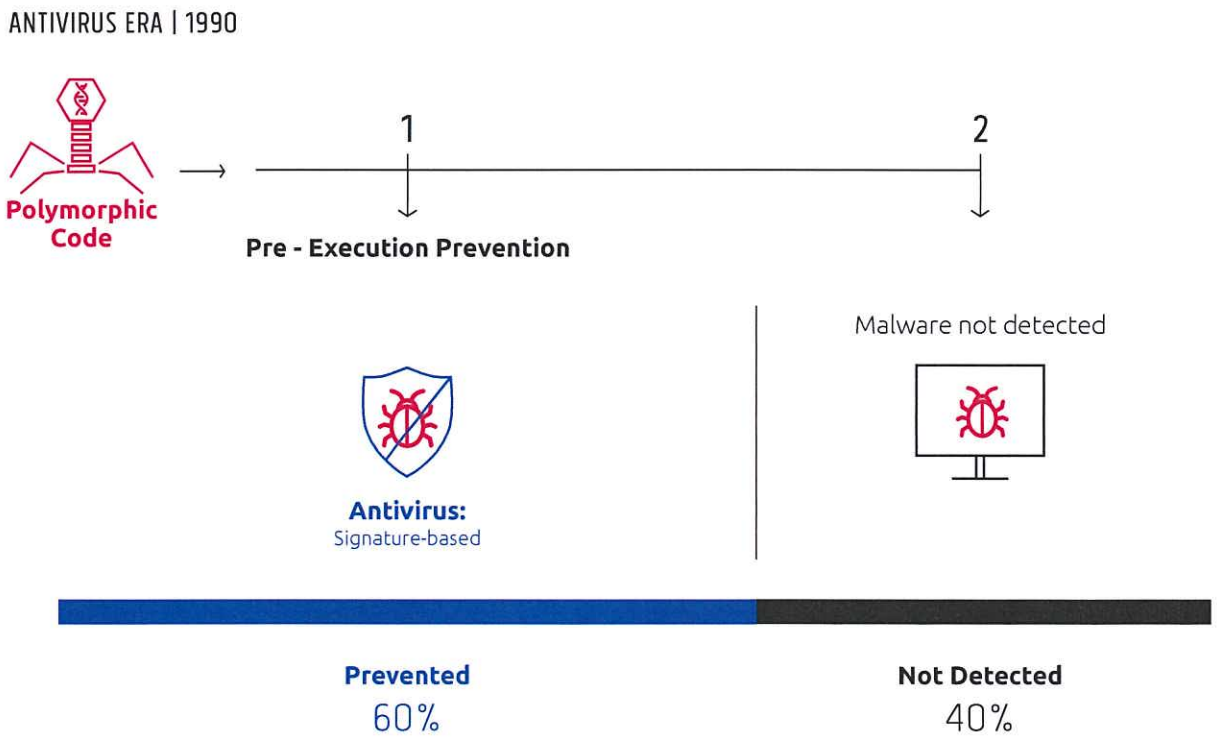
Once new Antivirus technologies started stopping malware families, malware authors were forced to think of new methods to bypass signatures. That's when the first evasion technique emerged, and the concept of unknown malware begun: polymorphism, known as malware variants, or malware mutations. In this technique, the same code is reused for each malware variant, but with very slight modifications. This was the first real challenge for Antivirus providers – they had issues in detecting those variants, and every time a new variant appeared, they had to push a new update to the definition file. In order to save that time, there was a necessary need to find a new method to mitigate malware.

For example, one variant can have the following assembly instruction:

```
add $1, %eax
```

While another variant can use the following instruction, which will act identically:

```
inc eax
```



Polymorphism → Static Heuristics

At this point, static heuristics became the main method. Instead of signing the whole bunch of files or the malicious code, random parts of the signature were inserted into the simple signature, to catch all possible variants that lean on the same original code.

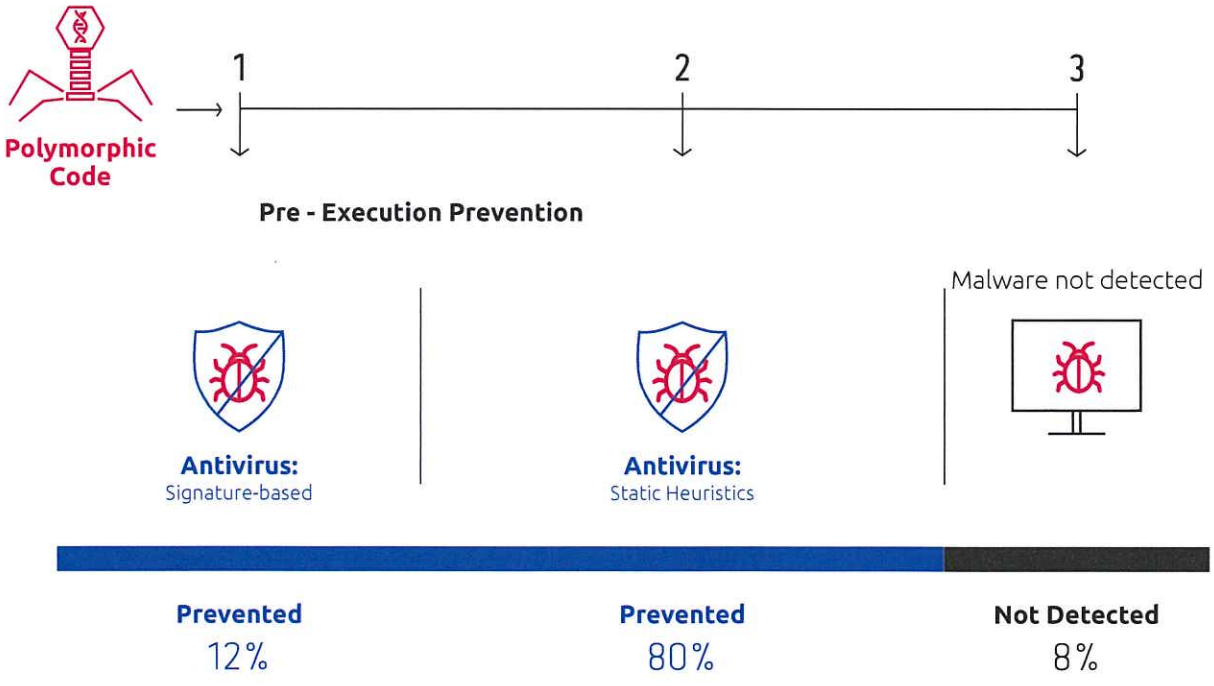
This was an extremely primitive form of Artificial Intelligence. The use of this type of logical decision making gave the advantage back to security vendors. With some basic decision making, prior to executing, the threat could still be prevented and neutralized. This was the first salvo from the bad guys that was ultimately fended off by an immature cybersecurity industry. However, since these exchanges between the good and the bad, we haven't seen detection rates reach 100% again.

For example,
malicious code could be
`3A 4E FF 91 A0 01 05`
and so the signature could apply to this byte sequence.

However, malware could also use the sequence
`3A 4E FF 92 E9 01 05`
to perform the same operation.

The static heuristic to catch both two variants could be
`3A 4E FF ?? ?? 01 05`
One heuristic to detect two variants.

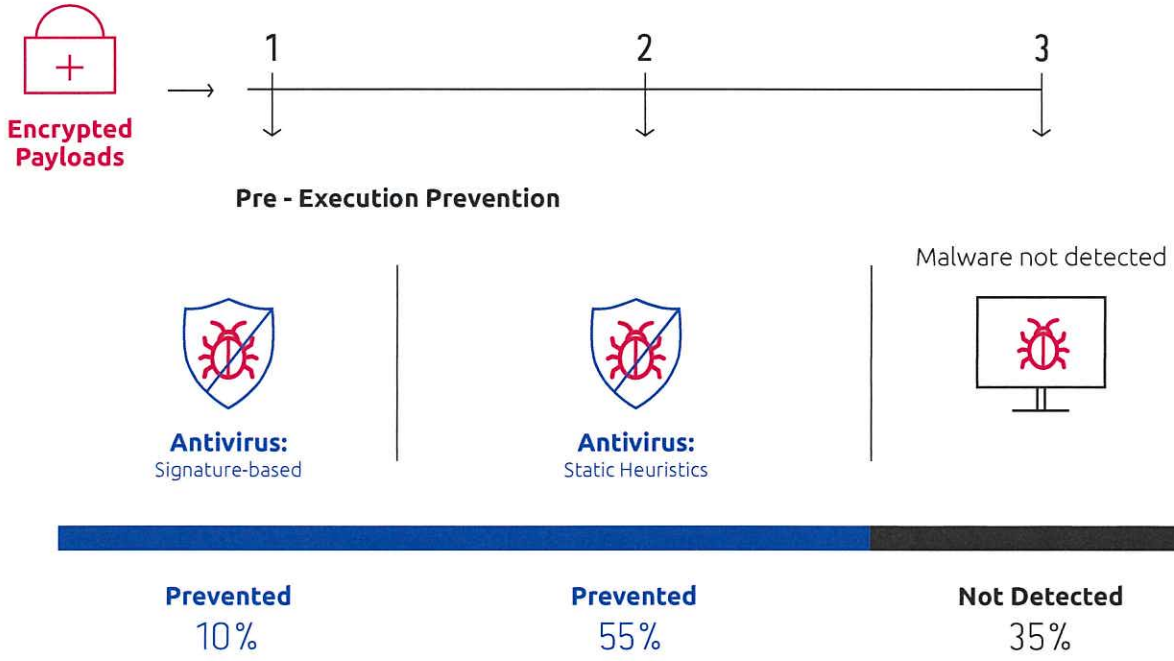
ANTIVIRUS ERA | 1992



Static Heuristics → Encrypted Payloads

The cat and mouse chase continued, and attackers searched for a new technique to defeat the Antivirus again. This time the method used was encrypted payloads. By encrypting the malicious part, or even the whole file, even the static heuristic method which was most common at the time, couldn't detect new variants. This is because each time a new encryption key was used, the whole bunch of data was completely different with no similarities to the previous variants. So instead of manually encrypting the payloads, the attackers started to use tools that do it automatically. Such tools include packers and crypters (UPX, Armadillo, PECompact), shellcodes obfuscators and injectors (MSFVenom, Shellter) and other FUD tools (custom-made crypters).

ANTIVIRUS ERA | 2000



THE SANDBOXING ERA

Encrypted Payloads → Sandboxing

In order to deal with the ever-increasing sophistication of threats, new protection layers were needed. The perception that preventative technologies could not cope with these threats was certainly not unfounded, and so various post-execution techniques were developed and used to cope with multiple security layers. The concept behind this was that if static analysis on the original file cannot find the malicious pattern, then the solution could be executing the file in a close container and matching behavioral heuristics on the process' behavior.

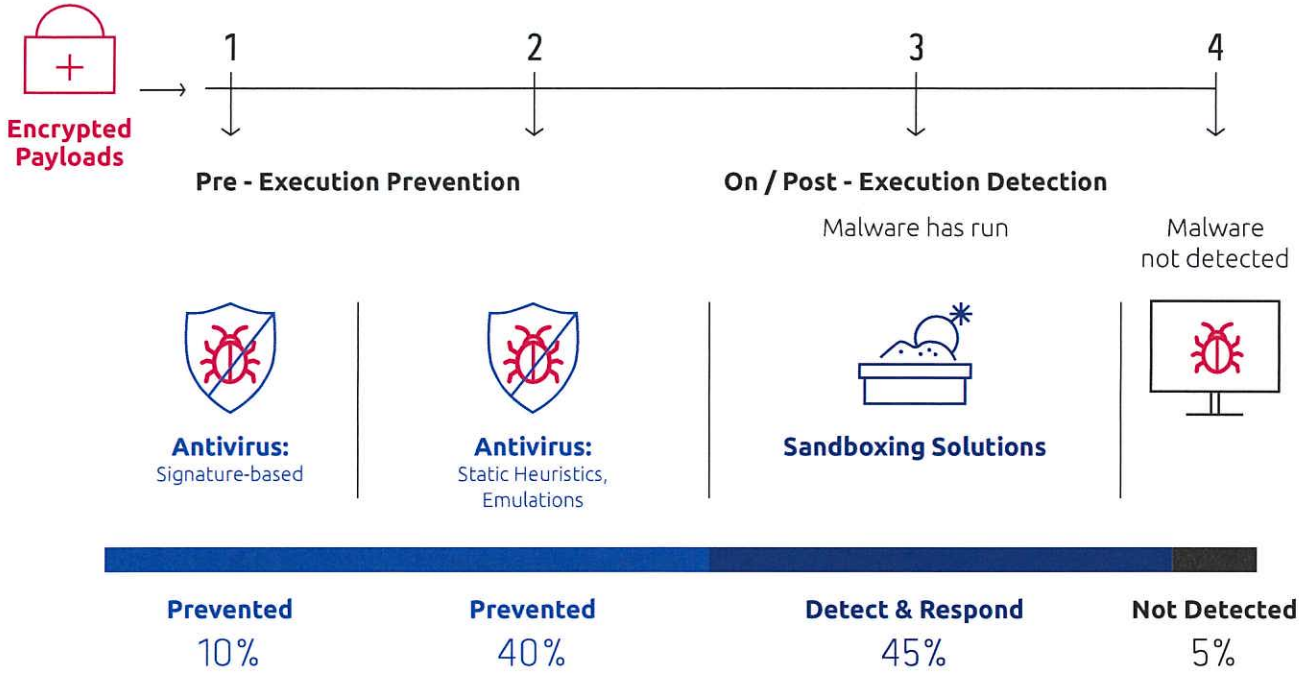
At this point in time, there was the greatest divergence of approaches by the industry. With the addition of techniques such as emulation, sandboxing, and cloud-based file reputation services, which tried to fill the void left by ineffective prevention capability, there was suddenly a real differentiation between solutions.

This is the first time a solution came out of the endpoint perimeter using an agent-less approach. Sandboxing solutions were widely deployed within the network whether as a network perimeter solution online-mode as a gateway or outline by spanning files into it using various protocols and integrations.

"Detection" was suddenly deemed good enough, and customers had to choose a security philosophy to go by – prevention or detection, automatic protection or creating a delay in work. Customers also considered combining the two approaches – "the more layers the merrier", in which many customers have decided to implement many solutions to their stack. The downside of this is a negative impact on performance and management overhead.

This era was the most important in the history of endpoint security. It not only experienced the shift of approaches by vendors incapable of innovation in the pre-execution/preventative discipline, but also an explosion of post-execution and detection-based technologies that in parallel saw the previously mentioned budget split between prevention and detection tip the other way too.

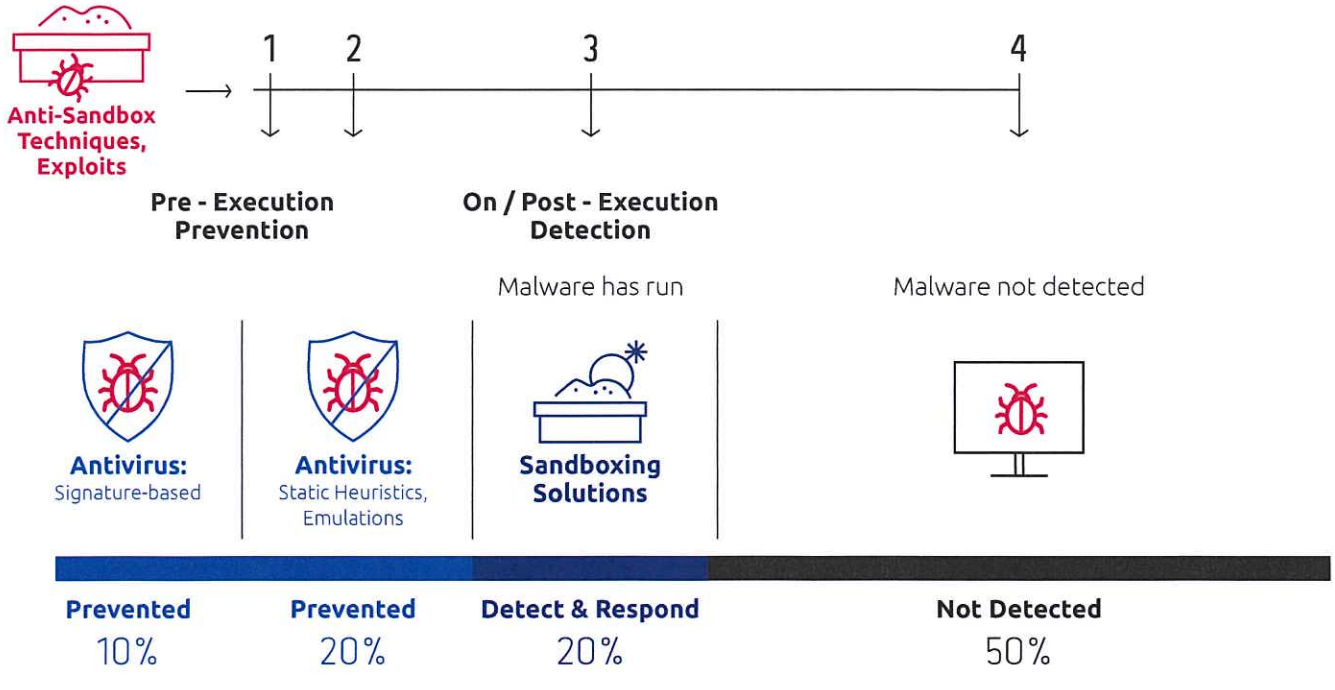
SANDBOXING ERA | 2008



Sandboxing → Anti-sandboxing, Exploits

With the new emulation and sandboxing techniques, attackers had to find new techniques to defeat them. Many anti-sandboxing capabilities emerged rapidly and were inserted into malware, which included time-delayed execution, and detections that the malware is running inside a container. Other attack vectors that sandboxing solutions were not aiming to target, such as client-side exploits of legitimate software installed on computers, grew rapidly. And so the new threat landscape completely bypassed the new promising layer in the market.

SANDBOXING ERA | 2010



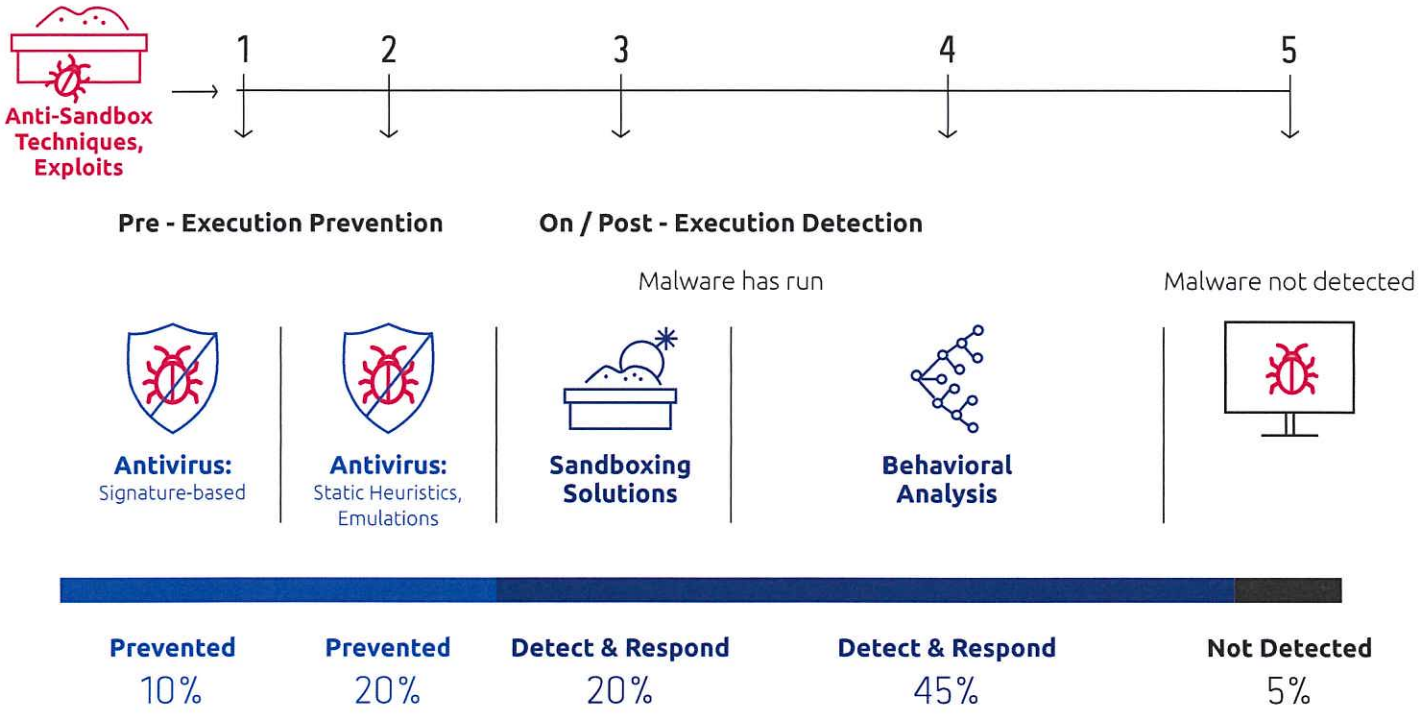
THE BEHAVIORAL ANALYSIS ERA

Anti-sandboxing →Behavioral Analysis

Within a few years, the agent-less approach has become the past. The behavioral analysis capabilities that were done inside the sandbox or emulator, were quickly deployed inside the endpoint security solution itself. And so anti-sandboxing techniques were not valid anymore and known attack vectors could be easily detected and mitigated – usually by killing the process. Though sometime this wasn't good enough, as the malicious activity could already be performed before the mitigation; ransomware could encrypt the computer, or spyware could leak all the important data.

As part of the behavioral analysis heuristics, one stood out – anti-exploitations techniques. These were looking for the exploitation stage (buffer overflow, heap spray for example) before the shellcode itself (the actual malicious business logic – ransomware, spyware) was executed. One of the major problems with this layer was the difficulty of managing it along with the very high false positive rates it provided. However, it did a good job protecting from this attack surface.

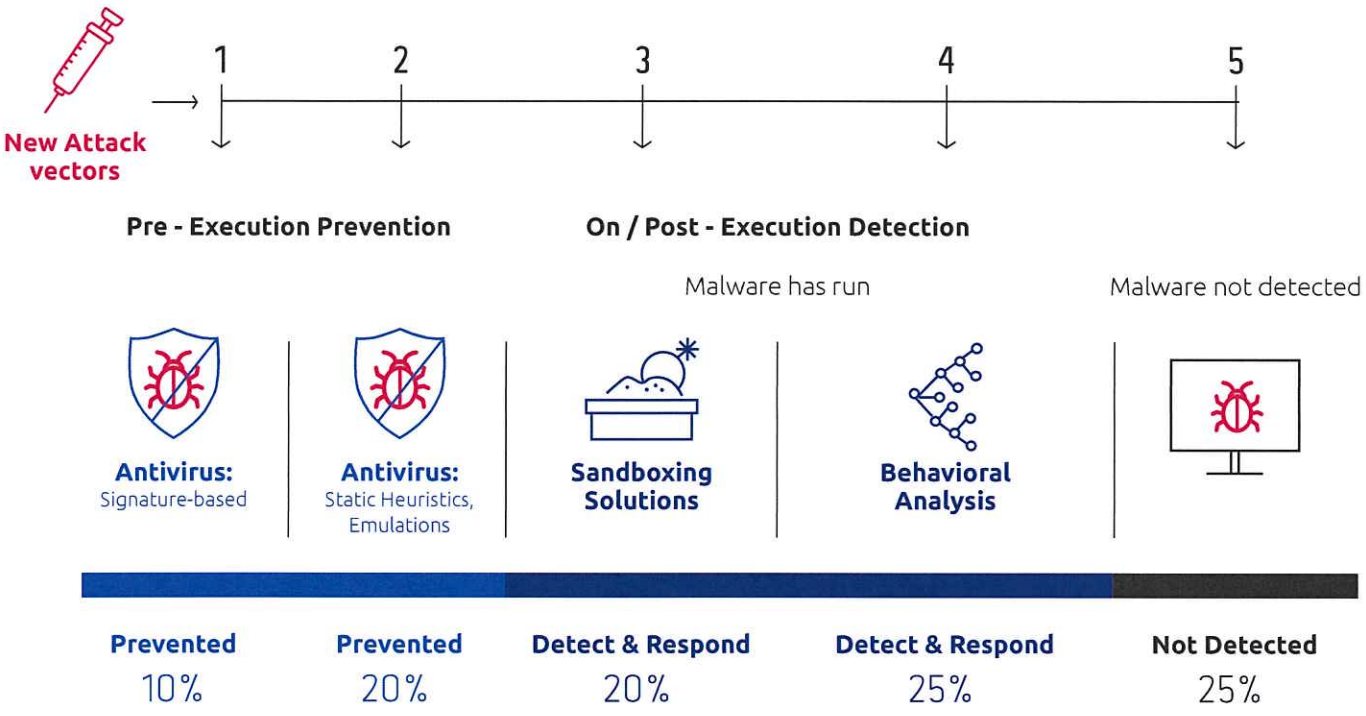
BEHAVIORAL ANALYSIS ERA | 2010



Behavioral Analysis →New Attack Implementations

As the cat & mouse game continued, new attack vectors based on the ones that were already mitigated by the behavioral analysis layer, were implemented by attackers. New information gathering techniques were invented, new keylogging techniques were published, and all the heuristics developed up until then were useless.

BEHAVIORAL ANALYSIS ERA | 2010



THE MACHINE LEARNING ERA

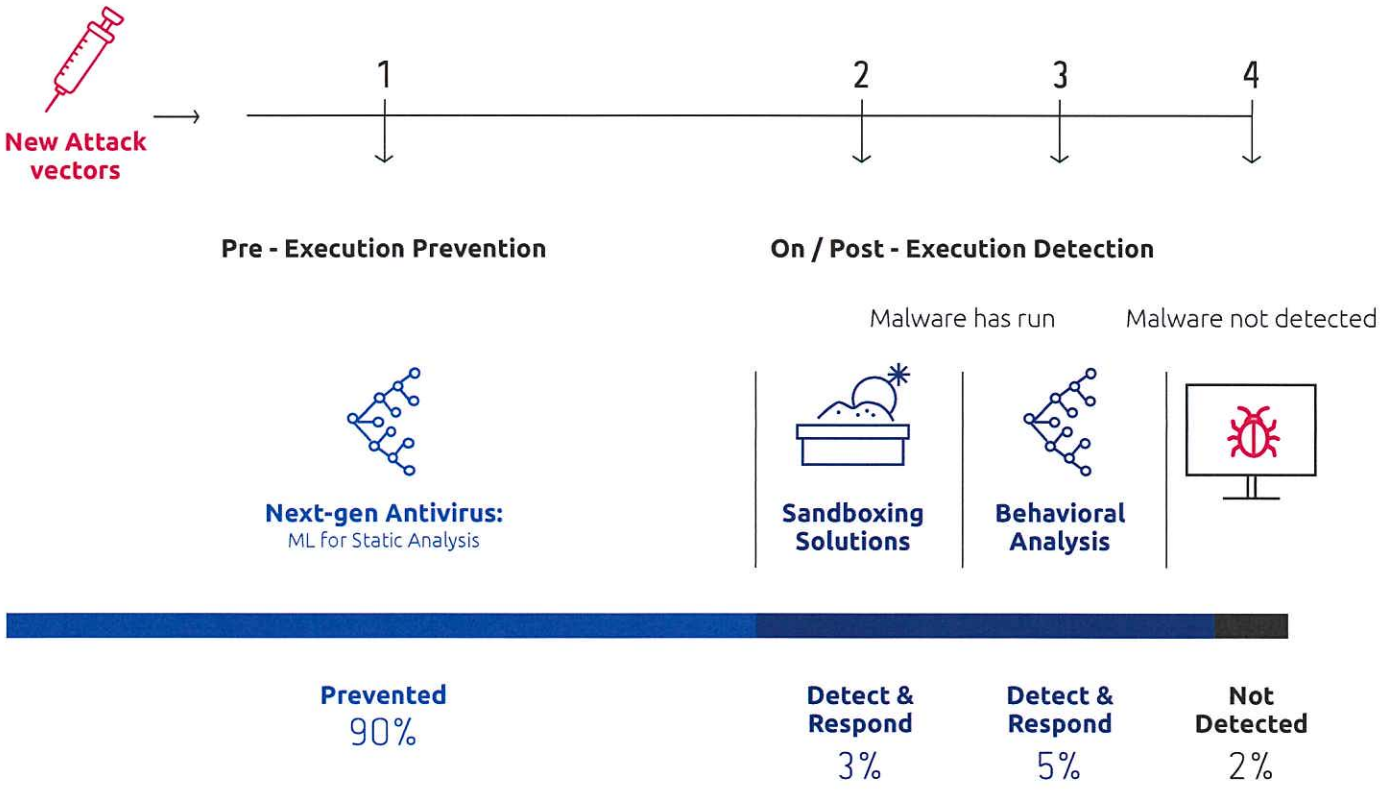
New Attack Implementations → Machine Learning for Static Analysis

If we were to engage with an engineering analyst who isn't involved in cybersecurity, and present him with the challenges the industry faces, many of them would tell us that prevention is the one aspect that has had the least to no real innovation for several years. This is when the machine learning era emerged.

Artificial intelligence, depending on the definition, was already used in the 1980s in cybersecurity in primitive forms. For example, for signature creation, by using static heuristics to move away from 1-1 to a 1-many signature. But the real innovation in cybersecurity artificial intelligence came with the adoption of machine learning algorithms. By wiping the slate clean and starting again, vendors shelfed the signature and the static heuristic methods, and implemented concepts from the academy in the form of machine learning models in order to detect malicious files. This is when prevention returned.

Although using prevention for APT attacks achieved higher results, the implementation was still not good enough, and it suffered from very high false positive rates and limitations when it came to the type of files it supports; as most of the vendors only supported PE files.

MACHINE LEARNING ERA | 2015



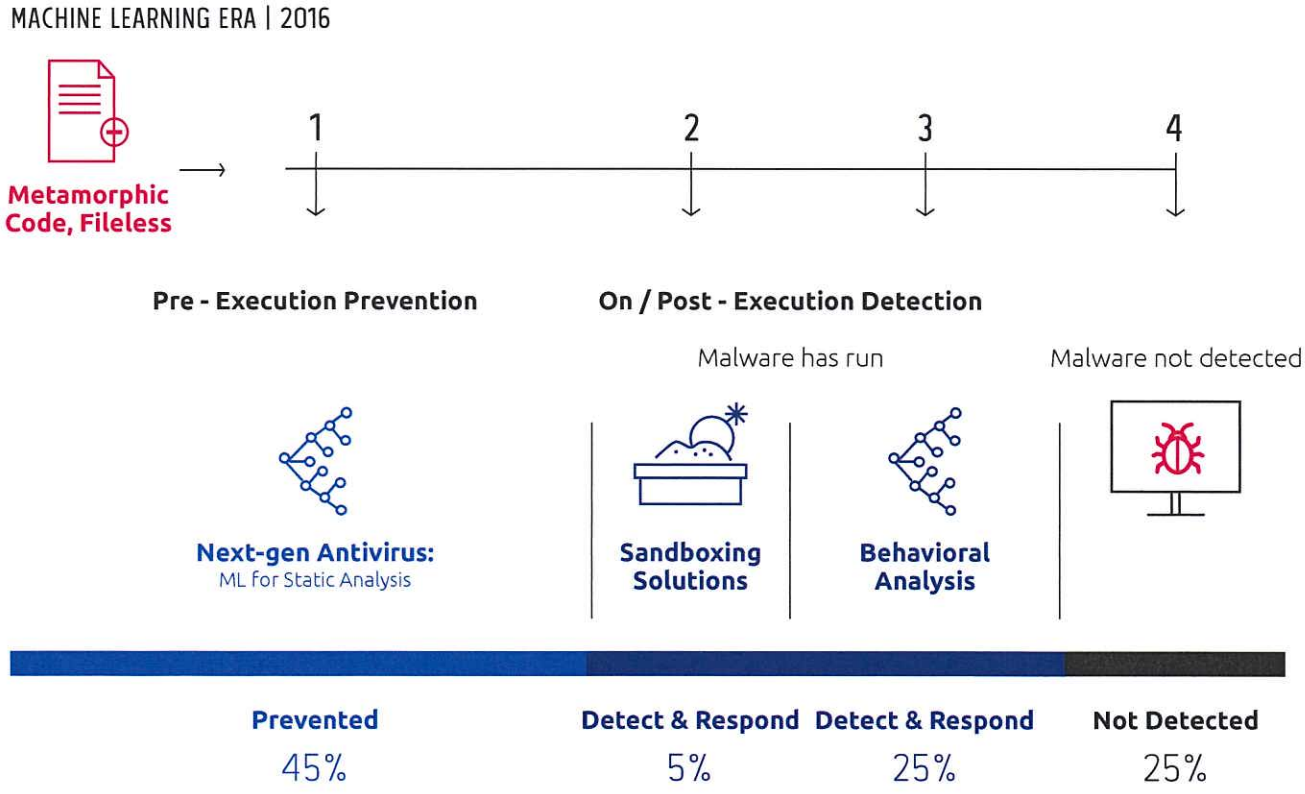
One must assume that it's difficult for cybersecurity professionals to admit there is now an approach or academic discipline from outside the world of computer science that we know little about. The past couple of years have seen the greatest minds in the field of AI and cybersecurity join forces, on some occasions successfully, on others disappointingly. The fact is though, there is a place for AI in cybersecurity, just as it does in every other industry in the world. The question is, how much of it do we use, what form of it do we use, and in combination with what other layers or technologies should we use it for?

Unfortunately, we are still in the early stages of the adoption of AI in cybersecurity solutions. While some vendors implemented machine learning capabilities during their early development stages, others added them at a later stage just to join the AI party.

Machine Learning for Static Analysis → Metamorphism, Fileless

As always, it didn't take long, and new attack vectors emerged to bypass machine learning-based solutions. The metamorphism concept, in which benign content ("junk") is added into malicious files, began to be used. This "junk" generated a wrong prediction result of the statistical, linear machine learning model by misleading it to think the malicious file is actually benign.

In addition, Fileless attack vectors, including scripts and code injection techniques, were also in the spotlight. As long as machine learning focused on files, attack vectors that don't focus on files can evade from that prevention layer.



THE DETECTION & RESPONSE ERA (EDR)

Metamorphism, Fileless → EDR

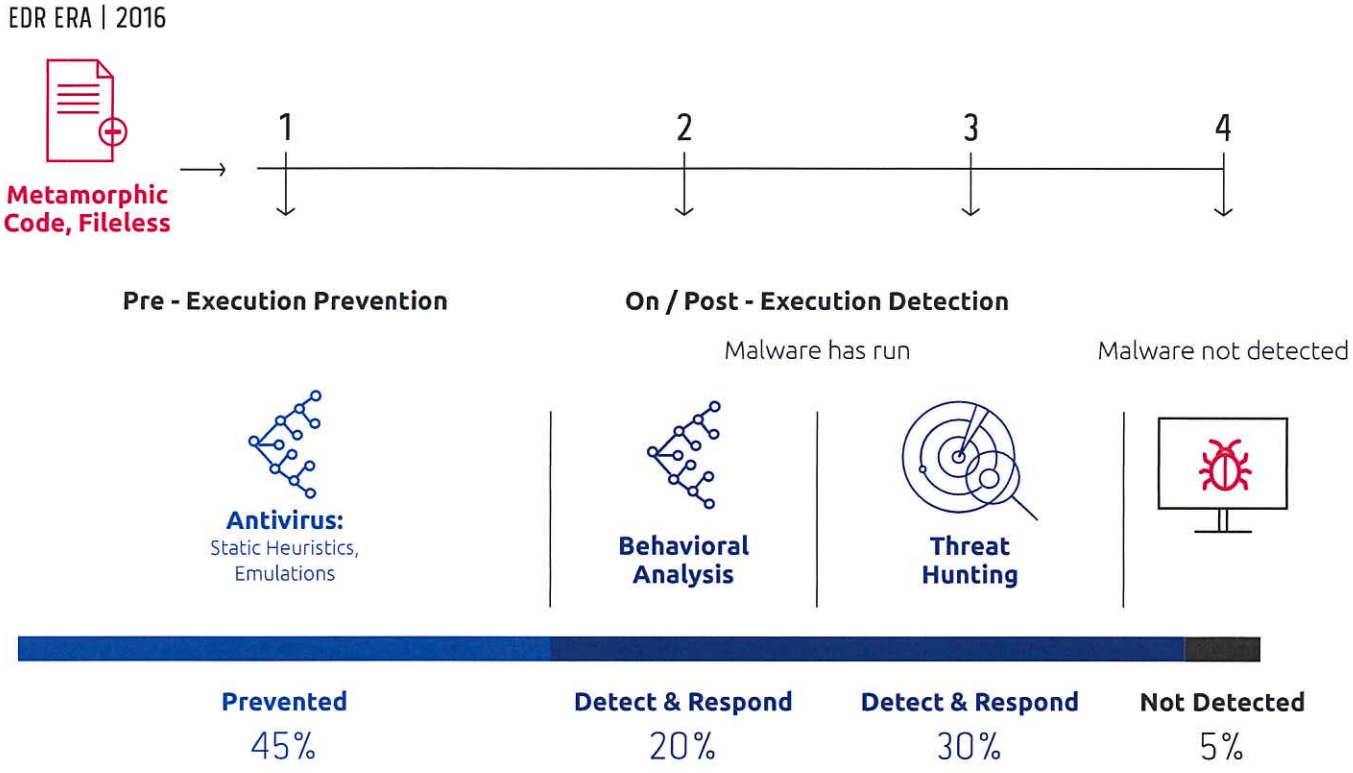
In parallel to the Machine Learning Era which brought the prevention approach back to the game, the EDR Era started, and claimed the detection approach has not yet been exhausted. The EDR Era was introduced by vendors that stated that prevention has failed.

This time it was applied by using threat hunting operations. Instead of giving the responsibility to statistical models or signatures created by the vendors, all the data needed to detect an attack was recorded and was presented to the customer for him to hunt after threats running in his organization.

THE HIDDEN COST OF EDR SOLUTIONS

- **Security Event Analysis:** hunting for security events is complex and requires an experienced team or managed services, leading to increased costs
- **Endpoint System Performance:** monitoring the endpoint continuously leads to performance bottlenecks, while collecting unnecessary data that can strain the endpoint
- **Network Bandwidth:** the cost of network data usage on-premise or to the cloud
- **On-Premises Analysis:** it is costly to host and manage an on-premise big data solution
- **Privacy Concerns:** by collecting and storing most of the system events for detection and response, there's a chance of collecting more information than is necessary or desired
- **IT Operations:** mitigating and remediating breaches that were detected lead to additional costs to operate

Although in theory the concept sounds good, it suffers from many problems. Firstly, it means finding a needle in the haystack, and in large organizations with many endpoints it is even harder. There's a need for ~~a~~dedicated, highly experienced team that understands the material, knows what to search for and specializes in detecting attackers. Another option is to outsource the operation to a managed security service provider (MSP/MSSP), which is usually expensive. Lastly, this still means detection and the remediation could take a long time after the attack has already been spotted manually.



THE DEEP LEARNING ERA

EDR → Deep Learning

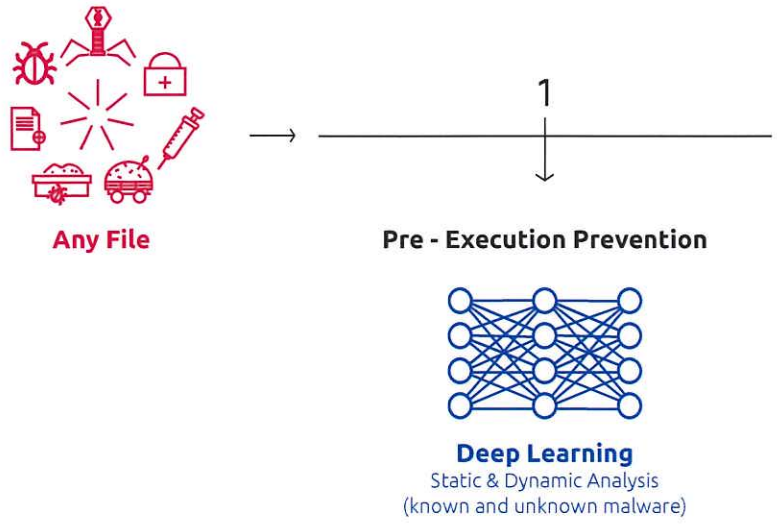
Will AI change the world? Unquestionably yes, it already has, and we have practical evidence of this. Will it change cybersecurity?

The most advanced form of machine learning, deep Learning, can. The revolution started over the 2010s, when deep neural networks were able to be trained by using GPUs. The fields that made the shift from using machine learning to deep learning started achieving much higher accuracy: for example, computer vision improved by 20-30%, and text understanding by 10-20%.

Deep learning also reshaped cybersecurity. By using a fully autonomous training, mistakes that may occur during the training phase by domain experts are mitigated. It also makes the process much faster, and by feeding much more data to the algorithm during the training phase, by processing 100% of the content and by using its non-linear correlation advantages, deep learning provides the benefits of machine learning, and mitigates all its limitations.

The advantages also include much higher detection rates and lower false positive rates, detection of any type of threat that other solutions fail to detect, including APTs, zero-days and any unknown malware. The agnostic approach also allows the support of any file type, including PE, various documents, flash files and even fonts or images.

DEEP LEARNING ERA | 2018



CONCLUSION

At the beginning, when known malware were the common attack vector, prevention based on simple signatures was the best approach. But with more and more sophisticated and unknown malware attacks, other approaches like detection started to emerge.

But detection has its own limitations, and today with new technologies based on AI, the prevention approach is back and stronger than ever. Analysts can officially predict the return of the natural order of prevention and shift the budget from detection-based solutions to prevention-based solutions due to the efficacy of protection. As long as the cat and mouse chase between the attackers and defenders continues to grow, the problem of unknown malware is here to stay, and prevention is the way to combat that.

In reality, there is a time and place for detection, response, forensics, and all other fashionable flavors of security approaches, but prevention is still a must. If you've been bombarded by every buzzword the industry has to offer, remember the good old days when prevention was king, and be aware that when looking at 'next-generation' security solutions, there are some that may surprise, and may yet justify the decision made back in the days of developing an AV solution for the Atari. Prevention will always be better than a cure.

MUST HAVE REQUIREMENTS TO ACHIEVE REAL-TIME UNKNOWN MALWARE PREVENTION

As cyber-attacks become more and more sophisticated, and the problem of unknown malware keep increasing, there is a growing need for real-time prevention. What is needed to achieve this?

- Prevention rates must be greater than 99% for unknown malware and false positive rates must be lower than 0.001%
- Pre-execution prevention must happen within milliseconds (as opposed to waiting for the attack to occur and then reacting)
- It must be applicable to any type of file or fileless attack, and not just portable executable files and able to run on any operating system
- Autonomous, on-device prevention without impacting the device performance
- Ability to perform automated analysis and classification of threats in real-time without needing the security analysis of an expert
- Platform agnostic so that it can be applied everywhere and not just on the endpoint (e.g. Data Center, cloud, network, mobile etc.).

*All the numbers stated in this whitepaper are based on third party analysis, statistics and our estimations.

ABOUT DEEP INSTINCT

Deep Instinct is the first company to apply deep learning to cybersecurity. Deep learning is inspired by the brain's ability to learn. Once a brain learns to identify an object, its identification becomes second nature. Similarly, as Deep Instinct's artificial neural network brain learns to detect any type of cyber threat, its prediction capabilities become instinctive. As a result, any kind of malware, known and new, first-seen malware, zero-days, ransomware and APT attacks are predicted and prevented in real-time with unmatched accuracy.

Using deep learning, Deep Instinct offers a predictive threat prevention platform with multi-layer protection against any known or unknown threat from any file or fileless attack. Deep Instinct protection can be applied on any device (endpoints, mobile devices and servers) with any operating system.

To learn more, visit | www.deepinstinct.com

GET A DEMO

deepinstinct
BEFORE YOU KNOW IT

www.deepinstinct.com